Name: Pourik Saha

College Roll Number: 703

Exam Roll Number: 2021141287

Supervisor: Dr. Biswajit Biswas

Department: Comp. Sc.

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchase exceeds INR 100000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Customer module : In this module we add a function which will take customer details as input and will output a unique customer number. The customer details along with the newly generated customer number is stored in the database for further use._
    Input: Customer details
    Output: Unique customer number
  - Purchase module : In this module we add a function for updating purchases of a particular customer using his/her customer number. At the time of checkout, the bill is shown to the checkout staff who enters customer number and bill amount as input. After entering, the function finds a record with customer number equal to the entered customer number in the database and after finding entry with the same customer number, the database is updated by adding bill amount to customer transaction attribute.
    Input: Purchase Summary
    Output: Updated Database
  - Promotional Offer module : In this module, on the last day of the year(31st December) after working hours at the supermarket, records in the database are sorted with respect to total transaction.
    A. Ten customers with the highest total transaction are selected, details taken output and are sent a message to their phone number about them winning the surprise gifts.
    B. The records with total transaction greater than 1,00,000 are selected and taken output as they had won a 22 carat gold coin. After the declaration of all names who won promotional offers, the total transaction is cleared for all customer numbers in the database for a fresh start in the new year.
    Input: List of customers and their transactions throughout the year
    Output: Names of 10 customers with highest transaction and names of customers with transaction greater than 10 thousand separately.

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

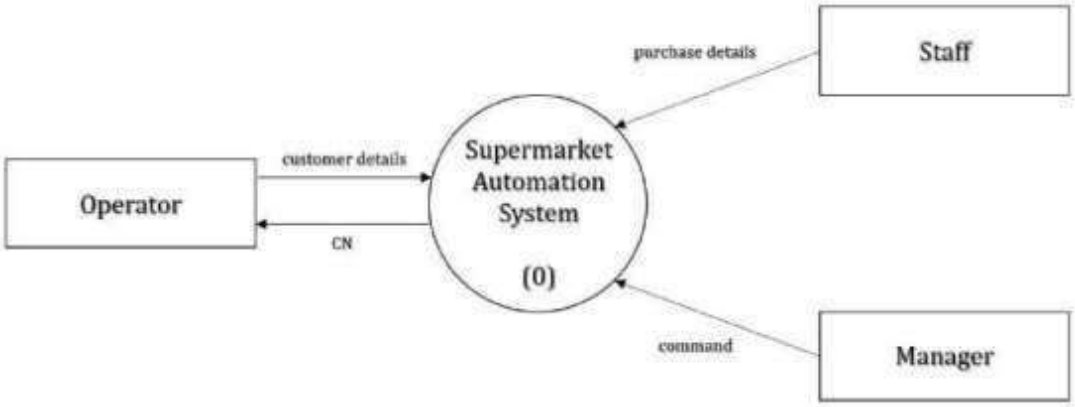
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

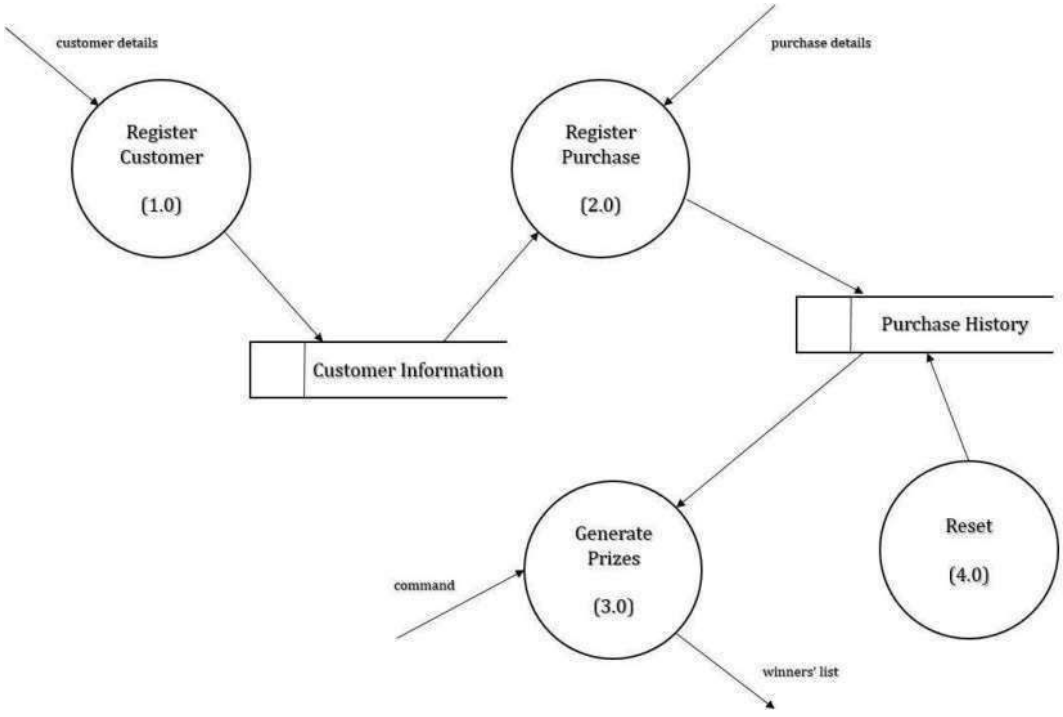
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

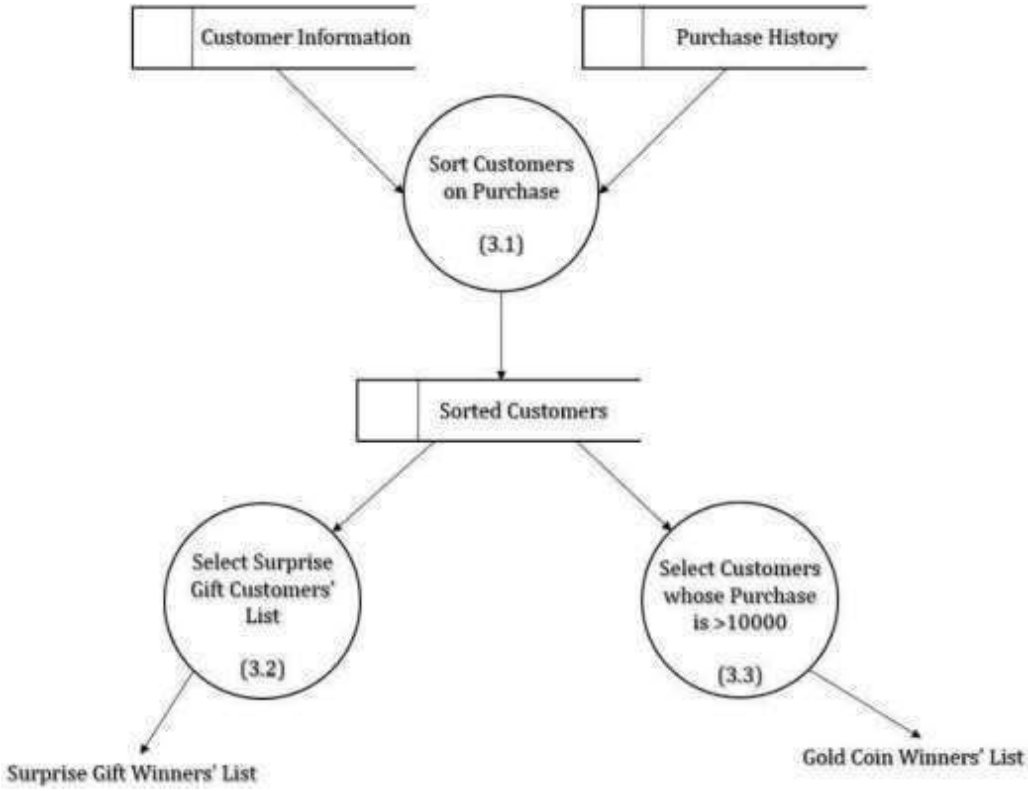
1. Level 0 DFD :



2. Level 1 DFD :



3. Level 2 DFD of process 3.0 :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 2

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

PROBLEM STATEMENT :

It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Incremental Model*** for the following reasons :
    - The fares , roads and buses may change frequently
    - It also requires generating reports after processing o there is a continuous evaluation

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Ticket Booking module : In this module, a customer can book a bus ticket which is mostly preferable to him/her. He can also choose the payment as online or offline (on the counter). The confirmation system will generate the ticket.
    Input: Preferable booking date, time, destination of the customer
    Output: Available bus lists according to the details, Bus Ticket
  - Customer module : In this module, the system will maintain a profile of a customer. At first, customers must register into the system with necessary details. System will maintain a list of previous bookings, current plans.
    Input: Personal details of customer
    Output: An organized profile of the customer, list of previously booked ticket, current bookings
  - Routing Scheduling : In this module, the system will maintain the details of a bus's route. It will handle a database which includes bus number, bus type, route, destination, schedules and driver's name such as if a customer gives his/her travelling details, the system will show the result according to the routing schedule details._
    Input: Details of bus with route and schedules
    Output: Organized database of buses

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

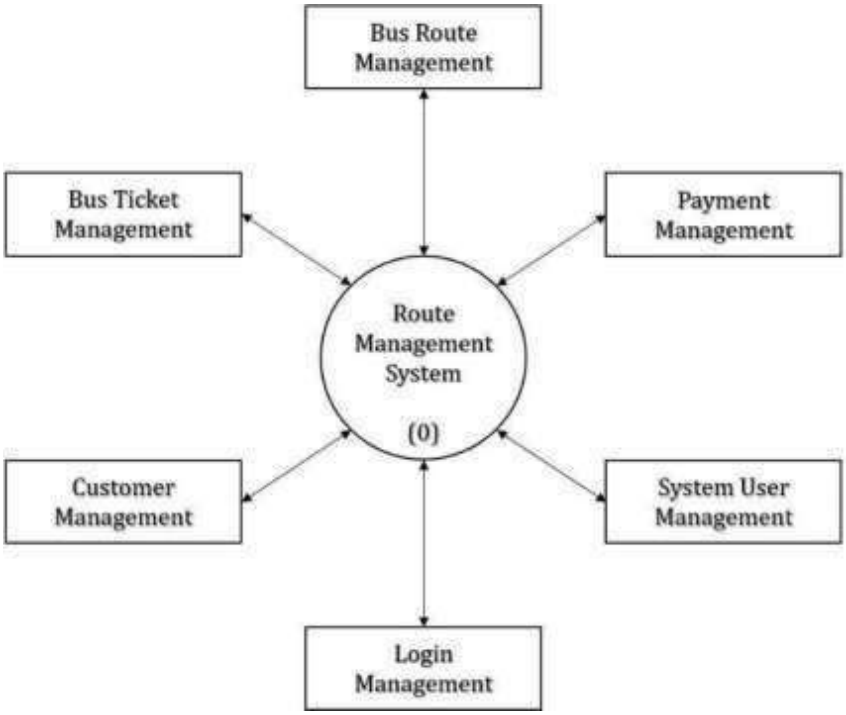
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

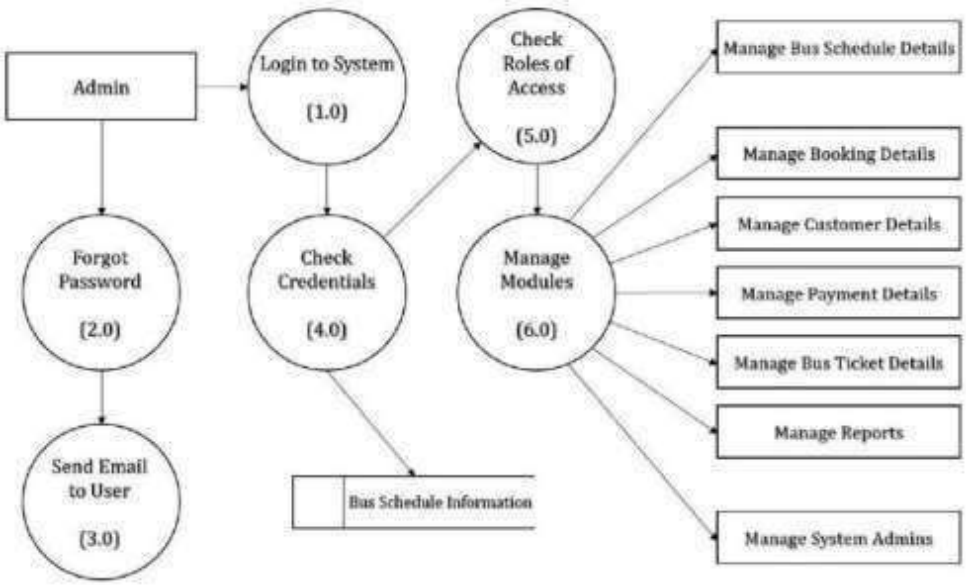
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Incremental Model*** for the following reasons :
    - The number of patients, doctors, diseases and medicines keeps on increasing and updating.
    - The system of the hospital gets updated frequently

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin module : In this module, admin has to give the personal details to register into the system at the very first time. System will create an account and will give the user id and password. System will also set the permission of access. For the next time, the admin can log in into the system and manage other details of a hospital or chamber according to his permission.

    Input: Details of Admin

    Output: User id, Password and Permission of Access details
  - Doctor's module : In this module, the system will maintain the details of the doctor like doctor's schedule, doctor to visit which patient, doctor's fees, replacement of a doctor, availability of a doctor etc.

    Input: Doctor's name

    Output: Doctor's Schedule, Doctor's fees
  - Patient module : In this module, the system will maintain the details of a patient. After admission, the admin continuously updates the details about his credential. At the time of discharge, the system will generate the bill.

    Input: Patient's details, medicine cost, doctor's fees, nursing fees

    Output: Entire Bill to be paid
  - Diagnostic module : In this module, the system will maintain the status of diagnosis of a patient. It will handle which doctor is visiting which patient. Patient is taking which medicines. What is the current health condition of the patient? After every update by the admin, the system will analyze. The system will also suggest medicine by taking information from past diagnostics.

    Input: Diagnostic details, Patient's disease

    Output: Status of the patient

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

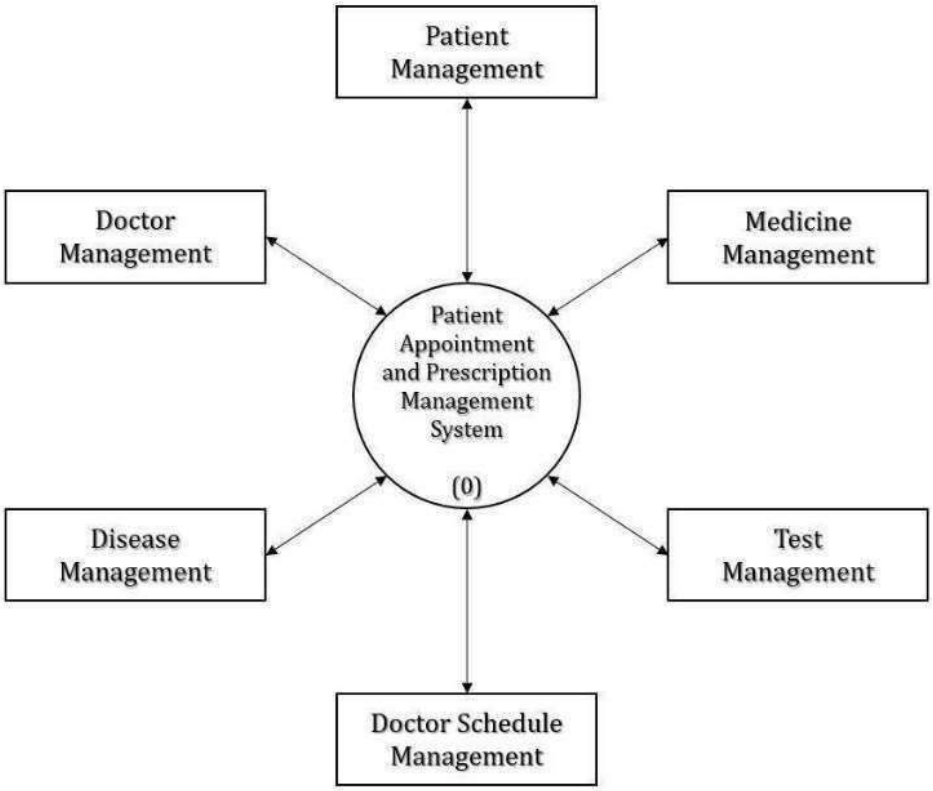
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

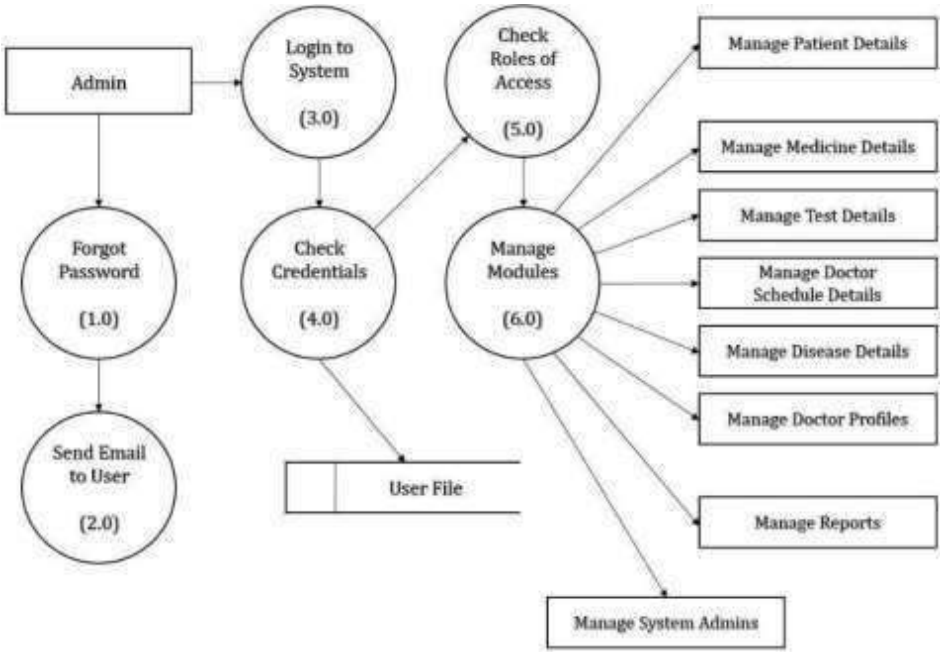
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

HOTEL AUTOMATION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on this spot depending on the availability of rooms. Operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, type room required. The computer should register these data and depending on the availability of suitable room, computer should provisionally allot room number to the guest and assign a unique token number to them. If the guests cannot be accommodating, the computer generates an apology message. The hotel catering services manager would input the quantity & type of food items as and when consumed by the guests along with the token number of the guests with date & time. You can assume that different food items had different code numbers and hence the catering service manager did not enter the full name of the food item. When a customer picks up bags to check out, the hotel automation software should generate the entire bill for the customer's balance amount payable by him.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The requirements can only be fulfilled by a continuous evaluation
    - New rooms, types gets added also more guests arrive time to time

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Guests module : This module takes the guest's personal information and knows whether s/he wants online allocation or spot booking.
    Input: Guest's personal information
    Output: Token Number
  - Staying Allocation module : This module takes the data of the guest about his lodging dates and allocates room (if available) or leaves a apology message (if not available)
    Input: Guest Information of lodging times
    Output: Allocated free room or Apology Message
  - Catering module : This module takes the food order of the guests according to its token number and gives a total food bill along with the grand bill of the hotel.
    Input: Token number, Food Statistics, Food items
    Output: Generate the food bill
  - Bill Generation module : This module generates the whole bill of the guest according to the token number.
    Input: Token number, Food item bills, lodging bill
    Output: The grand bill

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

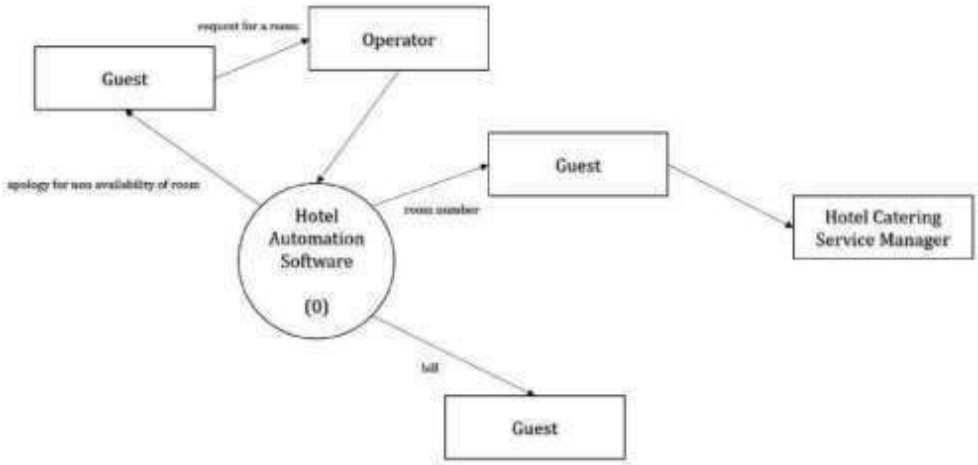
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R)processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

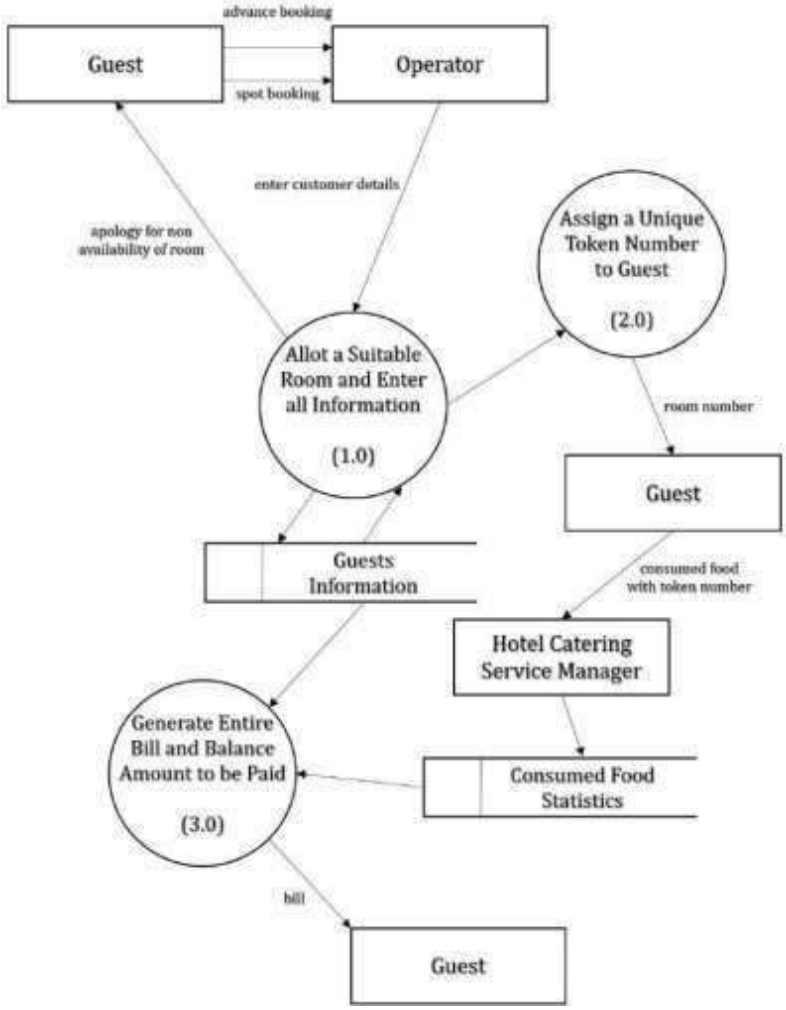
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Crime file management system identifies internal data
stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The rules don't change that frequently

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin module : In this module admin has to register into the system by giving the self information. After the registration, admin can login to the system to handle the crime file details. If the admin forgot the password, then the system will provide a system to change the password.
    Input: Personal Information of Admin to Login
    Output: Access Permission
  - Crime module : In this module admin adds the whole details about the crime of a criminal which includes crime records, complaint records, FIR records, Charge sheet records etc. System will produce organized details about the criminal.
    Input: Individual Crime Details
    Output: Organized Crime Details Report
  - Report module : This module generates the file report for the court according to the codes by using the Crime Details records.
    Input: Crime Details of Criminal
    Output: Court File Report

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

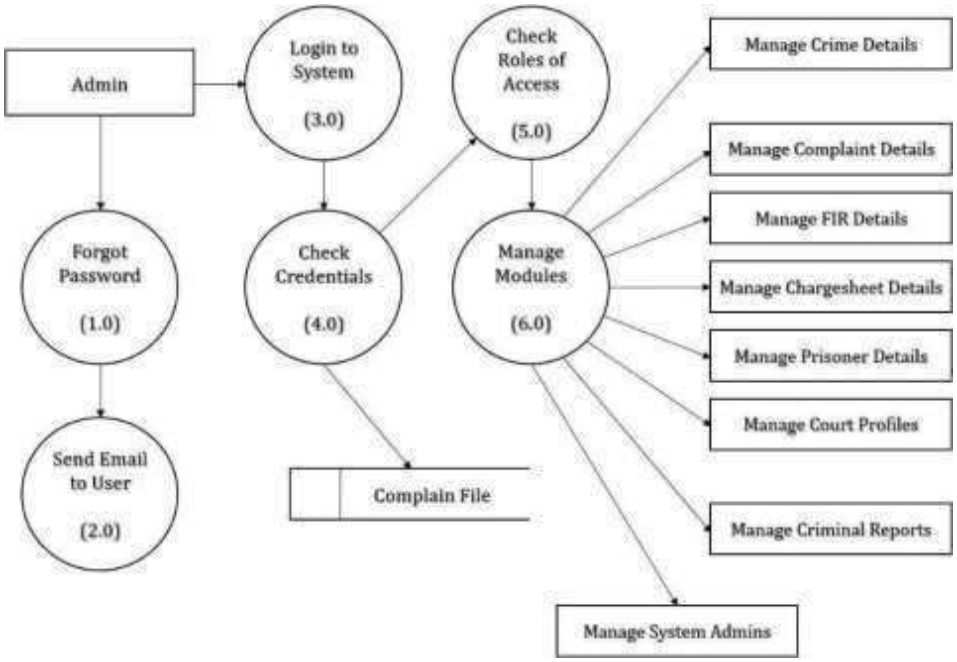
- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :

2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SYSTEM

PROBLEM STATEMENT :

Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are - processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate reports of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The students gets evaluated on yearly basis
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin Registration module : Admin or the controller of examination has to register into the system by giving the personal details with the institutional details. After registration, the system will create a profile and provide a login name and password to update the time table and other details. There must be on option for "Forgot Password" to regenerate the password if the admin forgot the password.
    Input: Details of Admin
    Output: New Profile with login id and password
  - Time Table Management : In this part of the system, Admin creates the time table of a particular class based on the subject, department, availability of faculty etc. Admin has to give detailed information and the system will handle the flow of the time table with classes and departments.
    Input: Details of class, department, branch, available faculty
    Output: Time Table of each class
  - Course and Student Database module : In this module, Admin adds or updates the data of student and course details. Submitting all the details of student and course details by the admin, the system manages the database of student and corresponding course details.
    Input: Details of students and courses
    Output: Database of students, courses, classes
  - Examination Computation module : In this part of the system, admin submits the examination details of the student. After submission, the admin can update the details also. It will compute the examination marks of the students and create the list for each student with their marks.
    Input: Exam details of students
    Output: Marksheet of students

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

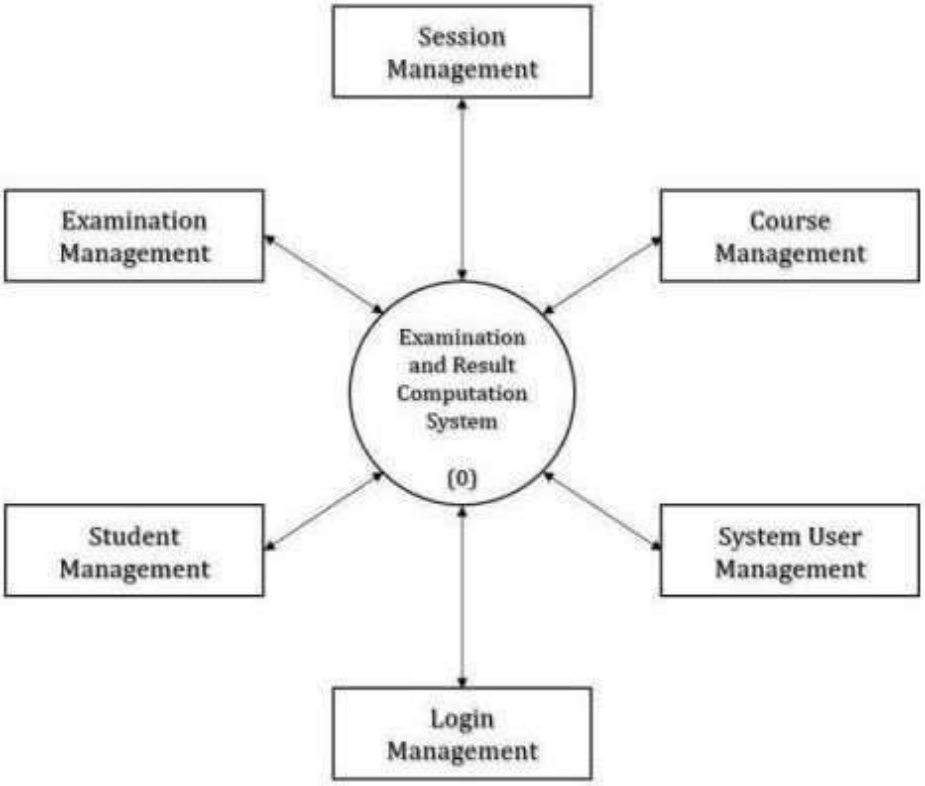
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

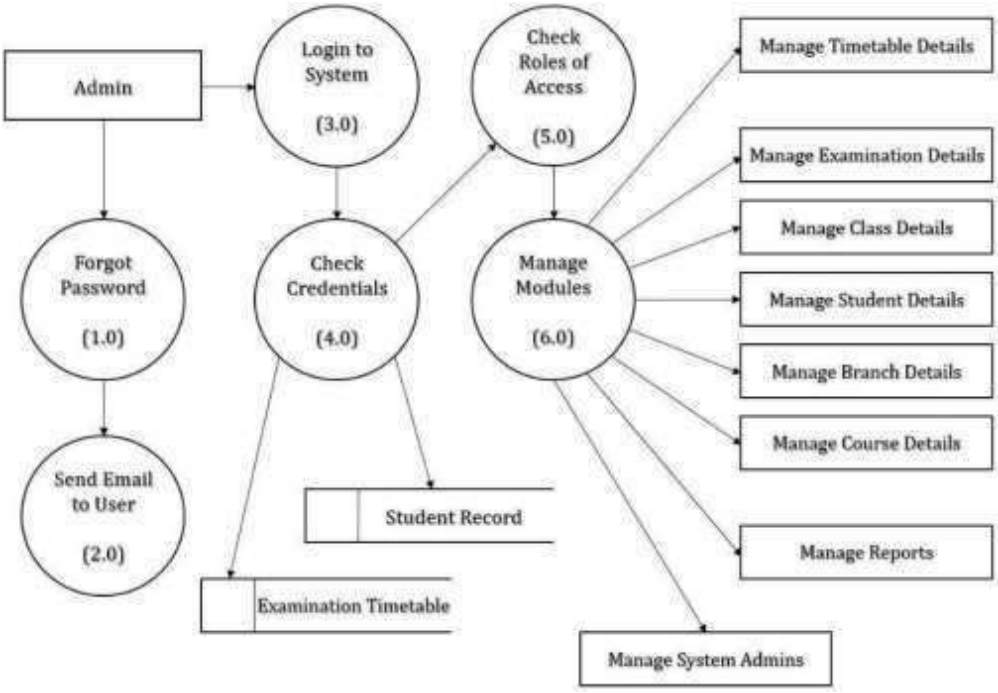
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 7

PARKING ALLOCATION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionality of parking system - processing of parking slots, vehicle records, parking fees, duration and generates the report of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
    - Car Details module : In this module, the user will store the every details of a parked car such as parked car number, car type, lobbies. Other than that, it will also calculate the parking fees according to the total time of parking.
      Input: Details of Car
      Output: Database of the parked car, parking time and fees
    - Bill Generation module : In this module, it generates the bill when a customer leaves the parking slot. It generates the parking fees from the data stored in the database.
      Input: The database of the parked car
      Output: The bill to be paid

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

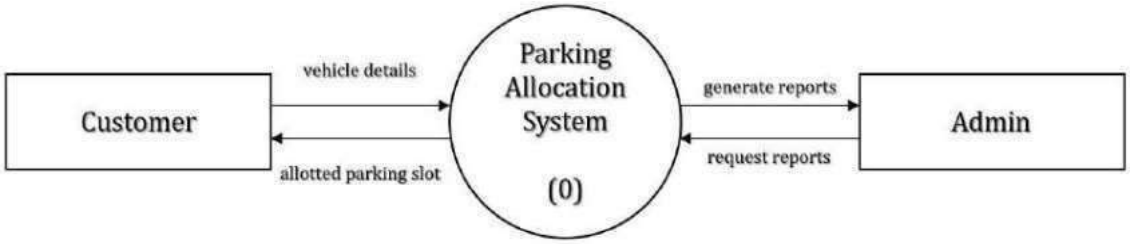
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

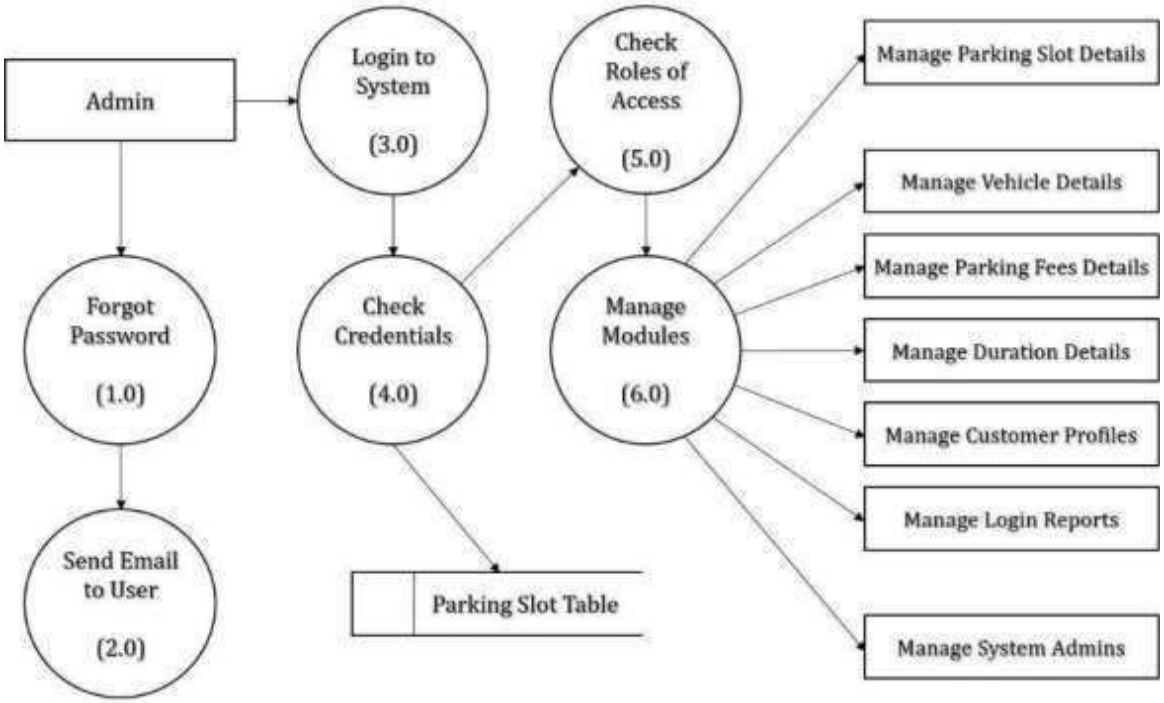
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 8

WHOLESALE MANAGEMENT SYSTEM

PROBLEM STATEMENT :

It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The requirements may increase or change in the future
    - The system needs an continuous evaluation

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
    - Customer module : In this module, a customer is registered and the profile is maintained.
      Input: Customer details
      Output: Unique profile
    - Stock module : In this module, the details of stocks are maintained. A database is maintained for keeping the stock information._
      Input: Stock Details
      Output: Updated Database
    - Bill module : In this module, all the bills are maintained. A database is maintained for keeping the bill information.
      Input: Customer Details, Stock Details
      Output: Generated Bills, Updated Database
    - Report module : In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.
      Input: Store, product, retail, bill, customer details
      Output: Generated Reports

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	14/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

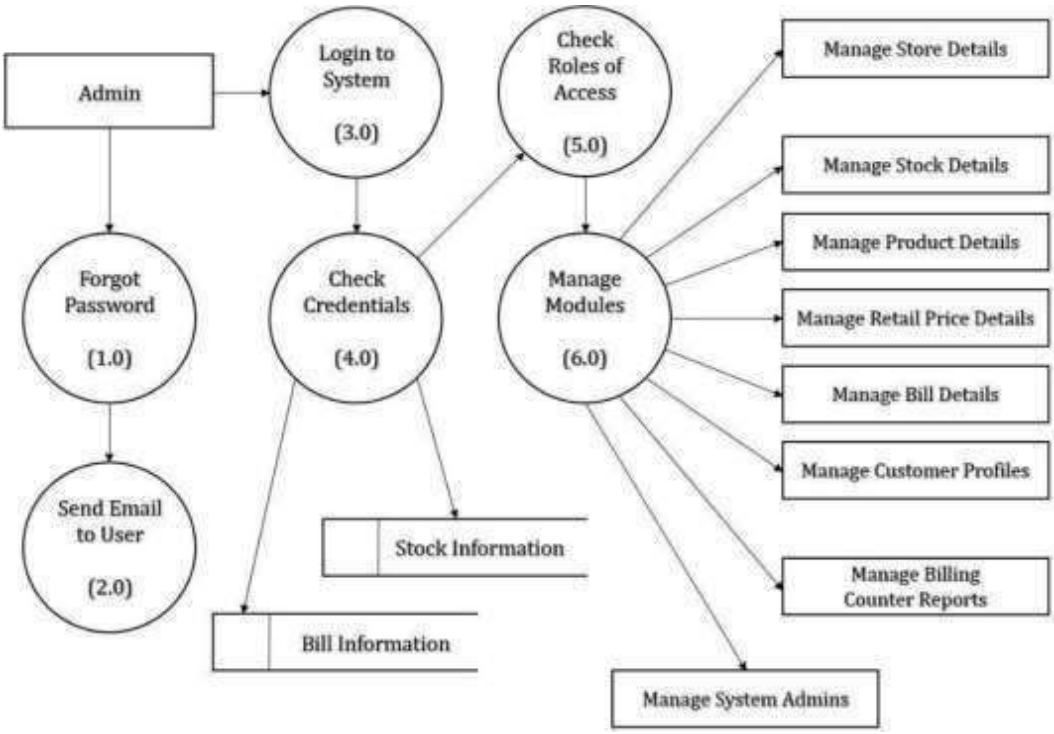
- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :

2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project On Organized Retail Shopping Management Software

Name: Sayantan Bhattacharyya

Roll: 701

Semester: V

Department: Computer Science (B.Sc.)

Supervisor's Name: Dr. Chayan Halder

Project -1

Organized Retail Shopping Management Software

Problem Statement:

A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number, driving license number. Each customer who registers for this scheme is assigned the customer number (CN) by the computer. A customer can present his CN to the checkout staff when he makes his purchase. In this case the value of this purchase is credited again in its CN. At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh /-.The entries against the CN are reset on the last day of each year after the prize winner's lists are generated.

Process Model:

Supermarket management software needs to change the information in a basis of regular interval. Every year the number of customers, the purchase history and other necessary information have to be changed. So the software needs to be added more new functionalities. A supermarket exists in a competitive environment, where it acts as a value-added intermediary between geographically dispersed supplier companies and the scattered individual customers who eventually buy their products. The supermarket management involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This incremental model combines the elements of the waterfall model with the iterative philosophy of prototyping. That is basic requirements are addressed, but many supplementary features (some known, others unknown) remain undelivered. The core product is used by the customer (or undergoes detailed review). As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the

delivery of additional features and functionality. So for a supermarket management software system the **Incremental model** is the most suitable model.

Requirement Analysis:

Basically three modules are used in a Supermarket management system. Consumer module, Purchase module, Promotional Module. On basis of these three modules supermarket management software can be designed.

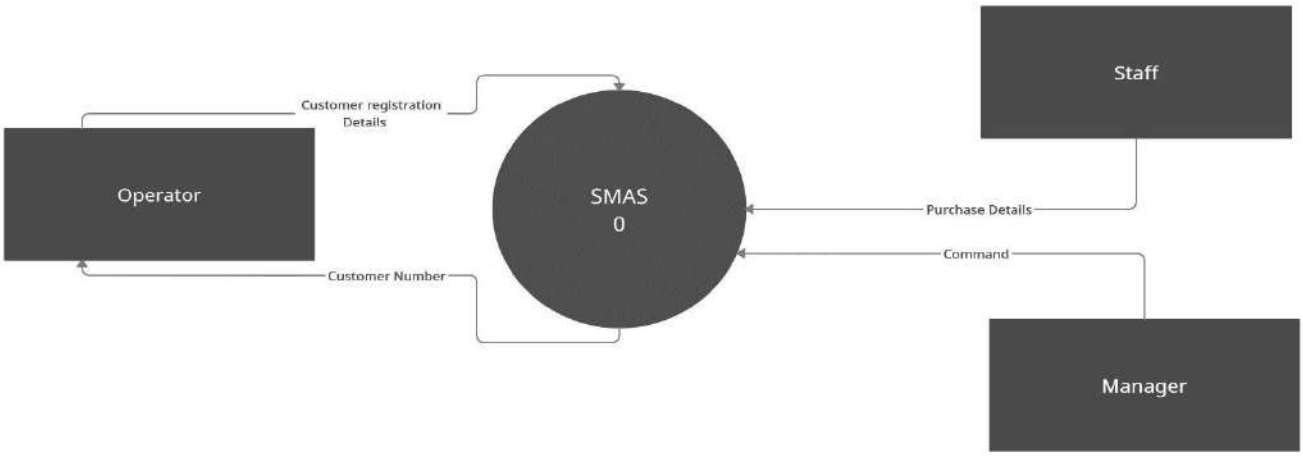
- Consumer Module:
  A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number. When a customer registers his name in the system he will be provided with a unique Customer number (CN).To maintain the record of the customer and their CN a database is maintained.
  Input: Name, residence number, telephone number, driving licenses all other necessaries.
  Output: Customer number (CN).

- Purchase Module:
  When a customer made his purchase his name, bought product, and buying time is recorded. To maintain the purchase history of every customer a separate database is maintained. According to that the amount is deducted from the buyer. **Input:** Customer name, bought product, time
  Output: Purchase history database

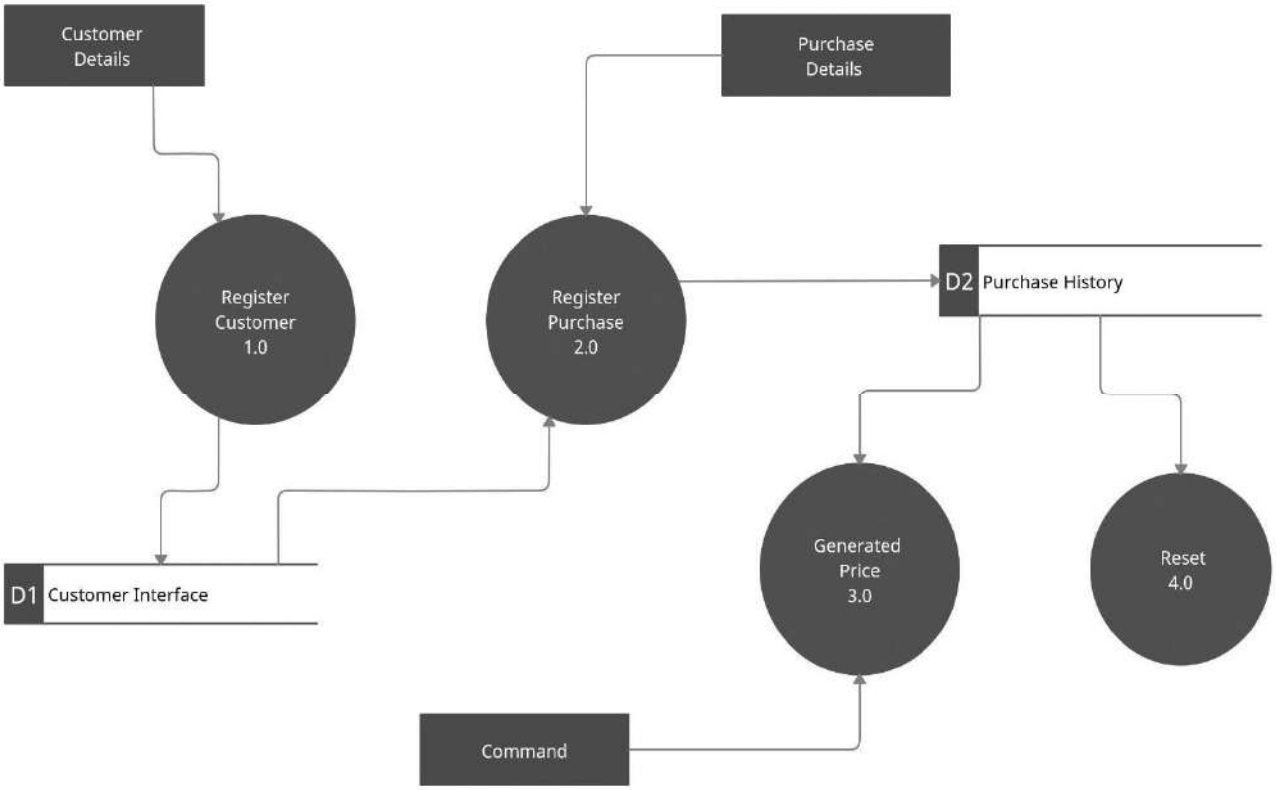
- Promotional module:
  At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh /-.The entries against the CN are reset on the last day of each year after the prize winner's lists are generated. After lists are created the all history is reset.
  Input: Purchase history according to the customer number
  Output: Surprise Gift Winner List, Gold coin Winner List

Design Engineering:

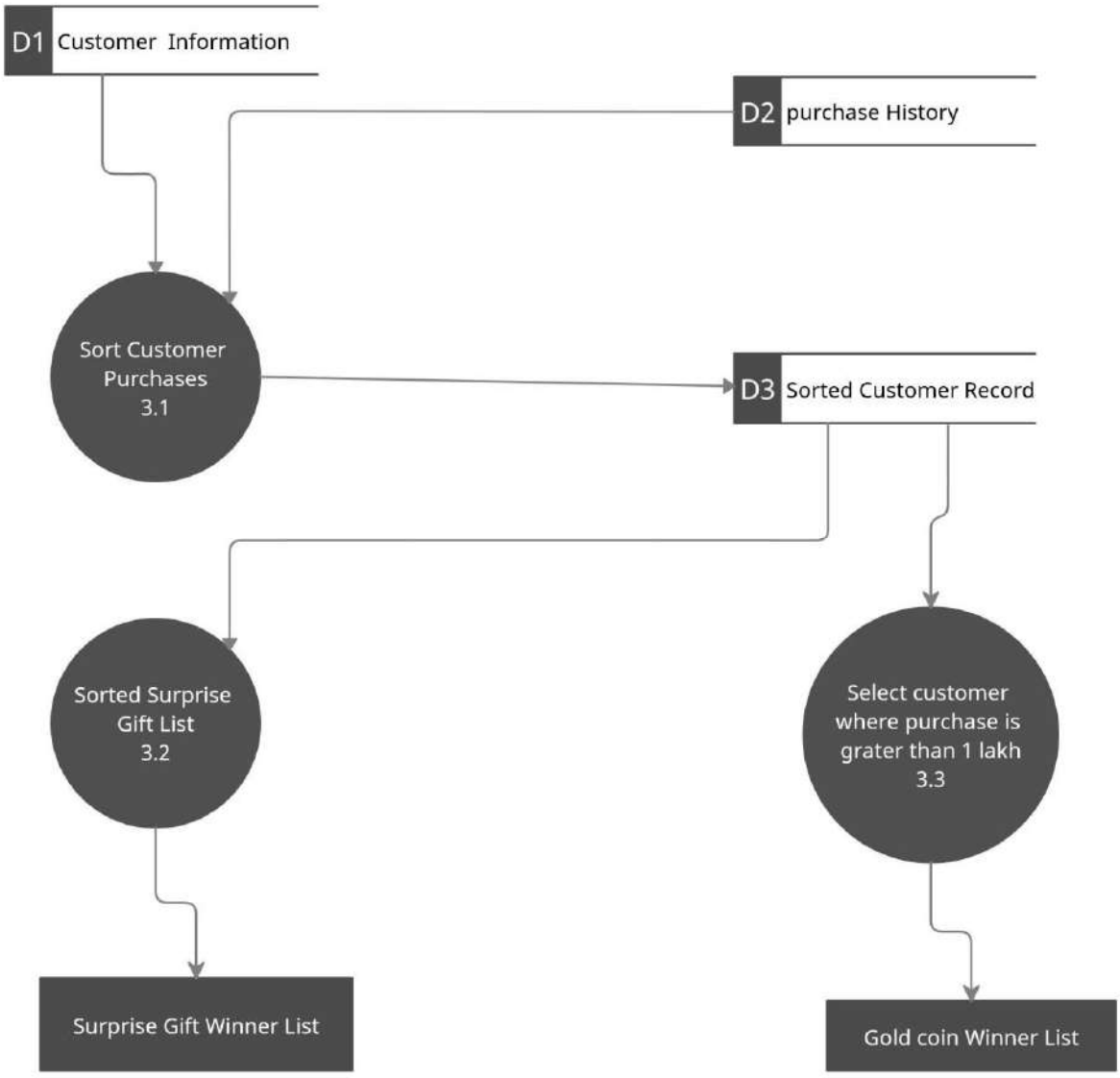
- Zero Level DFD(Context Diagram):



level One DFD:



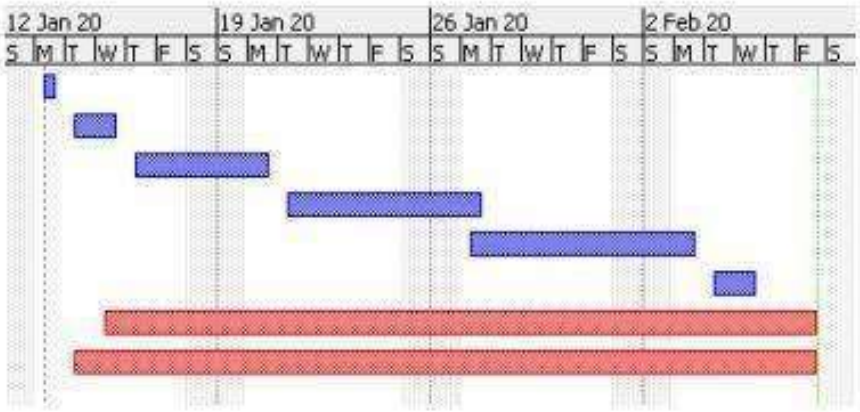
- Level Two of process 3:



Project Management:

Project Scheduling:

	Name	Duration	Start	Finish
1	Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2	Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3	Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4	Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5	Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6	Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7	Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8	User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



Project Resources:

- Human Resources :
  - project manager
  - System analyst
  - Developer
  - Designer
  - Tester
  - Document creator

- Software Resource : ▪ MS Word
  - MS Project

- C (For forehand programming)
- SQL (For backhand programming)

- Hardware Resource :
  - Processor: Intel() Core(TM) i5-4310U 2.60Ghz
  - Ram : 12.00 GB

Risk Management plan:

Risk factor =

$$\frac{\text{(Risk exposure before reduction-Risk exposure after reduction)}}{\text{Cost of reduction}}$$

No such risk is there in the supermarket management system as it uses incremental model.

Testing:
Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

Maintenance: Maintenance stands for all modifications and updation done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

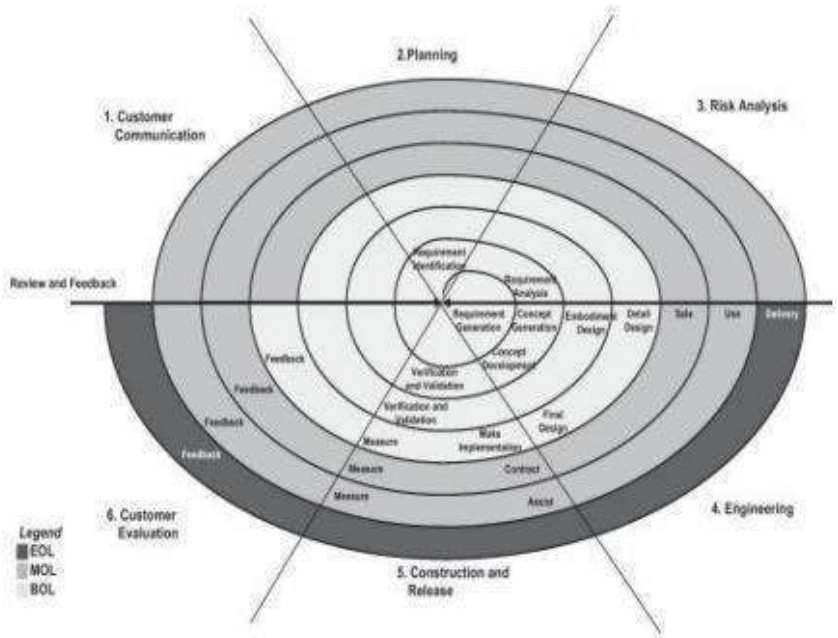
Project 2:

Route Management System

PROBLEM STATEMENT

It identifies internal datastores of payment, bus route, customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule. It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL:



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:
  - ★ Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
  - ★ Spiral Model consists of Risk Analysis task region.
  - ★ In this model, we systematically prepare prototype and release for customer feedback.
- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

Software Requirements Specifications

1. Goal of the Project:

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. Functional Requirements:

As per the requirements of the software from client end. By analysis we realise that we need five modules: A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

-This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password. B. Customer

Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details. C. Schedule Module:

This module maintains timings of the bus.
This include function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried. D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

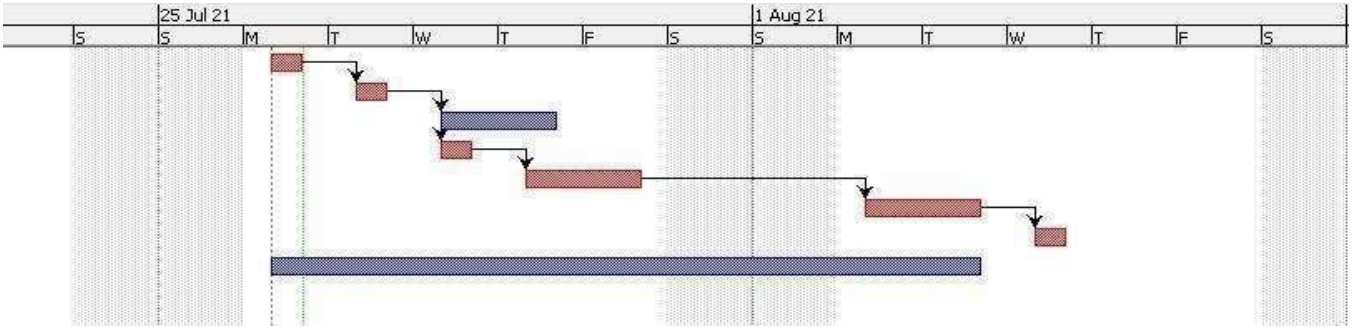
B. Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT:

a)Project Scheduling:-

	Name	Duration	Start	Finish	Predecessors
1	Probleam Statement	1 day	7/26/21 8:00 AM	7/26/21 5:00 PM	
2	Customer Commmunication	1 day	7/27/21 8:00 AM	7/27/21 5:00 PM	1
3	Planning	2 days	7/28/21 8:00 AM	7/29/21 5:00 PM	2
4	Risk Analysis	1 day	7/28/21 8:00 AM	7/28/21 5:00 PM	2
5	Engineering	2 days	7/29/21 8:00 AM	7/30/21 5:00 PM	4
6	Construction and Release	2 days	8/2/21 8:00 AM	8/3/21 5:00 PM	5
7	Consumer Evaluation	1 day?	8/4/21 8:00 AM	8/4/21 5:00 PM	6
8	Write User manual	7 days	7/26/21 8:00 AM	8/3/21 5:00 PM	



b. Project Resources

a. **ManPower Resources:**
   i. **Project Manager ii. System Analyst iii. Designer iv. Developer v. Tester vi. Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above ii. Ram: 4Gb or above iii. Java Development Kit - 13 iv. Java Runtime Environment v. Sublime Text 3 (Text Editor) vi. Ms - Project vii. Create.ly viii. Google Docs**

c. Risk Management Plan

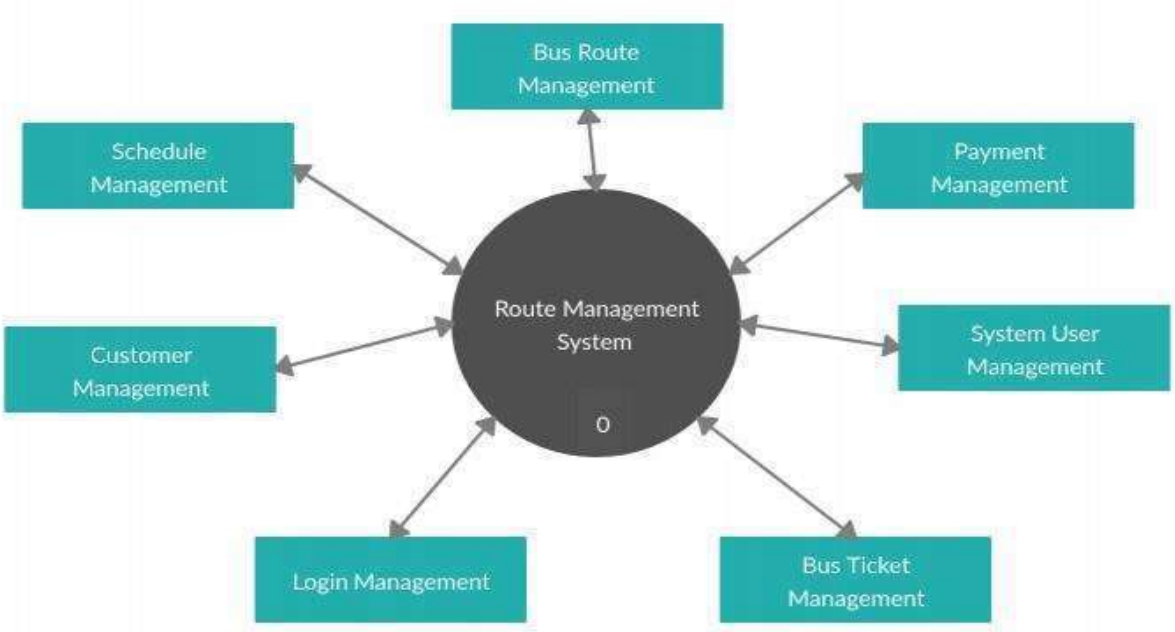
All risks are measured in terms of : A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk Priority of each risk is multiplication of above two Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

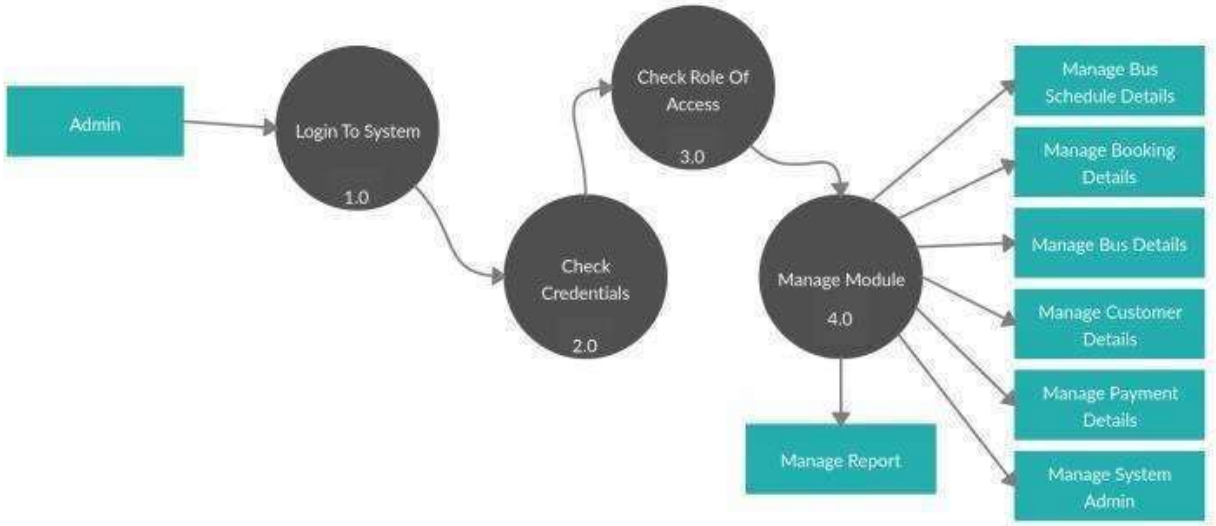
DESIGN ENGINEERING

1. *Architectural Design*



2. Component Level Design

Level 1 DFD



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation 2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.

C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

Integration testing is done to verify the functional, performance,and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance 2. Adaptive
Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, delete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - ○ Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes. ○ Requires low up-front commitment
        - ○ The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

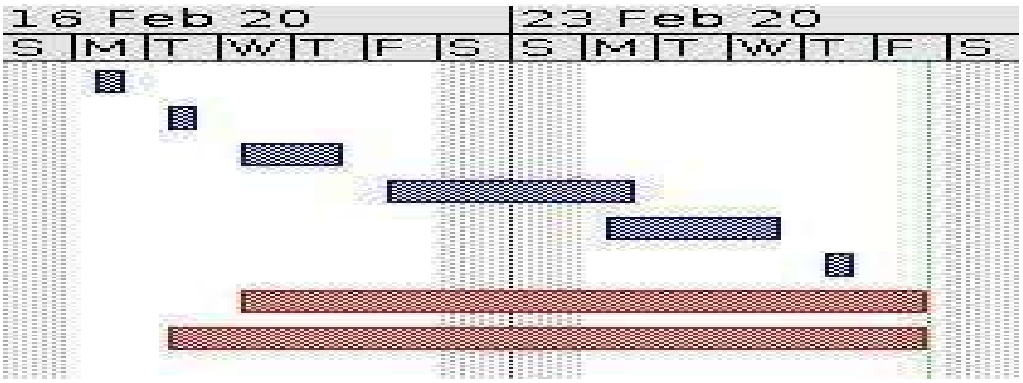
As per the requirements of the software from client end by analysis we realize that we need three modules.

- Goal of the project : The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.
- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
    - Doctor module : This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.
    - Report module : This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.
    - Diagnostic module : This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
    - Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



- Project Resources :
  - ○ Hardware Resources : Several computer machines.
  - ○ Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

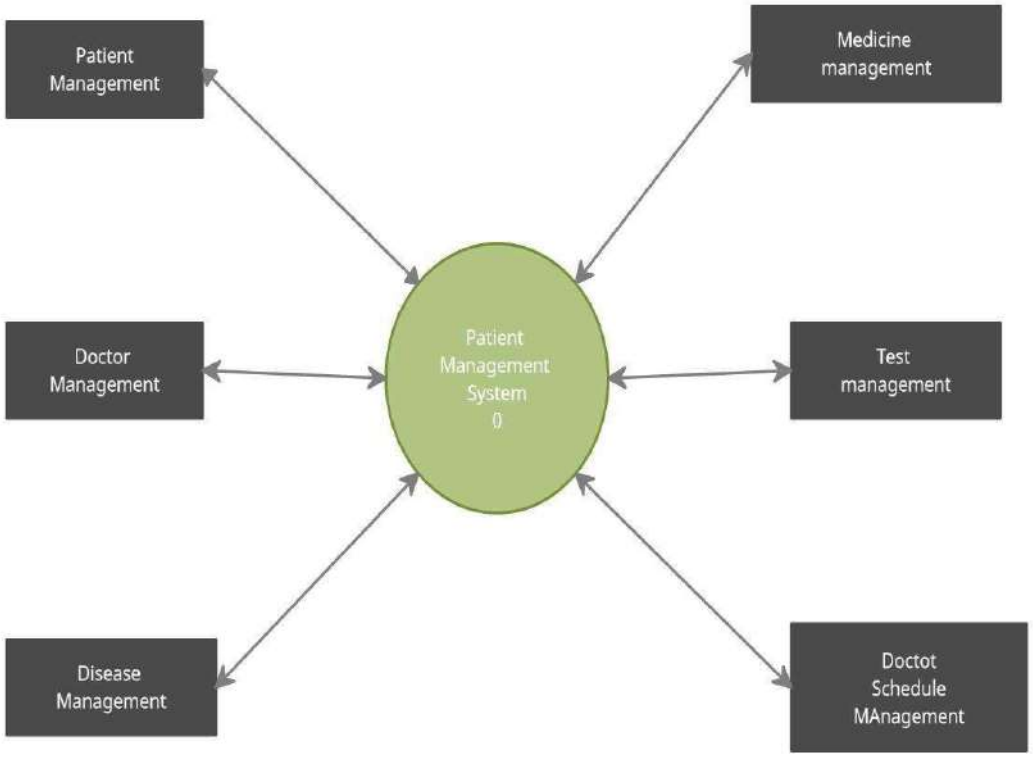
- Risk management plan : Risks are measured by 2 parameters.
  - ○ Likelihood of a risk coming through (Probability of occurrence).
  - ○ Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - ○ Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

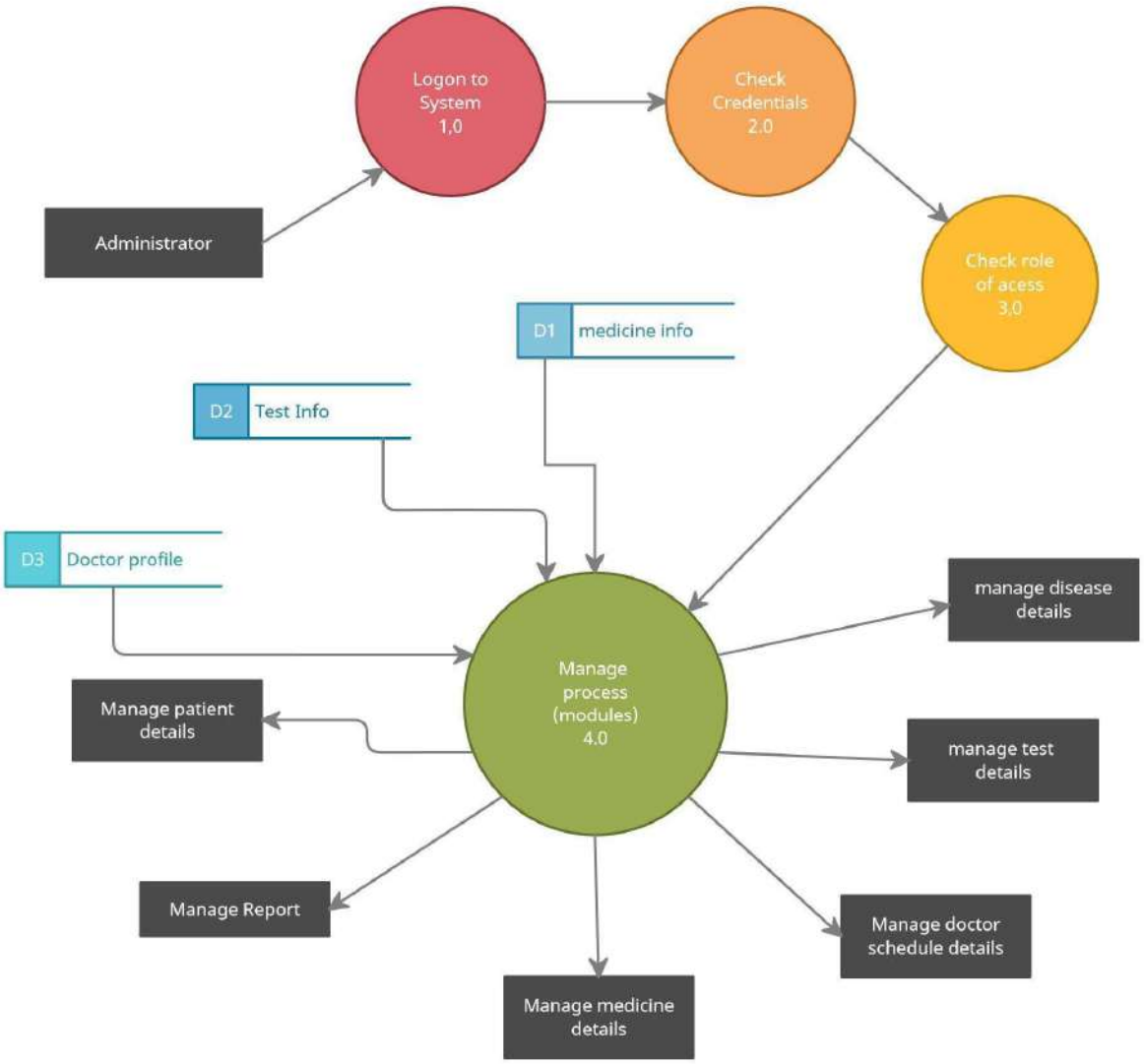
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT:

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to check out the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - ○ Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - ○ Spiral Model consists of risk analysis task region.
    - ○ In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

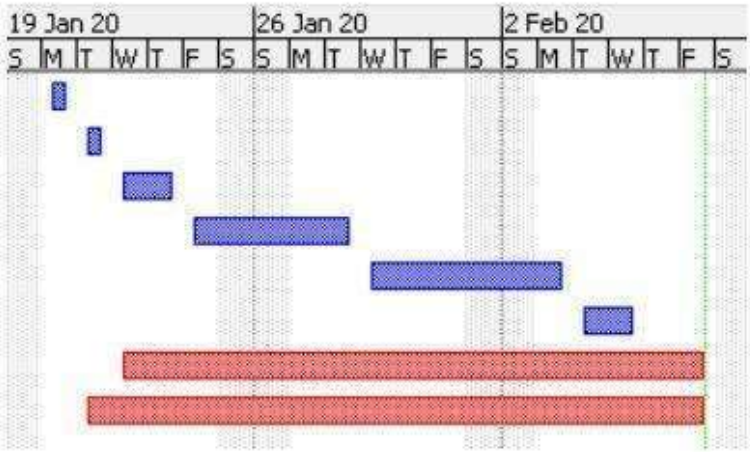
As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. **Guest module:** In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.
2. **Stay module:** In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.
3. **Catering module:** In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no.. Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.
4. **Bill generation module:** At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2		Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3		Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4		Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5		Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM

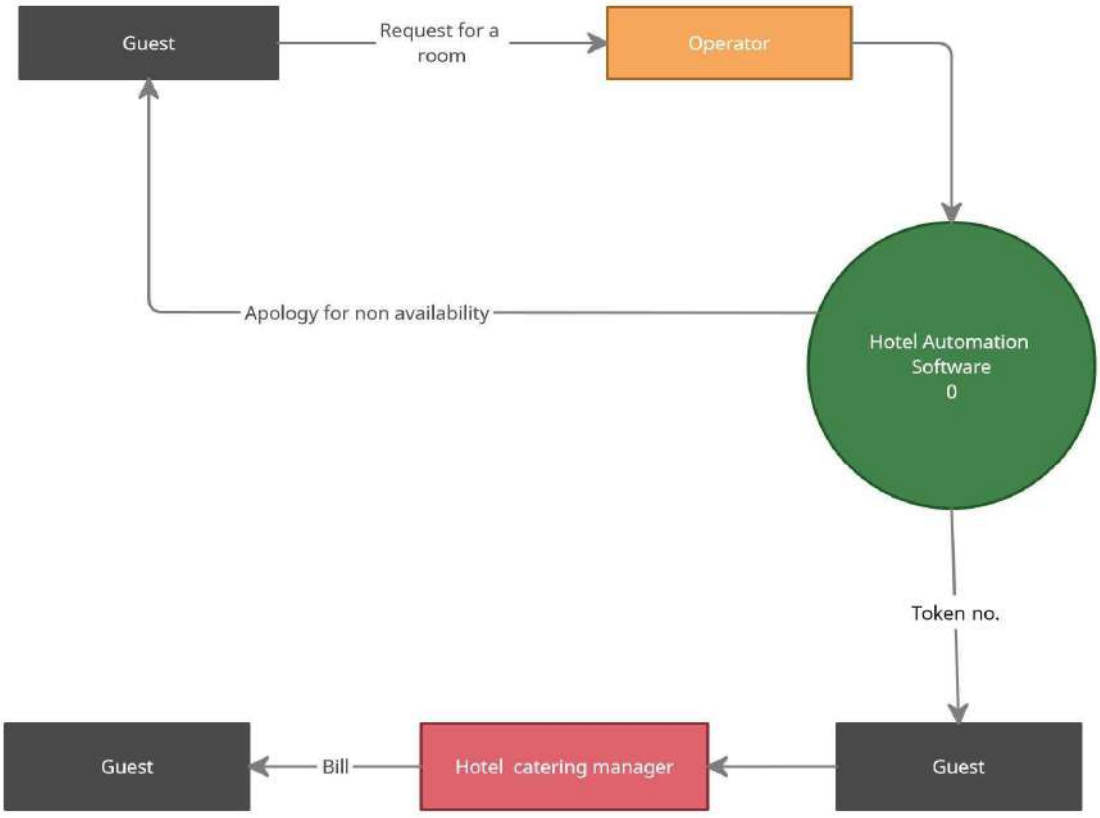


- **Project Resources :**
  - **Hardware Resources:** Several computer machines.
  - **Human Resources:** Manpower resources like project manager, designer, analysist, programmer, tester etc.
- **Risk management plan:** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

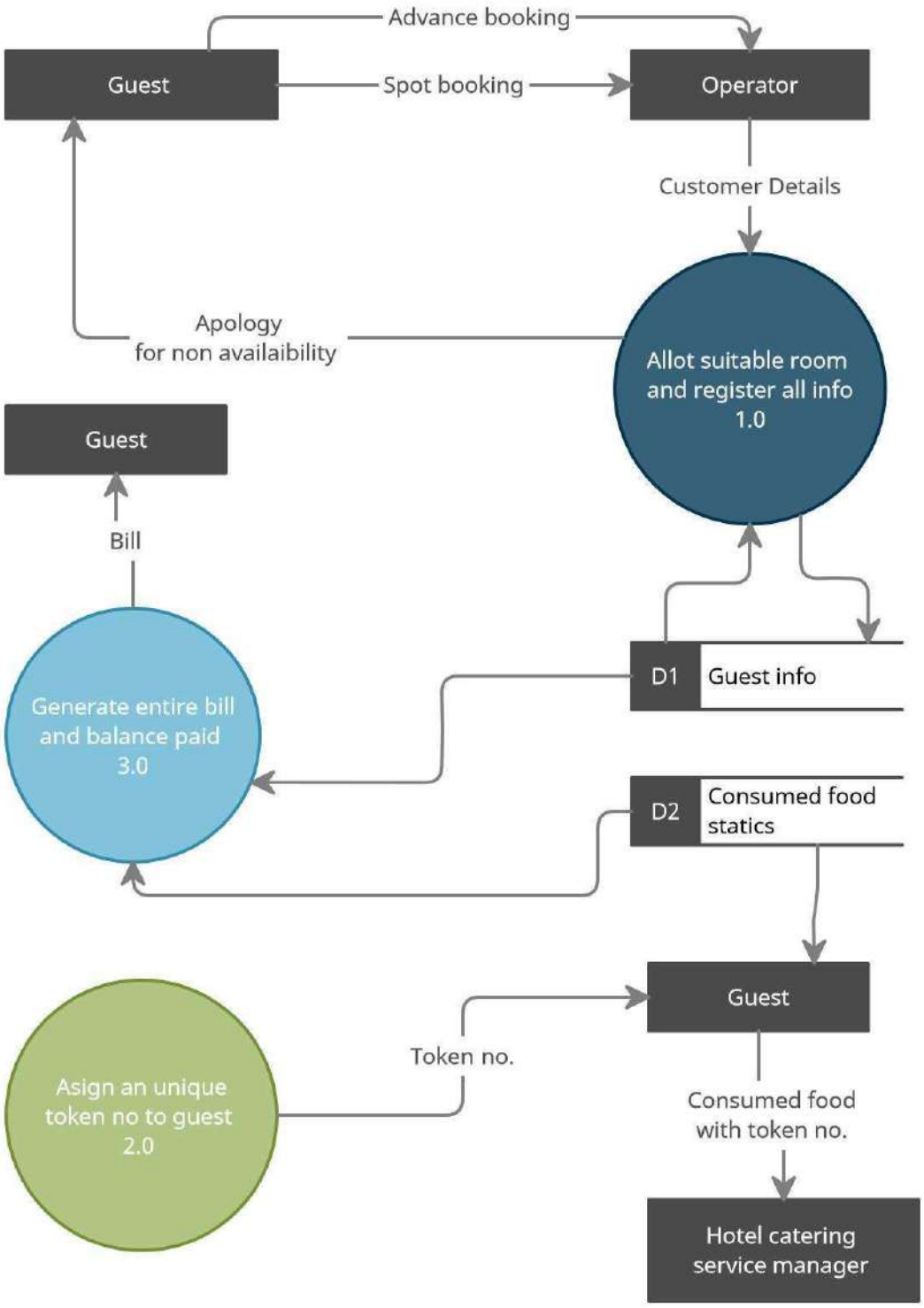
    (Risk exposure before reduction - Risk exposure before reduction)/ (Cost of reduction)

Design Engineering : Context Diagram

1. LEVEL 0 DFD:



Level 1 DFD :


Advance booking
Guest
Spot booking
Operator
Customer Details
Apology
for non availaibility
Allot suitable room
and register all info
1.0
Guest
Bill
Generate entire bill
and balance paid
3.0
D1
Guest info
D2
Consumed food
statics
Guest
Asign an unique
token no to guest
2.0
Token no.
Consumed food
with token no.
Hotel catering
service manager

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT:

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, charge-sheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL:

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons:
    - ○ Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes. ○ Requires low up-front commitment
        - ○ The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS):

As per the requirements of the software from client end by analysis we realize that we need three modules.

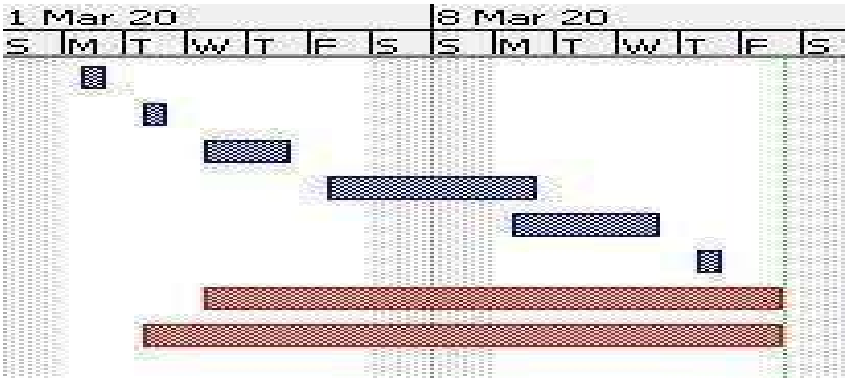
- Goal of the project: The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.
- Functional requirements: The three modules needed are.
    - Administrator module: This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
        - Criminal module: This module accesses data from the database of criminal record details and manages it properly.
        - FIR module: This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
        - Charge-sheet module: This module accesses data from the database of charge-sheet record details and manages it properly.
        - Court module: The main function of this module is to manage the court profiles.

- Non-Functional Requirements:
    - Usability: The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
        - Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
        - Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT:

- Project Scheduling:
-

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



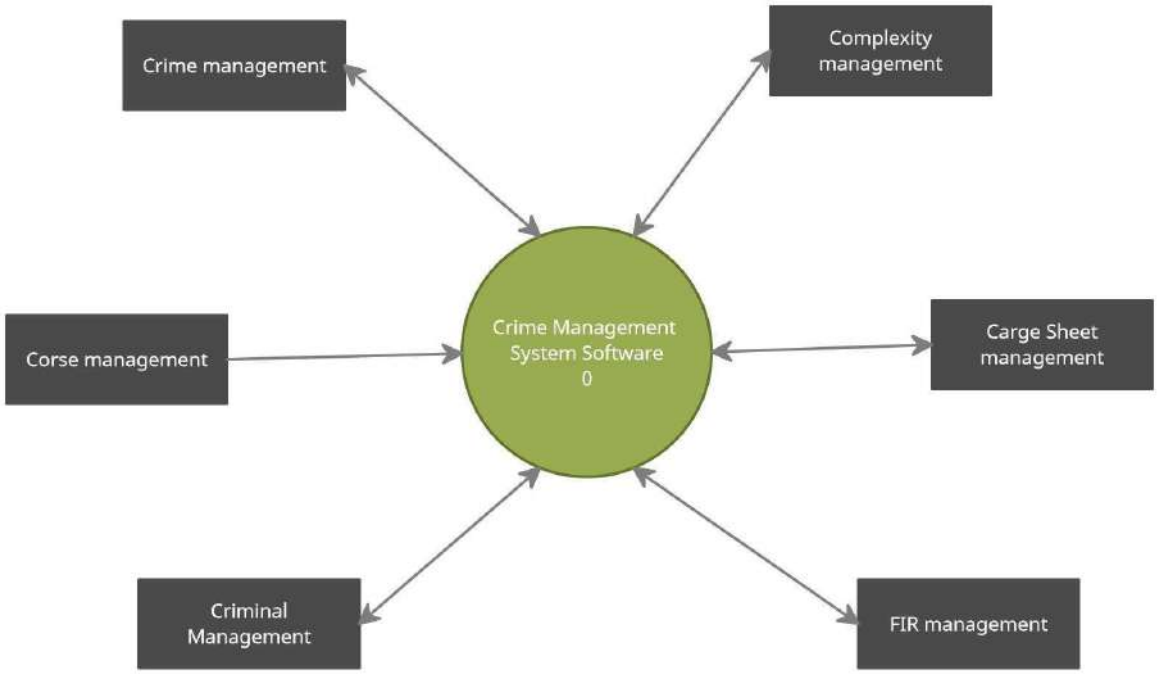
- Project Resources:
    - Hardware Resources: Several computer machines.
    - Human Resources: Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan: Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
    - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

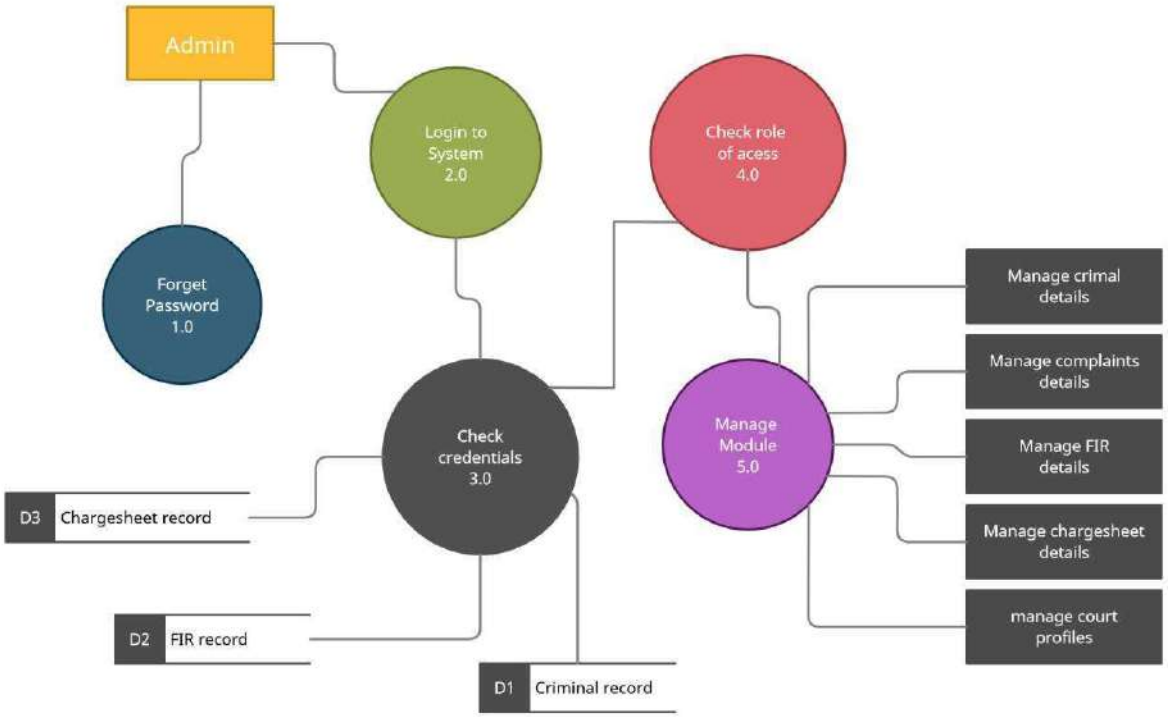
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram 1. Level 0

DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.

  This time this testing is done in three parts.

  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT:

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
        - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realize that we need three modules.

- Goal of the project : The aim and objectives for students' examination

  Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed

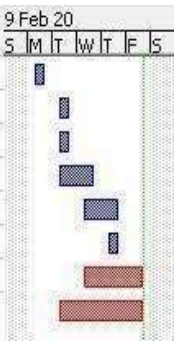
of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements : The three modules needed are.
    - Login and recovery module : This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
    - Credentials module : This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
    - Administrator module : This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

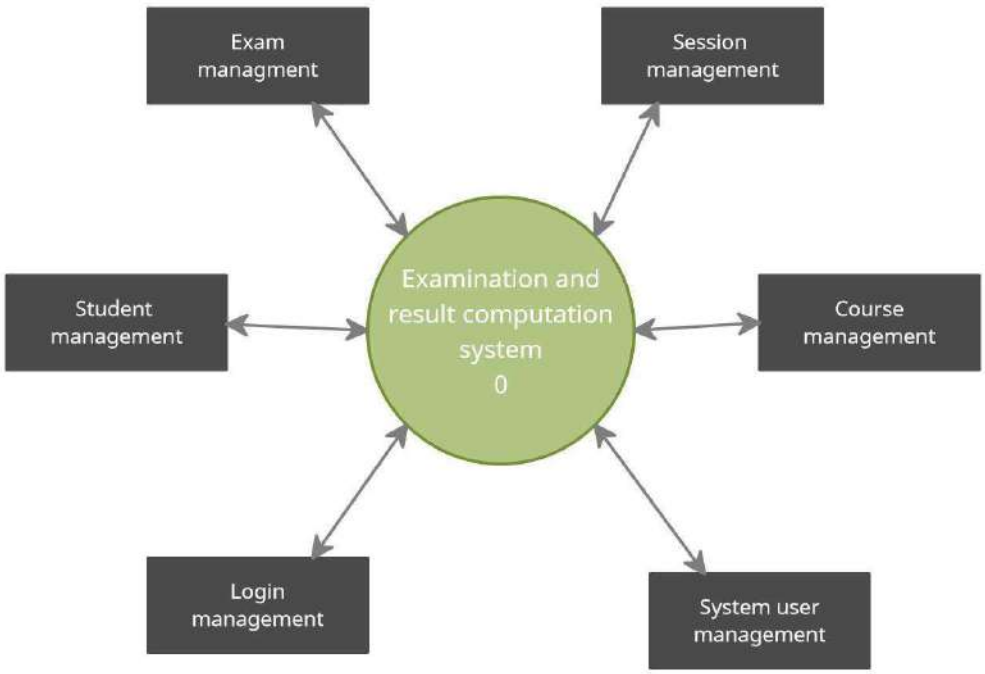


- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

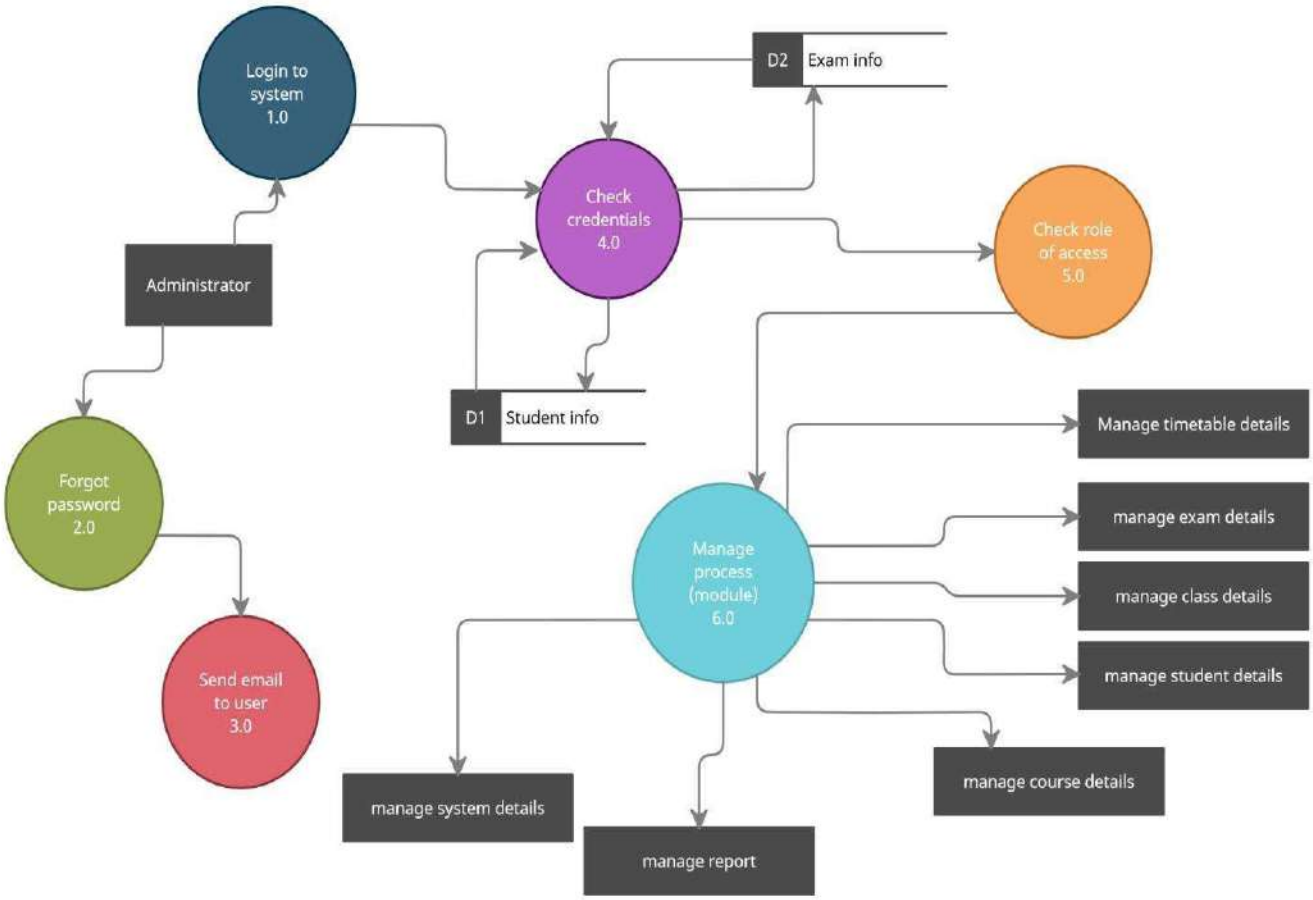
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested.

  This time this testing is done in three parts.

  - Alpha testing (Verification): It is done by the development team.
  - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.

○ Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE:

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

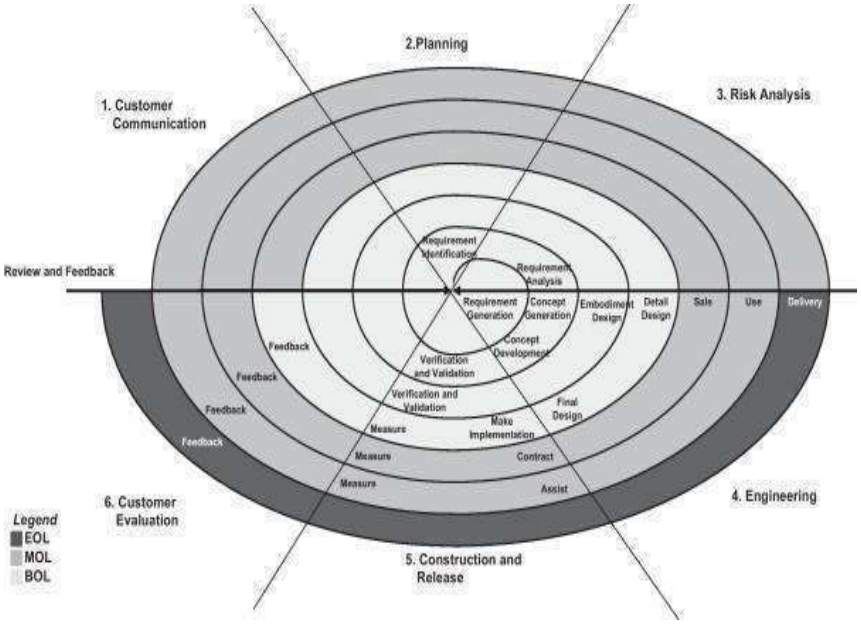
Project 7:

Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. ***Goal of the Project:***

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc. which was a hefty work.

2. ***Functional Requirements:***

As per the requirements of the software from client end.

By analysis we realize that we need three modules:

A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software. Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. Parking Module:

This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

*3. **Non-Functional Requirements:***

A. **Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.**

B. **Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.**

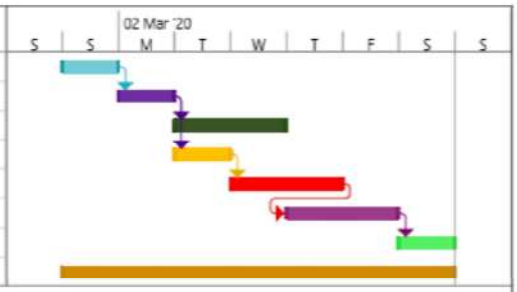
C. **Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.**

PROJECT MANAGEMENT

a.Project Scheduling

b.Project Resources

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



a. **ManPower Resources:**

i. **Project Manager ii. System Analyst iii. Designer iv. Developer v. Tester vi. Document Writer**

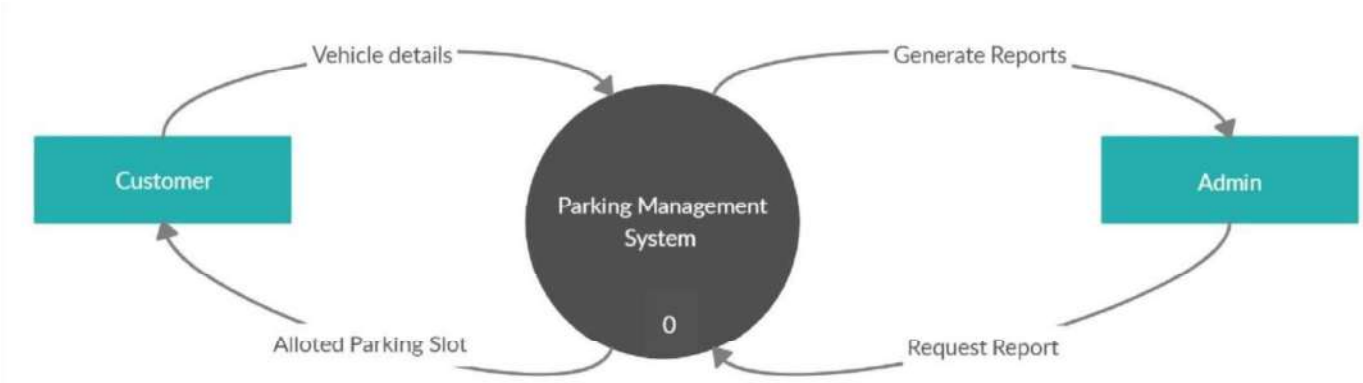
b. **Hardware-Software Resources**

i. **Processor: Intel i3 4th gen or above**

ii.Ram: 4Gb or above

iii. Java Development Kit 13 iv.Java Runtime Environment v.Sublime Text 3 (Text Editor) vi. Ms - Project vii. Creately viii. Google Docs.

Level 1 DFD:



Level-2 DFD

c. Risk Management Plan:

All risks are measured in terms of

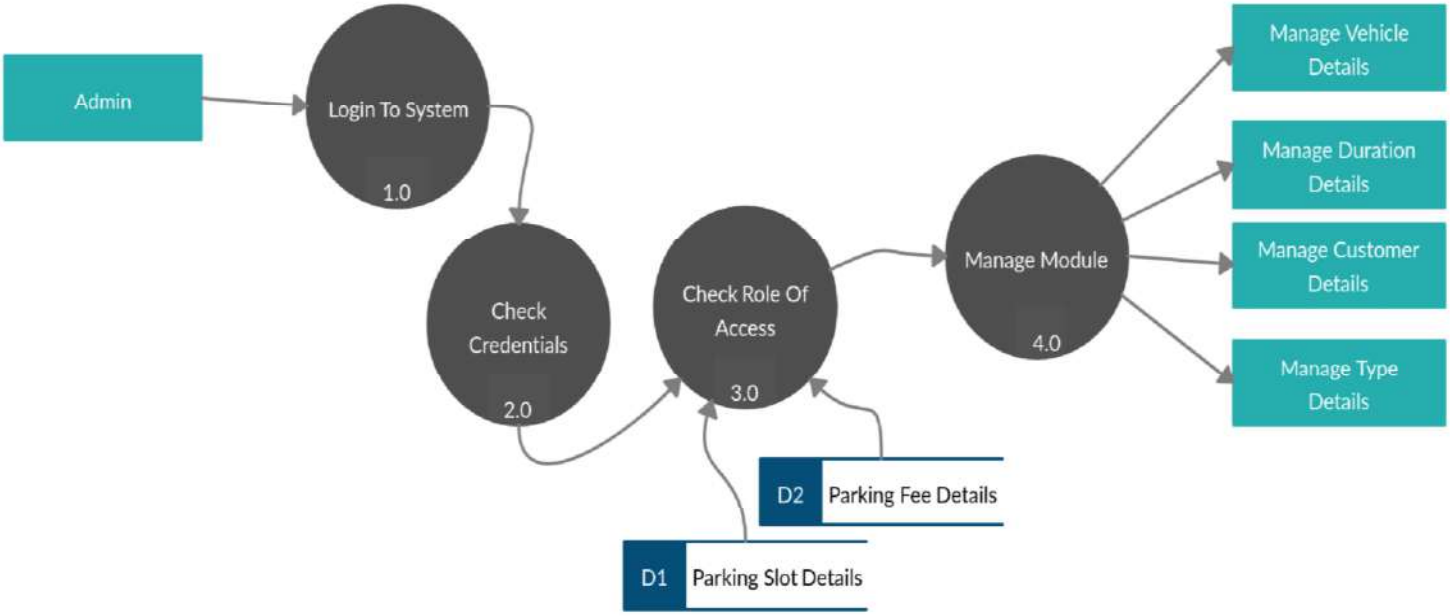
A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk Priority of each risk is multiplication of above two.

- Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation **2. System Testing:**

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team

B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.

C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE:

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project 8:

Wholesale Management System

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

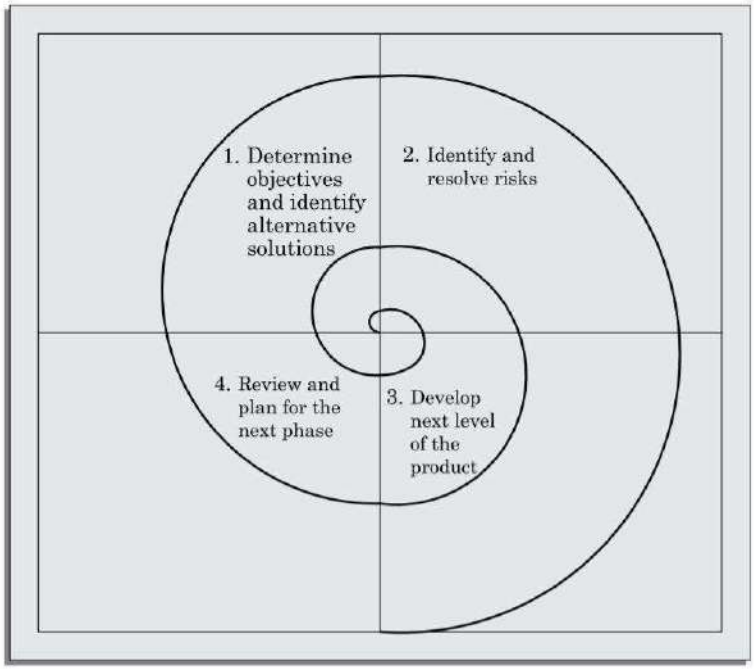


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core.

Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module** – In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module** – In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module** – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

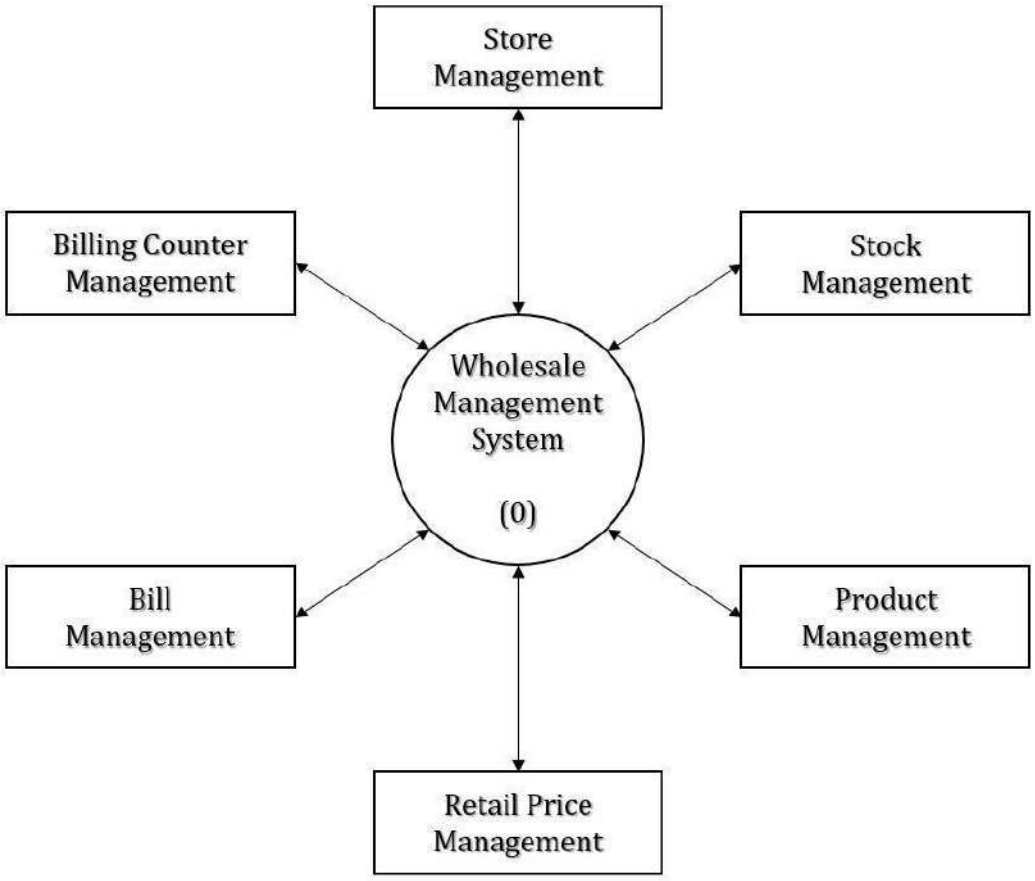


Fig – DFD level zero

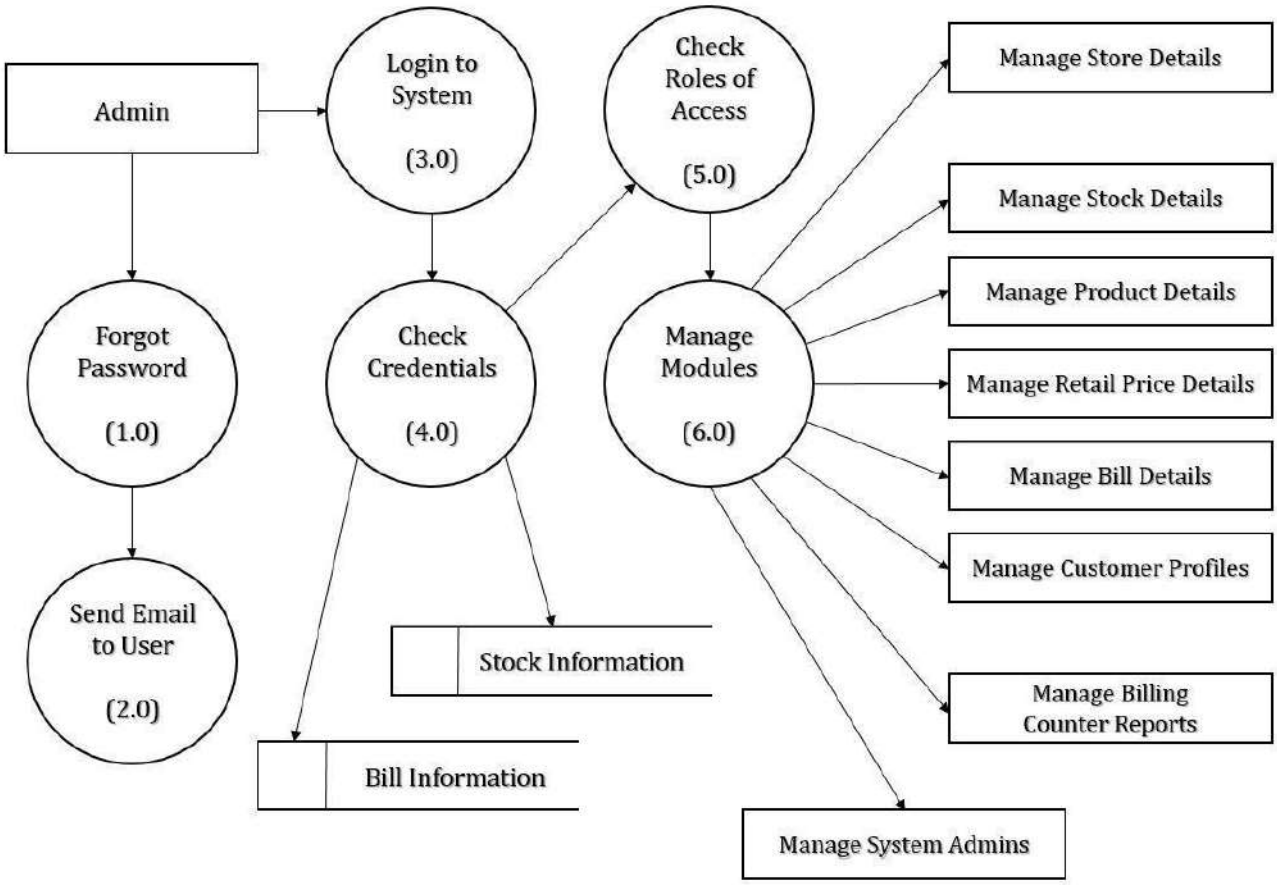


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

Attached file – GanttWholesale.pdf

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$
where **p** – priority with which the risk must be handled **r** – the probability of the risk becoming true **s** – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60. Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Name: Debpritam Roy

Semester: V

Department: Computer Science (B.Sc.)

Collage Roll No.: 704

Registration No.: A01-111-1112-003-2019

Subject: Cloud Computing

Google App Engine

Supervisor's Name: Dr. Chayan Halder

CONTENTS

Introduction

A cloud service provider, or CSP, is a company that offers some component of cloud computing; typically when you search the internet a cloud service is defined as, **infrastructure as a service (IaaS)**, **software as a service (SaaS)** or **platform as a service (PaaS)** to other businesses or individuals.

"A Cloud Service is any system that provides on-demand availability of computer system resources, e.g.; data storage and computing power, without direct active management by the user".

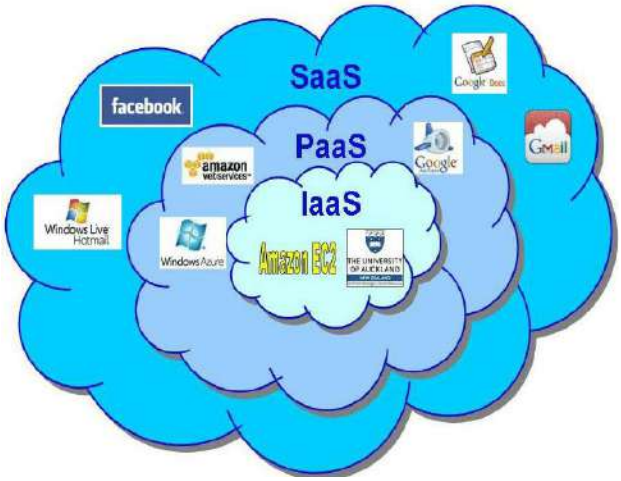


Fig. Basic Cloud Model.

What are the Models Present?

There are the mainly three types of cloud service models –

1. Infrastructure as a Service (IaaS).
2. Platform as a Service (PaaS).
3. Software as a Service (SaaS).

Some other cloud service models are –

- Business Provider as a Service (BPaaS).
- Function as a Service (FaaS).
- Database as a Service (DBaaS).
- Identity as a Service (IDaaS).
- Analytics as a Service (AaaS).
- Backend as a Service (BaaS).

Infrastructure as a Service (IaaS)

(https://en.wikipedia.org/wiki/Infrastructure_as_a_service)

Infrastructure as a service (IaaS) is a type of cloud computing service that offers essential compute, storage and networking resources on demand, on a pay-as-you-go basis. IaaS is one of the four types of cloud services, along with software as a service (SaaS), platform as a service (PaaS) and serverless.

Migrating your organization's infrastructure to an IaaS solution helps you reduce maintenance of on-premises data centers, save money on hardware costs and gain real-time business insights. IaaS solutions give you the flexibility to scale your IT resources up and down with demand. They also help you quickly provision new applications and increase the reliability of your underlying infrastructure.

Popular examples of IaaS include:

- DigitalOcean.
- Linode.
- Rackspace.
- Amazon Web Services (AWS)
- Cisco Metacloud.
- Microsoft Azure.
- Google Compute Engine (GCE)

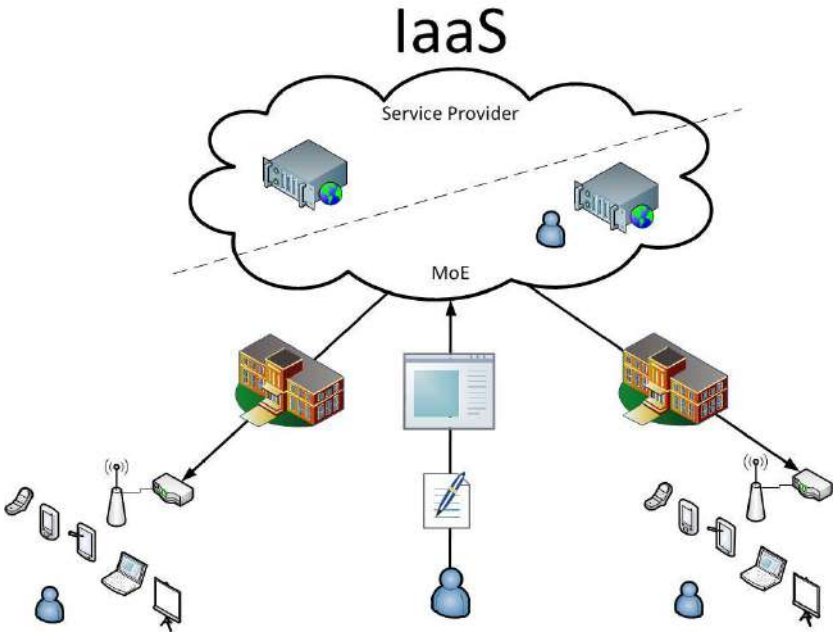


Fig. Infrastructure as a Service

Platform as a Service (PaaS) (https://en.wikipedia.org/wiki/Platform_as_a_service)

Platform as a service (PaaS) is a cloud computing model where a third-party provider delivers hardware and software tools to users over the internet. Usually, these tools are needed for application development. A PaaS provider hosts the hardware and software on its own **infrastructure**. As a result, PaaS frees developers from having to install in-house hardware and software to develop or run a new application.

PaaS tools tend to be touted as simple to use and convenient. Users will normally have to pay on a per-use basis. An organization may find the move to a **PaaS compelling considering** potential cost savings over using on-premises alternatives.

Popular examples of PaaS:

- AWS Elastic Beanstalk.
- Windows Azure.
- Heroku.
- Force.com.
- Google App Engine.
- OpenShift.

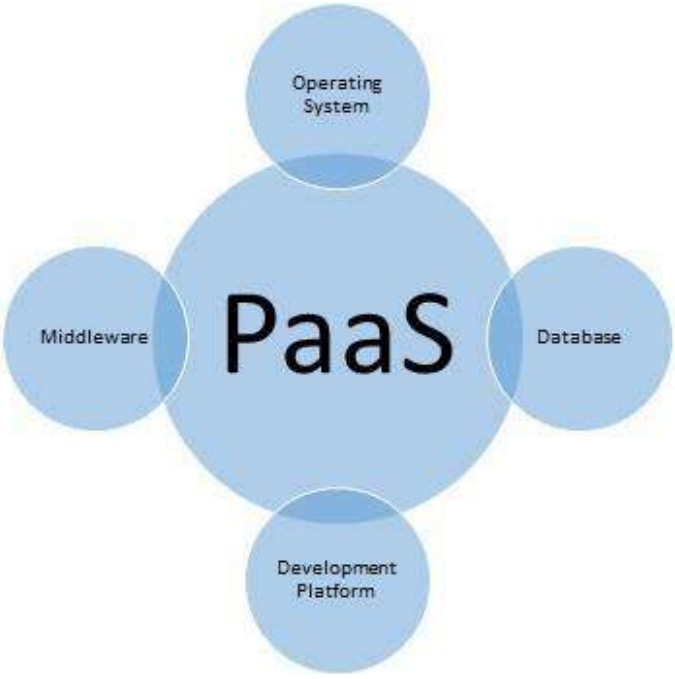


Fig. Platform as a Service

Software as a Service (SaaS) (https://en.wikipedia.org/wiki/Software_as_a_service)

Software as a service (SaaS) is a **software distribution model in which a cloud provider hosts applications and makes them available to end users over the internet**. In this model, an **independent software vendor (ISV)** may contract a third-party cloud provider to host the application. Or, with larger companies, such as Microsoft, the cloud provider might also be the software vendor.

Software-as-a-Service is typically accessed through a **web browser**, with users logging into the system using a username and password. Instead of each user having to install the software on their computer, the user is able to access the program via the Internet.

Popular examples of SaaS

- Google Workspace (formerly GSuite)
- Dropbox.
- Salesforce.
- Cisco WebEx.
- SAP Concur.
- GoToMeeting.

Fig. Software as a Service

What is Google App Engine?

(https://appengine.google.com/) (https://en.wikipedia.org/wiki/Google_App_Engine)

Google App Engine is a Platform as a Service (PaaS) product that provides Web app developers and enterprises with access to Google's scalable hosting and tier 1 Internet service.

The App Engine requires that apps be written in Java or Python, store data in Google BigTable and use the Google query language. Non-compliant applications require modification to use App Engine.

Google App Engine provides more infrastructure than other scalable hosting services such as Amazon Elastic Compute Cloud (EC2). The App Engine also eliminates some system administration and developmental tasks to make it easier to write scalable applications.

Google App Engine is free up to a certain amount of resource usage. Users exceeding the per-day or per-minute usage rates for CPU resources, storage, number of API calls or requests and concurrent requests can pay for more of these resources.

Fig. Services of Google App Engine

How is service provided?

When a file is uploaded on the Google cloud, the unique metadata is inserted into a file. It helps identify the different files and track the changes made across all the copies of any particular file. All the changes made by individuals get synchronized automatically to the main file, also called a master file. GCP further updates all the downloaded files using metadata to maintain the correct records.

Let's understand the working of GCP with a general example:

Suppose that MS Office is implemented on Cloud to enable several people to work together. The primary aim of using cloud technology is to work on the same project at the same time. We can create and save a file on the cloud once we install a plugin for the MS Office suite. This will allow several people to edit a document at the same time. The owner can assign access to specific people to allow them to download and start editing the document in MS Office.

Once users are assigned as an editor, they can use and edit the document's cloud copy as desired. The combined, edited copy is generated that is known as the master document. GCP helps to assign a unique URL to each specific copy of the existing document given to different users. However, any of the authorized users' changes will be visible on all the copies of documents shared over the cloud. In case multiple changes are made to the same document, then GCP allows the owner to select the appropriate changes to keep.

Google Cloud Platform Services

Google provides a considerable number of services with several unique features. That is the reason why Google Cloud Platform is continually expanding across the globe. Some of the significant services of GCP are:

- Compute Services
- Networking
- Storage Services
- Big Data
- Security and Identity Management
- Management Tools
- Cloud AI
- IoT (Internet of Things)

Compute Services

GCP offers a scalable range of computing services, such as:

- **Google App Engine**: It is a cloud computing platform that follows the concept of Platform-as-a-Service to deploy PHP, Java and other software. It is also used to develop and deploy web-based software in Google-managed data centers. The most significant advantage of Google App Engine is its automatic scaling capability. This means that the App Engine automatically allocates more resources for the application when there is an increase in requests.
- **Compute Engine**: It is a cloud computing platform that follows the concept of Infrastructure-as-a-Service to run Windows and Linux based virtual machines. It is an essential component of GCP. It is designed on the same infrastructure used by Google search engine, YouTube and other Google services.
- **Kubernetes Engines**: This computing service is responsible for offering a platform for automatic deployment, scaling, and other operations of application containers across clusters of hosts. The engine supports several container tools like a docker, etc.

Networking

GCP includes the following network services:

- **VPC**: VPC stands for Virtual Private Network. The primary function of VPC is to offer a private network with routing, IP allocation, and network firewall policies. This will help to create a secure environment for the application deployments.
- **Cloud Load Balancing**: As its name states, Cloud balancing are used to distribute workload across different computing resources to balance the entire system performance. This also results in cost-reduction. The process also helps in minimizing the availability and maximizing the capability of the resources.
- **Content Delivery Network**: CDN is a geographically distributed network of proxy servers and their data centers. The primary aim of using CDN is to provide maximum performance to the users. Additionally, it also helps deliver high availability of resources by equally distributing the related services to the end-users.

Storage Services

GCP has the following storage services:

- **Google Cloud Storage**: It is an online data storage web service that Google provides to its users to store and access data from anywhere. The service also includes a wide range of features like maximum performance, scalability, security and sharing.
- **Cloud SQL**: It is a web-service that enables users to create, manage, and use relational databases stored on Google Cloud servers. The service itself maintains and protects the databases, which helps users focus on their applications and other operations.
- **Cloud Bigtable**: It is known for its fast performance and highly manageable feature. It is a highly scalable NoSQL database service that allows collecting and retaining data from as low as 1 TB to hundreds of PB.

Big Data

GCP provides a variety of services related to big data; they are:

- **BigQuery**: It is a fully managed data analysis service by Google. The primary aim of Google BigQuery service is to helps businesses to analyze Big Data. It offers a highly scalable data management option. This means BigQuery allows users to perform ad-hoc queries and share data insights across the web.
- **Google Cloud Datastore**: Google Cloud Datastore is a kind of datastore service that is fully managed, schema-less, and non-relational. This service enables businesses to perform automatic transactions and a rich set of queries. The main advantage of Google Cloud Datastore is the capability of automatic scaling. This means that the service can itself scale up and down, depending on the requirement of resources.
- **Google Cloud Dataproc**: It is a very fast and easy to use big data service offered by Google. It mainly helps in managing Hadoop and Spark services for distributed data processing. The service allows users to create Hadoop or Spark clusters sized according to the overall workload and can be accessed whenever users want them.

Security and Identity Management

GCP includes the following services related to Security and Identity management:

- **Cloud Data Loss Prevention API**: It is mainly designed to manage sensitive data. It helps users manage sensitive data elements like credit card details, debit card details, passport numbers, etc. It offers fast and scalable classification for sensitive data.
- **Cloud IAM**: It stands for Cloud Identity and Access Management. It is a framework that contains rules and policies and validates the authentication of the users for accessing the technology resources. That is why it is also known as Identity Management (IdM).

Management Tools

GCP includes the following services related to management tools:

- **Google Stackdriver**: Google Stackdriver service is primarily responsible for displaying the overall performance and diagnostics information. This may include insights of data monitoring, tracing, logging, error reporting, etc. The service also prompts an alert notification to the public cloud users.
- **Google Cloud Console App**: It is a native mobile application powered by Google. The primary aim of this service is to enable users to manage the core features of Google Cloud services directly from their mobile devices anytime, anywhere. The primary functions of this service are alerting, monitoring, and performing critical actions on resources.

Cloud AI

When it comes to Cloud AI, GCP offers these services:

- **Cloud Machine Learning Engine**: It is another fully managed service that allows users to create Machine Learning models. The service is mainly used for those ML models, which are based on mainstream frameworks.
- **Cloud AutoML**: It is the type of service that is based on Machine Learning. It helps users to enter their data sets and gain access to quality trained pre-designed ML models. The service works by following Google's transfer learning and Neural Architecture Search method.

IoT (Internet of Things)

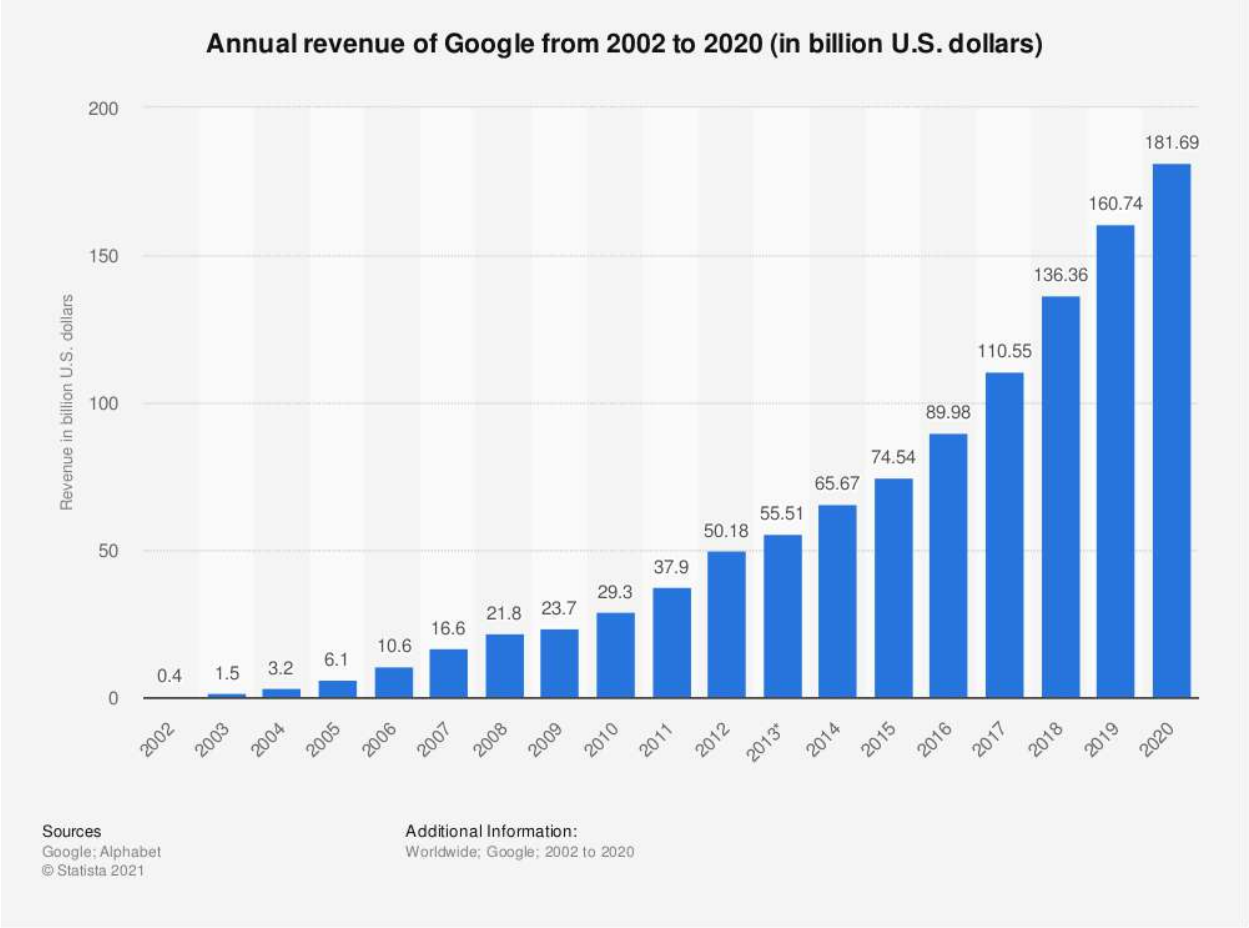
GCP contains the following IoT services:

Cloud IoT Core: It is one of the fully managed core services. It allows users to connect, control, and ingest data from various devices that are securely connected to the Internet. This allows other Google cloud services to analyze process, collect and visualize IoT data in real-time.

Cloud IoT Edge: The Edge computing service brings memory and other computing-power resources near to the location where it is required.

Fig. Google Cloud Platform Services

Annual Graph of Google Cloud Services



Advantages and Disadvantages

(https://stackoverflow.com/questions/1306279/pros-cons-of-google-app-engine)

Advantages of Google Cloud Platform

There are several advantages of using Google Cloud Platform, such as:

- **Google Cloud Offers Quick and Easy Collaboration**: Multiple users can access the data and simultaneously contribute their information. This is possible because the data is stored on the cloud servers, not on the user's personal computers.

- **Higher Productivity with Continuous Development**: Google is always working on adding new features and functionalities to provide higher productivity to the customers. Therefore, Google delivers frequent updates to its products and services.

- **Less Disruption with Adopting New Features**: Instead of pushing huge disruptive updates of changes, Google provides small updates weekly. This helps users to understand and adopt new features easily.

- **Least or Minimal Data is stored on Vulnerable Devices**: Google does not store data on local devices unless a user explicitly tries to do it. This is because the data stored on local devices may get compromised compared to the cloud's data.

- **Users can access Google Cloud from Anywhere**: The best thing is that a user can easily access the information stored on Google cloud from anywhere because it is operated through web-based applications.

- **Google provides Maximum Security with its Robust Structure**: Google hires leading security professionals to protect user's data. Users get process-based and physical security features made by Google.

- **Users have Full Control over their Data**: Users gain full control over services and the data stored in Google Cloud. If a user does not want to use Google services any longer and wants to delete the cloud data, it can be easily performed.

- **Google provides Higher Uptime and Reliability**: Google uses several resources to provide higher and reliable up-time servers. If a data center is not working for technical issues, the system will automatically communicate with the secondary center without interruption visible to users.

Disadvantages of Google Cloud Platform

There are some disadvantages of using Google Cloud Platform, such as:

- Locked into Google App Engine?
- Developers have read-only access to the file system on App Engine.
- App Engine can only execute code called from an HTTP request (except for scheduled background tasks).
- Users may upload arbitrary Python modules, but only if they are pure-Python; C and Pyrex modules are not supported.
- App Engine limits the maximum rows returned from an entity get to 1000 rows per Datastore call. (**Update** - App Engine now supports cursors for accessing larger queries)
- Java applications may only use a subset (The JRE Class White List) of the classes from the JRE standard edition.
- Java applications cannot create new threads.

Microsoft Azure

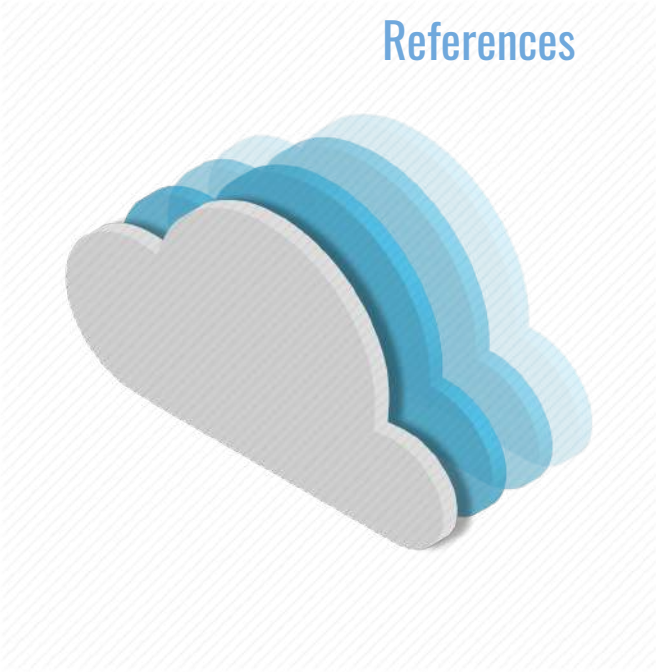
Case Study

Name	Alapan Ghosh
Dept.	Computer Science
Semester	V
Subject	Cloud Computing (DSE3)
Roll	705
Reg.No.	A01-1112-117-004-2019

Supervisor's Name: Dr. Chayan Halder

Ramakrishna Mission Vivekananda
Centenary College Rahara

CONTENTS



What is Cloud Computing ?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end users. Following are the working models of cloud computing:

- ➢ Deployment Models
- ➢ Service Models

Fig 1.1 Representation of Cloud Computing

What are the types of Deployment Model ?

Public Cloud

Public clouds are owned and operated by third-party cloud service providers, which deliver their computing resources, like servers and storage, over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software, and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

Private Cloud

Private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company's on-site datacenter. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

Hybrid Cloud

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options, and helps optimize your existing infrastructure, security, and compliance.

What are the types of Service Model ?

Infrastructure as a service (IaaS)

The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps.

Software as a service (SaaS)

Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching.

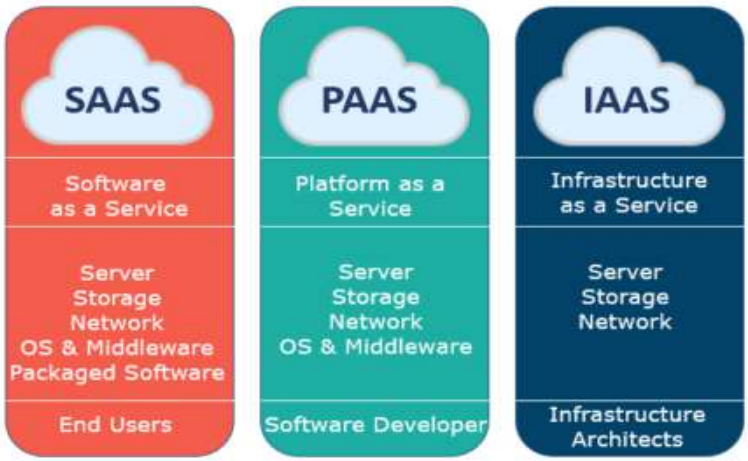


Fig 3.1 Different Types of Cloud as a Service

What is Microsoft Azure ?

Microsoft Windows Azure is a cloud operating system built on top of Microsoft datacenters' infrastructure and provides developers with a collection of services for building applications with cloud technology. Services range from compute, storage, and networking to application connectivity, access control, and business intelligence. Any application that is built on the Microsoft technology can be scaled using the Azure platform, which integrates the scalability features into the common Microsoft technologies such as Microsoft Windows Server 2008, SQL Server, and ASP.NET. The services provided are managed and controlled through the Windows Azure Management Portal, which acts as an administrative console for all the services offered by the Azure platform. In this section, we present the core features of the major services available with Azure.

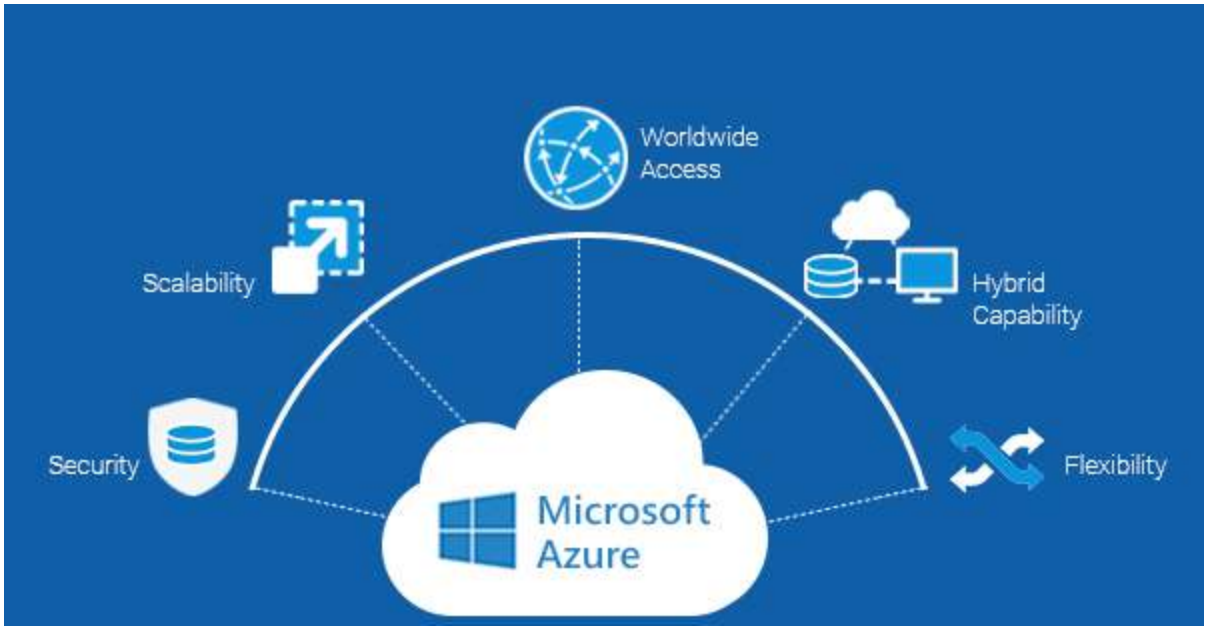


Fig 4.1 Representation of Microsoft Azure

How does Microsoft Azure work ?

The Windows Azure platform is made up of a foundation layer and a set of developer services that can be used to build scalable applications. These services cover compute, storage, networking, and identity management, which are tied together by middleware called AppFabric. This scalable computing environment is hosted within Microsoft datacenters and accessible through the Windows Azure Management Portal. Alternatively, developers can recreate a Windows Azure environment (with limited capabilities) on their own machines for development and testing purposes. In this section, we provide an overview of the Azure middleware and its services.

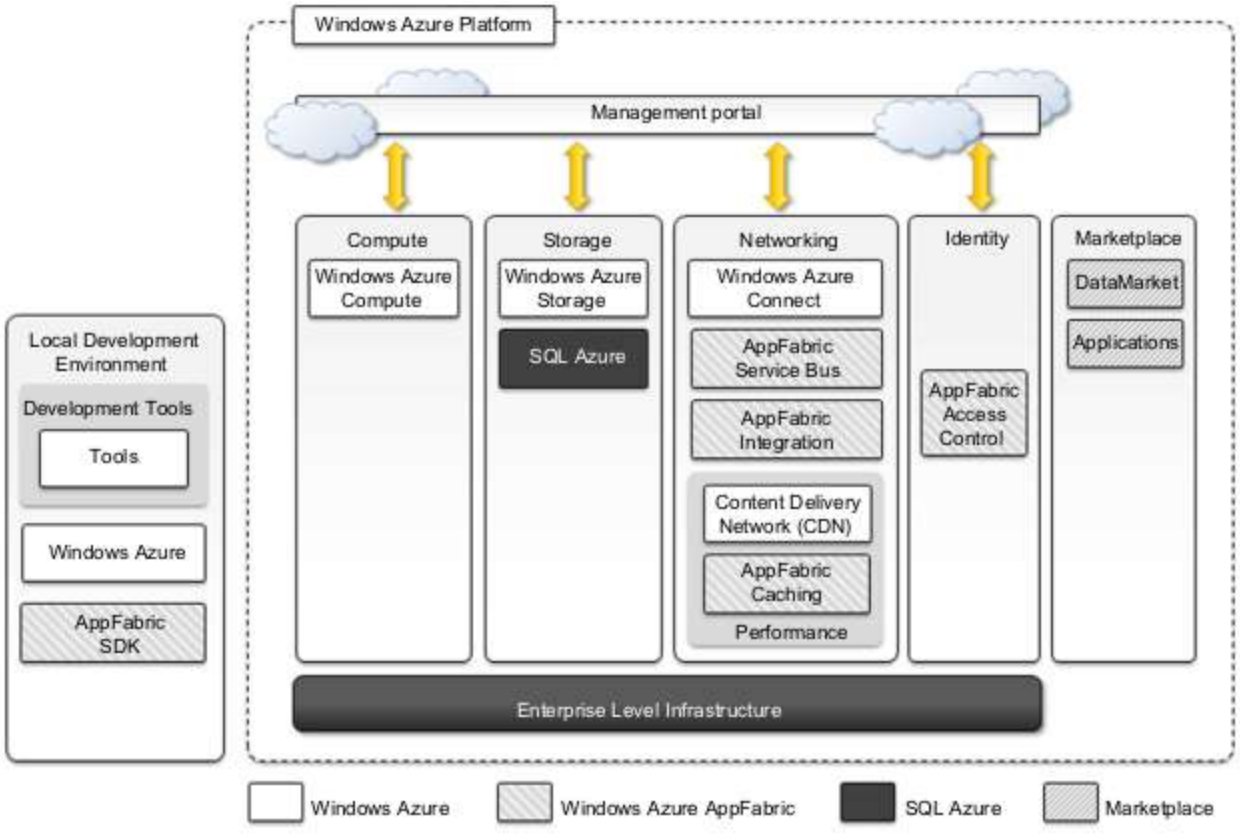


Fig 5.1 Microsoft Azure Platform Architecture

What are the services provided ?

Compute Services

Compute services are the core components of Microsoft Windows Azure, and they are delivered by means of the abstraction of roles. A role is a runtime environment that is customized for a specific computer task. Roles are managed by the Azure operating system and instantiated on demand in order to address surges in application demand. Currently, there are three different roles: Web role, Worker role, and Virtual Machine (VM) role.

Web Role

The Web role is designed to implement scalable Web applications. Web roles represent the units of deployment of Web applications within the Azure infrastructure. They are hosted on the IIS 7 Web Server, which is a component of the infrastructure that supports Azure. When Azure detects peak loads in the request made to a given application, it instantiates multiple Web roles for that application and distributes the load among them by means of a load balancer.

Worker Role

Worker roles are designed to host general compute services on Azure. They can be used to quickly provide compute power or to host services that do not communicate with the external world through HTTP. A common practice for Worker roles is to use them to provide background processing for Web applications developed with Web roles.

Virtual Machine Role

The Virtual Machine role allows developers to fully control the computing stack of their computer service by defining a custom image of the Windows Server 2008 R2 operating system and all the service stack required by their applications. The Virtual Machine role is based on the Windows Hyper-V virtualization technology, which is natively integrated in the Windows server technology at the base of Azure.

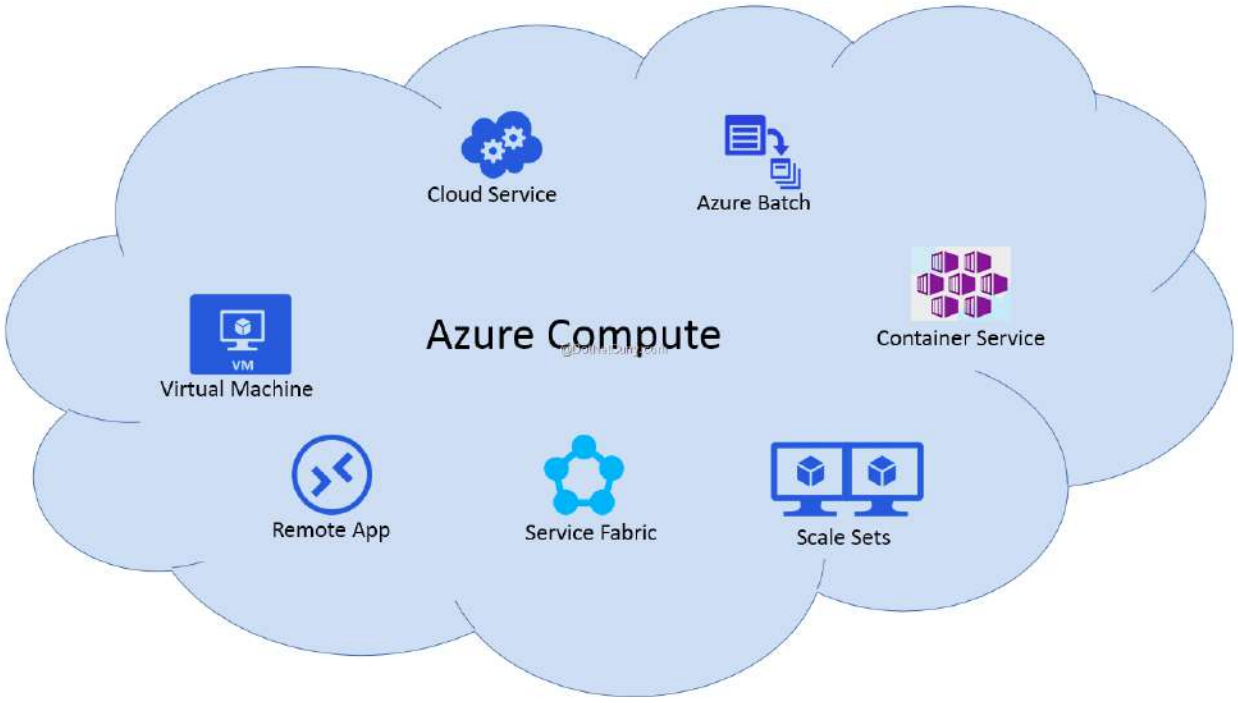


Fig 6.1 Types of Microsoft Azure Compute Services

Storage Services

Compute resources are equipped with local storage in the form of a directory on the local file system that can be used to temporarily store information that is useful for the current execution cycle of a role. If the role is restarted and activated on a different physical machine, this information is lost. Windows Azure provides different types of storage solutions that complement compute services with a more durable and redundant option compared to local storage. Compared to local storage, these services can be accessed by multiple clients at the same time and from everywhere, thus becoming a general solution for storage.

Blobs

Azure allows storing large amounts of data in the form of binary large objects (BLOBs) by means of the blobs service. This service is optimal to store large text or binary files. Two types of blobs are available Block blobs and Page blobs

Azure Drive

Page blobs can be used to store an entire file system in the form of a single Virtual Hard Drive (VHD) file. This can then be mounted as a part of the NTFS file system by Azure compute resources, thus providing persistent and durable storage. A page blob mounted as part of an NTFS tree is called an Azure Drive.

Tables

Tables constitute a semi structured storage solution, allowing users to store information in the form of entities with a collection of properties. Entities are stored as rows in the table and are identified by a key, which also constitutes the unique index built for the table. Users can insert, update, delete, and select a subset of the rows stored in the table. Unlike SQL tables, there are no schema enforcing constraints on the properties of entities and there is no facility for representing relationships among entities. For this reason, tables are more similar to spreadsheets rather than SQL tables.

Queues

Queue storage allows applications to communicate by exchanging messages through durable queues, thus avoiding lost or unprocessed messages. Applications enter messages into a queue, and other applications can read them in a first-in, first-out (FIFO) style.

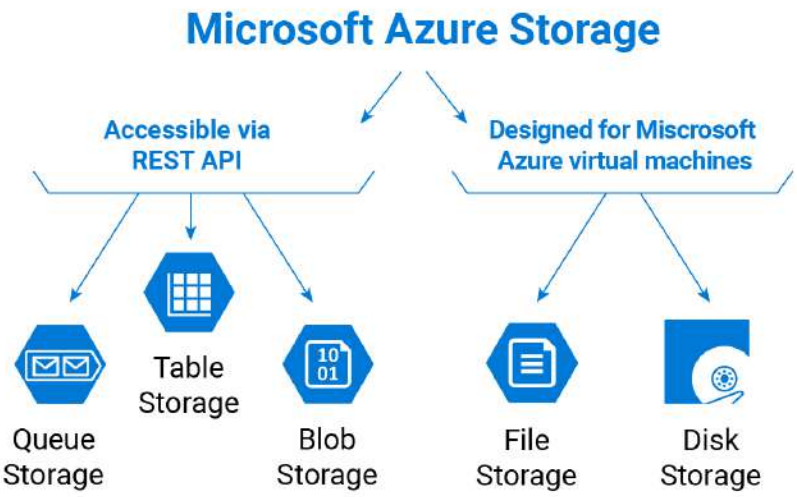


Fig 6.2 Types of Microsoft Azure Storage Services

Core Infrastructure: AppFabric

AppFabric is a comprehensive middleware for developing, deploying, and managing applications on the cloud or for integrating existing applications with cloud services. AppFabric implements an optimized infrastructure supporting scaling out and high availability; sandboxing and multi-tenancy; state management; and dynamic address resolution and routing. On top of this infrastructure, the middleware offers a collection of services that simplify many of the common tasks in a distributed application, such as communication, authentication and authorization, and data access.

Access Control

AppFabric provides the capability of encoding access control to resources in Web applications and services into a set of rules that are expressed outside the application code base. These rules give a great degree of flexibility in terms of the ability to secure components of the application and define access control policies for users and groups.

Service Bus

Service Bus constitutes the messaging and connectivity infrastructure provided with AppFabric for building distributed and disconnected applications in the Azure Cloud and between the private premises and the Azure Cloud. Service Bus allows applications to interact with different protocols and patterns over a reliable communication channel that guarantees delivery.

Azure cache

Windows Azure provides a set of durable storage solutions that allow applications to persist their data. These solutions are based on disk storage, which might constitute a bottleneck for the applications that need to gracefully scale along the clients' requests and dataset size dimensions. Azure Cache is a service that allows developers to quickly access data persisted on Windows Azure storage or in SQL Azure. The service implements a distributed in-memory cache of which the size can be dynamically adjusted by applications according to their needs.

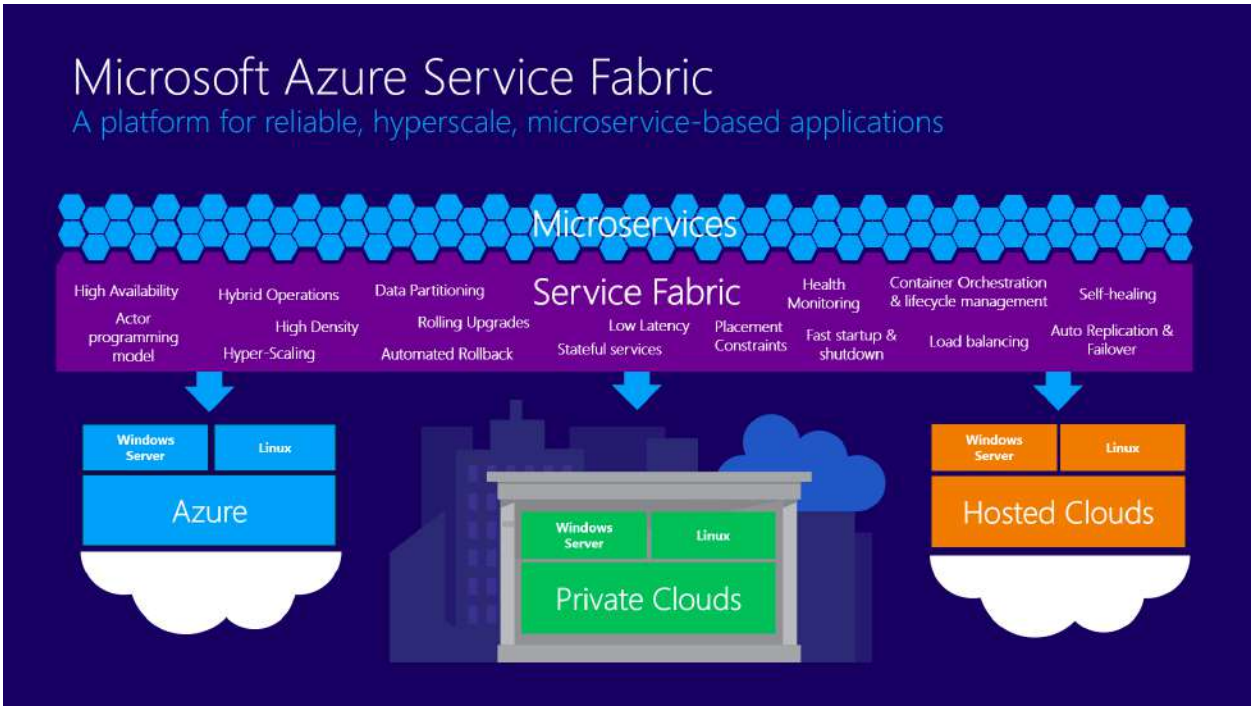


Fig 6.3 Representation of Microsoft Azure Service Fabric

Other Services

Compute, storage, and middleware services constitute the core components of the Windows Azure platform. Besides these, other services and components simplify the development and integration of applications with the Azure Cloud. An important area for these services is application connectivity, including virtual networking and content delivery.

Windows Azure Virtual Network

Networking services for applications are offered under the name Windows Azure Virtual Network, which includes Windows Azure Connect and Windows Azure Traffic Manager. Windows Azure Connect allows easy setup of IP-based network connectivity among machines hosted on the private premises and the roles deployed on the Azure Cloud. This service is particularly useful in the case of VM roles, where machines hosted in the Azure Cloud become part of the private network of the enterprise and can be managed with the same tools used in the private premises.

Windows Azure Content Delivery Network

Windows Azure Content Delivery Network (CDN) is the content delivery network solution that improves the content delivery capabilities of Windows Azure Storage and several other Microsoft services, such as Microsoft Windows Update and Bing maps. The service allows serving of Web objects (images, static HTML, CSS, and scripts) as well as streaming content by using a network of 24 locations distributed across the world.

SQL Azure

SQL Azure is a relational database service hosted on Windows Azure and built on the SQL Server technologies. The service extends the capabilities of SQL Server to the cloud and provides developers with a scalable, highly available, and fault-tolerant relational database. SQL Azure is accessible from either the Windows Azure Cloud or any other location that has access to the Azure Cloud. It is fully compatible with the interface exposed by SQL Server, so applications built for SQL Server can transparently migrate to SQL Azure. Moreover, the service is fully manageable using REST APIs, allowing developers to control databases deployed in the Azure Cloud as well as the firewall rules set up for their accessibility.

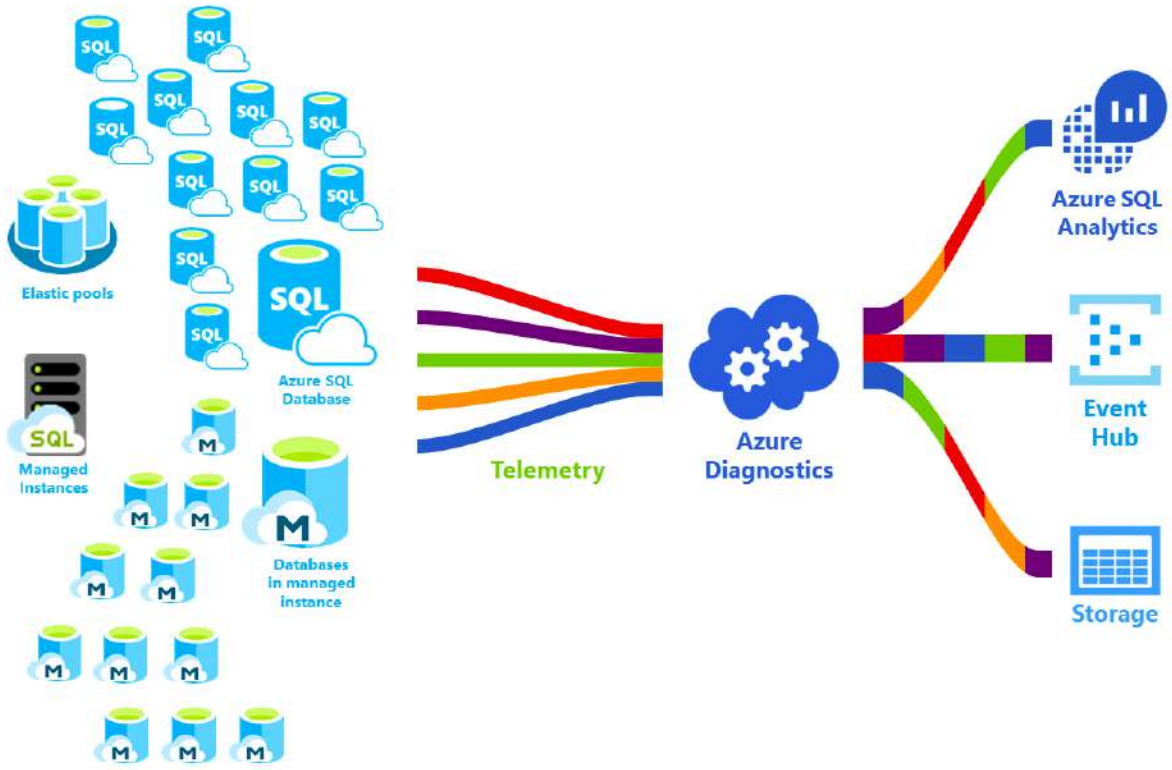


Fig 6.4 Representation of Microsoft Azure SQL

Microsoft Azure Usage and Revenue

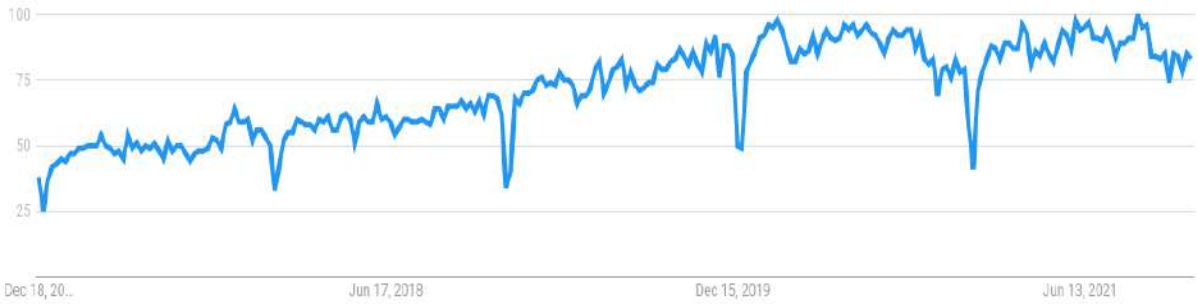


Fig 7.1 Popularity of Microsoft Azure over the last 5 years

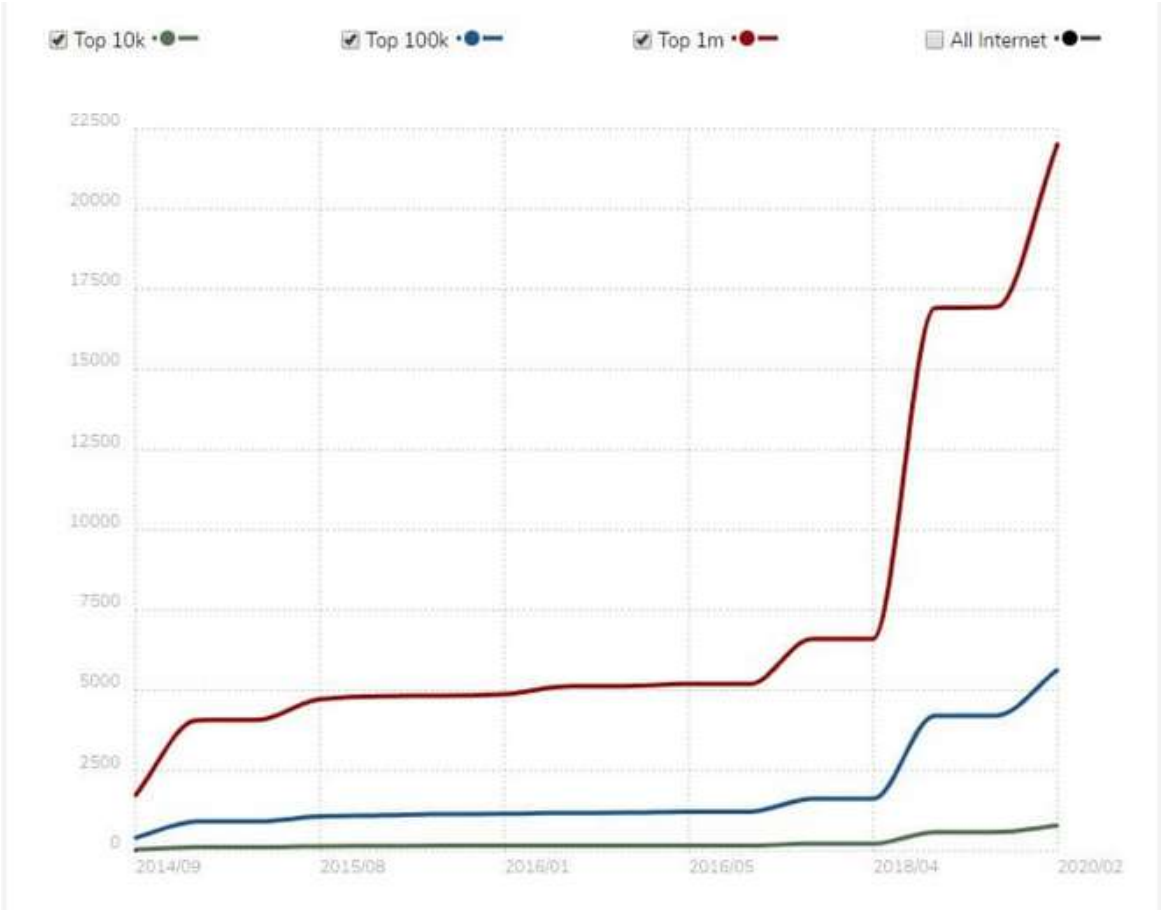


Fig 7.2 Microsoft Azure Usage Statistics

Fig 7.3 Microsoft Azure Revenue over the years

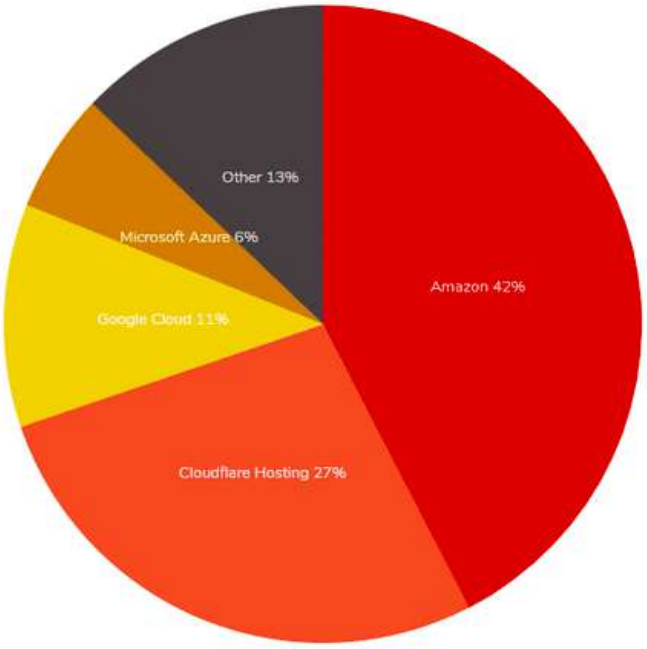


Fig 7.4 Cloud Hosting Usage Distribution

Advantages of Microsoft Azure

High Availability and Uptime

While Microsoft is not as large as AWS, Azure still is the second largest cloud platform in the industry today, with data centers found in several different regions, making it ideal for international businesses. Azure also promises a 99.995% uptime rate — an impressive accomplishment in the Cloud Industry.

Flexibility

Moving to the cloud can be an expensive experience, so it is important for businesses to make the most of their cloud platform once they are there. This flexibility is important, as it will enable you to scale up your projects as your business continues to grow. Azure proves to be an easily scalable platform and barely a few clicks of a button will get you the additional licenses you may need. Imagine being able to scale down your software environment over the weekend or scaling up only for your busiest days of the year.

Security

One of the most appealing features that Azure has to offer its clients is a state of the art security system following a ADADSC approach: Detect, Assess, Diagnose, Stabilize, and Close. They have proven to be the leading force in IaaS security and have received multiple compliance certifications for their high standards. Their security features are both reliable and user-friendly with protections like multi-factor authentication and password requirements.

Disadvantages of Microsoft Azure

Complexity

As a SaaS platform, Azure can easily become an extremely complicated environment for larger companies. Before the cloud, there was an extremely rigorous process when it came to purchasing more licenses, usually in the form of a negotiation or a contract renewal. Many companies do not have any sort of processes to regulate the spending of employees when confronted with their cloud platform. It will require management and strict processes to make sure purchasing is controlled, environments are well managed, and projects are closed after they have reached their conclusion.

Data Transfer

Azure services are all subject to data transfer fees that are often the cause of stacked hidden fees. This is not unique to Azure as all of the large cloud services like AWS and Google do this same gouging of their customer base. This separate fee for in and out data can prove quite costly for large companies, so you should be aware of this to avoid any surprises.

Support

Despite their high-quality products and global reach, Microsoft is not very good at dealing with the sheer volume of their customers and treating each customer as a unique individual. However, as a cloud service provider, that is one thing that Microsoft will have to do on a regular basis as companies run into technical issues and server problems that must be handled quickly.

Conclusion

Microsoft Azure is Microsoft's solution for developing cloud computing applications. Azure is an implementation of the PaaS layer and provides the developer with a collection of services and scalable middleware hosted on Microsoft datacenters that address compute, storage, networking, and identity management needs of applications. The services Azure offers can be used either individually or all together for building both applications that integrate cloud features and elastic computing systems completely hosted in the cloud.

The core components of the platform are composed of compute services, storage services, and middleware. Compute services are based on the abstraction of roles, which identify a sandboxed environment where developers can build their distributed and scalable components. These roles are useful for Web applications, back-end processing, and virtual computing. Storage services include solutions for static and dynamic content, which is organized in the form of tables with fewer constraints than those imposed by the relational model. These and other services are implemented and made available through AppFabric, which constitutes the distributed and scalable middleware of Azure. The platform is mostly based on the .NET technology and Windows systems, even though other technologies and systems can be supported. For this reason, Azure constitutes the solution of choice for migrating to the cloud applications that are already based on the .NET technology.

References

- Mastering CLoud Computing Foundations and Applications Programming by Rajkumar Buyya, Christian Vecchiola, S. Thamar Selvi
- Microsoft Azure Official Website https://azure.microsoft.com/en-us/overview/

Name: Sawan Bhattacharyya

College Roll Number: 706

Exam Roll Number: 2021141290

Supervisor: Dr. Biswajit Biswas

Department: Comp. Sc.

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchase exceeds INR 100000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Customer module : In this module we add a function which will take customer details as input and will output a unique customer number. The customer details along with the newly generated customer number is stored in the database for further use.
    Input: Customer details
    Output: Unique customer number
  - Purchase module : In this module we add a function for updating purchases of a particular customer using his/her customer number. At the time of checkout, the bill is shown to the checkout staff who enters customer number and bill amount as input. After entering, the function finds a record with customer number equal to the entered customer number in the database and after finding entry with the same customer number, the database is updated by adding bill amount to customer transaction attribute.
    Input: Purchase Summary
    Output: Updated Database
  - Promotional Offer module : In this module, on the last day of the year(31st December) after working hours at the supermarket, records in the database are sorted with respect to total transaction.
    A. Ten customers with the highest total transaction are selected, details taken output and are sent a message to their phone number about them winning the surprise gifts.
    B. The records with total transaction greater than 1,00,000 are selected and taken output as they had won a 22 carat gold coin.
    After the declaration of all names who won promotional offers, the total transaction is cleared for all customer numbers in the database for a fresh start in the new year.
    Input: List of customers and their transactions throughout the year
    Output: Names of 10 customers with highest transaction and names of customers with transaction greater than 10 thousand separately.

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

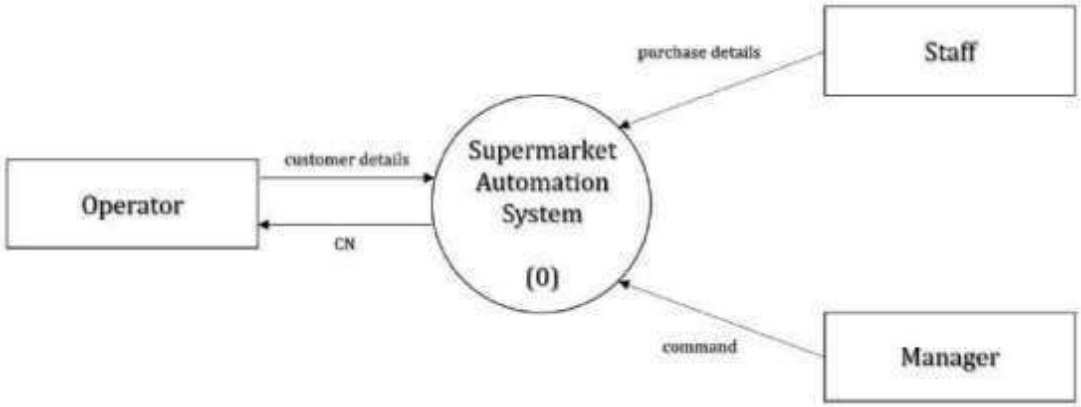
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

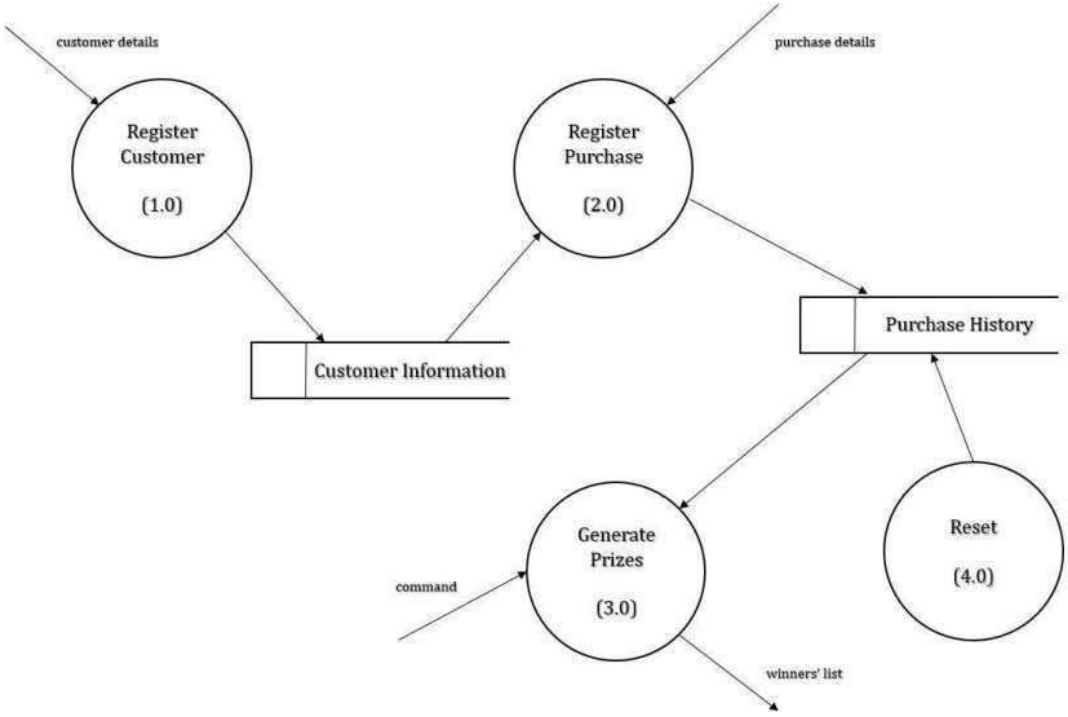
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

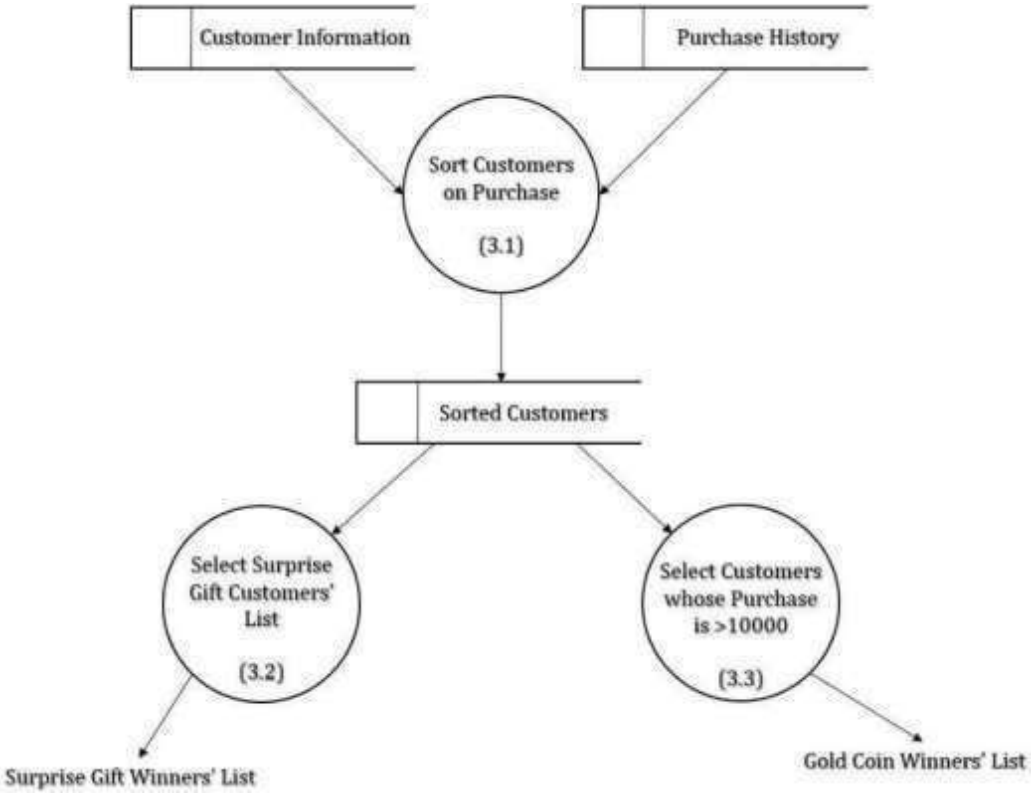
1. Level 0 DFD :



2. Level 1 DFD :



3. Level 2 DFD of process 3.0 :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 2

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

PROBLEM STATEMENT :

It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Incremental Model*** for the following reasons :
    - The fares , roads and buses may change frequently
    - It also requires generating reports after processing o there is a continuous evaluation

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Ticket Booking module : In this module, a customer can book a bus ticket which is mostly preferable to him/her. He can also choose the payment as online or offline (on the counter). The confirmation system will generate the ticket.
    Input: Preferable booking date, time, destination of the customer
    Output: Available bus lists according to the details, Bus Ticket
  - Customer module : In this module, the system will maintain a profile of a customer. At first, customers must register into the system with necessary details. System will maintain a list of previous bookings, current plans.
    Input: Personal details of customer
    Output: An organized profile of the customer, list of previously booked ticket, current bookings
  - Routing Scheduling : In this module, the system will maintain the details of a bus's route. It will handle a database which includes bus number, bus type, route, destination, schedules and driver's name such as if a customer gives his/her travelling details, the system will show the result according to the routing schedule details._
    Input: Details of bus with route and schedules
    Output: Organized database of buses

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

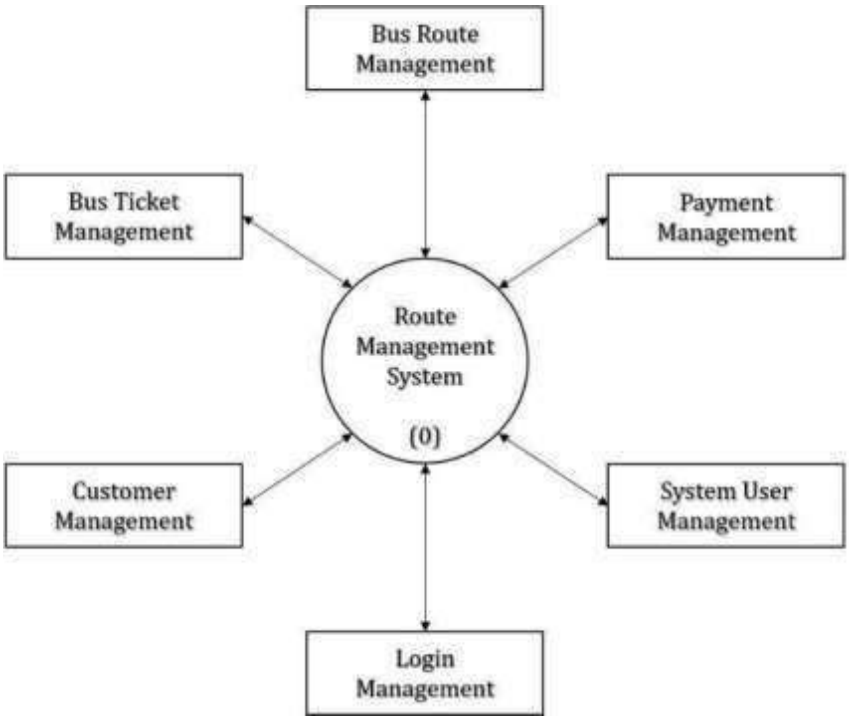
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

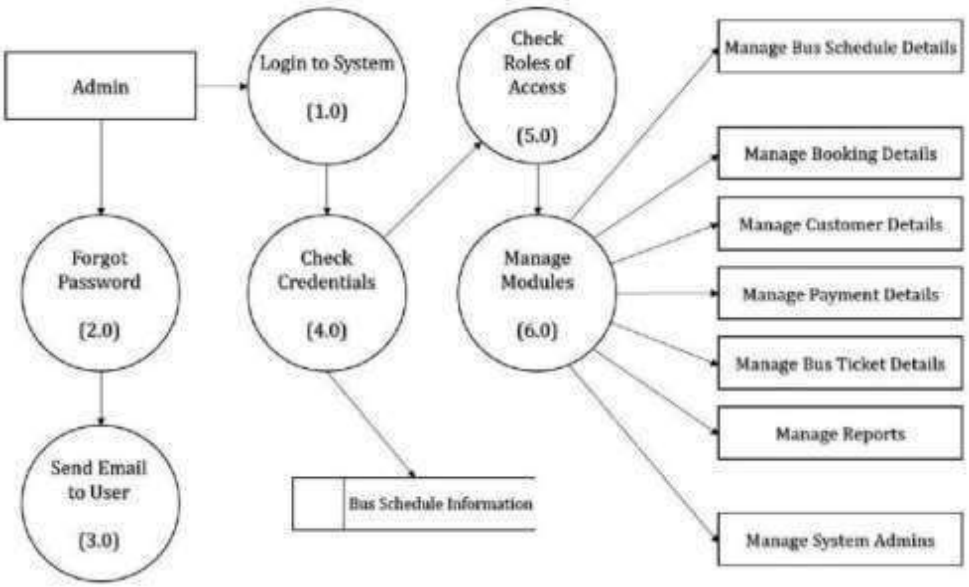
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Incremental Model*** for the following reasons :
    - The number of patients, doctors, diseases and medicines keeps on increasing and updating.
    - The system of the hospital gets updated frequently

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin module : In this module, admin has to give the personal details to register into the system at the very first time. System will create an account and will give the user id and password. System will also set the permission of access. For the next time, the admin can log in into the system and manage other details of a hospital or chamber according to his permission.

    Input: Details of Admin

    Output: User id, Password and Permission of Access details
  - Doctor's module : In this module, the system will maintain the details of the doctor like doctor's schedule, doctor to visit which patient, doctor's fees, replacement of a doctor, availability of a doctor etc.

    Input: Doctor's name

    Output: Doctor's Schedule, Doctor's fees
  - Patient module : In this module, the system will maintain the details of a patient. After admission, the admin continuously updates the details about his credential. At the time of discharge, the system will generate the bill.

    Input: Patient's details, medicine cost, doctor's fees, nursing fees

    Output: Entire Bill to be paid
  - Diagnostic module : In this module, the system will maintain the status of diagnosis of a patient. It will handle which doctor is visiting which patient. Patient is taking which medicines. What is the current health condition of the patient? After every update by the admin, the system will analyze. The system will also suggest medicine by taking information from past diagnostics.

    Input: Diagnostic details, Patient's disease

    Output: Status of the patient

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

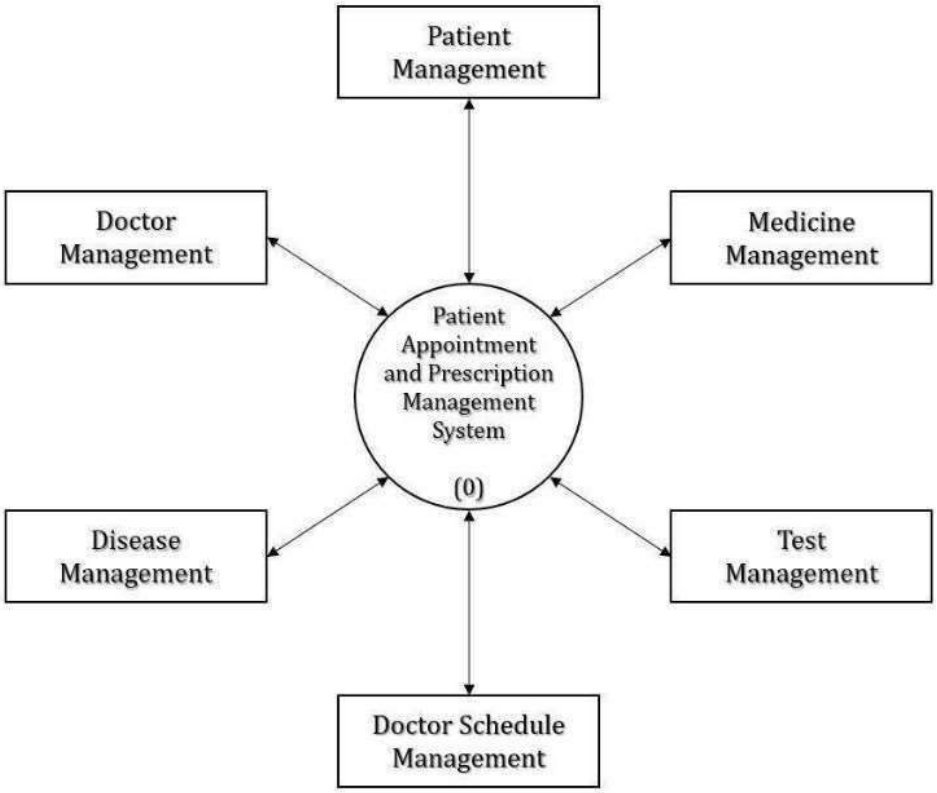
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

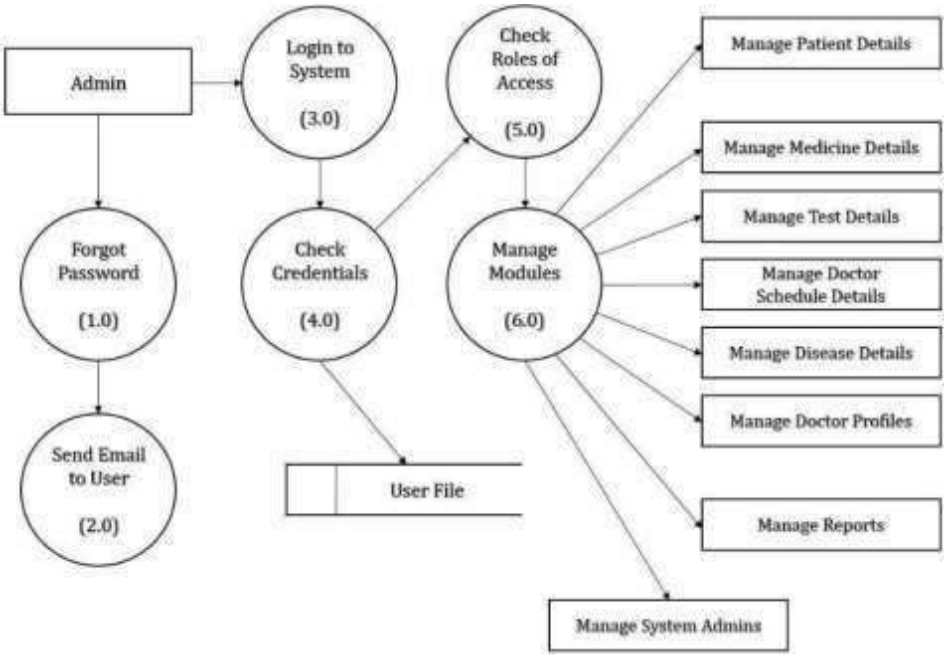
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

HOTEL AUTOMATION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on this spot depending on the availability of rooms. Operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay,type room required. The computer should register these data and depending on the availability of suitable room, computer should provisionally allot room number to the guest and assign a unique token number to them. If the guests cannot be accommodating, the computer generates an apology message. The hotel catering services manager would input the quantity & type of food items as and when consumed by the guests along with the token number of the guests with date & time. You can assume that different food items had different code numbers and hence the catering service manager did not enter the full name of the food item. When a customer picks up bags to check out, the hotel automation software should generate the entire bill for the customer's balance amount payable by him.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The requirements can only be fulfilled by a continuous evaluation
    - New rooms, types gets added also more guests arrive time to time

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Guests module : This module takes the guest's personal information and knows whether s/he wants online allocation or spot booking.
    Input: Guest's personal information
    Output: Token Number
  - Staying Allocation module : This module takes the data of the guest about his lodging dates and allocates room (if available) or leaves a apology message (if not available)
    Input: Guest Information of lodging times
    Output: Allocated free room or Apology Message
  - Catering module : This module takes the food order of the guests according to its token number and gives a total food bill along with the grand bill of the hotel.
    Input: Token number, Food Statistics, Food items
    Output: Generate the food bill
  - Bill Generation module : This module generates the whole bill of the guest according to the token number.
    Input: Token number, Food item bills, lodging bill
    Output: The grand bill

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

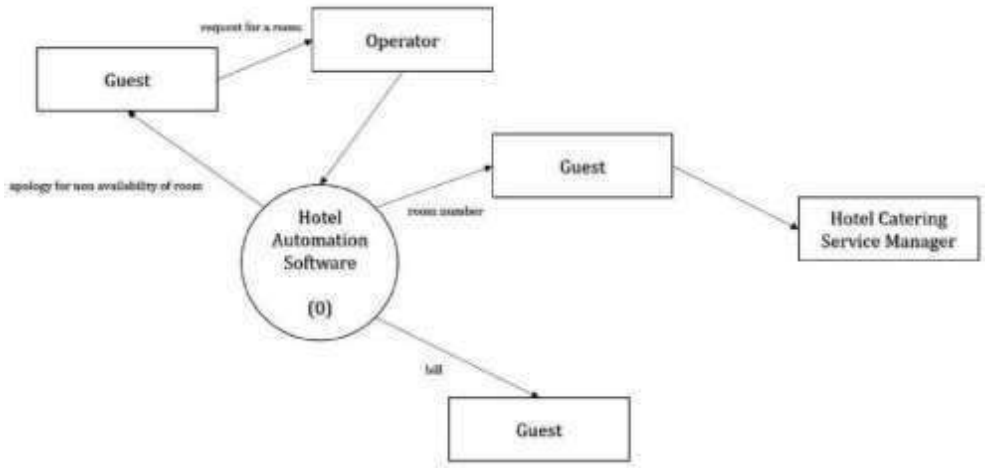
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R)processor, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

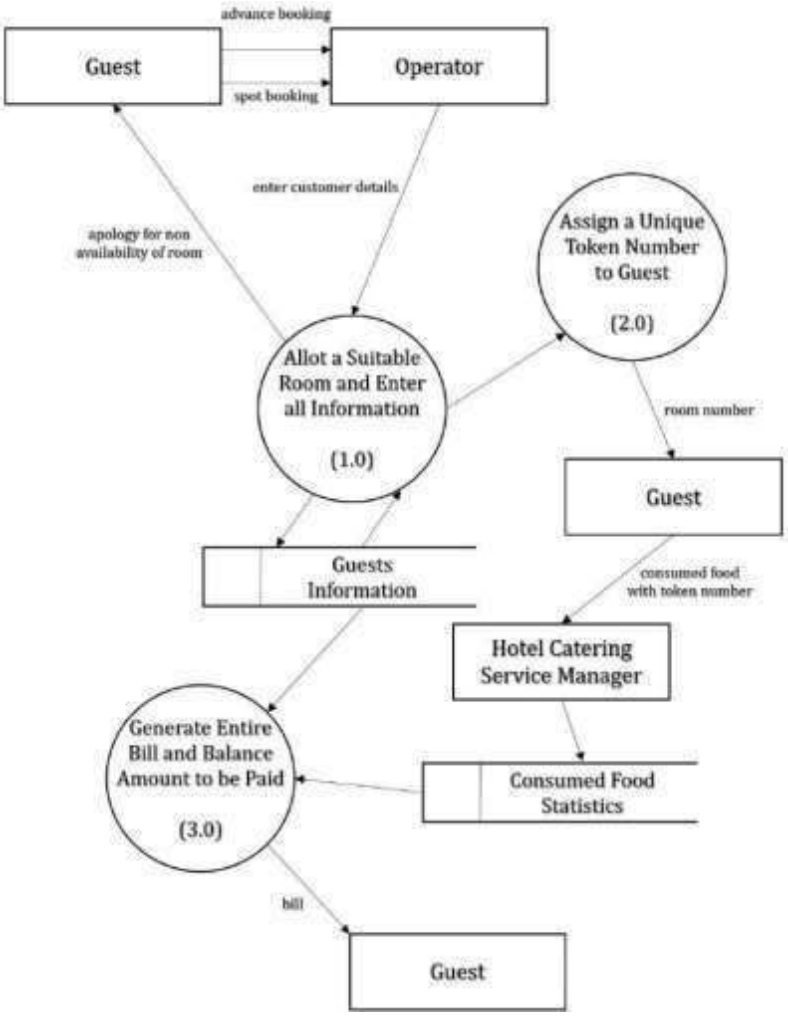
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Crime file management system identifies internal data
stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The rules don't change that frequently

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin module : In this module admin has to register into the system by giving the self information. After the registration, admin can login to the system to handle the crime file details. If the admin forgot the password, then the system will provide a system to change the password.
    Input: Personal Information of Admin to Login
    Output: Access Permission
  - Crime module : In this module admin adds the whole details about the crime of a criminal which includes crime records, complaint records, FIR records, Charge sheet records etc. System will produce organized details about the criminal.
    Input: Individual Crime Details
    Output: Organized Crime Details Report
  - Report module : This module generates the file report for the court according to the codes by using the Crime Details records.
    Input: Crime Details of Criminal
    Output: Court File Report

- Non-Functional Requirements :
  - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

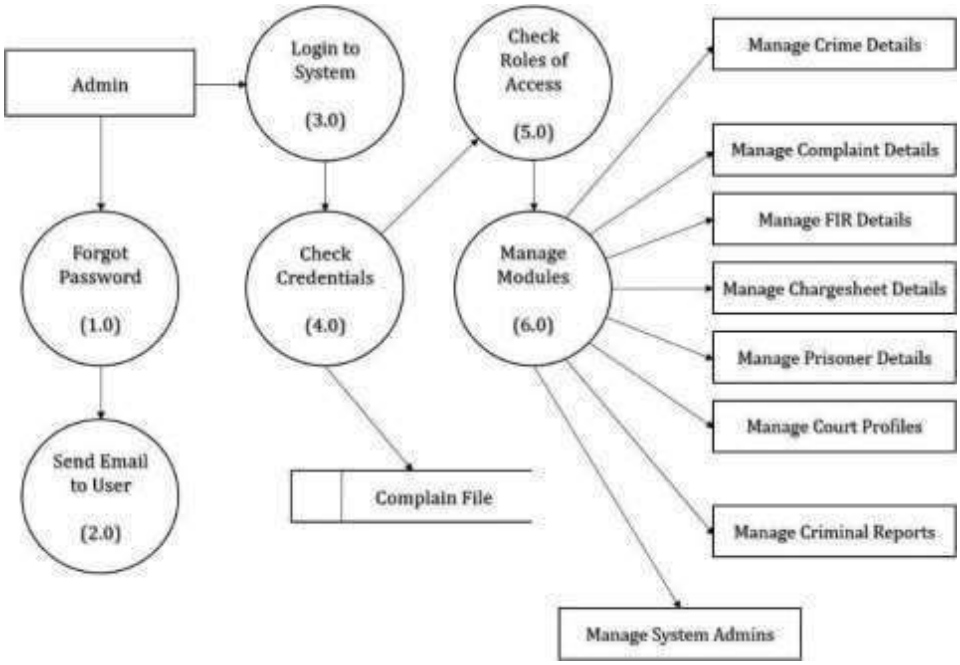
- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :

2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SYSTEM

PROBLEM STATEMENT :

Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are - processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate reports of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The students gets evaluated on yearly basis
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
  - Admin Registration module : Admin or the controller of examination has to register into the system by giving the personal details with the institutional details. After registration, the system will create a profile and provide a login name and password to update the time table and other details. There must be on option for "Forgot Password" to regenerate the password if the admin forgot the password.
    Input: Details of Admin
    Output: New Profile with login id and password
  - Time Table Management : In this part of the system, Admin creates the time table of a particular class based on the subject, department, availability of faculty etc. Admin has to give detailed information and the system will handle the flow of the time table with classes and departments.
    Input: Details of class, department, branch, available faculty
    Output: Time Table of each class
  - Course and Student Database module : In this module, Admin adds or updates the data of student and course details. Submitting all the details of student and course details by the admin, the system manages the database of student and corresponding course details.
    Input: Details of students and courses
    Output: Database of students, courses, classes
  - Examination Computation module : In this part of the system, admin submits the examination details of the student. After submission, the admin can update the details also. It will compute the examination marks of the students and create the list for each student with their marks.
    Input: Exam details of students
    Output: Marksheet of students

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

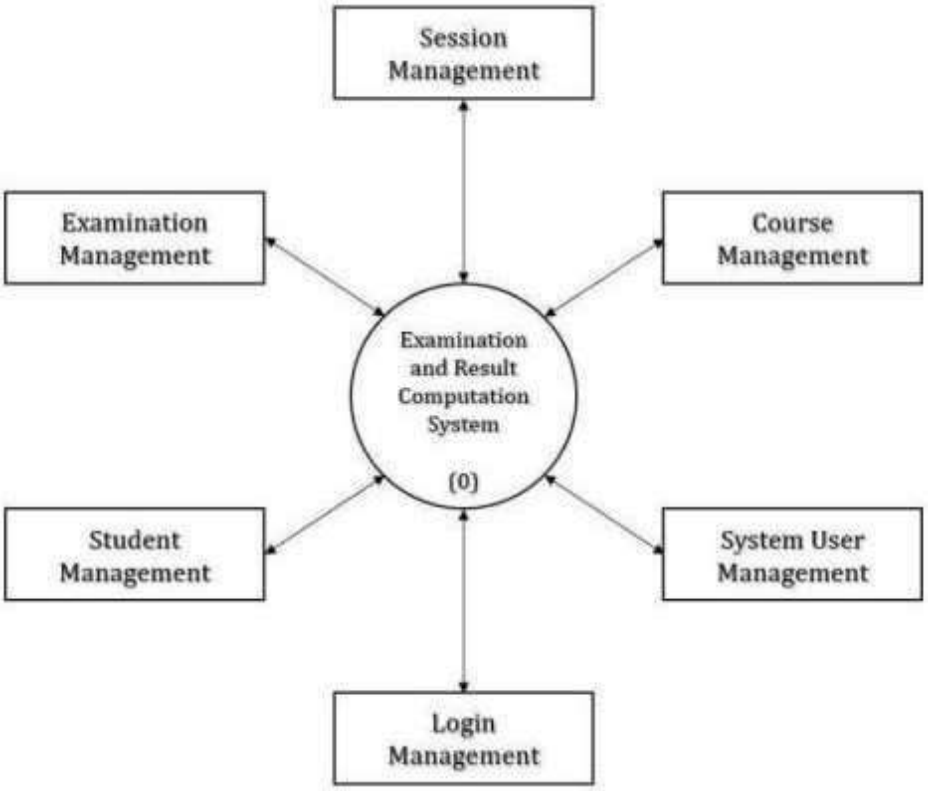
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

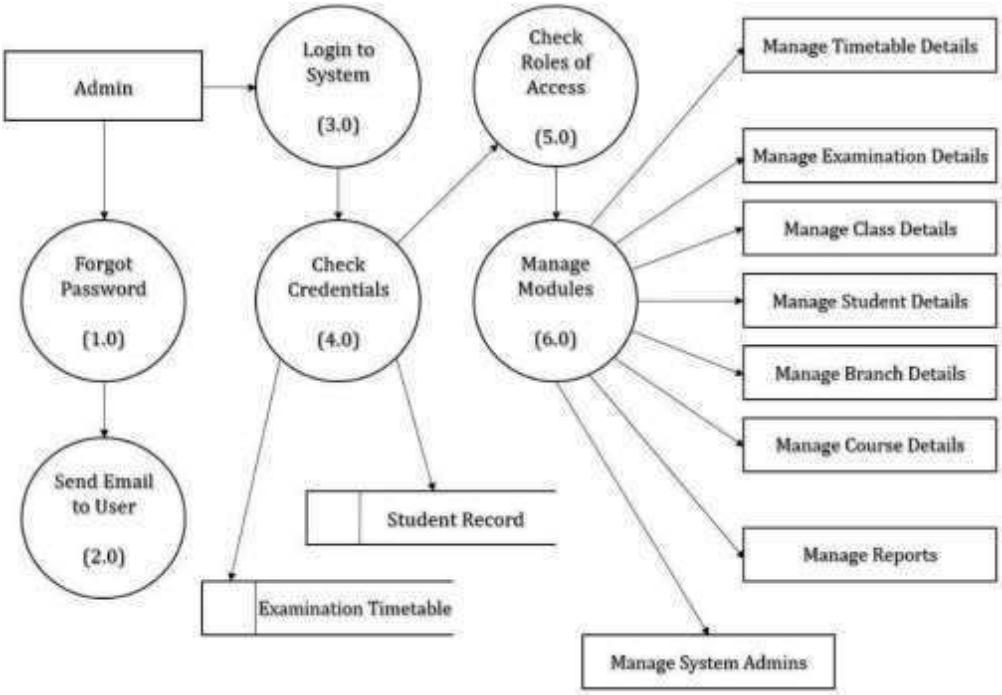
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 7

PARKING ALLOCATION MANAGEMENT SYSTEM

PROBLEM STATEMENT :

Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionality of parking system - processing of parking slots, vehicle records, parking fees, duration and generates the report of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Prototyping*** for the following reasons :
    - The requirements are not clearly mentioned.
    - The requirements may change on a yearly basis depending on the results.

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
    - Car Details module : In this module, the user will store the every details of a parked car such as parked car number, car type, lobbies. Other than that, it will also calculate the parking fees according to the total time of parking.
      Input: Details of Car
      Output: Database of the parked car, parking time and fees
    - Bill Generation module : In this module, it generates the bill when a customer leaves the parking slot. It generates the parking fees from the data stored in the database.
      Input: The database of the parked car
      Output: The bill to be paid

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

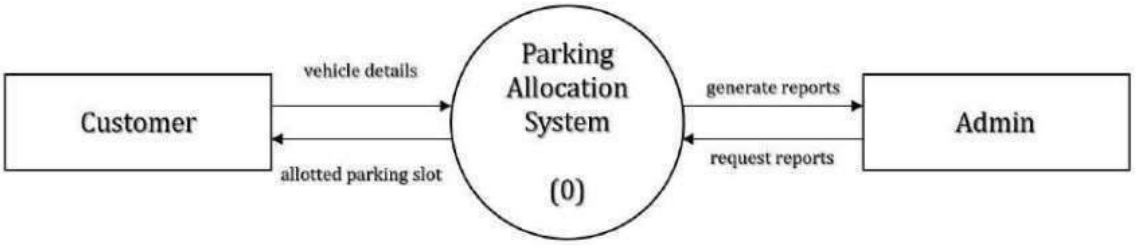
Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Feasibility Study	T	14/04/2021	19/04/2021	4 days
Requirement Analysis and Specifications	T	20/04/2021	27/04/2021	6 days
Design	T	28/04/2021	07/05/2021	8 days
Coding and Unit Testing	T	04/05/2021	22/06/2021	36 days
Integration and System Testing	T	15/06/2021	01/07/2021	13 days
Maintenance	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

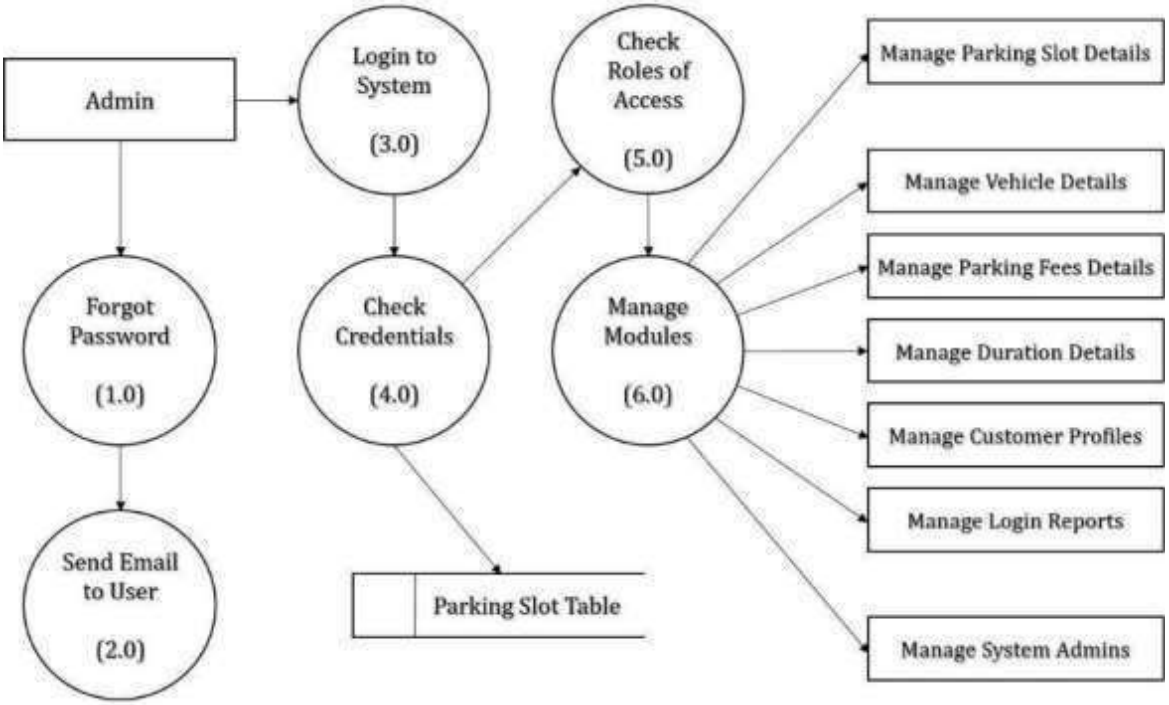
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 8

WHOLESALE MANAGEMENT SYSTEM

PROBLEM STATEMENT :

It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use ***Spiral Model*** for the following reasons :
    - The requirements may increase or change in the future
    - The system needs an continuous evaluation

REQUIREMENT ANALYSIS (SRS) :

- Functional requirements :
    - Customer module : In this module, a customer is registered and the profile is maintained.
      Input: Customer details
      Output: Unique profile
    - Stock module : In this module, the details of stocks are maintained. A database is maintained for keeping the stock information._
      Input: Stock Details
      Output: Updated Database
    - Bill module : In this module, all the bills are maintained. A database is maintained for keeping the bill information.
      Input: Customer Details, Stock Details
      Output: Generated Bills, Updated Database
    - Report module : In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.
      Input: Store, product, retail, bill, customer details
      Output: Generated Reports

- Non-Functional Requirements :
    - Usability : The software should have a user-friendly interface for ease of access. It ensures more customers get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that it uses memory efficiently and takes optimal storage for being portable.
    - After every purchase, a transaction message is to be sent to the mobile number of the customer, it will ensure the customer that his data is stored at the right location.

PROJECT MANAGEMENT :

- Project Scheduling :

Title	T/M	Start	End	
Problem Statement	T	12/04/2021	13/04/2021	2 days
Customer Communication	T	14/04/2021	19/04/2021	4 days
Planning	T	20/04/2021	27/04/2021	6 days
Risk Analysis	T	28/04/2021	07/05/2021	8 days
Engineering	T	04/05/2021	22/06/2021	36 days
Construction and Release	T	15/06/2021	01/07/2021	13 days
Customer Evaluation	T	02/07/2021	16/07/2021	11 days
Documentation and User Manual	T	11/05/2021	23/07/2021	54 days

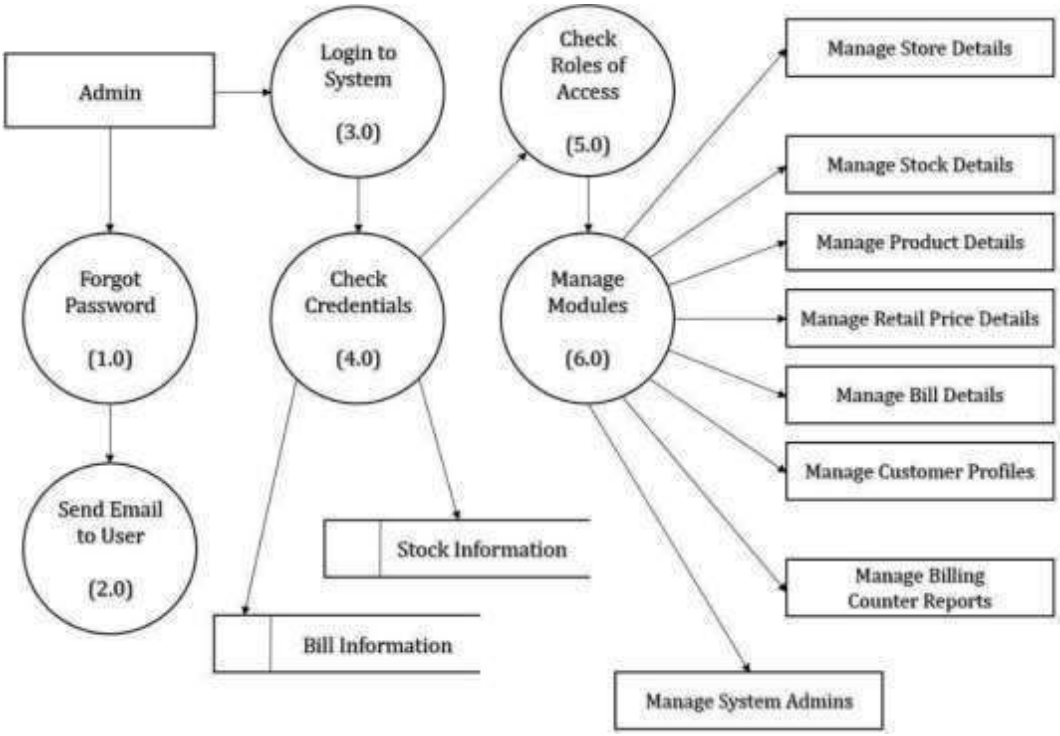
- Project Resources :
  - Human Resources : Project Manager, System Analyst, Developer, Designer, Tester, Document Creator
  - Software Resources : MS Word, MS Project, C, SQL
  - Hardware Resources : Intel(R)Pentium(R), 3.10 GHz, 6 GB RAM
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :

2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.
  This time this testing is done in three parts.
  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Abhishek Dutta
Roll Number: 707
Exam roll number: 2021141291
Registration number: A01-1112-117- 006-2019
Semester: V

Department: Computer Science

Supervisor: Prof. Manas Pal

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a “Customer Number” (CN). A customer can present ‘CN’ to customer staff when he makes purchases. In this case, the value of his purchase is credited against its ‘CN’. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

REQUIREMENT ANALYSIS (SRS) :

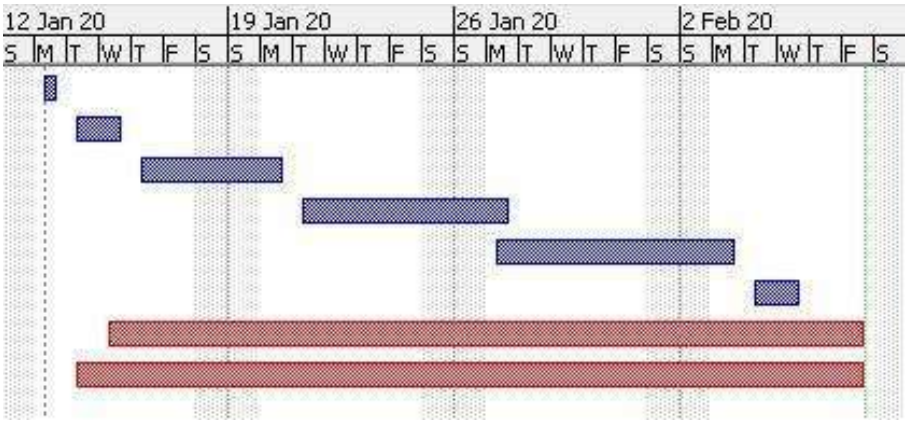
As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account’s purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh

the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



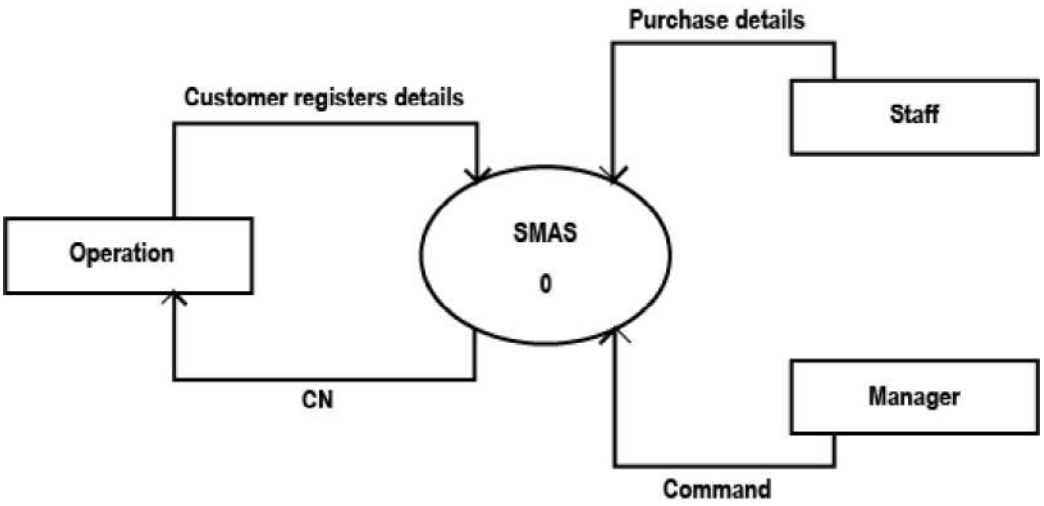
- Project Resources :
  - Hardware Resources :Several computer machines.
  - Human Resources :Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan :Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.

○ Risk leverage =

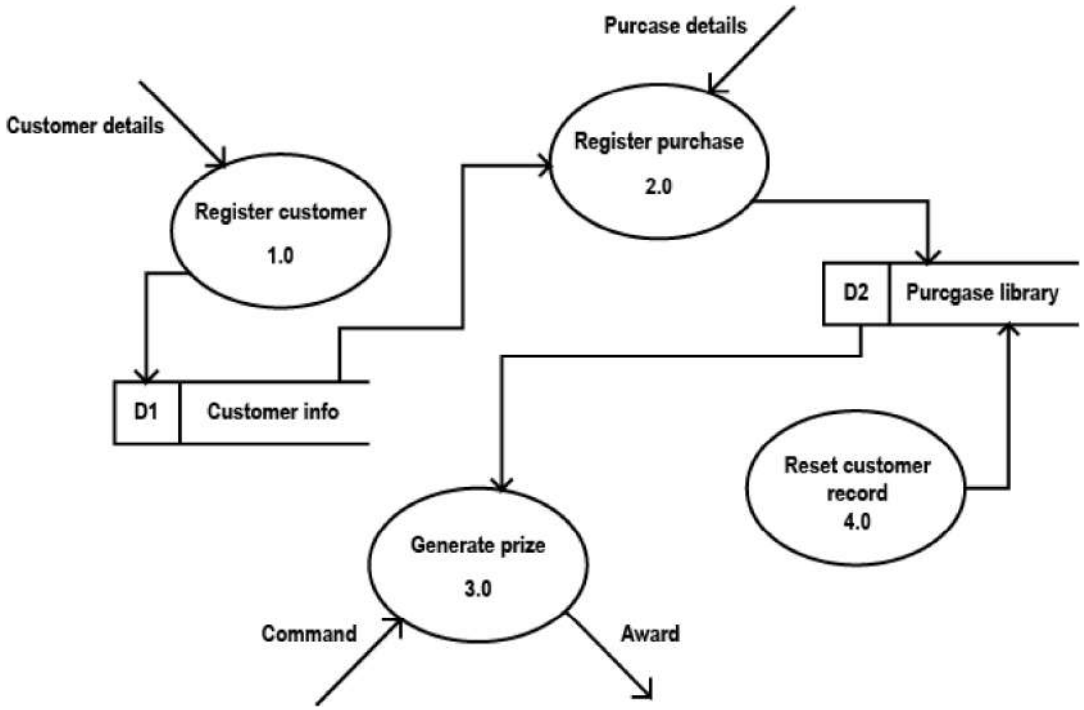
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

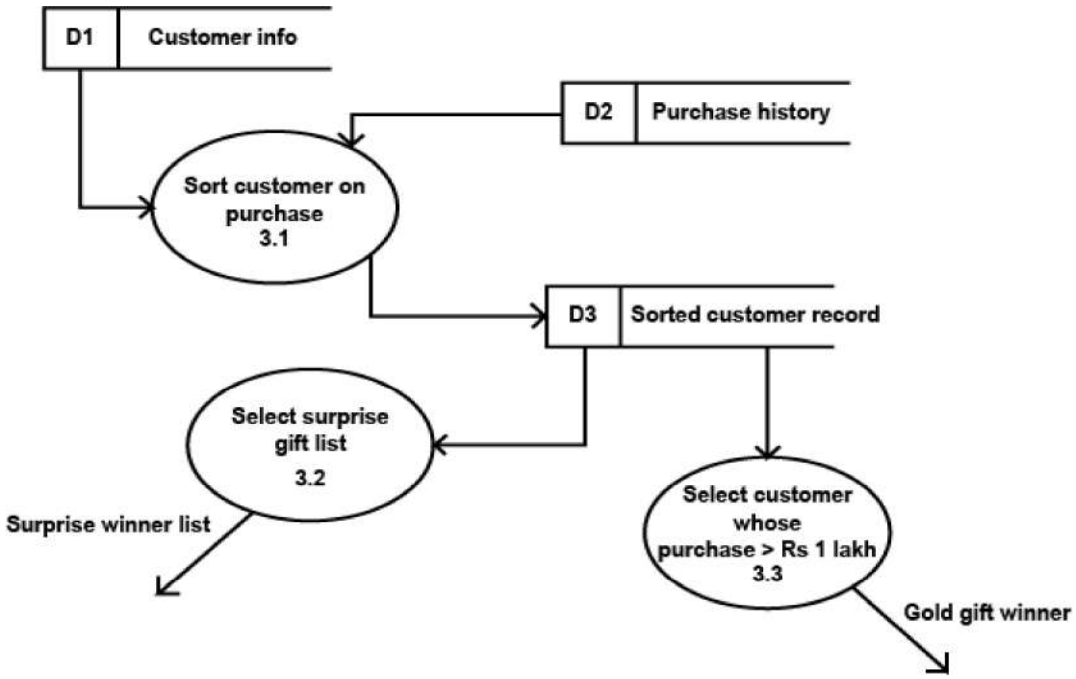
1. Level 0 DFD :



3. Level 1 DFD :



4. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

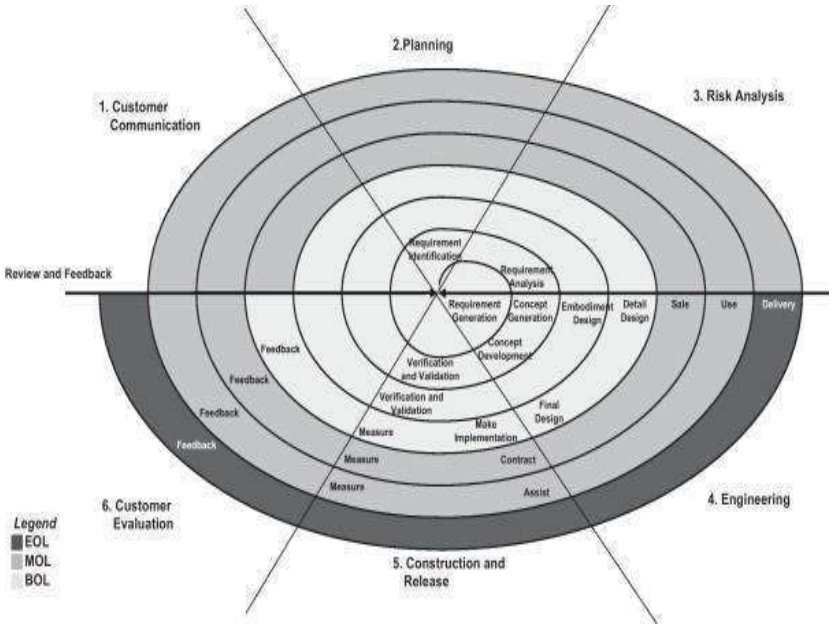
PROJECT 2

Route Management System

It identifies internal datastores of payment, bus route,

customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule.It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔**Software Requirements Specifications**

1. ***Goal of the Project:***

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. ***Functional Requirements:***

As per the requirements of the software from client end. By analysis we realise that we need five modules: A.

Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Customer Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. Schedule Module:

This module maintains timings of the bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

---

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

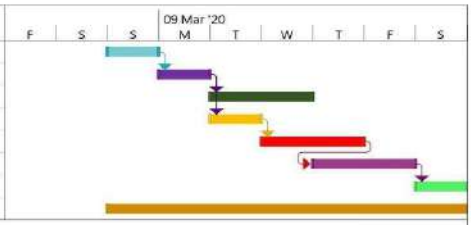
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 08-03-20	Sun 08-03-20	
2	Customer Communication	1 day	Mon 09-03-20	Mon 09-03-20	1
3	Planning	2 days	Tue 10-03-20	Wed 11-03-20	2
4	Risk Analysis	1 day	Tue 10-03-20	Tue 10-03-20	2
5	Engineering	2 days	Wed 11-03-20	Thu 12-03-20	4
6	Construction and Release	2 days	Thu 12-03-20	Fri 13-03-20	5
7	Customer Evaluation	1 day	Sat 14-03-20	Sat 14-03-20	6
8	Write User Manual	7 days	Sun 08-03-20	Sat 14-03-20	



Project: Project8
Date: Sat 14-03-20

Task, Split, Milestone, Summary, Project Summary, Inactive Task, Inactive Milestone
Inactive Summary, Manual Task, Duration-only, Manual Summary Rollup, Manual Summary, Start-only, Finish-only
External Tasks, External Milestone, Deadline, Progress, Manual Progress

b. Project Resources

a. **ManPower Resources:**
   i. **Project Manager**
   ii. **System Analyst**
   iii. **Designer** iv. **Developer** v. **Tester**
vi. **Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above**
   ii. Ram: 4Gb or above
   iii. **Java Development Kit -**
   13 iv. **Java Runtime**
   Environment v. **Sublime**
   Text 3 (Text Editor)
   vi. **Ms - Project**
   vii. **Creately** viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. **Likelihood of a risk coming through**
B. **Consequence of the problem associated with that risk**

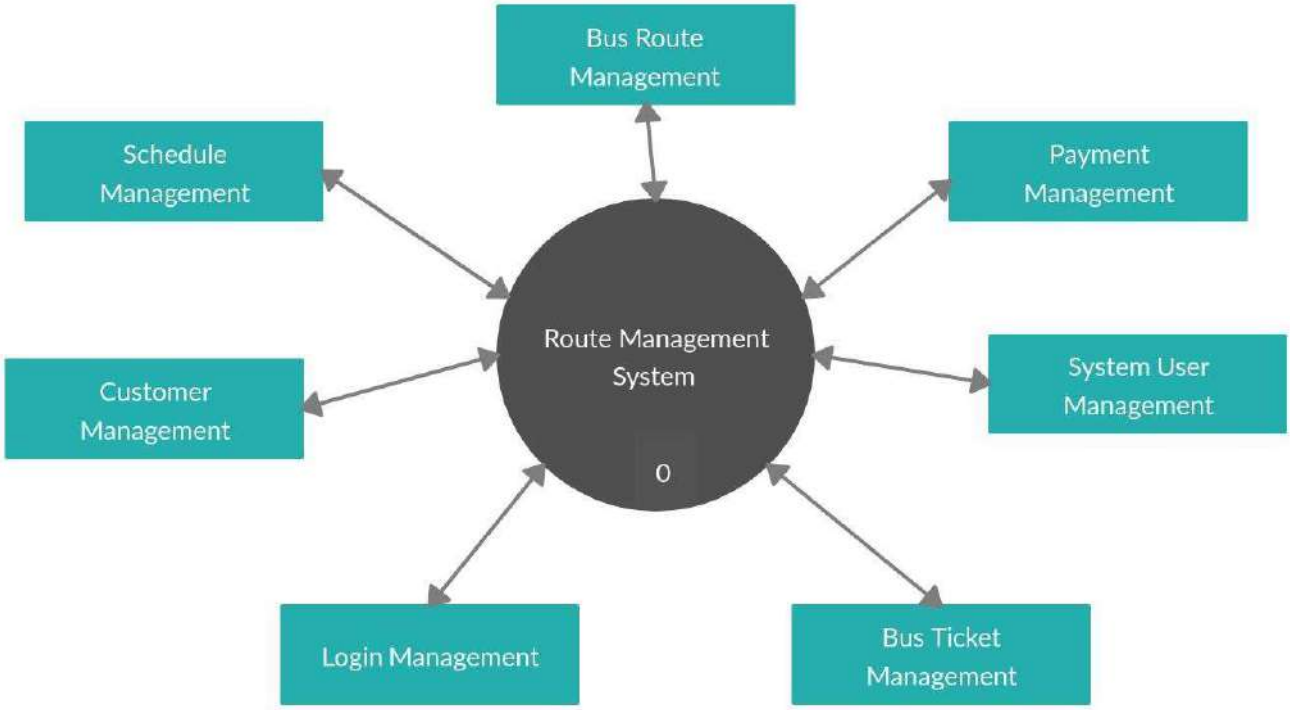
Priority of each risk is multiplication of above two

Risk Leverage =

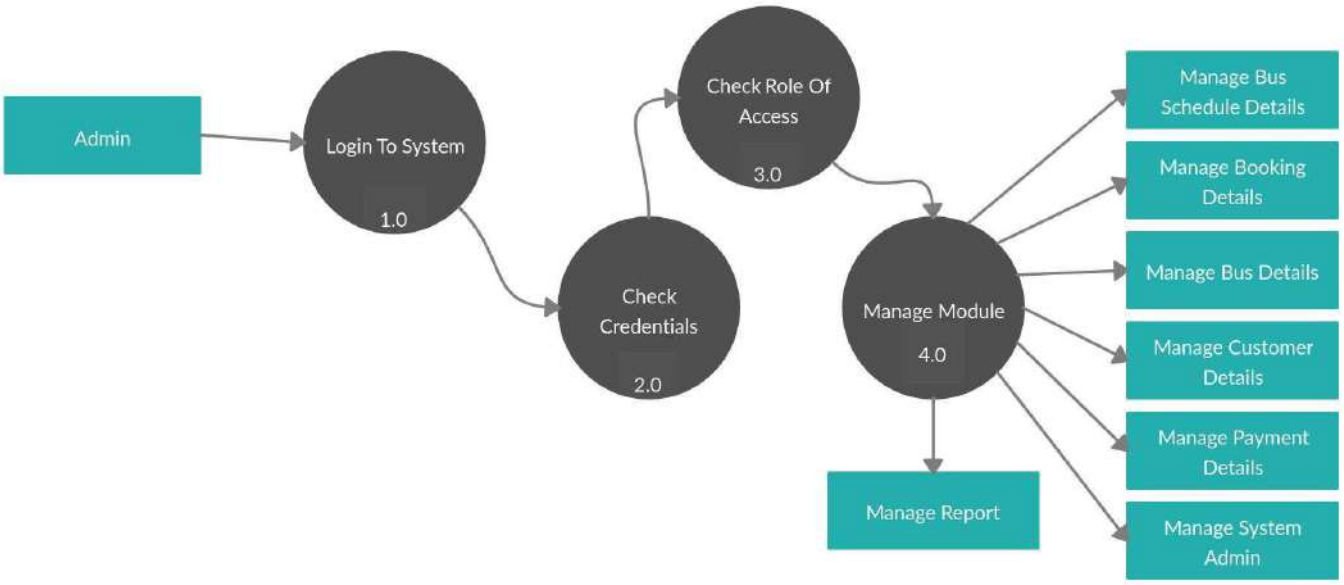
$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduct}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. **Architectural Design**



2. **Component Level Design**
   - **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing: Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance
4. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project :The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

- Functional requirements :The three modules needed are.
    - Administrator module :This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

    ○ Doctor module :This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.

    ○ Report module :This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.

    ○ Diagnostic module :This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.
- Non-Functional Requirements :
    - Usability :The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.

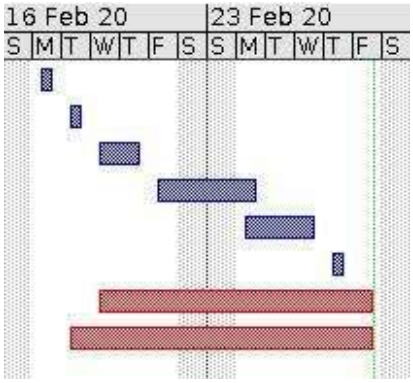
    ○ Portability :The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

    ○ Robustness :The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



- Project Resources :
  - Hardware Resources :Several computer machines.
  - Human Resources :Manpower resources like project manager, designer, analysist, programmer, tester etc.

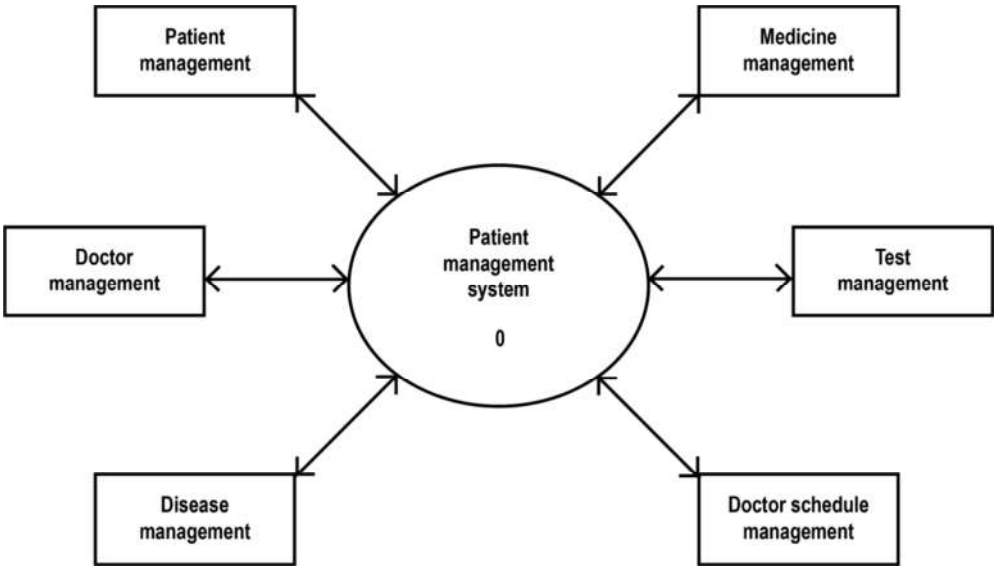
- Risk management plan :Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

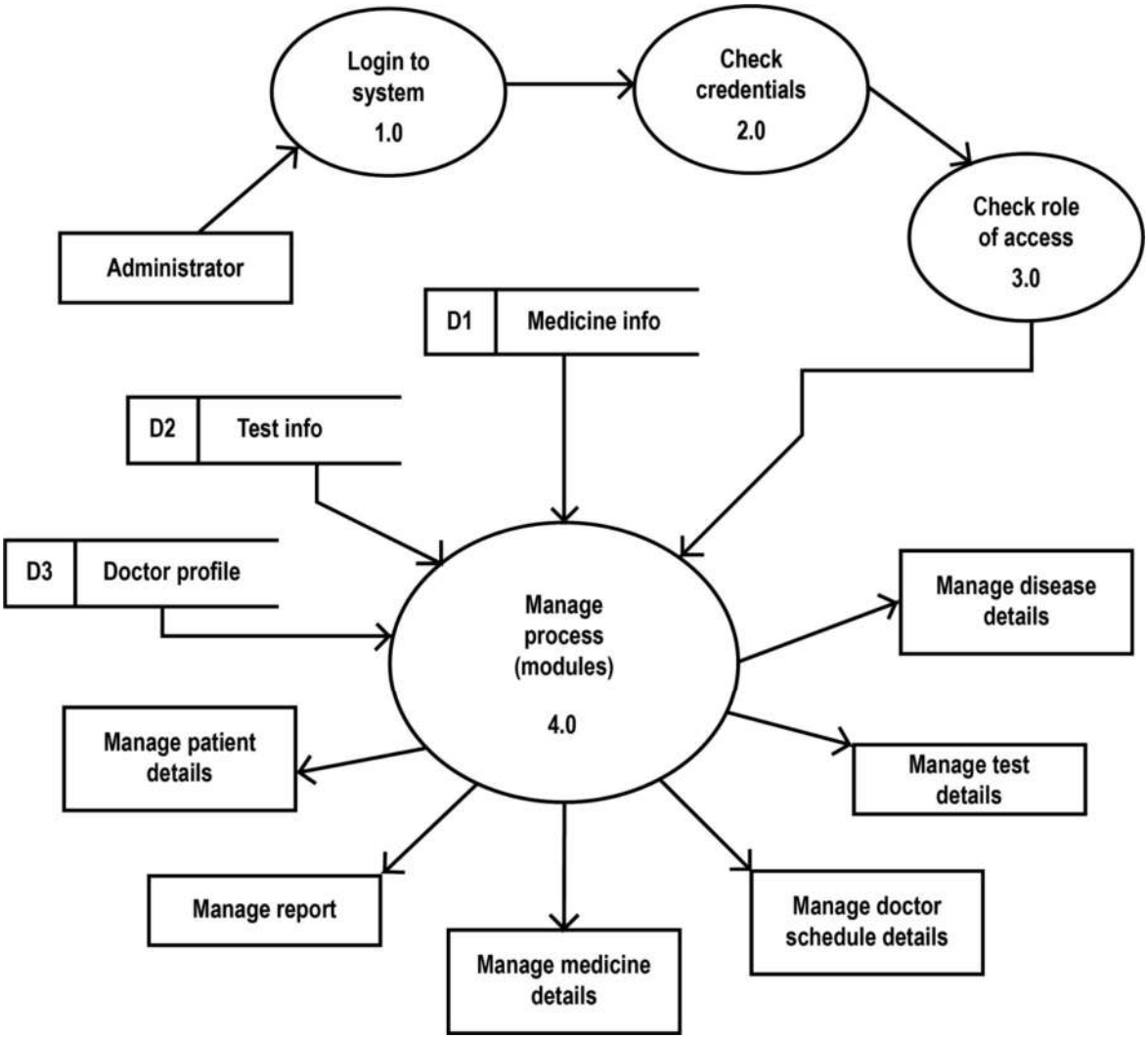
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing :Each module is tested in isolation.
- System Testing :The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
        - Beta testing (Validation) :It is done by a special group of friendly customers in an uncontrolled environment.

- Acceptance testing :It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. Guest module :In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. Stay module :In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

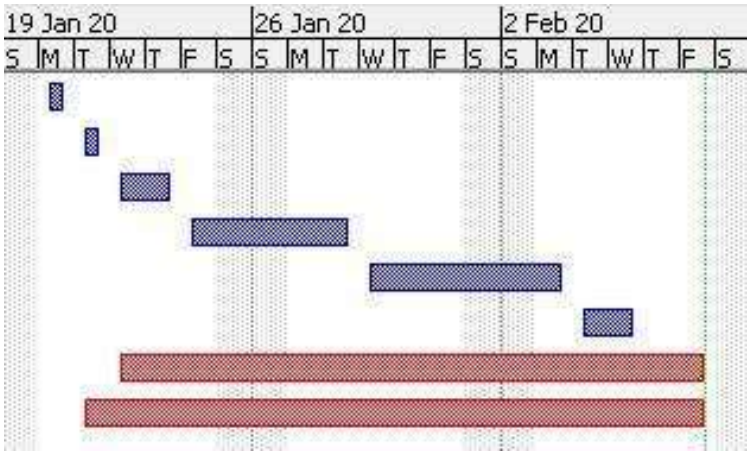
3. Catering module :In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.

4. Bill generation module :At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2		Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3		Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4		Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5		Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



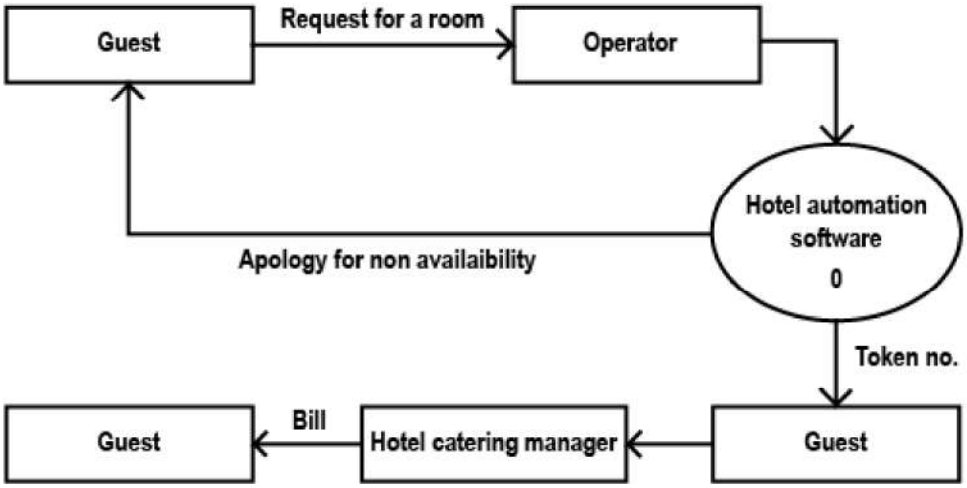
- Project Resources :
    - Hardware Resources :Several computer machines.
    - Human Resources :Manpower resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan :Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

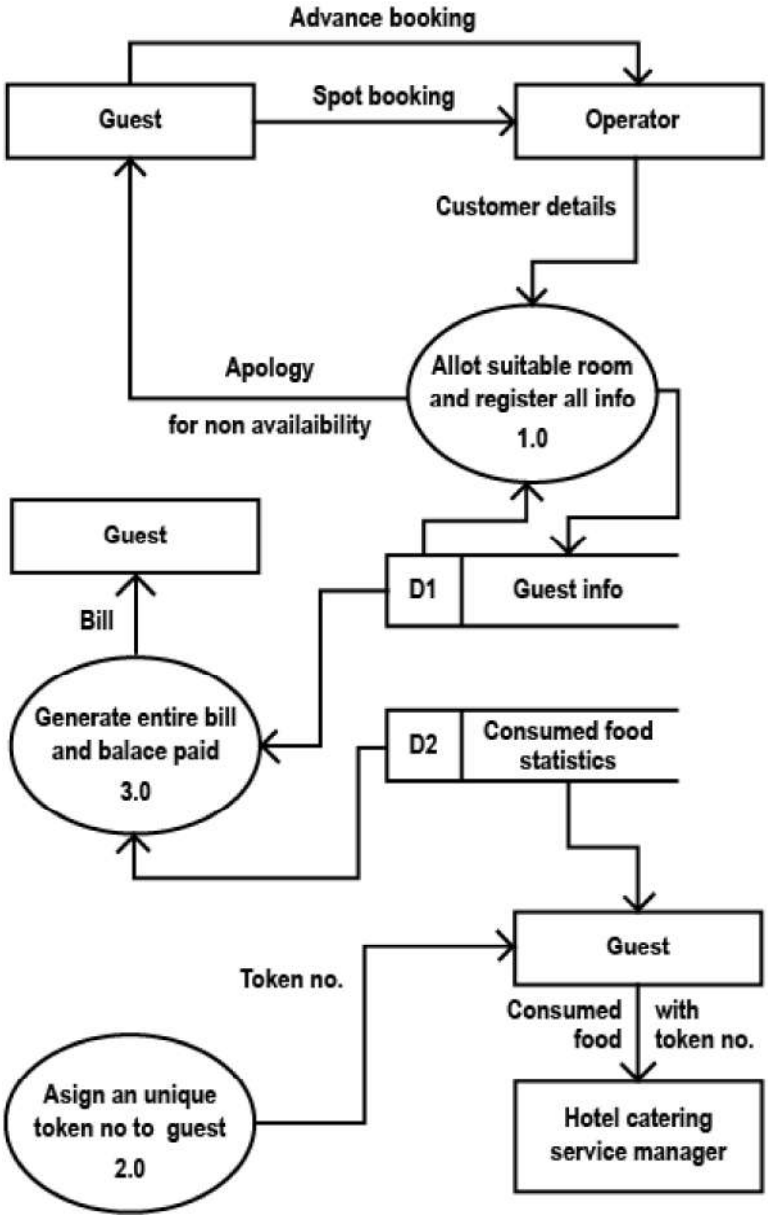
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing :Each module is tested in isolation.
- System Testing :The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.

- Beta testing (Validation) :It is done by a special group of friendly customers in an uncontrolled environment.
- Acceptance testing :It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing :Each module is tested in isolation.
- System Testing :The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) :It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing :It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Road maintenance is the performance of adaptive, preventive and corrective maintenance for a given

road in a locality. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project :The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

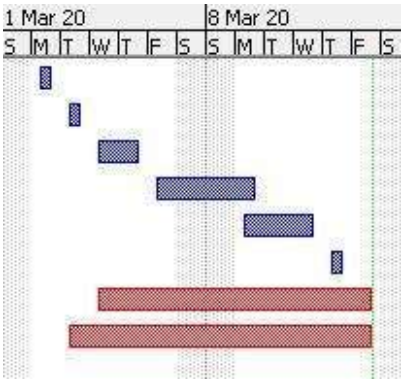
- Functional requirements :The three modules needed are.
    - Administrator module :This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
    - Criminal module :This module accesses data from the database of criminal record details and manages it properly.
    - FIR module :This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
    - Chargesheet module :This module accesses data from the database of chargesheet record details and manages it properly.
    - Court module :The main function of this module is to manage the court profiles.

- Non-Functional Requirements :
    - Usability :The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability :The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
    - Robustness :The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



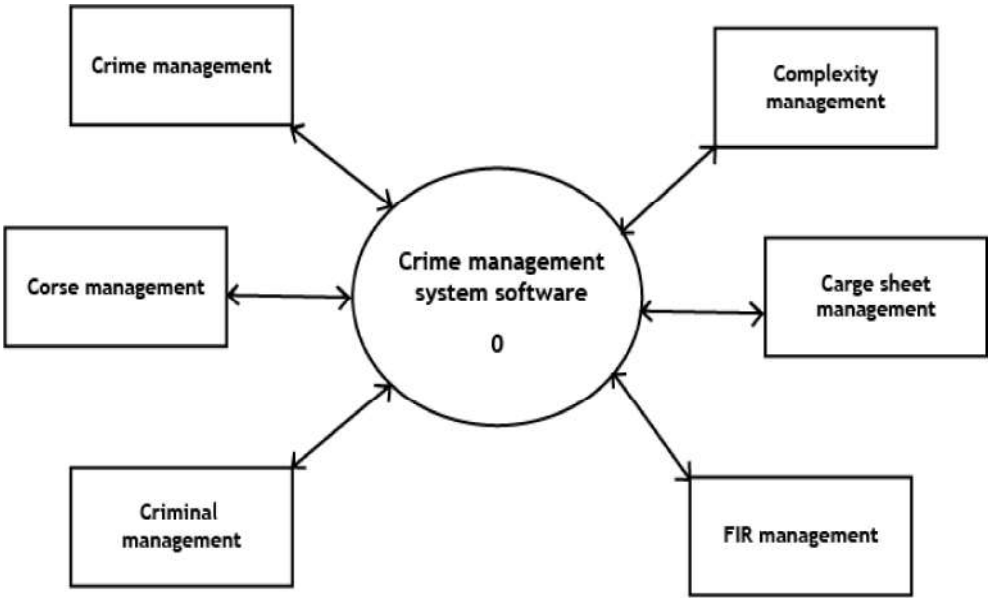
- Project Resources :
  - Hardware Resources :Several computer machines.
  - Human Resources :Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan :Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{}}$$

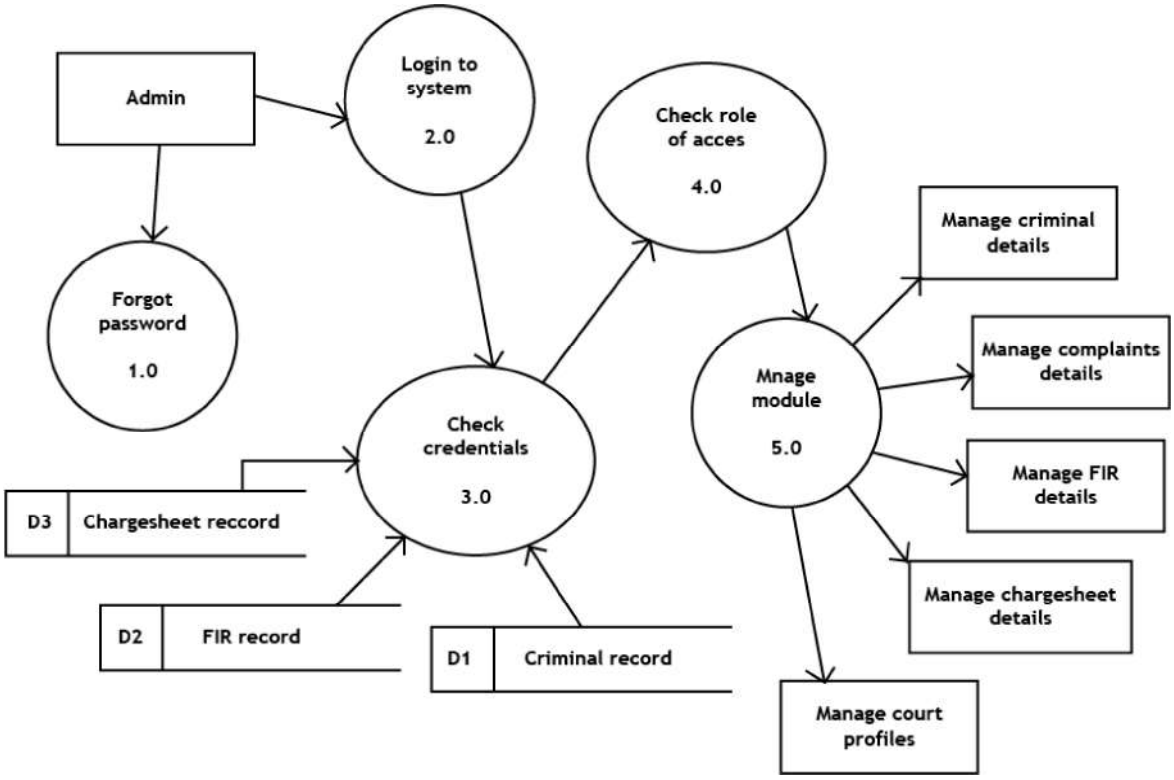
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing :Each module is tested in isolation.
- System Testing :The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
        - Beta testing (Validation) :It is done by a special group of friendly customers in an uncontrolled environment.
        - Acceptance testing :It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT :

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

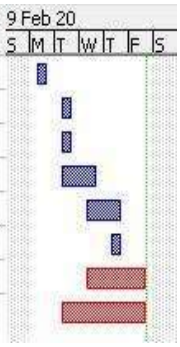
- Goal of the project :The aim and objectives for students' examination
  Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements :The three modules needed are.
  - Login and recovery module :This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
  - Credentials module :This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
  - Administrator module :This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.
- Non-Functional Requirements :
  - Usability :The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - Portability :The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

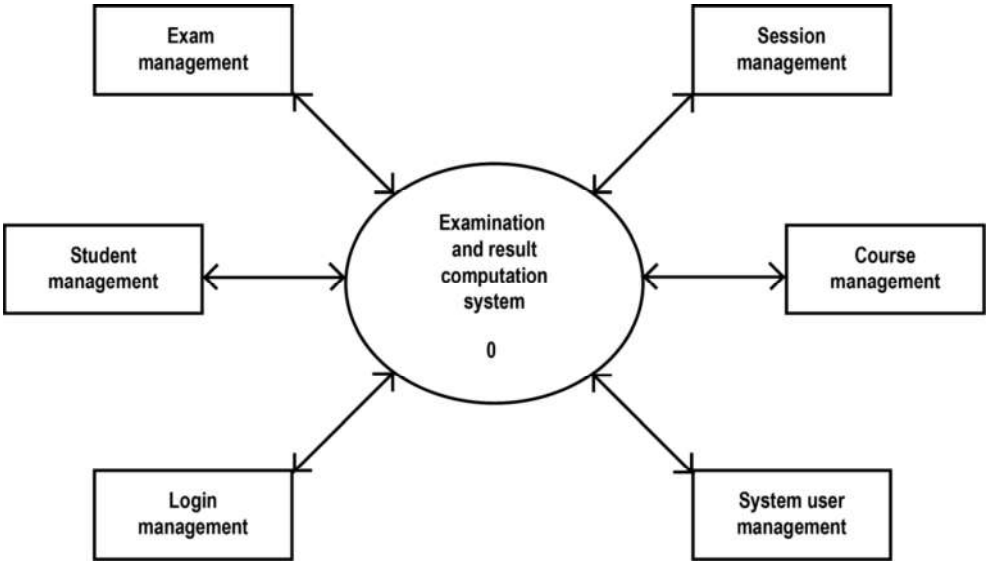


- Project Resources :
  - Hardware Resources :Several computer machines.
  - Human Resources :Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan :Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

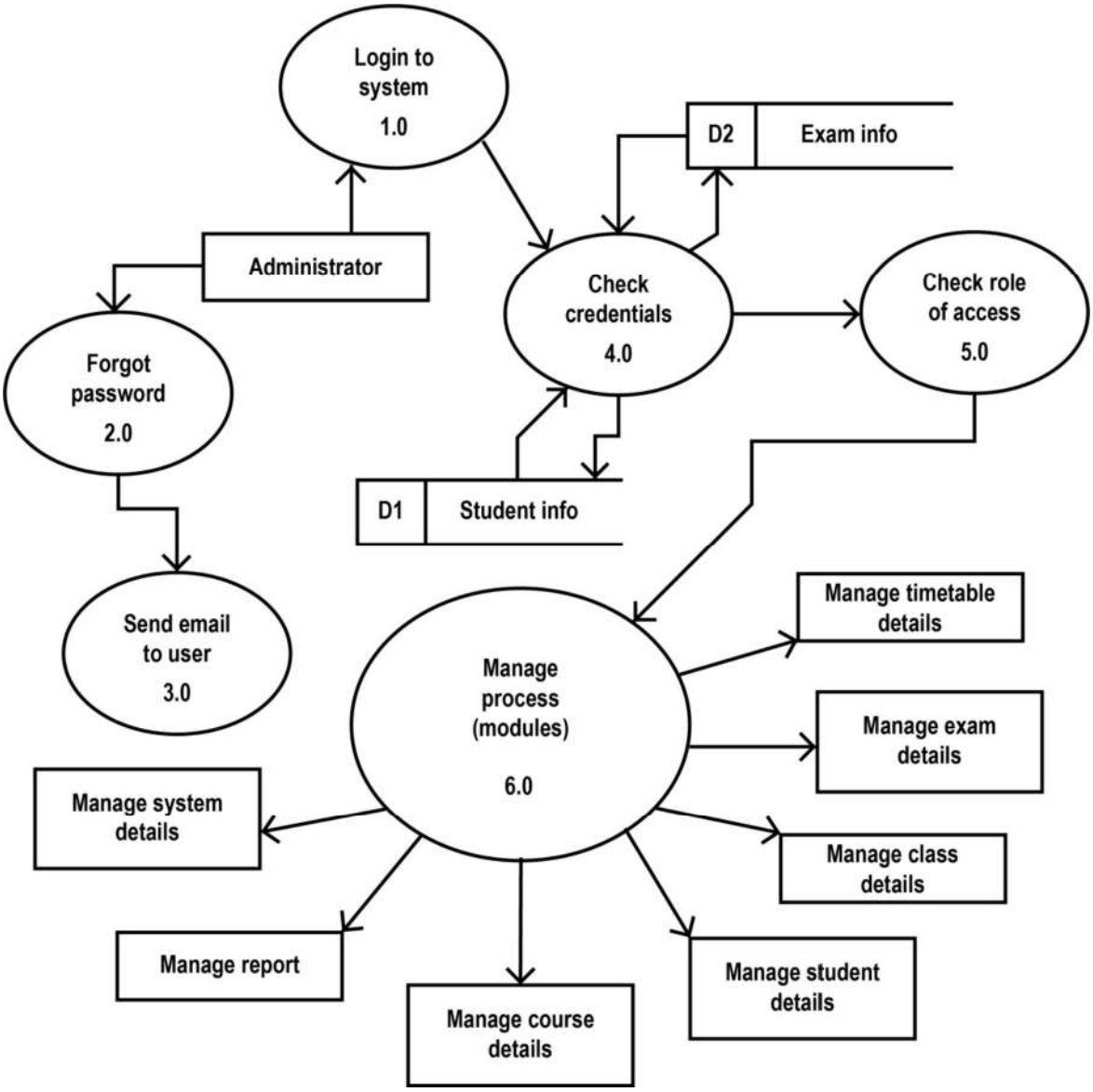
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :


Exam management
Session management
Examination and result computation system
0
Student management
Course management
Login management
System user management

2. Level 1 DFD :


Login to system
1.0
D2
Exam info
Administrator
Check credentials
4.0
Check role of access
5.0
Forgot password
2.0
D1
Student info
Send email to user
3.0
Manage process (modules)
6.0
Manage timetable details
Manage exam details
Manage system details
Manage class details
Manage report
Manage course details
Manage student details

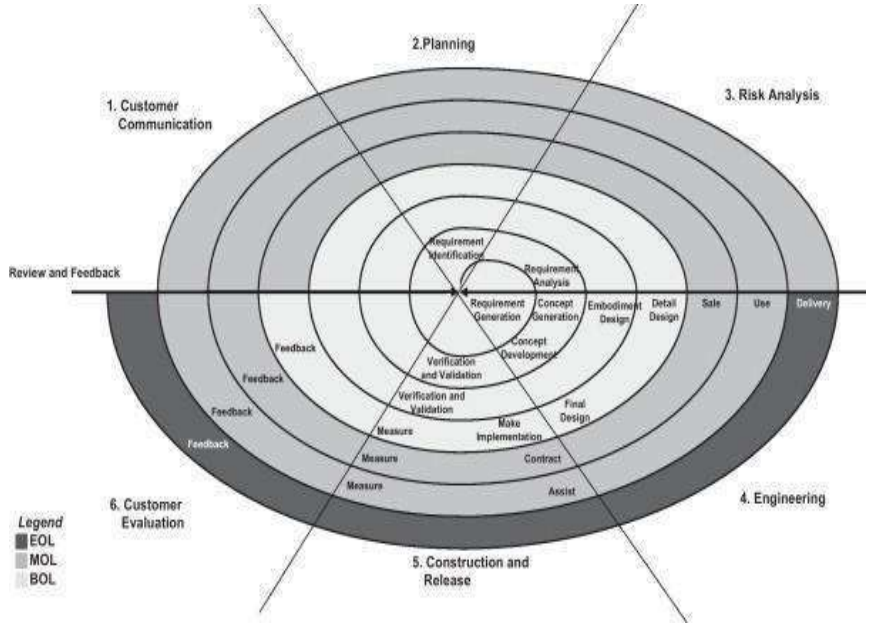
Project 7

Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔Software Requirements

Specifications *1. Goal of the Project:*

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc which was a hefty work.

2. Functional Requirements:

As per the requirements of the software from client end.

By analysis we realise that we need three modules: A. Admin Module:

This module asks username and password as input and if the

input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. Parking Module:

This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

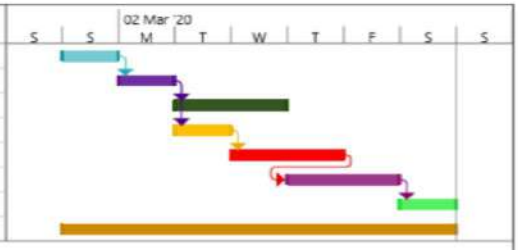
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a.Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



Project: Project7
Date: Fri 13-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

b.Project Resources

a. ManPower Resources:

i. Project Manager

ii. System Analyst

iii. Designer iv. Developer v. Tester

vi. Document Writer

b. Hardware-Software Resources

i. Processor: Intel i3 4th gen or above

ii. Ram: 4Gb or above

iii. Java Development Kit - 13 iv. Java Runtime Environment v. Sublime Text 3 (Text Editor)

vi. Ms - Project

vii. Creately viii. Google Docs

c. Risk Management Plan

All risks are measured in terms of :

A. Likelihood of a risk coming through
B. Consequence of the problem associated with that risk

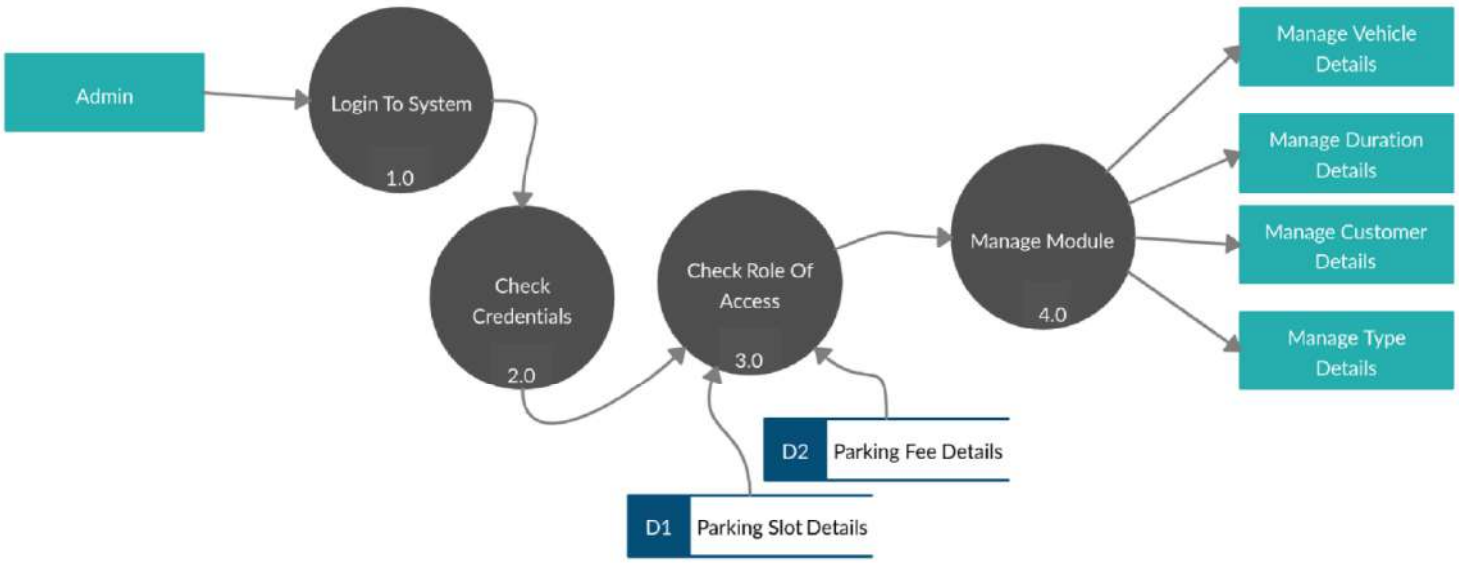
Priority of each risk is multiplication of above two

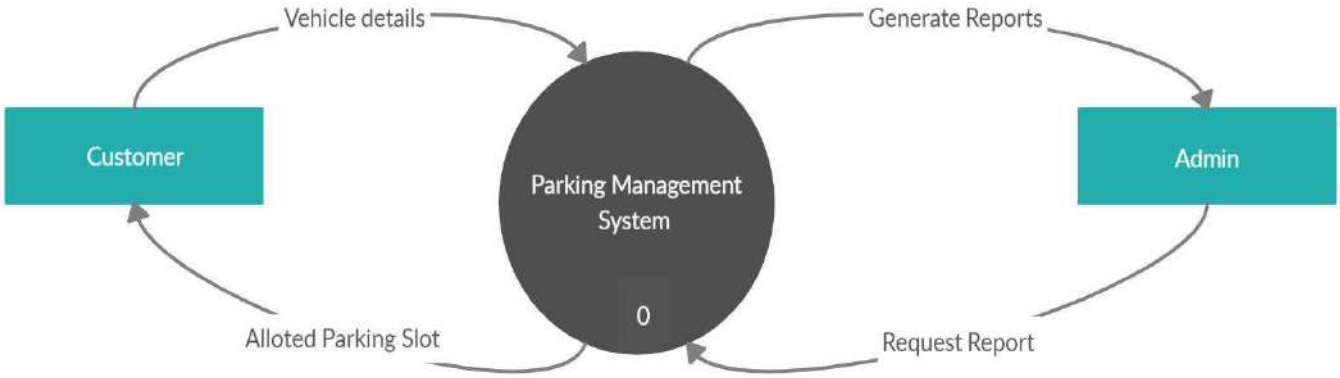
Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. Architectural Design





2. **Component Level Design**

   - **Level 1 DFD**

TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation **2. System Testing:**

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

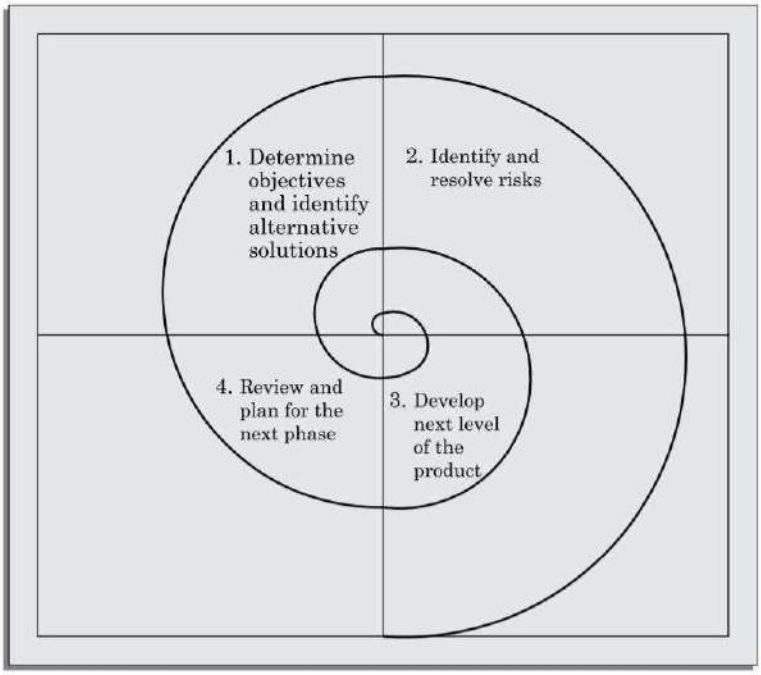


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Customer Module – In this module, a customer is registered and the profile is Maintained.

Input: customer details
Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained. A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

Report Module – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

Input: store details, product details, retail price details, bill details, billing counter details, customer details
Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

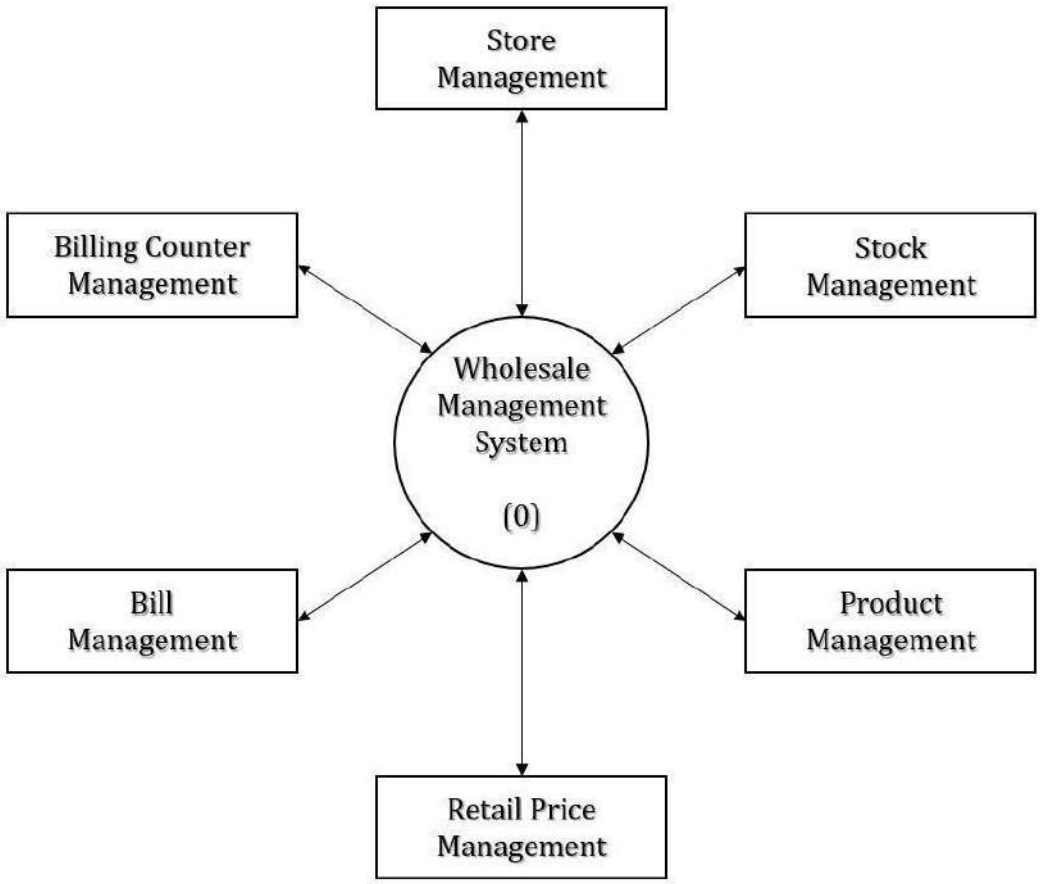


Fig – DFD level zero

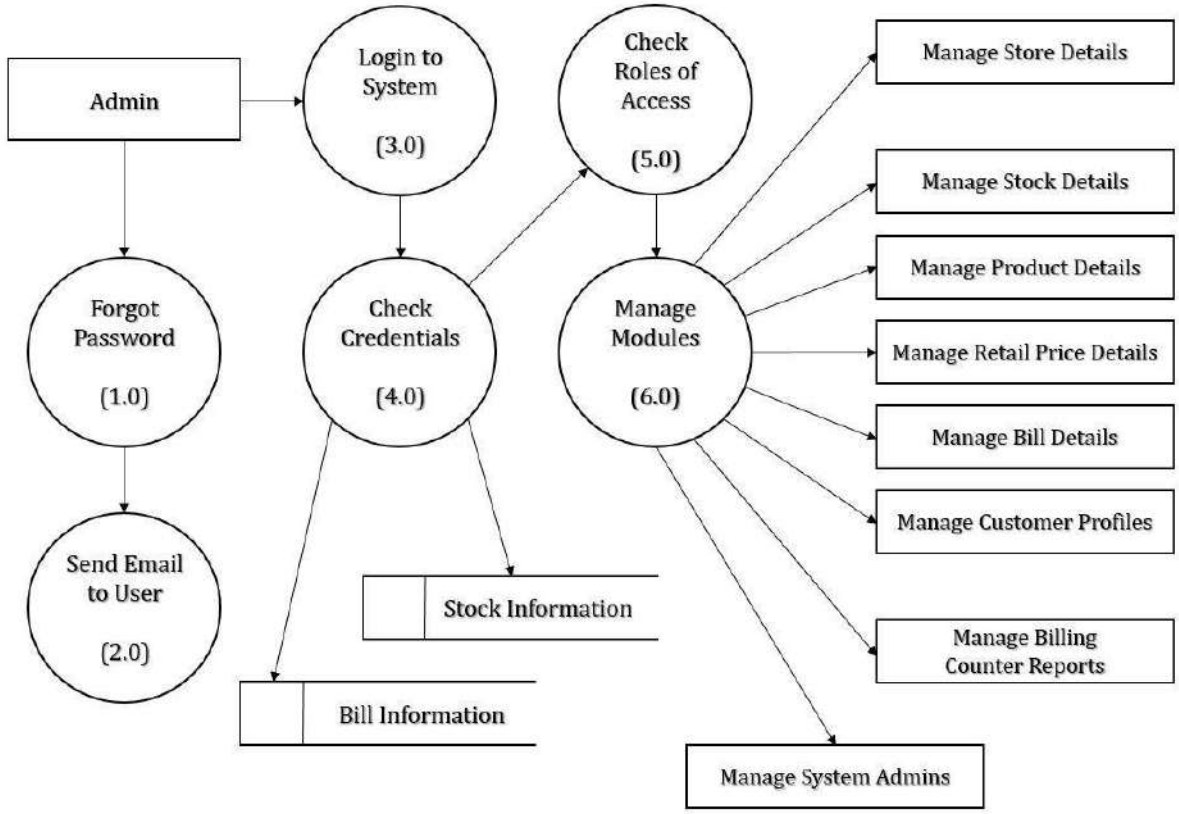
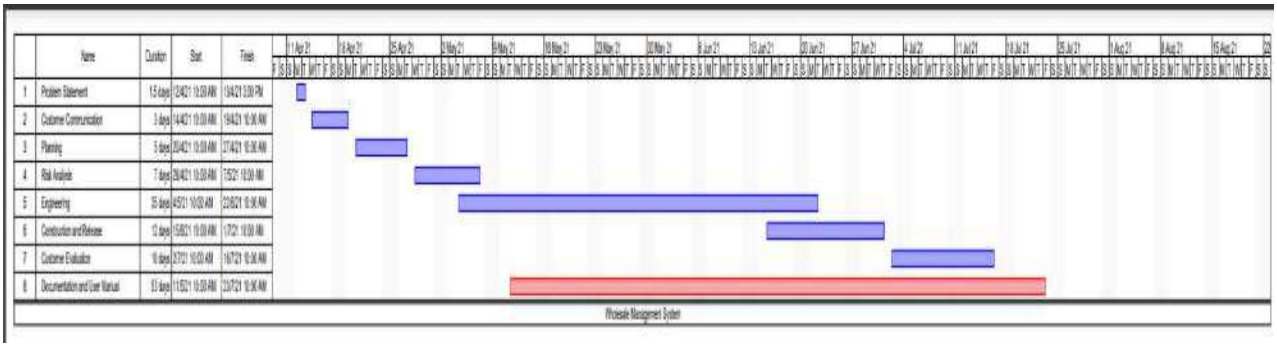


Fig – DFD level one

Project Management

Project Schedule –



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

The likelihood of a risk coming true (**r**)

The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

risk leverage= risk exposure before reduction-risk exposure after reductioncost of reduction

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

α (Alpha) Testing – It is the system testing performed by the development team in a controlled environment.

ii. **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii. **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i. **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii. **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii. **Adaptive maintenance –** Porting the software to work in a new environment.

PROJECT ON RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Trinanjan Daw

College Roll No. : 708

Examination Roll No. : 2021141292

Department: Comp. Sc.

Supervisor: Prof. Manas Pal

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement – A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

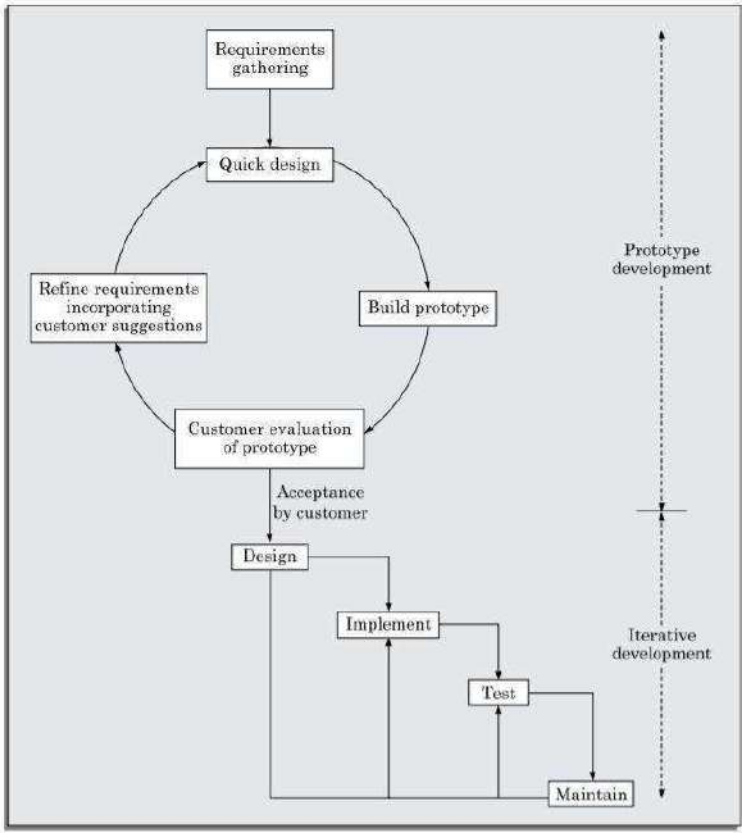


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: unique CN

2. **Purchase Module** – In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input: purchase made by the customer, CN
   Output: updated database

3. **Promotional Module** – In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output: prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

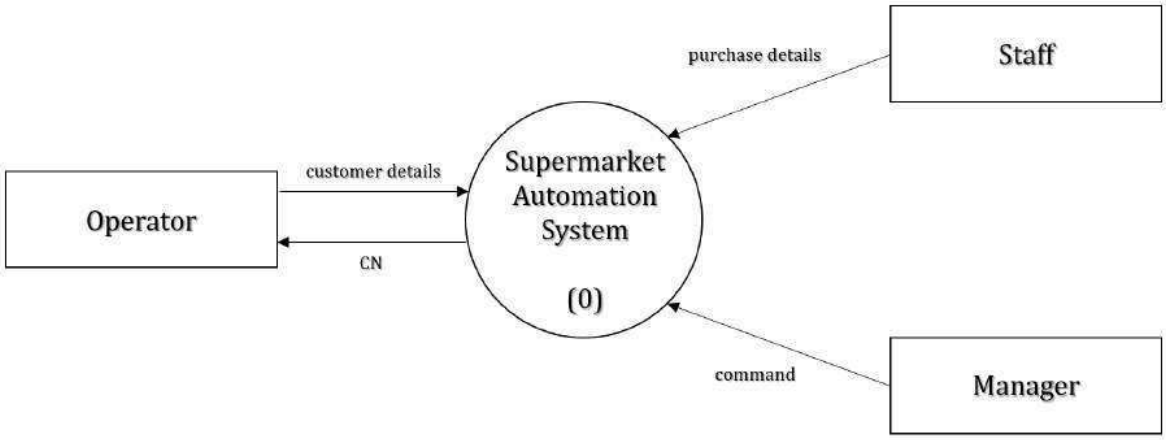


Fig - DFD level zero

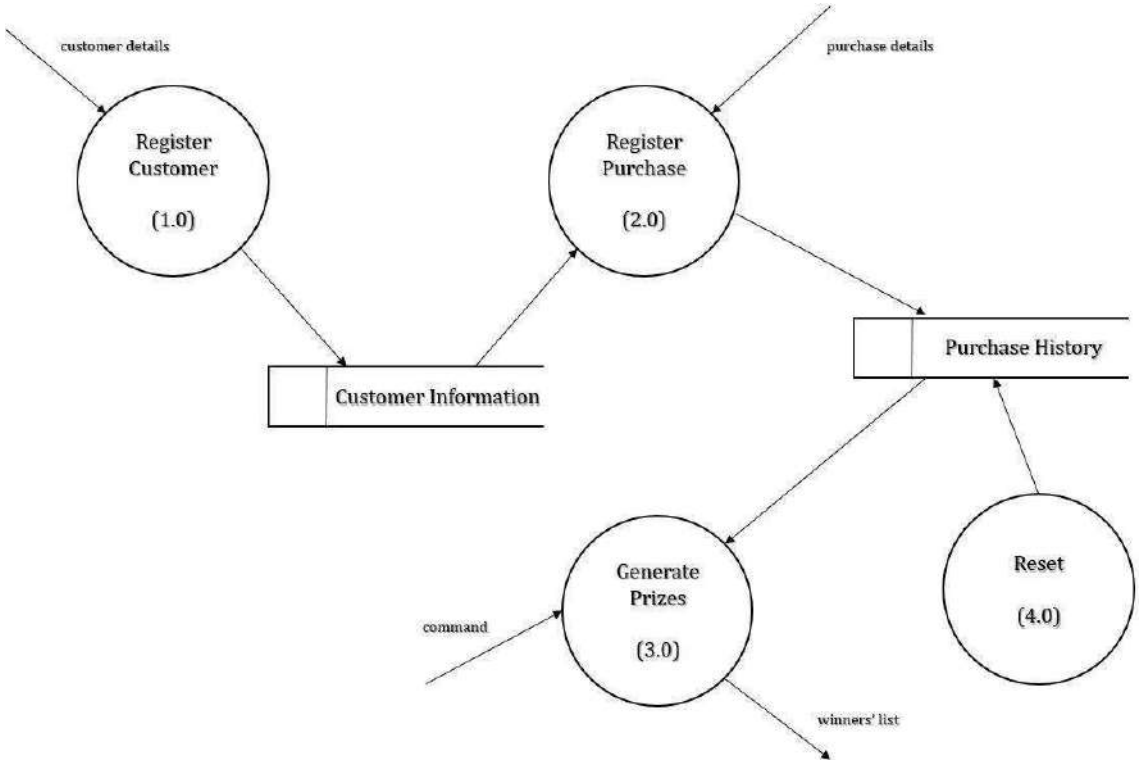


Fig - DFD level one

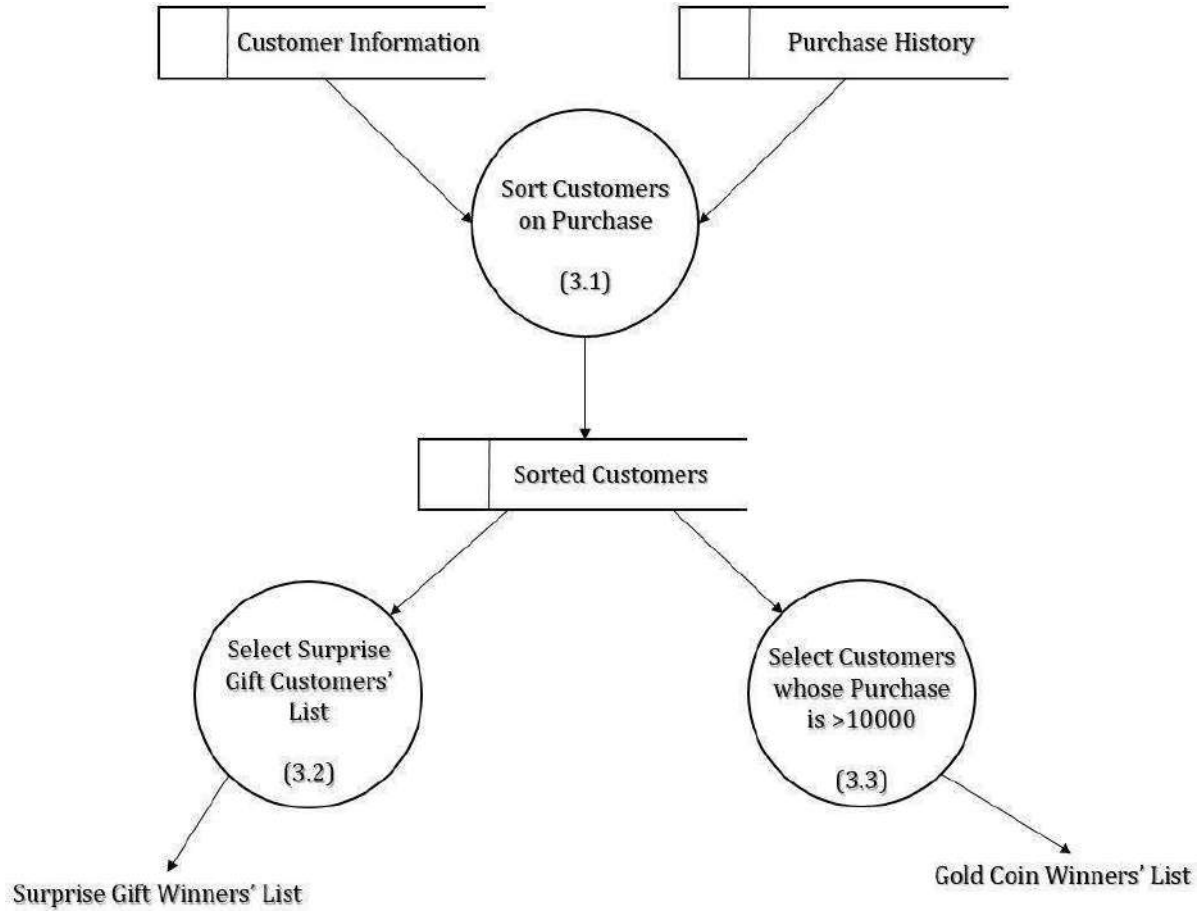
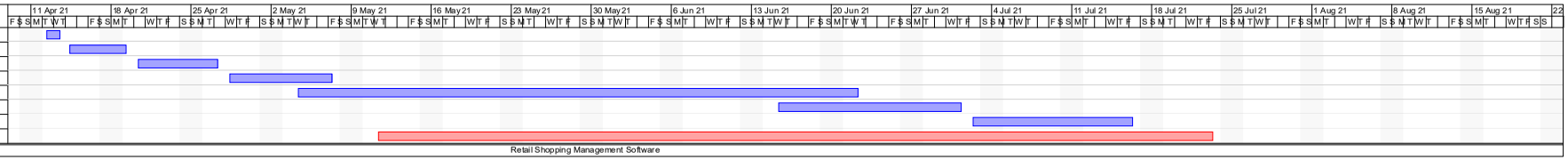


Fig – DFD level two of Process 3.0

Project Management

Project Schedule –The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and UserManual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Retail Shopping Management Software

Risk Analysis

Risk Assessment - Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p -** priority with which the risk must be handled

r - the probability of the risk becoming true

s - the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject

the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement – It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model – Incremental model

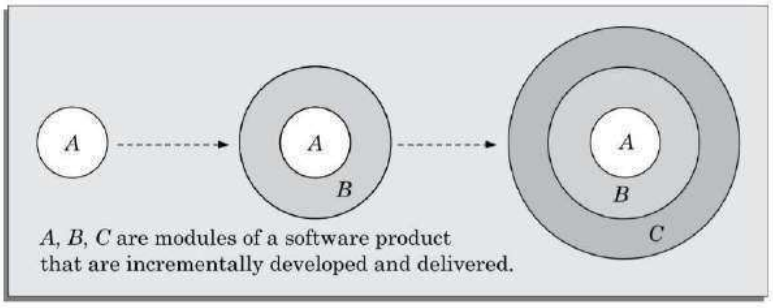


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module** – In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module** – In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module** – In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

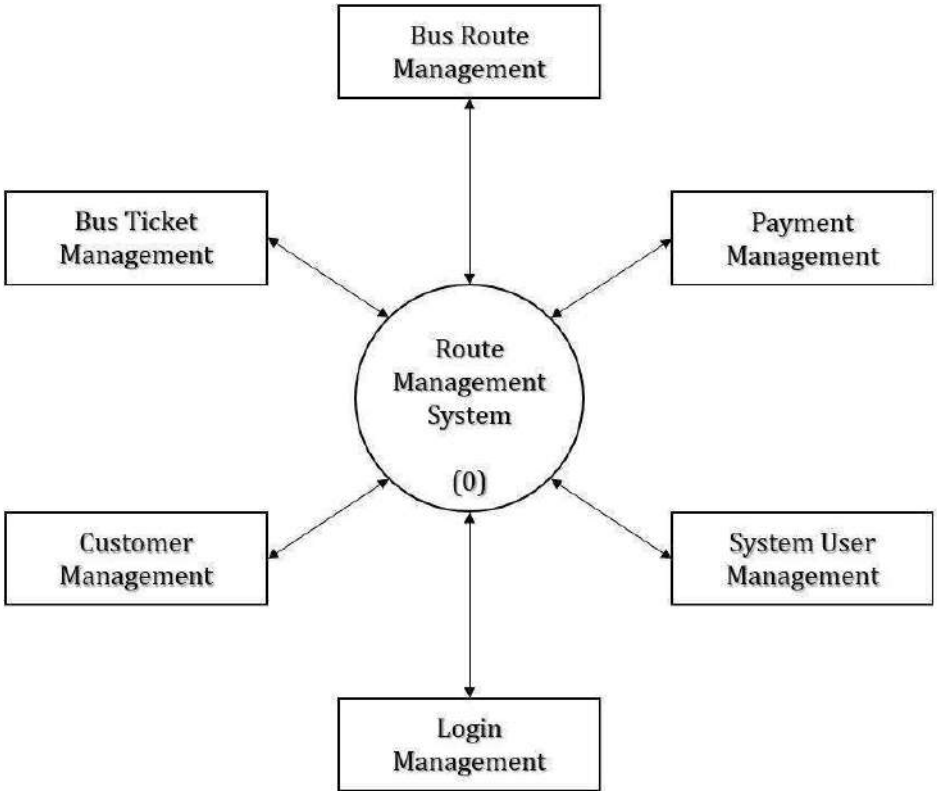


Fig – DFD level zero

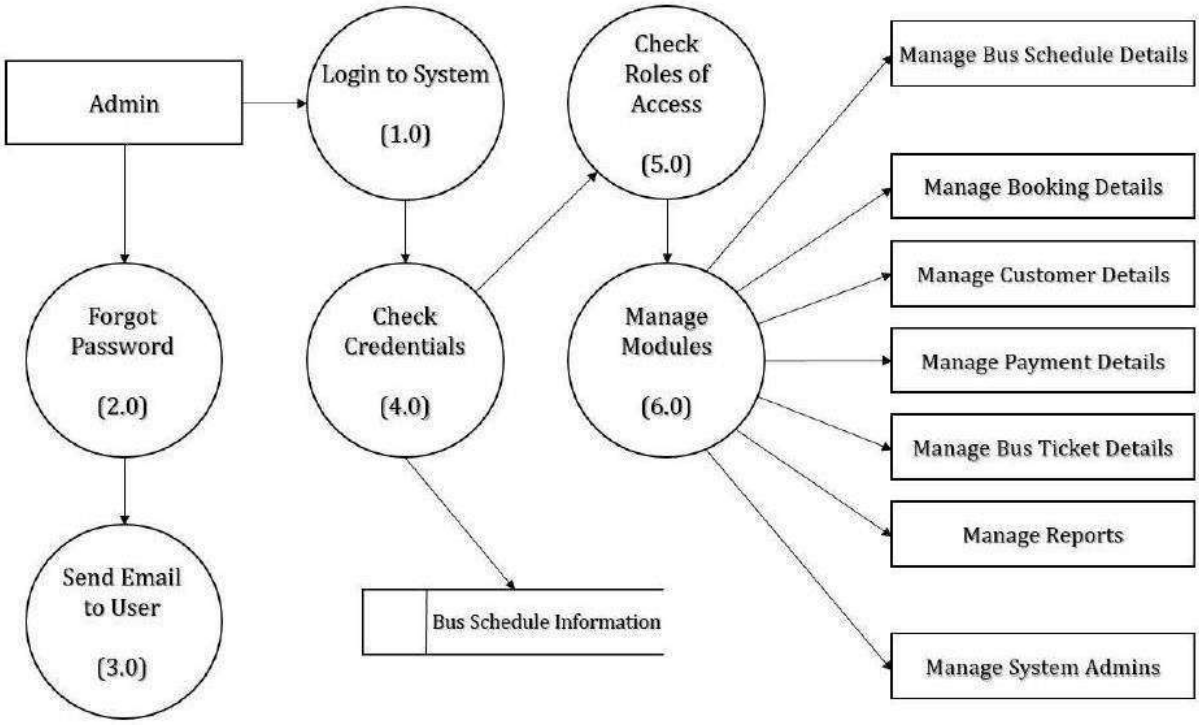
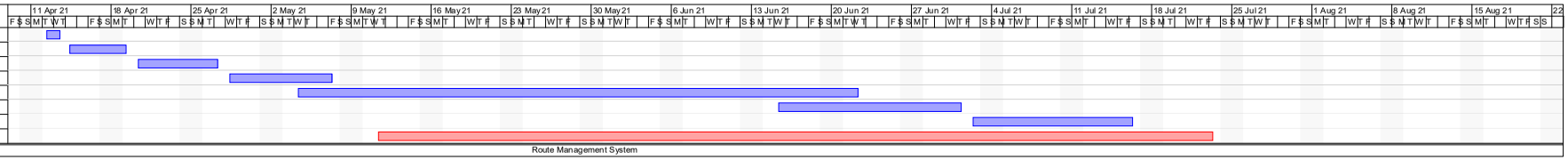


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and UserManual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement – Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model – Incremental development model

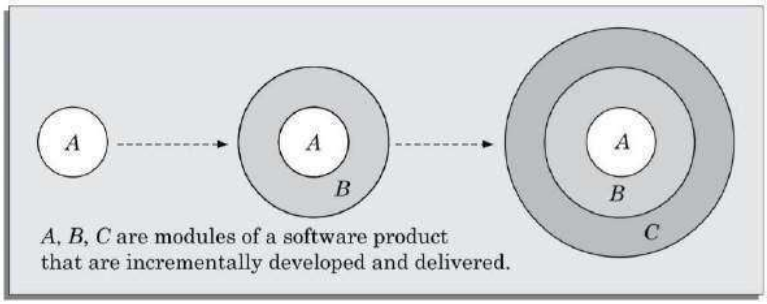


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module** – In this module, a patient is registered and the profile is maintained.

   Input: patient details
   Output: unique profile

2. **Doctor Module** – In this module, the details of doctors are maintained.

   Input: patient details
   Output: updated database

3. **Diagnosis Module** – In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input: patient details, doctor details, test details
   Output: diagnosed disease, updated database

4. **Report Module** – In this module, patient records, medicine and drug records, test records are processed and reports are generated.

   Input: patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

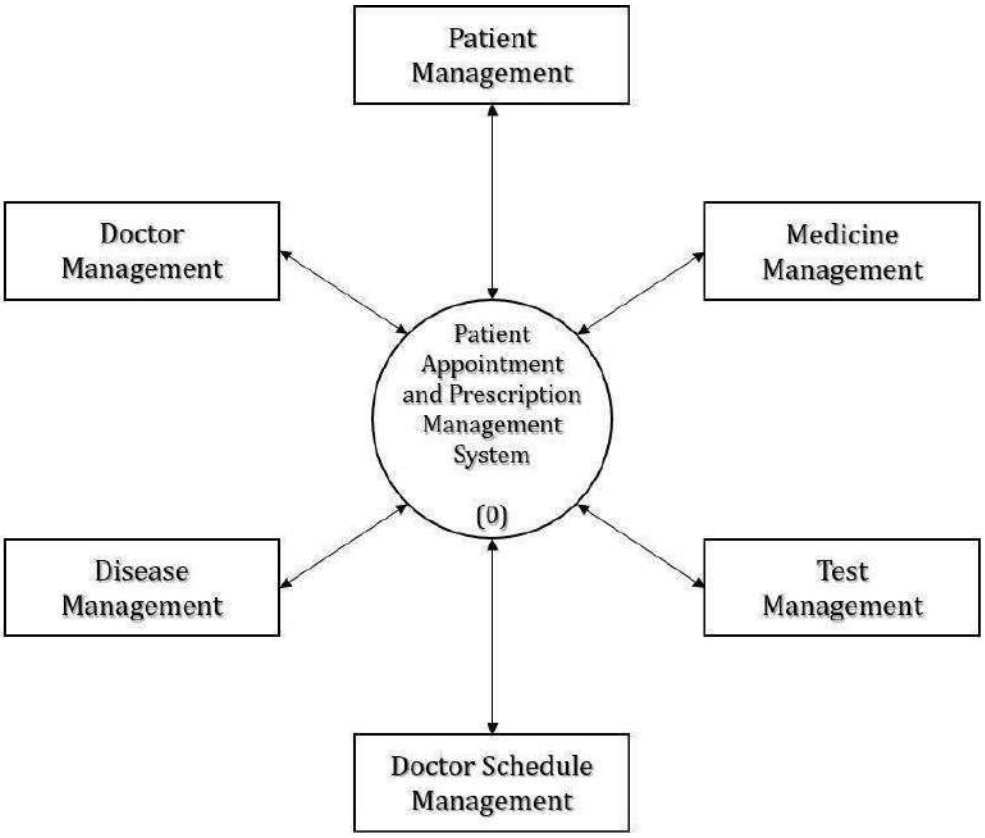


Fig – DFD level zero

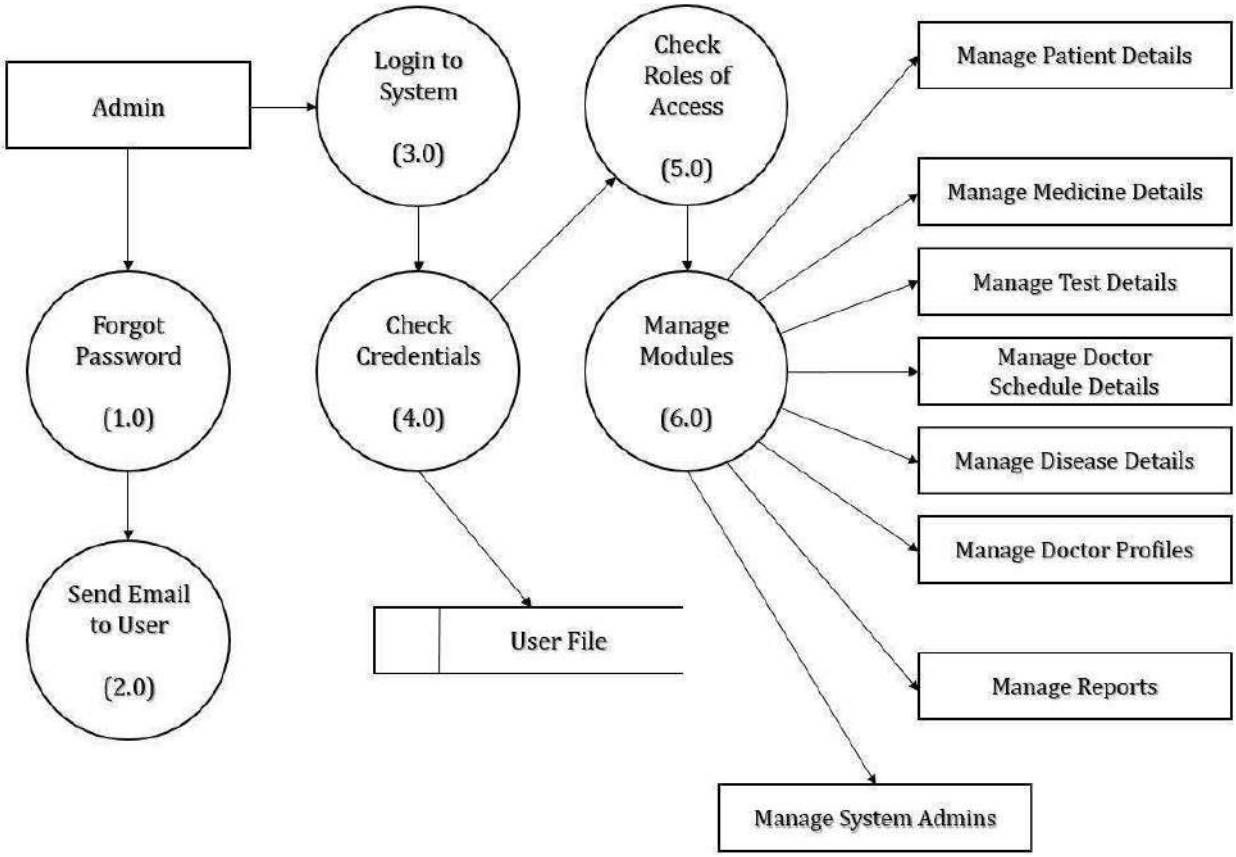
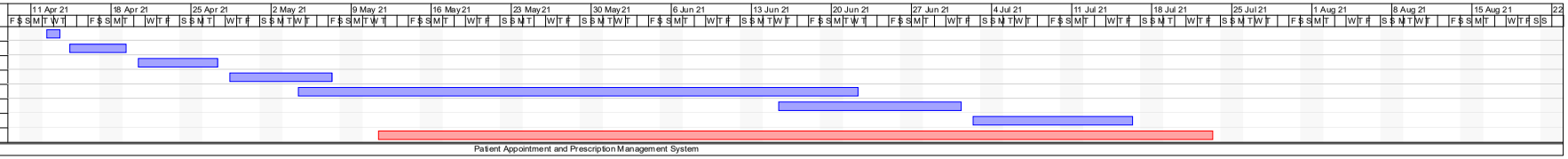


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and UserManual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement – Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model – Spiral development model

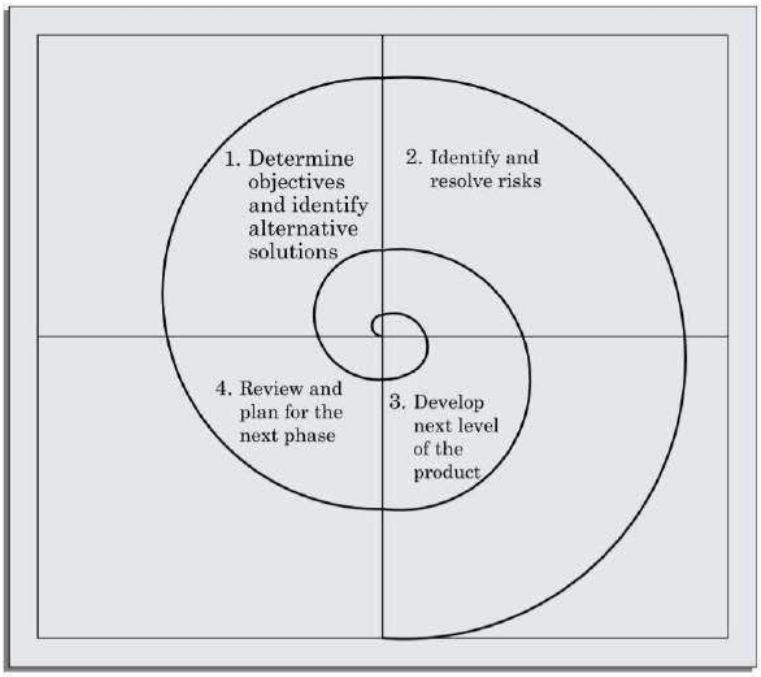


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module** – In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input: guest details
   Output: unique profile, updated database

2. **Catering Module** – In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input: guest details, room number, token number
   Output: updated database

3. **Bill Module** – In this module, the total bill is generated when the guest prepares to check out.

   Input: guest details, room number, token number, food items consumed
   Output: generated bill

Design Engineering

Data Flow Diagrams (DFDs)

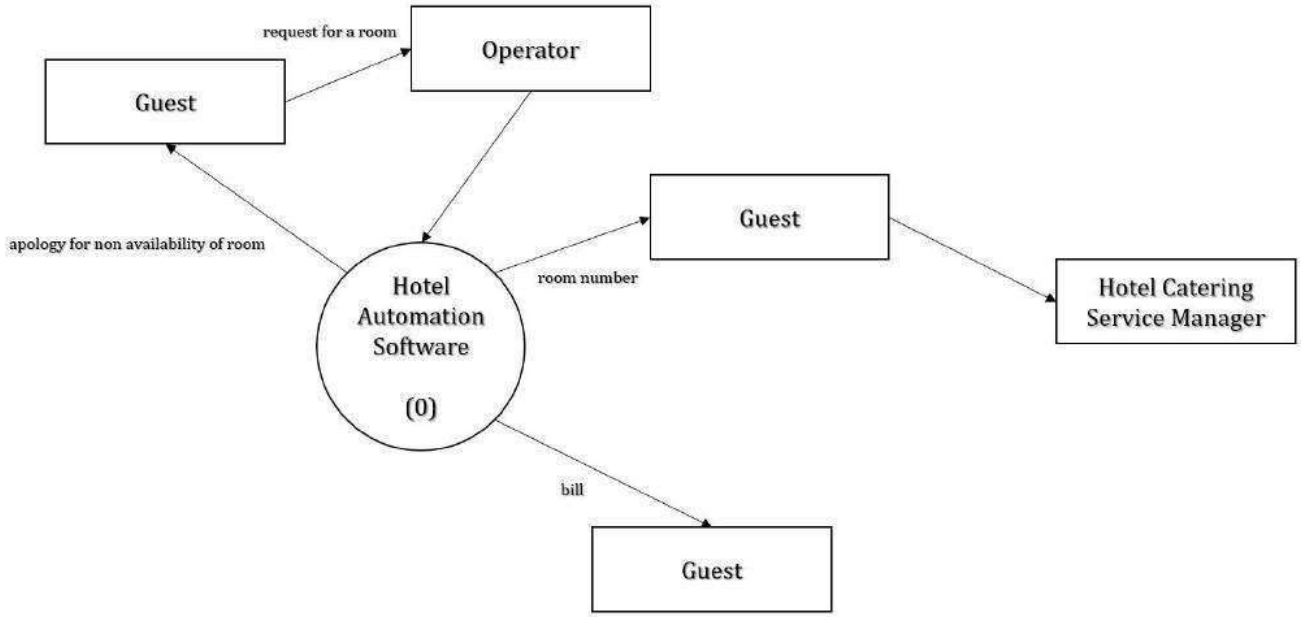


Fig – DFD level zero

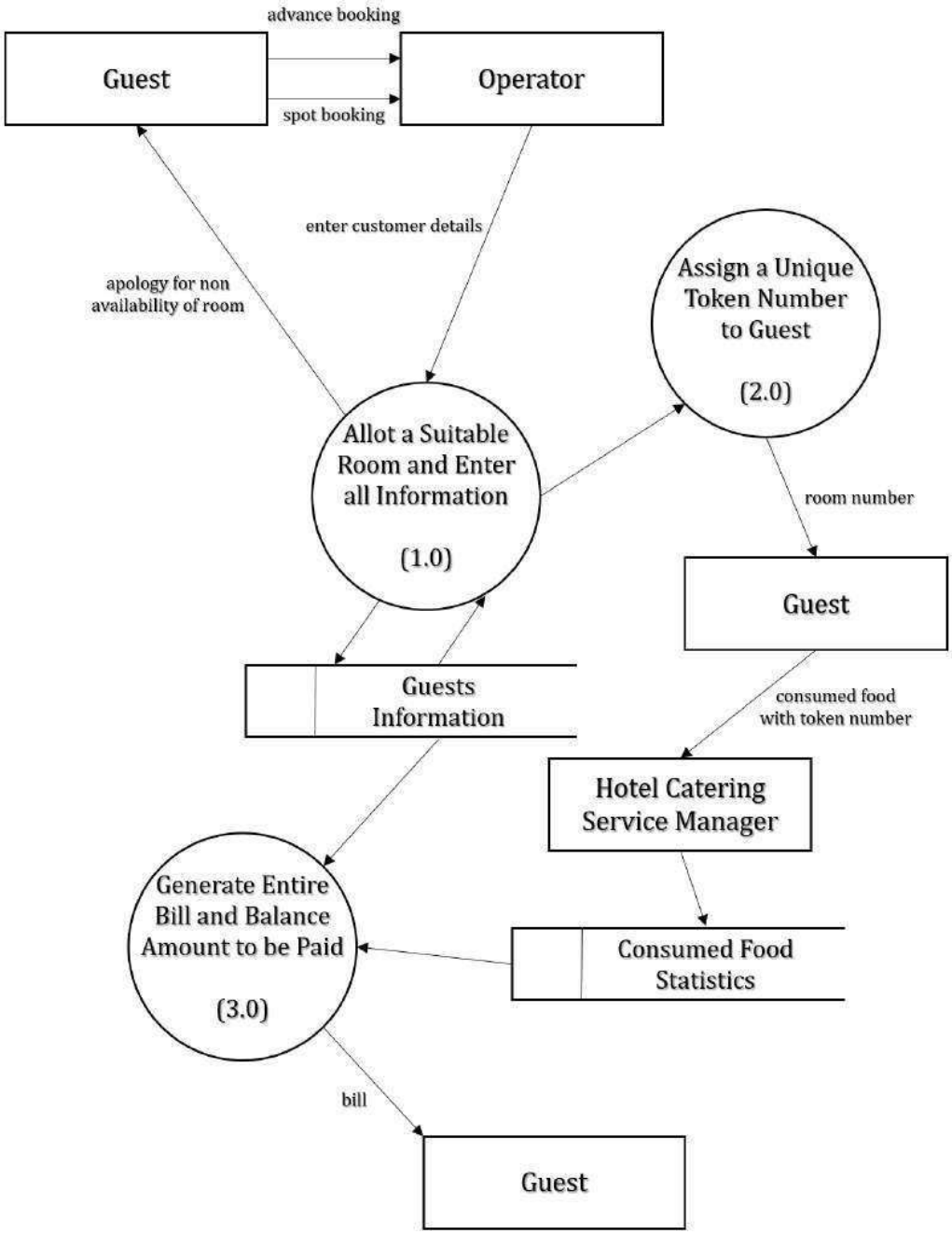
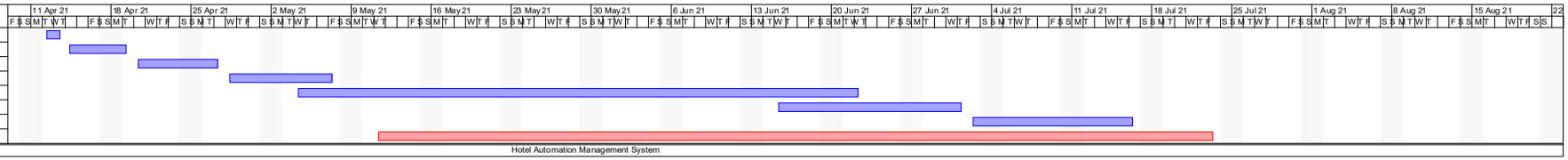


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Constructionand Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentationand User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Hotel Automation Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement–Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

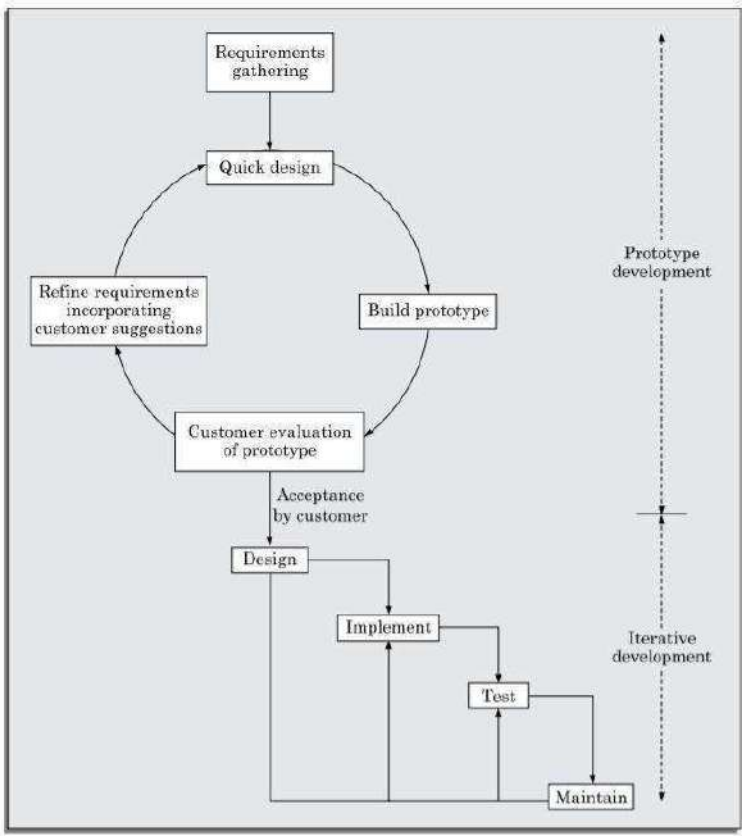


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module** – In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input: admin details
   Output: unique profile, granted level of access

2. **Complain Module** – In this module, the complaint is registered and FIR is filed. A database is maintained for keeping the complaint details.

   Input: complaint details
   Output: filed FIR, updated database

3. **Crime Module** – In this module, the details of the crime are maintained and chargesheet is filed.

   Input: complaint details, FIR details
   Output: filed chargesheet

4. **Court Module** – In this module, the case is taken to the court and legal action is taken accordingly.

   Input: complaint details, FIR details, chargesheet details
   Output: legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

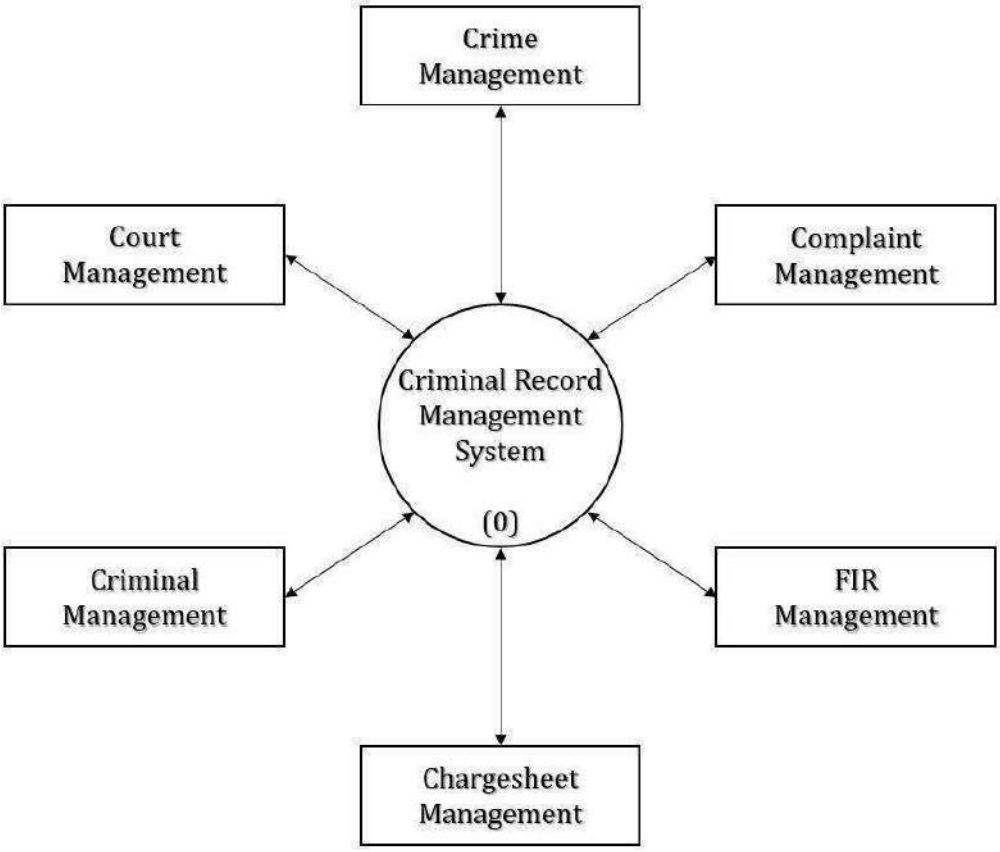


Fig – DFD level zero

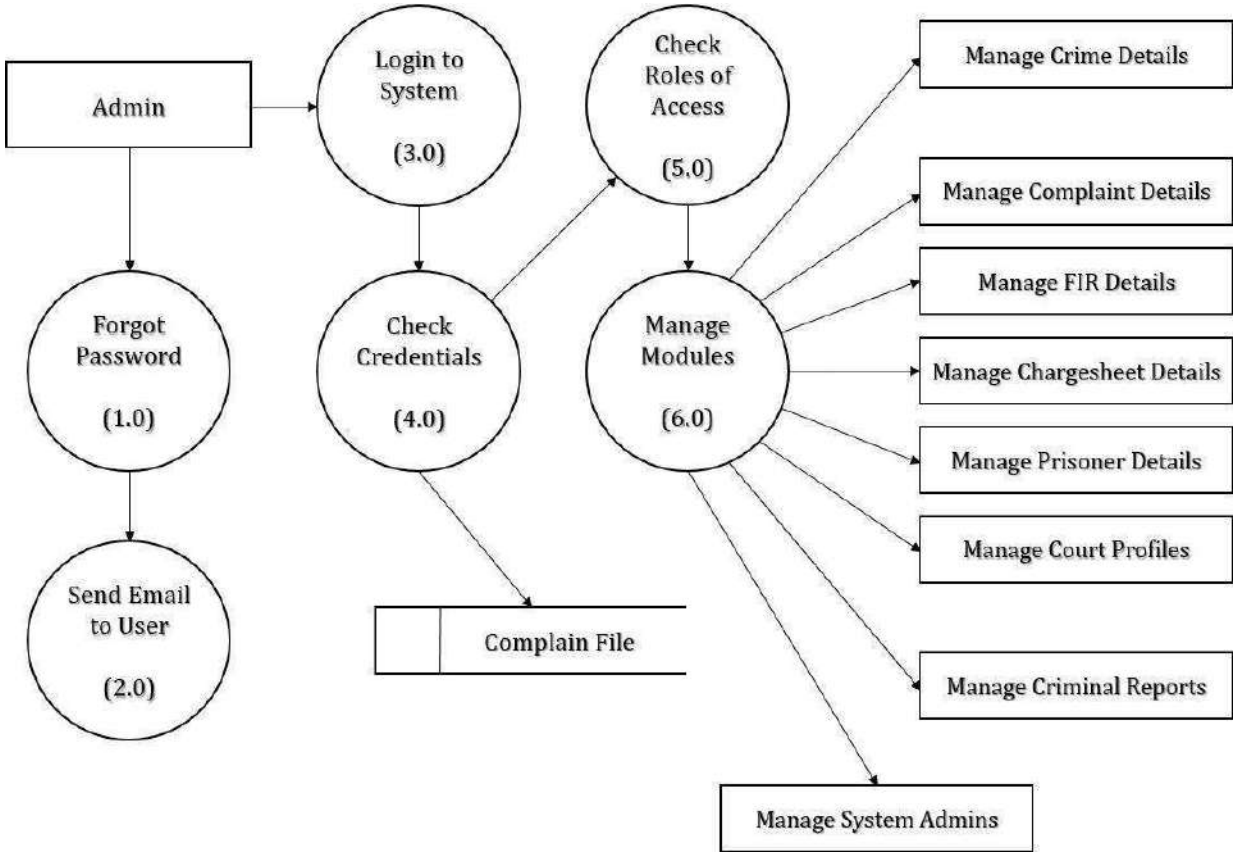
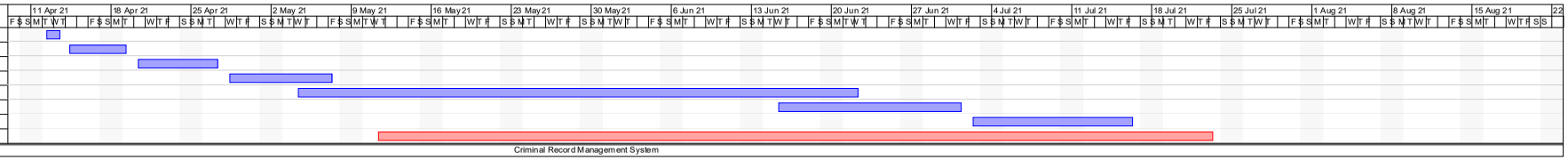


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and UserManual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Criminal Record Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement – Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model – Spiral development model

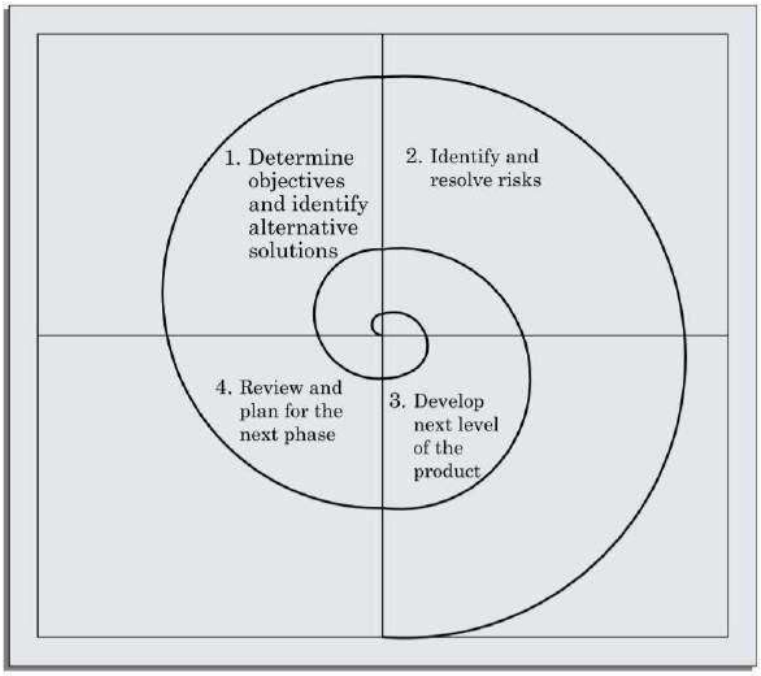


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module** – In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module** – In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module** – In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module** – In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

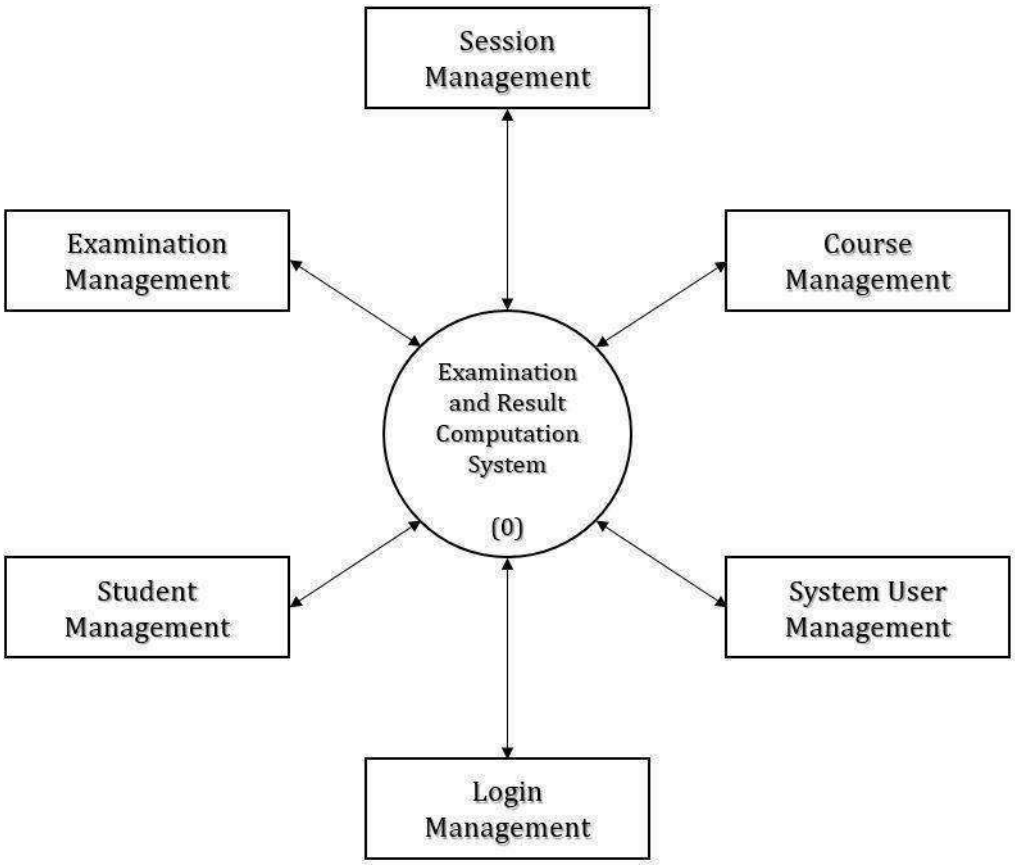


Fig – DFD level zero

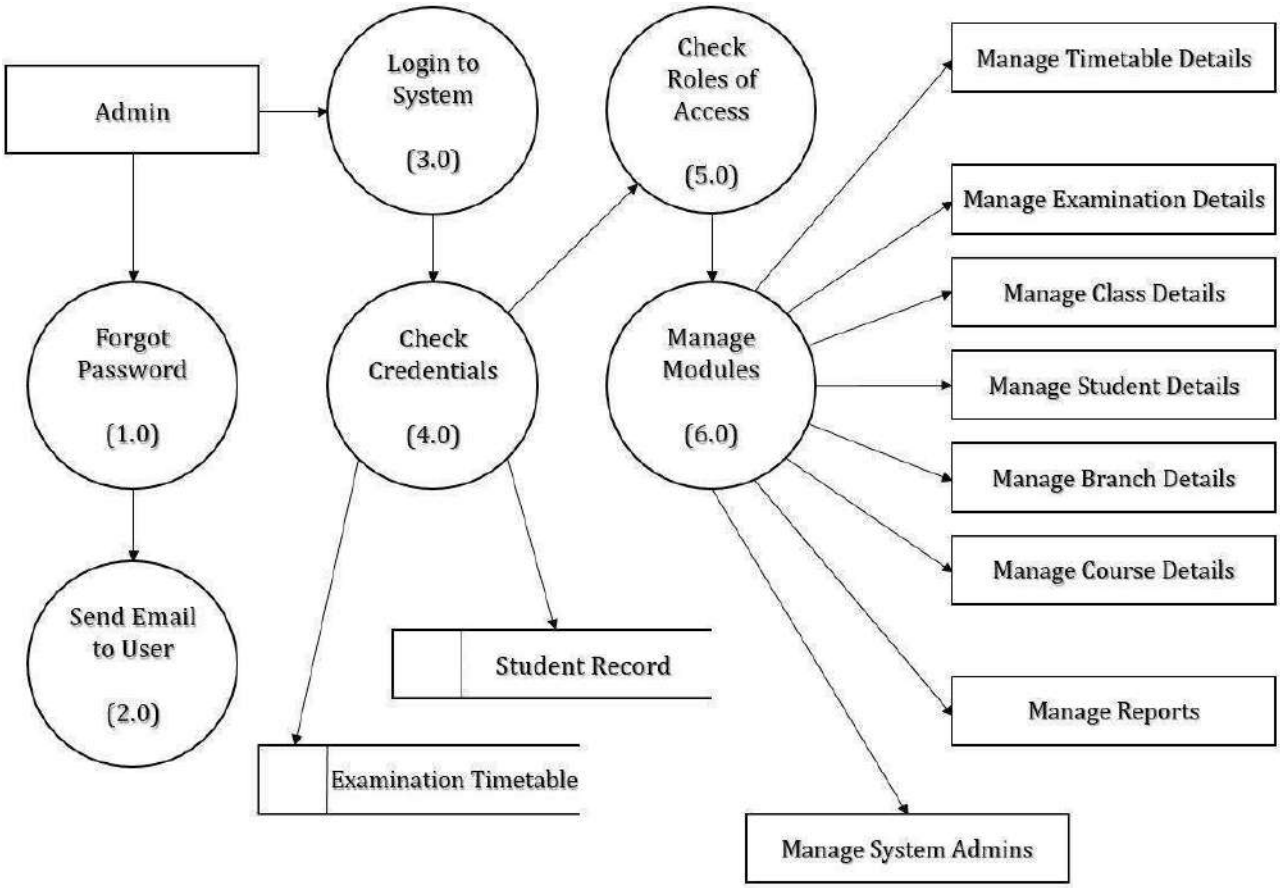
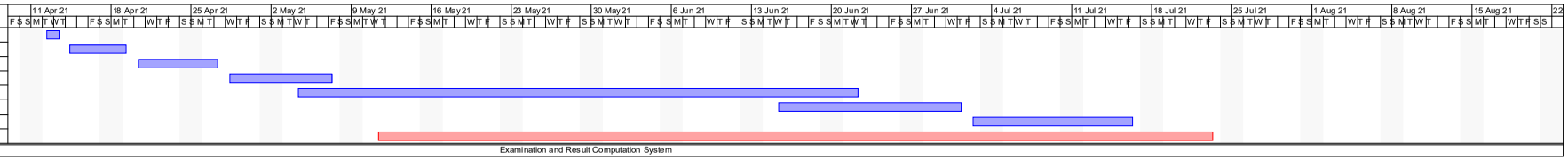


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Constructionand Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentationand User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

PARKING ALLOCATION SYATEM

Problem Statement–Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model – Spiral development model

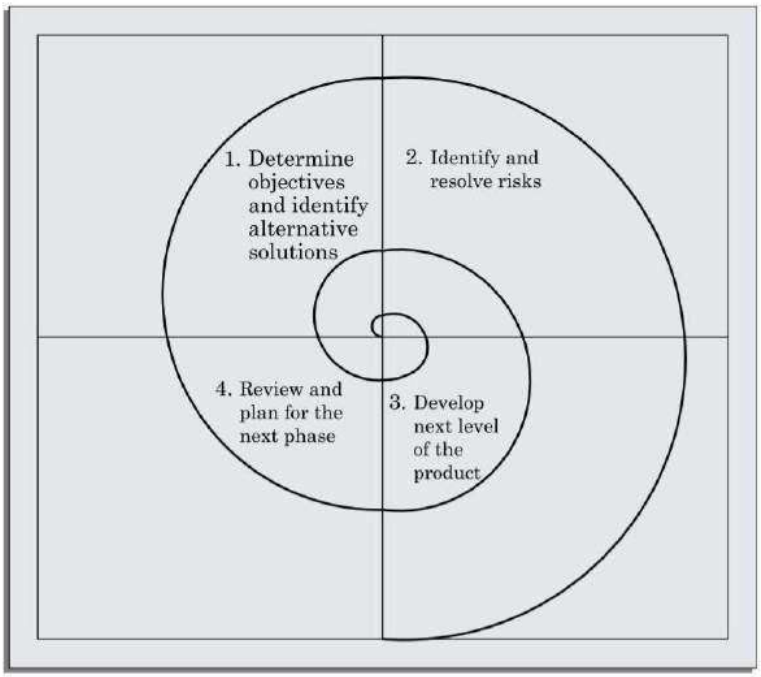


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module** – In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module** – In this module, parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module** – In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

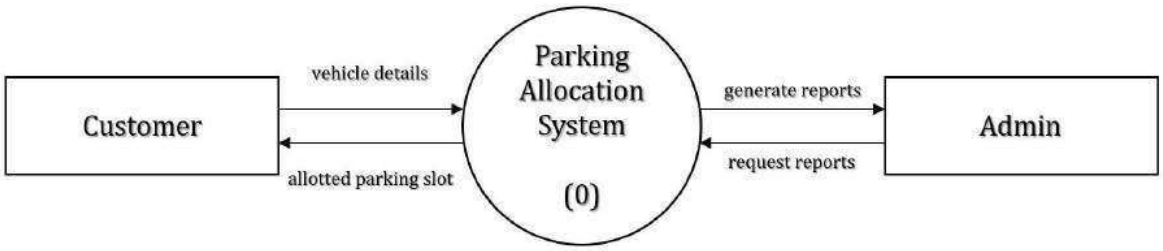


Fig – DFD level zero

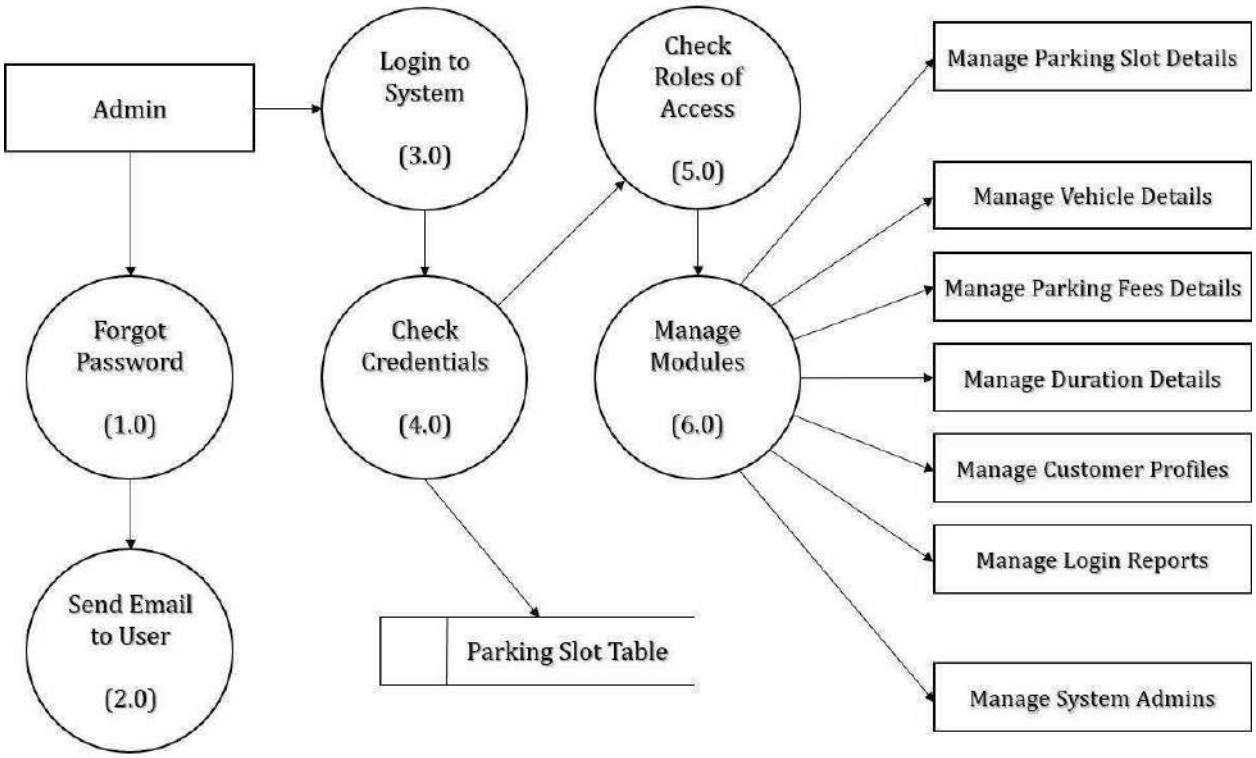
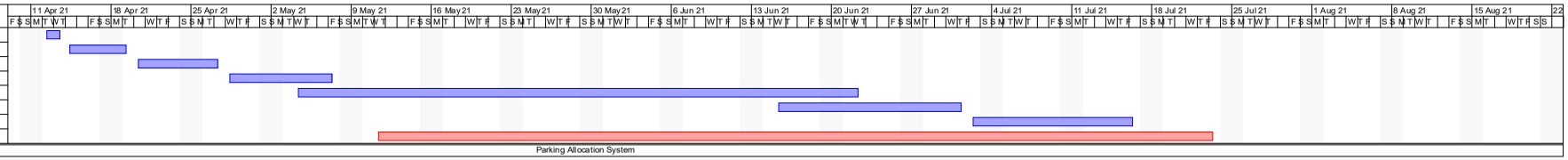


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Constructionand Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentationand User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Parking Allocation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

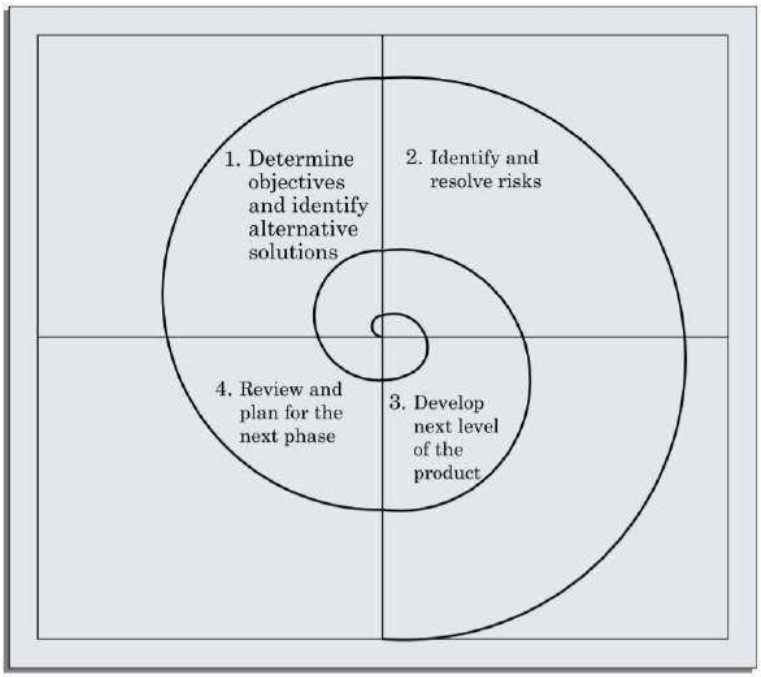


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module** – In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module** – In this module, all the bills are maintained. A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module** – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

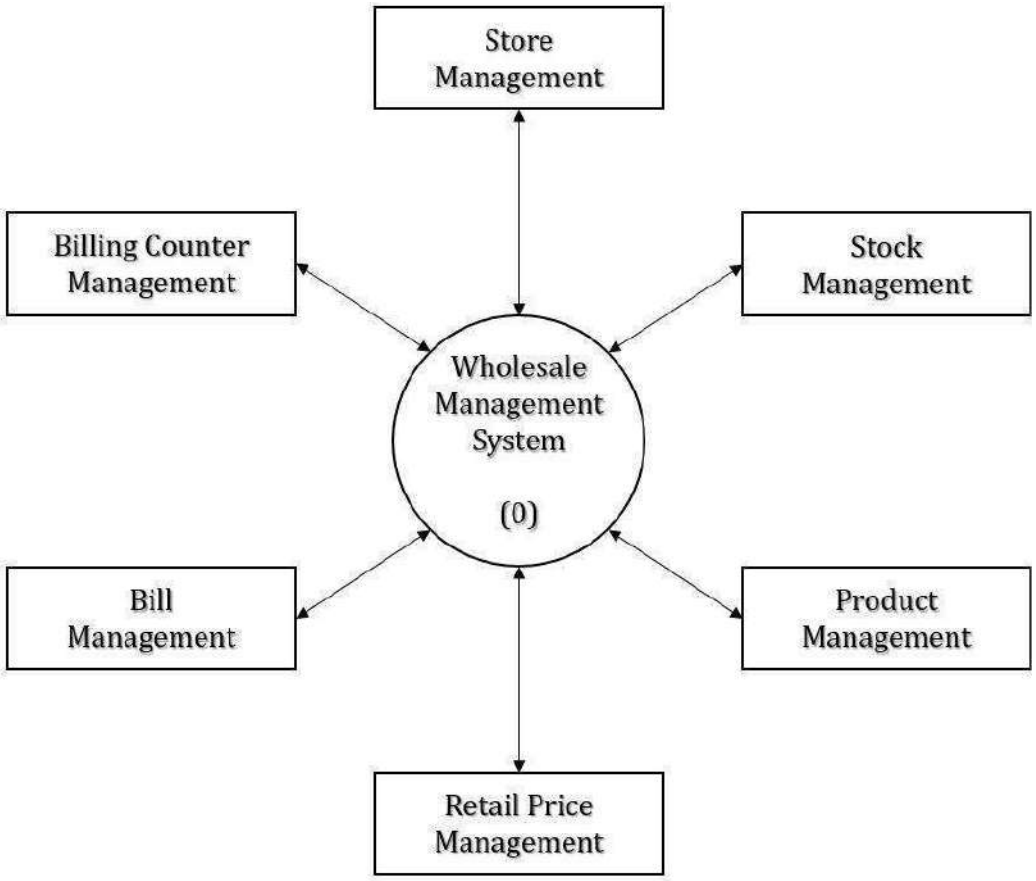


Fig – DFD level zero

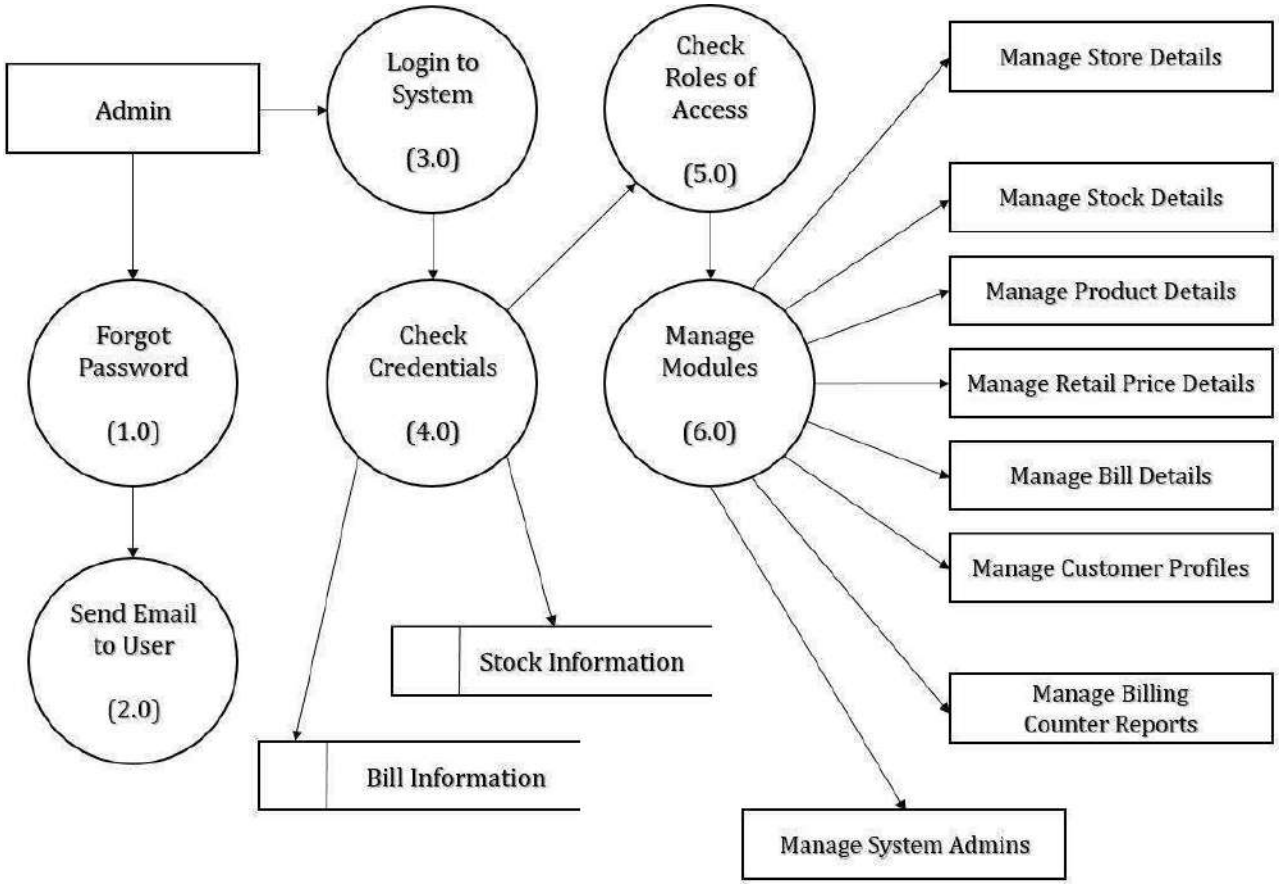
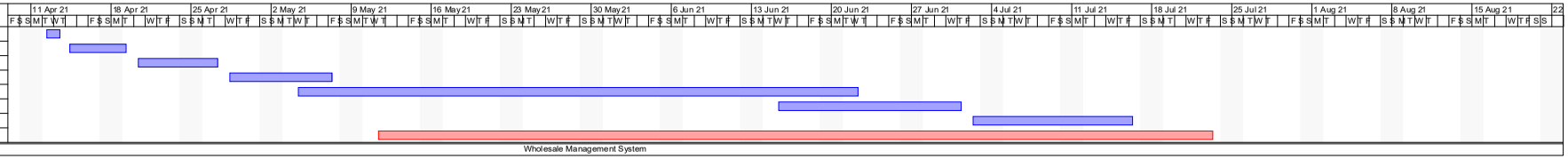


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Constructionand Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentationand User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Wholesale Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r ×s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction\ -\ Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

PROJECT ON RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Biswajit Manna

College Roll No.: 709

Examination Roll No.: 2021141293

Paper Code: CMSADSE3

Semester: V

Department: Computer Science

Supervisor: Prof. Manas Pal

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement – A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

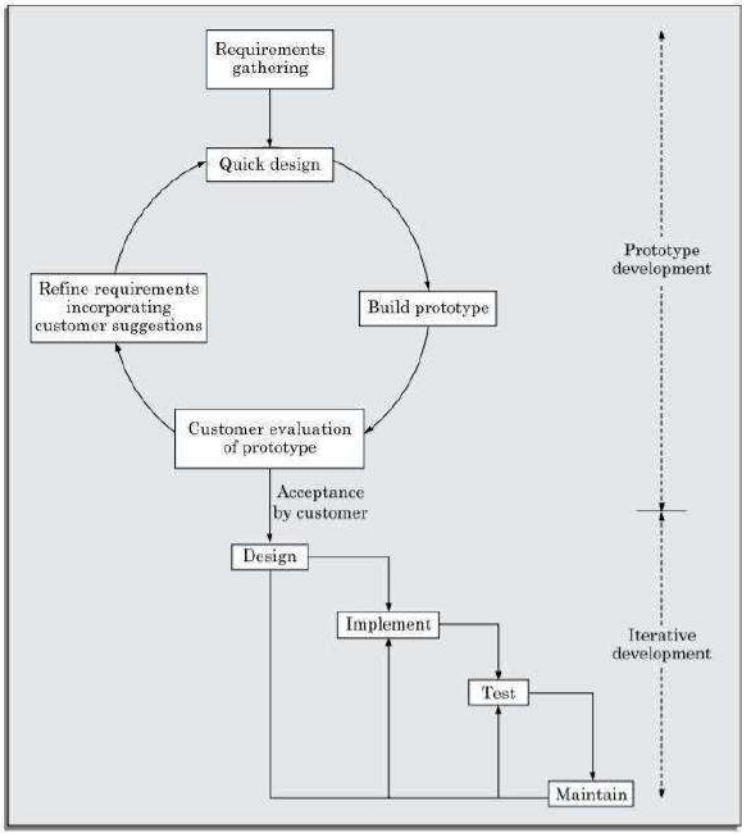


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: unique CN

2. **Purchase Module –** In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input: purchase made by the customer, CN
   Output: updated database

3. **Promotional Module –** In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output: prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

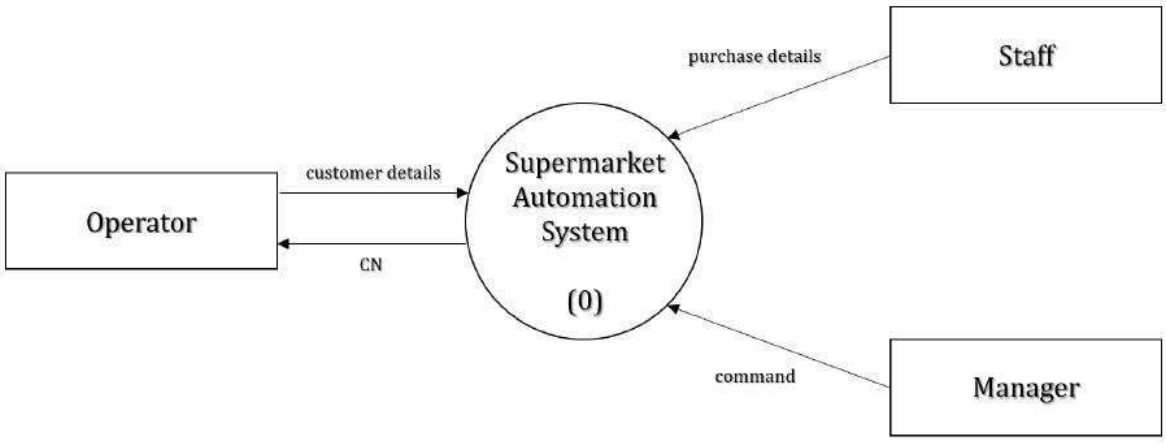


Fig – DFD level zero

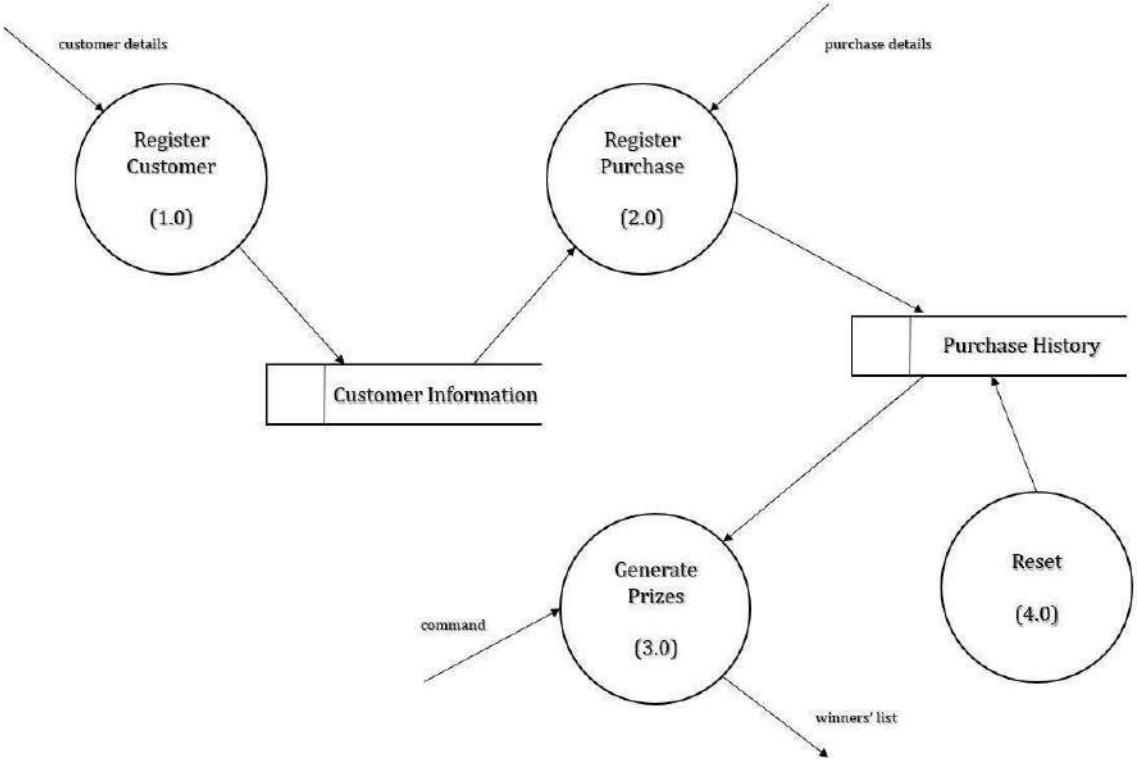


Fig – DFD level one

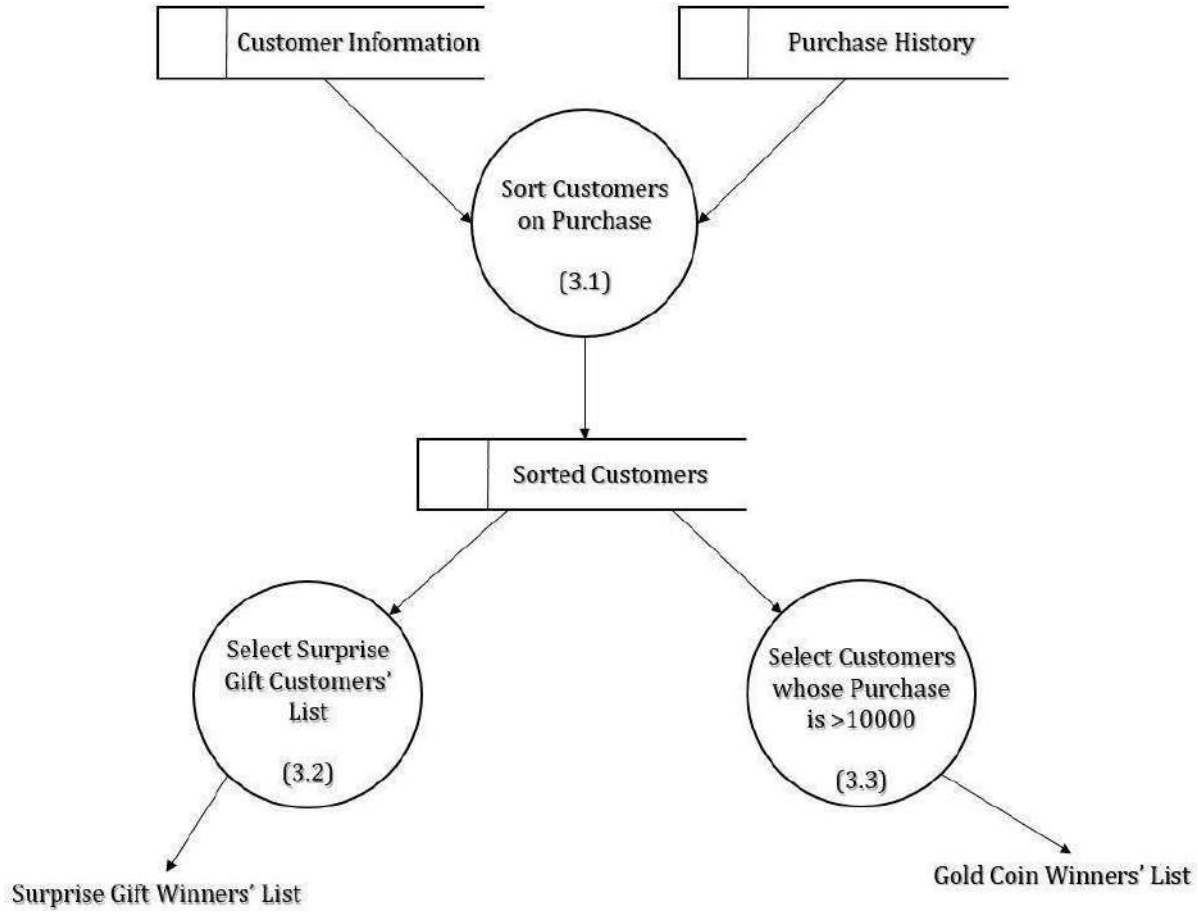


Fig – DFD level two of Process 3.0

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S

Retail Shopping Management Software

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement – It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model – Incremental model

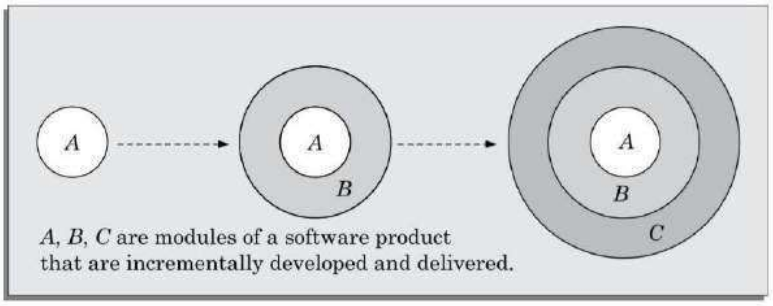


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module –** In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module –** In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module –** In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

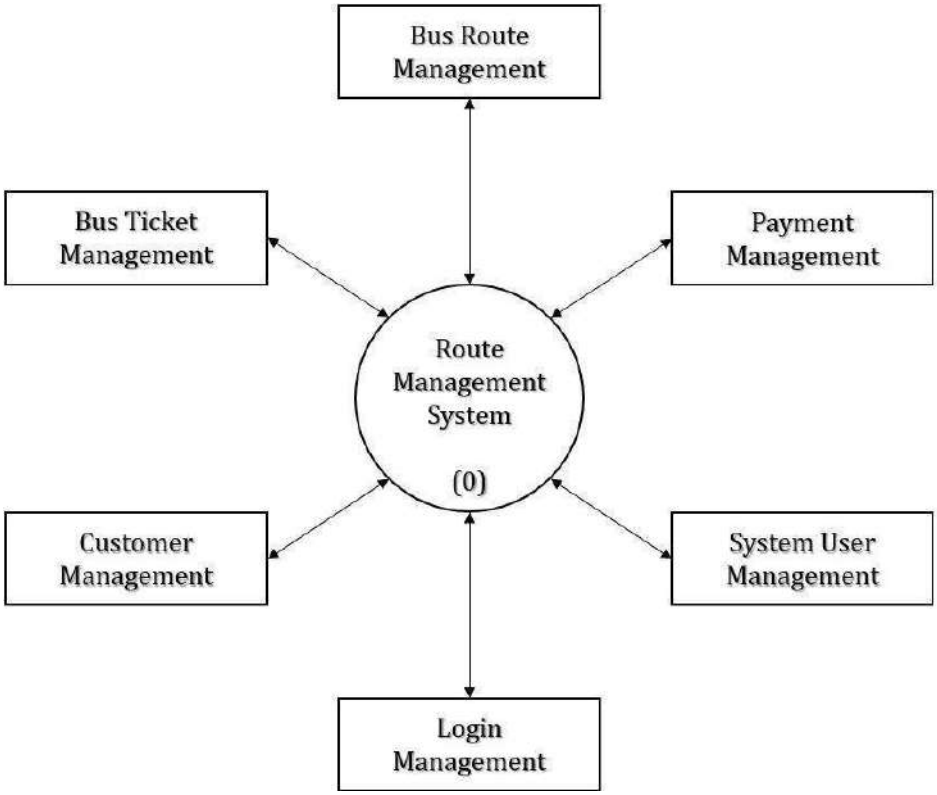


Fig – DFD level zero

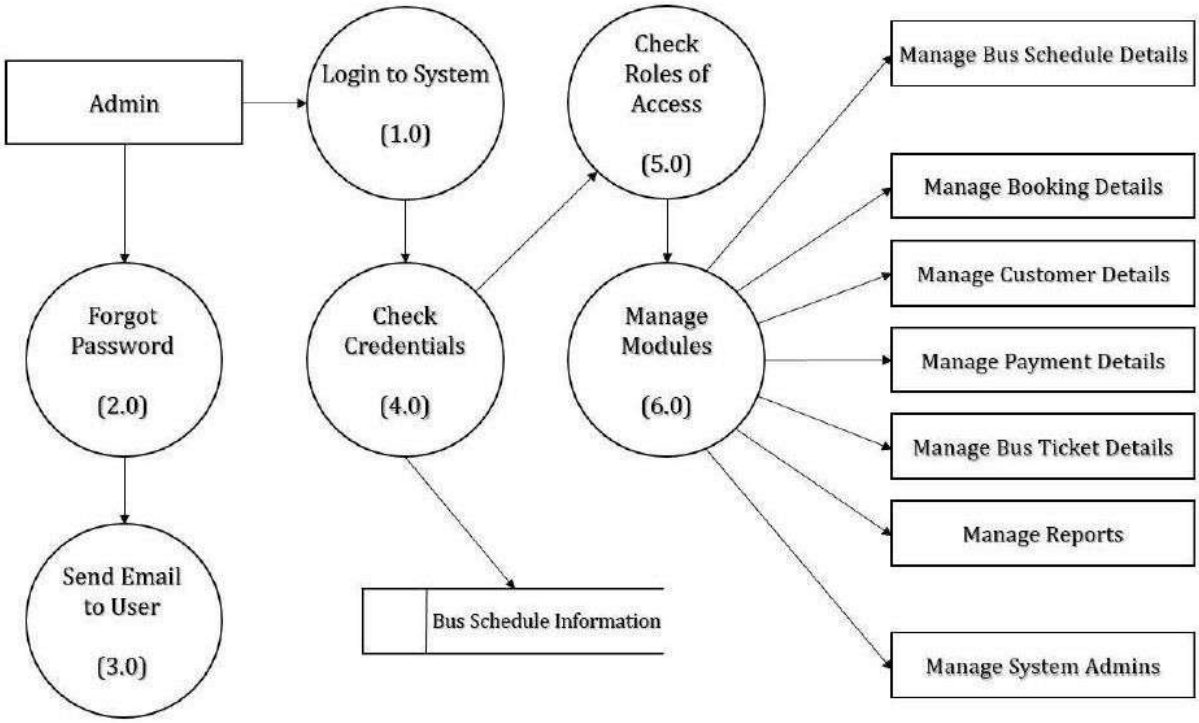


Fig – DFD level one

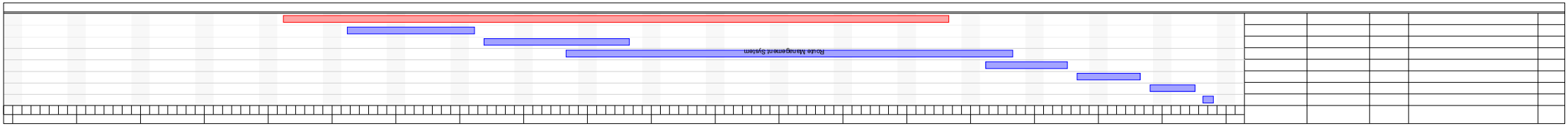
Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S


Route Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement – Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model – Incremental development model

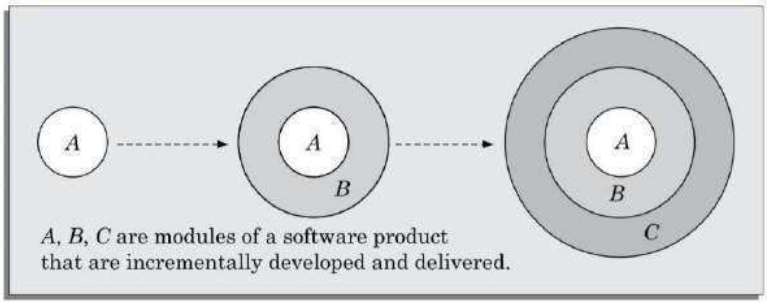


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module –** In this module, a patient is registered and the profile is maintained.

   Input: patient details
   Output: unique profile

2. **Doctor Module –** In this module, the details of doctors are maintained.

   Input: patient details
   Output: updated database

3. **Diagnosis Module –** In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input: patient details, doctor details, test details
   Output: diagnosed disease, updated database

4. **Report Module –** In this module, patient records, medicine and drug records, test records are processed and reports are generated.

   Input: patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

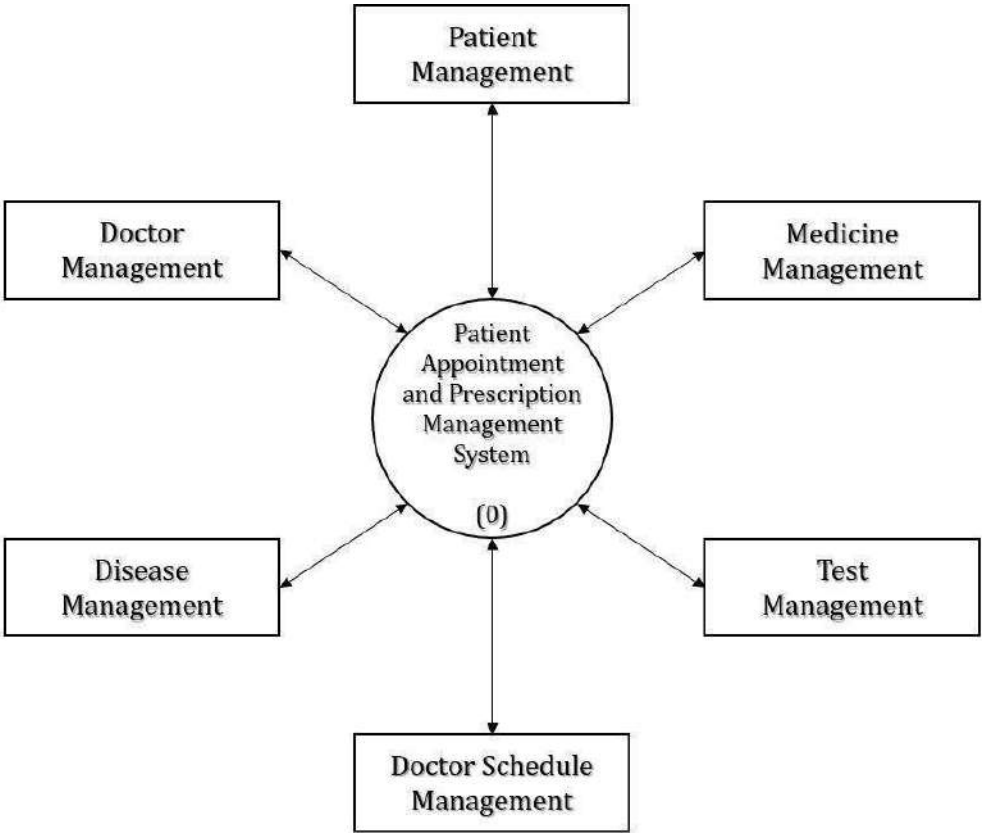


Fig – DFD level zero

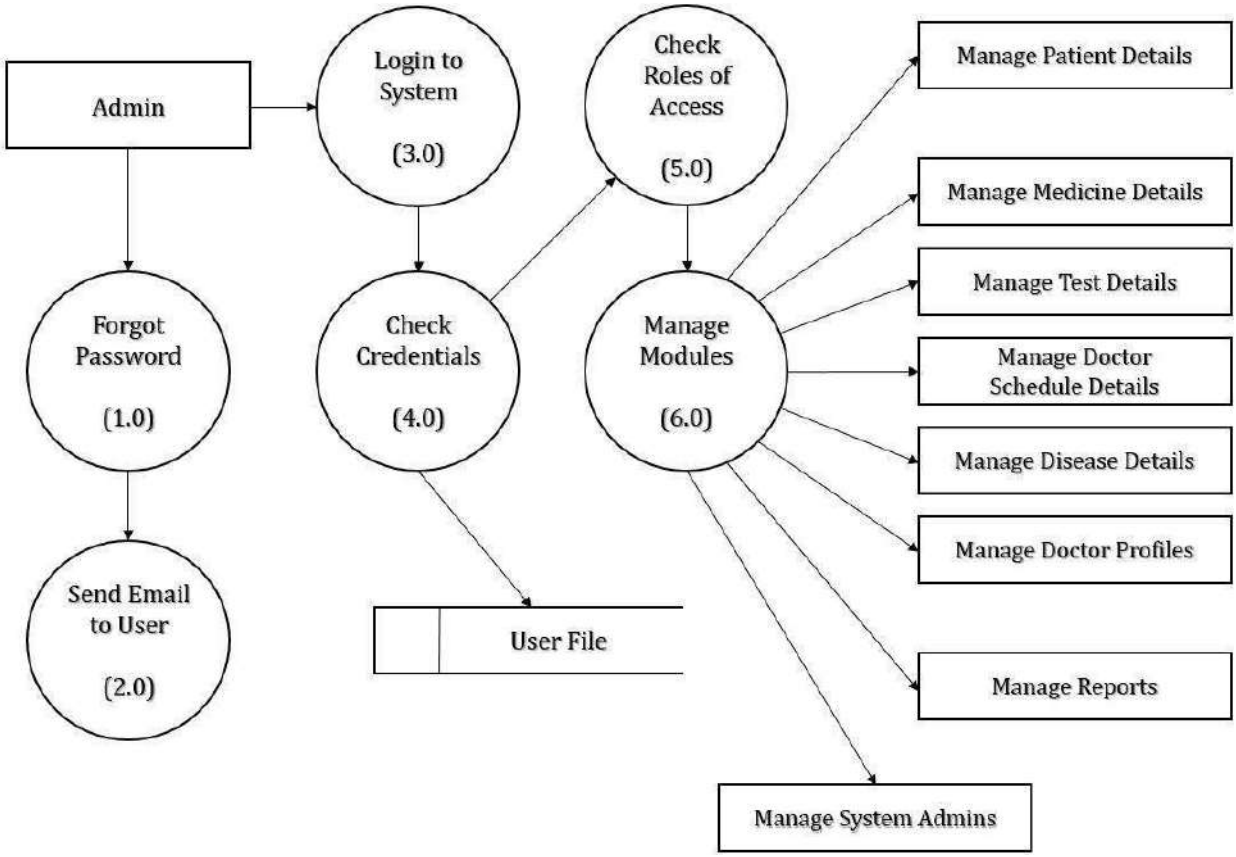


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 | 18 Apr 21 | 25 Apr 21 | 2 May 21 | 9 May 21 | 16 May 21 | 23 May 21 | 30 May 21 | 6 Jun 21 | 13 Jun 21 | 20 Jun 21 | 27 Jun 21 | 4 Jul 21 | 11 Jul 21 | 18 Jul 21 | 25 Jul 21 | 1 Aug 21 | 8 Aug 21 | 15 Aug 21 | 22

F S S M T W T F S S

Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement – Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model – Spiral development model

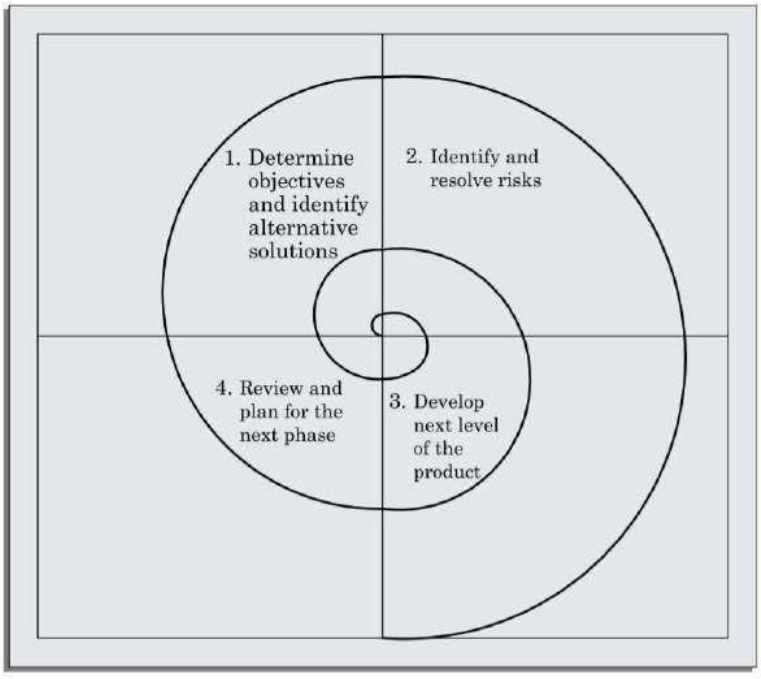


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module –** In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input: guest details
   Output: unique profile, updated database

2. **Catering Module –** In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input: guest details, room number, token number
   Output: updated database

3. **Bill Module –** In this module, the total bill is generated when the guest prepares to check out.

   Input: guest details, room number, token number, food items consumed
   Output: generated bill

Design Engineering

Data Flow Diagrams (DFDs)

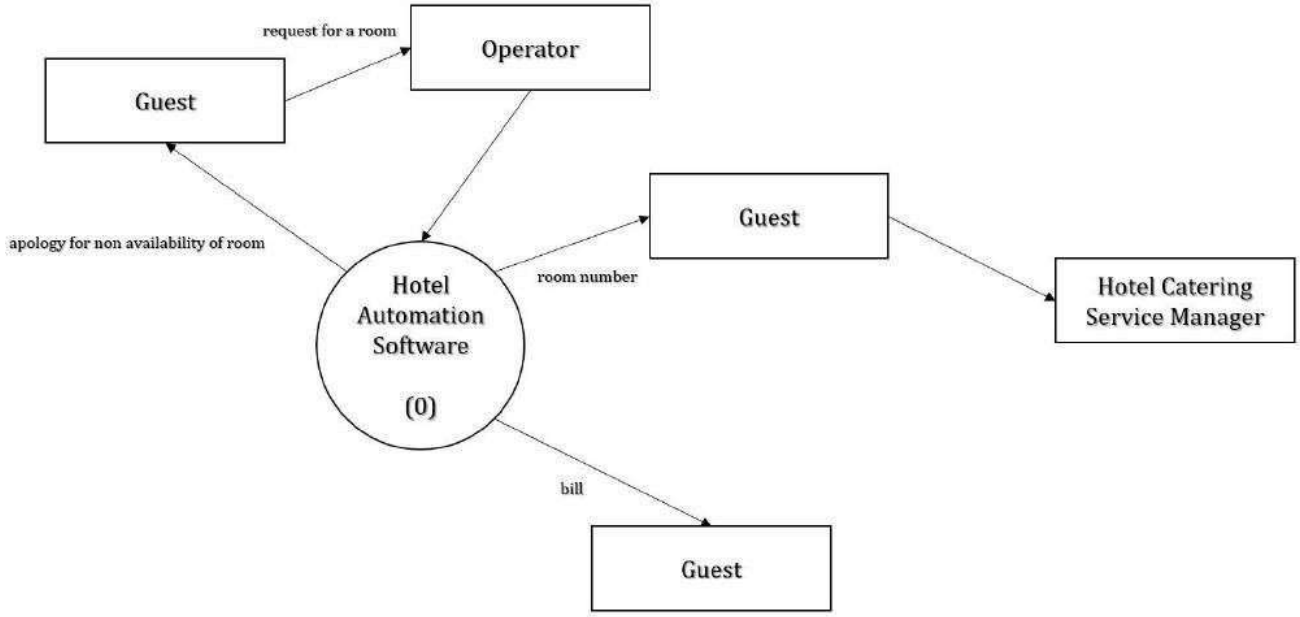


Fig – DFD level zero

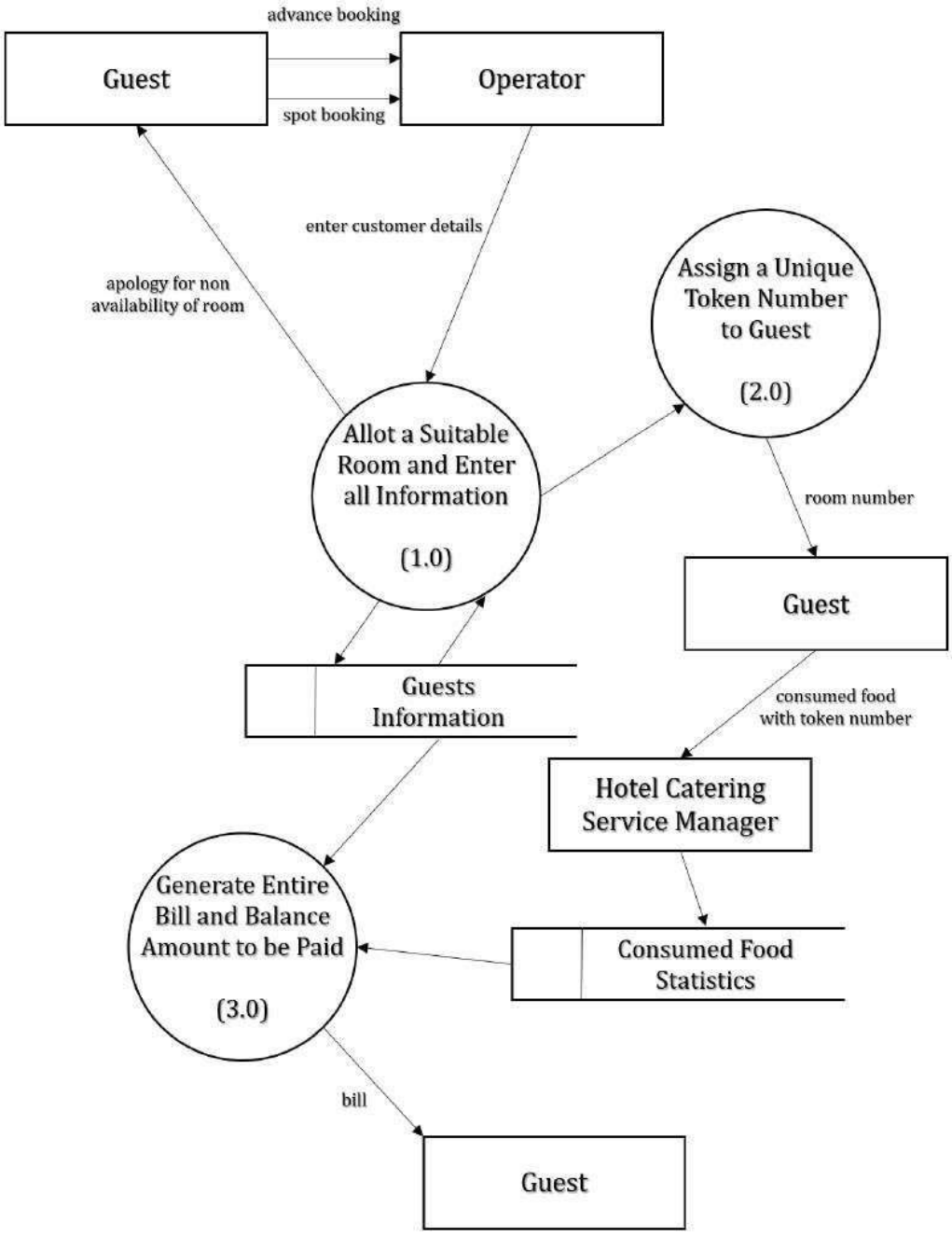


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S

Hotel Automation Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement – Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

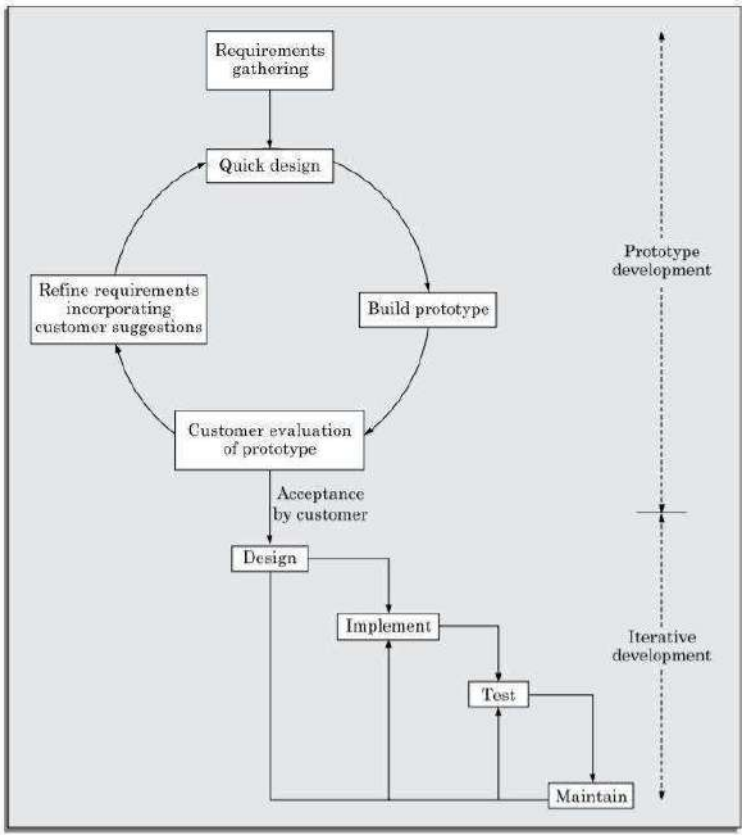


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module –** In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input: admin details
   Output: unique profile, granted level of access

2. **Complain Module –** In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output: filed FIR, updated database

3. **Crime Module –** In this module, the details of the crime are maintained and chargesheet is filed.

   Input: complaint details, FIR details
   Output: filed chargesheet

4. **Court Module –** In this module, the case is taken to the court and legal action is taken accordingly.

   Input: complaint details, FIR details, chargesheet details
   Output: legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

Fig – DFD level zero

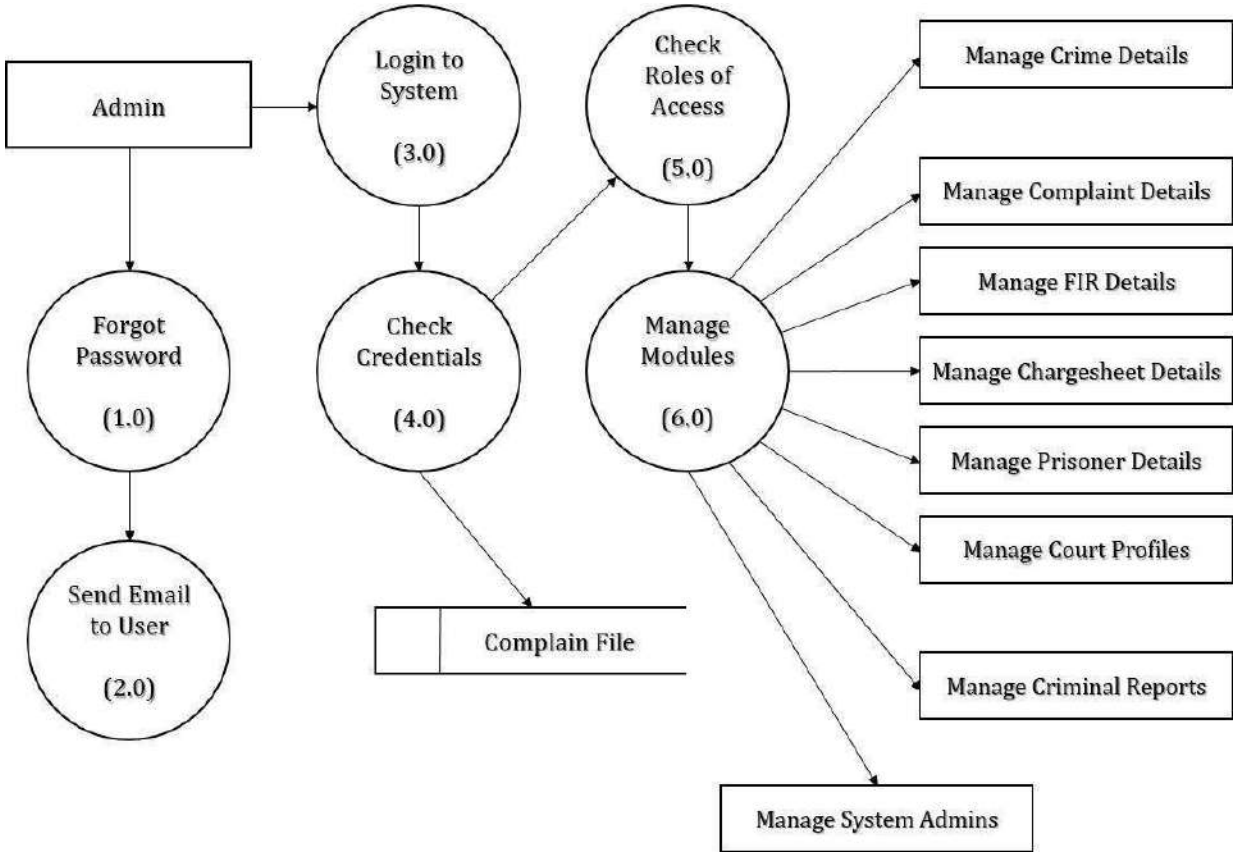


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S

Criminal Record Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement – Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model – Spiral development model

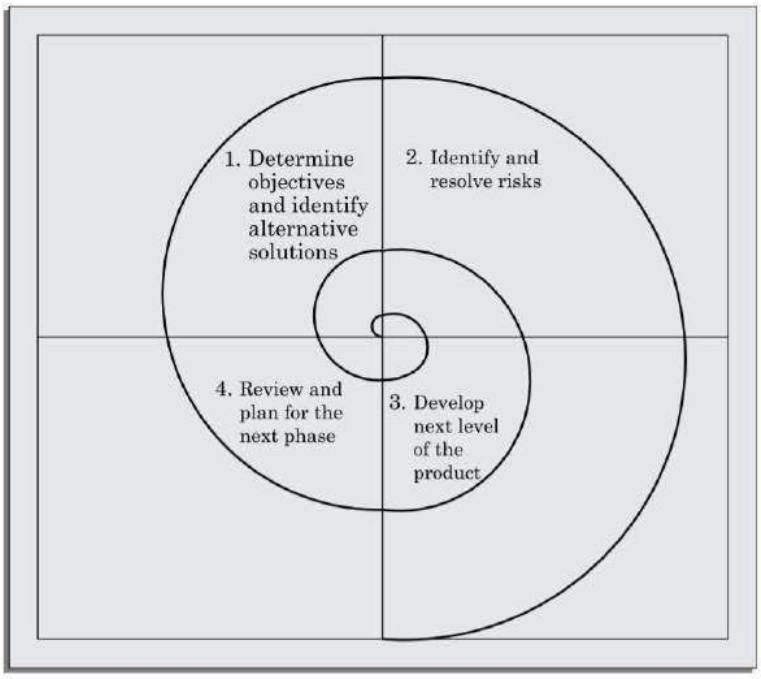


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module –** In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module –** In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module –** In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module –** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

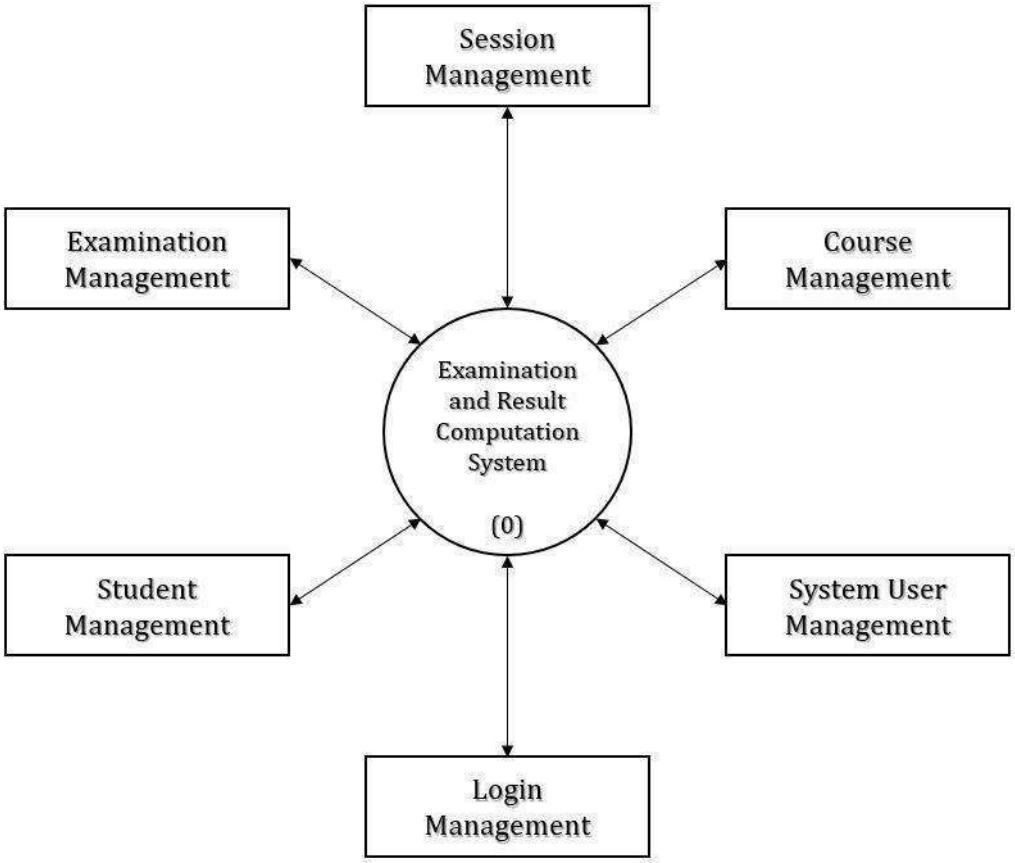


Fig – DFD level zero

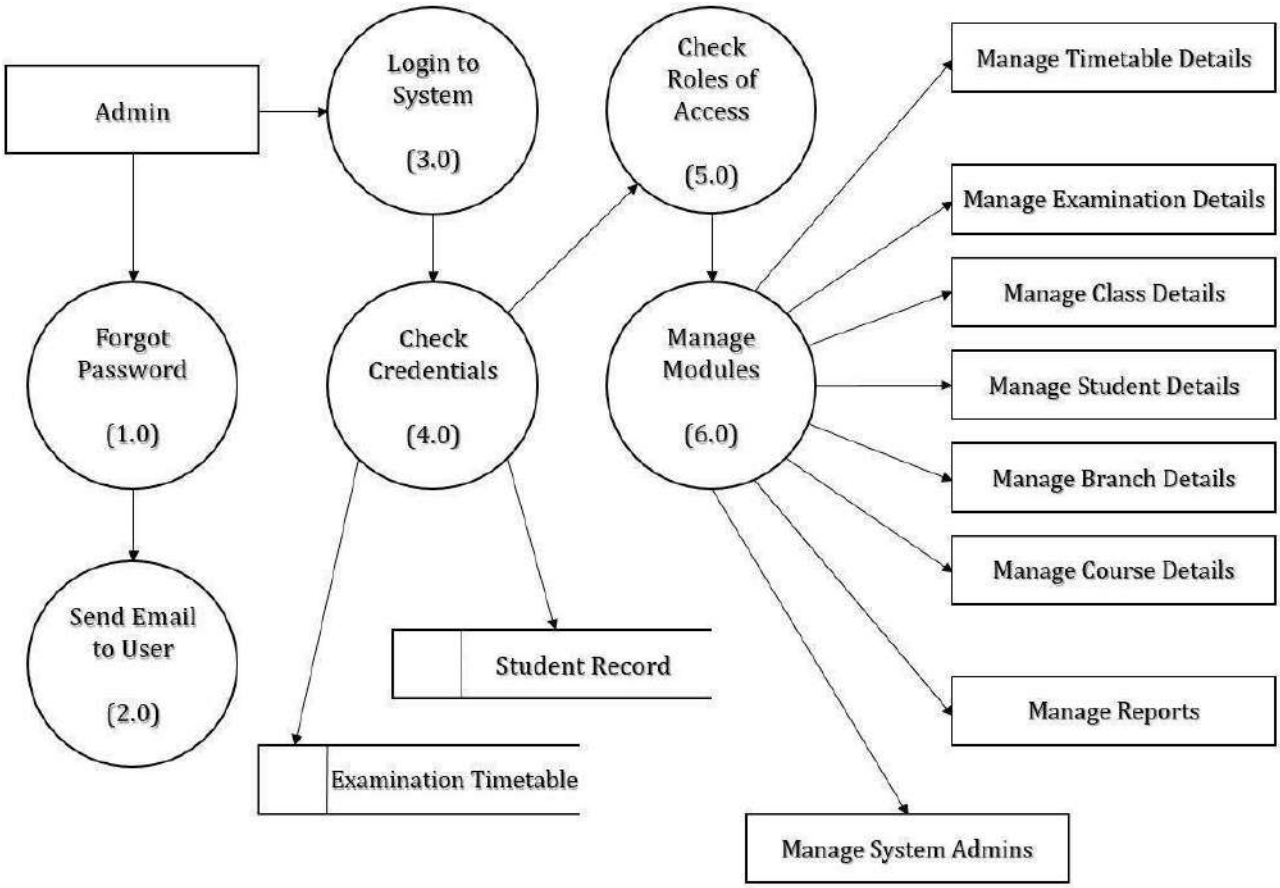


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S

Examination and Result Computation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PARKING ALLOCATION SYATEM

Problem Statement – Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model – Spiral development model

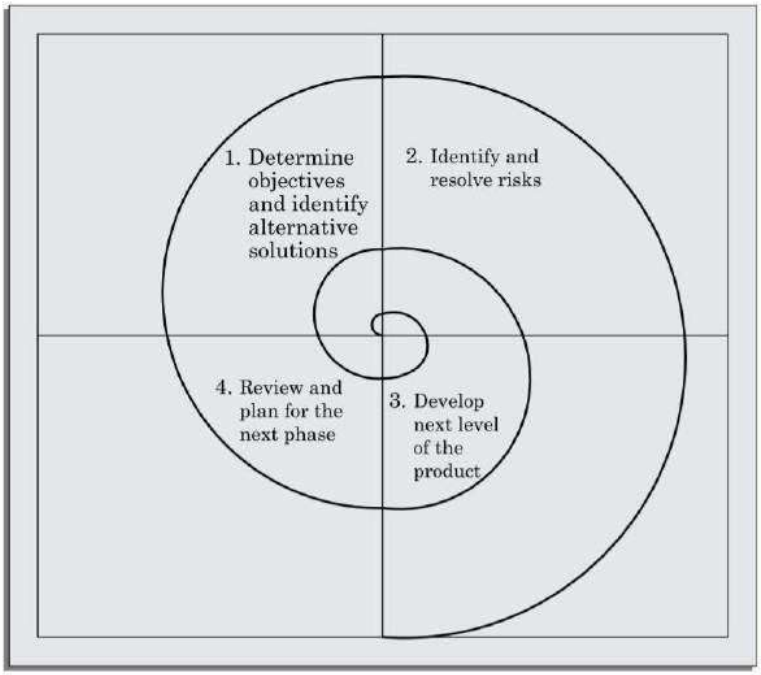


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module –** In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module –** In this module, parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module –** In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

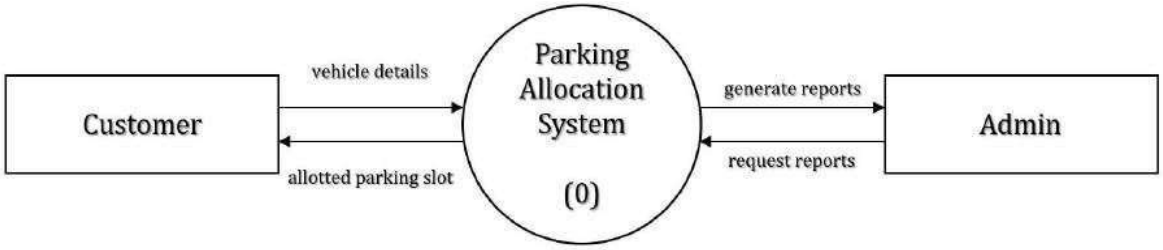


Fig – DFD level zero

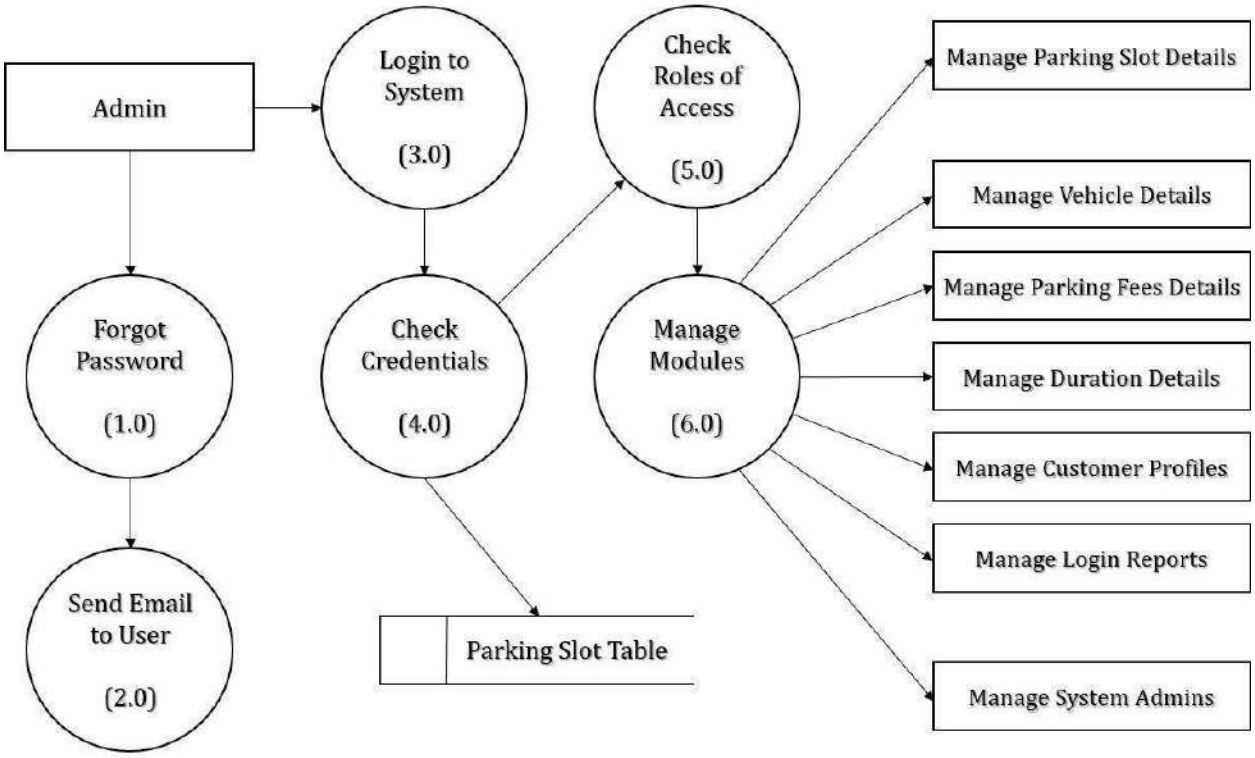


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 18 Apr 21 25 Apr 21 2 May 21 9 May 21 16 May 21 23 May 21 30 May 21 6 Jun 21 13 Jun 21 20 Jun 21 27 Jun 21 4 Jul 21 11 Jul 21 18 Jul 21 25 Jul 21 1 Aug 21 8 Aug 21 15 Aug 21 22

F S S M T W T F S S

Parking Allocation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

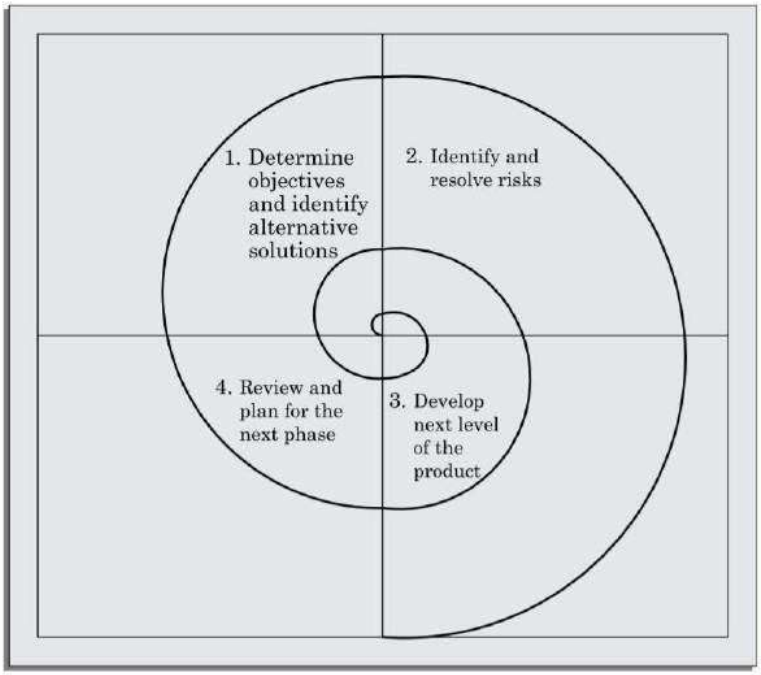


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

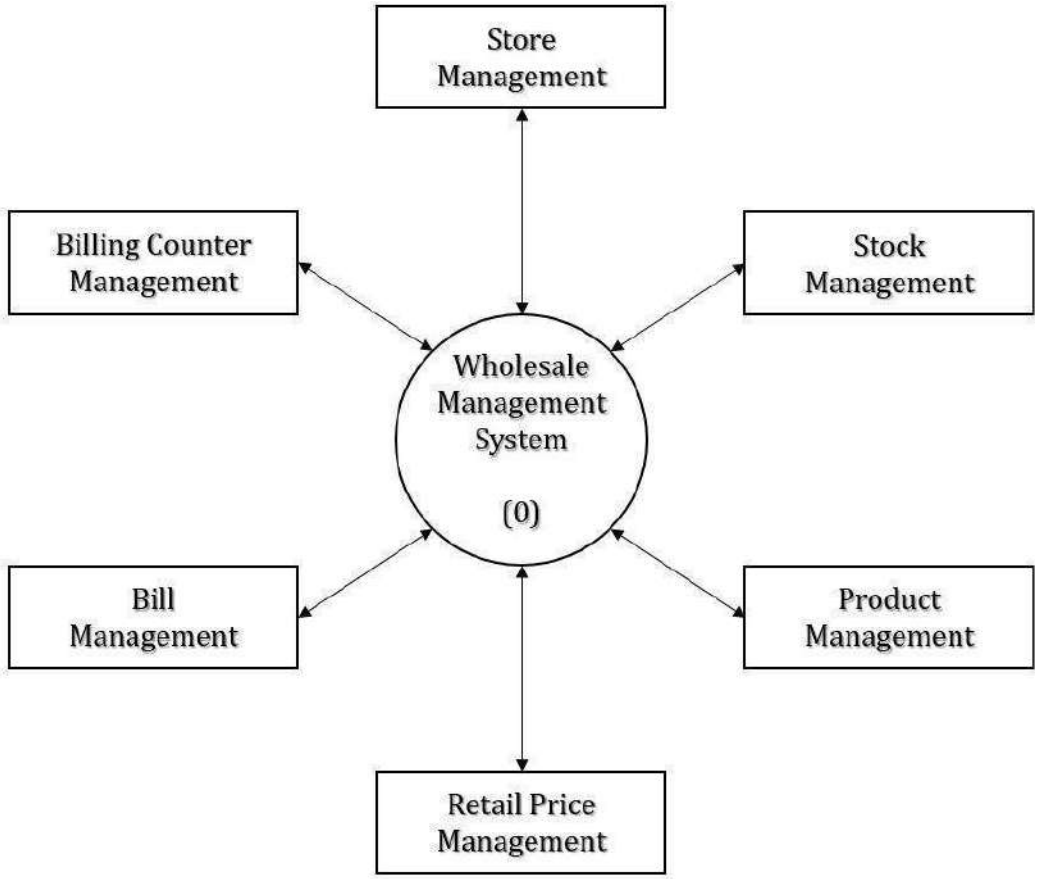


Fig – DFD level zero

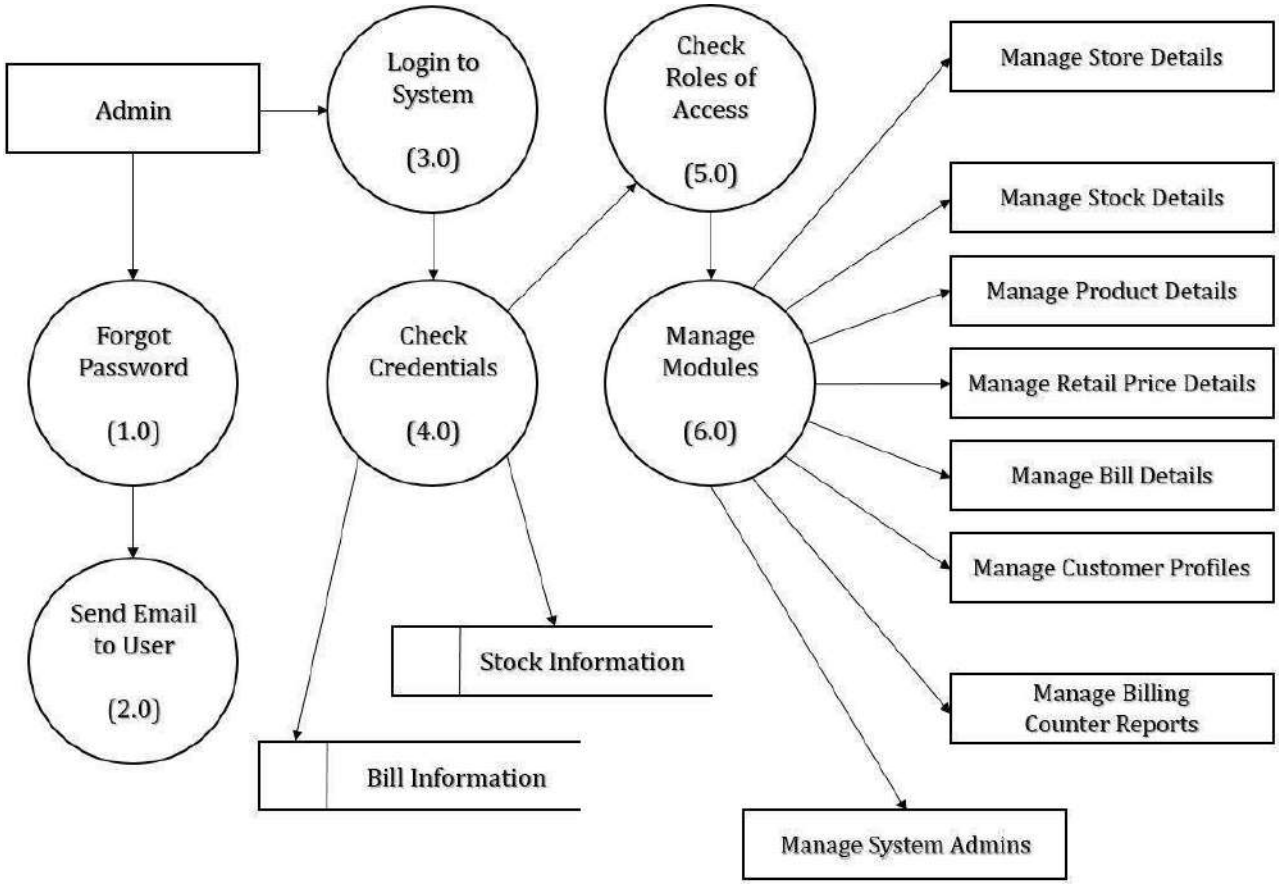


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

11 Apr 21 · 18 Apr 21 · 25 Apr 21 · 2 May 21 · 9 May 21 · 16 May 21 · 23 May 21 · 30 May 21 · 6 Jun 21 · 13 Jun 21 · 20 Jun 21 · 27 Jun 21 · 4 Jul 21 · 11 Jul 21 · 18 Jul 21 · 25 Jul 21 · 1 Aug 21 · 8 Aug 21 · 15 Aug 21 · 22

F S S M T W T F S S

Wholesale Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Microsoft Azure

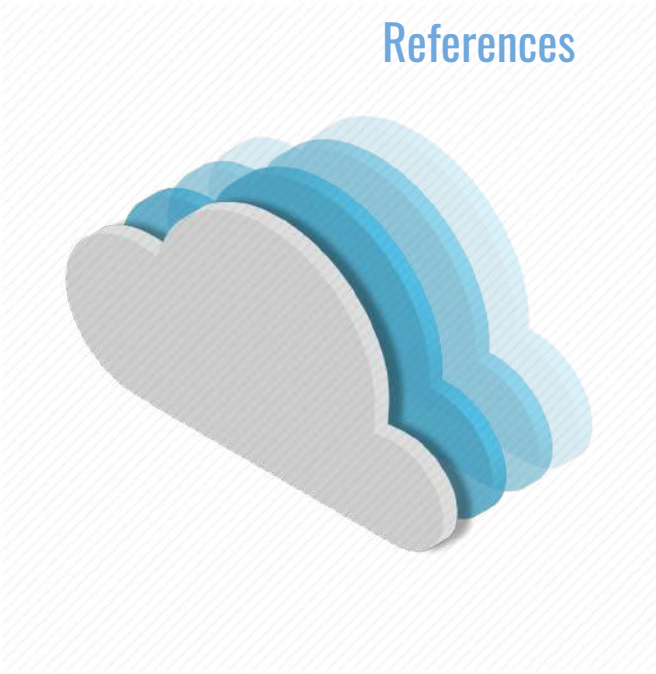
Case Study

Name Venkatesh Aanand
Dept. Computer Science
Semester V
Subject Cloud Computing (DSE3)
Roll 711
Reg.No. A01-1112-117-009-2019

Supervisor's Name: Dr. Chayan Halder

Ramakrishna Mission Vivekananda Centenary College Rahara

CONTENTS



What is Cloud Computing ?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end users. Following are the working models of cloud computing:

- ➢ Deployment Models
- ➢ Service Models

Fig 1.1 Representation of Cloud Computing

What are the types of Deployment Model ?

Public Cloud

Public clouds are owned and operated by third-party cloud service providers, which deliver their computing resources, like servers and storage, over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software, and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

Private Cloud

Private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company's on-site datacenter. Some companies also pay third-party service providers to host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

Hybrid Cloud

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options, and helps optimize your existing infrastructure, security, and compliance.

What are the types of Service Model ?

Infrastructure as a service (IaaS)

The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps.

Software as a service (SaaS)

Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching.

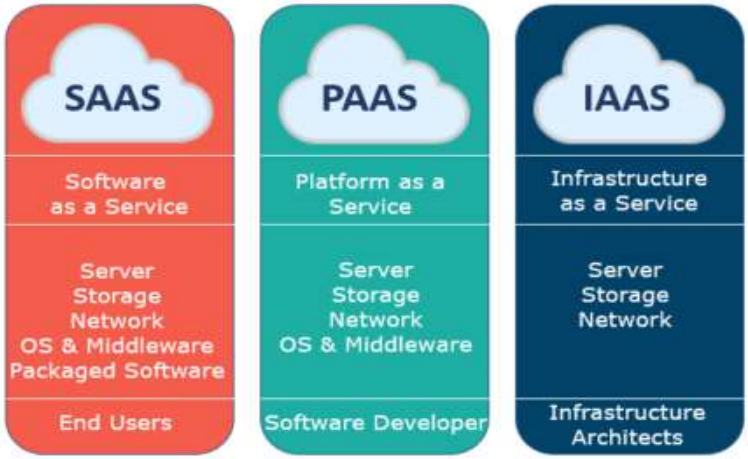


Fig 3.1 Different Types of Cloud as a Service

What is Microsoft Azure ?

Microsoft Windows Azure is a cloud operating system built on top of Microsoft datacenters' infrastructure and provides developers with a collection of services for building applications with cloud technology. Services range from compute, storage, and networking to application connectivity, access control, and business intelligence. Any application that is built on the Microsoft technology can be scaled using the Azure platform, which integrates the scalability features into the common Microsoft technologies such as Microsoft Windows Server 2008, SQL Server, and ASP.NET. The services provided are managed and controlled through the Windows Azure Management Portal, which acts as an administrative console for all the services offered by the Azure platform. In this section, we present the core features of the major services available with Azure.

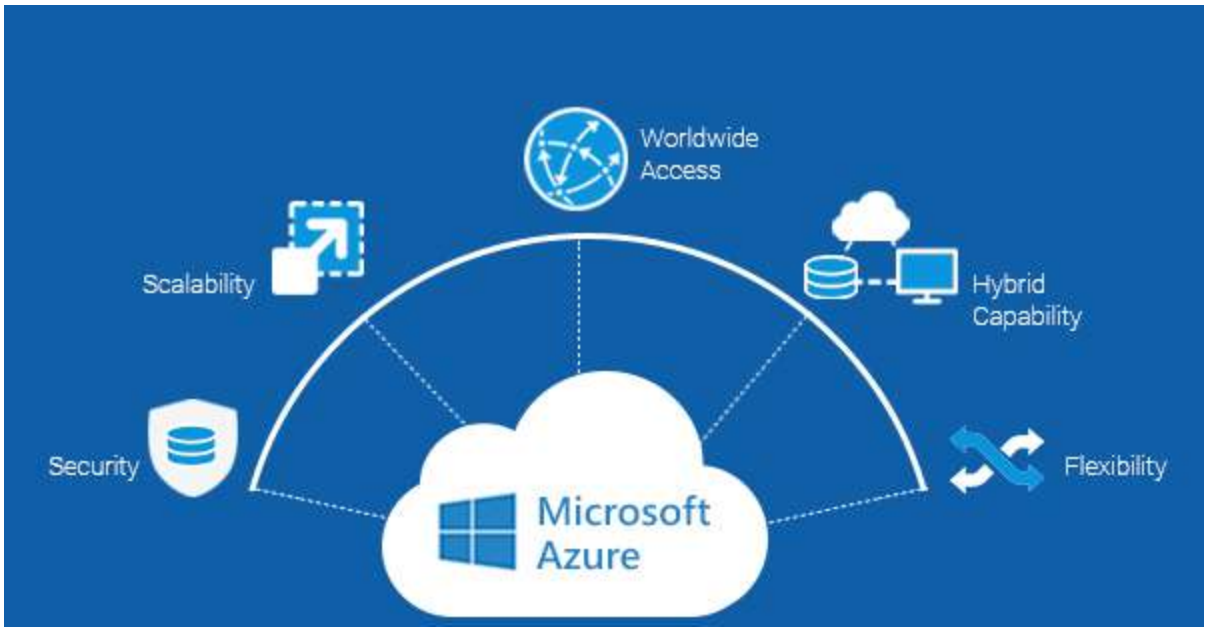


Fig 4.1 Representation of Microsoft Azure

How does Microsoft Azure work ?

The Windows Azure platform is made up of a foundation layer and a set of developer services that can be used to build scalable applications. These services cover compute, storage, networking, and identity management, which are tied together by middleware called AppFabric. This scalable computing environment is hosted within Microsoft datacenters and accessible through the Windows Azure Management Portal. Alternatively, developers can recreate a Windows Azure environment (with limited capabilities) on their own machines for development and testing purposes. In this section, we provide an overview of the Azure middleware and its services.

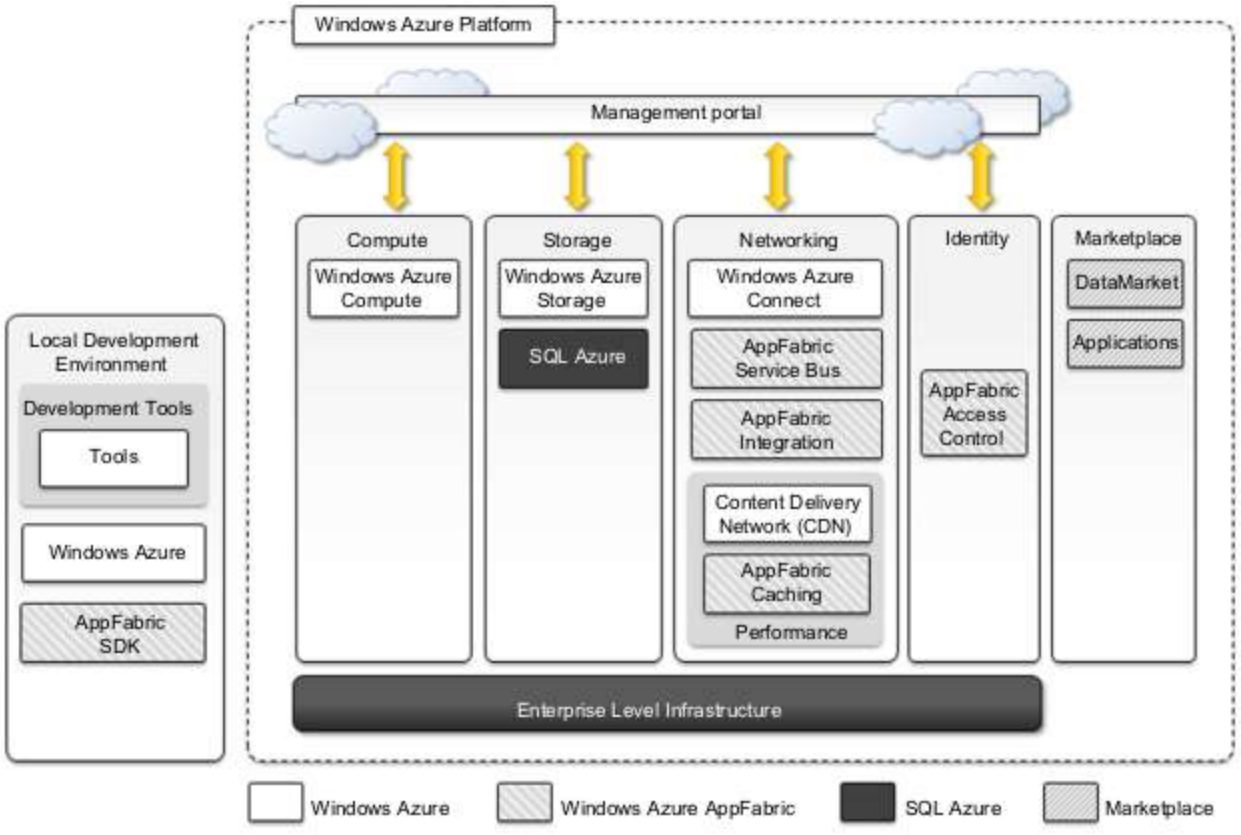


Fig 5.1 Microsoft Azure Platform Architecture

What are the services provided ?

Compute Services

Compute services are the core components of Microsoft Windows Azure, and they are delivered by means of the abstraction of roles. A role is a runtime environment that is customized for a specific computer task. Roles are managed by the Azure operating system and instantiated on demand in order to address surges in application demand. Currently, there are three different roles: Web role, Worker role, and Virtual Machine (VM) role.

Web Role

The Web role is designed to implement scalable Web applications. Web roles represent the units of deployment of Web applications within the Azure infrastructure. They are hosted on the IIS 7 Web Server, which is a component of the infrastructure that supports Azure. When Azure detects peak loads in the request made to a given application, it instantiates multiple Web roles for that application and distributes the load among them by means of a load balancer.

Worker Role

Worker roles are designed to host general compute services on Azure. They can be used to quickly provide compute power or to host services that do not communicate with the external world through HTTP. A common practice for Worker roles is to use them to provide background processing for Web applications developed with Web roles.

Virtual Machine Role

The Virtual Machine role allows developers to fully control the computing stack of their computer service by defining a custom image of the Windows Server 2008 R2 operating system and all the service stack required by their applications. The Virtual Machine role is based on the Windows Hyper-V virtualization technology, which is natively integrated in the Windows server technology at the base of Azure.

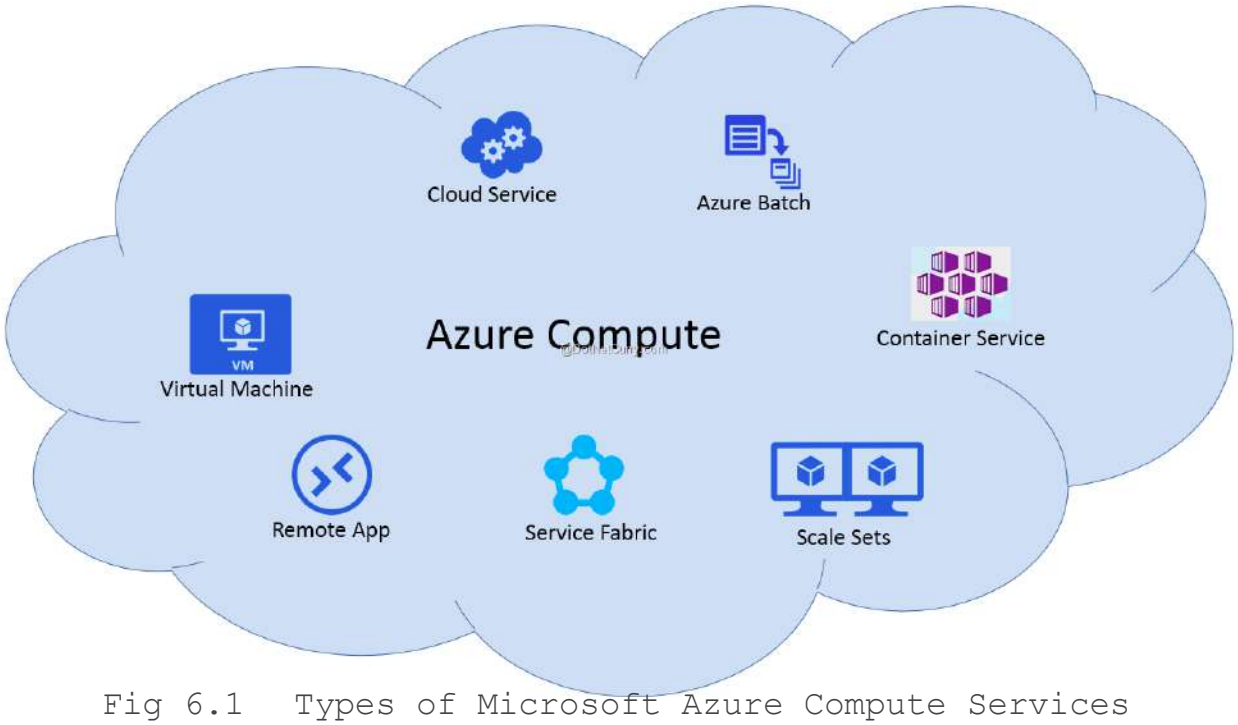


Fig 6.1 Types of Microsoft Azure Compute Services

Storage Services

Compute resources are equipped with local storage in the form of a directory on the local file system that can be used to temporarily store information that is useful for the current execution cycle of a role. If the role is restarted and activated on a different physical machine, this information is lost. Windows Azure provides different types of storage solutions that complement compute services with a more durable and redundant option compared to local storage. Compared to local storage, these services can be accessed by multiple clients at the same time and from everywhere, thus becoming a general solution for storage.

Blobs

Azure allows storing large amounts of data in the form of binary large objects (BLOBs) by means of the blobs service. This service is optimal to store large text or binary files. Two types of blobs are available Block blobs and Page blobs

Azure Drive

Page blobs can be used to store an entire file system in the form of a single Virtual Hard Drive (VHD) file. This can then be mounted as a part of the NTFS file system by Azure compute resources, thus providing persistent and durable storage. A page blob mounted as part of an NTFS tree is called an Azure Drive.

Tables

Tables constitute a semi structured storage solution, allowing users to store information in the form of entities with a collection of properties. Entities are stored as rows in the table and are identified by a key, which also constitutes the unique index built for the table. Users can insert, update, delete, and select a subset of the rows stored in the table. Unlike SQL tables, there are no schema enforcing constraints on the properties of entities and there is no facility for representing relationships among entities. For this reason, tables are more similar to spreadsheets rather than SQL tables.

Queues

Queue storage allows applications to communicate by exchanging messages through durable queues, thus avoiding lost or unprocessed messages. Applications enter messages into a queue, and other applications can read them in a first-in, first-out (FIFO) style.

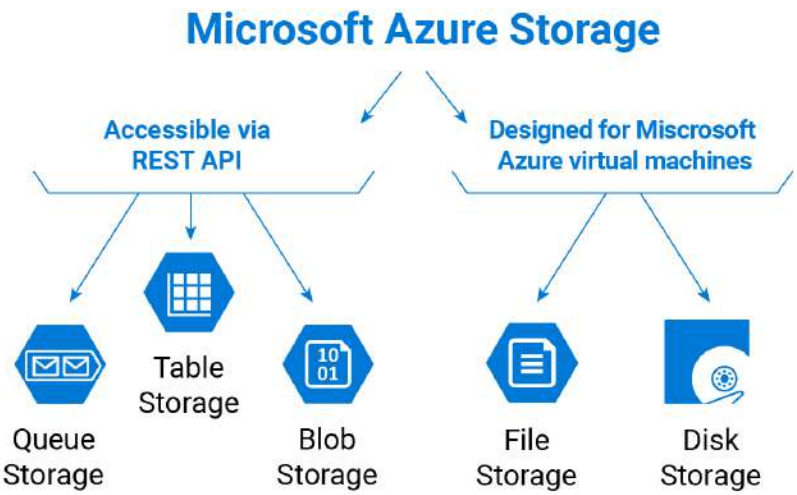


Fig 6.2 Types of Microsoft Azure Storage Services

Core Infrastructure: AppFabric

AppFabric is a comprehensive middleware for developing, deploying, and managing applications on the cloud or for integrating existing applications with cloud services. AppFabric implements an optimized infrastructure supporting scaling out and high availability; sandboxing and multi-tenancy; state management; and dynamic address resolution and routing. On top of this infrastructure, the middleware offers a collection of services that simplify many of the common tasks in a distributed application, such as communication, authentication and authorization, and data access.

Access Control

AppFabric provides the capability of encoding access control to resources in Web applications and services into a set of rules that are expressed outside the application code base. These rules give a great degree of flexibility in terms of the ability to secure components of the application and define access control policies for users and groups.

Service Bus

Service Bus constitutes the messaging and connectivity infrastructure provided with AppFabric for building distributed and disconnected applications in the Azure Cloud and between the private premises and the Azure Cloud. Service Bus allows applications to interact with different protocols and patterns over a reliable communication channel that guarantees delivery.

Azure cache

Windows Azure provides a set of durable storage solutions that allow applications to persist their data. These solutions are based on disk storage, which might constitute a bottleneck for the applications that need to gracefully scale along the clients' requests and dataset size dimensions. Azure Cache is a service that allows developers to quickly access data persisted on Windows Azure storage or in SQL Azure. The service implements a distributed in-memory cache of which the size can be dynamically adjusted by applications according to their needs.

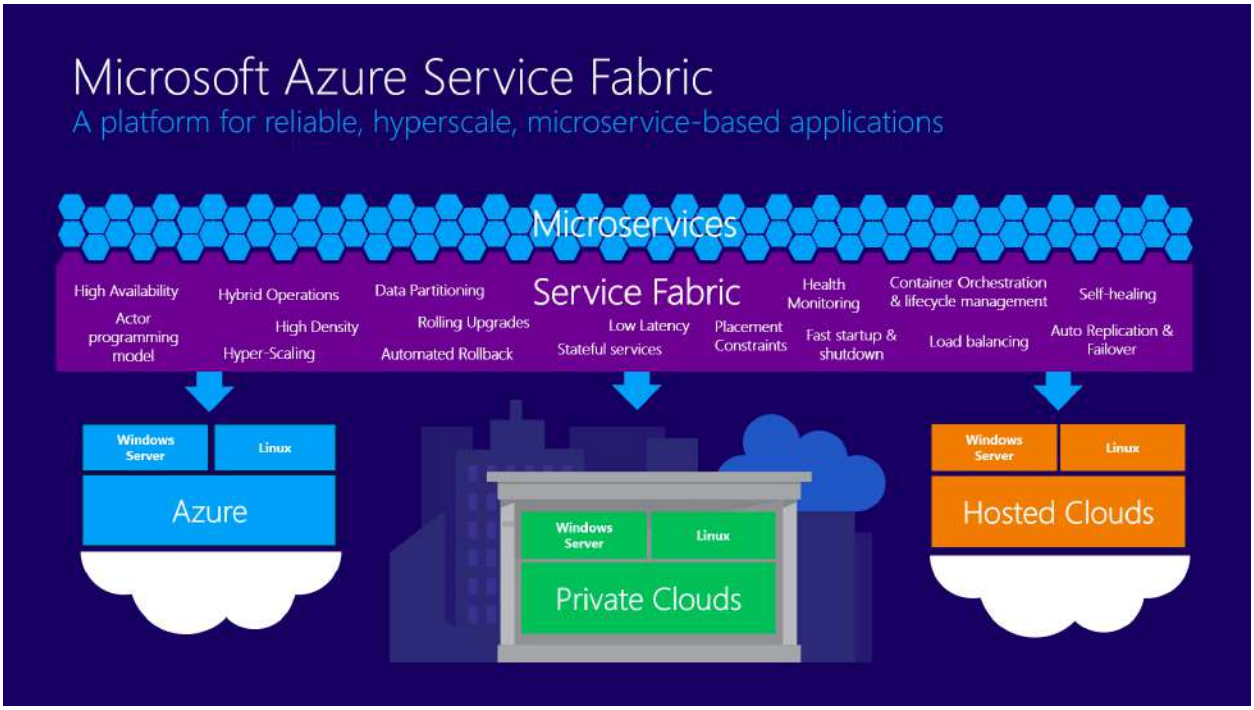


Fig 6.3 Representation of Microsoft Azure Service Fabric

Other Services

Compute, storage, and middleware services constitute the core components of the Windows Azure platform. Besides these, other services and components simplify the development and integration of applications with the Azure Cloud. An important area for these services is application connectivity, including virtual networking and content delivery.

Windows Azure Virtual Network

Networking services for applications are offered under the name Windows Azure Virtual Network, which includes Windows Azure Connect and Windows Azure Traffic Manager. Windows Azure Connect allows easy setup of IP-based network connectivity among machines hosted on the private premises and the roles deployed on the Azure Cloud. This service is particularly useful in the case of VM roles, where machines hosted in the Azure Cloud become part of the private network of the enterprise and can be managed with the same tools used in the private premises.

Windows Azure Content Delivery Network

Windows Azure Content Delivery Network (CDN) is the content delivery network solution that improves the content delivery capabilities of Windows Azure Storage and several other Microsoft services, such as Microsoft Windows Update and Bing maps. The service allows serving of Web objects (images, static HTML, CSS, and scripts) as well as streaming content by using a network of 24 locations distributed across the world.

SQL Azure

SQL Azure is a relational database service hosted on Windows Azure and built on the SQL Server technologies. The service extends the capabilities of SQL Server to the cloud and provides developers with a scalable, highly available, and fault-tolerant relational database. SQL Azure is accessible from either the Windows Azure Cloud or any other location that has access to the Azure Cloud. It is fully compatible with the interface exposed by SQL Server, so applications built for SQL Server can transparently migrate to SQL Azure. Moreover, the service is fully manageable using REST APIs, allowing developers to control databases deployed in the Azure Cloud as well as the firewall rules set up for their accessibility.

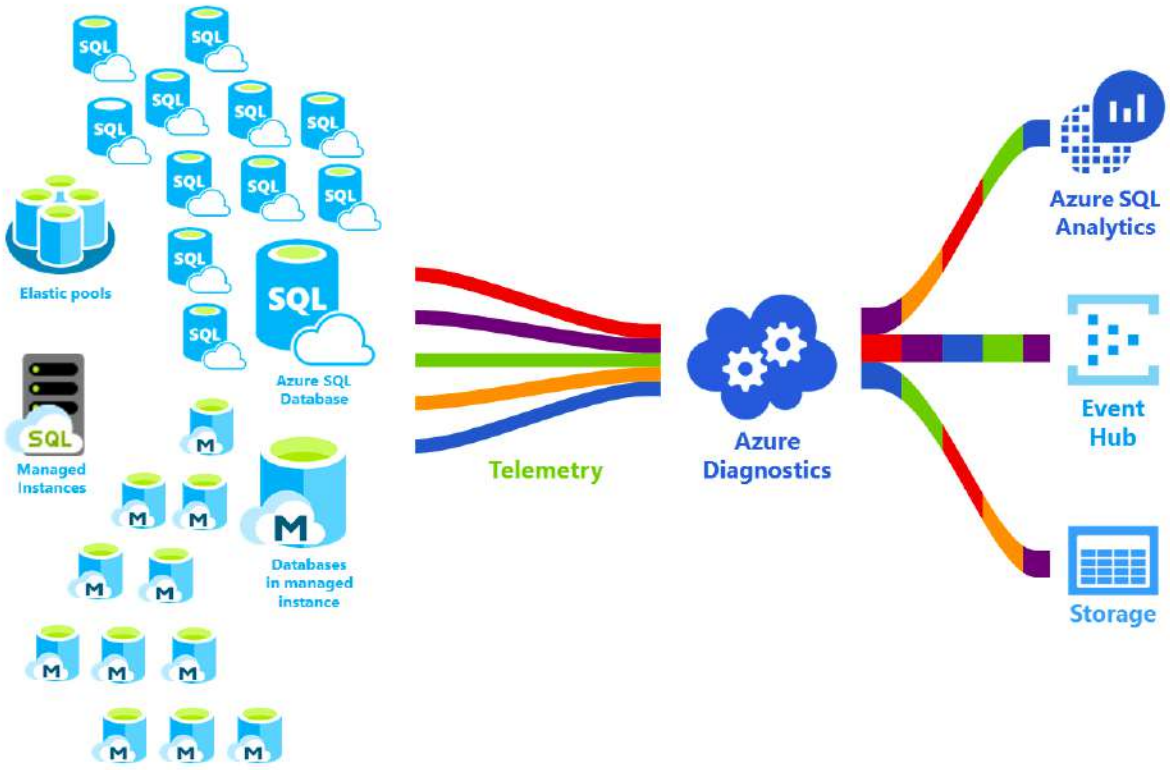


Fig 6.4 Representation of Microsoft Azure SQL

Microsoft Azure Usage and Revenue

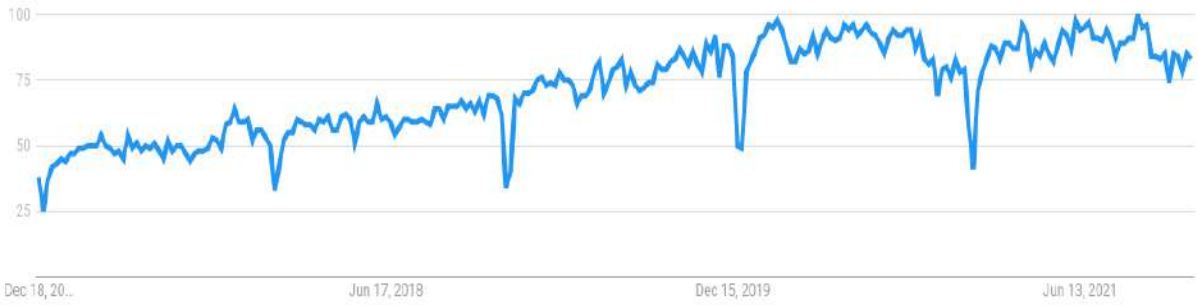


Fig 7.1 Popularity of Microsoft Azure over the last 5 years

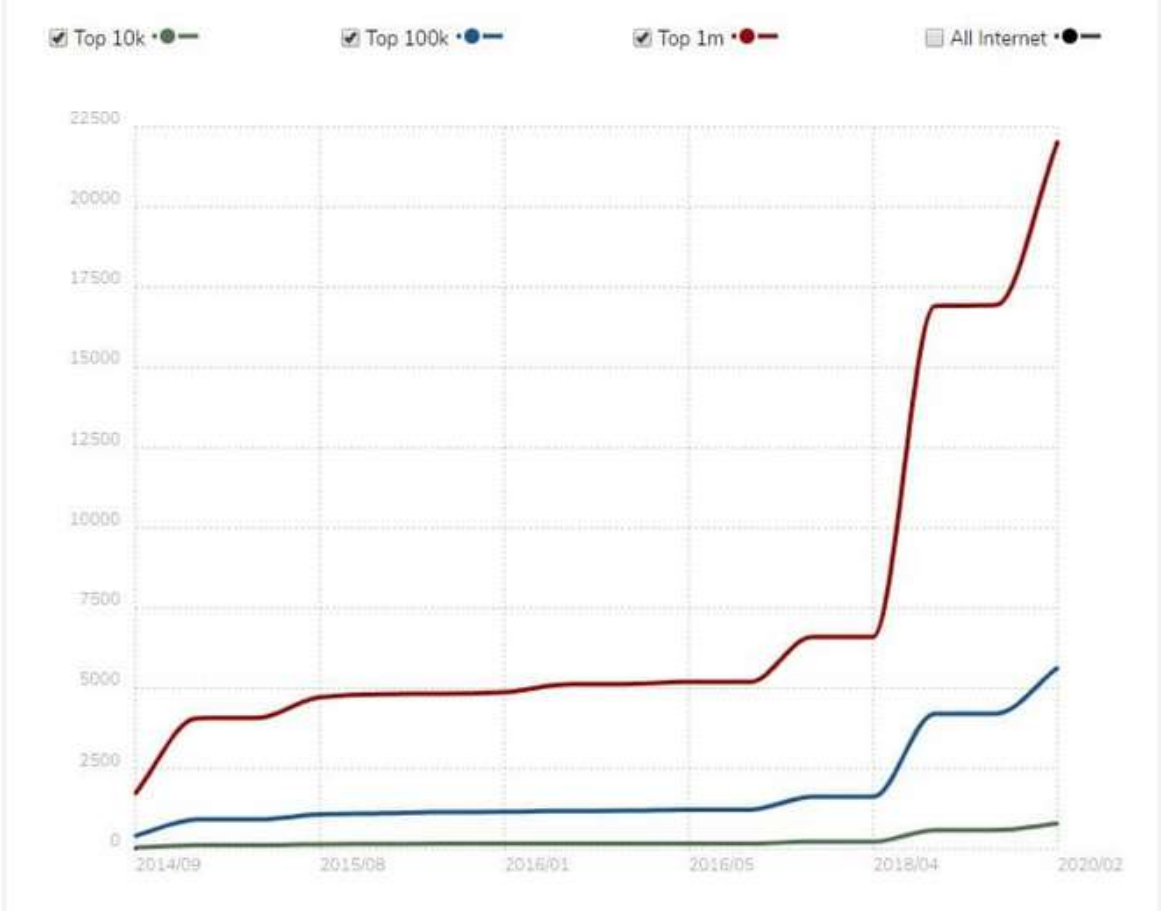


Fig 7.2 Microsoft Azure Usage Statistics

Fig 7.3 Microsoft Azure Revenue over the years

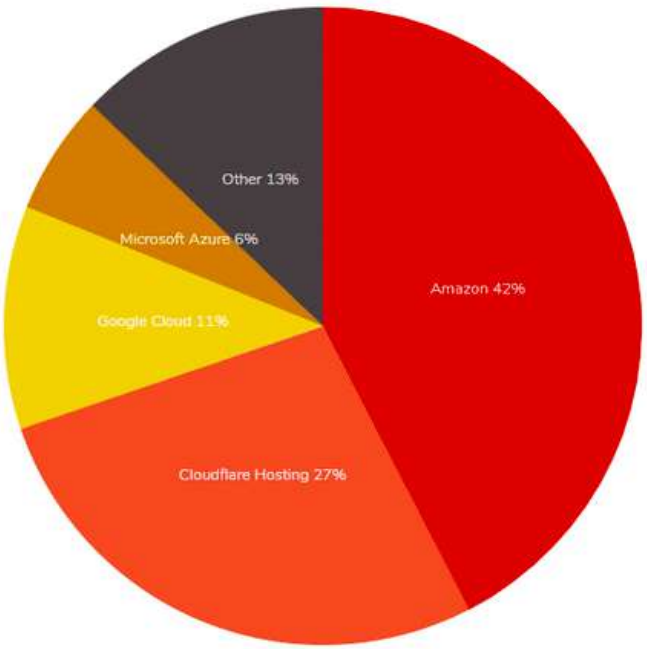


Fig 7.4 Cloud Hosting Usage Distribution

Advantages of Microsoft Azure

High Availability and Uptime

While Microsoft is not as large as AWS, Azure still is the second largest cloud platform in the industry today, with data centers found in several different regions, making it ideal for international businesses. Azure also promises a 99.995% uptime rate — an impressive accomplishment in the Cloud Industry.

Flexibility

Moving to the cloud can be an expensive experience, so it is important for businesses to make the most of their cloud platform once they are there. This flexibility is important, as it will enable you to scale up your projects as your business continues to grow. Azure proves to be an easily scalable platform and barely a few clicks of a button will get you the additional licenses you may need. Imagine being able to scale down your software environment over the weekend or scaling up only for your busiest days of the year.

Security

One of the most appealing features that Azure has to offer its clients is a state of the art security system following a ADADSC approach: Detect, Assess, Diagnose, Stabilize, and Close. They have proven to be the leading force in IaaS security and have received multiple compliance certifications for their high standards. Their security features are both reliable and user-friendly with protections like multi-factor authentication and password requirements.

Disadvantages of Microsoft Azure

Complexity

As a SaaS platform, Azure can easily become an extremely complicated environment for larger companies. Before the cloud, there was an extremely rigorous process when it came to purchasing more licenses, usually in the form of a negotiation or a contract renewal. Many companies do not have any sort of processes to regulate the spending of employees when confronted with their cloud platform. It will require management and strict processes to make sure purchasing is controlled, environments are well managed, and projects are closed after they have reached their conclusion.

Data Transfer

Azure services are all subject to data transfer fees that are often the cause of stacked hidden fees. This is not unique to Azure as all of the large cloud services like AWS and Google do this same gouging of their customer base. This separate fee for in and out data can prove quite costly for large companies, so you should be aware of this to avoid any surprises.

Support

Despite their high-quality products and global reach, Microsoft is not very good at dealing with the sheer volume of their customers and treating each customer as a unique individual. However, as a cloud service provider, that is one thing that Microsoft will have to do on a regular basis as companies run into technical issues and server problems that must be handled quickly.

Conclusion

Microsoft Azure is Microsoft's solution for developing cloud computing applications. Azure is an implementation of the PaaS layer and provides the developer with a collection of services and scalable middleware hosted on Microsoft datacenters that address compute, storage, networking, and identity management needs of applications. The services Azure offers can be used either individually or all together for building both applications that integrate cloud features and elastic computing systems completely hosted in the cloud.

The core components of the platform are composed of compute services, storage services, and middleware. Compute services are based on the abstraction of roles, which identify a sandboxed environment where developers can build their distributed and scalable components. These roles are useful for Web applications, back-end processing, and virtual computing. Storage services include solutions for static and dynamic content, which is organized in the form of tables with fewer constraints than those imposed by the relational model. These and other services are implemented and made available through AppFabric, which constitutes the distributed and scalable middleware of Azure. The platform is mostly based on the .NET technology and Windows systems, even though other technologies and systems can be supported. For this reason, Azure constitutes the solution of choice for migrating to the cloud applications that are already based on the .NET technology.

References

- Mastering CLoud Computing Foundations and Applications Programming by Rajkumar Buyya, Christian Vecchiola, S. Thamar Selvi
- Microsoft Azure Official Website https://azure.microsoft.com/en-us/overview/

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Santanu Mukherjee
Roll Number: 713
Exam roll number: 2021141295
Registration number: A01-1112-117- 011-2019
Semester: V

Department: Computer Science

Supervisor: Prof. Manas Pal

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a “Customer Number” (CN). A customer can present ‘CN’ to customer staff when he makes purchases. In this case, the value of his purchase is credited against its ‘CN’. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

REQUIREMENT ANALYSIS (SRS) :

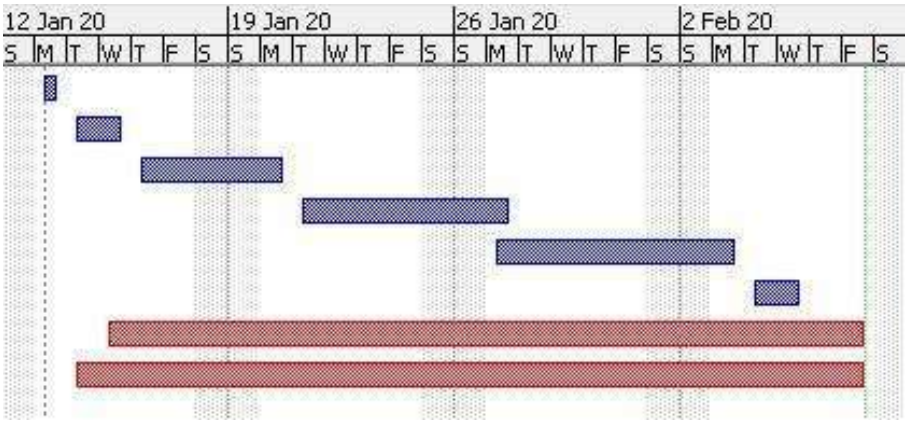
As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account’s purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh

the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



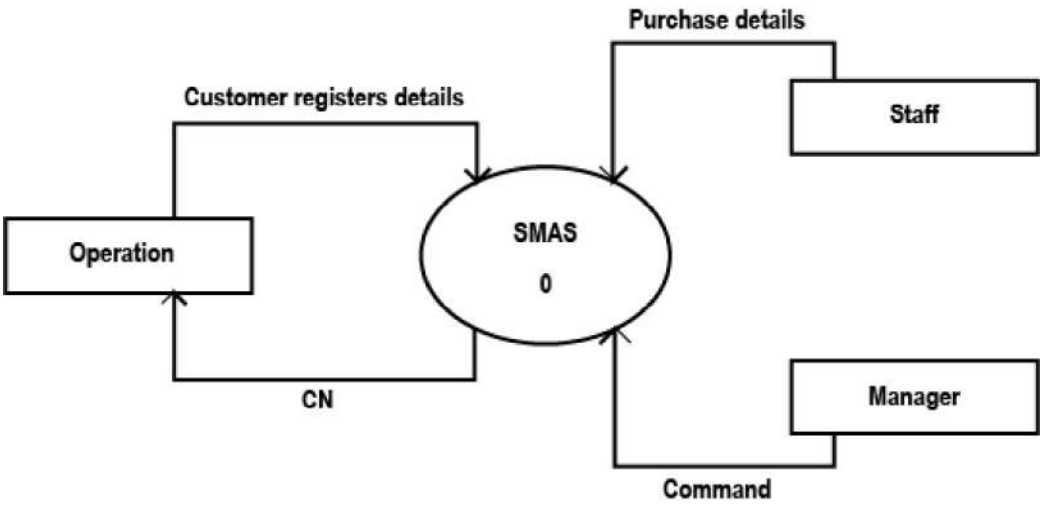
- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.

○ Risk leverage =

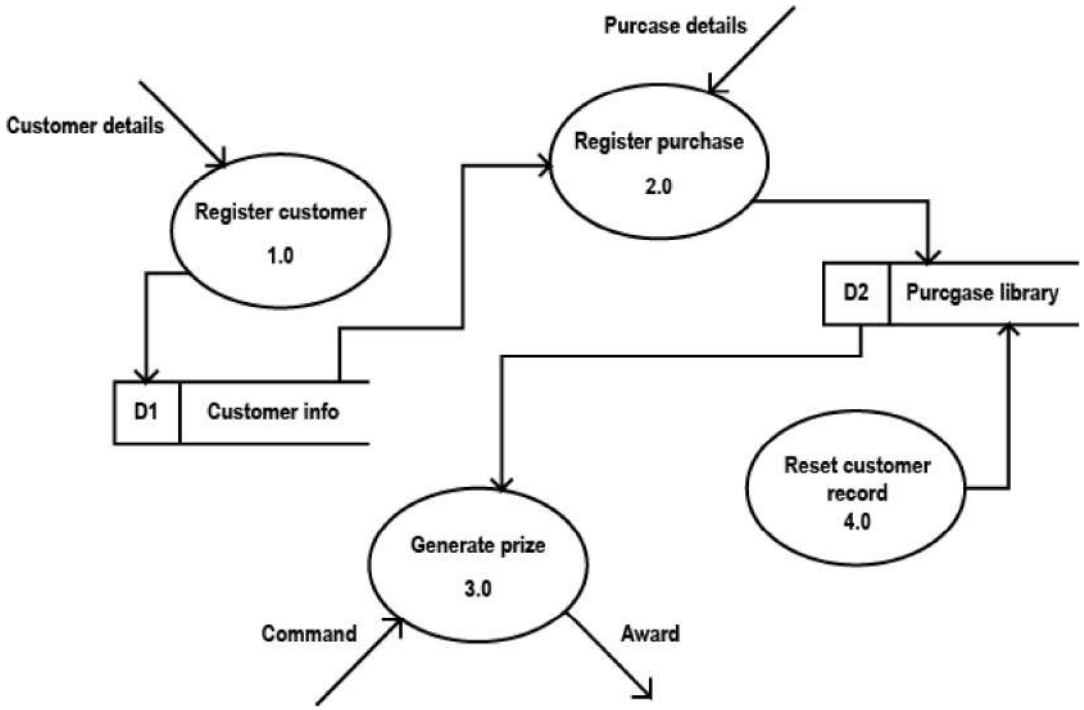
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

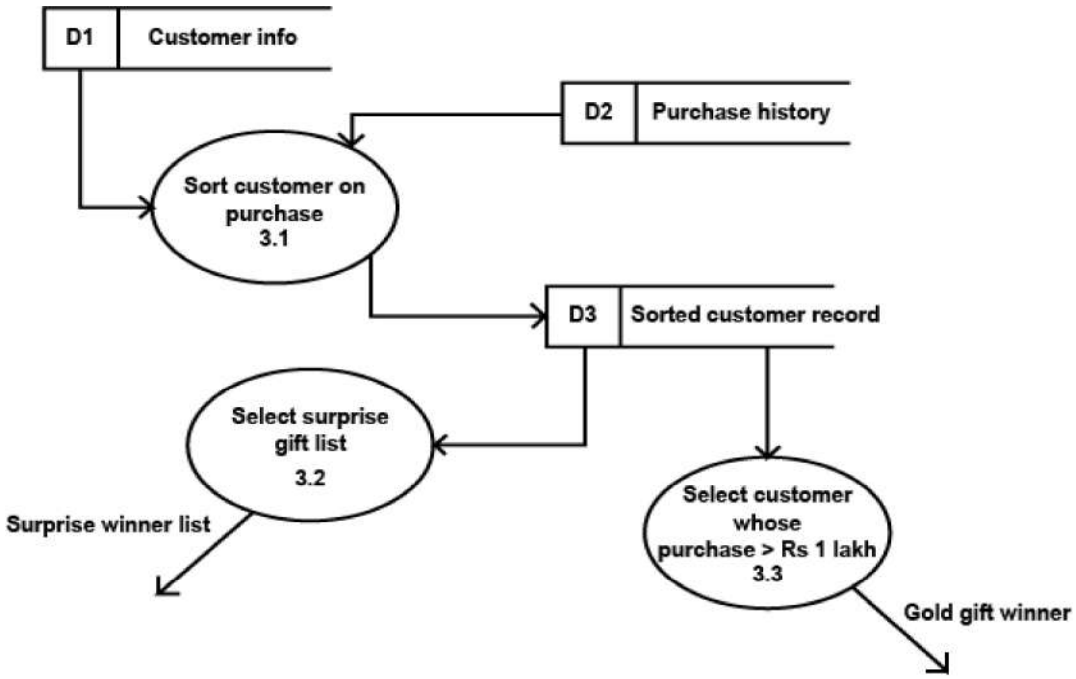
1. Level 0 DFD :



3. Level 1 DFD :



4. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

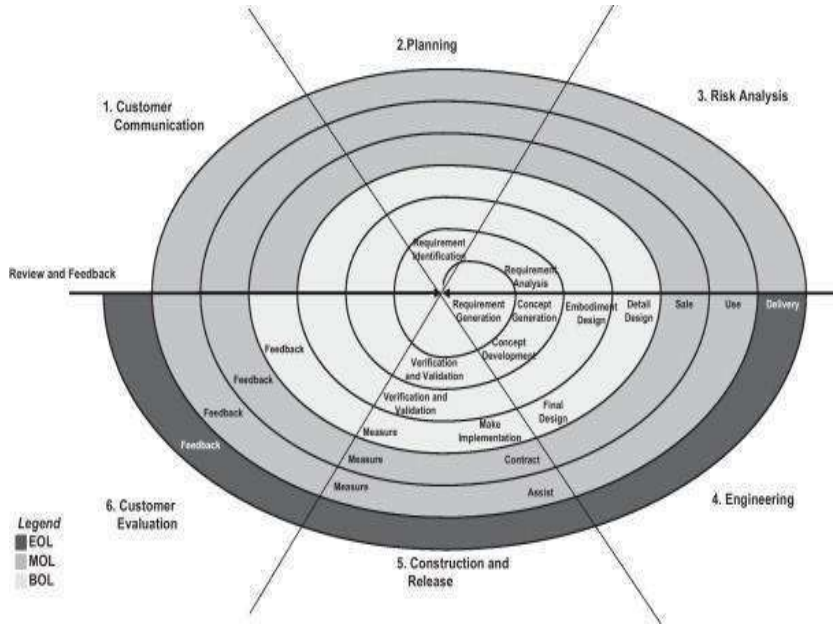
PROJECT 2

Route Management System

It identifies internal datastores of payment, bus route,

customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule.It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. ***Goal of the Project:***

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. ***Functional Requirements:***

As per the requirements of the software from client end. By analysis we realise that we need five modules: A.

Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Customer Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. Schedule Module:

This module maintains timings of the bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

---

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

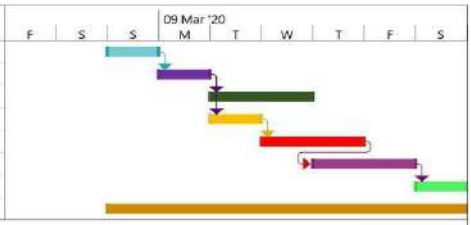
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 08-03-20	Sun 08-03-20	
2	Customer Communication	1 day	Mon 09-03-20	Mon 09-03-20	1
3	Planning	2 days	Tue 10-03-20	Wed 11-03-20	2
4	Risk Analysis	1 day	Tue 10-03-20	Tue 10-03-20	2
5	Engineering	2 days	Wed 11-03-20	Thu 12-03-20	4
6	Construction and Release	2 days	Thu 12-03-20	Fri 13-03-20	5
7	Customer Evaluation	1 day	Sat 14-03-20	Sat 14-03-20	6
8	Write User Manual	7 days	Sun 08-03-20	Sat 14-03-20	



Project: Project8
Date: Sat 14-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

b. Project Resources

a. **ManPower Resources:**
   i. **Project Manager**
   ii. **System Analyst**
   iii. **Designer** iv. **Developer** v. **Tester**
vi. **Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above**
   ii. Ram: 4Gb or above
   iii. **Java Development Kit -**
   13 iv. **Java Runtime**
   Environment v. **Sublime**
   Text 3 (Text Editor)
   vi. **Ms - Project**
   vii. **Creately** viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. **Likelihood of a risk coming through**
B. **Consequence of the problem associated with that risk**

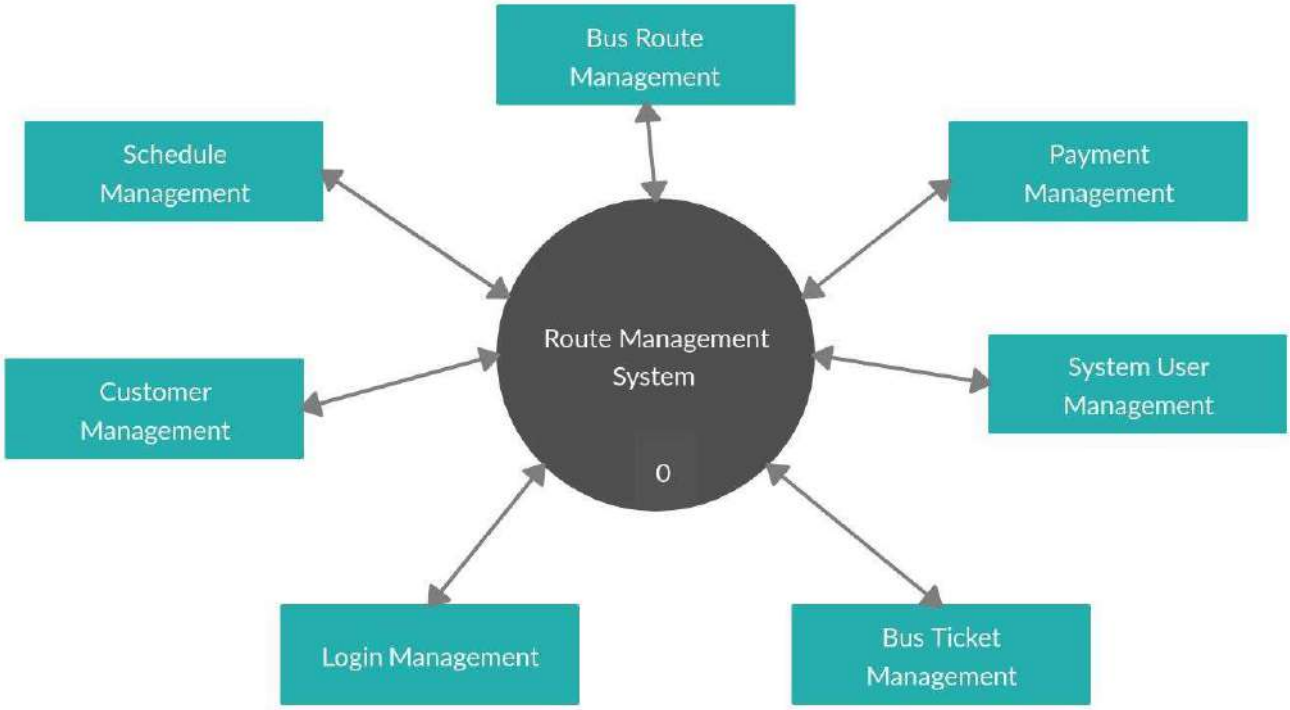
Priority of each risk is multiplication of above two

Risk Leverage =

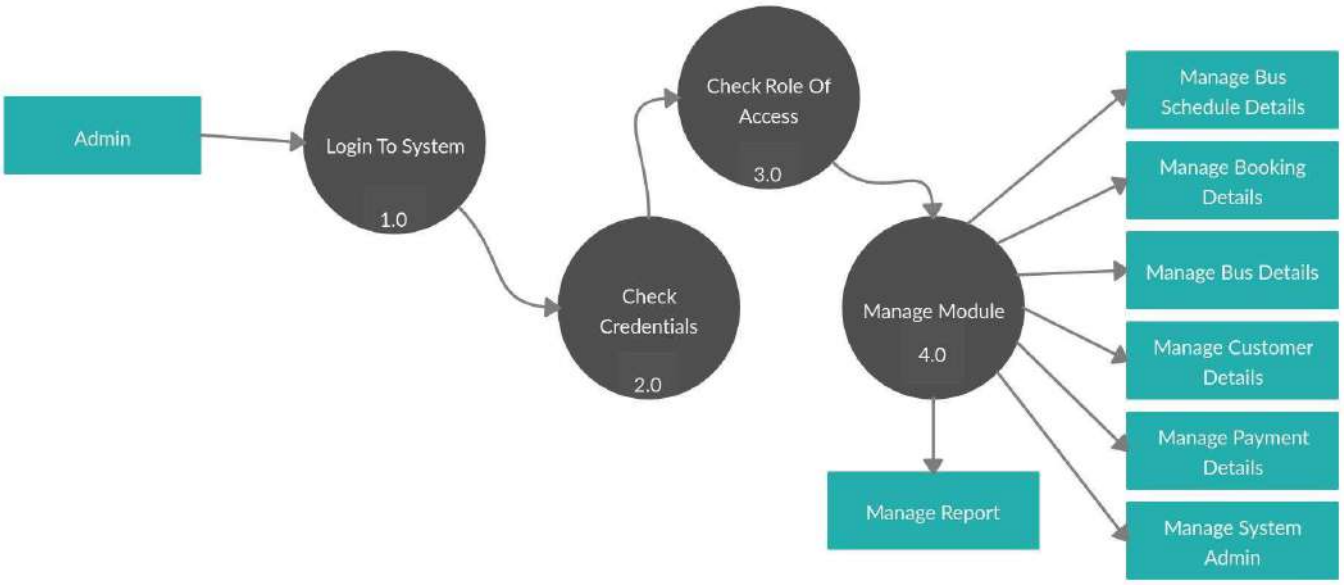
$$\frac{\textit{Risk exposure before Reduction} - \textit{Risk exposure after Reduct}}{\textit{Cost of Reduction}}$$

DESIGN ENGINEERING

1. **Architectural Design**



2. **Component Level Design**
   - **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing: Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance
4. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

      ○ Doctor module : This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.

      ○ Report module : This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.

      ○ Diagnostic module : This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.

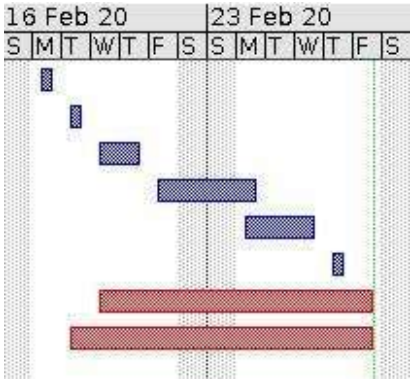
      ○ Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

      ○ Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

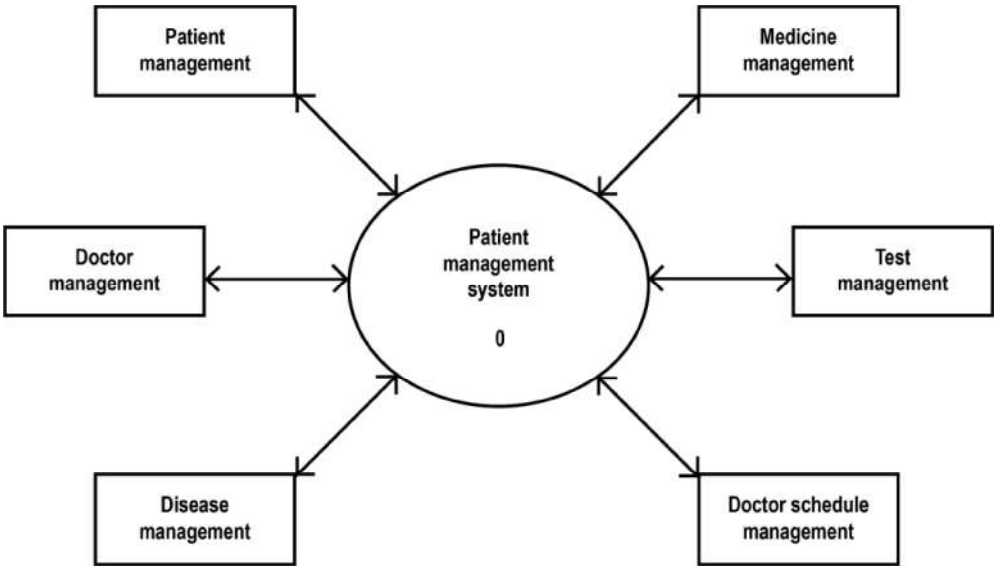
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

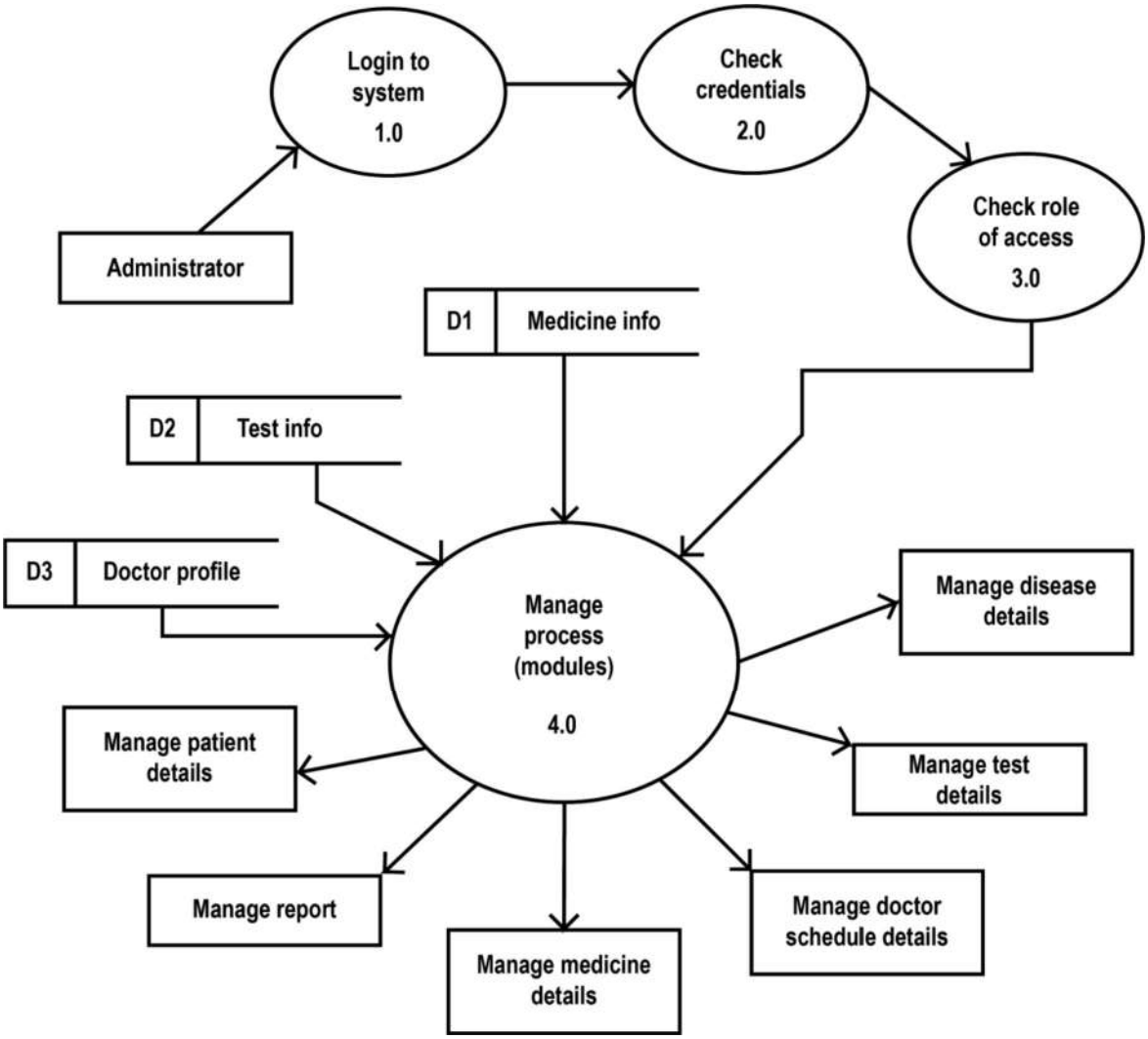
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.

- Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. Guest module : In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. Stay module : In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

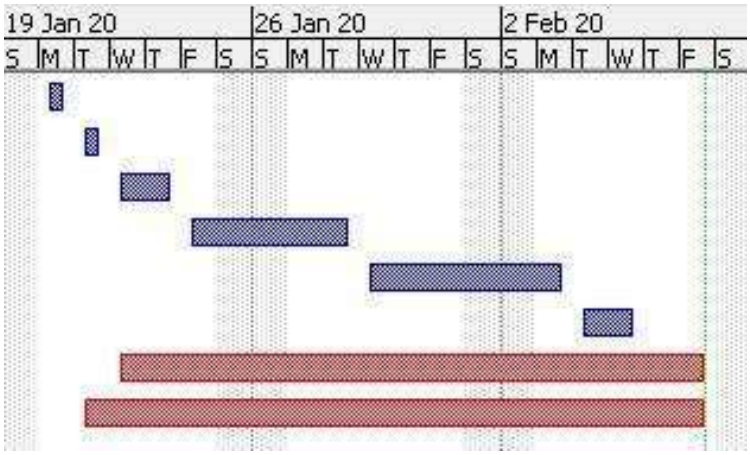
3. Catering module : In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.

4. Bill generation module : At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2		Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3		Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4		Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5		Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



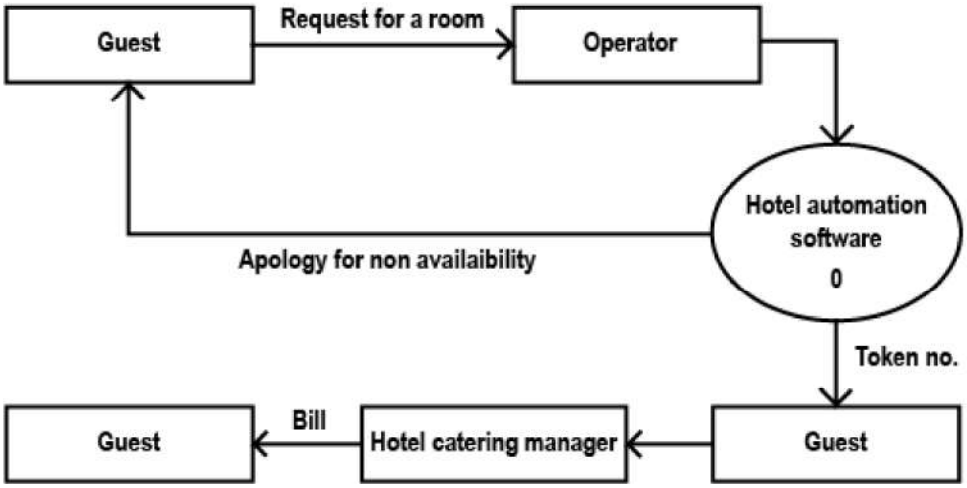
- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

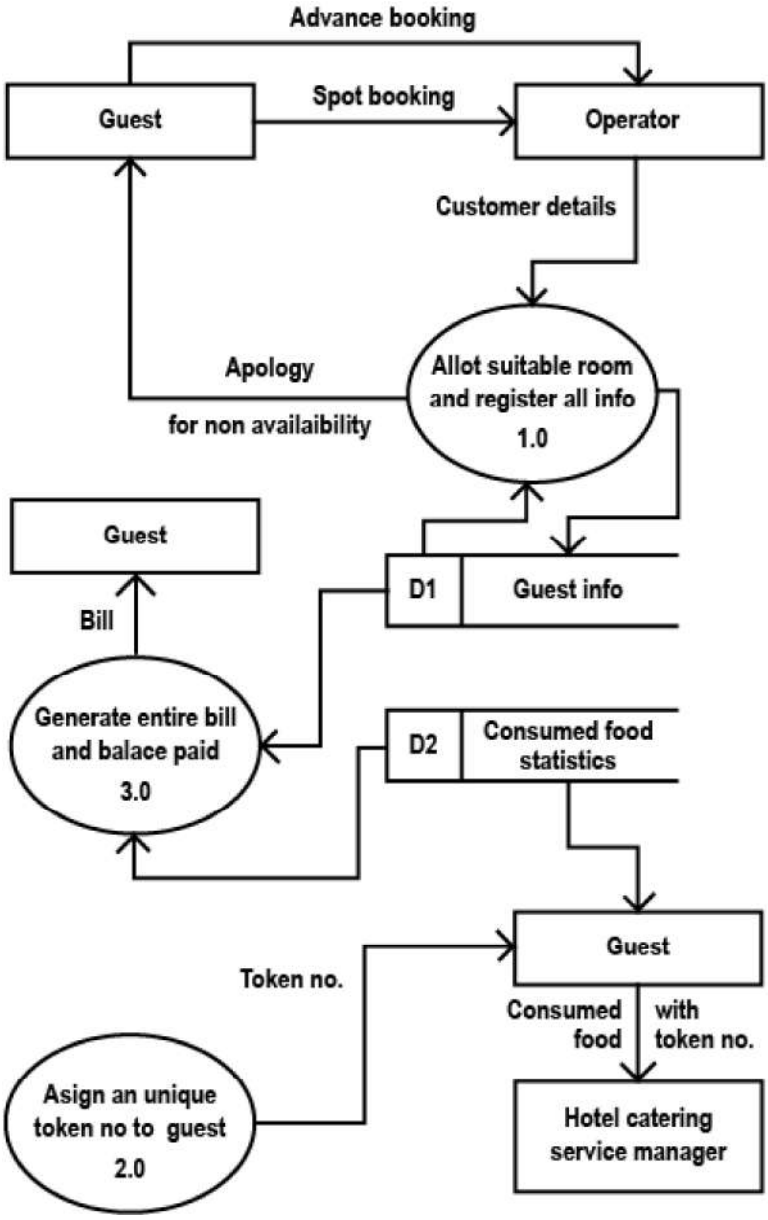
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.

    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Road maintenance is the performance of adaptive, preventive and corrective maintenance for a given

road in a locality. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

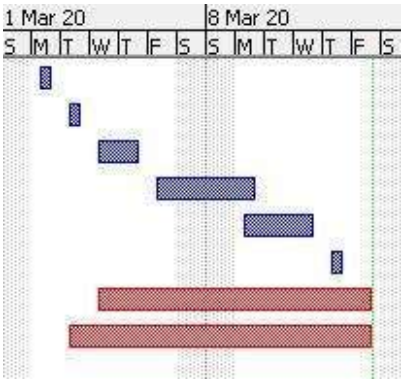
- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
    - Criminal module : This module accesses data from the database of criminal record details and manages it properly.
    - FIR module : This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
    - Chargesheet module : This module accesses data from the database of chargesheet record details and manages it properly.
    - Court module : The main function of this module is to manage the court profiles.

- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
    - Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



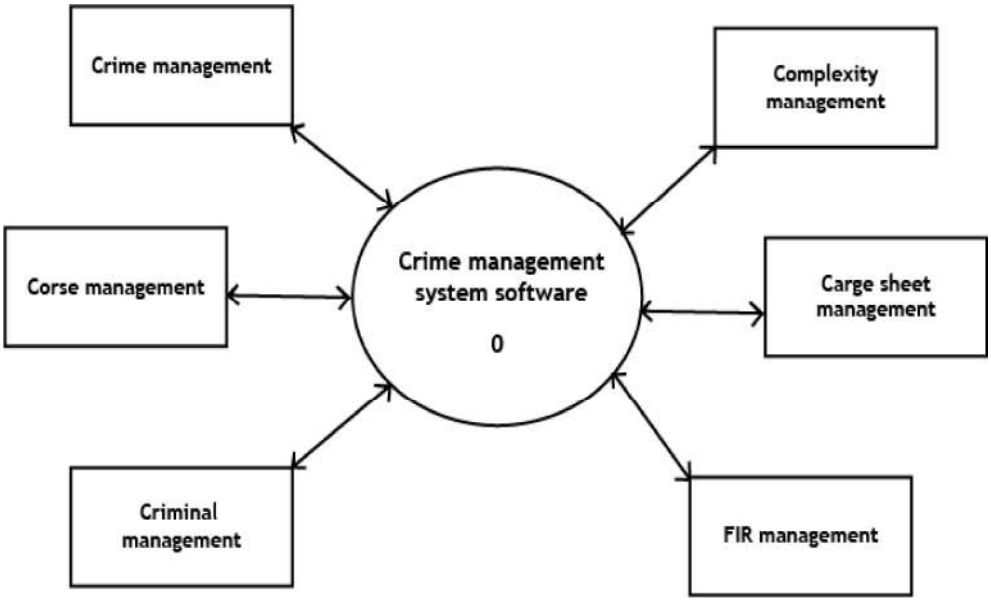
- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
    - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{}$$

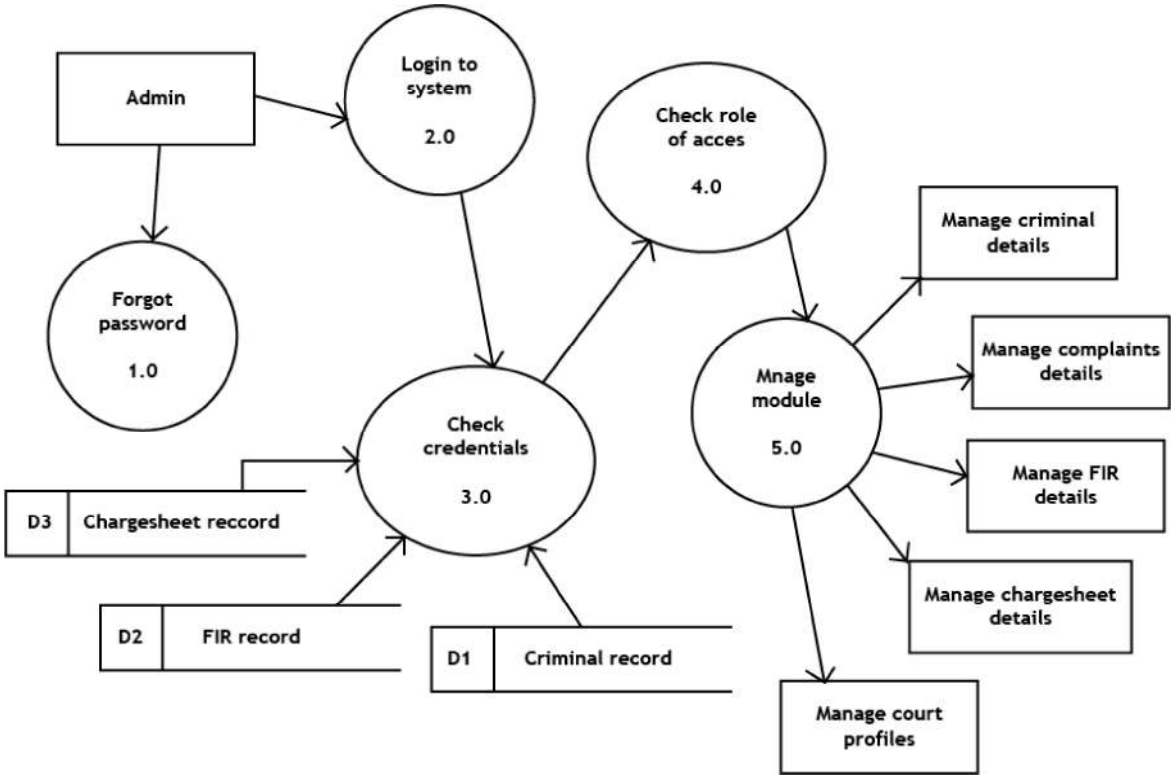
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
        - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
        - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT :

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

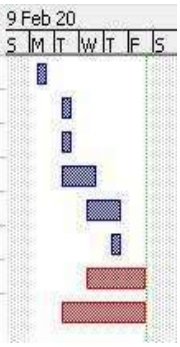
- Goal of the project : The aim and objectives for students' examination
  Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements : The three modules needed are.
  - Login and recovery module : This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
  - Credentials module : This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
  - Administrator module : This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.
- Non-Functional Requirements :
  - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

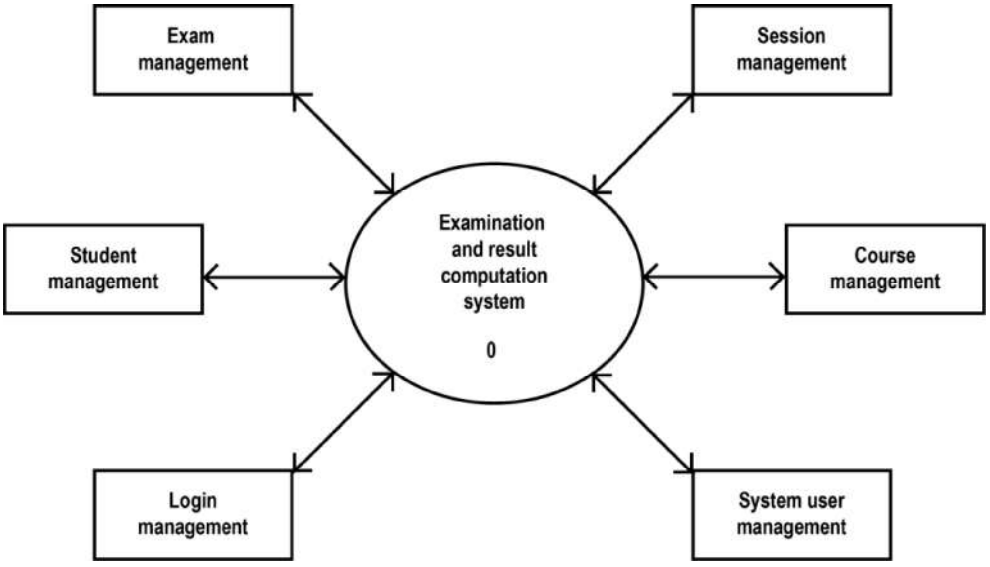


- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
    - Risk leverage =

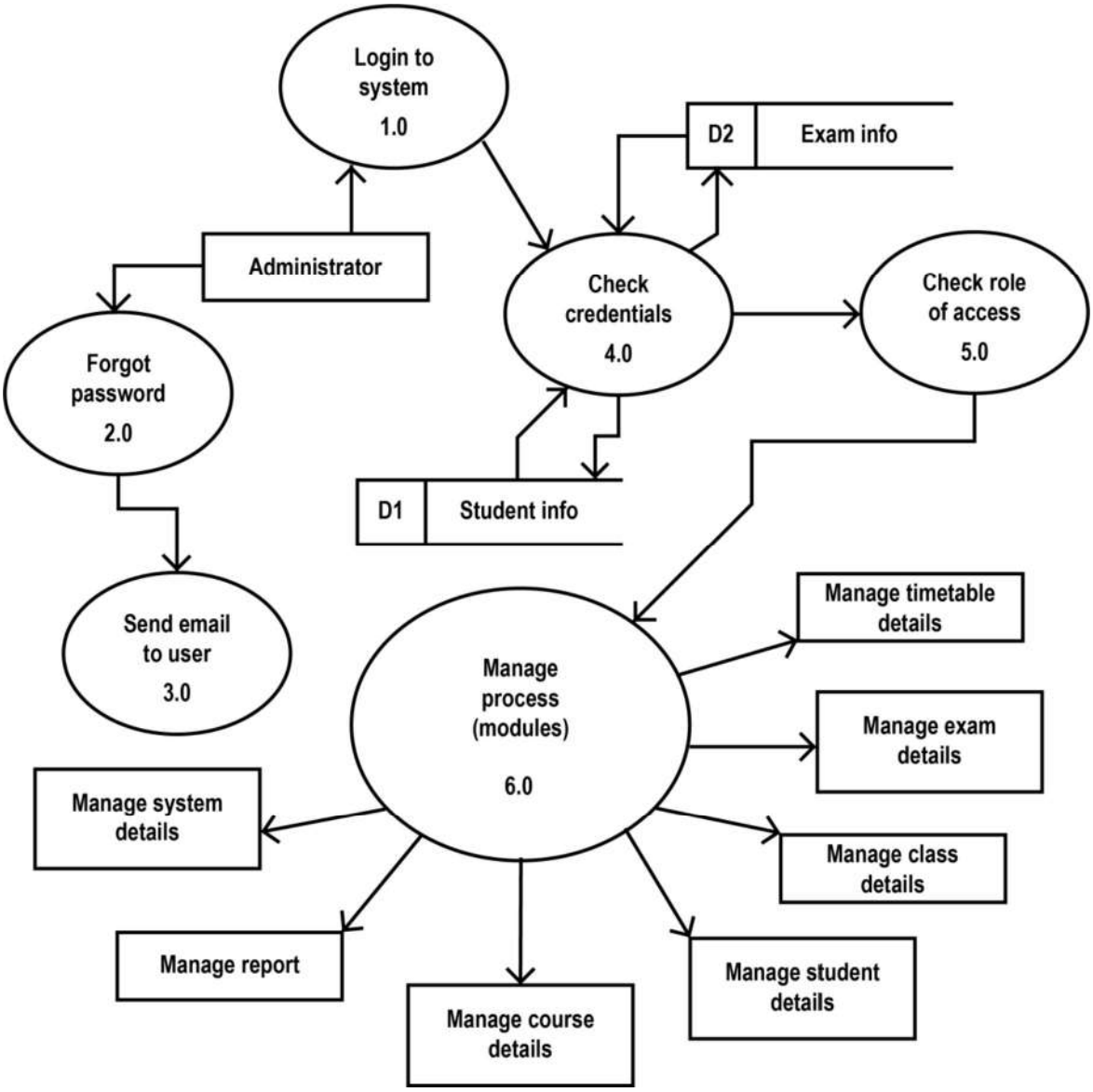
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :


Exam management
Session management
Examination and result computation system
0
Student management
Course management
Login management
System user management

2. Level 1 DFD :


Login to system
1.0
D2
Exam info
Administrator
Check credentials
4.0
Check role of access
5.0
Forgot password
2.0
D1
Student info
Send email to user
3.0
Manage process (modules)
6.0
Manage timetable details
Manage exam details
Manage system details
Manage class details
Manage report
Manage course details
Manage student details

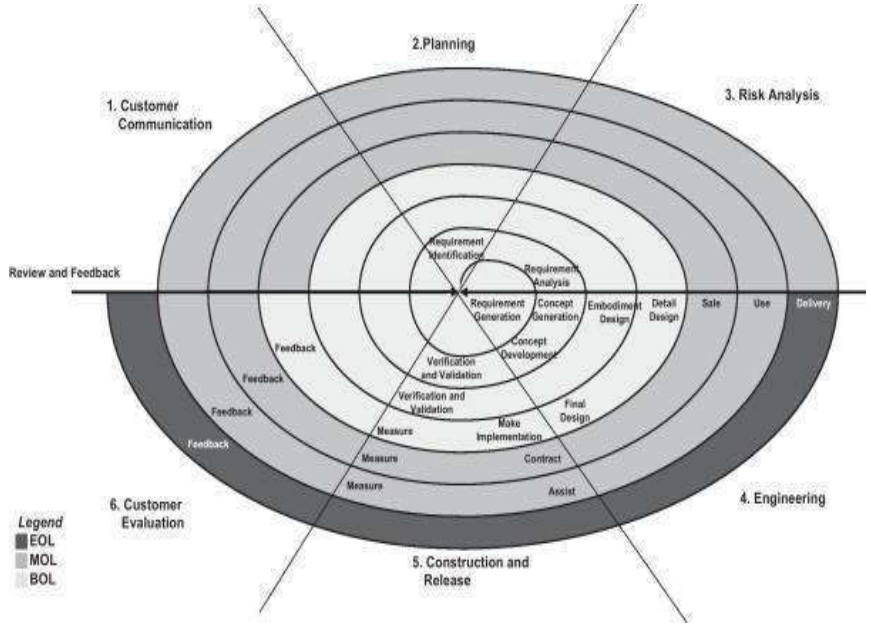
Project 7

Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications** ***1. Goal of the Project:***

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc which was a hefty work.

2. Functional Requirements:

As per the requirements of the software from client end.

By analysis we realise that we need three modules:

A. Admin Module:

This module asks username and password as input and if the

input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.
Input: Answer to security question
Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. Parking Module:

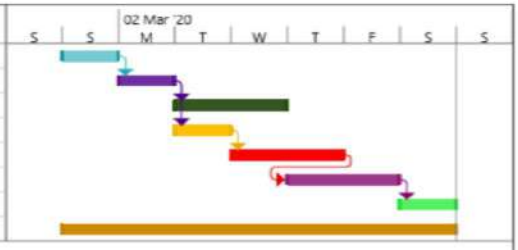
This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.
C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



Project: Project7
Date: Fri 13-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

a. Project Scheduling

b. Project Resources

a. ManPower Resources:

i. Project Manager

ii. System Analyst

iii. Designer iv. Developer v. Tester

vi. Document Writer

b. Hardware-Software Resources

i. Processor: Intel i3 4th gen or above

ii. Ram: 4Gb or above

iii. Java Development Kit - 13 iv. Java Runtime Environment v. Sublime Text 3 (Text Editor)

vi. **Ms - Project**

vii. **Creately** viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk

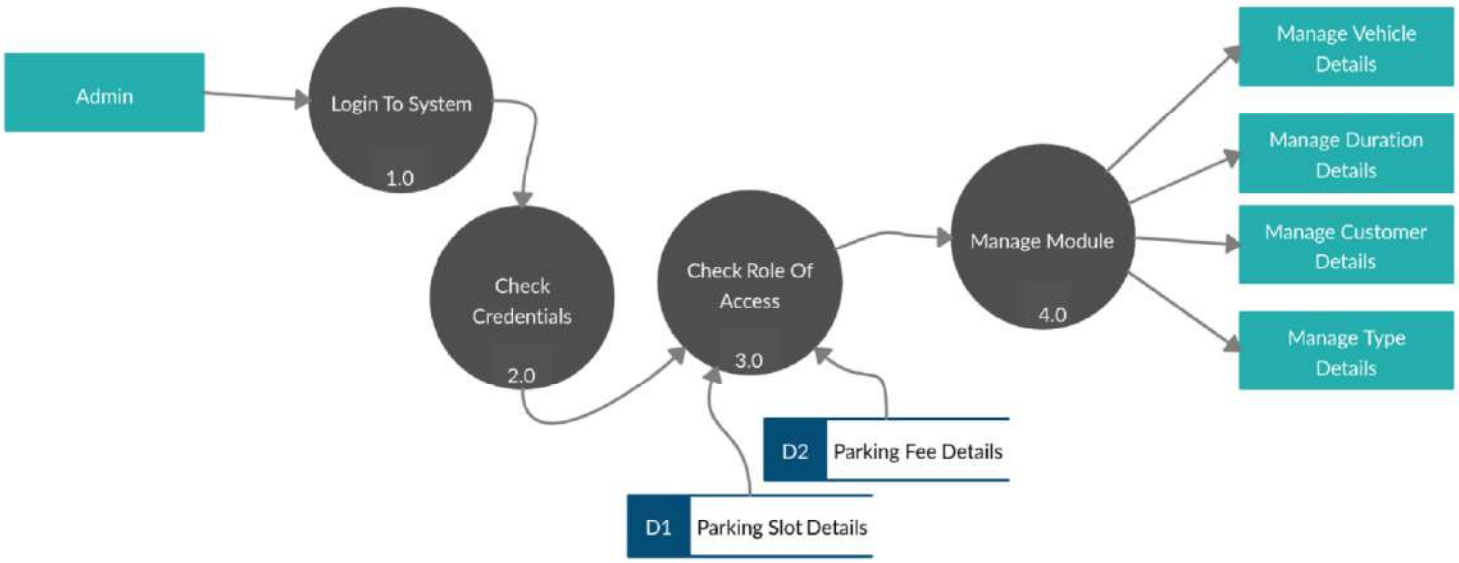
Priority of each risk is multiplication of above two

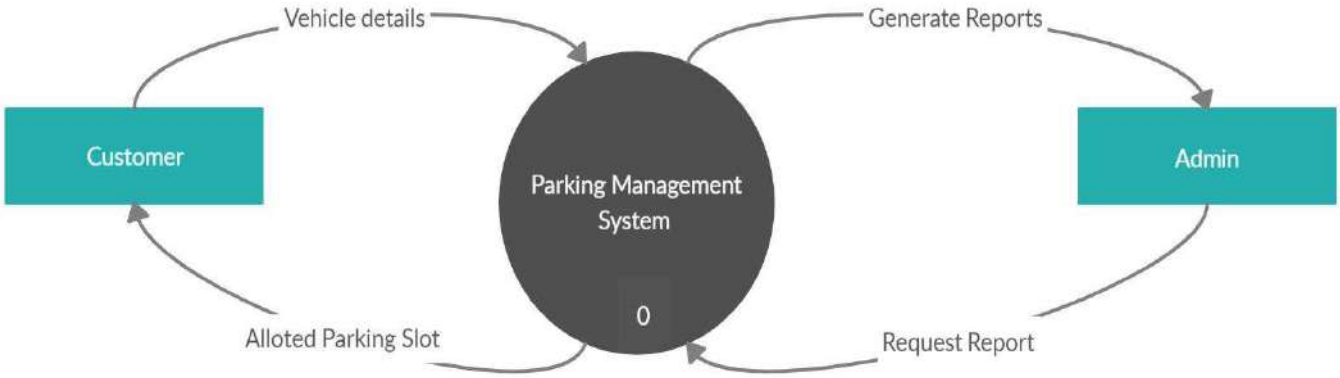
Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. Architectural Design





2. **Component Level Design**

   - **Level 1 DFD**

TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation **2. System Testing:**

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

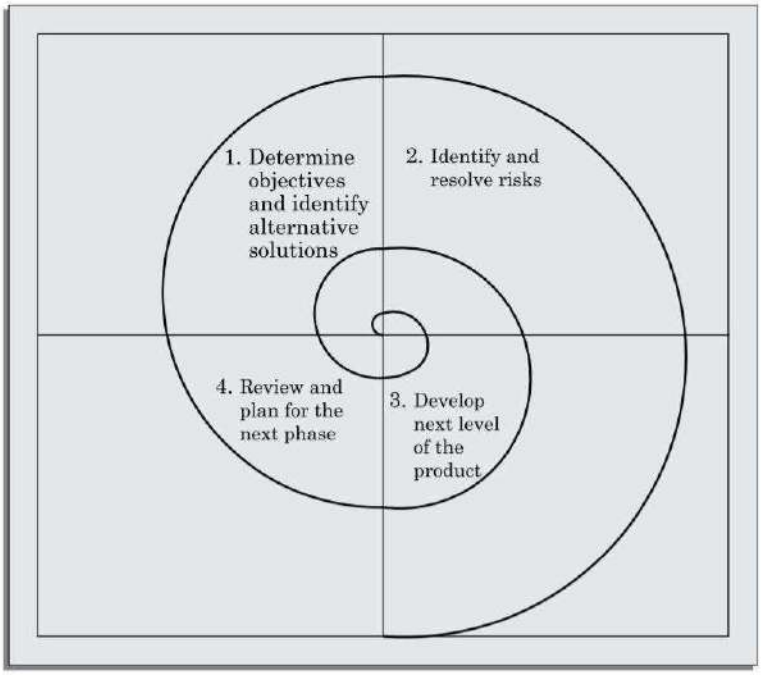


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Customer Module – In this module, a customer is registered and the profile is Maintained.

Input: customer details
Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained. A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

Report Module – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

Input: store details, product details, retail price details, bill details, billing counter details, customer details
Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

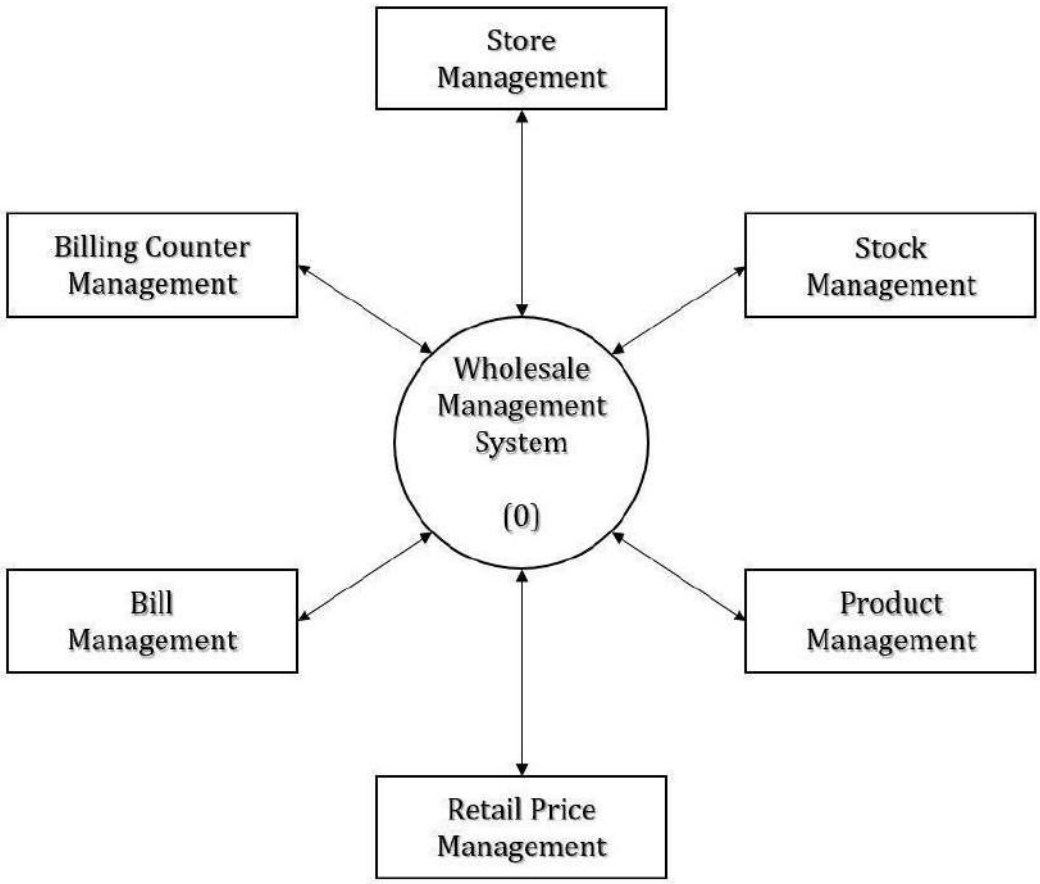


Fig – DFD level zero

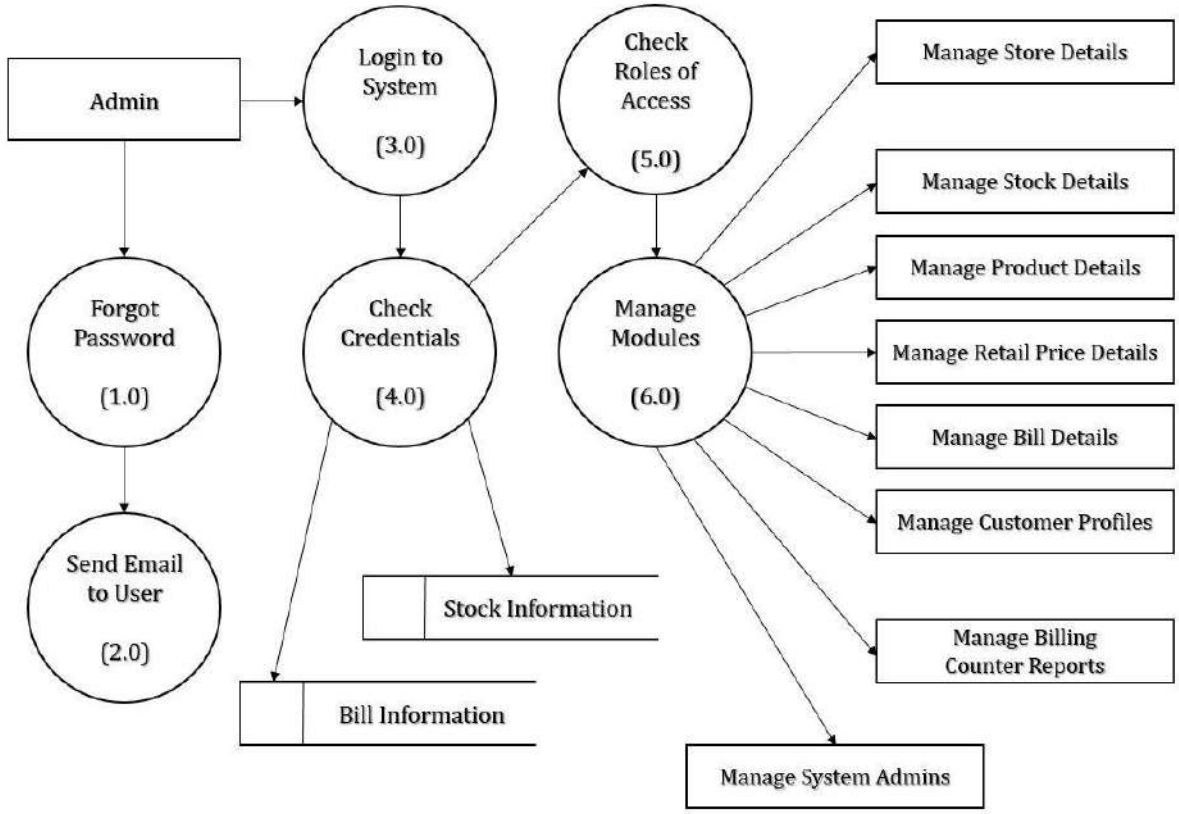
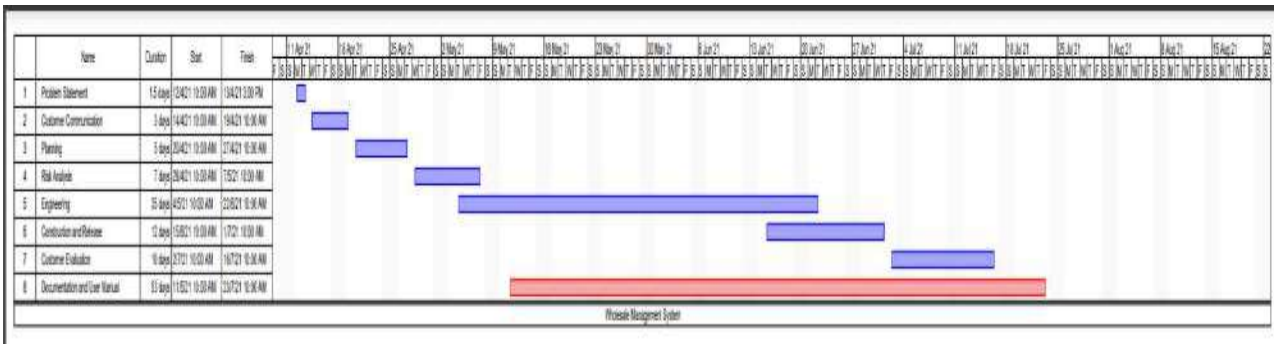


Fig – DFD level one

Project Management

Project Schedule –



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

The likelihood of a risk coming true (**r**)

The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

risk leverage= risk exposure before reduction-risk exposure after reductioncost of reduction

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

α (Alpha) Testing – It is the system testing performed by the development team in a controlled environment.

ii. **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii. **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i. **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii. **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii. **Adaptive maintenance –** Porting the software to work in a new environment.

Name: Sudipta Ghosh

Department: Computer Science

Semester: V

Subject: Cloud Computing (DSE III)

Roll: 714

Reg: A01-1142-117-012-2019

Google App Engine

Case study

Date Created:
10/12/2021

Supervisor's Name:
Dr. Biswajeet Biswas

Contents

What is Cloud Computing?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end users. Following are the working models of cloud computing:

- Deployment Model
- Service Model

Deployment Models

Public Cloud

Public cloud is an IT model where on-demand computing services and infrastructure are managed by a third-party provider and shared with multiple organizations using the public internet. Public cloud makes computing resources available to anyone for purchase. Multiple users typically share the use of a public cloud.

Private Cloud

Private cloud provides a high level of security and privacy to data through firewalls and internal hosting. It also ensures that operational and sensitive data are not accessible to third-party providers. HP Data Centers, Microsoft, Elastra-private cloud, and Ubuntu are the example of a private cloud.

Hybrid Cloud

Hybrid cloud refers to a mixed computing, storage, and services environment made up of on-premises infrastructure, private cloud services, and a public cloud—such as Amazon Web Services (AWS) or Microsoft Azure—with orchestration among the various platforms.

Service Models

Infrastructure as a Service (IaaS)

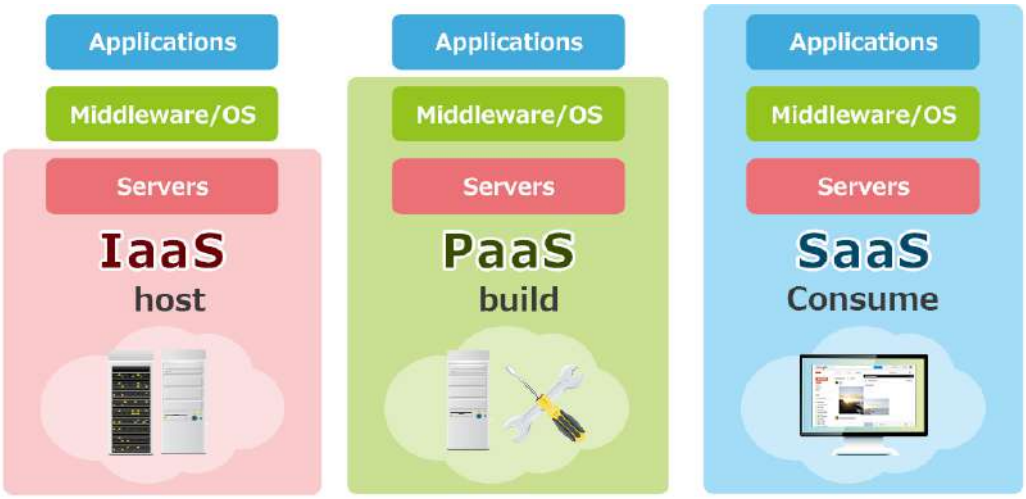
Infrastructure as a service are online services that provide high-level APIs used to dereference various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc.

Platform as a Service (PaaS)

Platform as a service (PaaS) is a cloud computing model where a third-party provider delivers hardware and software tools to users over the internet. As a result, PaaS frees developers from having to install in-house hardware and software to develop or run a new application. For e.g. Google App Engine

Software as a Service (SaaS)

Software as a service (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.



What is Google App Engine?

Google App Engine (often referred to as GAE or simply App Engine) is a cloud computing platform as a service for developing and hosting web applications in Google-managed data centers. Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand.

Google App Engine primarily supports Go, PHP, Java, Python, Node.js, .NET, and Ruby applications, although it can also support other languages via "custom runtimes".[4] The service is free up to a certain level of consumed resources and only in standard environment but not in flexible environment. Fees are charged for additional storage, bandwidth, or instance hours required by the application. It was first released as a preview version in April 2008 and came out of preview in September 2011.

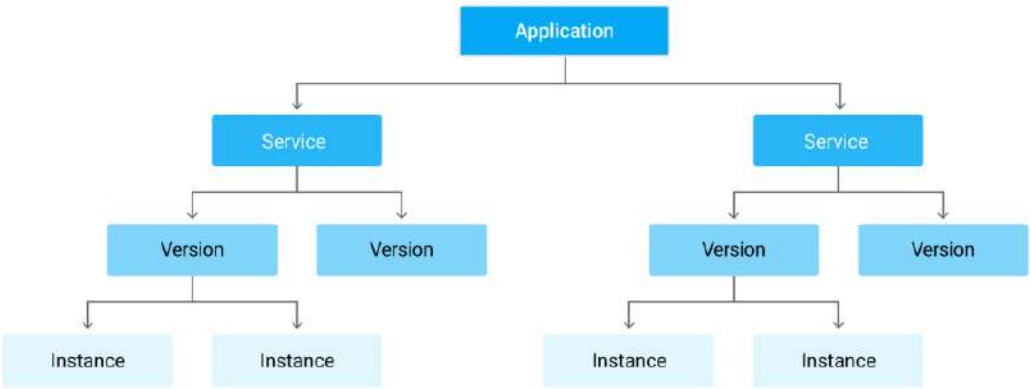
How GAE works?

An App Engine app is made up of a single application resource that consists of one or more services. Each service can be configured to use different runtimes and to operate with different performance settings. Within each service, you deploy versions of that service. Each version then runs within one or more instances, depending on how much traffic you configured it to handle.

Your App Engine app is created under your Google Cloud project when you create an application resource. The App Engine application is a top-level container that includes the service, version, and instance resources that make up your app. When you create your App Engine app, all your resources are created in the region that you choose, including your app code along with a collection of settings, credentials, and your app's metadata.

Each App Engine application includes at least one service, the default service, which can hold many versions, depending on your app's billing status. For more information, see Limits below.

The following diagram illustrates the hierarchy of an App Engine app running with multiple services. In this diagram, the app has two services that contain multiple versions, and two of those versions are actively running on multiple instances:



Services Provided by GAE

Google Cloud Platform Services

Google provides a considerable number of services with several unique features. That is the reason why Google Cloud Platform is continually expanding across the globe. Some of the significant services of GCP are:

- Compute Services
- Networking
- Storage Services
- Big Data
- Security and Identity Management
- Management Tools
- Cloud AI
- IoT (Internet of Things)

Compute Services

GCP offers a scalable range of computing services, such as:

- **Google App Engine**: It is a cloud computing platform that follows the concept of Platform-as-a-Service to deploy PHP, Java and other software. It is also used to develop and deploy web-based software in Google-managed data centers. The most significant advantage of Google App Engine is its automatic scaling capability. This means that the App Engine automatically allocates more resources for the application when there is an increase in requests.
- **Compute Engine**: It is a cloud computing platform that follows the concept of Infrastructure-as-a-Service to run Windows and Linux based virtual machines. It is an essential component of GCP. It is designed on the same infrastructure used by Google search engine, YouTube and other Google services.
- **Kubernetes Engines**: This computing service is responsible for offering a platform for automatic deployment, scaling, and other operations of application containers across clusters of hosts. The engine supports several container tools like a docker, etc.

Networking

GCP includes the following network services:

- **VPC**: VPC stands for Virtual Private Network. The primary function of VPC is to offer a private network with routing, IP allocation, and network firewall policies. This will help to create a secure environment for the application deployments.
- **Cloud Load Balancing**: As its name states, Cloud balancing are used to distribute workload across different computing resources to balance the entire system performance. This also results in cost-reduction. The process also helps in minimizing the availability and maximizing the capability of the resources.
- **Content Delivery Network**: CDN is a geographically distributed network of proxy servers and their data centers. The primary aim of using CDN is to provide maximum performance to the users. Additionally, it also helps deliver high availability of resources by equally distributing the related services to the end-users.

Storage Services

GCP has the following storage services:

- **Google Cloud Storage**: It is an online data storage web service that Google provides to its users to store and access data from anywhere. The service also includes a wide range of features like maximum performance, scalability, security and sharing.
- **Cloud SQL**: It is a web-service that enables users to create, manage, and use relational databases stored on Google Cloud servers. The service itself maintains and protects the databases, which helps users focus on their applications and other operations.
- **Cloud Bigtable**: It is known for its fast performance and highly manageable feature. It is a highly scalable NoSQL database service that allows collecting and retaining data from as low as 1 TB to hundreds of PB.

Big Data

GCP provides a variety of services related to big data; they are:

- **BigQuery**: It is a fully managed data analysis service by Google. The primary aim of Google BigQuery service is to helps businesses to analyze Big Data. It offers a highly scalable data management option. This means BigQuery allows users to perform ad-hoc queries and share data insights across the web.
- **Google Cloud Datastore**: Google Cloud Datastore is a kind of datastore service that is fully managed, schema-less, and non-relational. This service enables businesses to perform automatic transactions and a rich set of queries. The main advantage of Google Cloud Datastore is the capability of automatic scaling. This means that the service can itself scale up and down, depending on the requirement of resources.
- **Google Cloud Dataproc**: It is a very fast and easy to use big data service offered by Google. It mainly helps in managing Hadoop and Spark services for distributed data processing. The service allows users to create Hadoop or Spark clusters sized according to the overall workload and can be accessed whenever users want them.

Security and Identity Management

GCP includes the following services related to Security and Identity management:

- **Cloud Data Loss Prevention API**: It is mainly designed to manage sensitive data. It helps users manage sensitive data elements like credit card details, debit card details, passport numbers, etc. It offers fast and scalable classification for sensitive data.
- **Cloud IAM**: It stands for Cloud Identity and Access Management. It is a framework that contains rules and policies and validates the authentication of the users for accessing the technology resources. That is why it is also known as Identity Management (IdM).

Management Tools

GCP includes the following services related to management tools:

- **Google Stack driver**: Google Stack driver service is primarily responsible for displaying the overall performance and diagnostics information. This may include insights of data monitoring, tracing, logging, error reporting, etc. The service also prompts an alert notification to the public cloud users.
- **Google Cloud Console App**: It is a native mobile application powered by Google. The primary aim of this service is to enable users to manage the core features of Google Cloud services directly from their mobile devices anytime, anywhere. The primary functions of this service are alerting, monitoring, and performing critical actions on resources.

Cloud AI

When it comes to Cloud AI, GCP offers these services:

- **Cloud Machine Learning Engine**: It is another fully managed service that allows users to create Machine Learning models. The service is mainly used for those ML models, which are based on mainstream frameworks.
- **Cloud AutoML**: It is the type of service that is based on Machine Learning. It helps users to enter their data sets and gain access to quality trained pre-designed ML models. The service works by following Google's transfer learning and Neural Architecture Search method.

IoT (Internet of Things)

GCP contains the following IOT services:

Cloud IoT Core: It is one of the fully managed core services. It allows users to connect, control, and ingest data from various devices that are securely connected to the Internet. This allows other Google cloud services to analyse process, collect and visualize IoT data in real-time.

Cloud IoT Edge: The Edge computing service brings memory and other computing-power resources near to the location where it is required.

Advantages of Google App Engine

1. The main advantage of GAE is it doesn't scale. However, even after you empower billing, the whole system is augmented to support only 500 requests per second. If you want more, you can reach Google's disposal to increase your thresholds, so you can have millions of users, but more than 500 requests per second. If you want to do more than 500 requests and your app is legit, they will remove those caps or make them higher.

2. GAE feature set is good enough to build a decent website and you don't need to do the maintenance work.

3. It doesn't require any server administration. It has free usage quotas and provides scalability. GAE has better access to Google user accounts and deployment process is very easy.

4. GAE has the highest admin load. Once you are set up, deploying and re-deploying is quick and they will auto-everything. For example, you don't have to worry about the number of servers your app is using, how to share the data and how to load-balance.

5. You can get any feature from the store with GAE. But with Azure, you get the feeling of SQL Azure database but at the same time it would be too expensive. Azure storage is likely to have more gotchas. No relational integrity, no order-by, you will scam with the in-memory setting more. GAE's store has far rarer margins and more features than Azure tables.

Disadvantages of Google App Engine

1. GAE is yet not stable enough. Even Google says that GAE is the same infrastructure of Google self-internal project. And the budget would increase much when your website becomes bulky.

2. Without native file system read/write access, it is hard to process some data transform with existing library, and it doesn't support some native file system base library as well.

3. It does not provide full text search API. Also, the SDK/Java is unfavourable with Maven, it is unsatisfactory to accomplish lots of external libraries. The SDK/Java depth rest on IDE and the default project directory structure is some difference with normal web app.

4. It is not easy to process unit test. It cannot fix the root cause and does not support add SSL to web site. The GAE may be the development for future web application, but it is not equipped for building a modern web site now.

5. It suffers from the inability to tweak server software. The File system and many standard library modules are inaccessible. Only Python and a few runs of Java Virtual machine.

EUCALYPTUS

Case Study

Name: Imon Kalyan Saha
Dept. : Computer Science
Semester : V
Subject : Cloud Computing (DSE3)
Roll : 716
Reg.No. : A01-1112-117-013-2019

Supervisor's Name: Dr. Chayan Halder

CONTENTS

What is Cloud Computing ?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end
users. Following are the working models of cloud Computing:

➢ Deployment Models
➢ Service Models

What are the types of Deployment Model ?

Public Cloud

Public clouds are owned and operated by third-party cloud service providers, which deliver their computing resources, like servers and storage, over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software, and other supporting infrastructure is owned and managed by the cloud provider. You access these services and manage your account using a web browser.

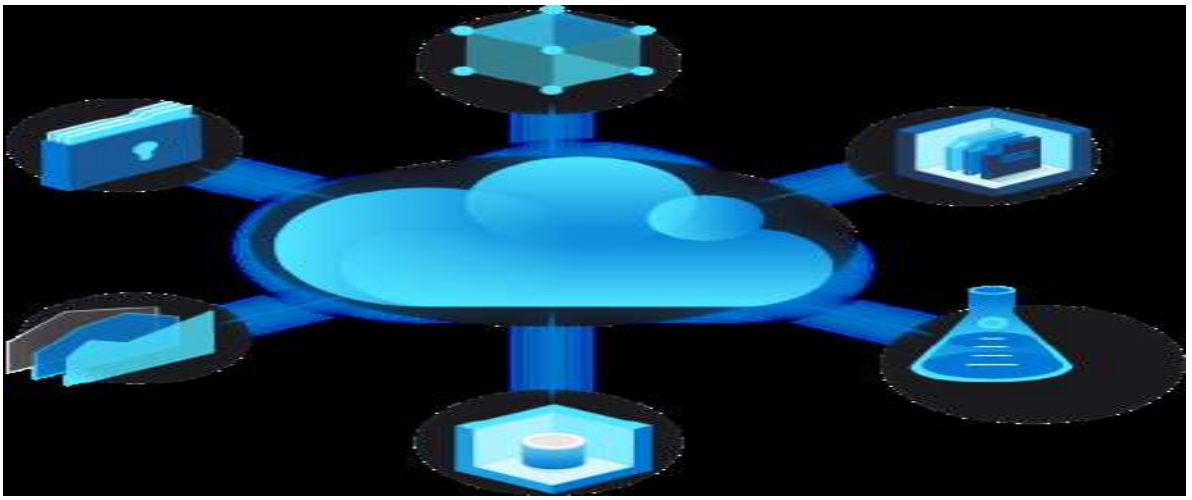
Private Cloud

Private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company's on-site datacenter. Some companies also pay third-party service providers to

host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

Hybrid Cloud

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options, and helps optimize your existing infrastructure, security, and compliance.



What are the types of Service Model ?

Infrastructure as a service (IaaS)

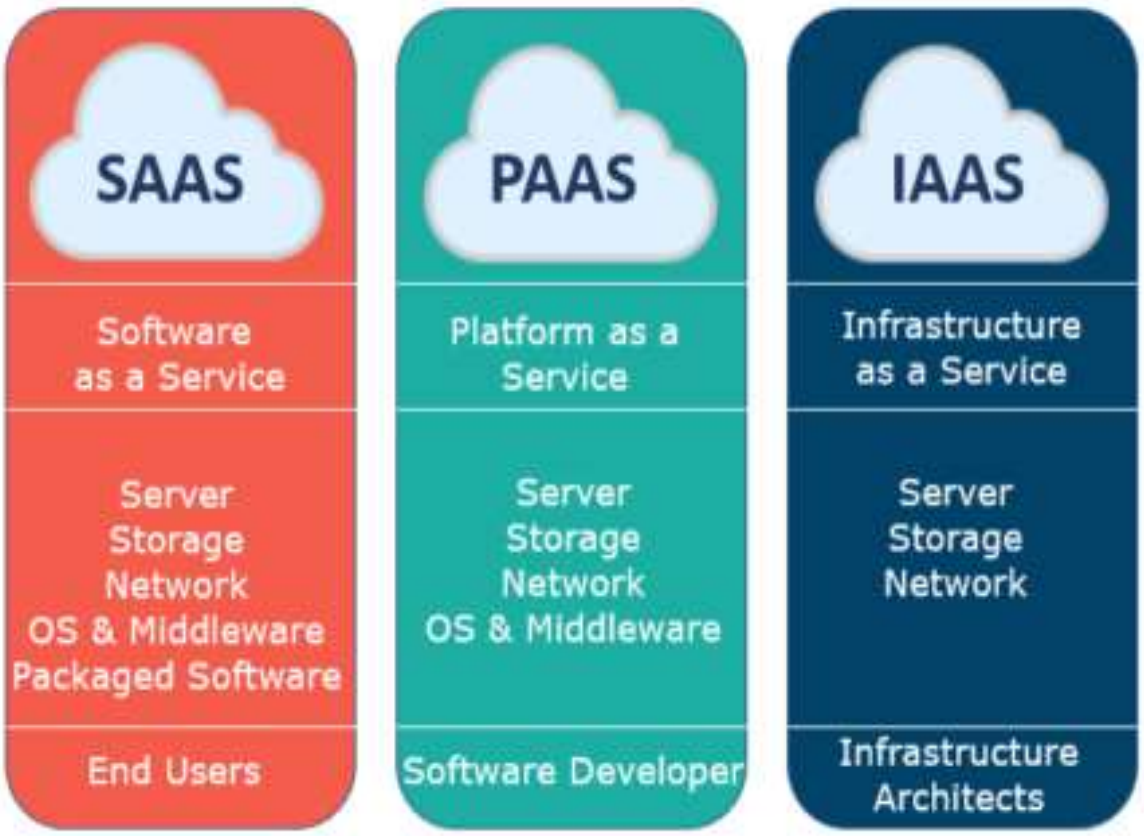
The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps.

Software as a service (SaaS)

Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching.



What is Eucalyptus?

Eucalyptus in cloud computing is an open-source software platform for carrying out IaaS or Infrastructure-as-a-Service in a hybrid cloud computing or private cloud computing environment.

Eucalyptus in cloud computing pools together existing virtualised framework to make cloud resources for storage as a service, network as a service and infrastructure as a service. Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems is short known as Eucalyptus in cloud computing.

Eucalyptus in cloud computing frameworks declared a conventional concurrence with AWS or Amazon Web Services in March 2012, permitting overseers to move cases between an Amazon Elastic Compute Cloud and the Eucalyptus private cloud to make a hybrid cloud. The organisation

additionally permits Eucalyptus to work with Amazon's product groups to create interesting Amazon Web Services viable highlights.

It tends to be effortlessly sent in existing IT frameworks to appreciate the advantages of both eucalyptus private cloud and eucalyptus public cloud models.

History of Eucalyptus

Improvement on Eucalyptus started as an examination project at US-based Rice University in the year 2003. In the year 2009, an organisation named Eucalyptus Systems was framed to market Eucalyptus software. Afterwards, in the year 2012, the firm went into a concurrence with Amazon Web Services for keeping up similarity and Application Programming Interface support. In the year 2014, it was procured by Hewlett-Packard or HP, which unexpectedly has its own cloud contributions under the HPE eucalyptus. The Helion portfolio has an assortment of cloud-related items, which incorporates HP's own kind of OpenStack called HP Helion OpenStack. Presently, Eucalyptus is a piece of the HPE portfolio and is known as HPE Helion Eucalyptus.

Eucalyptus architecture

Eucalyptus CLIs can oversee both Amazon Web Services and their own private occasions. Clients can undoubtedly relocate cases from Eucalyptus to Amazon Elastic Cloud. Network, storage, and compute are overseen by the virtualisation layer. Occurrences are isolated by hardware virtualisation. The following wording is utilised by Eucalyptus architecture in cloud computing.

1. Images: Any software application is work in , configuration, module software or framework software packaged and conveyed in the Eucalyptus cloud is known as a Eucalyptus Machine Image.

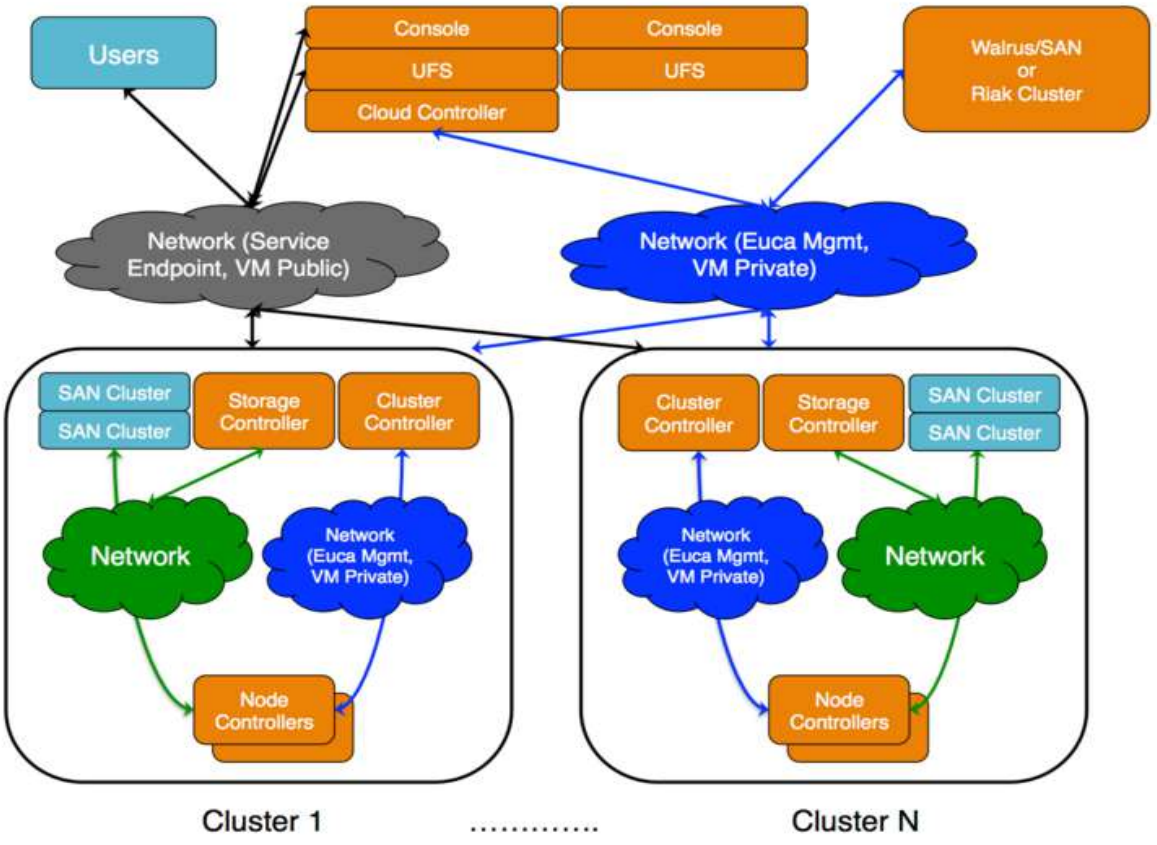
2. Instances: When we run the picture and utilise it, it turns into an instance.

3. Networking: The Eucalyptus network is to partitioned into three modes: Static mode, System mode, and Managed mode.

4. Access control: It is utilised to give limitation to clients.

5. Eucalyptus elastic block storage: It gives block-level storage volumes to connect to an instance.

6. Auto-scaling and load adjusting: It is utilised to make or obliterate cases or administrations dependent on necessities.



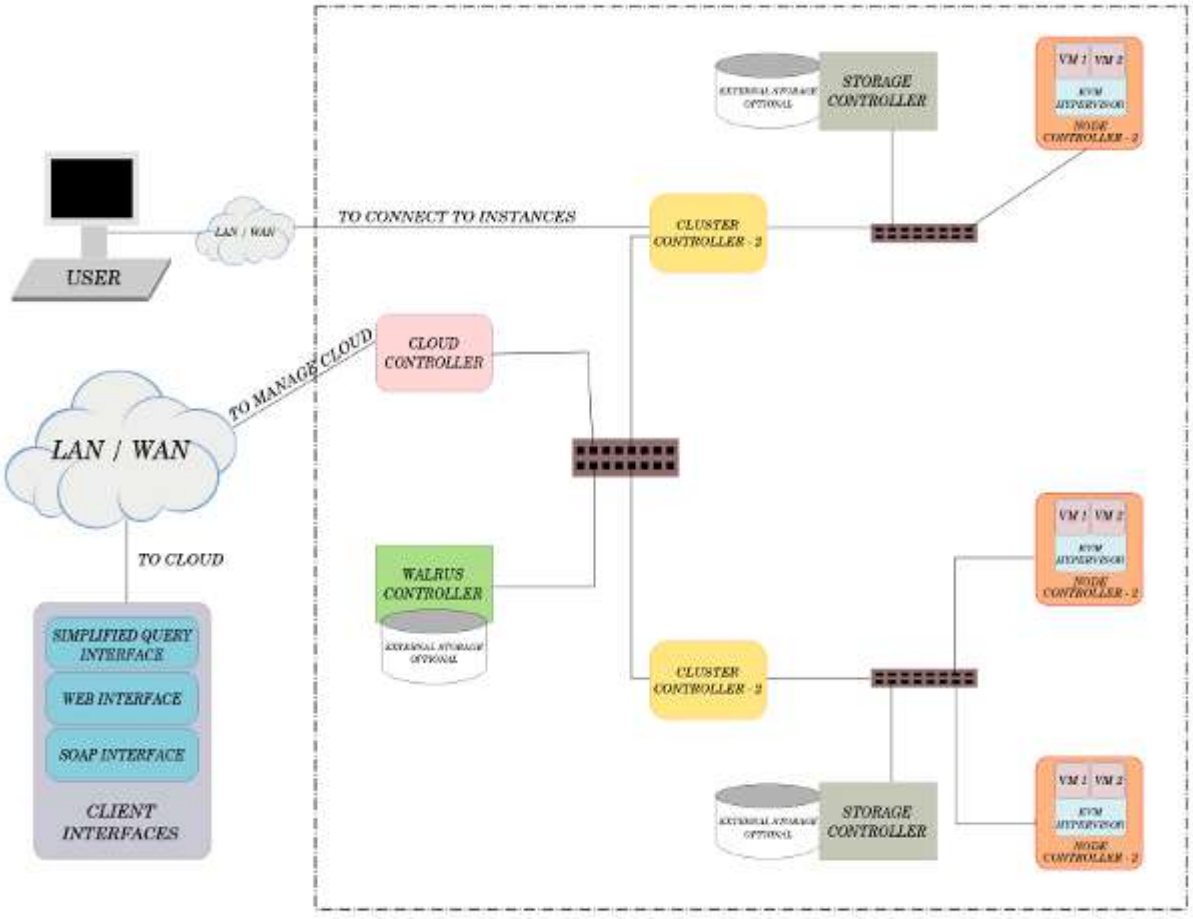
The advantages of the Eucalyptus cloud

The benefits of Eucalyptus in cloud computing are:

1. Eucalyptus can be utilised to benefit both the eucalyptus private cloud and the eucalyptus public cloud.
2. Clients can run Amazon or Eucalyptus machine pictures as examples on both clouds.
3. It isn’t extremely mainstream on the lookout yet is a solid contender to CloudStack and OpenStack.
4. It has 100% Application Programming Interface similarity with all the Amazon Web Services.
5. Eucalyptus can be utilised with DevOps apparatuses like Chef and Puppet.

What is the use of eucalyptus in cloud computing?

It is utilised to assemble hybrid, public and private clouds . It can likewise deliver your own data centre into a private cloud and permit you to stretch out the usefulness to numerous different organisations.



EUCALYPTUS BASED CLOUD

Conclusion

Eucalyptus in cloud computing is open-source programming that carries out an AWS viable cloud, which is financially savvy, secure and flexible. It tends to be effectively sent in existing IT frameworks to appreciate both private and public cloud models' advantages.

References

↞ https://docs.eucalyptus.cloud/eucalyptus/4.4.4/install-guide/euca_architecture.html

↞ https://www.jigsawacademy.com/blogs/cloud-computing/eucalyptus-in-cloud-computing/amp/

↞ https://www.wikipedia.org

Project on

RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Soumyadip Maity
College Roll: 717
Exam Roll: 2021141298
Registration number: A01 – 1112 – 117 – 014 – 2019
Paper Code: CMSADSE3
Semester: V
Supervisor: Dr. Chayan Halder

Project 1

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement – A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

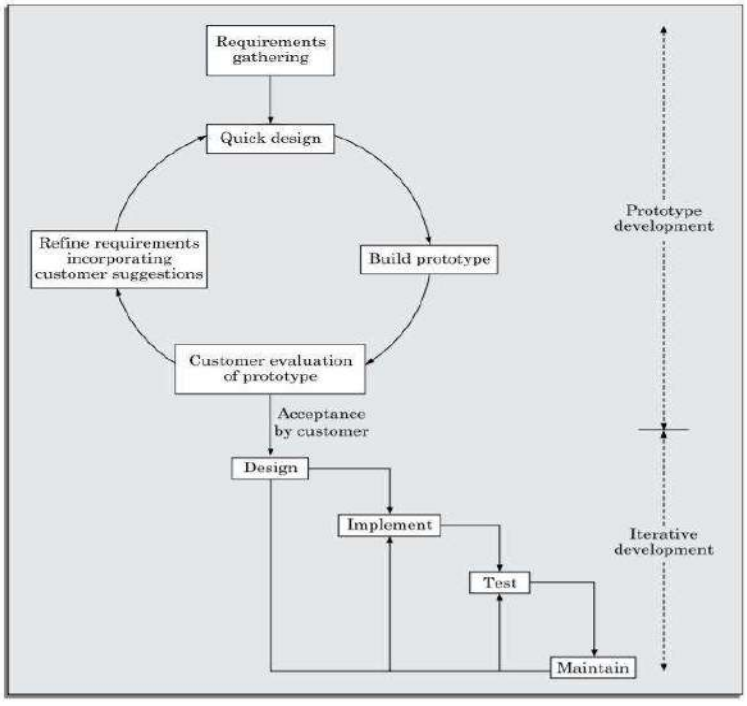


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: unique CN

2. **Purchase Module –** In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input: purchase made by the customer, CN
   Output: updated database

3. **Promotional Module –** In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output: prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

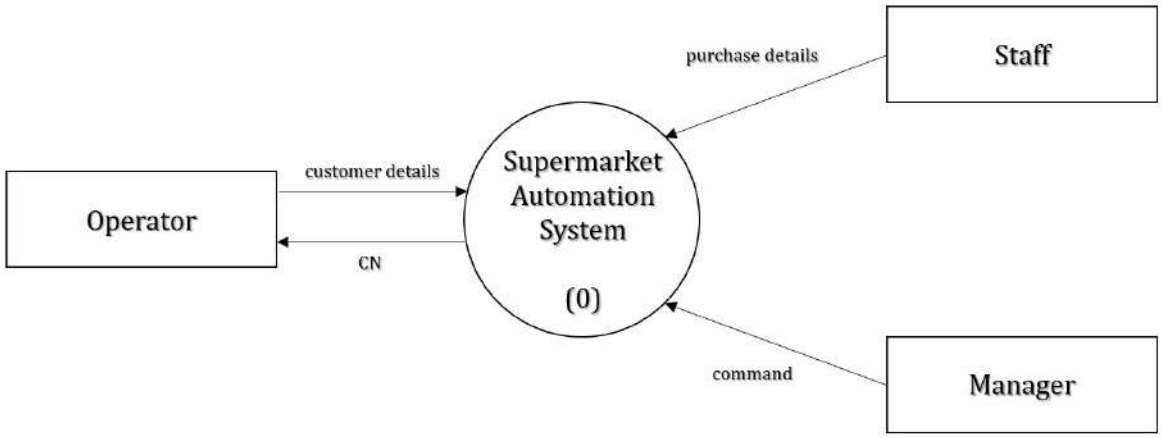


Fig – DFD level zero

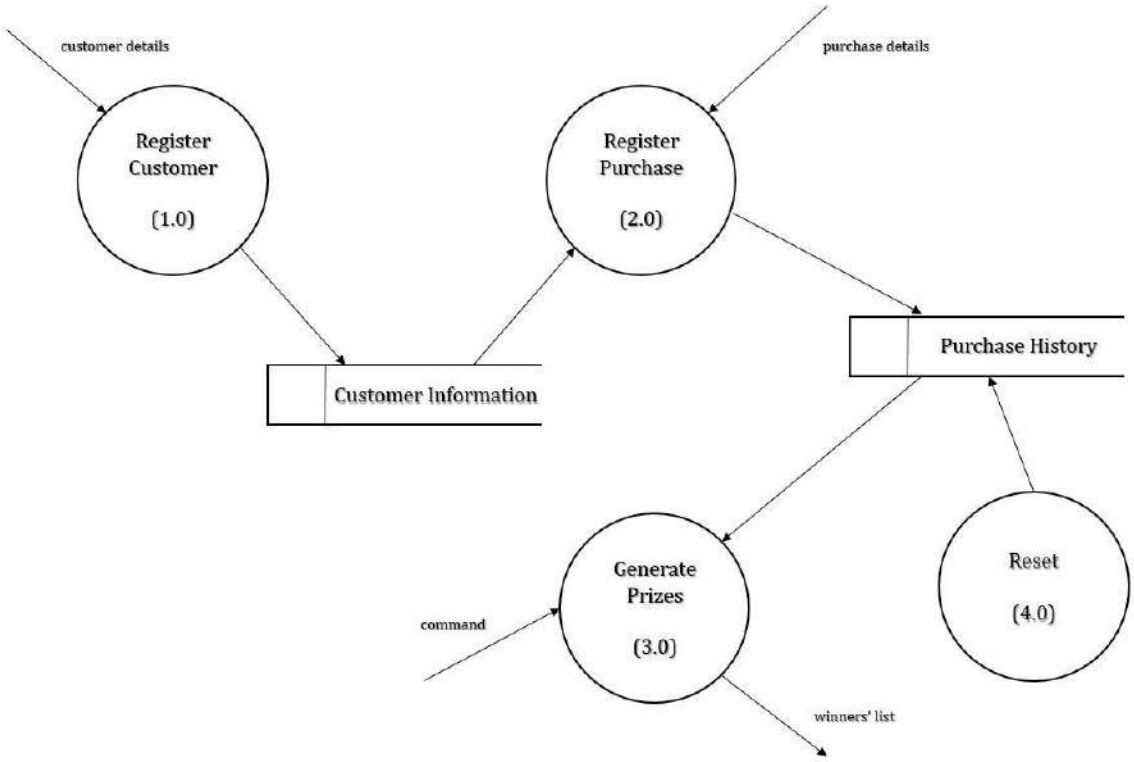


Fig – DFD level one

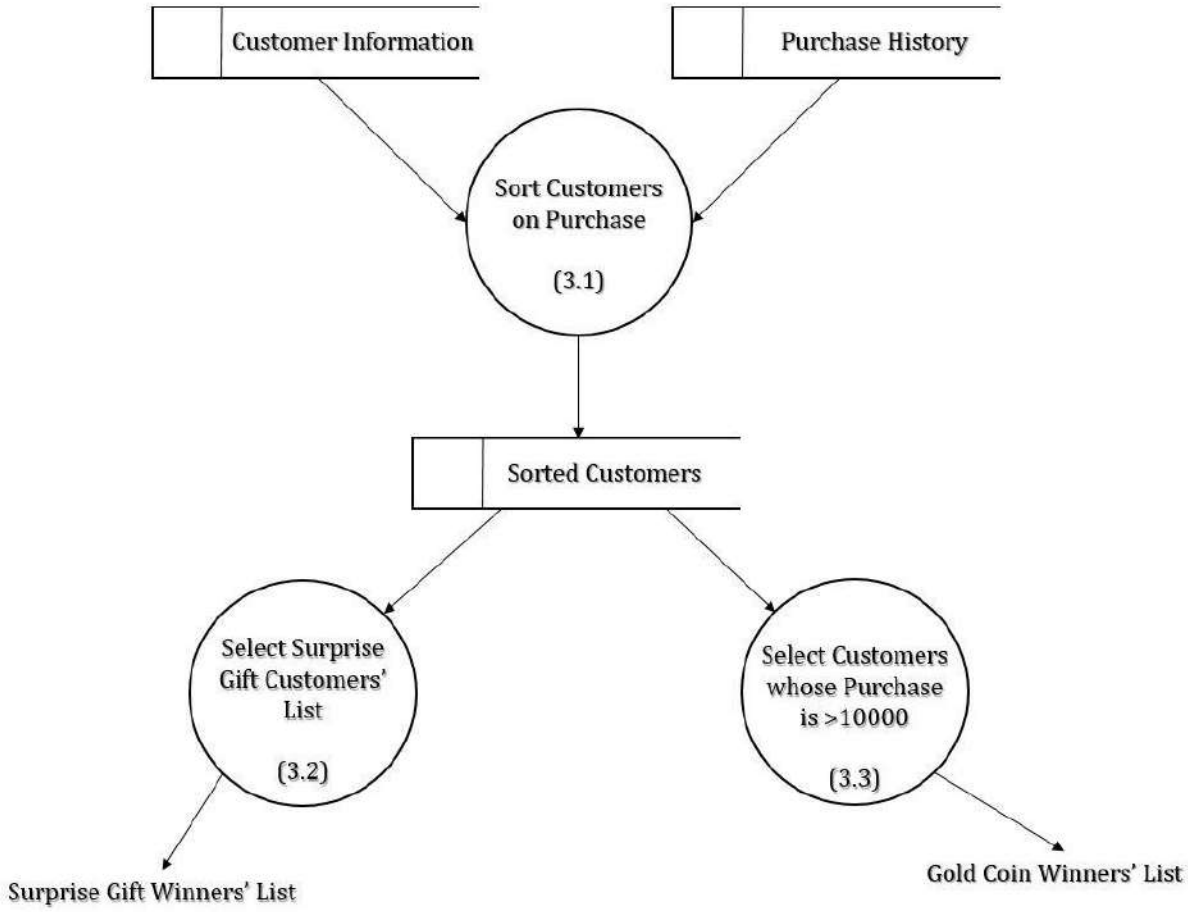
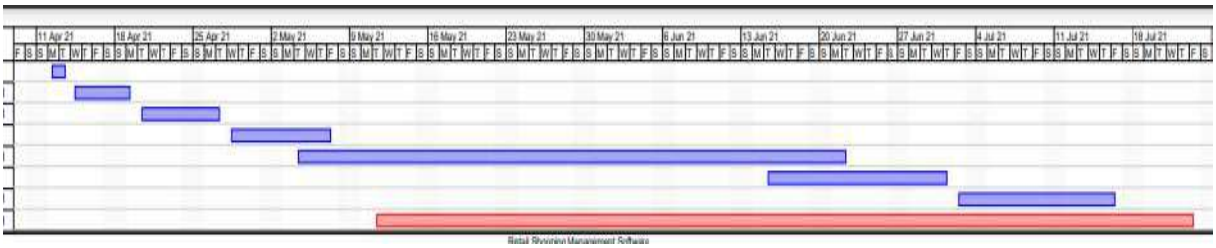


Fig – DFD level two of Process 3.0

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 2

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement – It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model – Incremental model

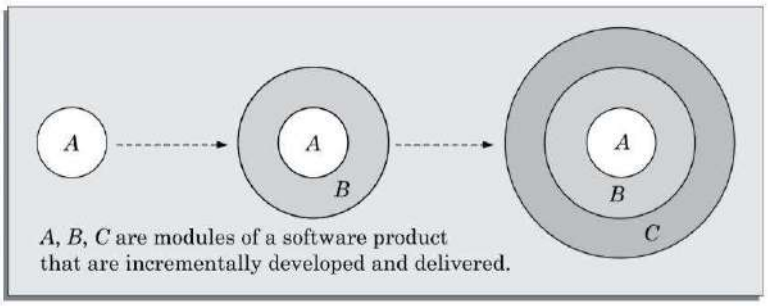


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module –** In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module –** In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module –** In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

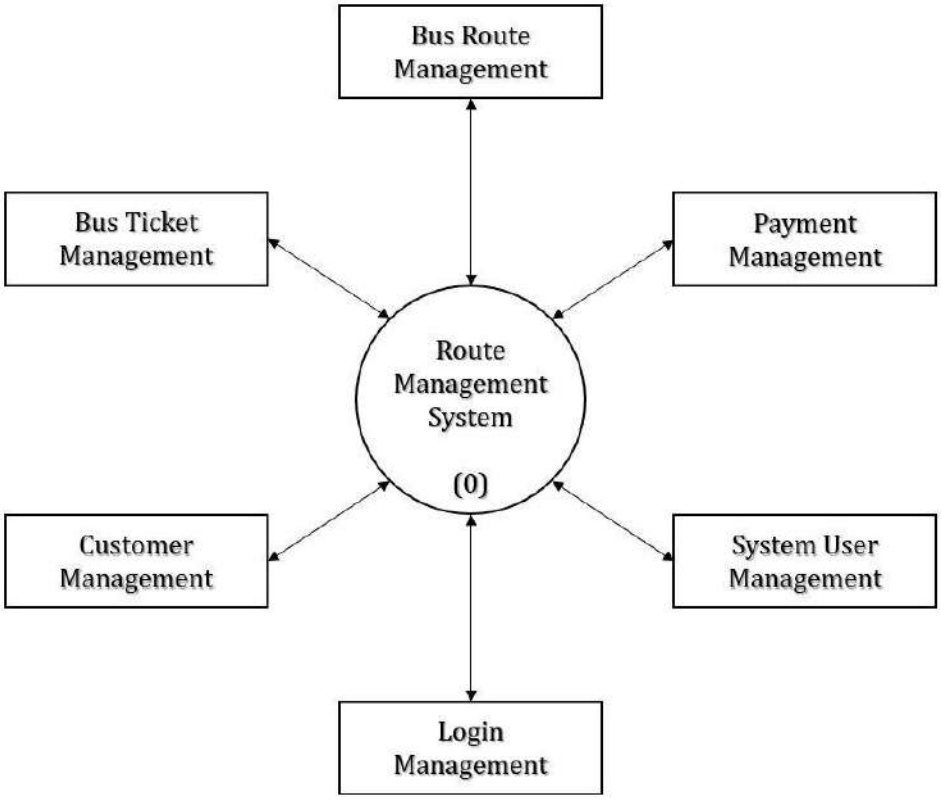


Fig – DFD level zero

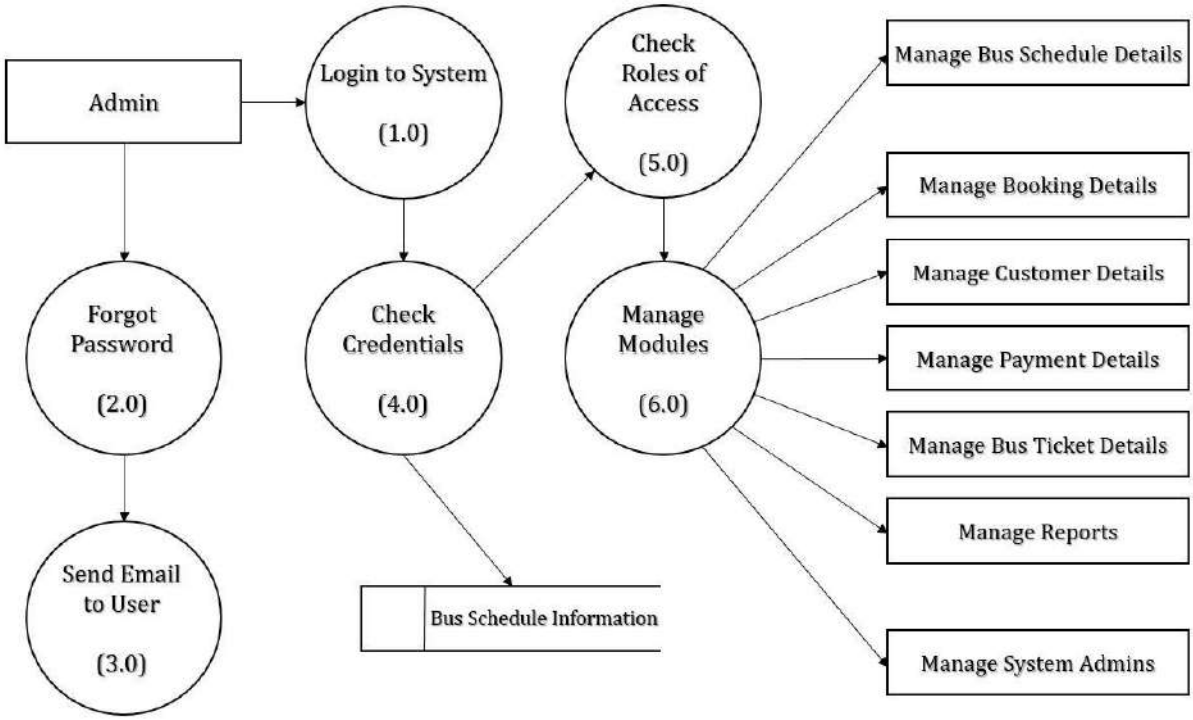


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 3

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement – Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model – Incremental development model

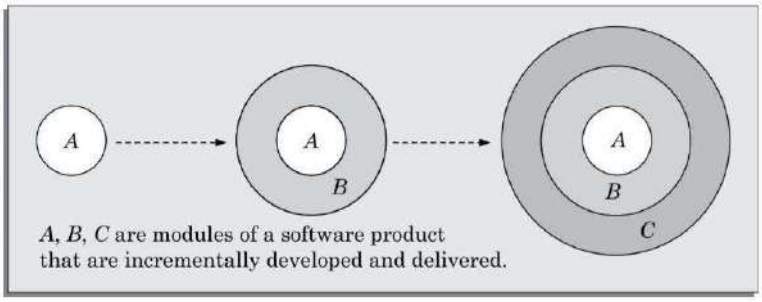


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module –** In this module, a patient is registered and the profile is maintained.

   Input: patient details
   Output: unique profile

2. **Doctor Module –** In this module, the details of doctors are maintained.

   Input: patient details
   Output: updated database

3. **Diagnosis Module –** In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input: patient details, doctor details, test details
   Output: diagnosed disease, updated database

4. **Report Module –** In this module, patient records, medicine and drug records, test records are processed and reports are generated.

   Input: patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

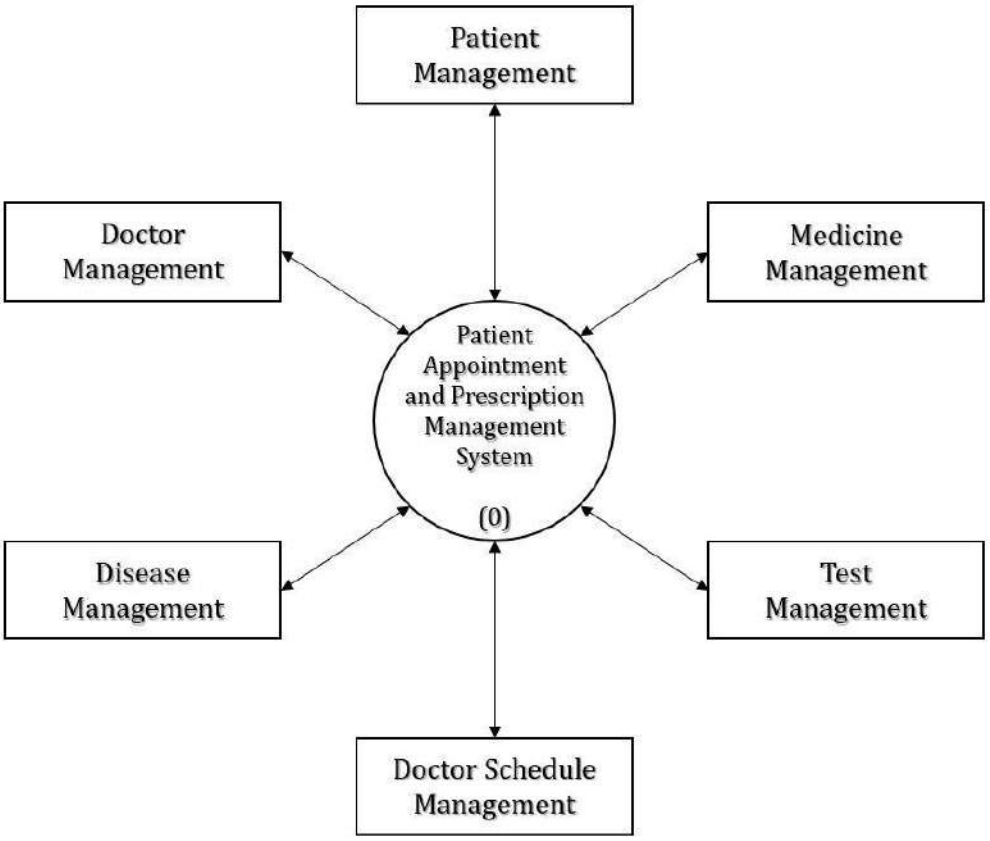


Fig – DFD level zero

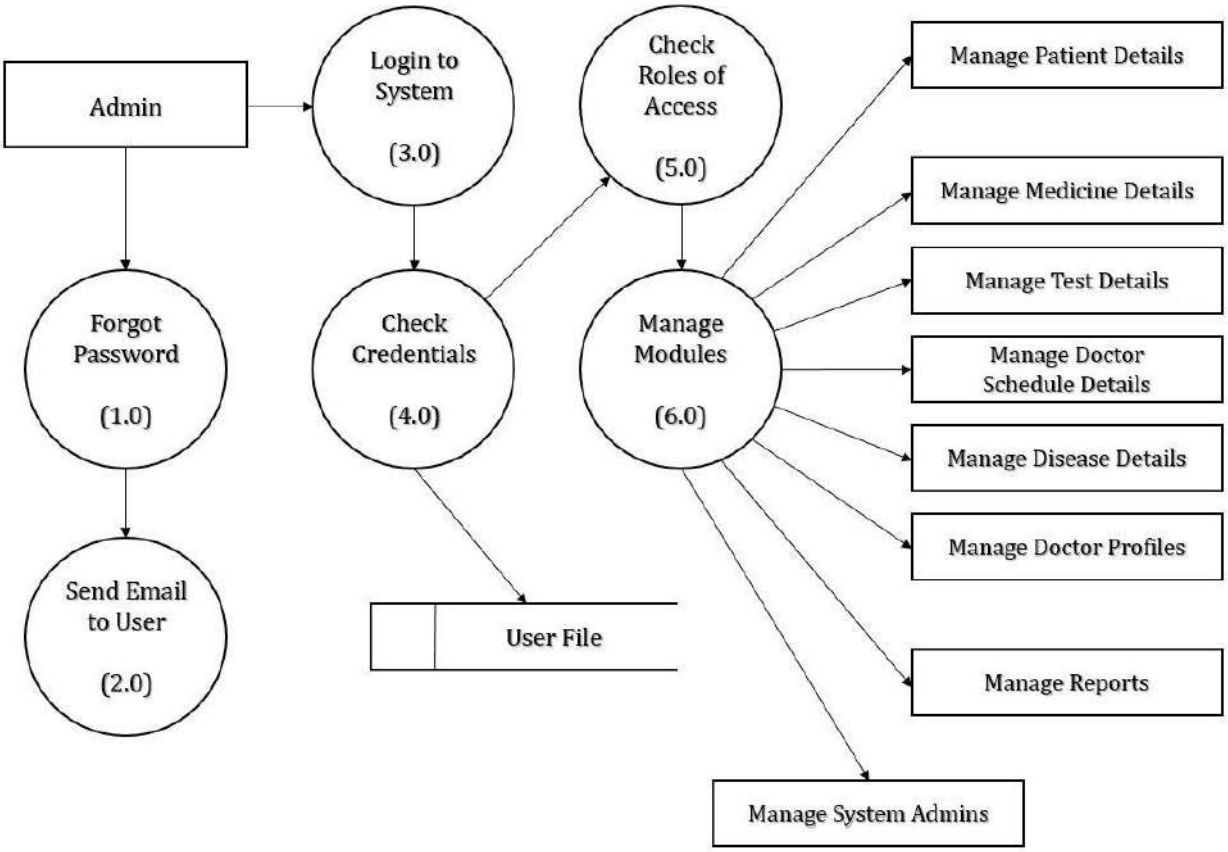
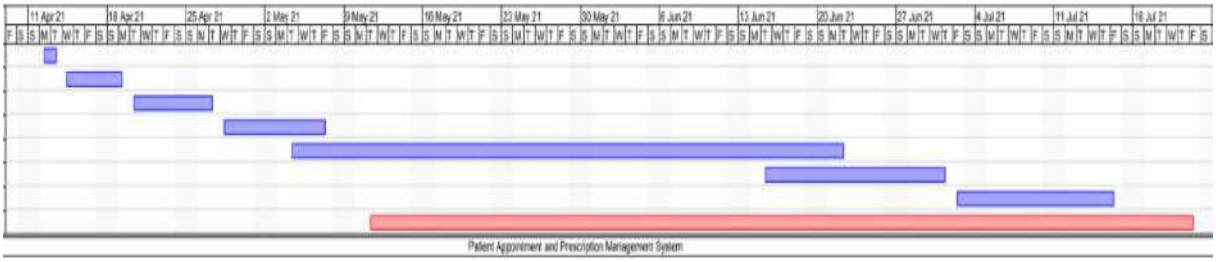


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 4

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement – Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model – Spiral development model

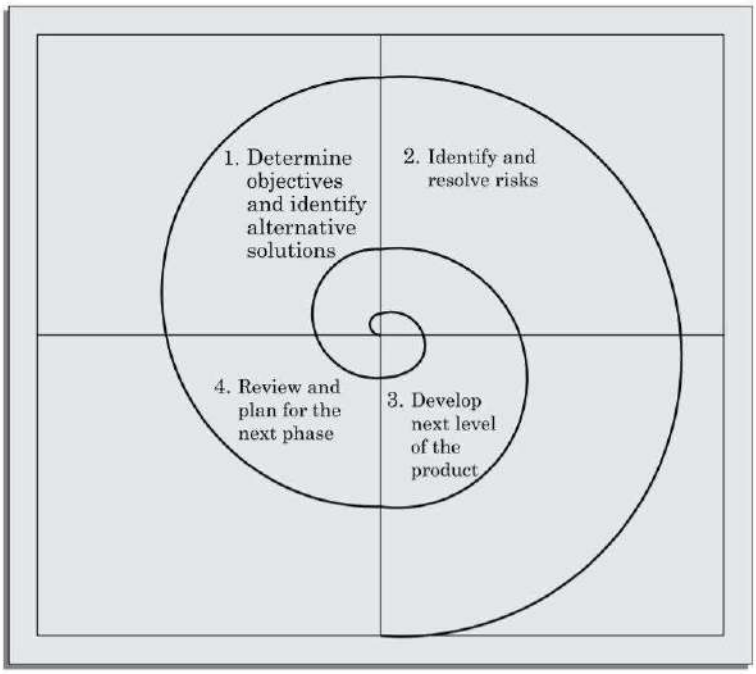


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module –** In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input: guest details
   Output: unique profile, updated database

2. **Catering Module –** In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input: guest details, room number, token number
   Output: updated database

3. **Bill Module –** In this module, the total bill is generated when the guest prepares to check out.

   Input: guest details, room number, token number, food items consumed
   Output: generated bill

Design Engineering

Data Flow Diagrams (DFDs)

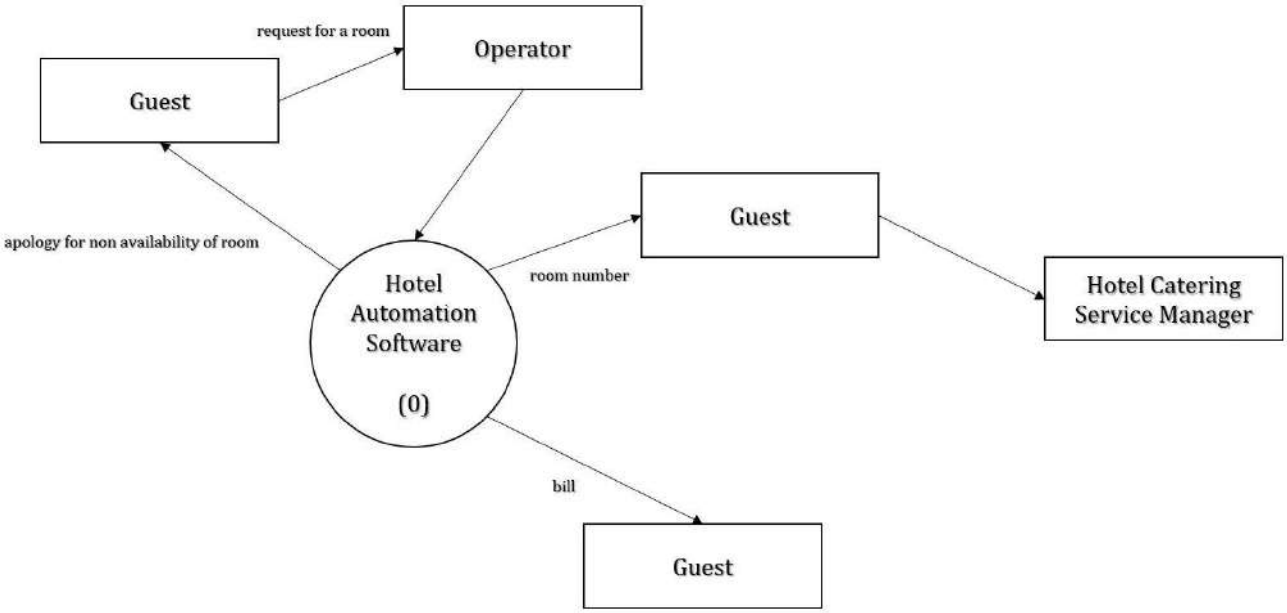


Fig – DFD level zero

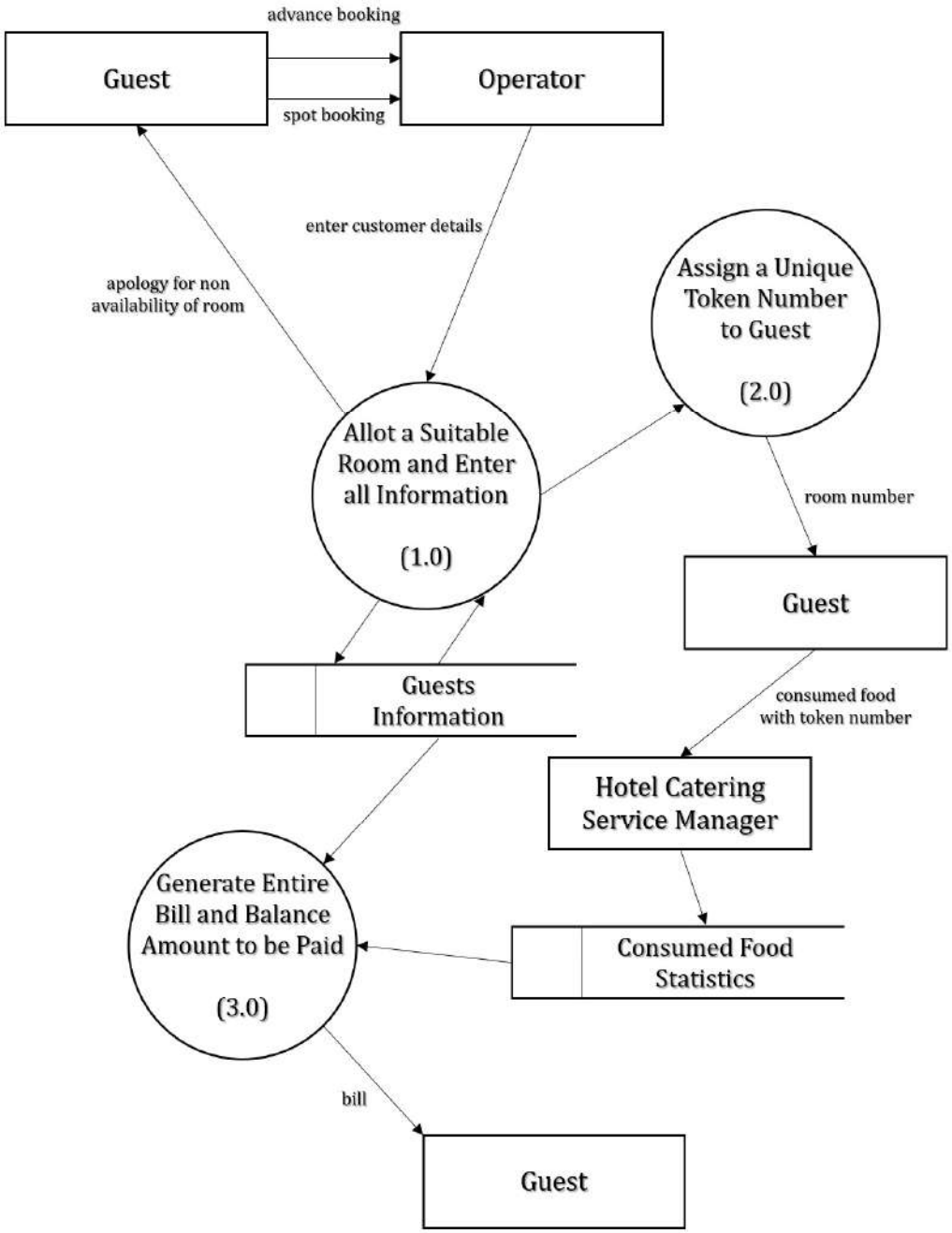


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 5

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement – Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

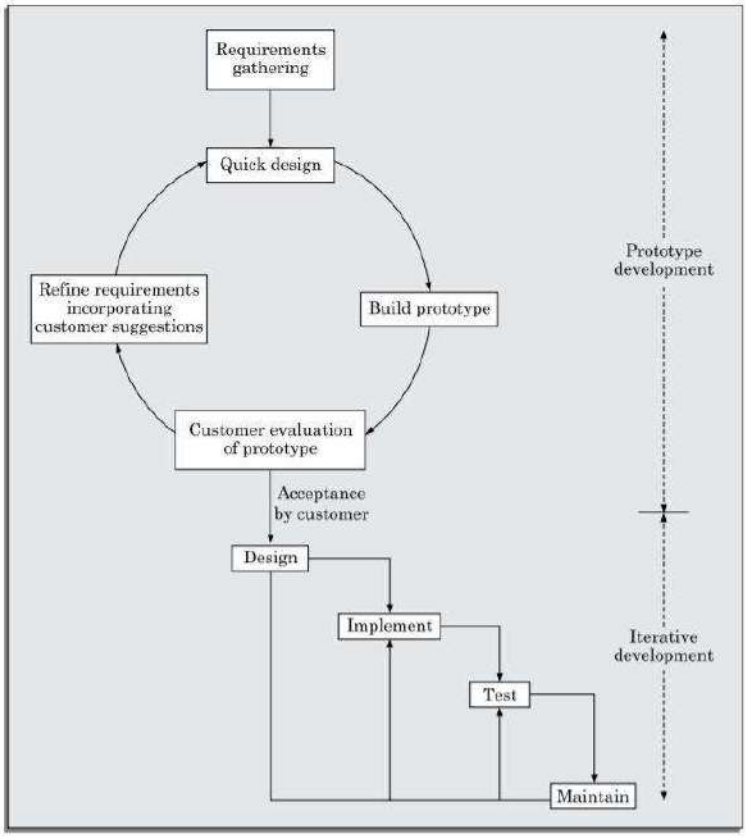


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module –** In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input: admin details
   Output: unique profile, granted level of access

2. **Complain Module –** In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output: filed FIR, updated database

3. **Crime Module –** In this module, the details of the crime are maintained and chargesheet is filed.

   Input: complaint details, FIR details
   Output: filed chargesheet

4. **Court Module –** In this module, the case is taken to the court and legal action is taken accordingly.

   Input: complaint details, FIR details, chargesheet details
   Output: legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

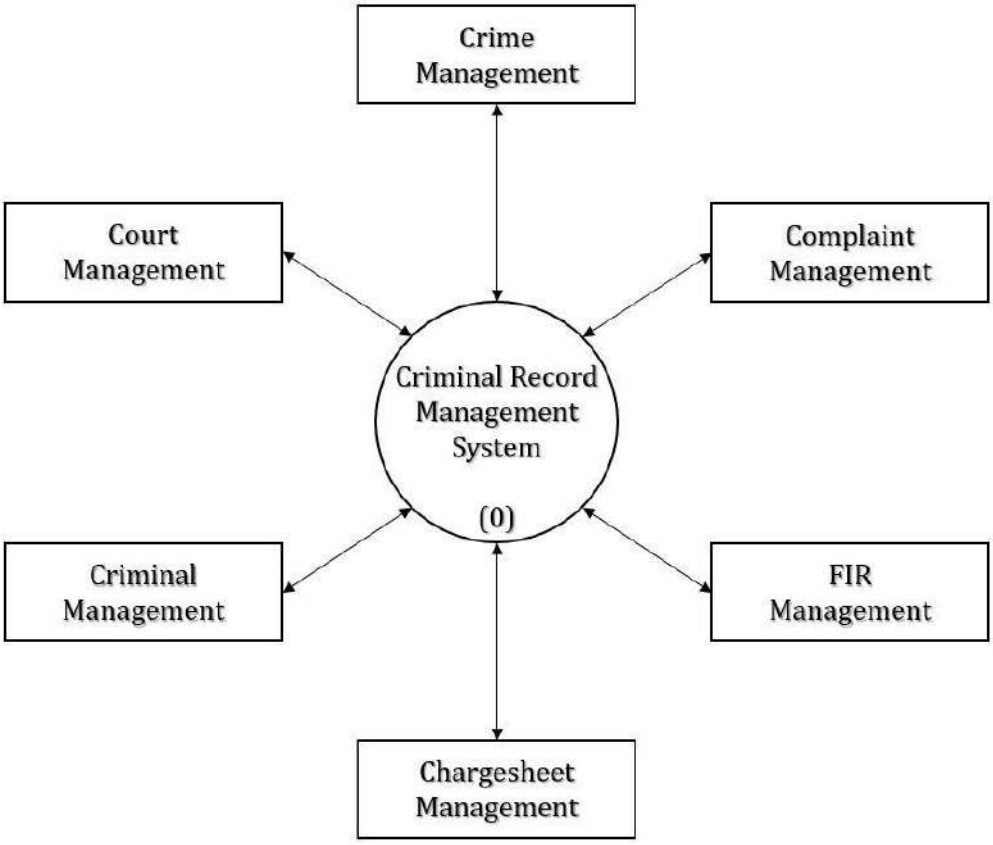


Fig – DFD level zero

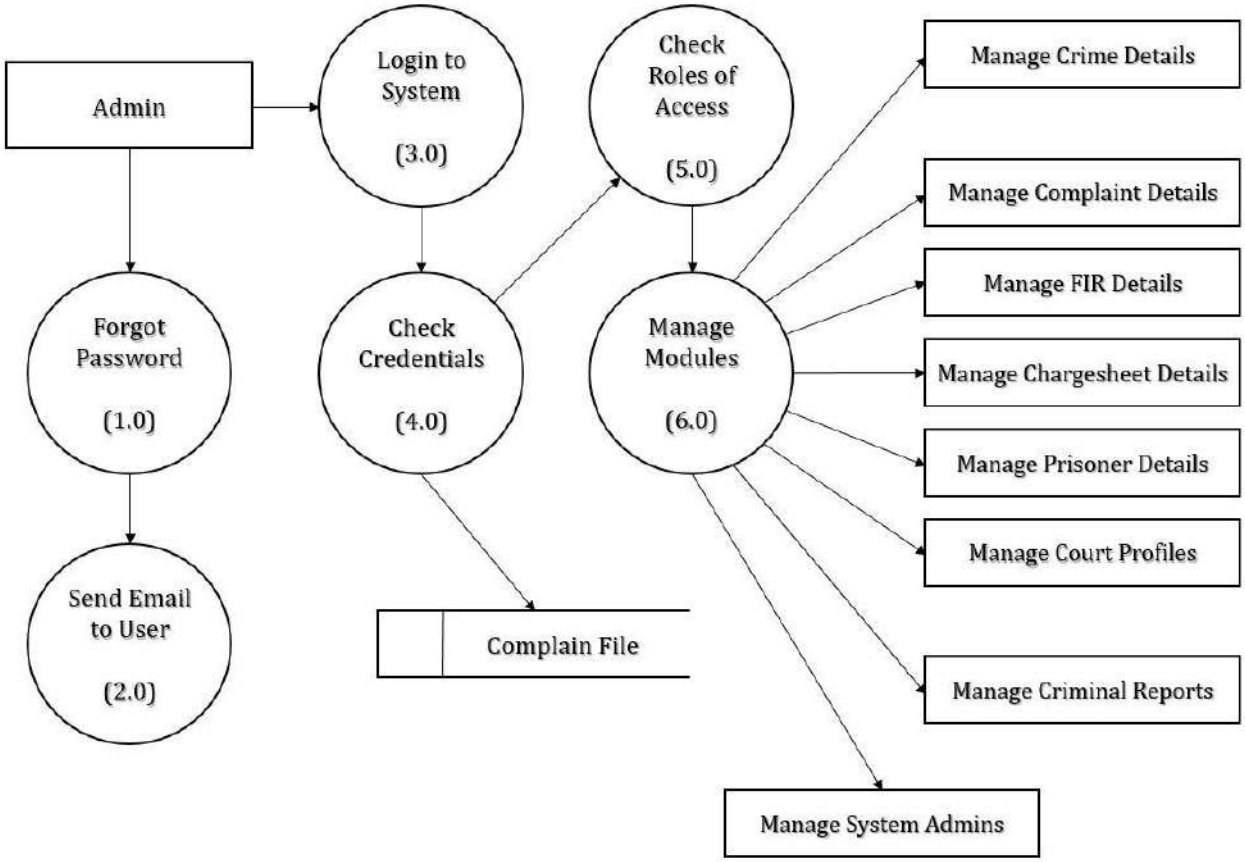
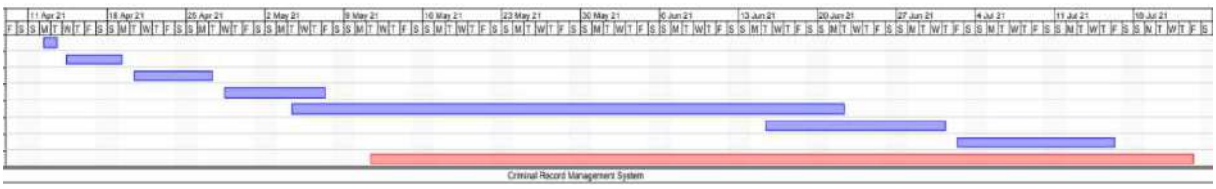


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers

themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 6

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement – Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model – Spiral development model

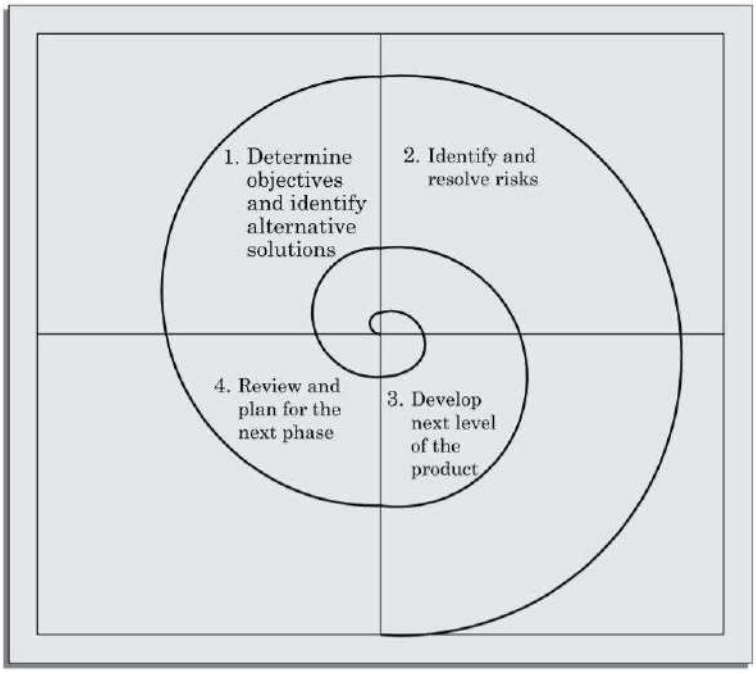


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the

software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module –** In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module –** In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module –** In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module –** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

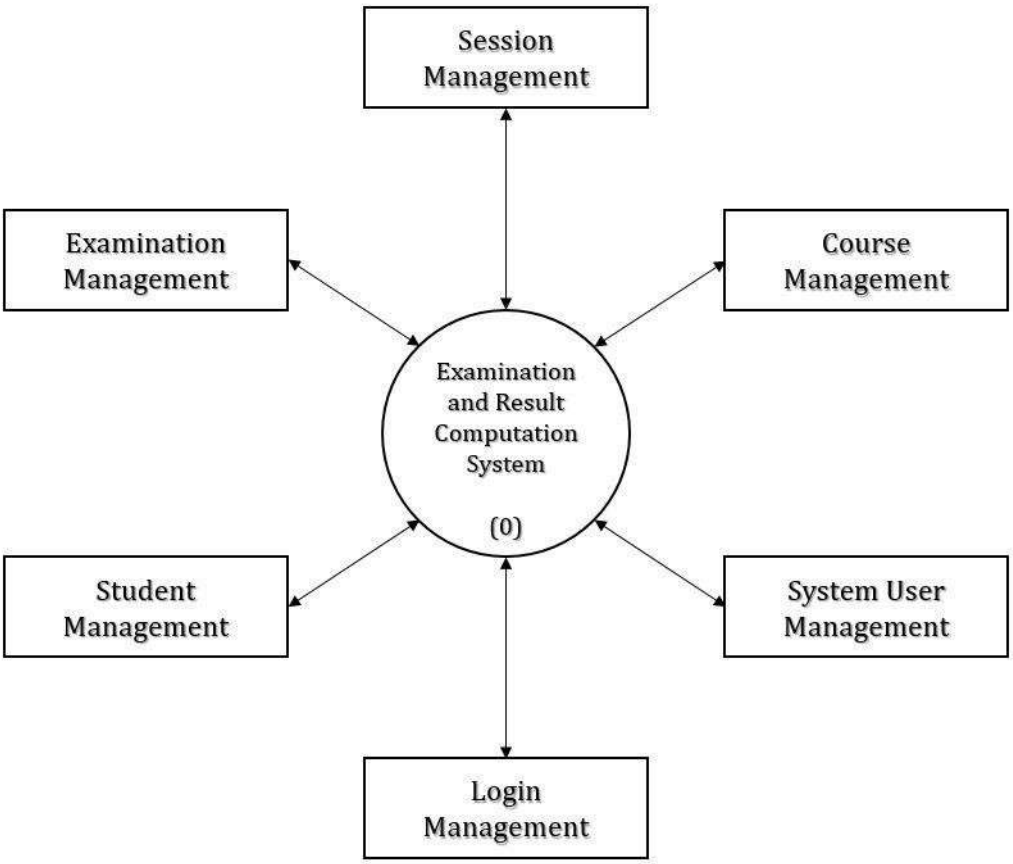


Fig – DFD level zero

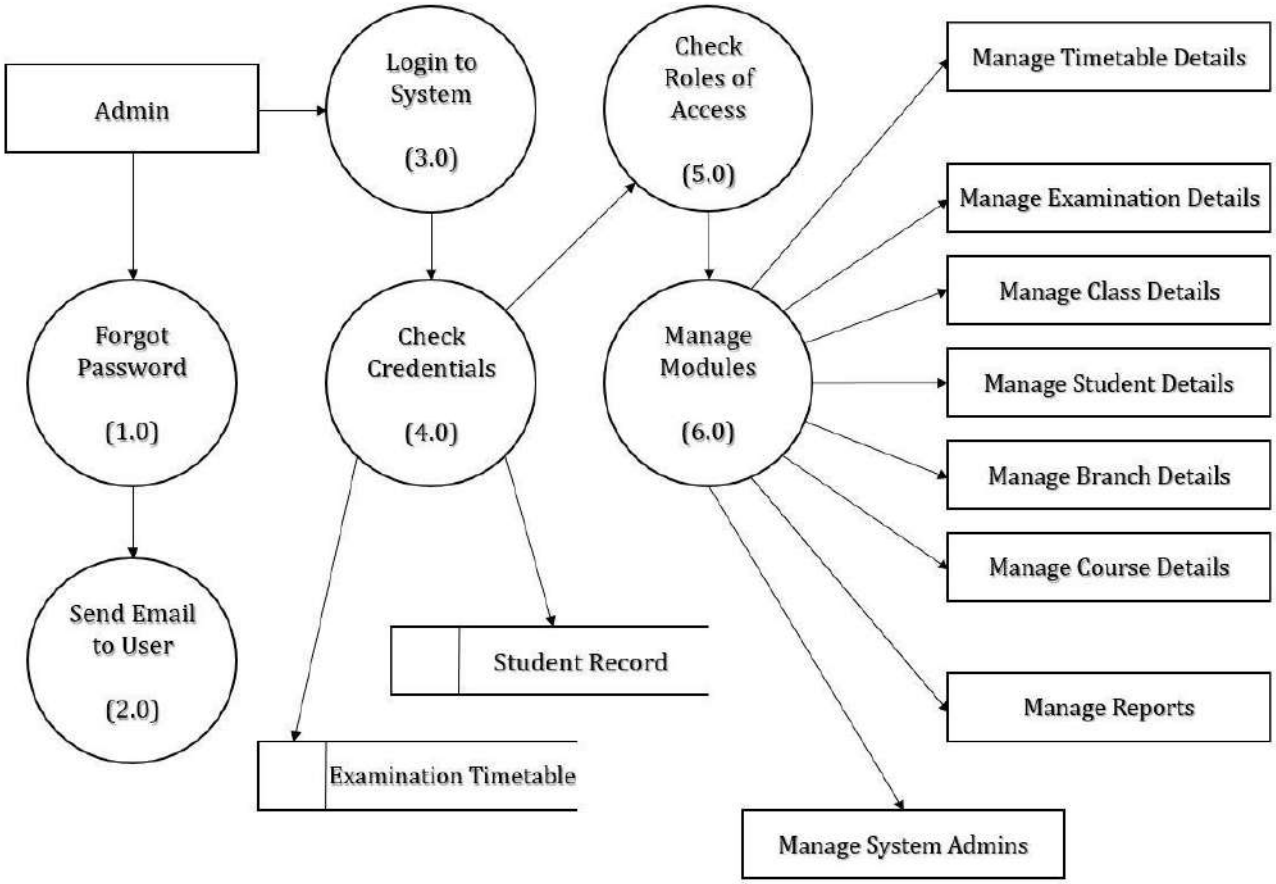
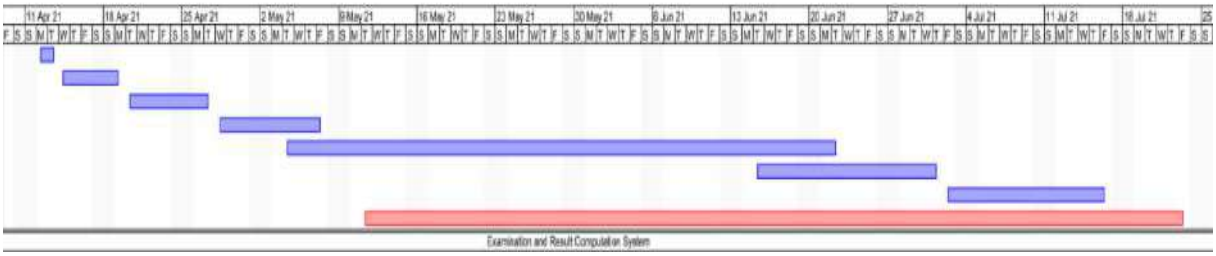


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 7

PARKING ALLOCATION SYSTEM

Problem Statement – Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model – Spiral development model

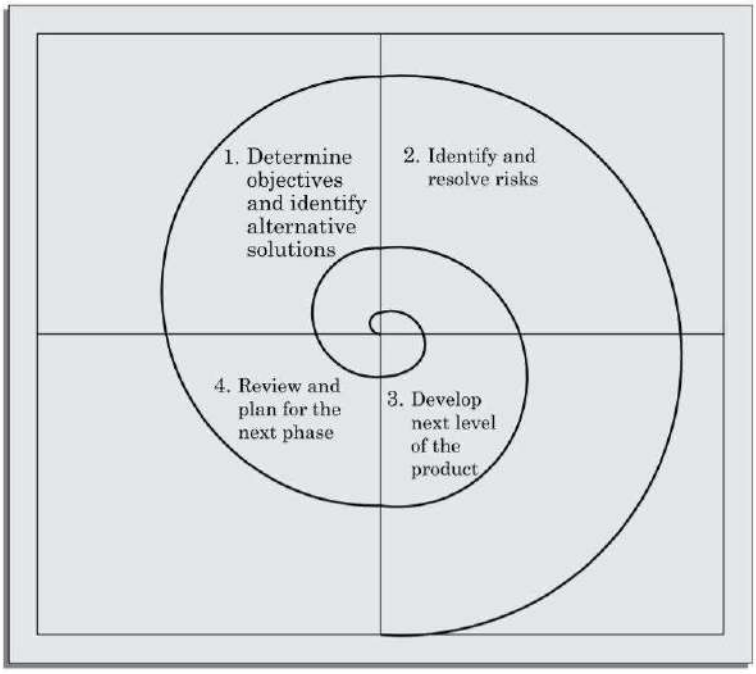


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at

the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module –** In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module –** In this module, parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module –** In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

Fig – DFD level zero

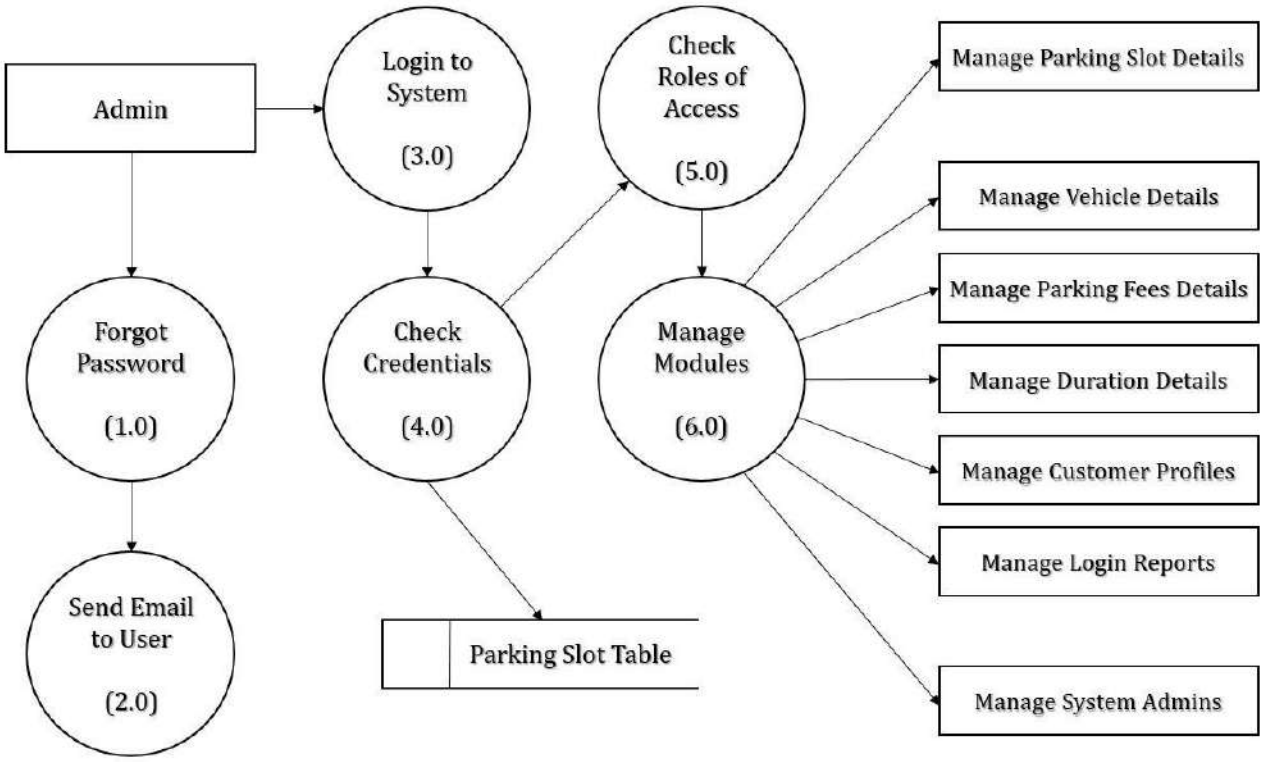
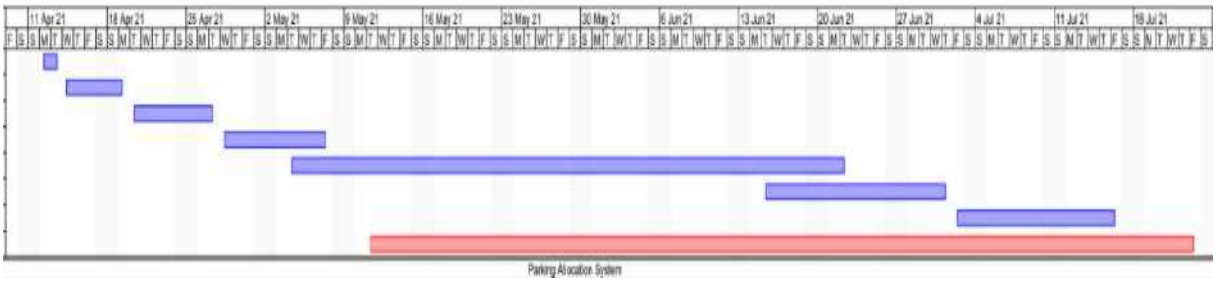


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Project 8

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

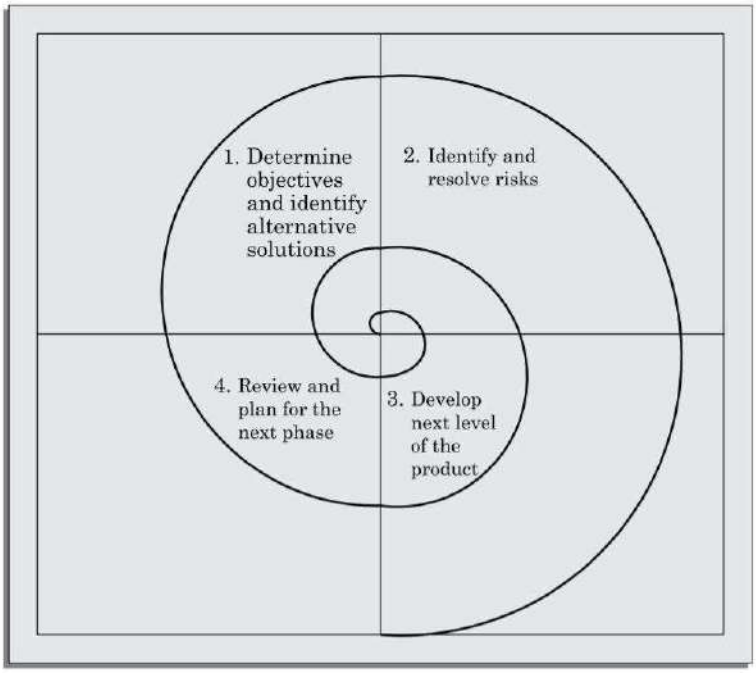


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a

prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

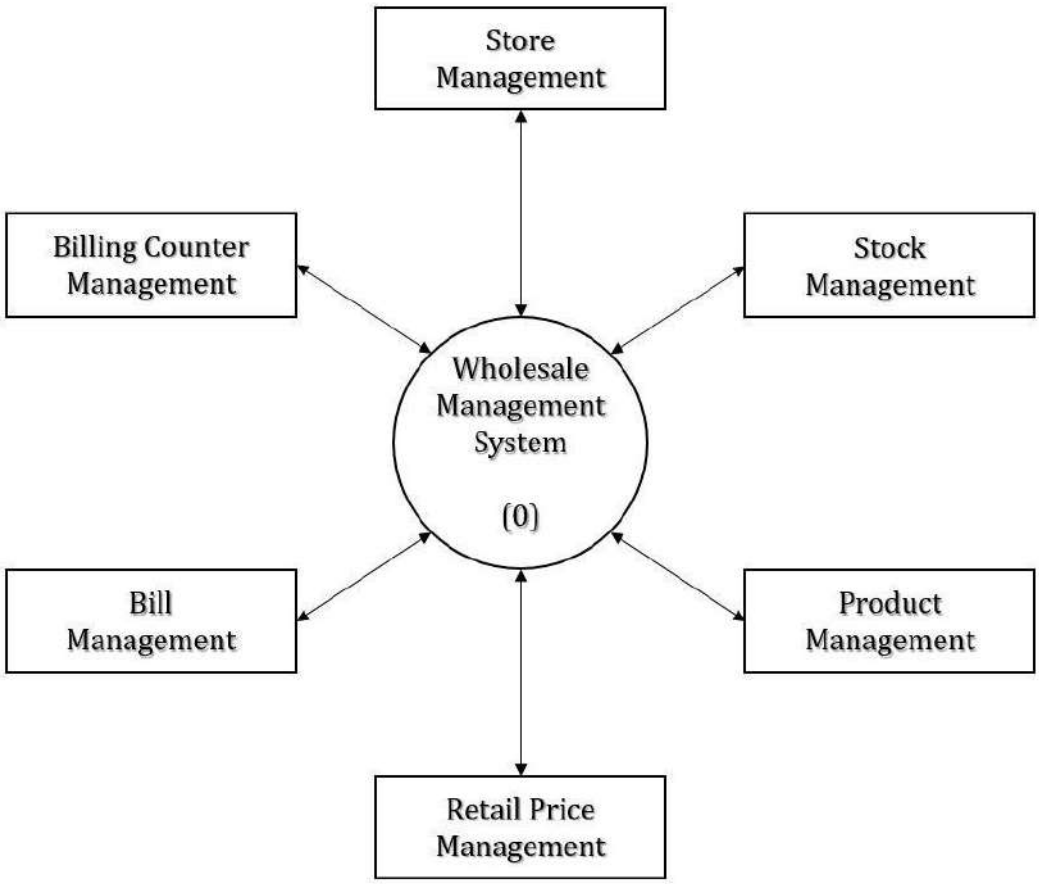


Fig – DFD level zero

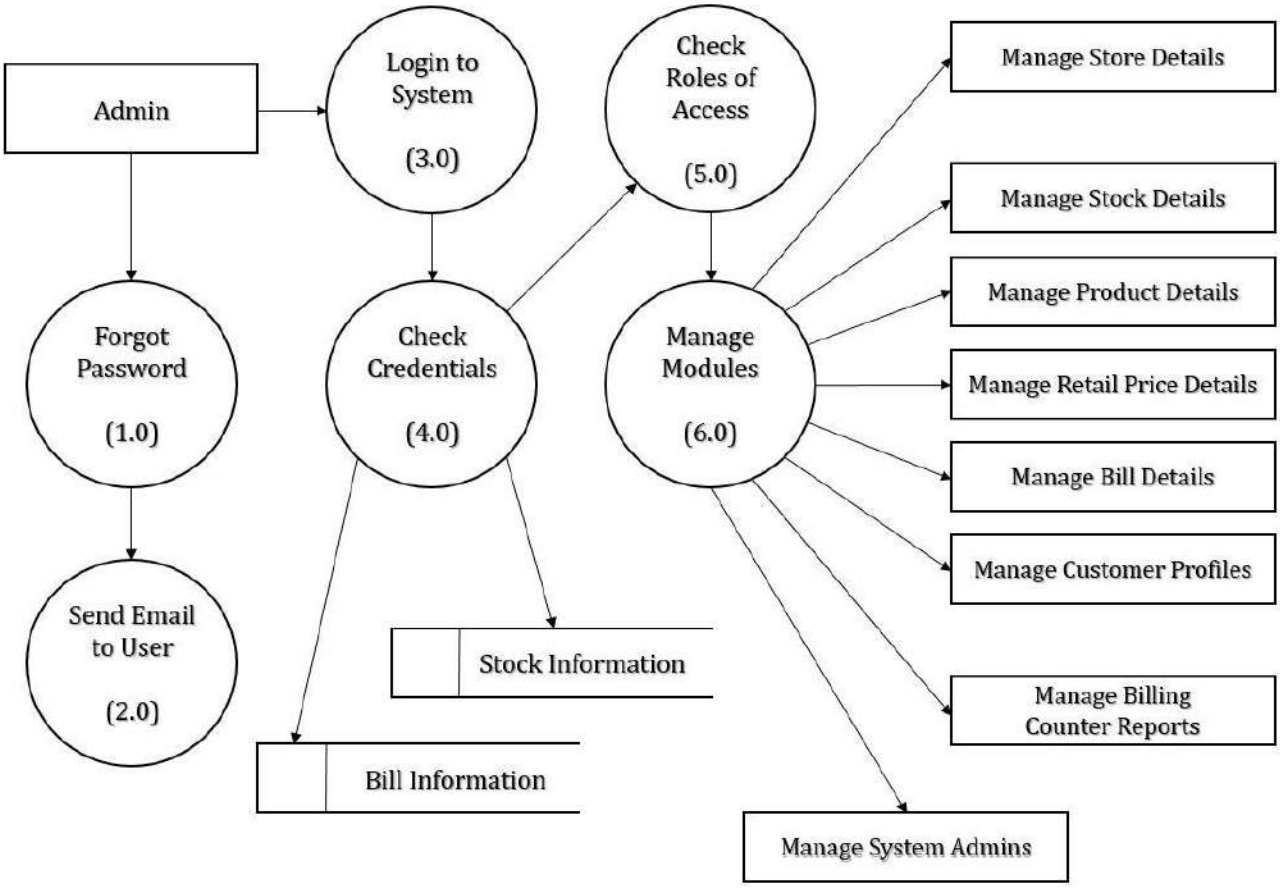


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

RAMAKRISHNA MISSION

VIVEKANANDA CENTENARY COLLEGE

COMPUTER SCIENCE DEPARTMENT

PROJECT ON SOFTWARE ENGINEERING

SUMBITTED BY:

NAME : SWAPNANIL GHOSH

COLLEGE ROLL NO. : 718

EXAMINATION ROLL NO. : 2021141299

REGISTRATION NO. : A01-1112-117-015-2019

SEMESTER : V

SUPERVISOR: Dr. BISWAJIT BISWAS

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement– A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

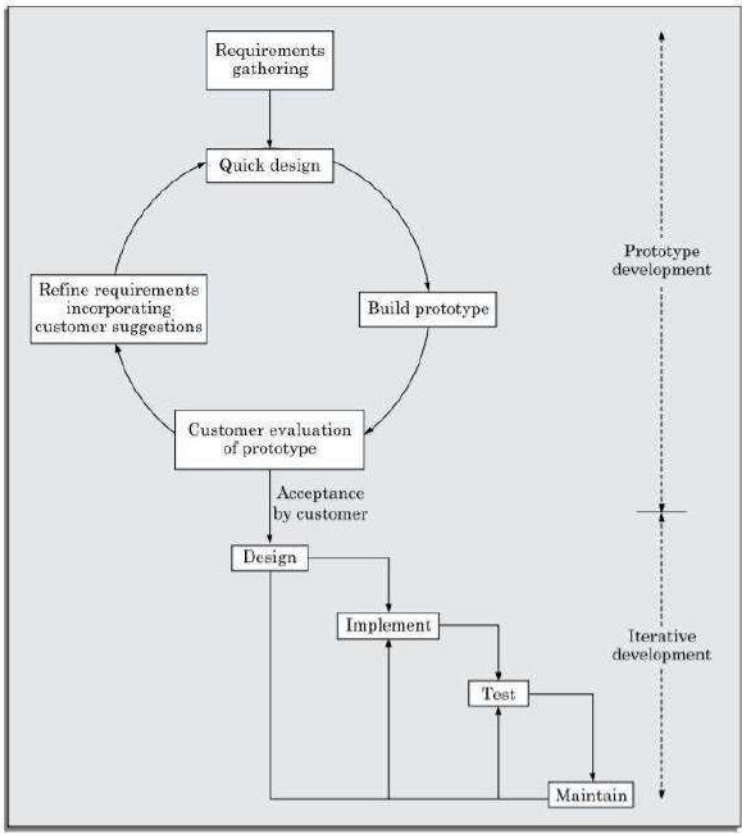


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly.Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module**–In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: uniqueCN

2. **Purchase Module**–In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input:purchase made by the customer, CN
   Output: updated database

3. **Promotional Module**–In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output:prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

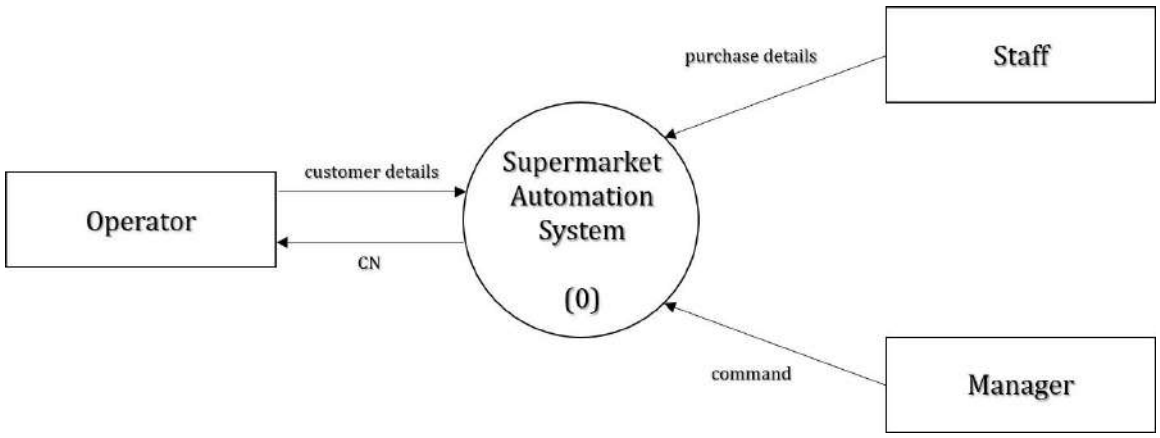


Fig – DFD level zero

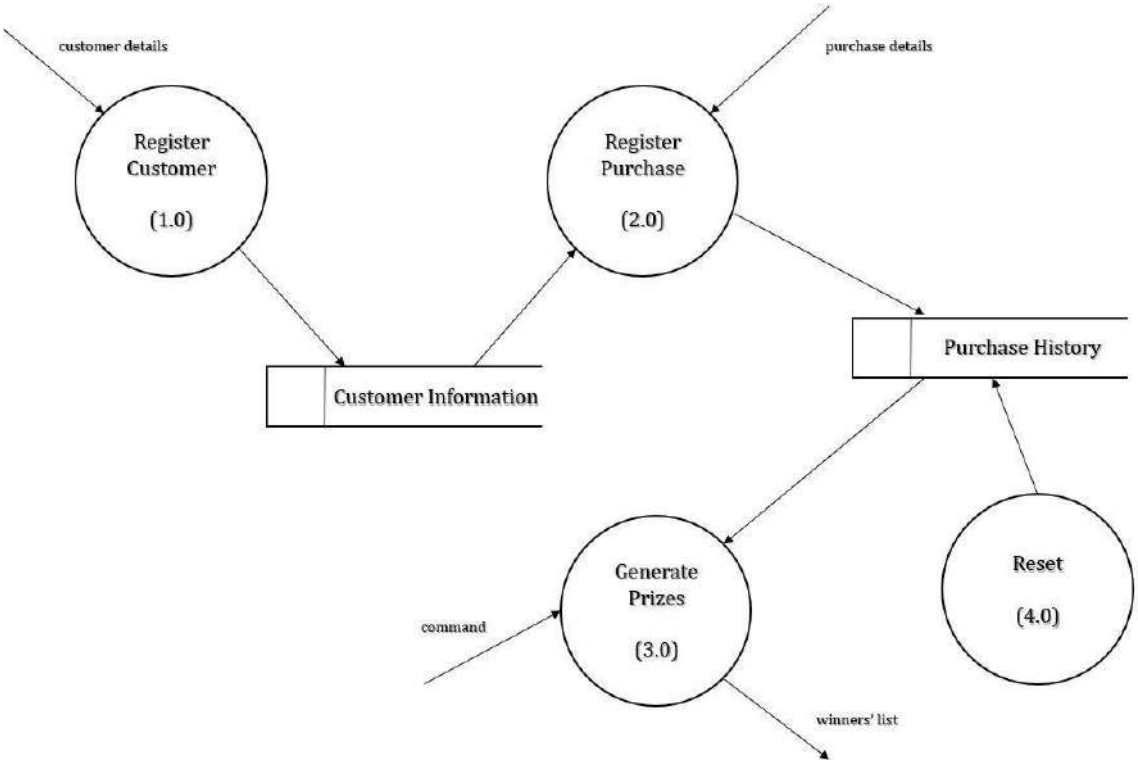


Fig – DFD level one

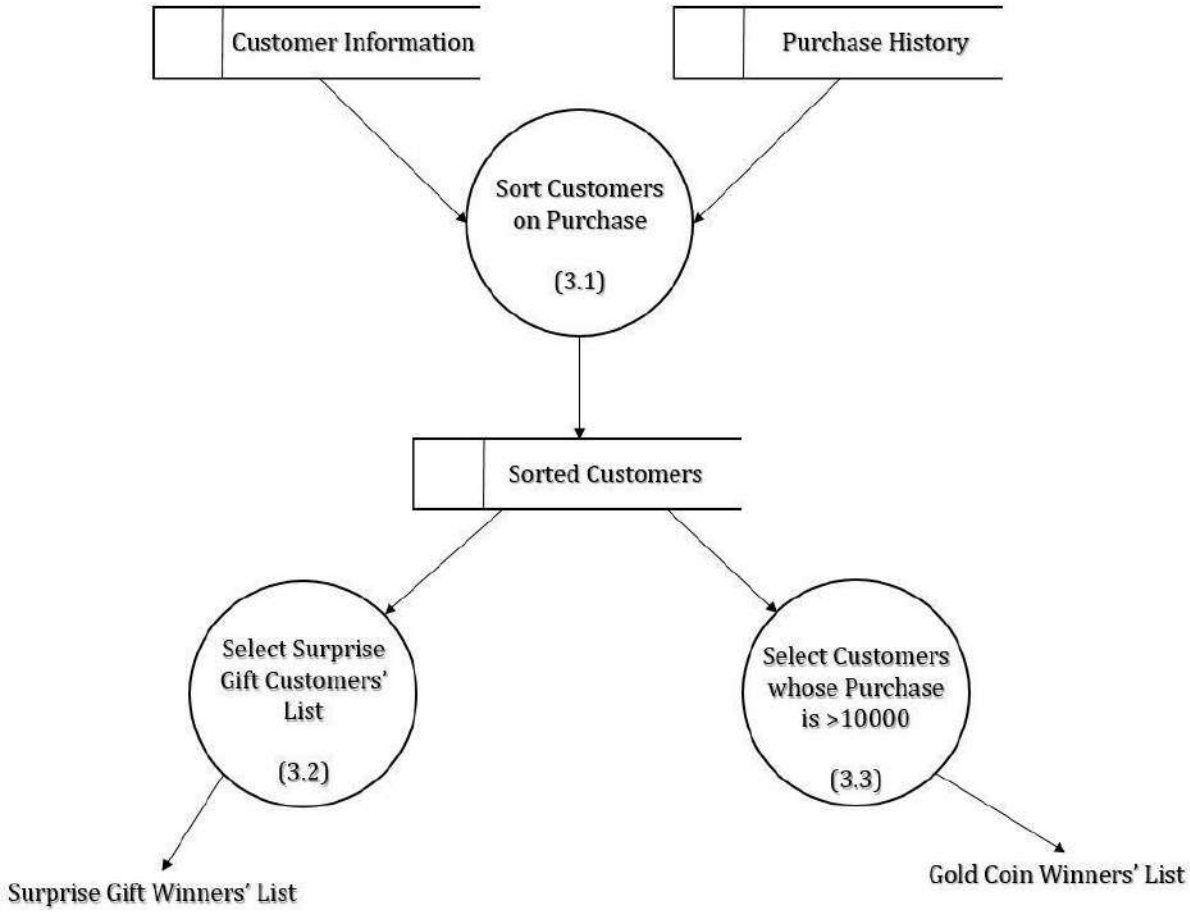
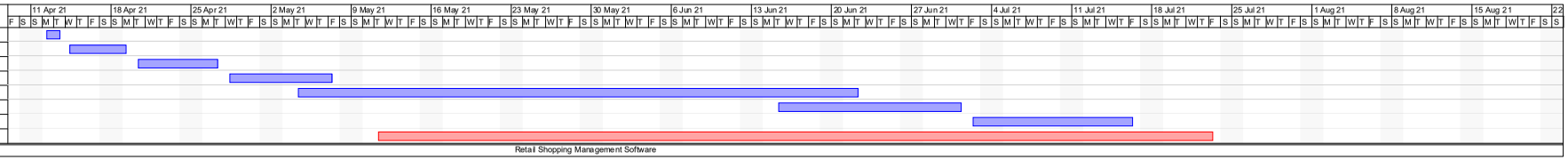


Fig – DFD level two of Process 3.0

Project Management

Project Schedule–The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Retail Shopping Management Software

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement –It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model –Incremental model

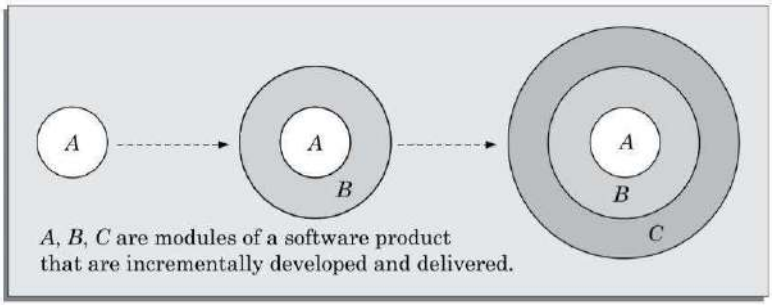


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module**–In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module**–In this module, any booking made by the customer is registered andthe ticket is generated.

   Input: customer details
   Output:generated ticket

3. **Information Module**–In this module,the busdetails of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module**–In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input:booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

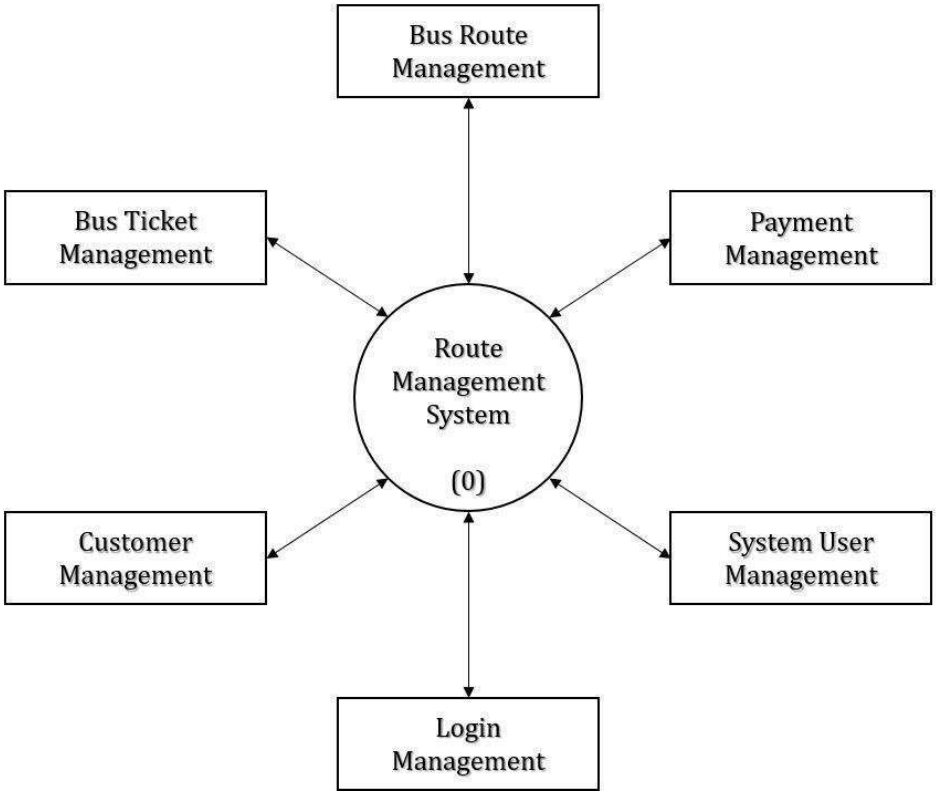


Fig – DFD level zero

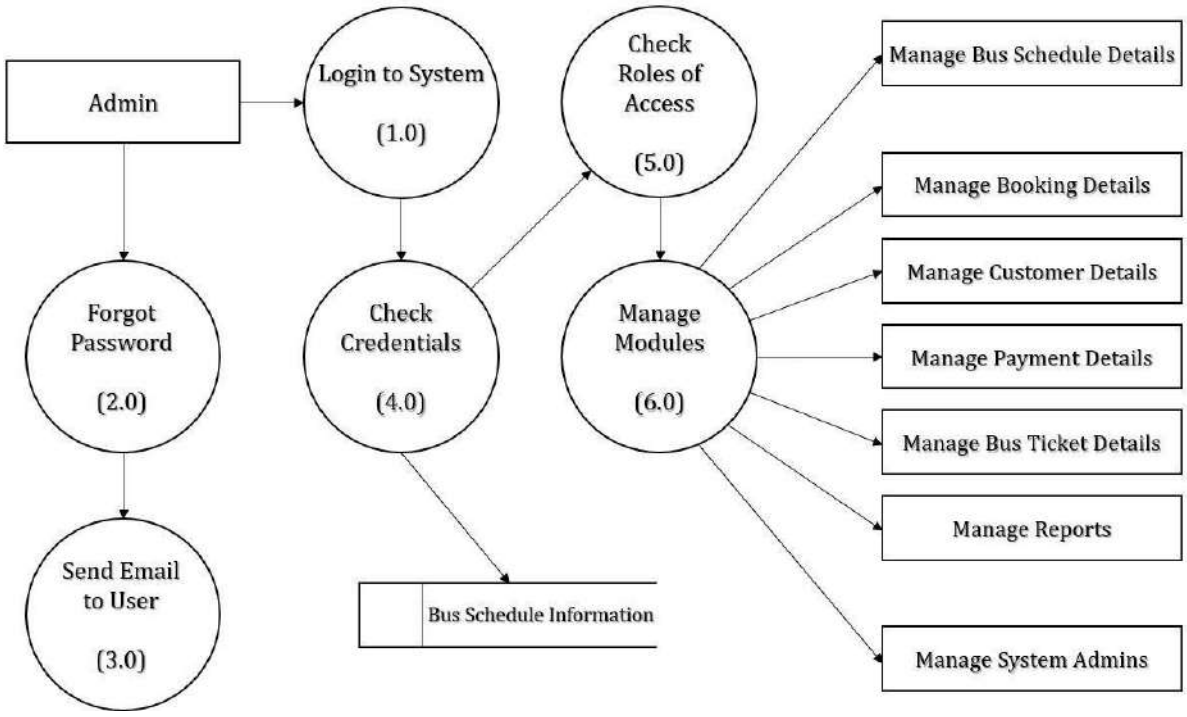
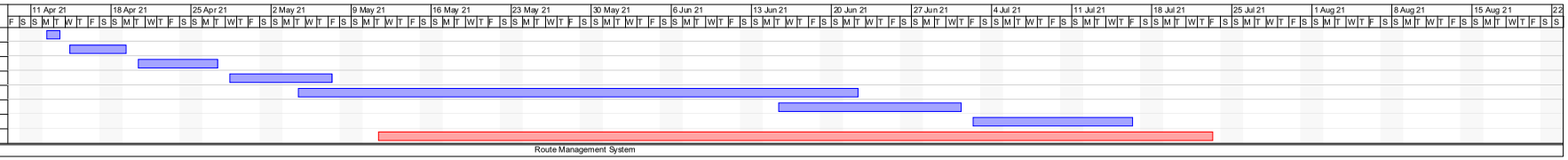


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement –Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model –Incremental development model

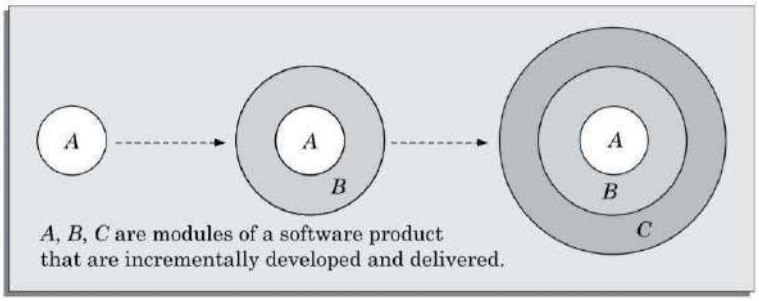


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model.In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered.The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model.The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module**–In this module, a patient is registered and the profile is maintained.

   Input:patient details
   Output: unique profile

2. **Doctor Module**–In this module, the details of doctors are maintained.

   Input:patient details
   Output:updated database

3. **Diagnosis Module**–In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input:patient details, doctor details, test details
   Output:diagnosed disease, updated database

4. **Report Module**–In this module, patient records, medicine and drug records, test recordsare processed and reports are generated.

   Input:patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

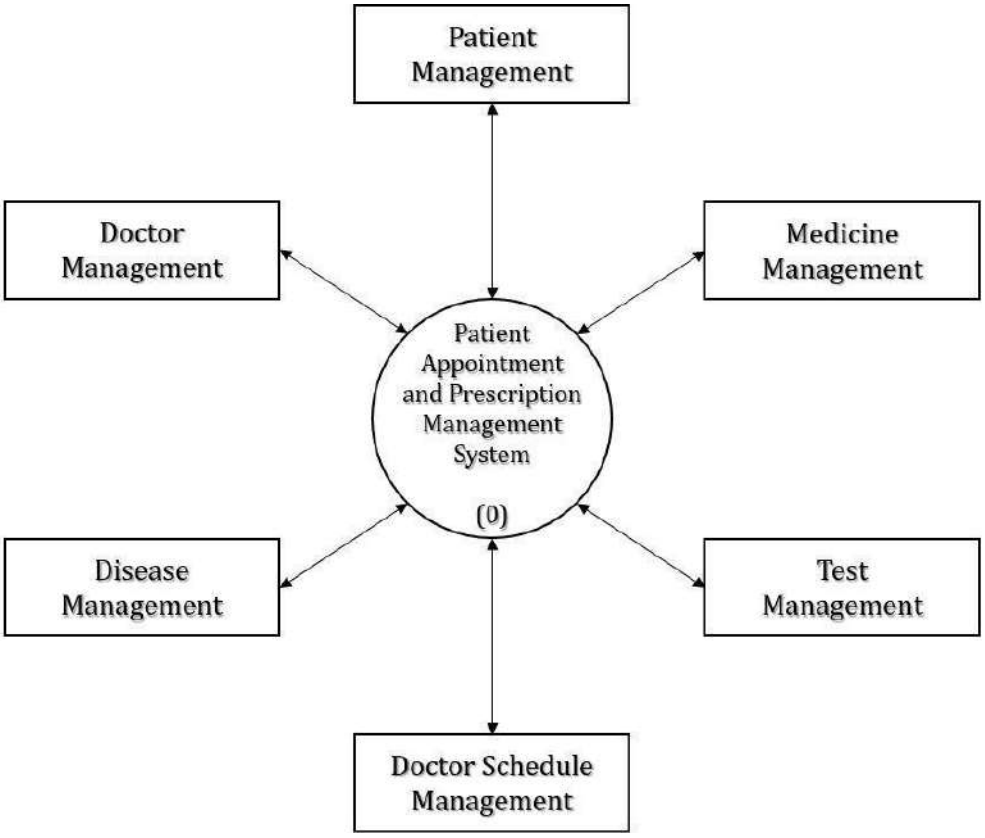


Fig – DFD level zero

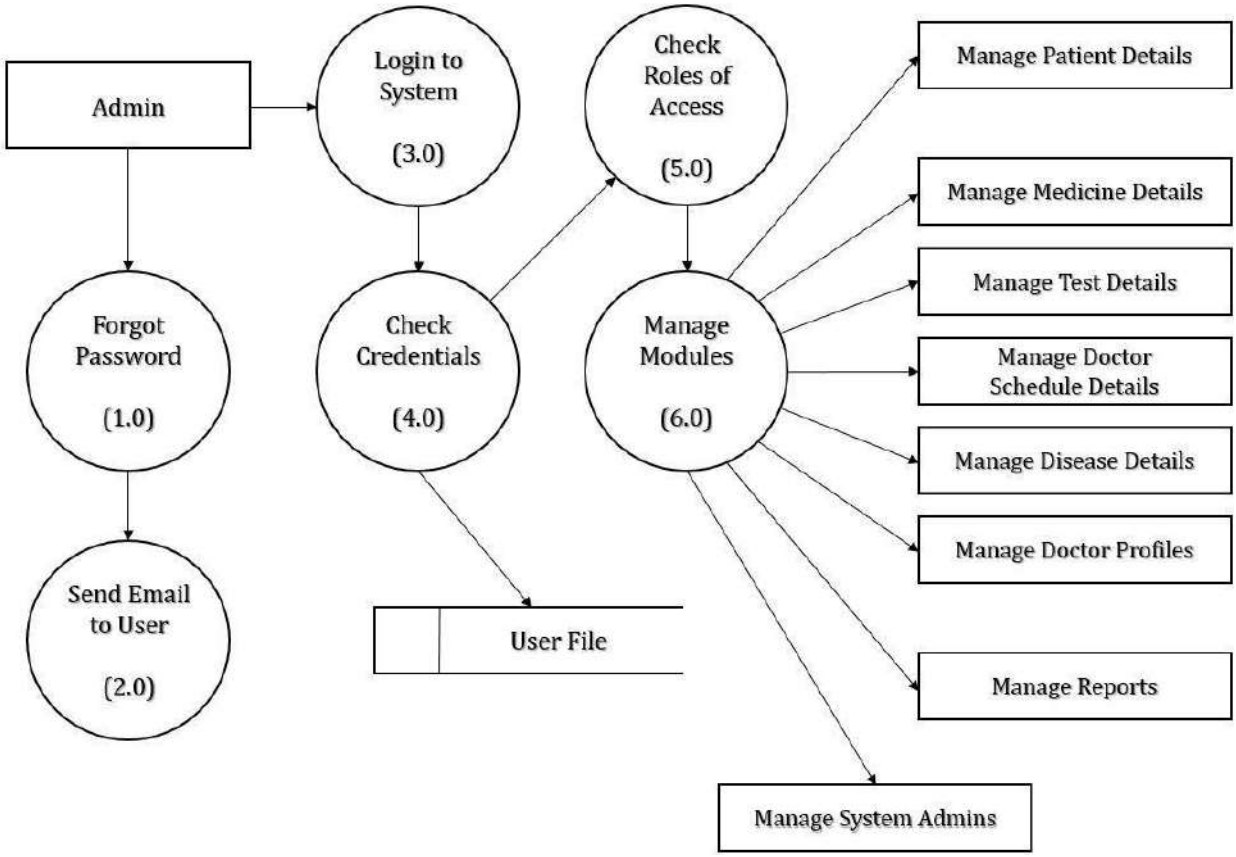
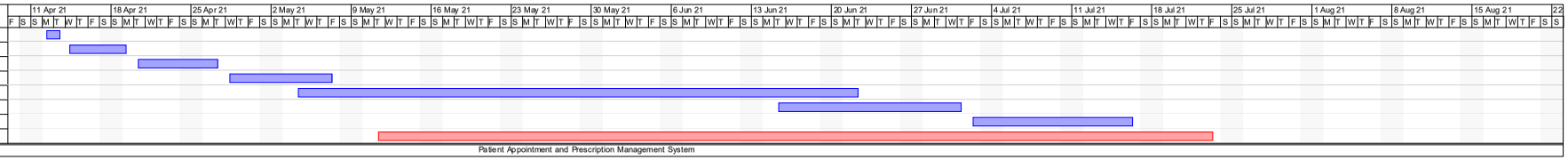


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement –Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model –Spiral development model

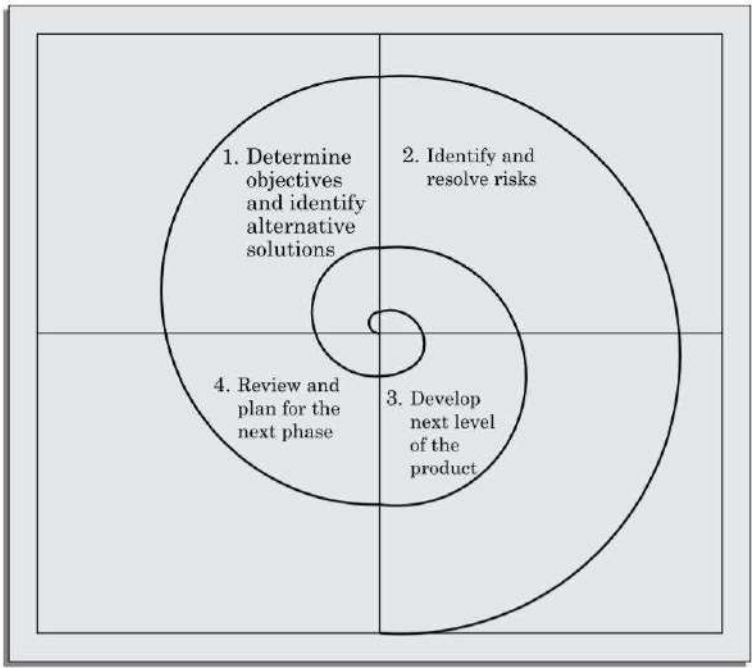


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module**–In this module, a guest is registered and the profile is maintained.If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input:guest details
   Output: unique profile, updated database

2. **Catering Module**–In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input:guest details, room number, token number
   Output:updated database

3. **Bill Module**–In this module, the totalbill is generated when the guest prepares to check out.

   Input:guest details, room number, token number, food items consumed
   Output:generated bill

Design Engineering

Data Flow Diagrams (DFDs)

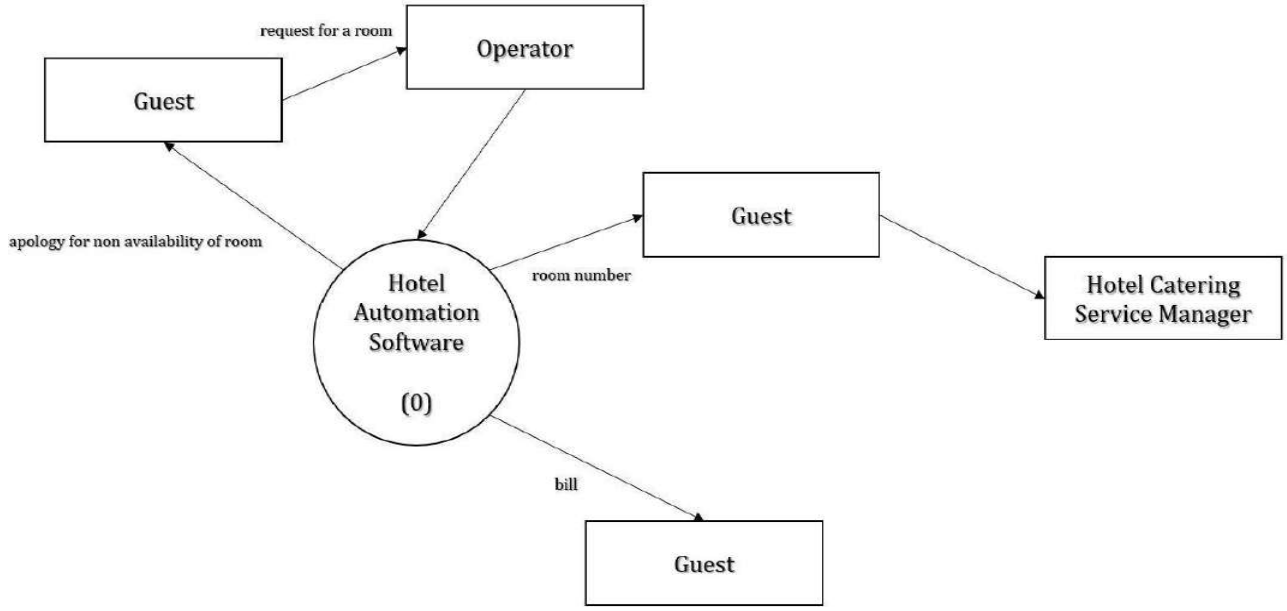


Fig – DFD level zero

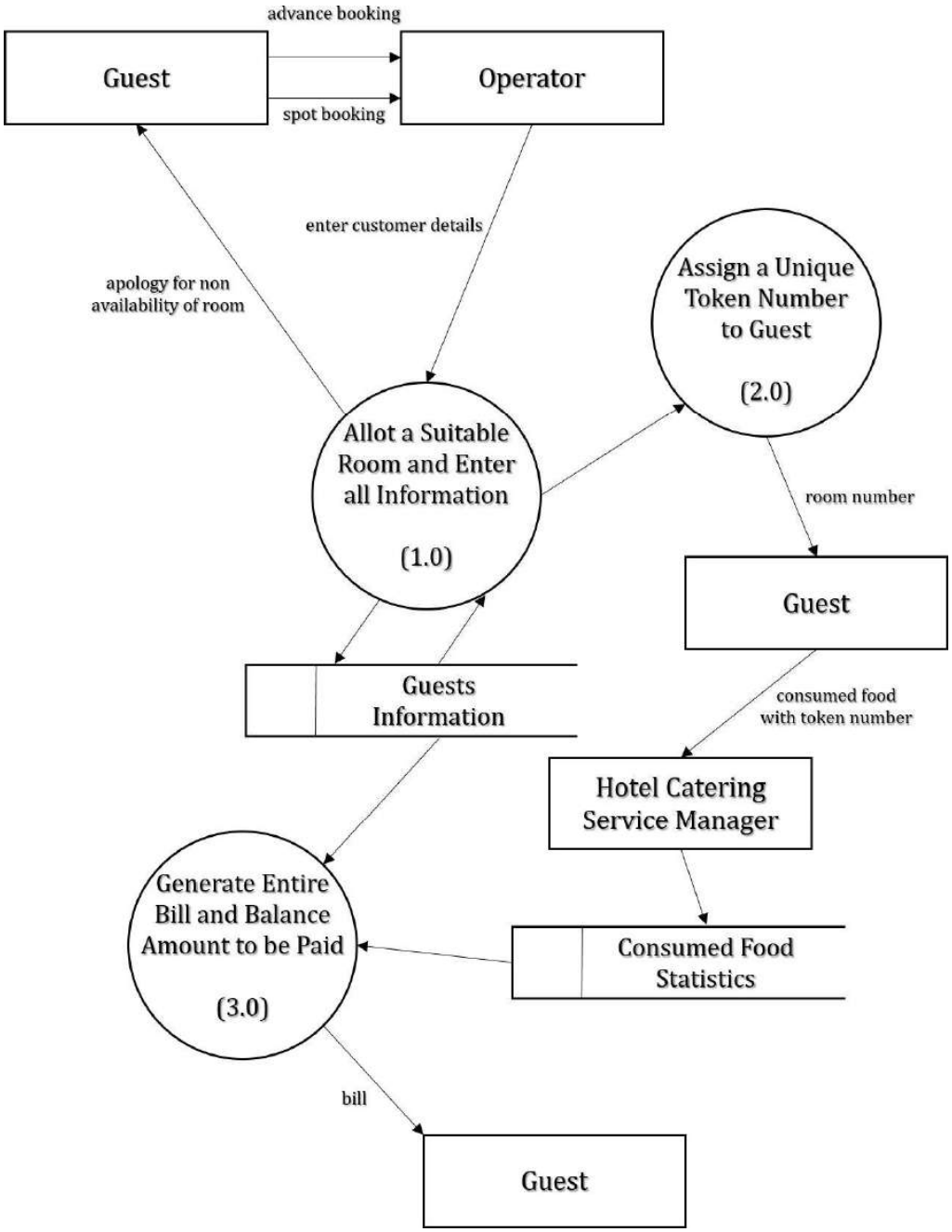
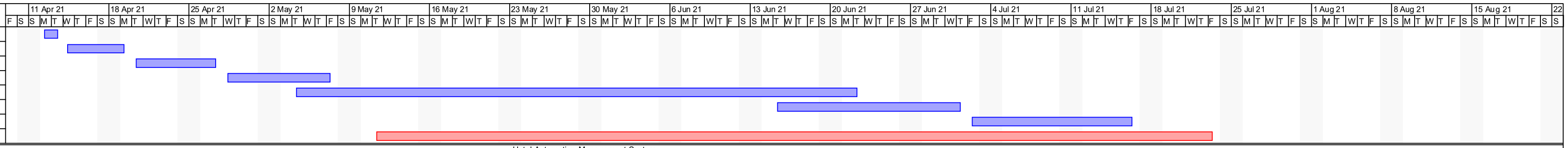


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Hotel Automation Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing–** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing–** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing–** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance–** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance–** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance–** Porting the software to work in a new environment.

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement –Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

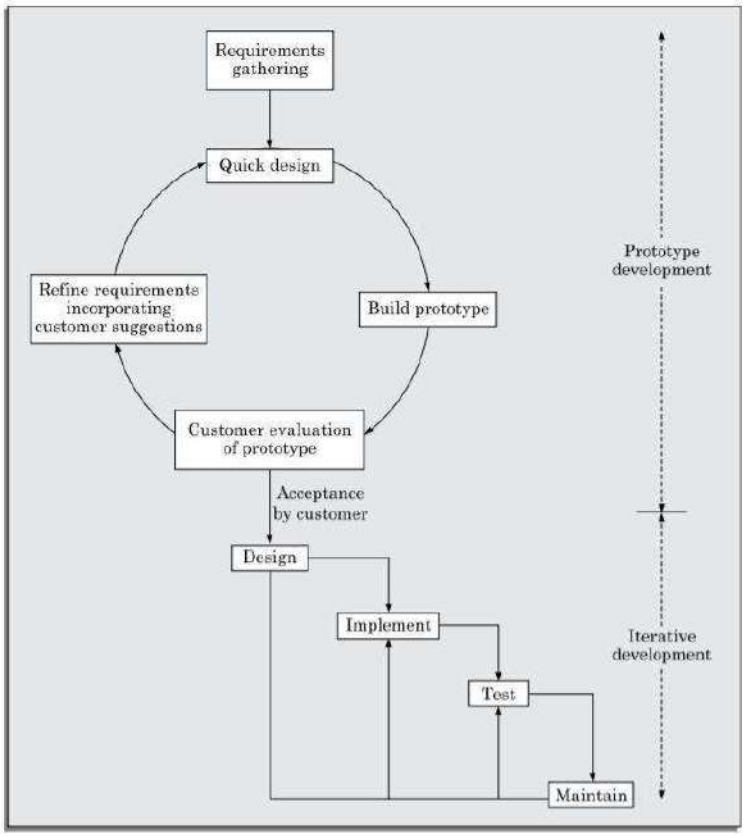


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly.Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module**–In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input:admin details
   Output: unique profile, granted level of access

2. **Complain Module**–In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output:filed FIR, updated database

3. **Crime Module**–In this module, the details of the crime are maintained and chargesheet is filed.

   Input:complaint details, FIR details
   Output:filed chargesheet

4. **Court Module**–In this module, the case is taken to the court and legal action is taken accordingly.

   Input:complaint details, FIR details, chargesheet details
   Output:legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

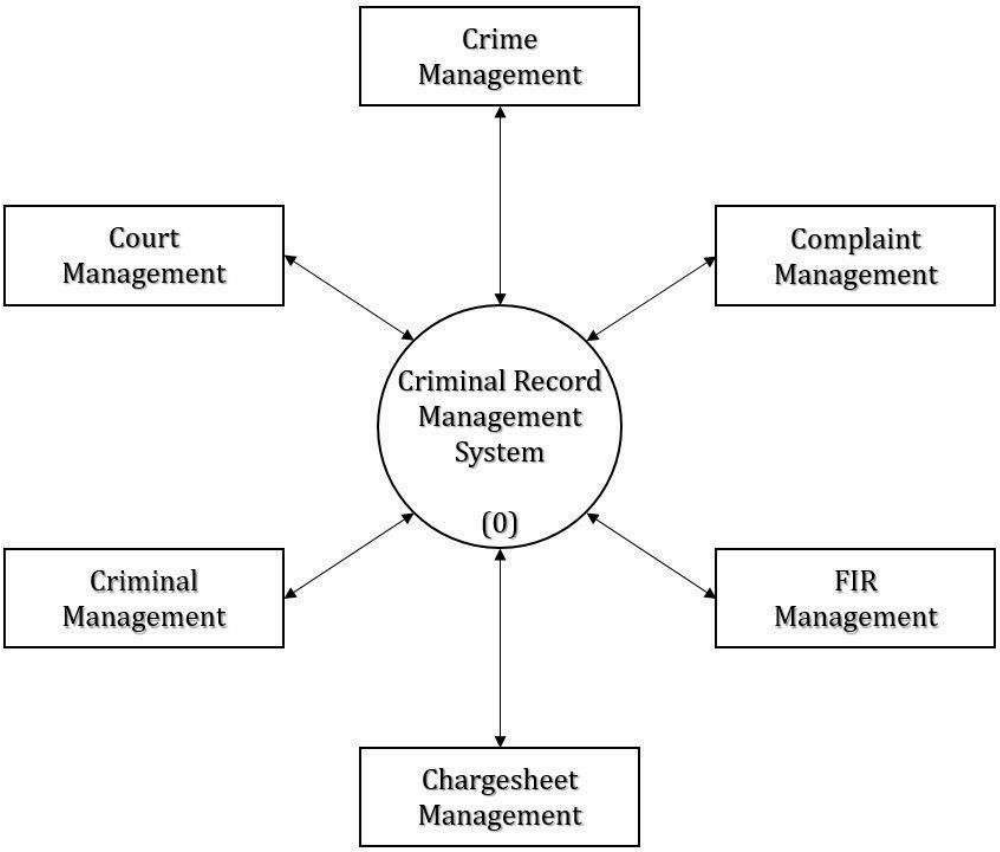


Fig – DFD level zero

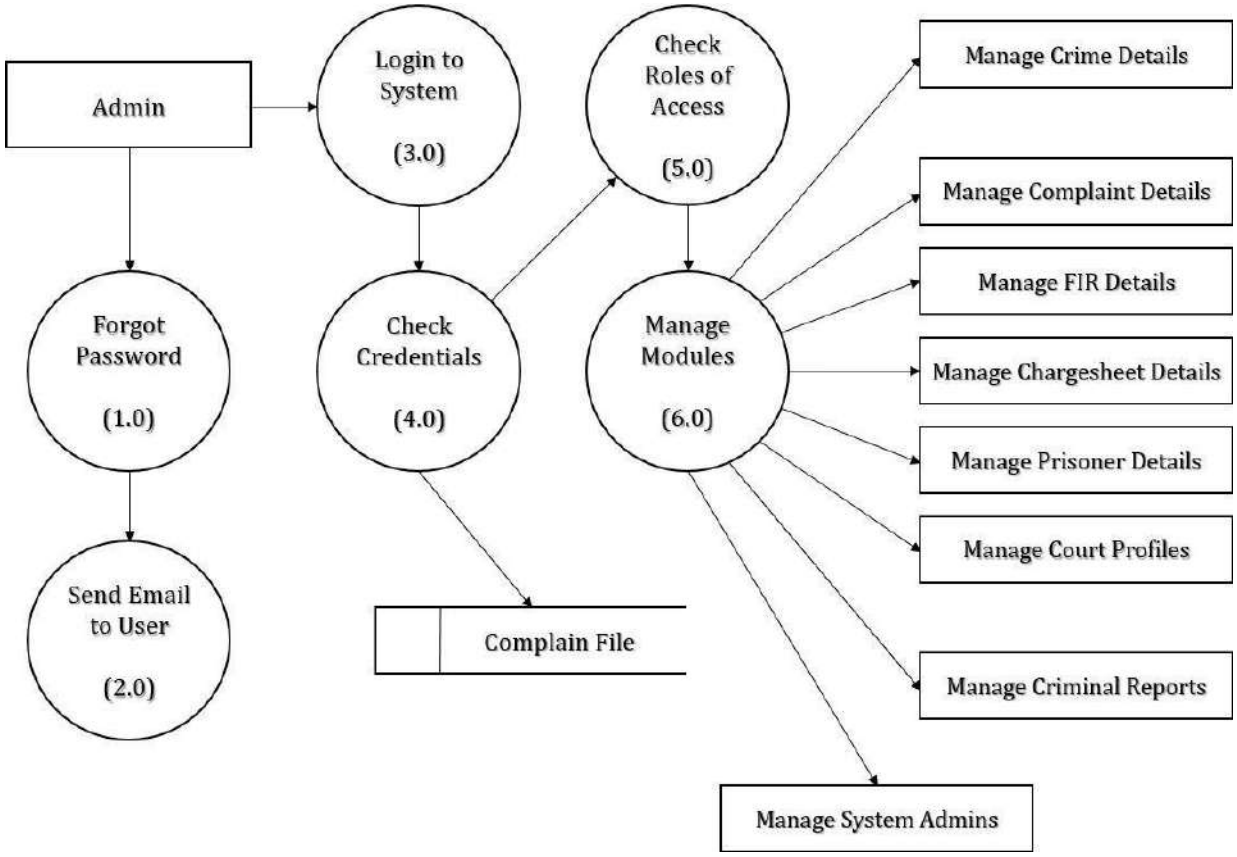
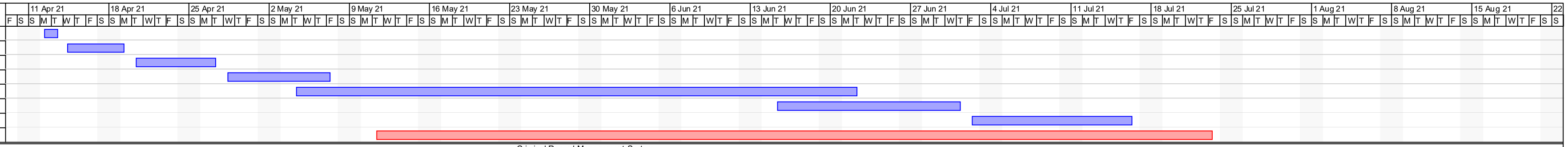


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Criminal Record Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement –Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model –Spiral development model

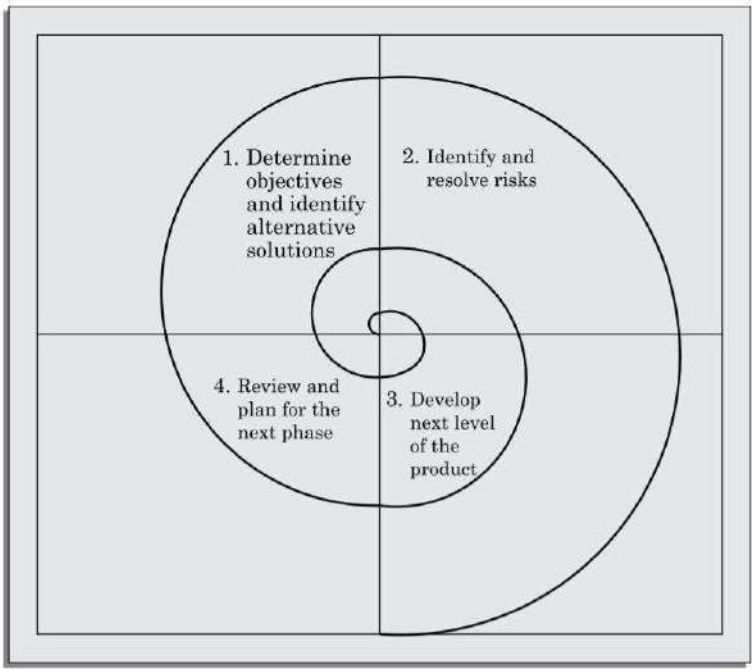


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module**–In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input:student details
   Output: unique profile

2. **Timetable Module**–In this module, the details of branch, course, class and subject ismaintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input:branchdetails, coursedetails, classdetails, subject details
   Output:updated database, created timetable

3. **Examination Module**–In this module,the examination takes place and the results are computed.

   Input:student details, timetable details
   Output:computed results, updated database

4. **Report Module**–In this module,timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input:guest details, room number, token number, food items consumed
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

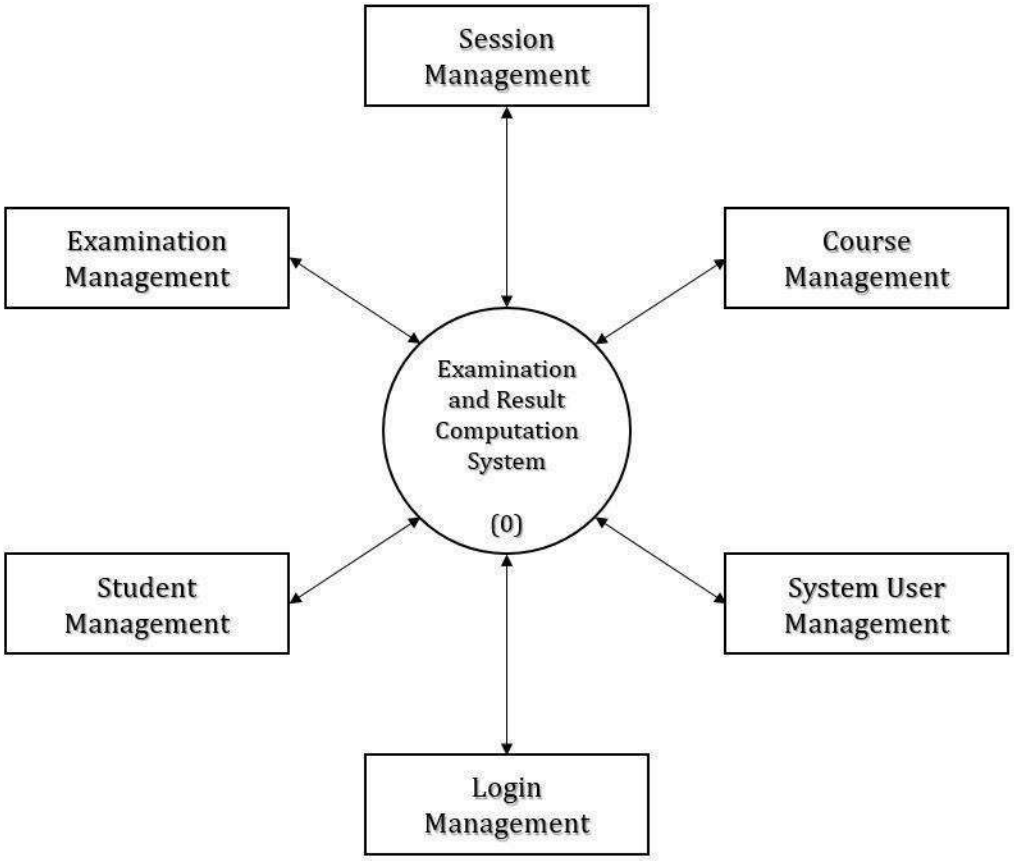


Fig – DFD level zero

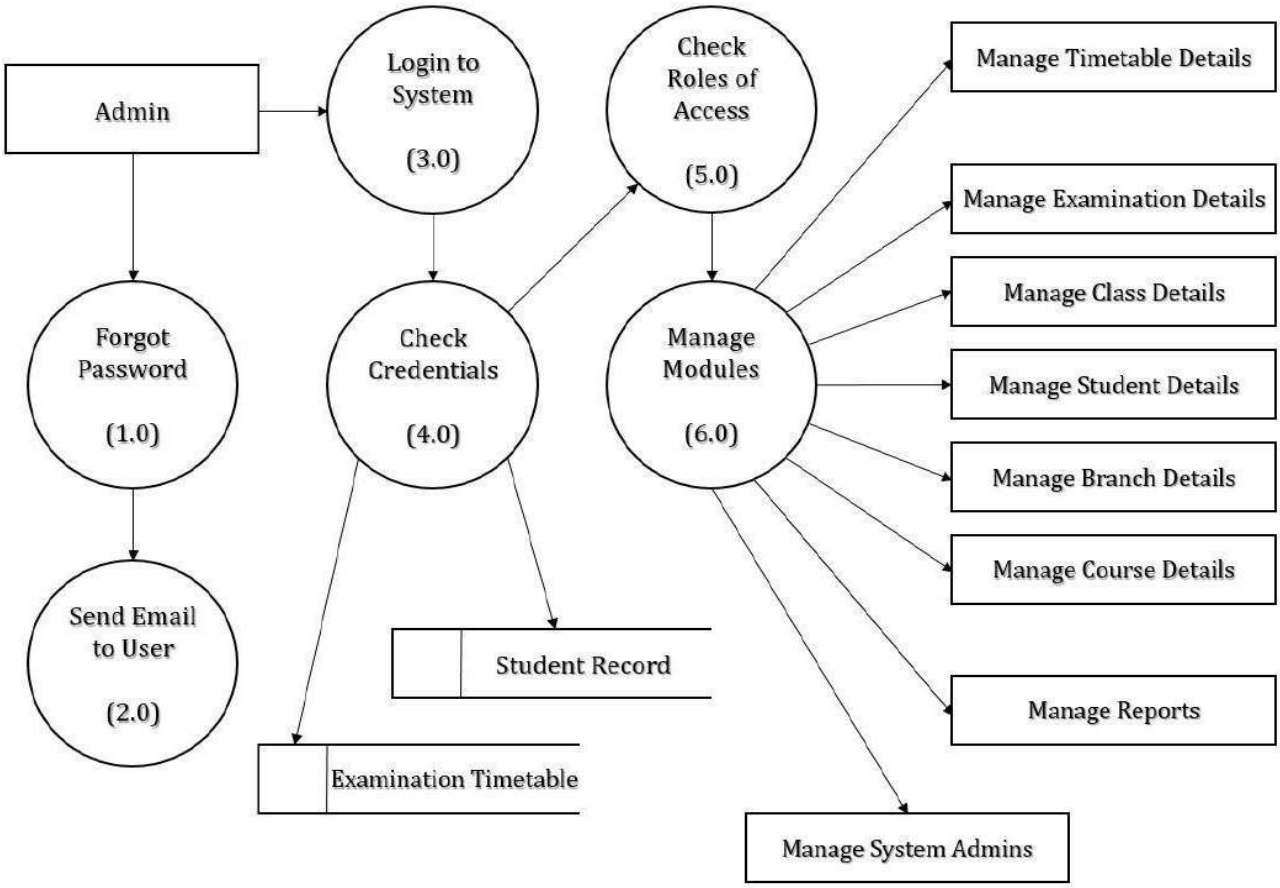
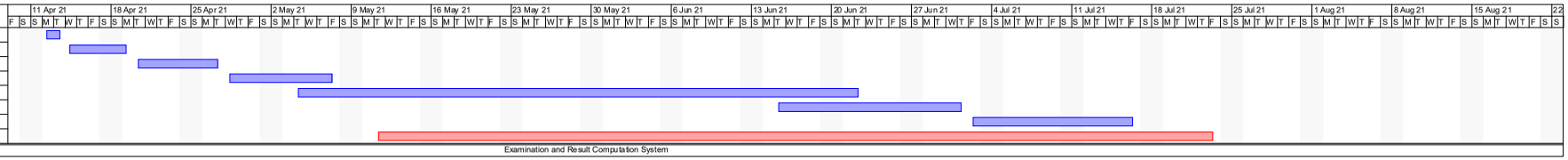


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

PARKING ALLOCATION SYATEM

Problem Statement –Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model –Spiral development model

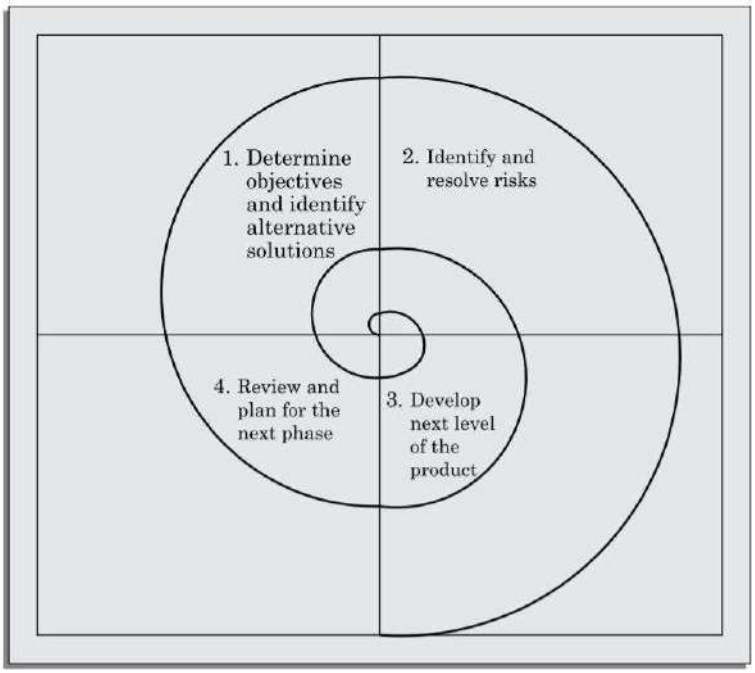


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module**–In this module, a customeris registered and the profile is maintained.

   Input:customerdetails
   Output: unique profile

2. **Vehicle Module**–In this module, the customer's vehicle is registered.

   Input:customer details, vehicle details
   Output:updated database

3. **Parking Module**–In this module,parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input:customer details, vehicle details, duration
   Output:updated database

4. **Report Module**–In this module,parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input:parking slot details, vehicle details, parking fee details, duration details
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

Fig – DFD level zero

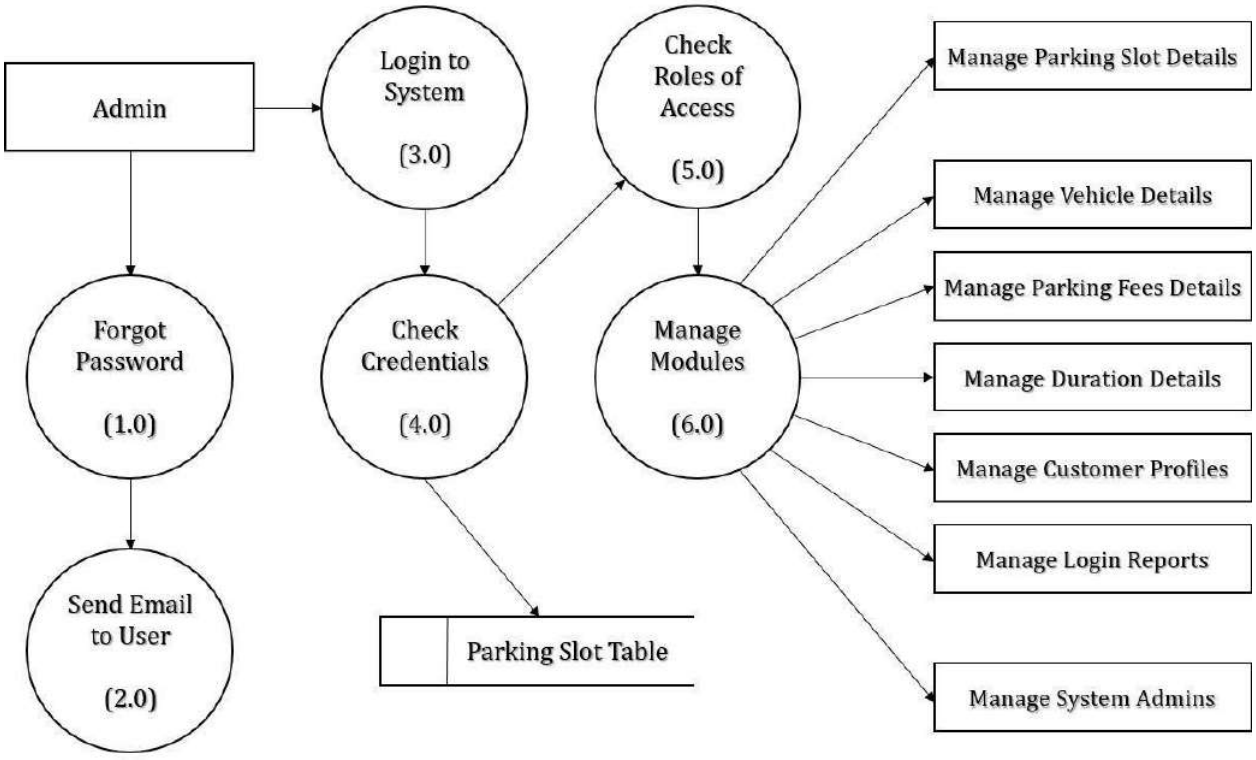
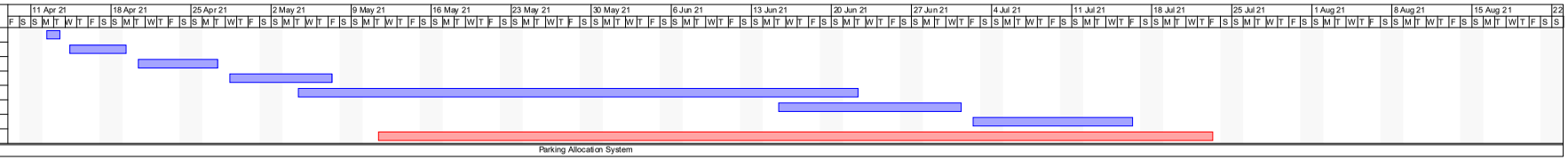


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement –It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model –Spiral development model

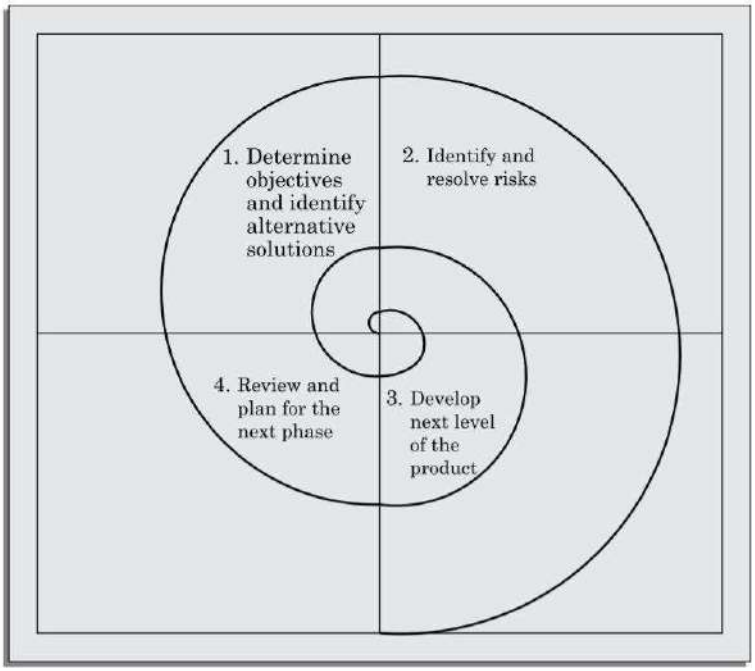


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module**–In this module, a customeris registered and the profile is maintained.

   Input:customerdetails
   Output: unique profile

2. **Stock Module**–In this module, the details of stocks are maintained.
   A database is maintained for keeping the stockinformation.

   Input:stockdetails
   Output:updated database

3. **Bill Module**–In this module,all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input:customer details, stock details
   Output:generated bills, updated database

4. **Report Module**–In this module,store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input:store details, product details, retail price details, bill details, billing counter details, customer details
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

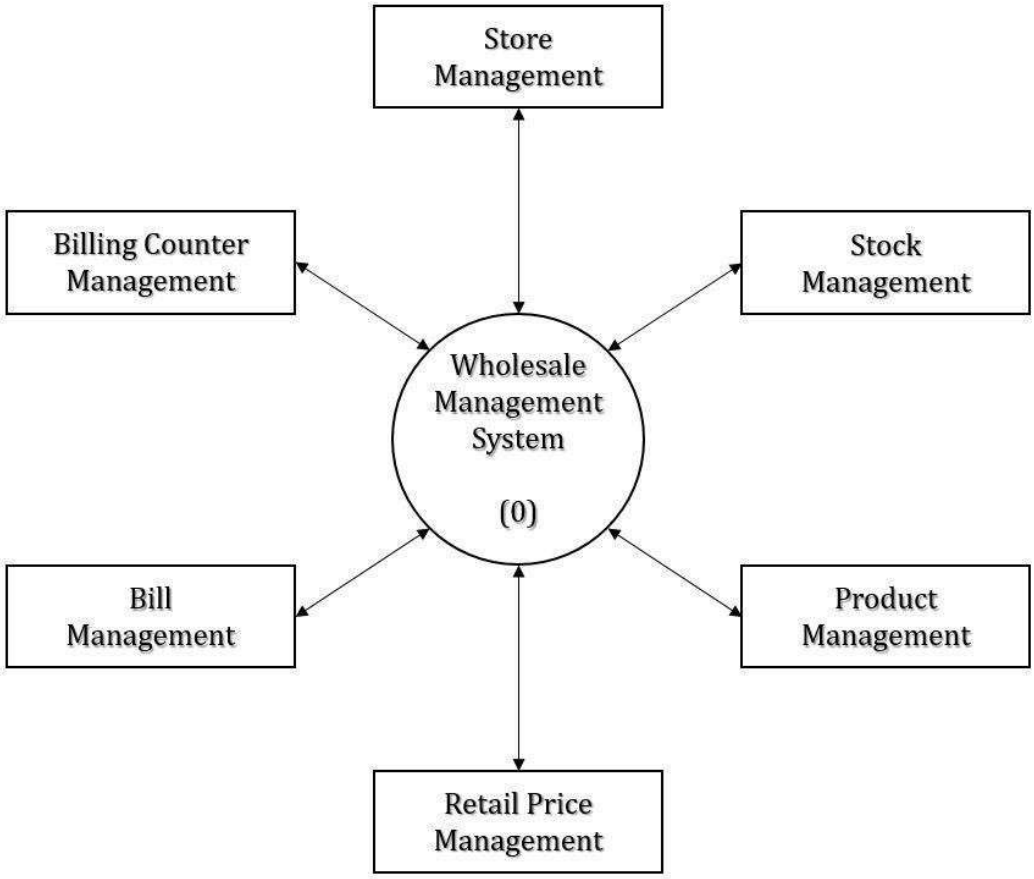


Fig – DFD level zero

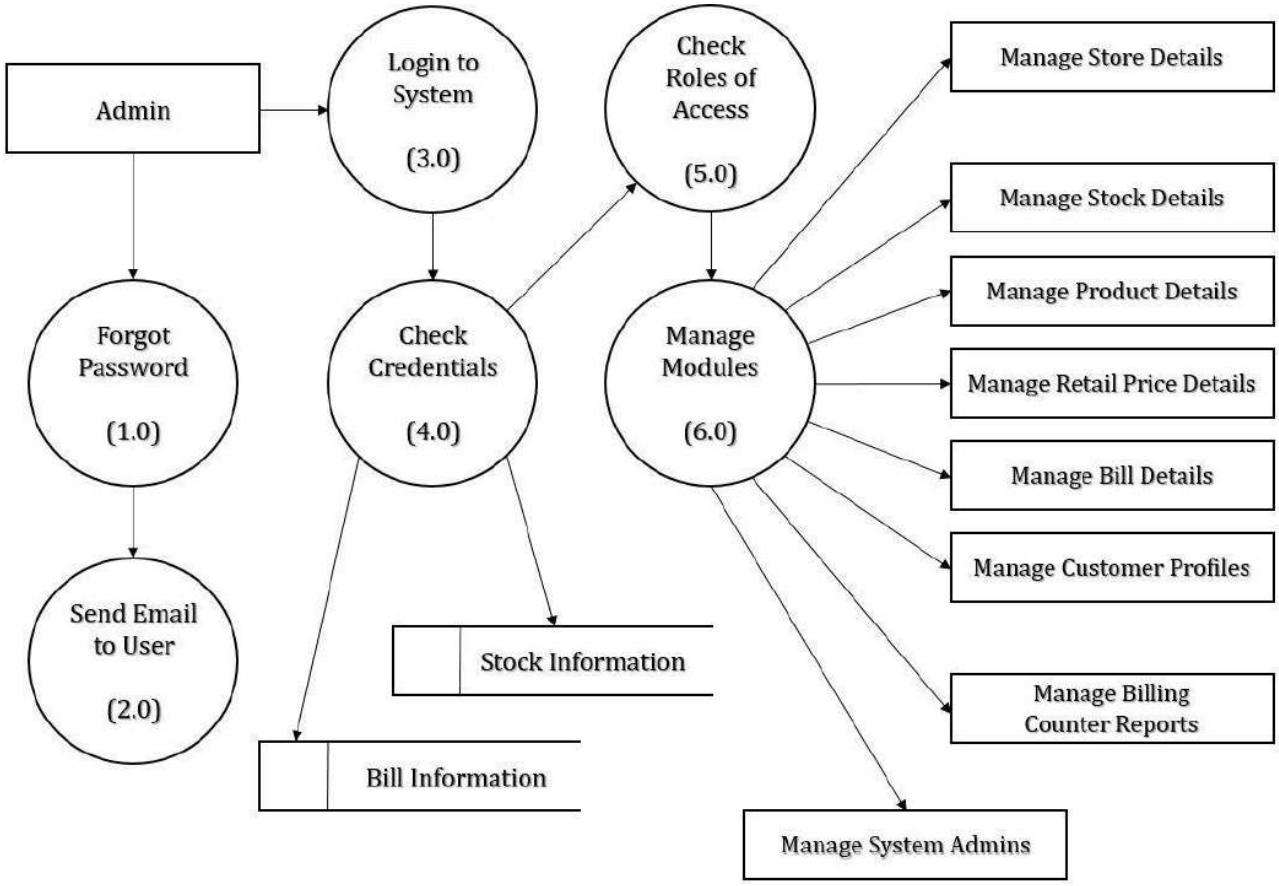
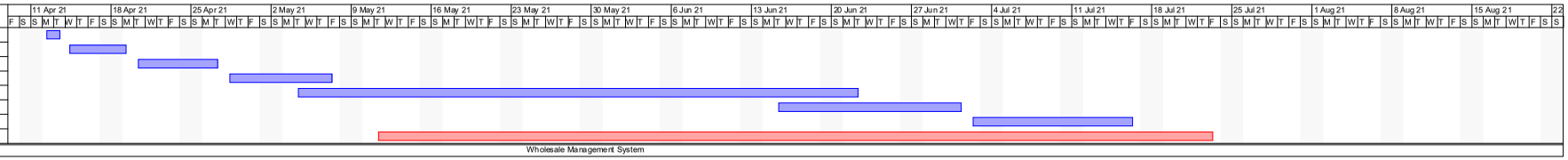


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

PROJECT ON RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Pratick Maiti.

College Roll No: 719

Examination Roll No.: 2021141300

Paper Code: CMSADSE3

Semester: V

Department: Computer Science

Supervisor: Prof. Manas Pal

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement – A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

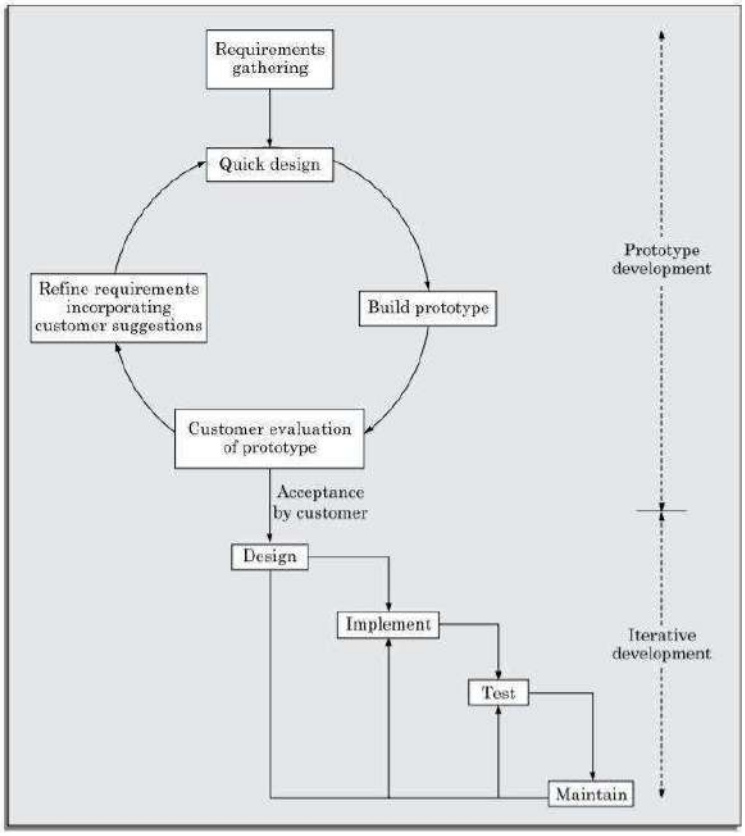


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: unique CN

2. **Purchase Module –** In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input: purchase made by the customer, CN
   Output: updated database

3. **Promotional Module –** In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output: prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

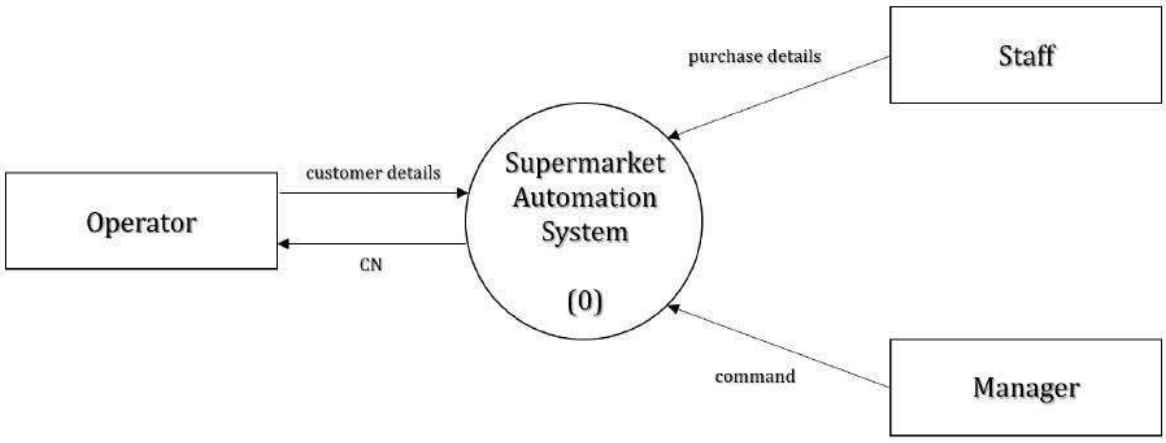


Fig – DFD level zero

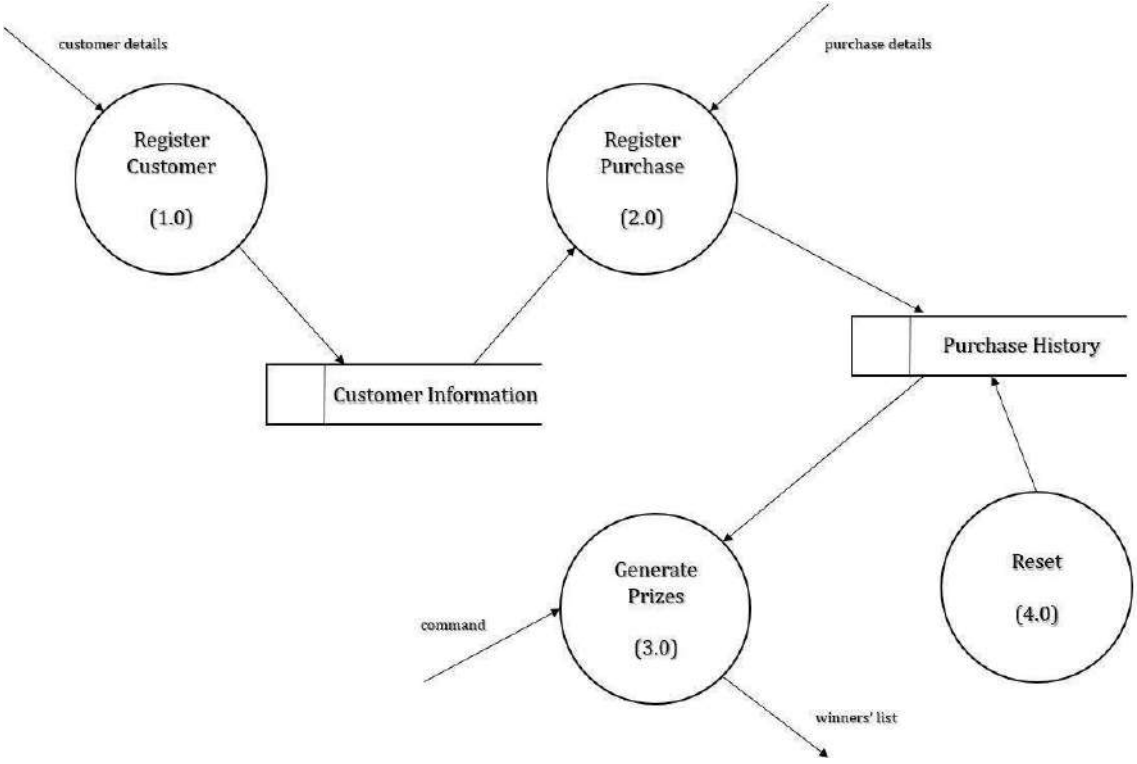


Fig – DFD level one

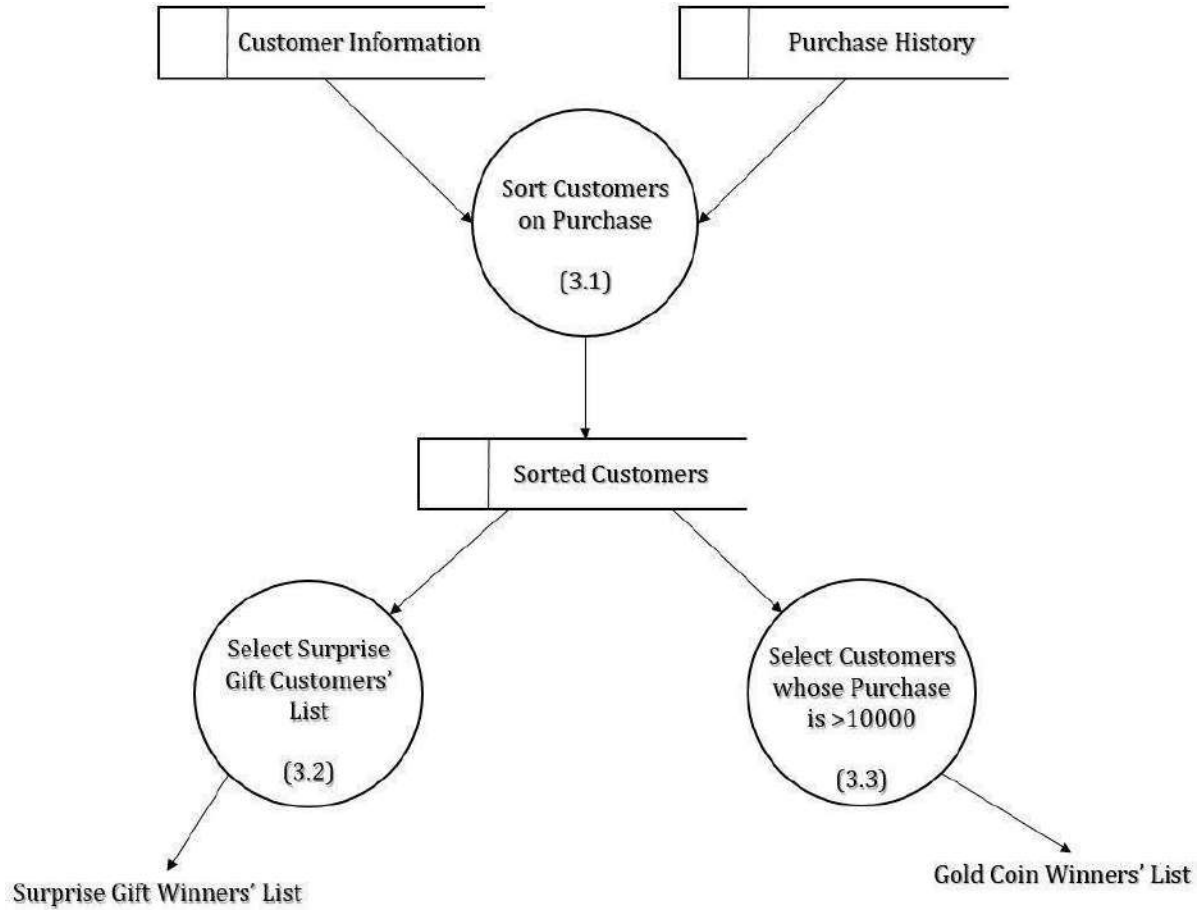
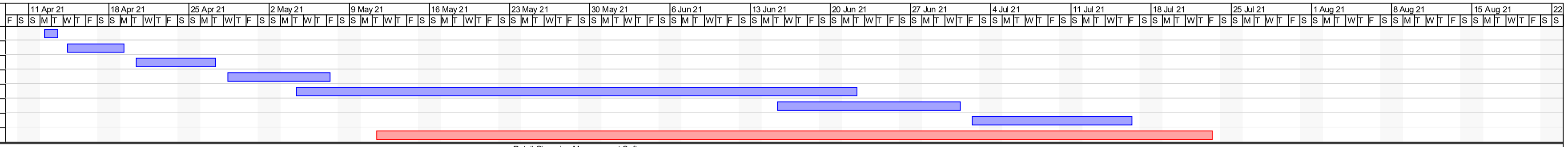


Fig – DFD level two of Process 3.0

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Retail Shopping Management Software

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement – It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model – Incremental model

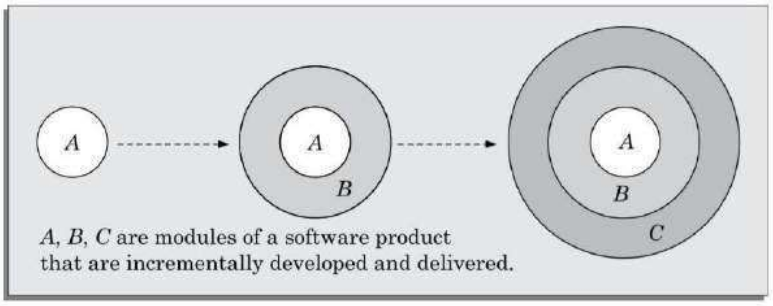


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module –** In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module –** In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module –** In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

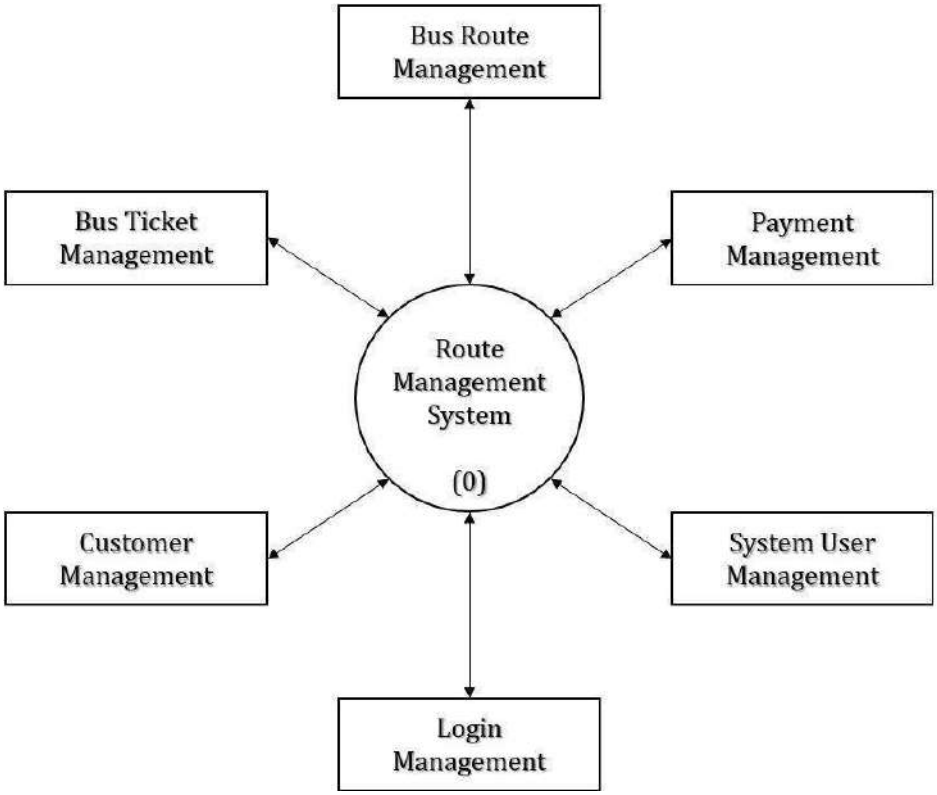


Fig – DFD level zero

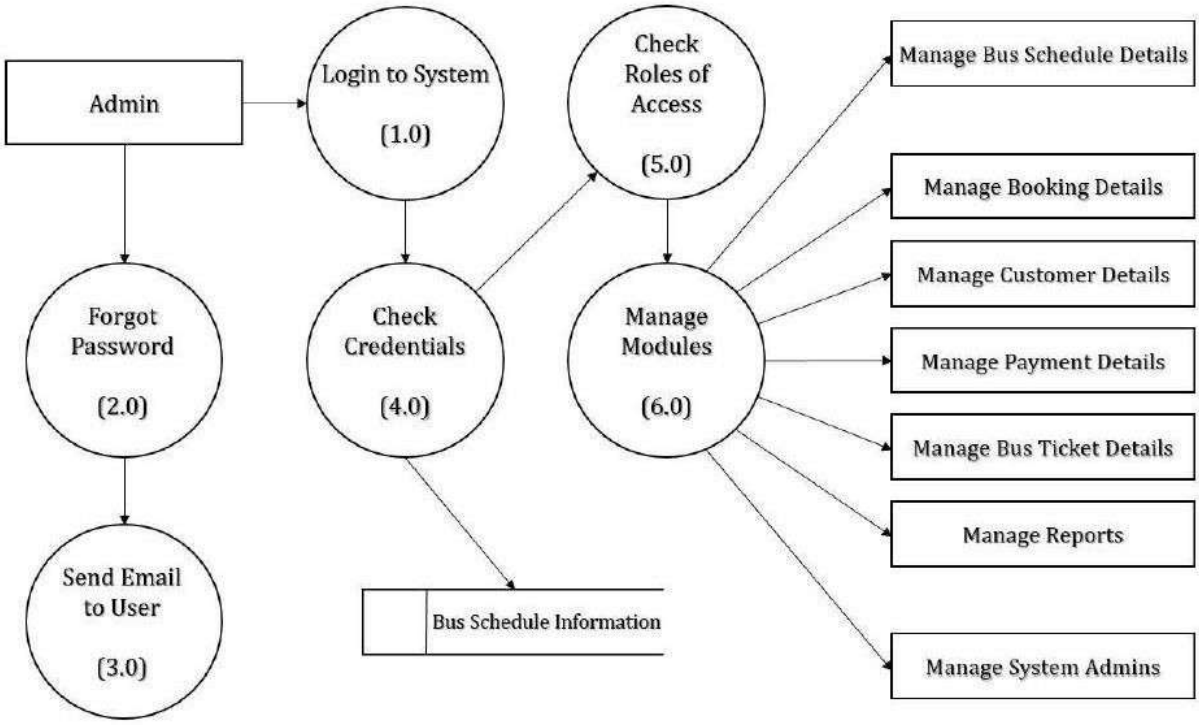
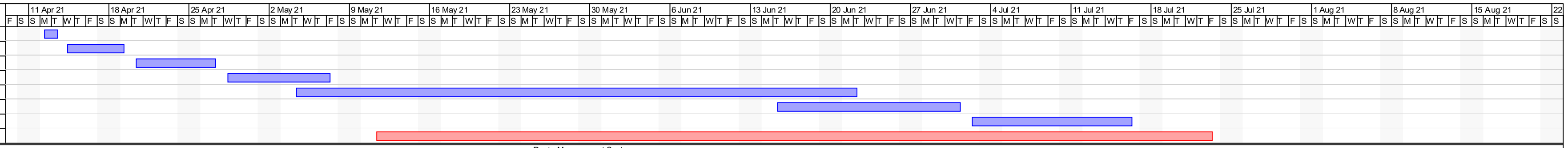


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement – Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model – Incremental development model

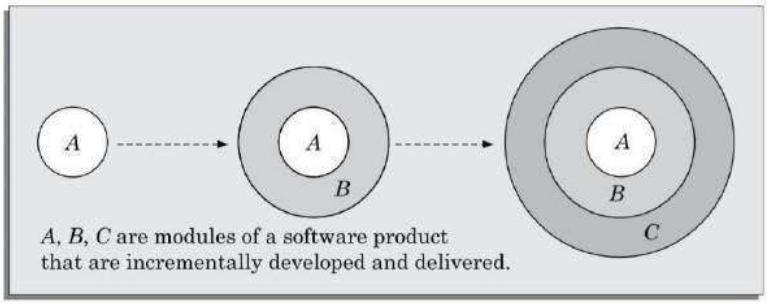


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module –** In this module, a patient is registered and the profile is maintained.

   Input: patient details
   Output: unique profile

2. **Doctor Module –** In this module, the details of doctors are maintained.

   Input: patient details
   Output: updated database

3. **Diagnosis Module –** In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input: patient details, doctor details, test details
   Output: diagnosed disease, updated database

4. **Report Module –** In this module, patient records, medicine and drug records, test records are processed and reports are generated.

   Input: patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

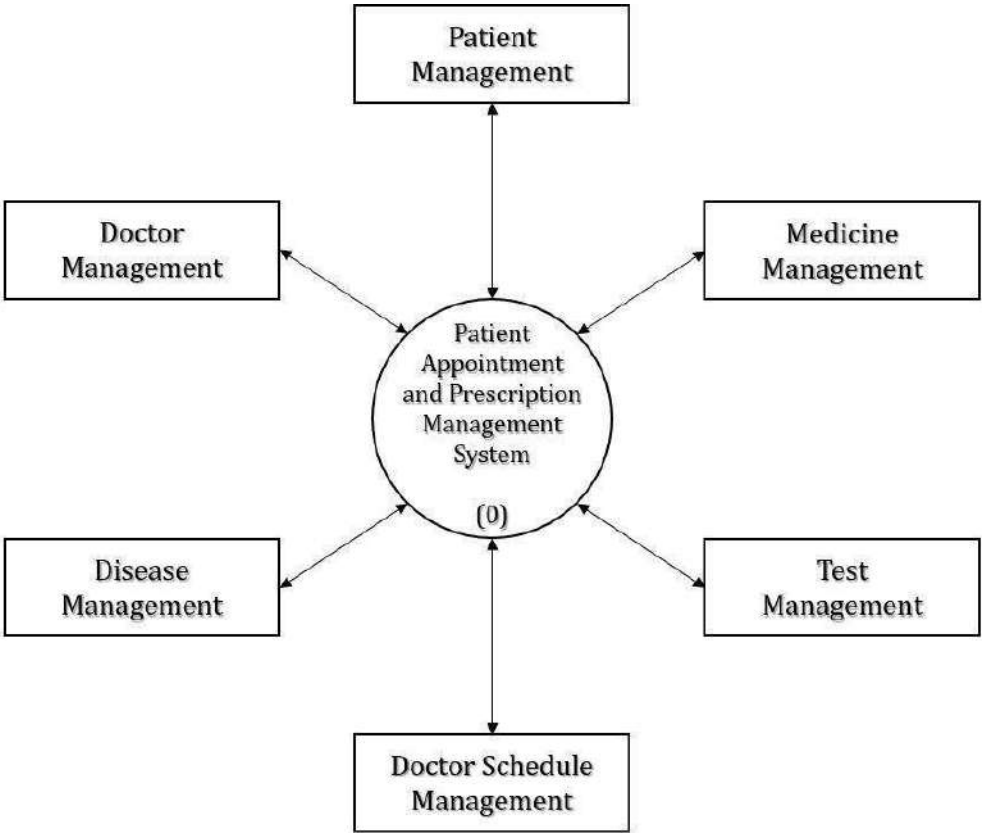


Fig – DFD level zero

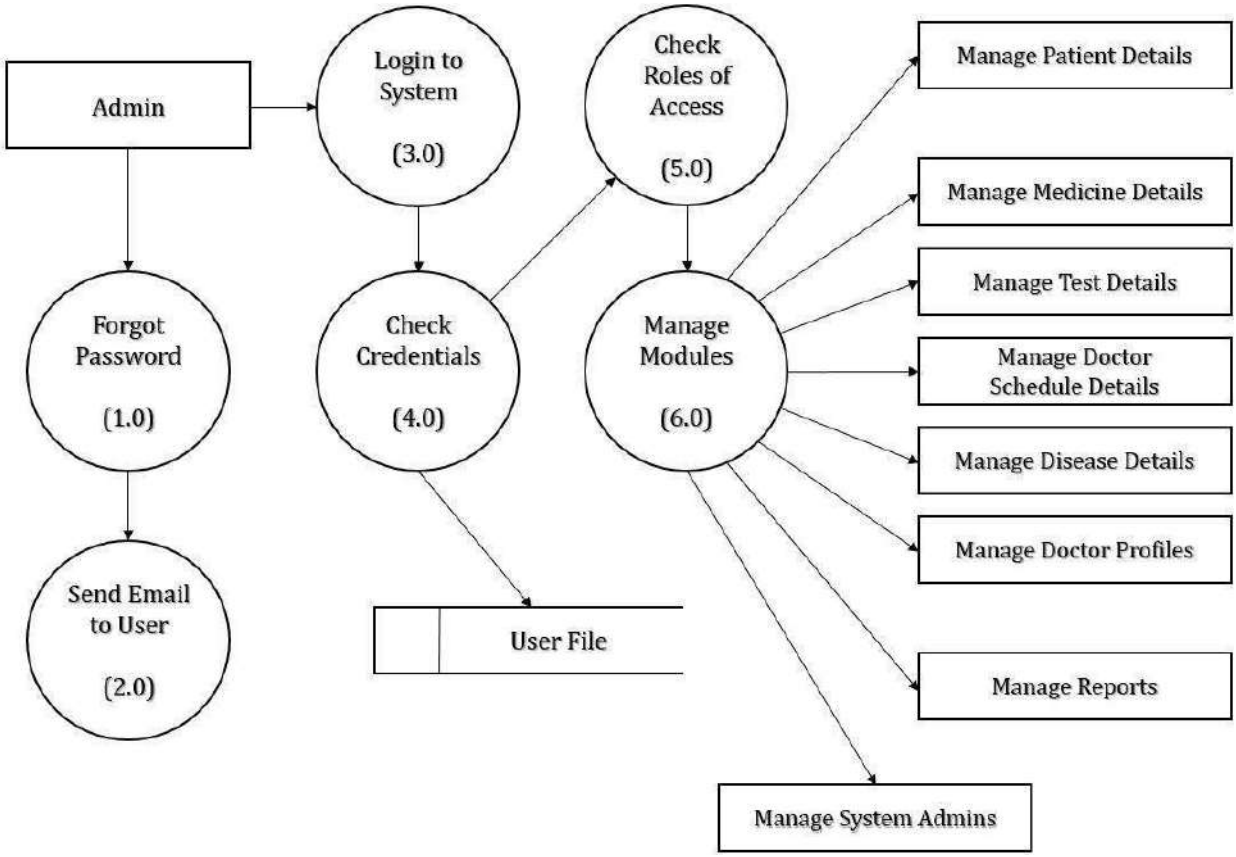
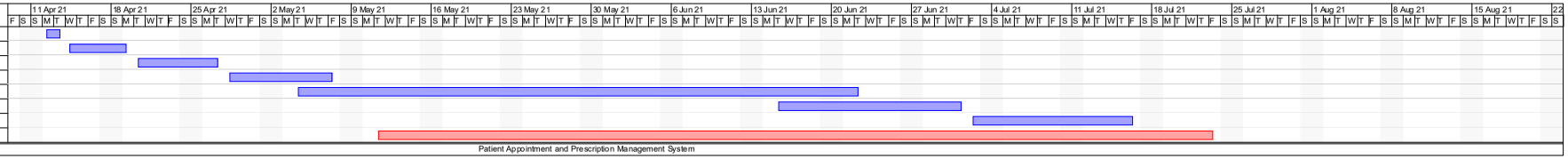


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement – Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model – Spiral development model

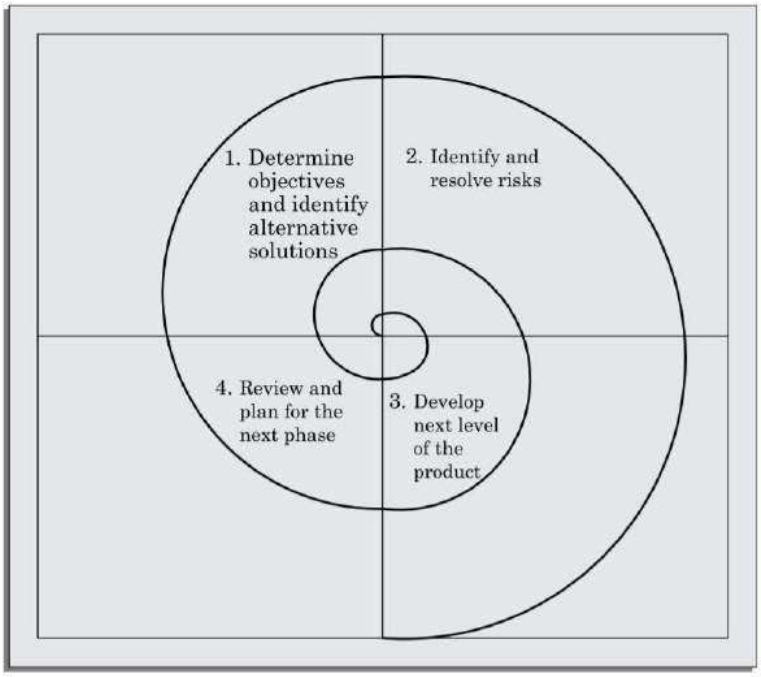


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module –** In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input: guest details
   Output: unique profile, updated database

2. **Catering Module –** In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input: guest details, room number, token number
   Output: updated database

3. **Bill Module –** In this module, the total bill is generated when the guest prepares to check out.

   Input: guest details, room number, token number, food items consumed
   Output: generated bill

Design Engineering

Data Flow Diagrams (DFDs)

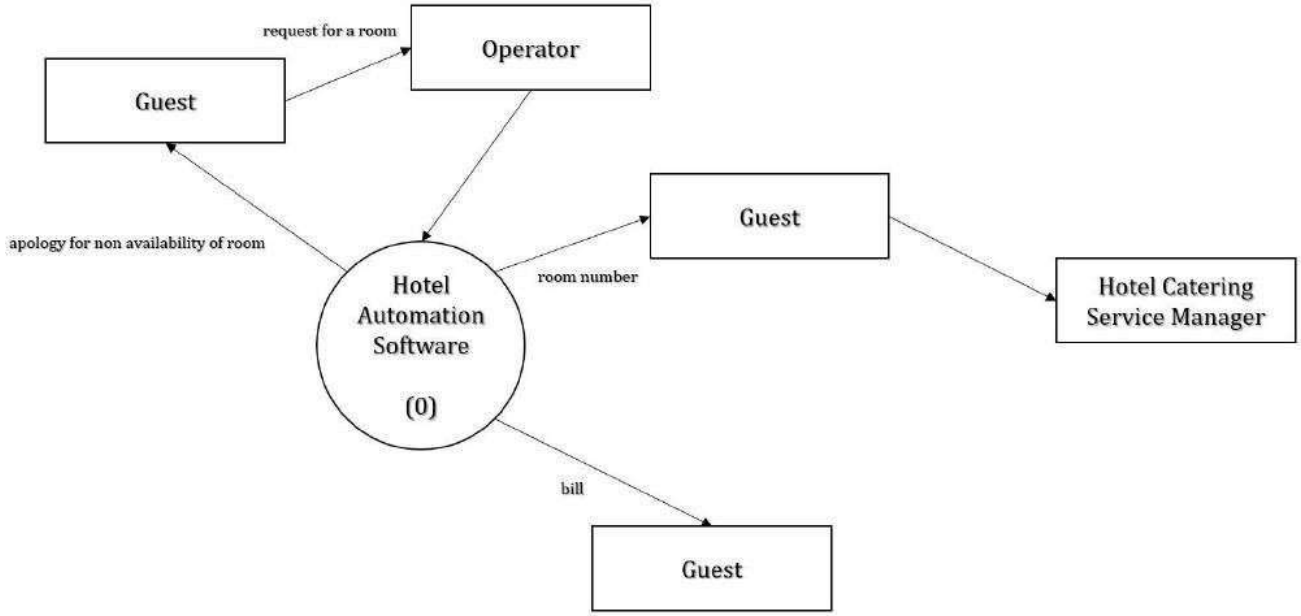


Fig – DFD level zero

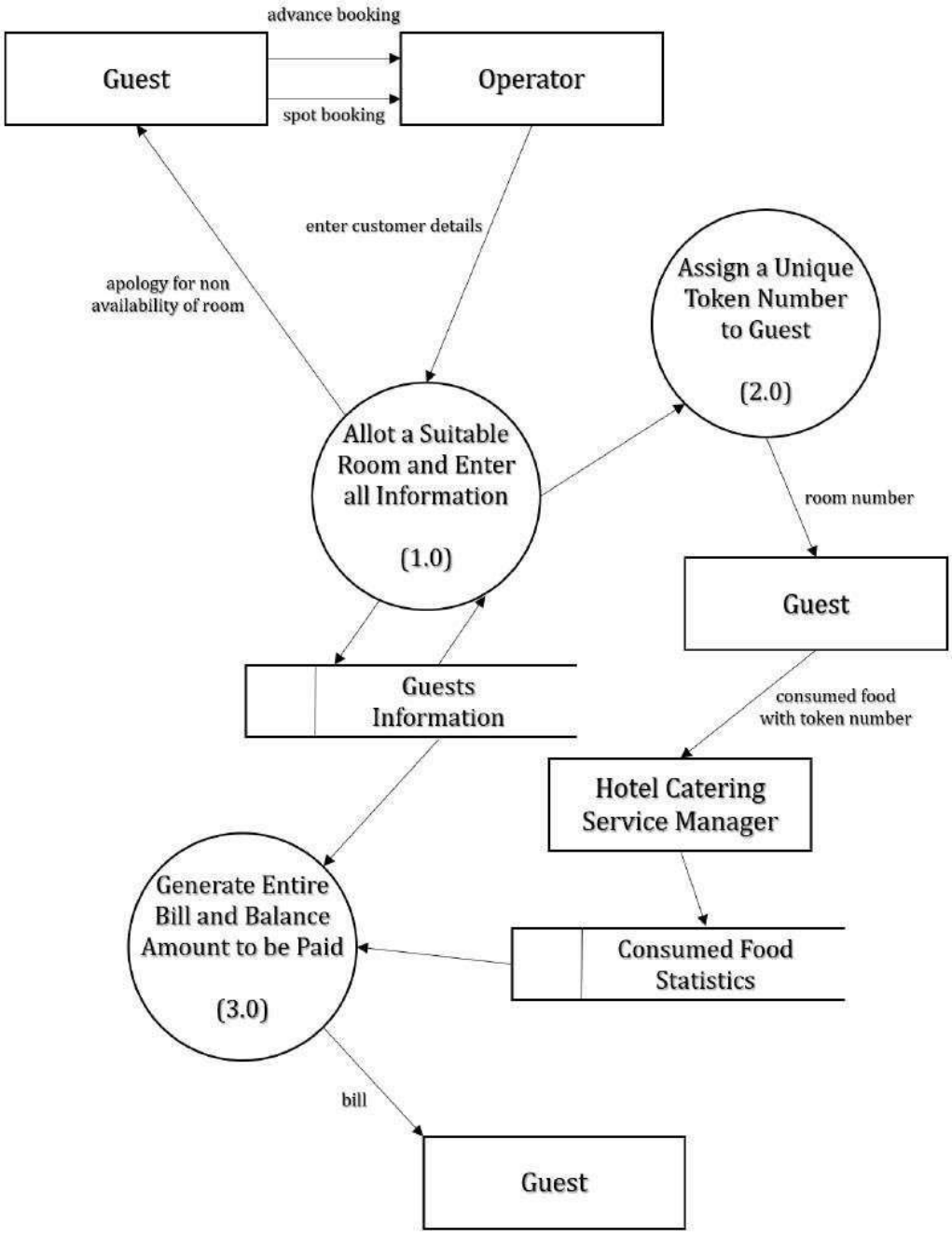
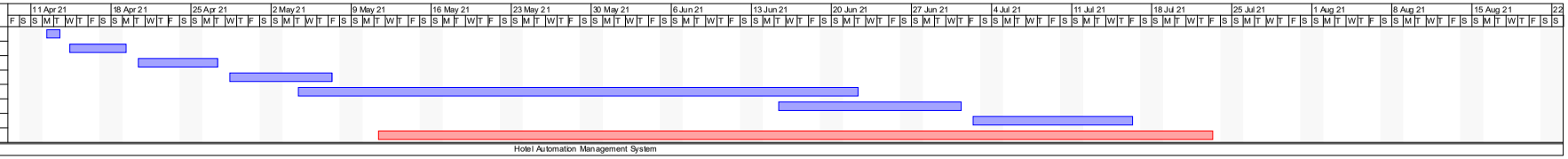


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Hotel Automation Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement – Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

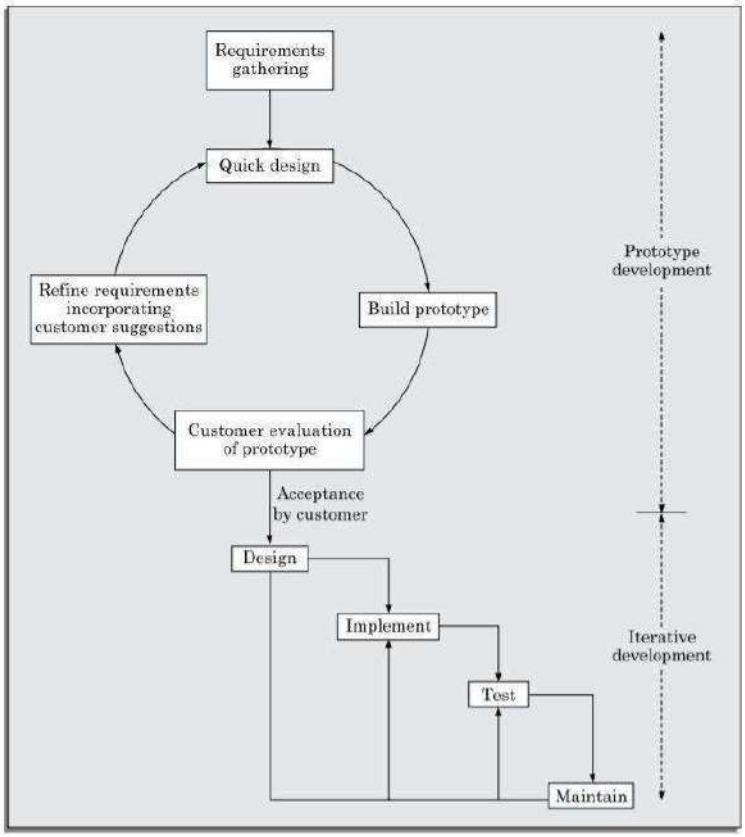


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module –** In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input: admin details
   Output: unique profile, granted level of access

2. **Complain Module –** In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output: filed FIR, updated database

3. **Crime Module –** In this module, the details of the crime are maintained and chargesheet is filed.

   Input: complaint details, FIR details
   Output: filed chargesheet

4. **Court Module –** In this module, the case is taken to the court and legal action is taken accordingly.

   Input: complaint details, FIR details, chargesheet details
   Output: legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

Fig – DFD level zero

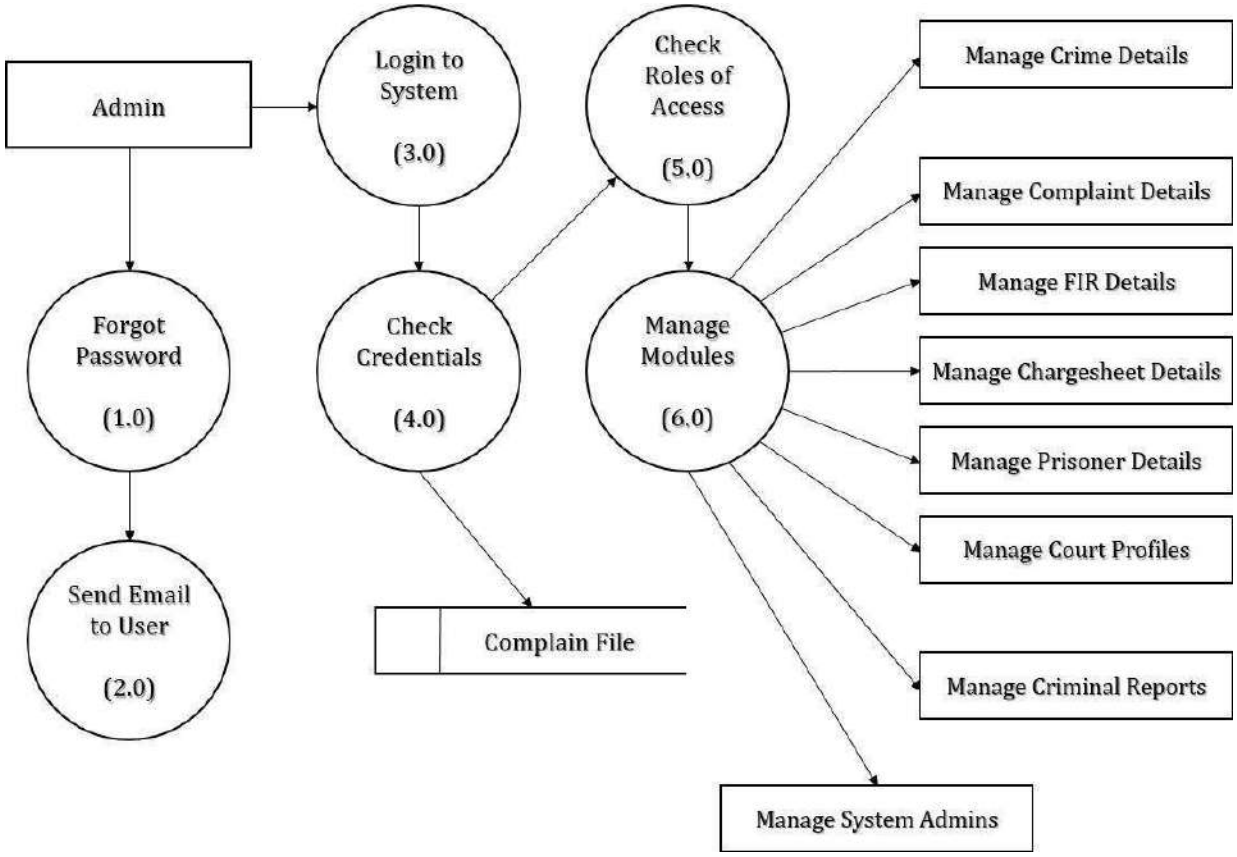
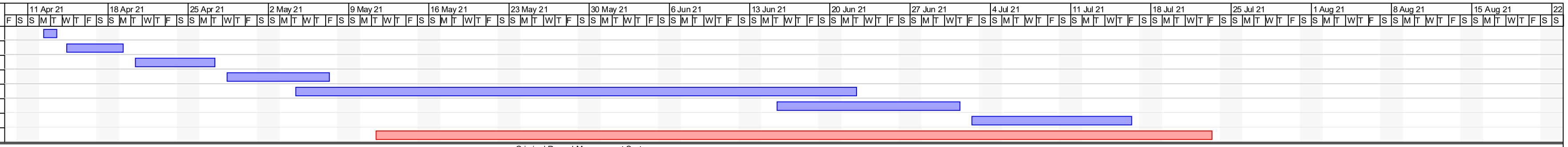


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Criminal Record Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement – Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model – Spiral development model

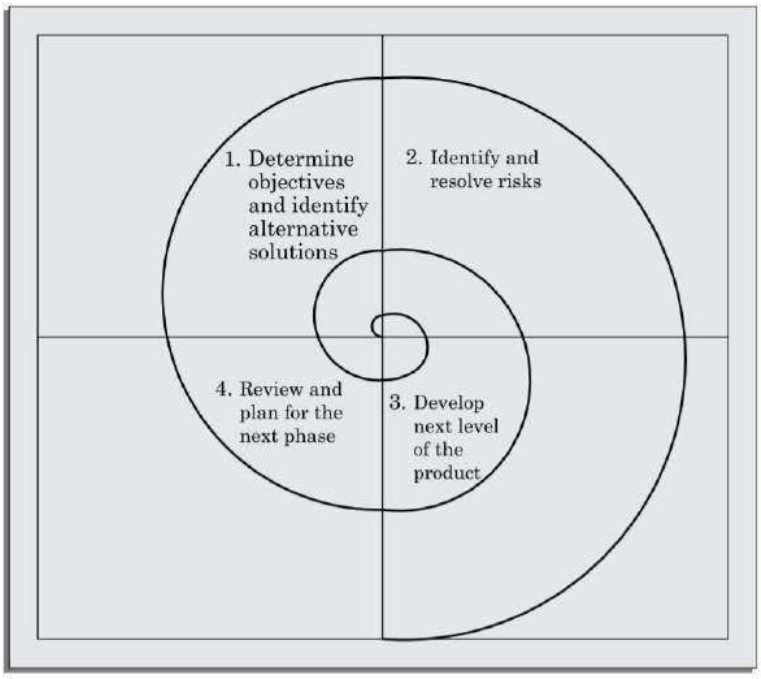


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module –** In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module –** In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module –** In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module –** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

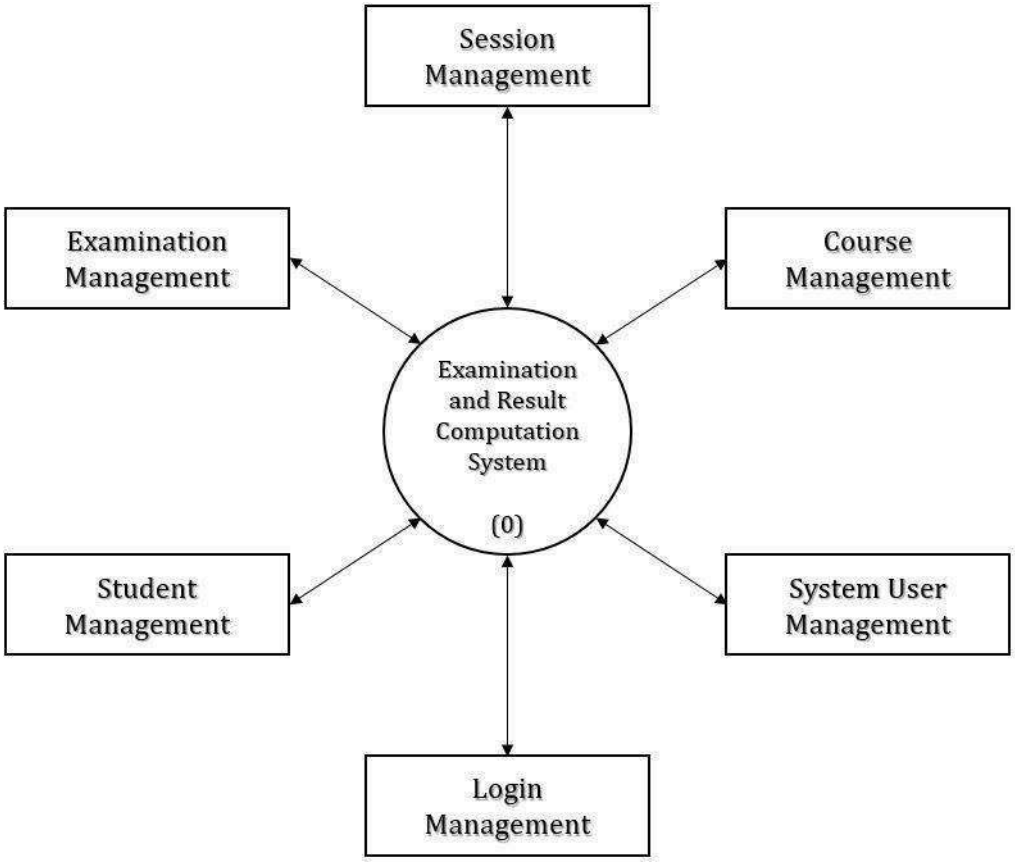


Fig – DFD level zero

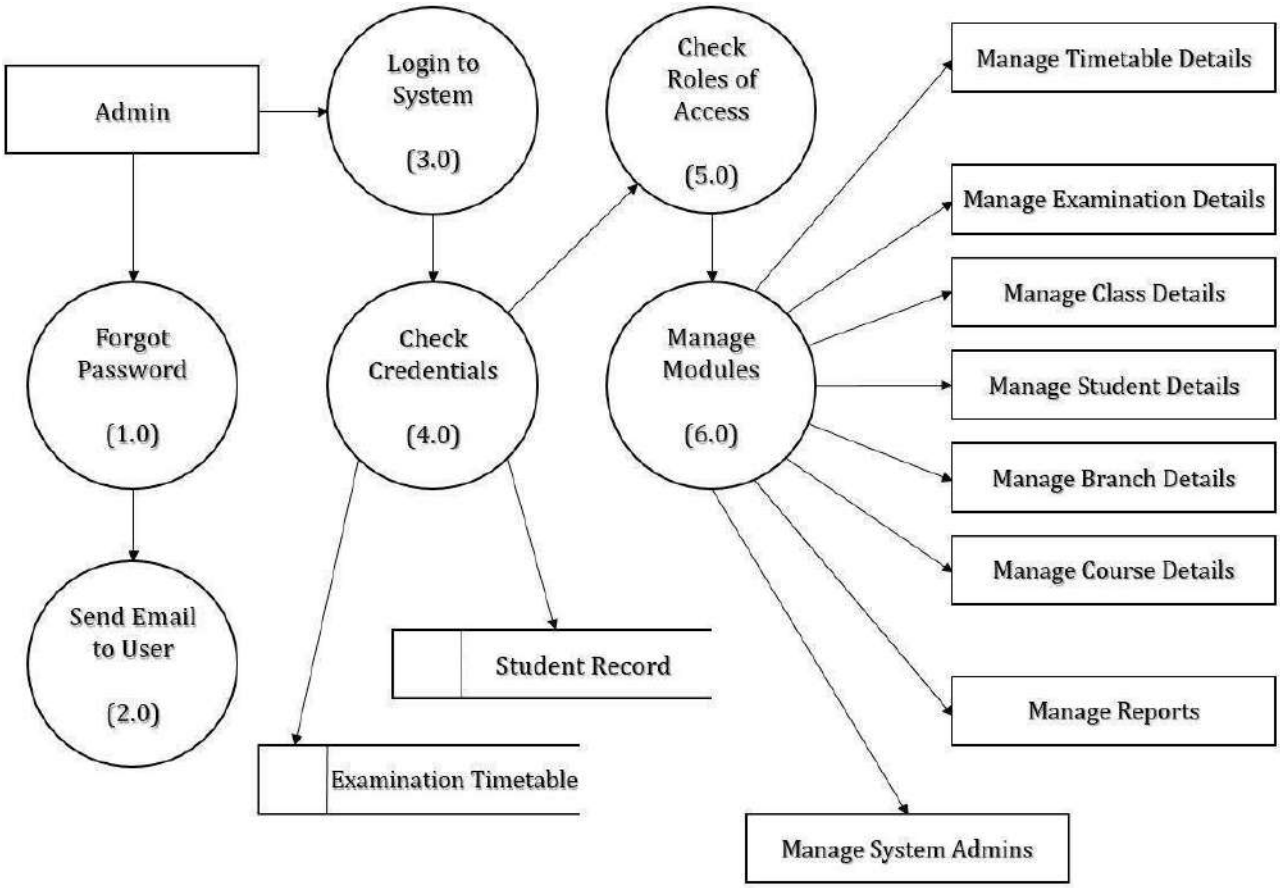
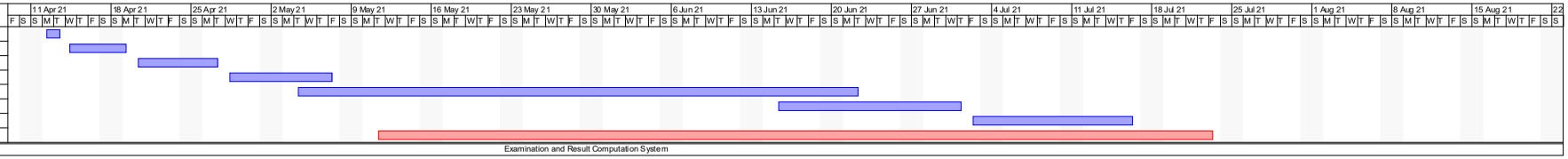


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PARKING ALLOCATION SYATEM

Problem Statement – Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model – Spiral development model

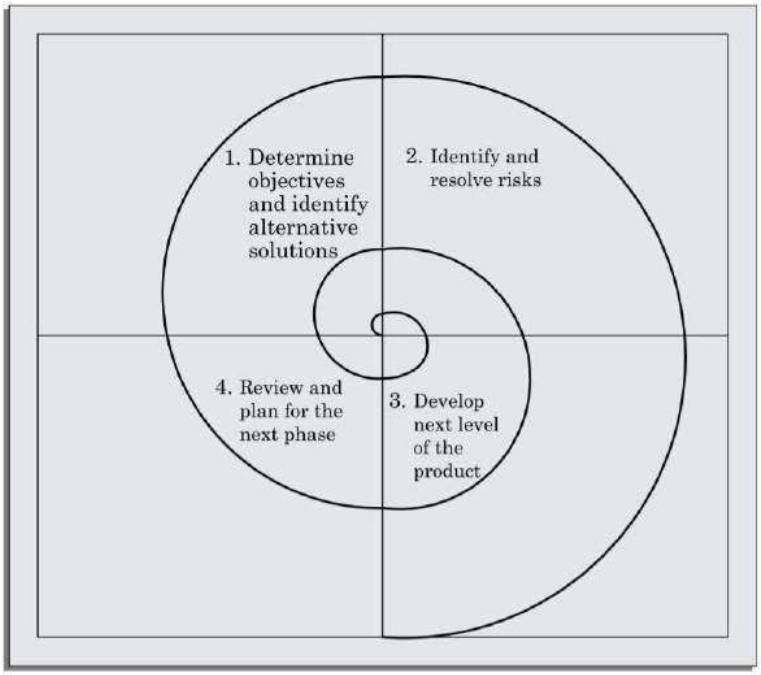


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module –** In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module –** In this module, parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module –** In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

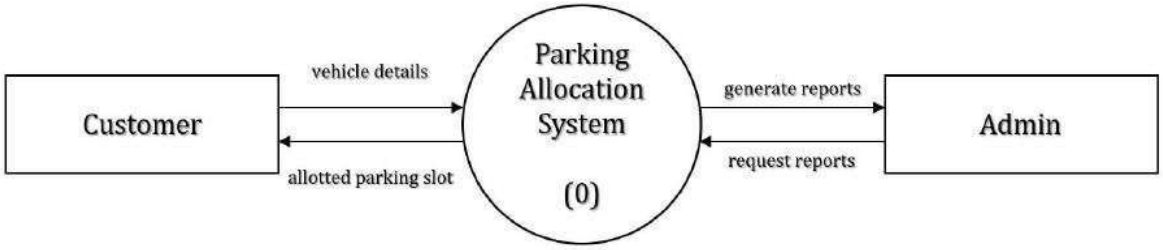


Fig – DFD level zero

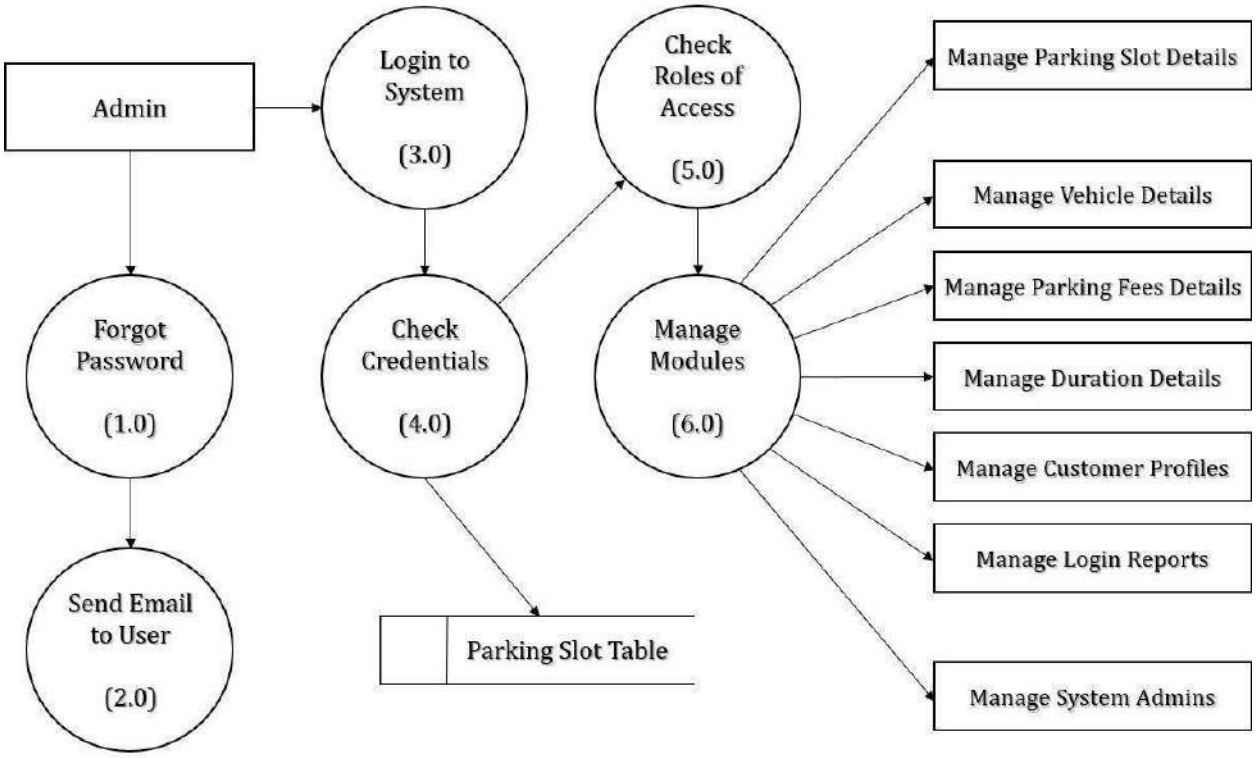
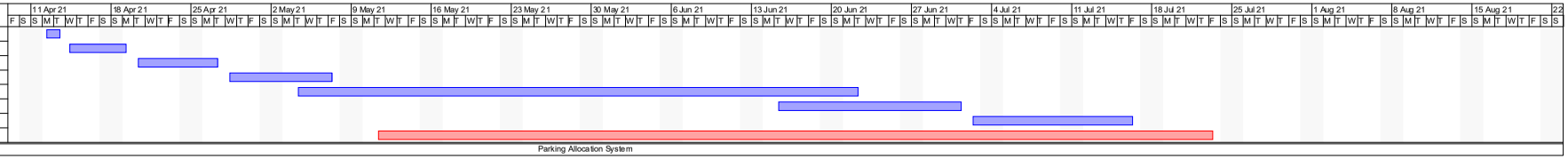


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Parking Allocation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

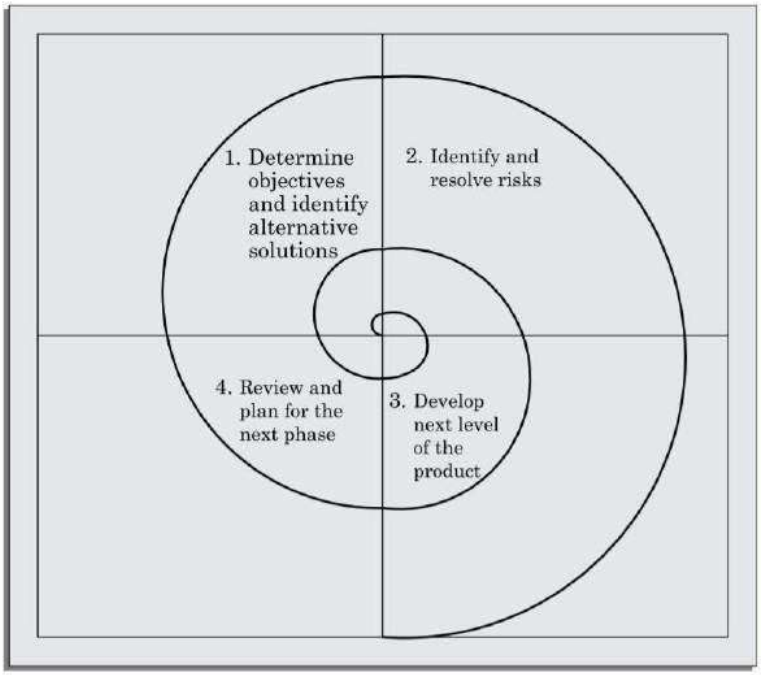


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

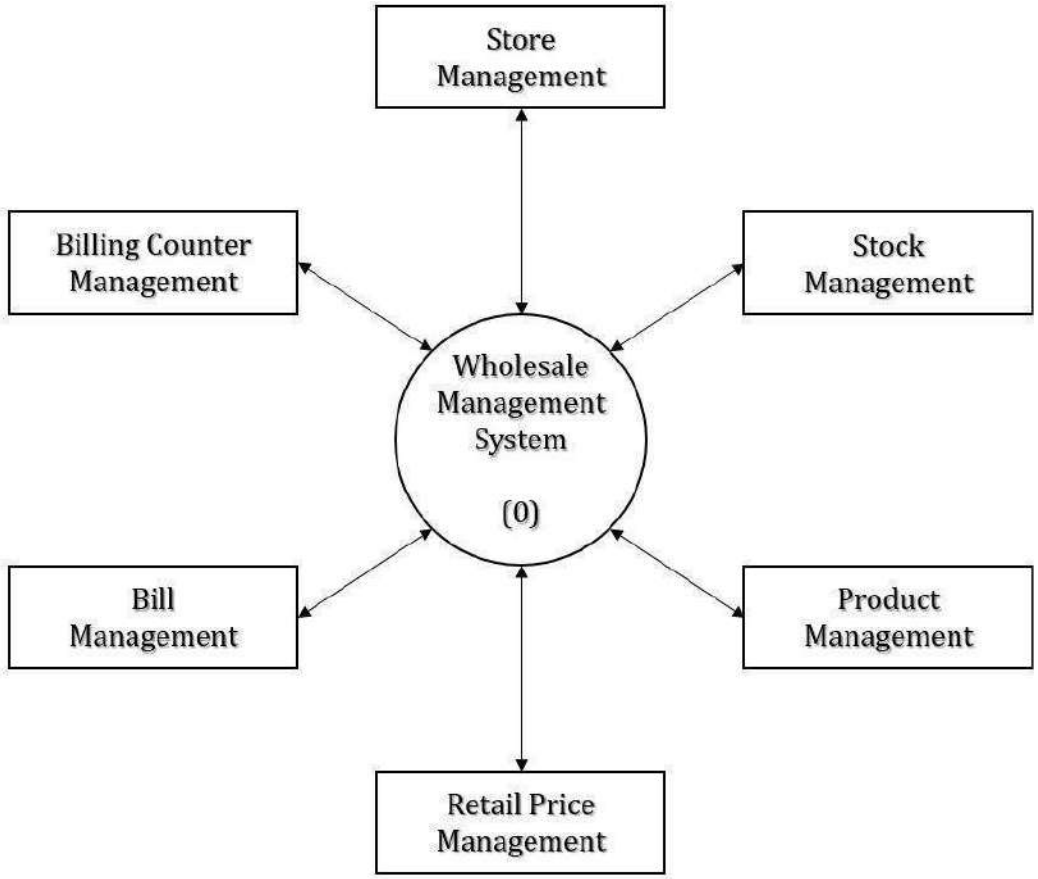


Fig – DFD level zero

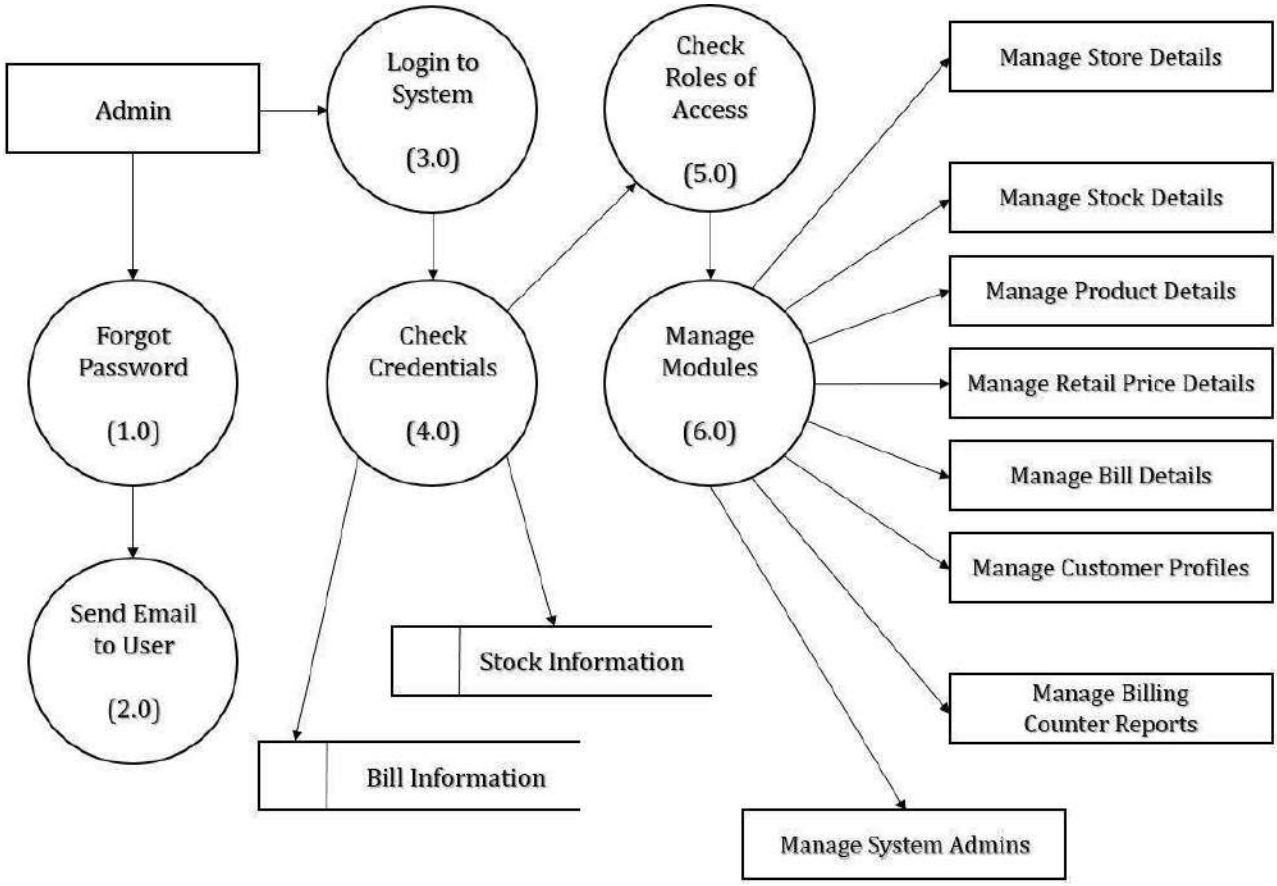
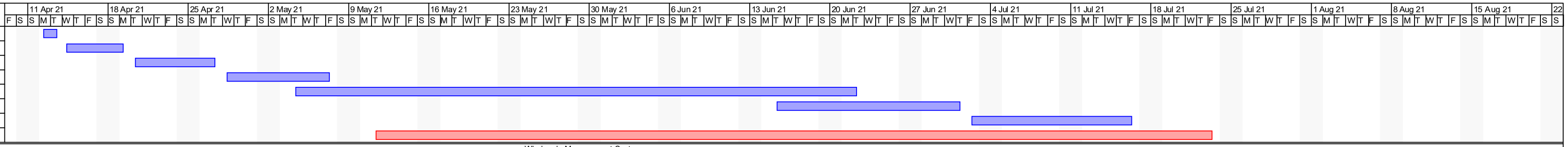


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Wholesale Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: **Debamoy Datta**
College Roll: **723**
Exam Roll: **2021141301**
Semester: **V**

Supervisor: **Dr. Chayan Halder**

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a "Customer Number" (CN). A customer can present 'CN' to customer staff when he makes purchases. In this case, the value of his purchase is credited against its 'CN'. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

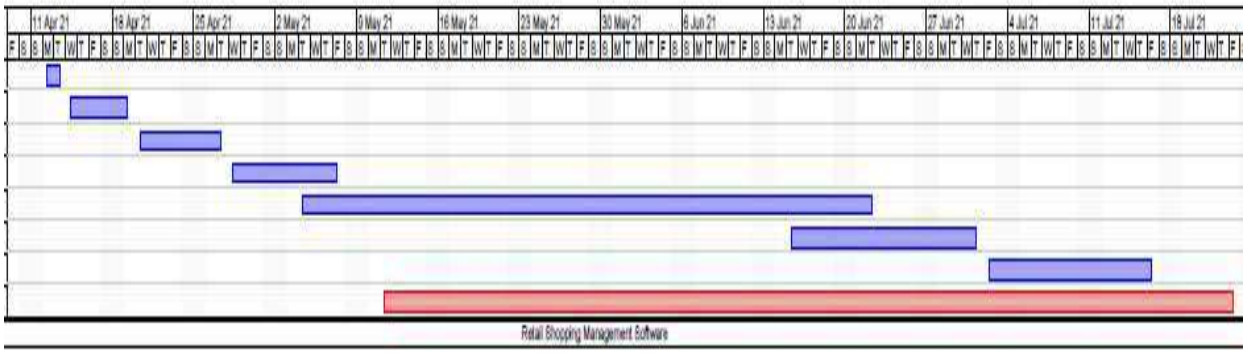
REQUIREMENT ANALYSIS (SRS):

As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account's purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- **Project Scheduling :**

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM

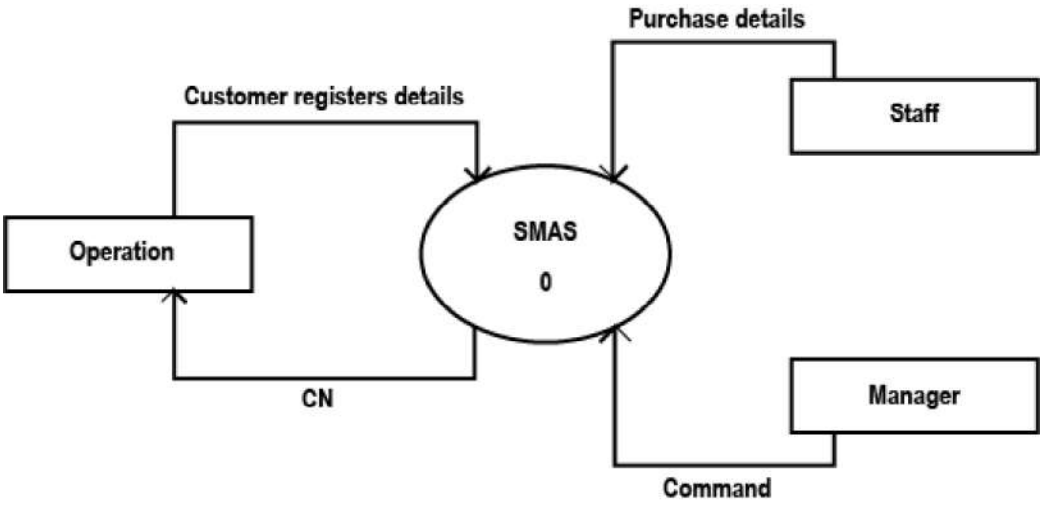


- **Project Resources :**
  - **Hardware Resources :** Several computer machines.
  - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, testeretc.
- **Risk management plan :** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

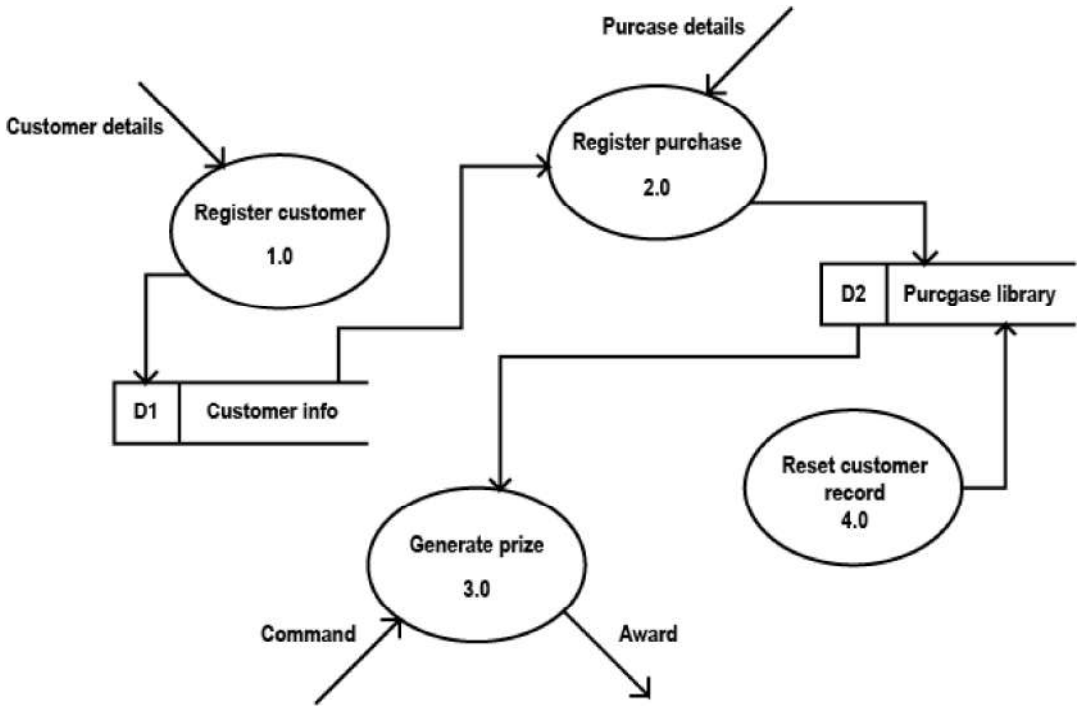
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

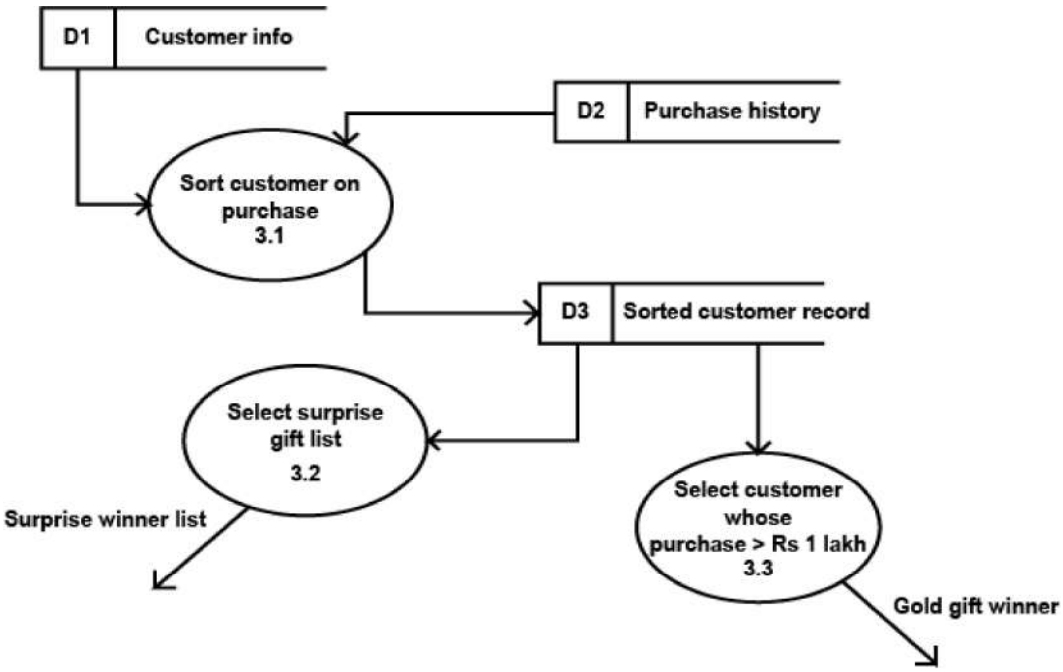
1. **Level 0 DFD :**



2. Level 1 DFD :



3. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

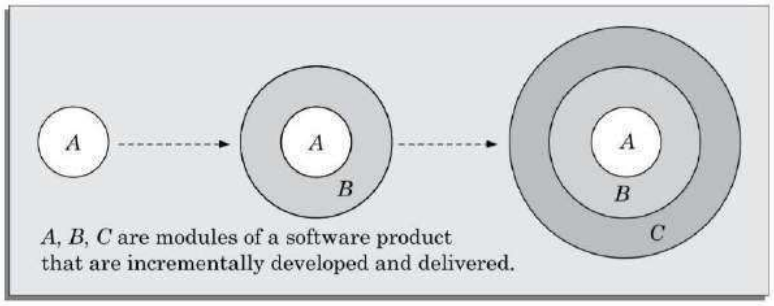
PROJECT-2

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement

It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule,ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model



Incremental model

Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module –** In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module –** In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module –** In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

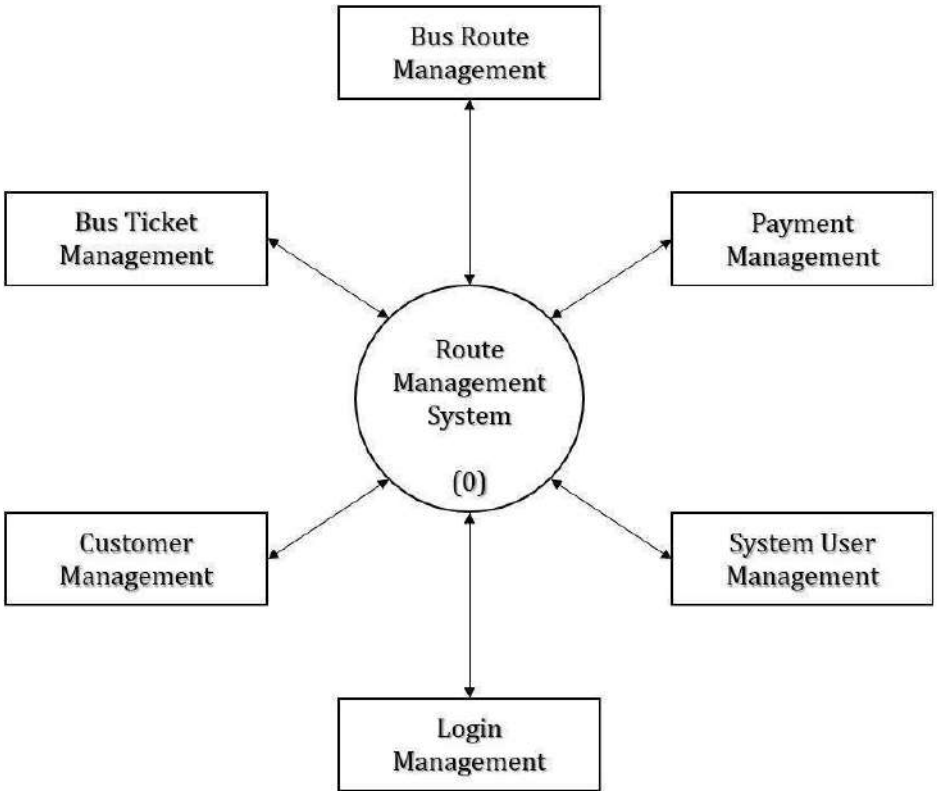


Fig – DFD level zero

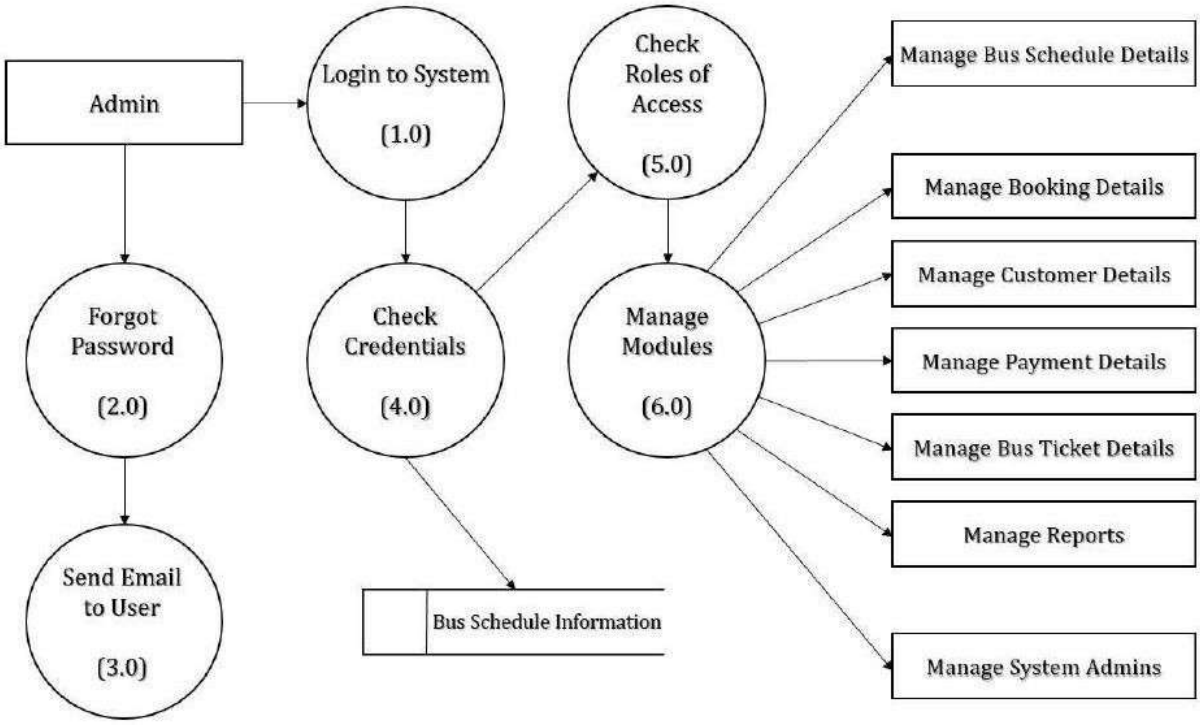
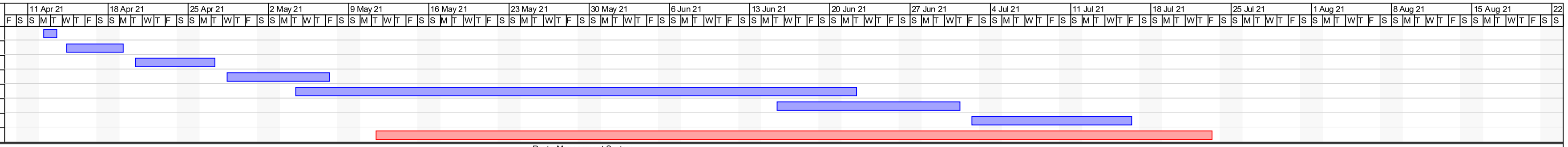


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance

The relative effort of development of a typical software product toits maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS):

As per the requirements of the software from client end by analysis we realise that we need three modules.

- **Goal of the project :** The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

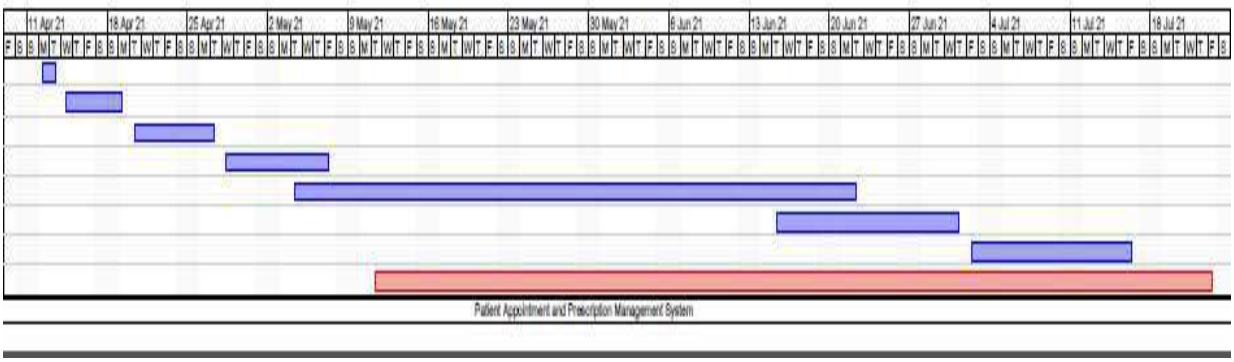
- **Functional requirements :** The three modules needed are.
    - **Administrator module :** This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
    - **Doctor module :** This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.
    - **Report module :** This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.
    - **Diagnostic module :** This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.

- **Non-Functional Requirements :**
  - **Usability :** The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - **Portability :** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
  - **Robustness :** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- **Project Scheduling :**

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



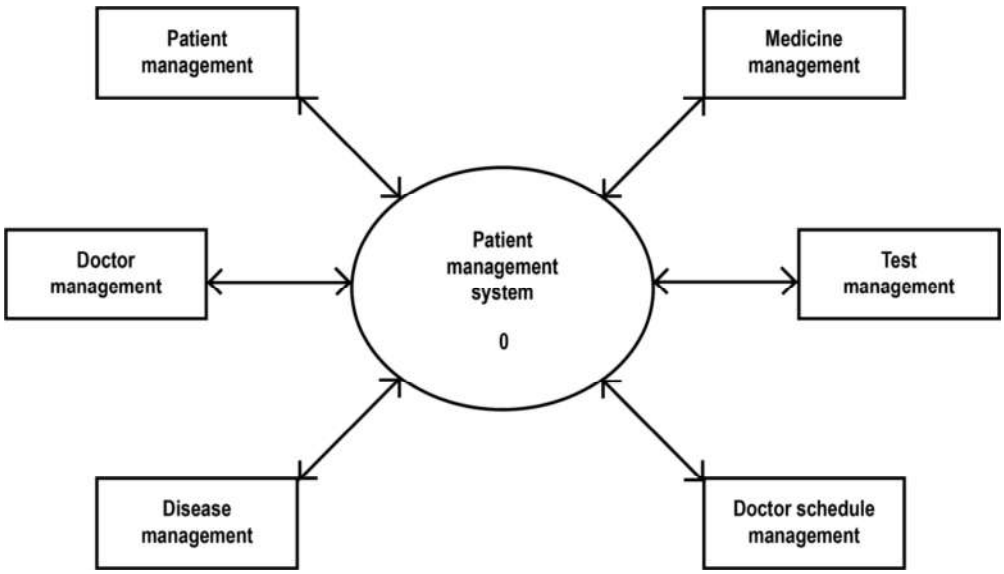
- **Project Resources :**
  - **Hardware Resources :** Several computer machines.
  - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, tester etc.

- **Risk management plan :** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

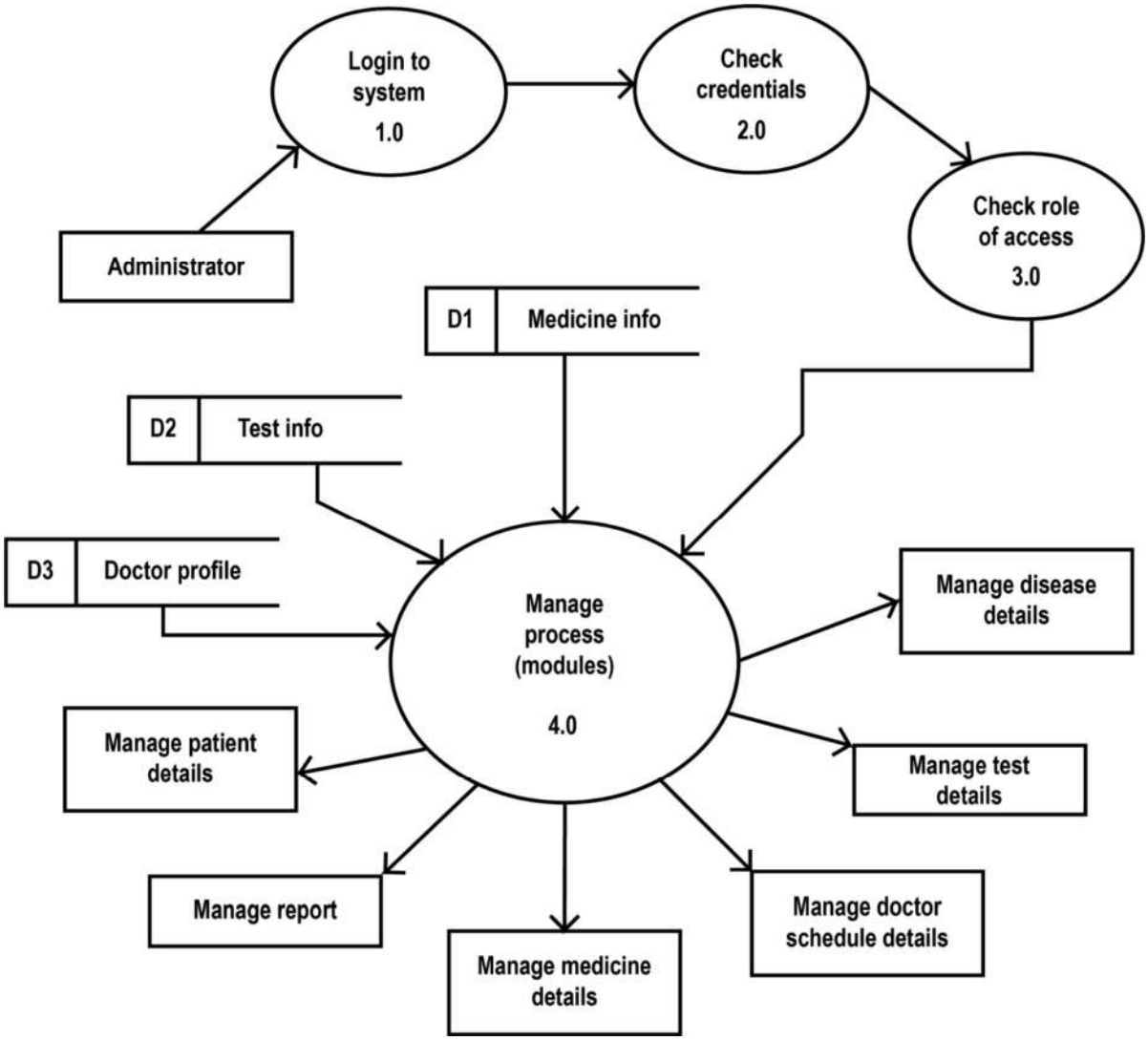
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. **Level 0 DFD :**



2. Level 1 DFD :


Login to system
1.0
Check credentials
2.0
Check role of access
3.0
Administrator
D1
Medicine info
D2
Test info
D3
Doctor profile
Manage process (modules)
4.0
Manage disease details
Manage patient details
Manage test details
Manage report
Manage medicine details
Manage doctor schedule details

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
    - **Alpha testing(Verification):** It is done by the development team.
    - **Beta testing(Validation):** It is done by a special group of friendly customers in an uncontrolled environment.
    - **Acceptance testing:** It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS):

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. **Guest module:** In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. **Stay module:** In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

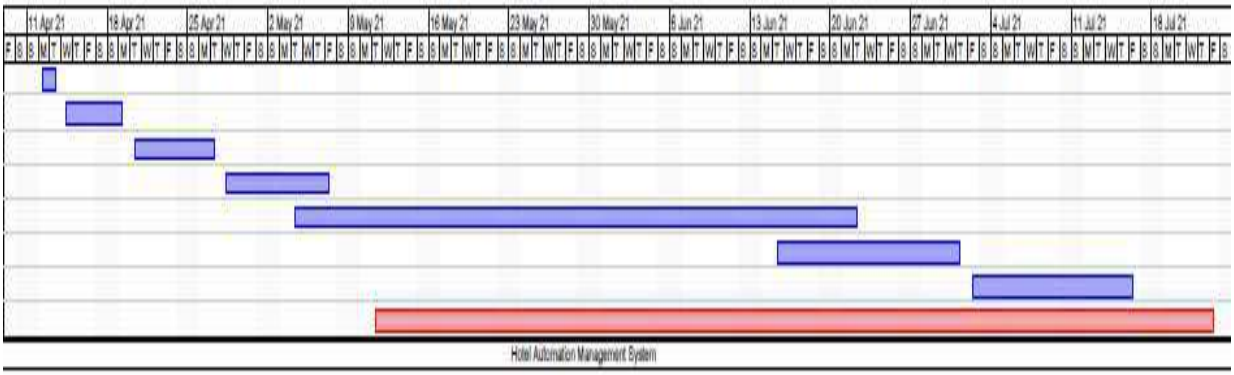
3. **Catering module:** In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.

4. **Bill generation module:** At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- **Project Scheduling :**

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



- **Project Resources :**
  - **Hardware Resources :** Several computer machines.
  - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, tester etc

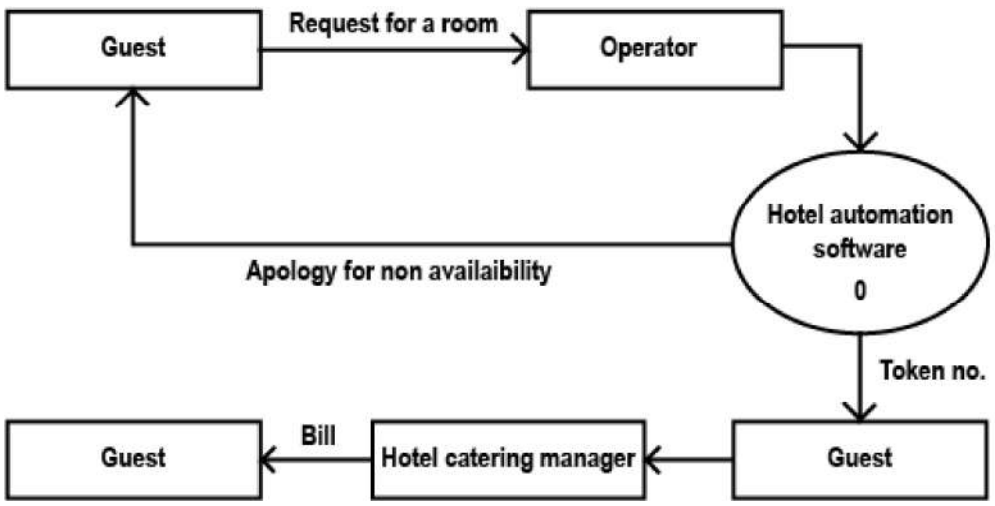
Risk management plan: Risks are measured by 2 parameters.

- Likelihood of a risk coming through (Probability of occurrence).
- Consequence of the problem associated with that risk.
  - Priority of risk is the multiplication of above two.
- Risk leverage =

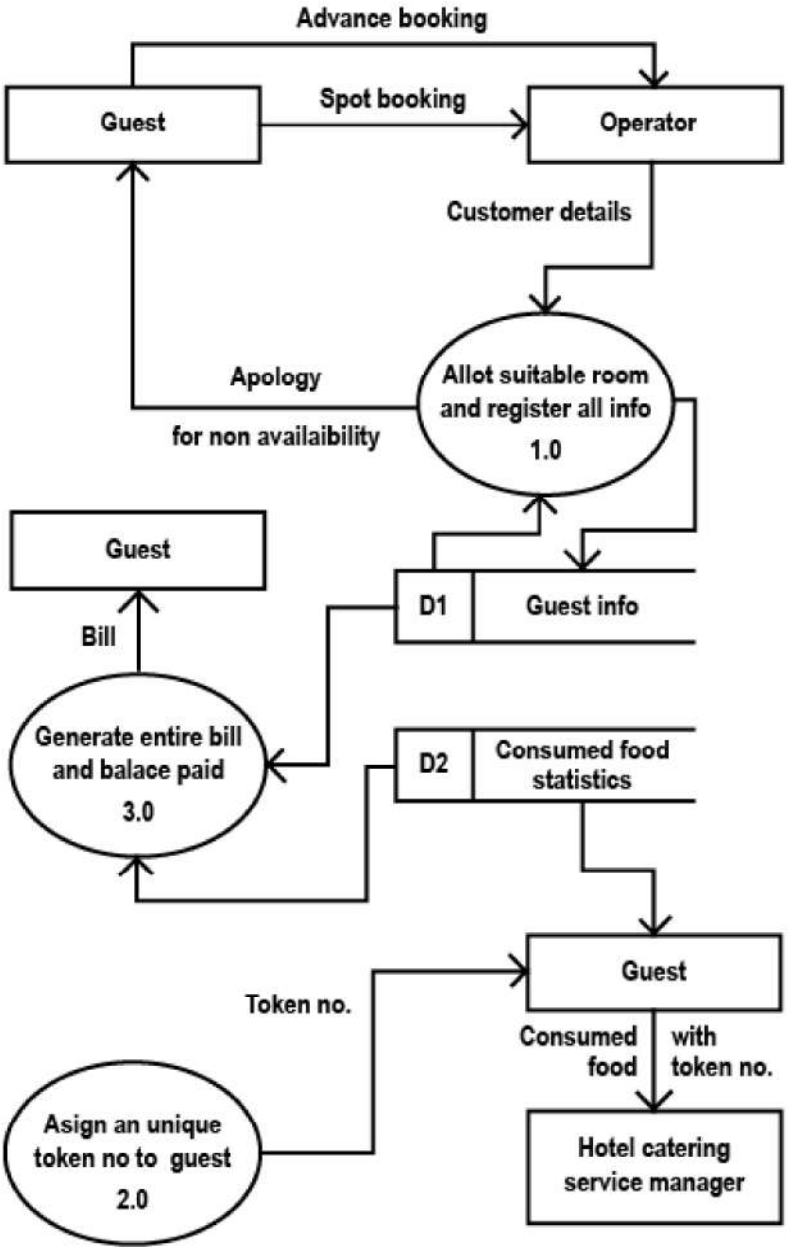
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Advance booking
Guest
Spot booking
Operator
Customer details
Apology
for non availaibility
Allot suitable room
and register all info
1.0
Guest
Bill
D1
Guest info
Generate entire bill
and balace paid
3.0
D2
Consumed food
statistics
Guest
Token no.
Consumed
food
with
token no.
Asign an unique
token no to guest
2.0
Hotel catering
service manager

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
    - **Alpha testing(Verification):** It is done by the development team.
    - **Beta testing(Validation):** It is done by a special group of friendly customers in an uncontrolled environment.
    - **Acceptance testing:** It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS):

As per the requirements of the software from client end by analysis we realise that we need three modules.

- **Goal of the project:** The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

- **Functional requirements:** The three modules needed are.
  - **Administrator module:** This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
  - **Criminal module:** This module accesses data from the database of criminal record details and manages it properly.
  - **FIR module:** This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
  - **Chargesheet module:** This module accesses data from the database of chargesheet record details and manages it properly.
  - **Court module:** The main function of this module is to manage the court profiles.

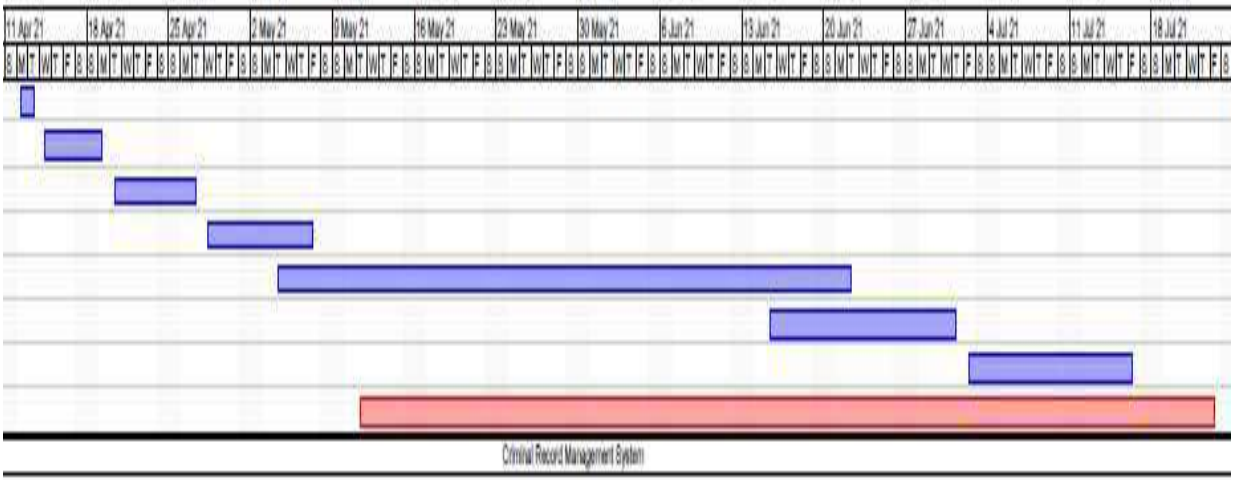
- **Non-Functional Requirements:**
  - **Usability:** The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

- **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- **Project Scheduling :**

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



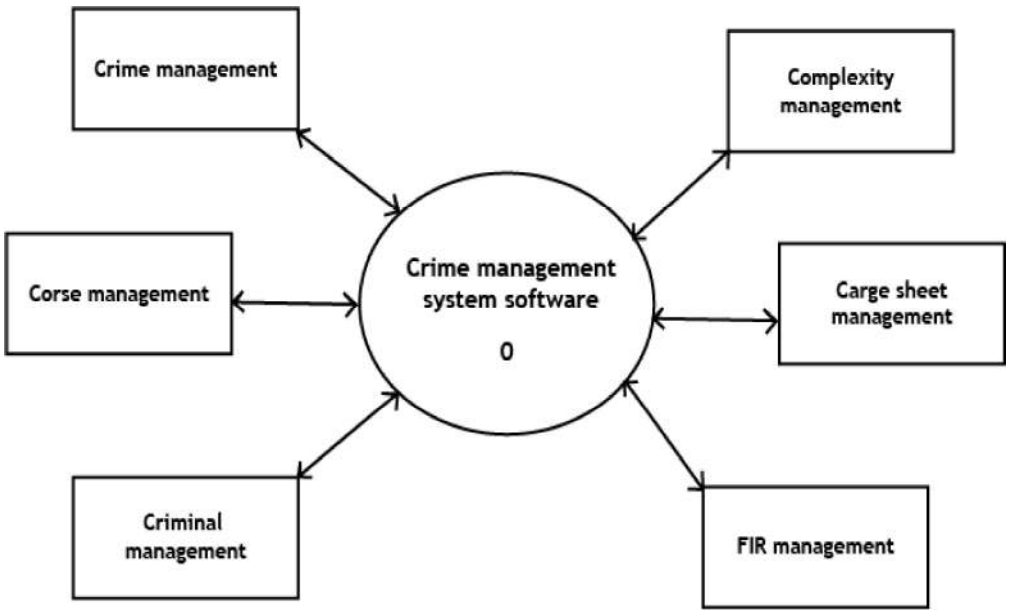
- **Project Resources :**
  - **Hardware Resources :** Several computer machines.
  - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, tester etc.

- **Risk management plan :** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

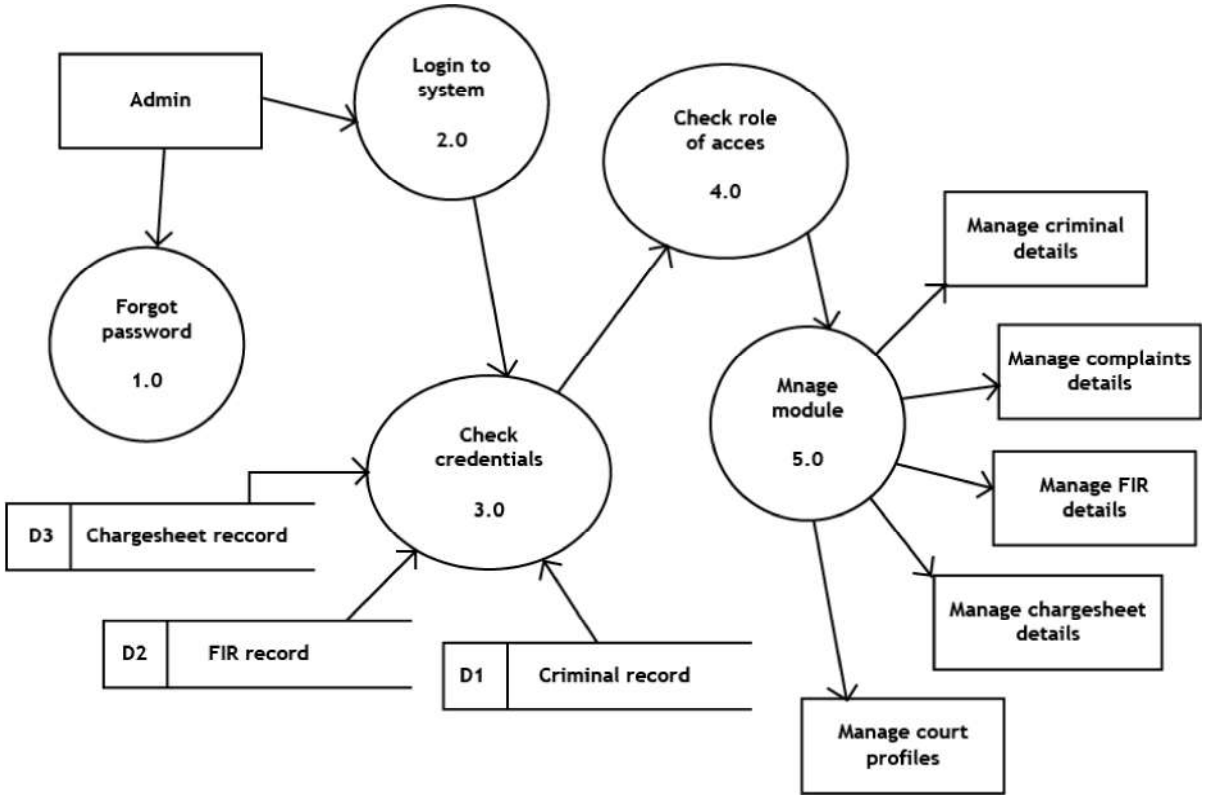
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
    - **Alpha testing(Verification):** It is done by the development team.
    - **Beta testing(Validation):** It is done by a special group of friendly customers in an uncontrolled environment.
    - **Acceptance testing:** It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement

Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for theexamination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model

Spiral development model

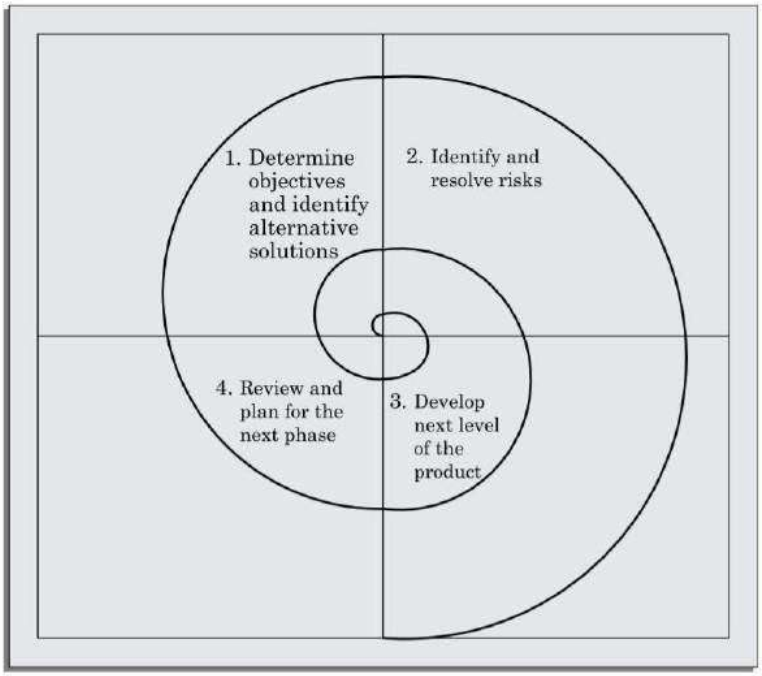


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module –** In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module –** In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module –** In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module –** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

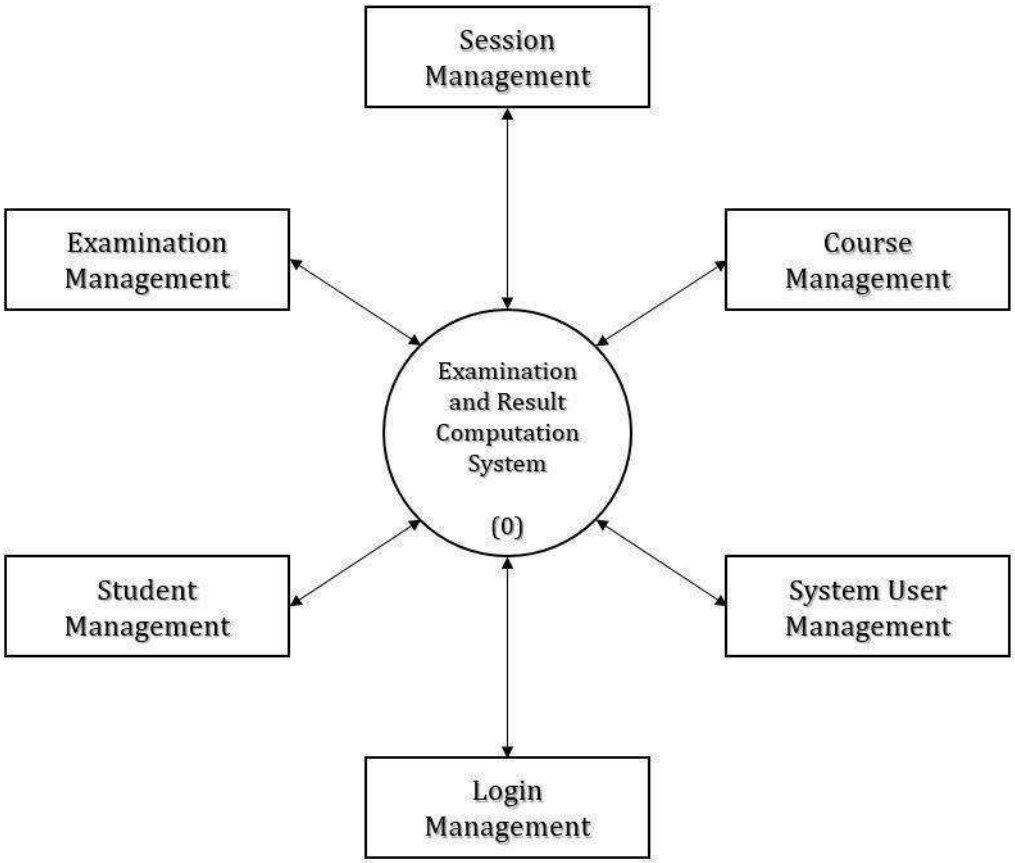


Fig – DFD level zero

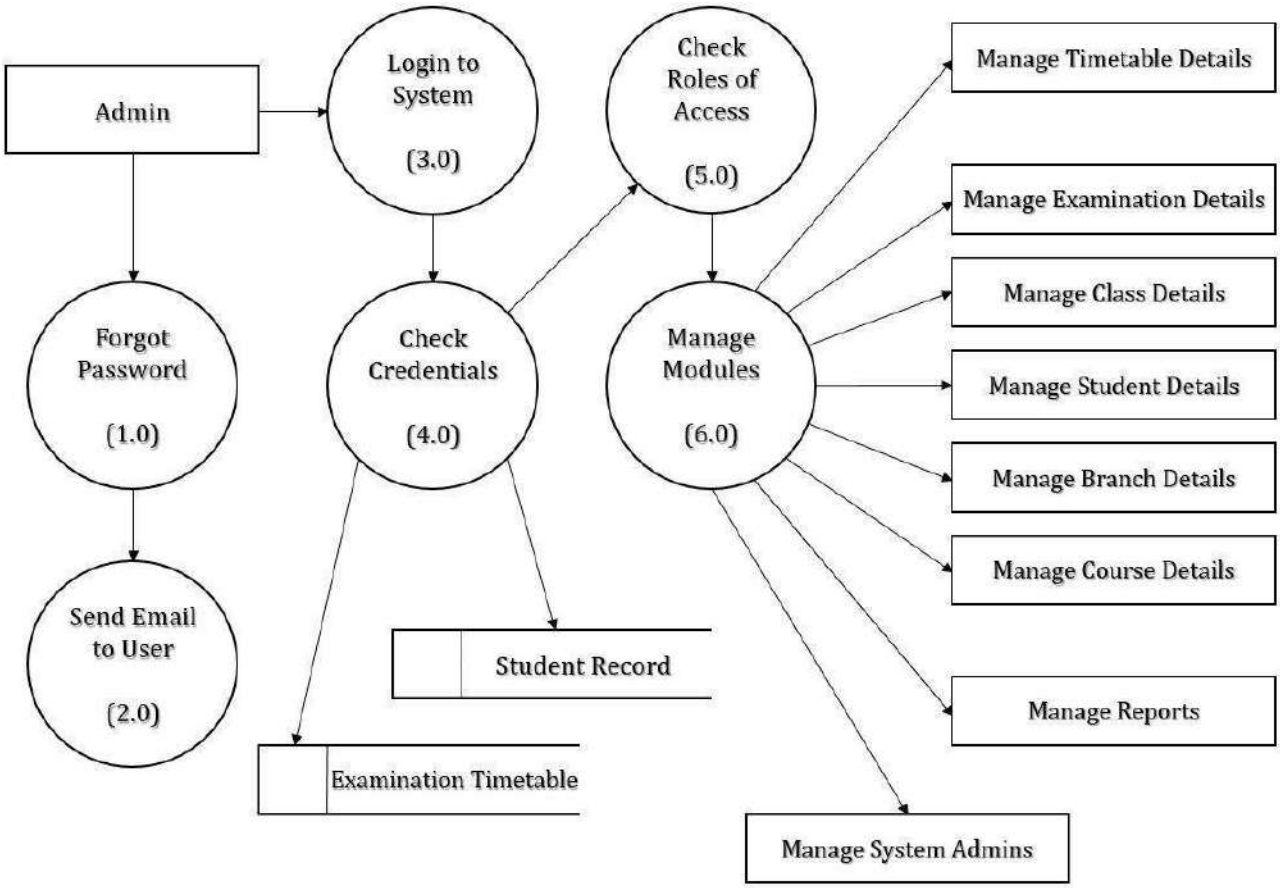
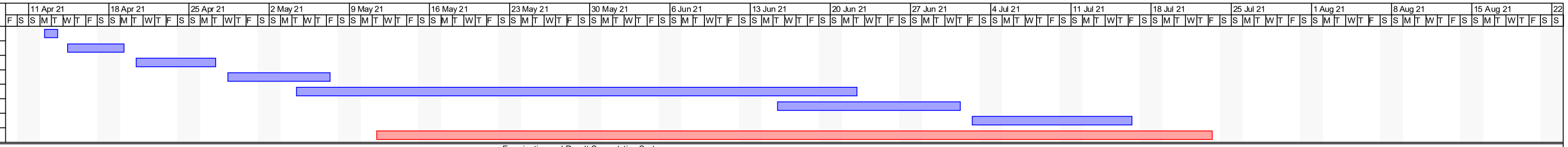


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product toits maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discoveredduring the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a newenvironment.

PARKING ALLOCATION SYSTEM

Problem Statement

Parking management system identifies internal data stores oflogin, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model

Spiral development model

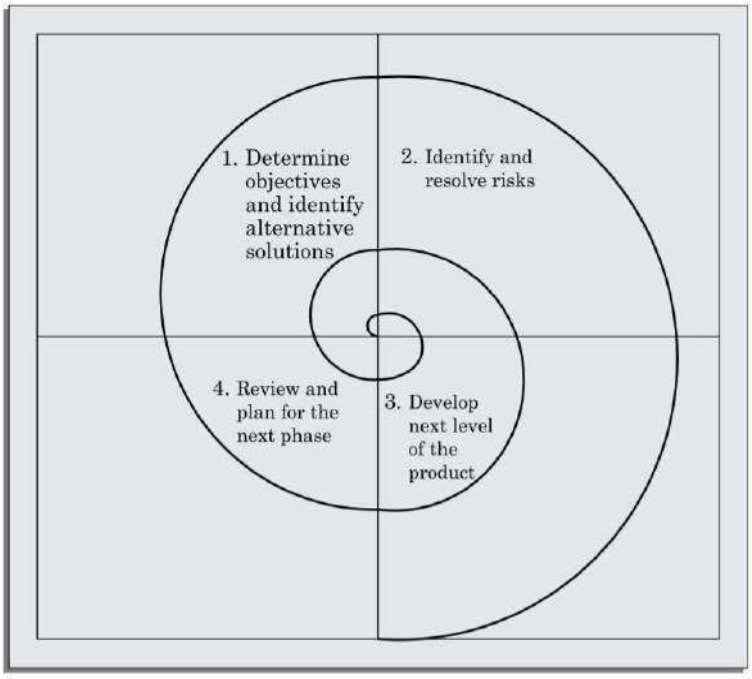


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module –** In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module –** In this module, parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module –** In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

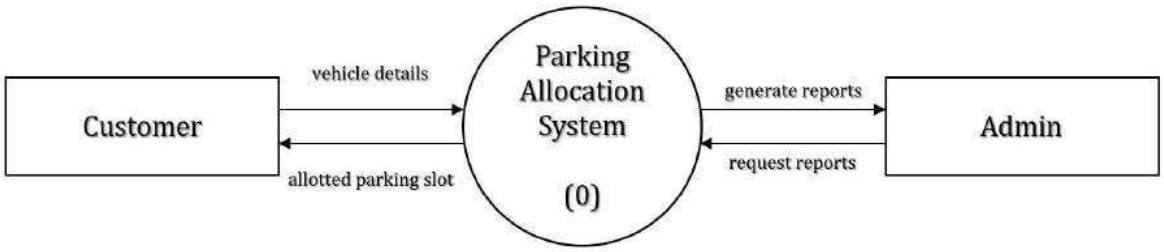


Fig – DFD level zero

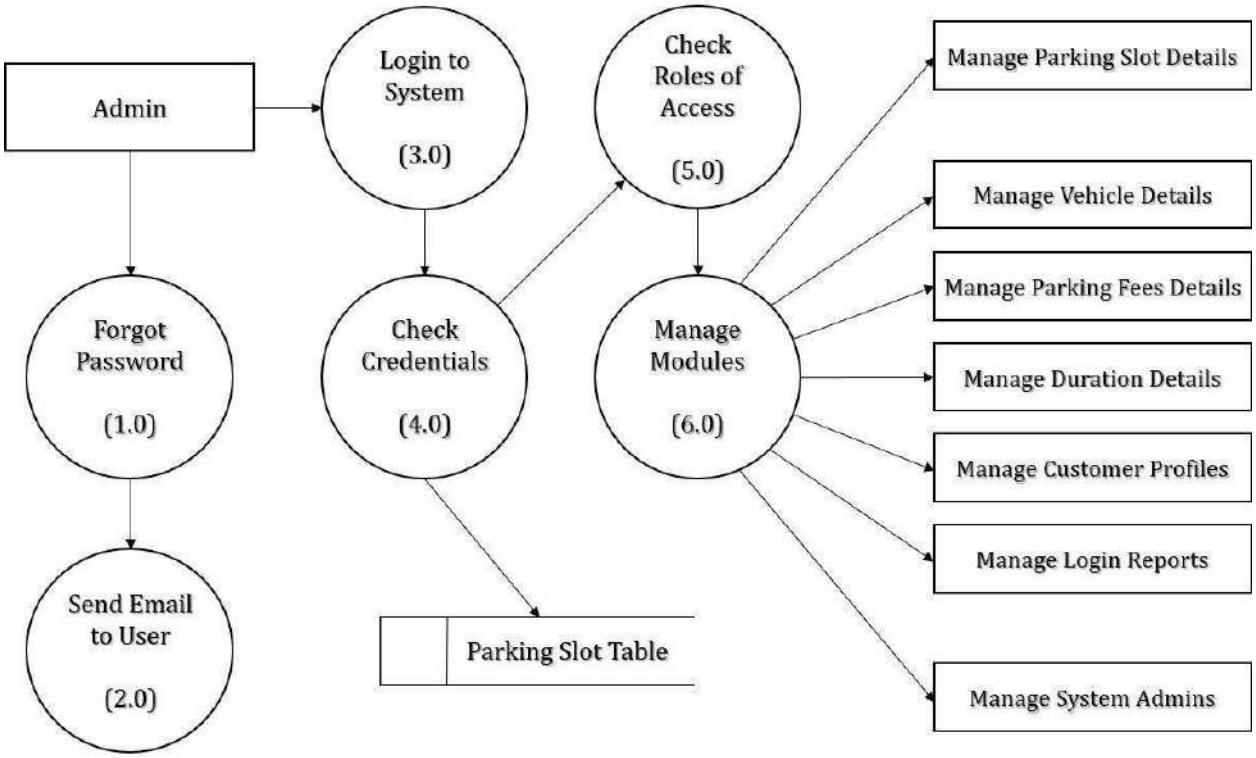
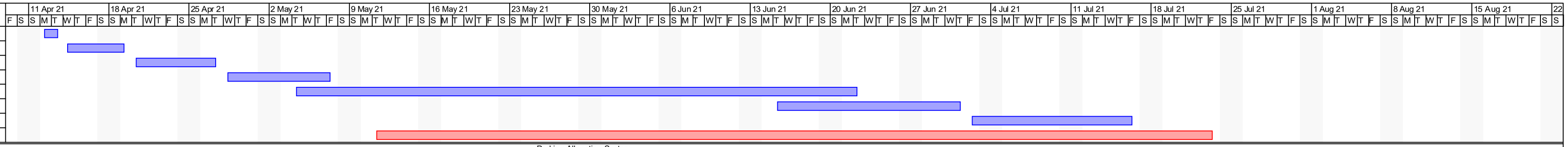


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Parking Allocation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance

The relative effort of development of a typical software product toits maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discoveredduring the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

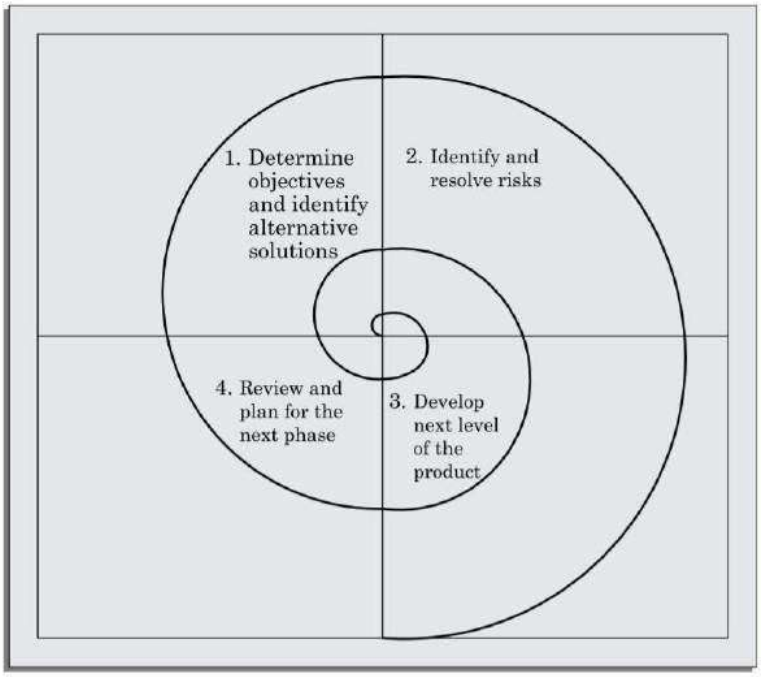
iii) **Adaptive maintenance –** Porting the software to work in a newenvironment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement

It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billingcounter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records,billing counter records and customer records are processed and reports are generated of the same.

Process Model



Spiral development model

Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

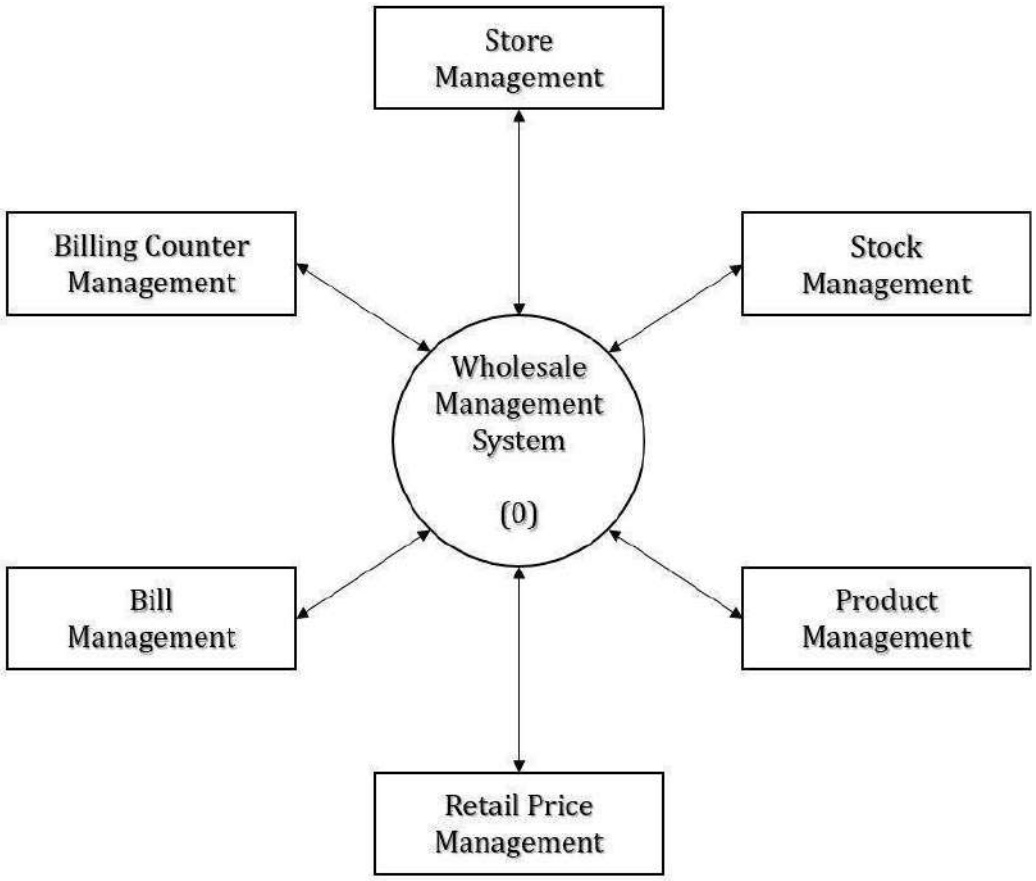


Fig – DFD level zero

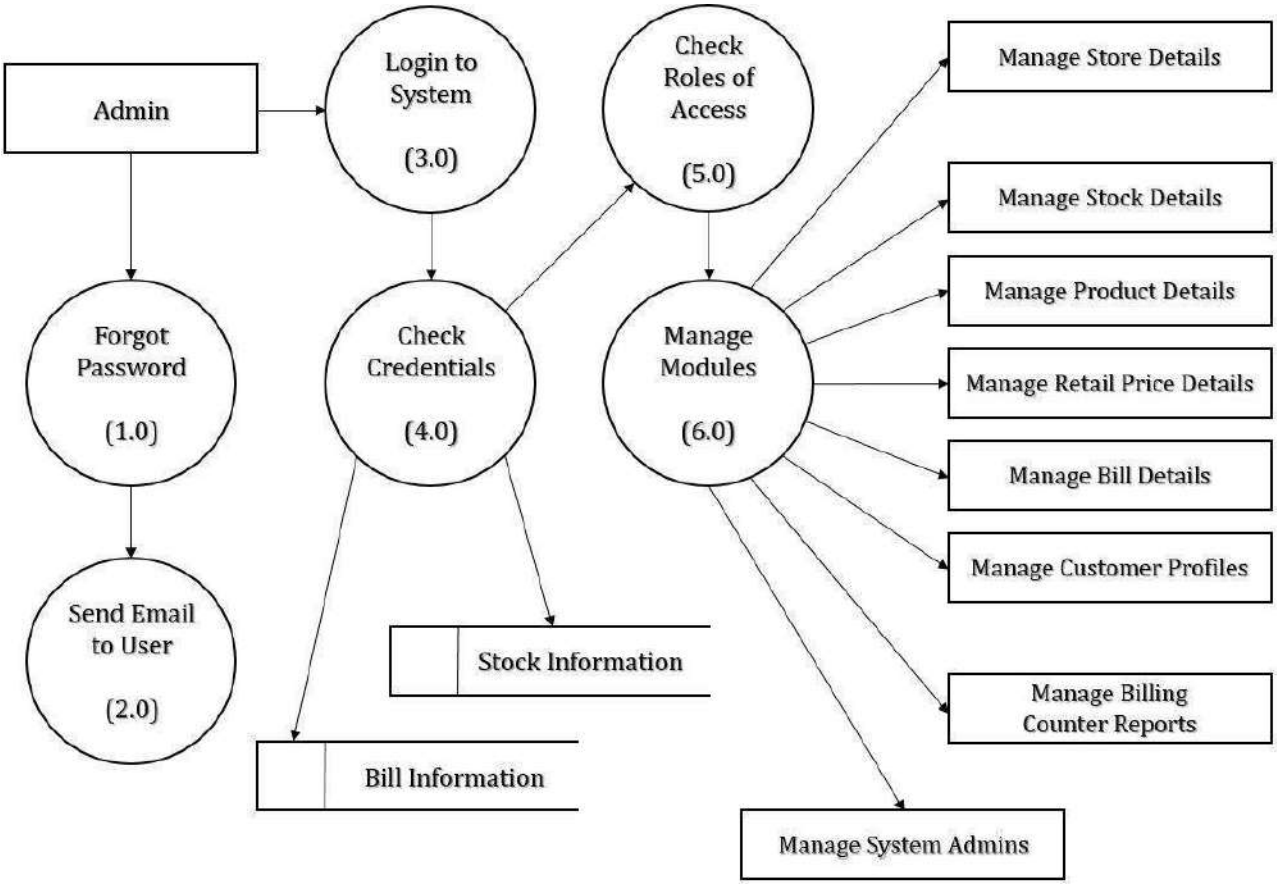
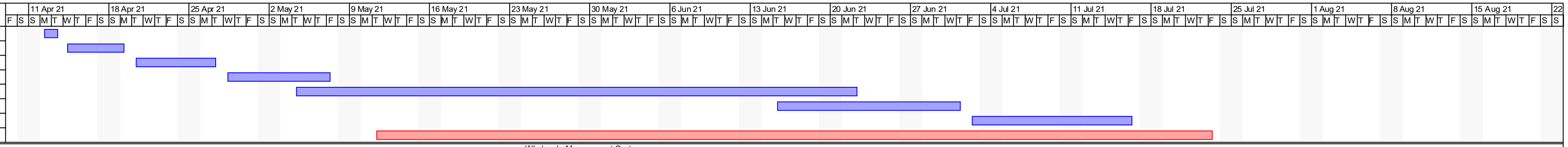


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Wholesale Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance

The relative effort of development of a typical software product toits maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

RAMAKRISHNA MISSION

VIVEKANANDA CENTENARY COLLEGE

COMPUTER SCIENCE DEPARTMENT

PROJECT ON SOFTWARE ENGINEERING

SUMBITTED BY

NAME : DIPENDU DHALI
COLLEGE ROLL NO. : 724
EXAMINATION ROLL NO. : 2021141302
REGISTRATION NO. : A01-1152-117-018-2019
SEMESTER : V
SUPERVISOR: Dr. BISWAJIT BISWAS

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement– A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

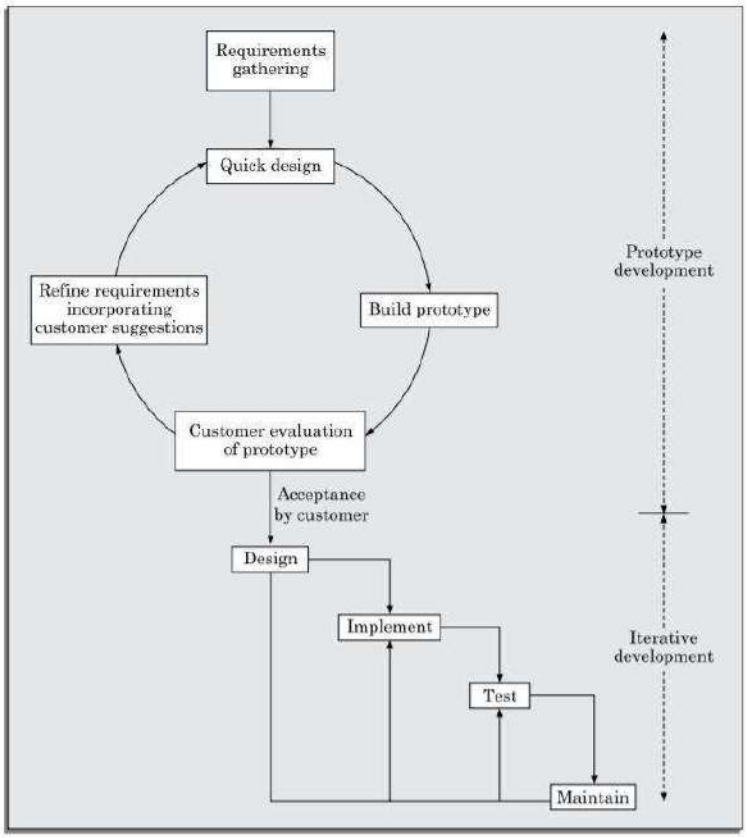


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly.Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Customer Module**–In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: uniqueCN

2. **Purchase Module**–In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input:purchase made by the customer, CN
   Output: updated database

3. **Promotional Module**–In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output:prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

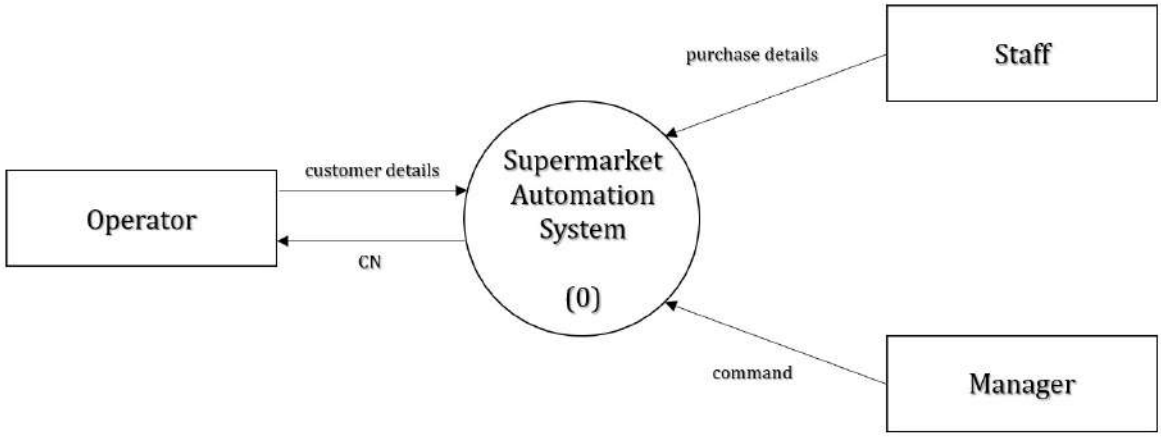


Fig – DFD level zero

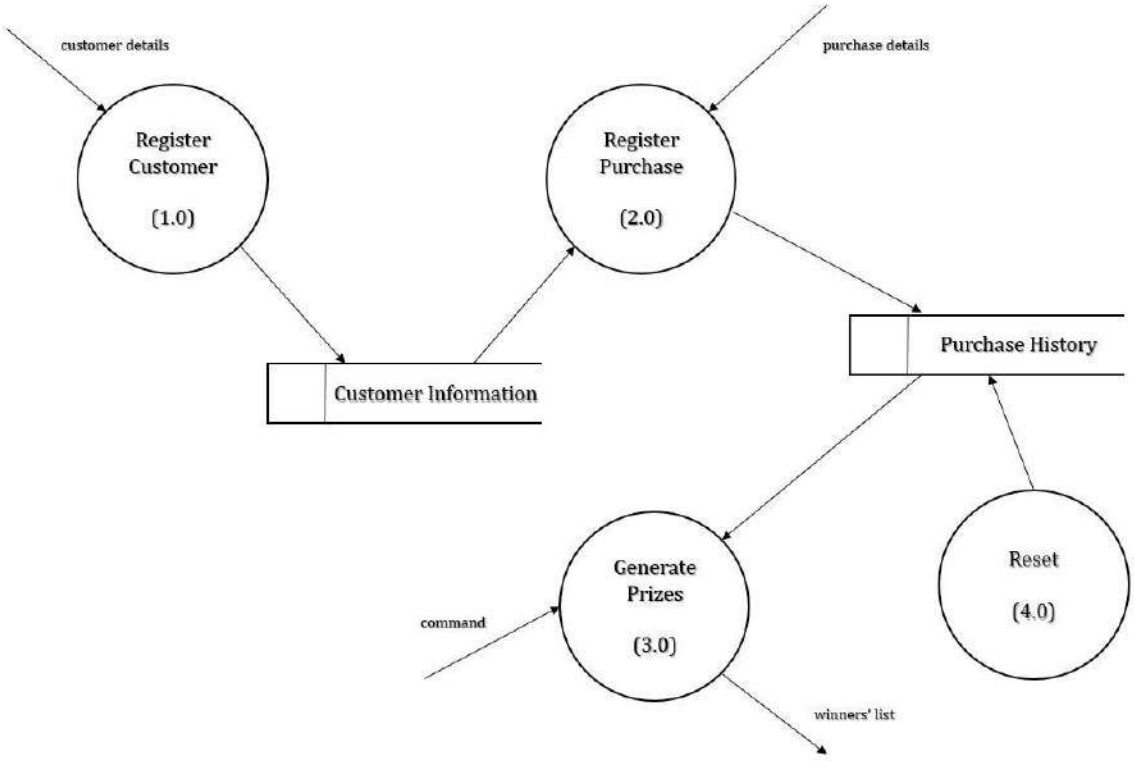


Fig – DFD level one

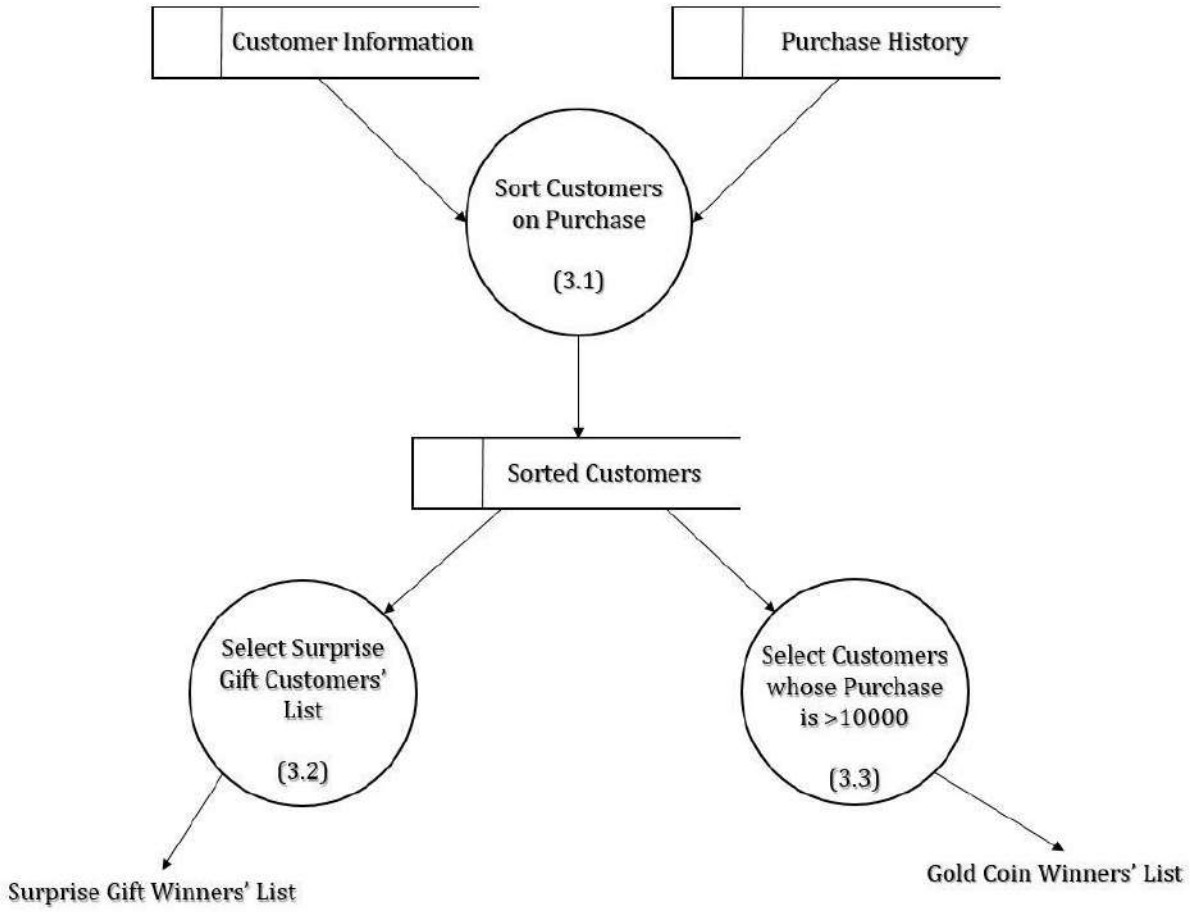
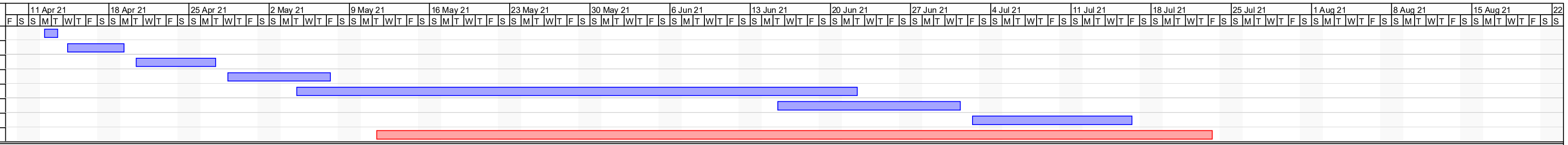


Fig – DFD level two of Process 3.0

Project Management

Project Schedule–The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Retail Shopping Management Software

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

ROUTE MANAGEMENT SYSTEM (DTC ROUTE INFORMATION)

Problem statement –It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process model –Incremental model

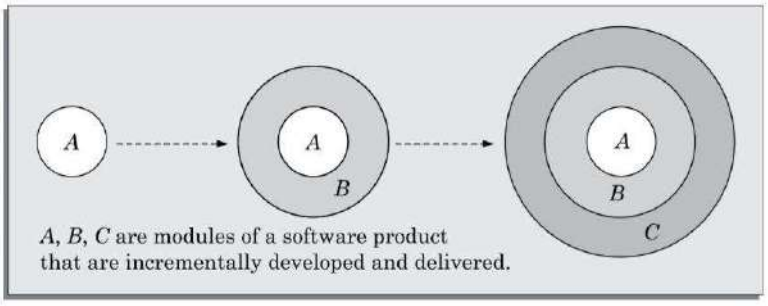


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module**–In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module**–In this module, any booking made by the customer is registered andthe ticket is generated.

   Input: customer details
   Output:generated ticket

3. **Information Module**–In this module,the busdetails of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module**–In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input:booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

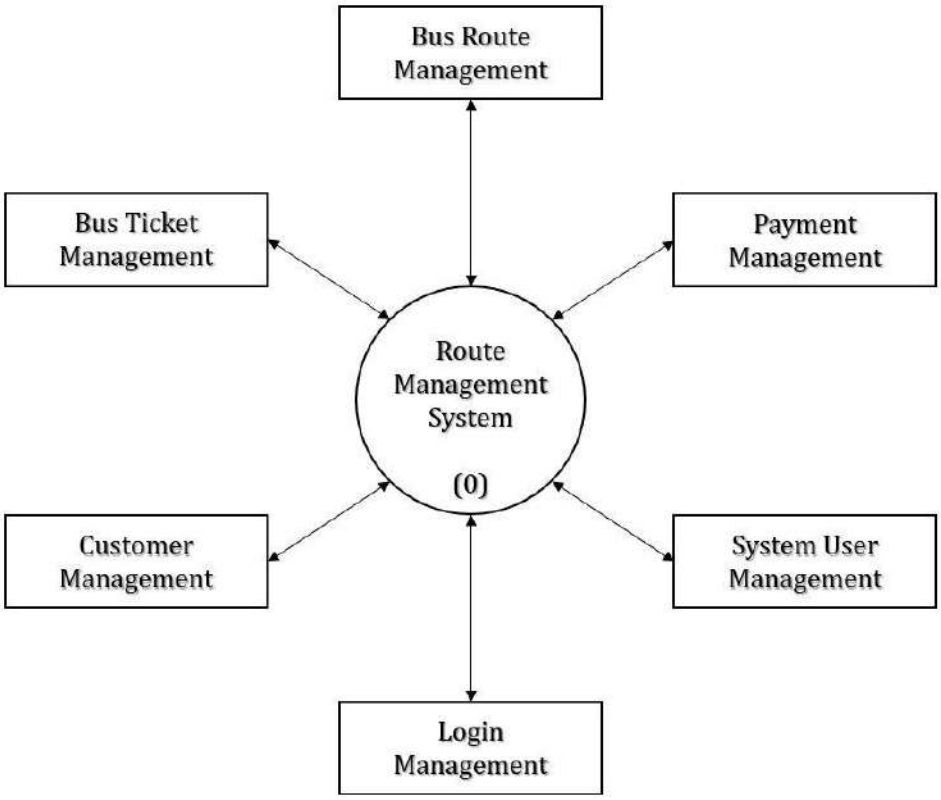


Fig – DFD level zero

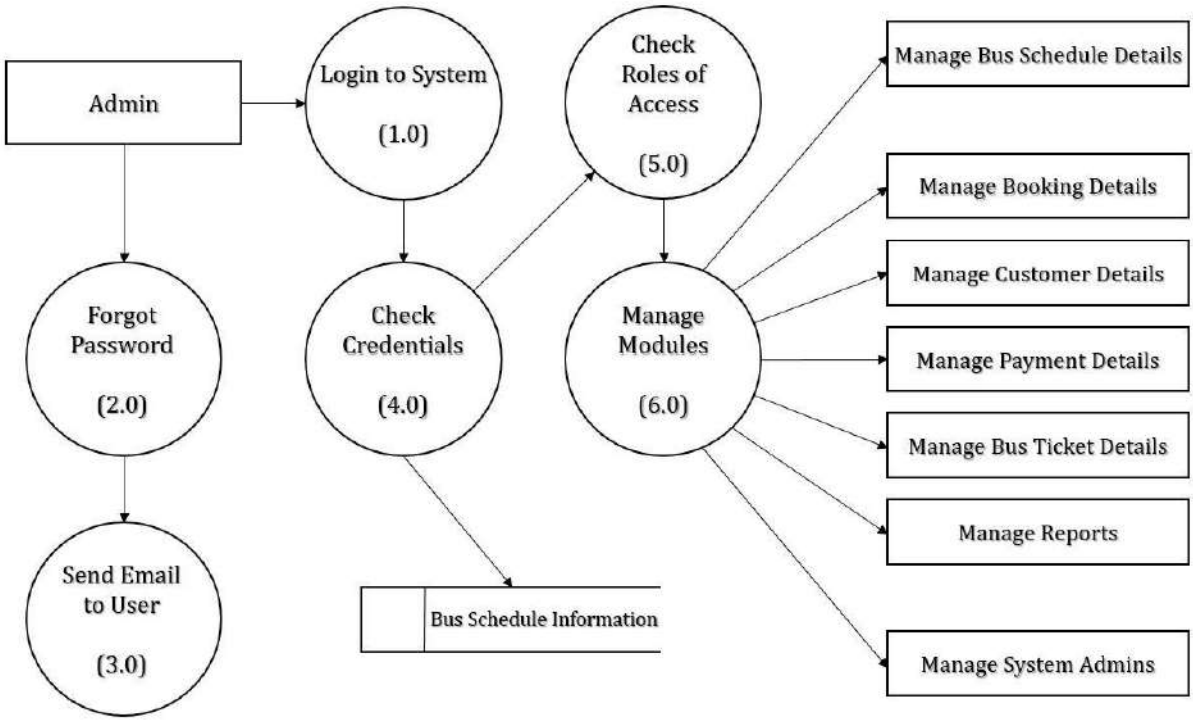
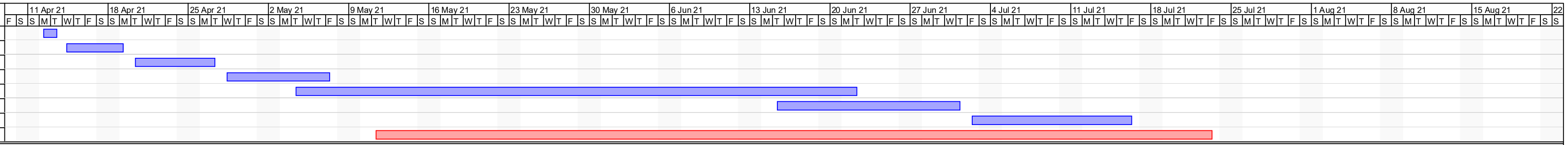


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

PATIENT APPOINTMENT AND PRESCRIPTION MANAGEMENT SYSTEM

Problem Statement –Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model –Incremental development model

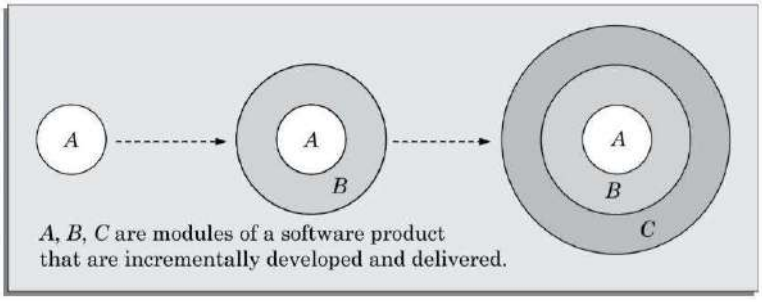


Fig – incremental model of software development

This life cycle model is also referred to as successive versions model.In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered.The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model.The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module**–In this module, a patient is registered and the profile is maintained.

   Input:patient details
   Output: unique profile

2. **Doctor Module**–In this module, the details of doctors are maintained.

   Input:patient details
   Output:updated database

3. **Diagnosis Module**–In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input:patient details, doctor details, test details
   Output:diagnosed disease, updated database

4. **Report Module**–In this module, patient records, medicine and drug records, test recordsare processed and reports are generated.

   Input:patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

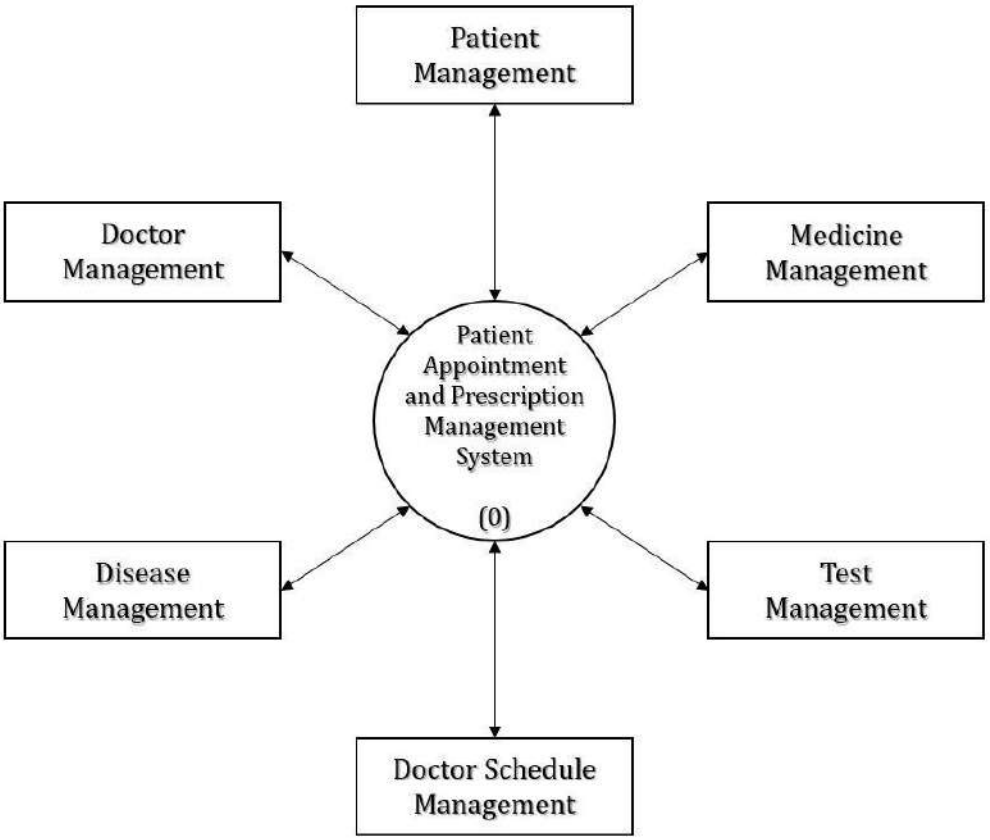


Fig – DFD level zero

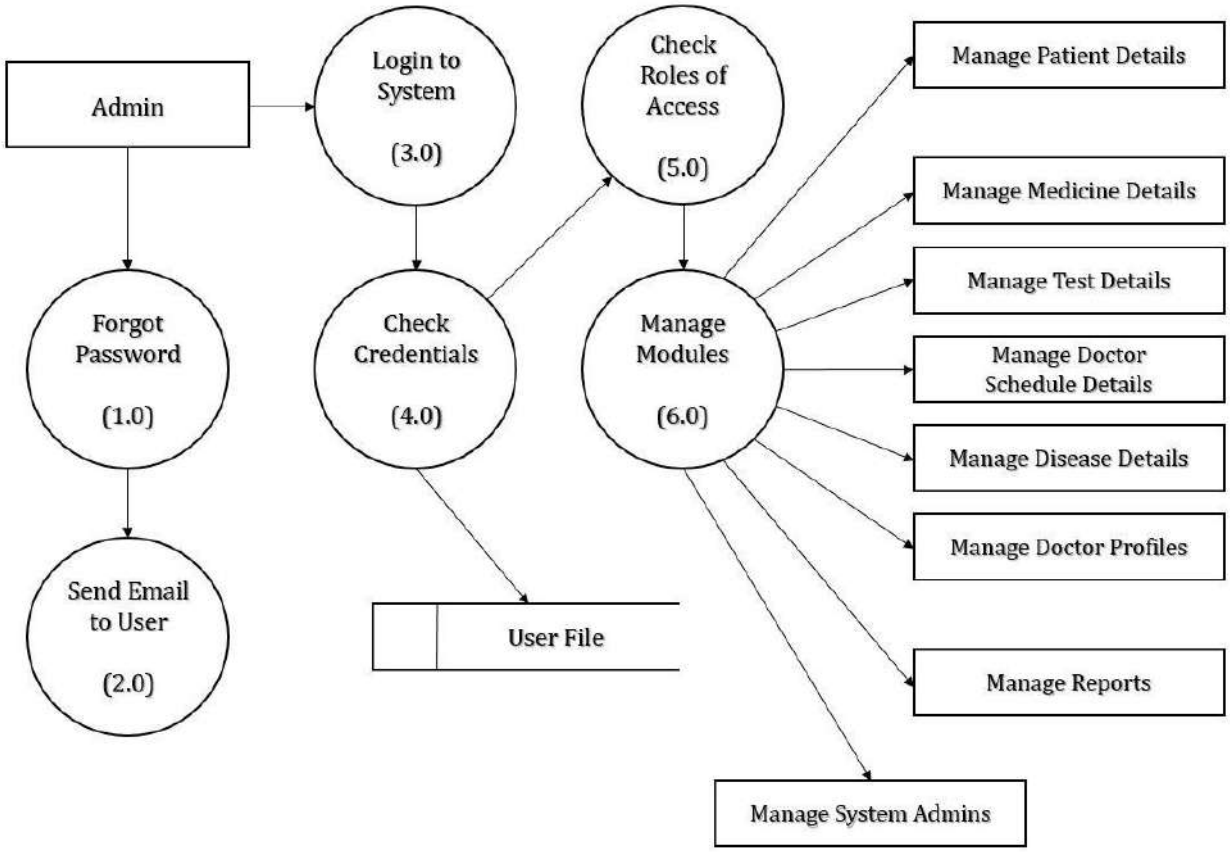
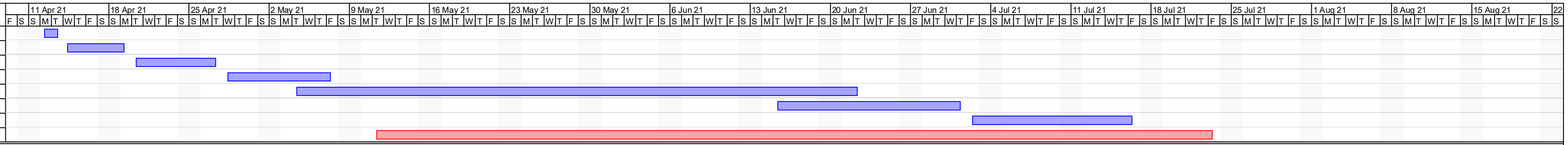


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

HOTEL AUTOMATION MANAGEMENT SYSTEM

Problem Statement –Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model –Spiral development model

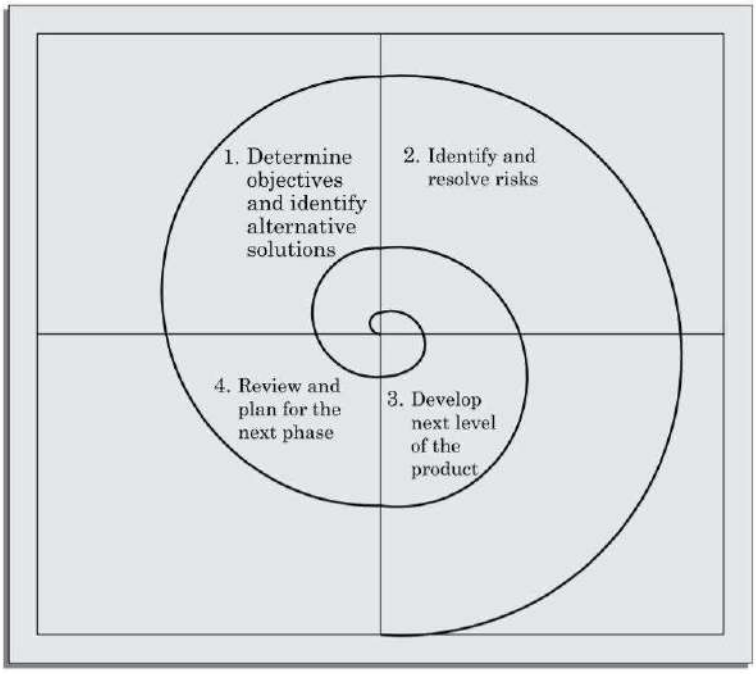


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module–**In this module, a guest is registered and the profile is maintained.If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input:guest details
   Output: unique profile, updated database

2. **Catering Module–**In this module, the details of food items consumed by the guest is maintained.
   A database is maintained for keeping the information.

   Input:guest details, room number, token number
   Output:updated database

3. **Bill Module–**In this module, the totalbill is generated when the guest prepares to check out.

   Input:guest details, room number, token number, food items consumed
   Output:generated bill

Design Engineering

Data Flow Diagrams (DFDs)

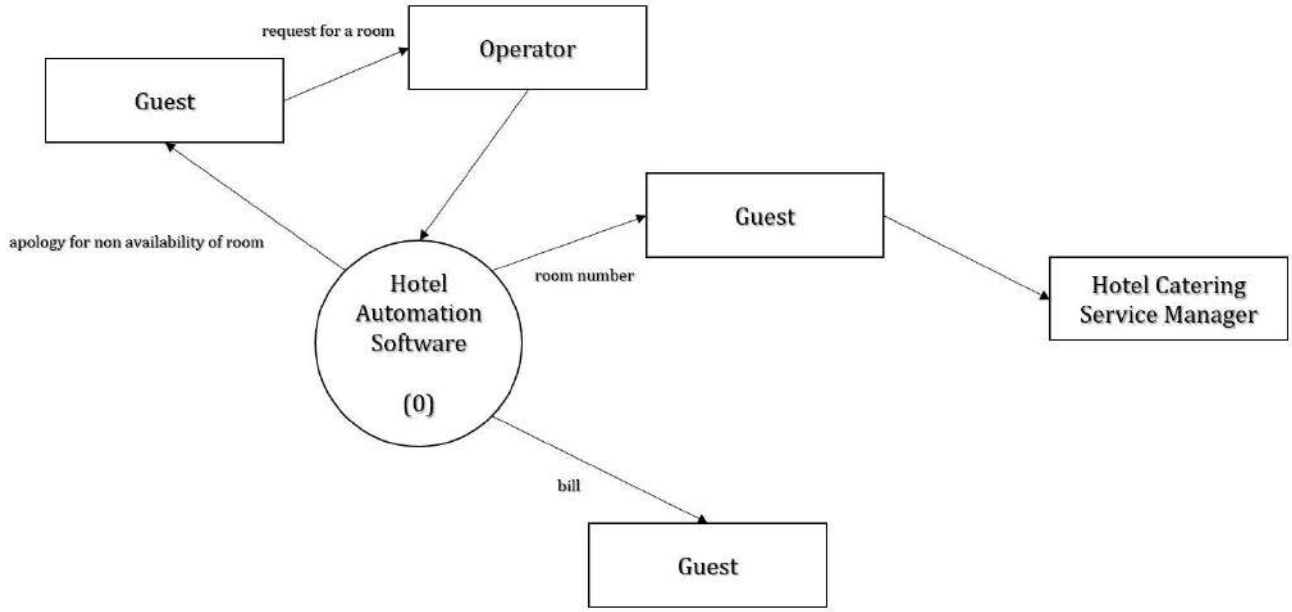


Fig – DFD level zero

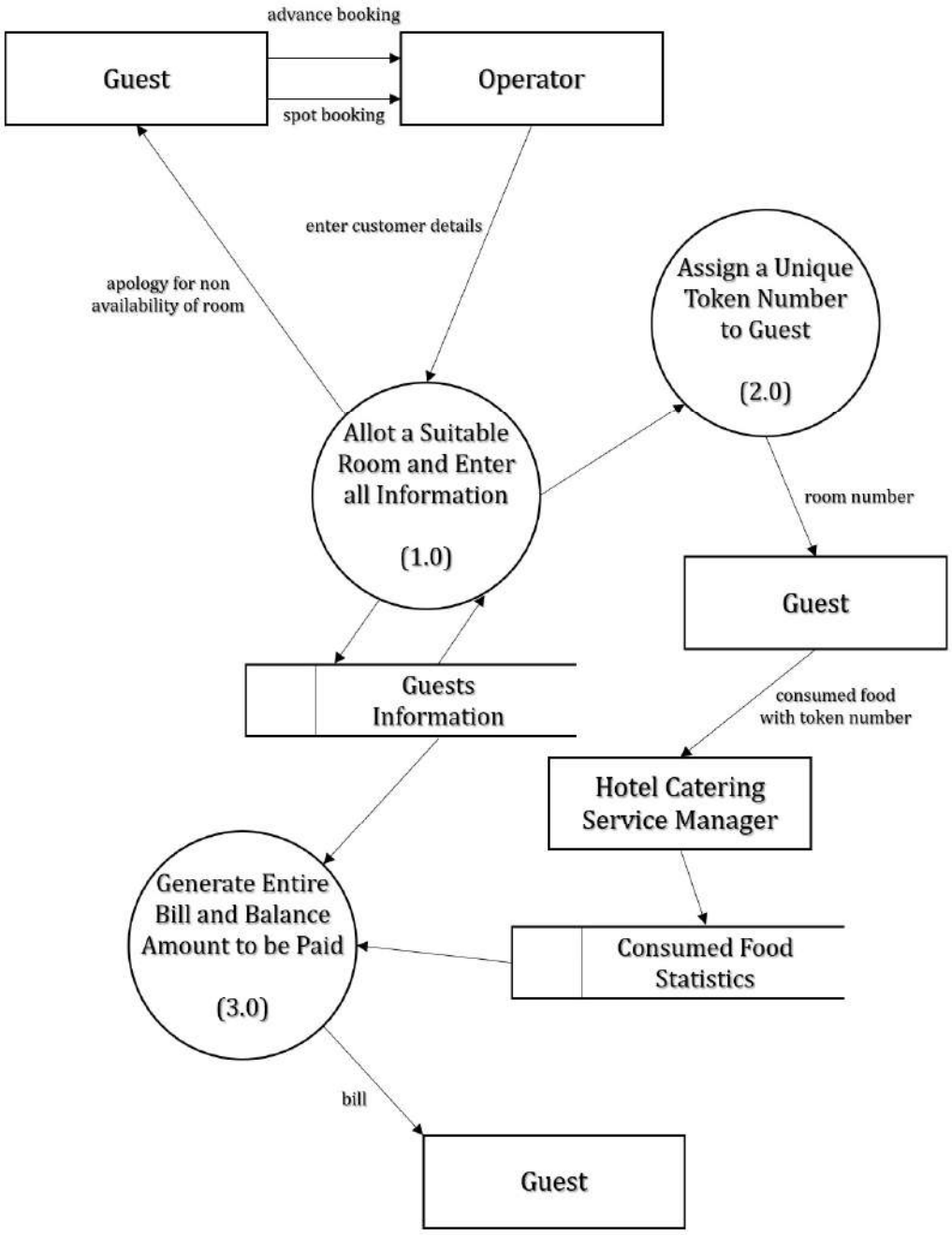
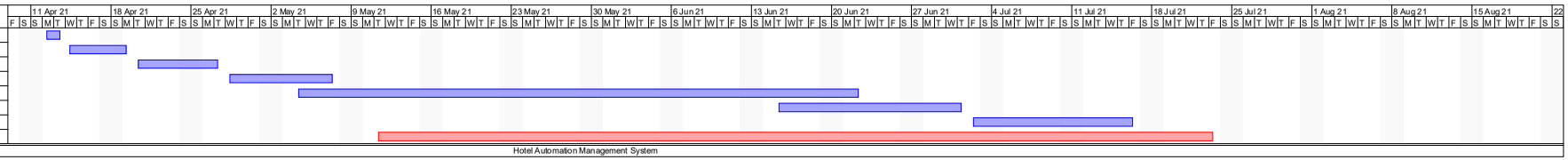


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Hotel Automation Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing–** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing–** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing–** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance–** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance–** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance–** Porting the software to work in a new environment.

CRIMINAL RECORD MANAGEMENT SYSTEM

Problem Statement –Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

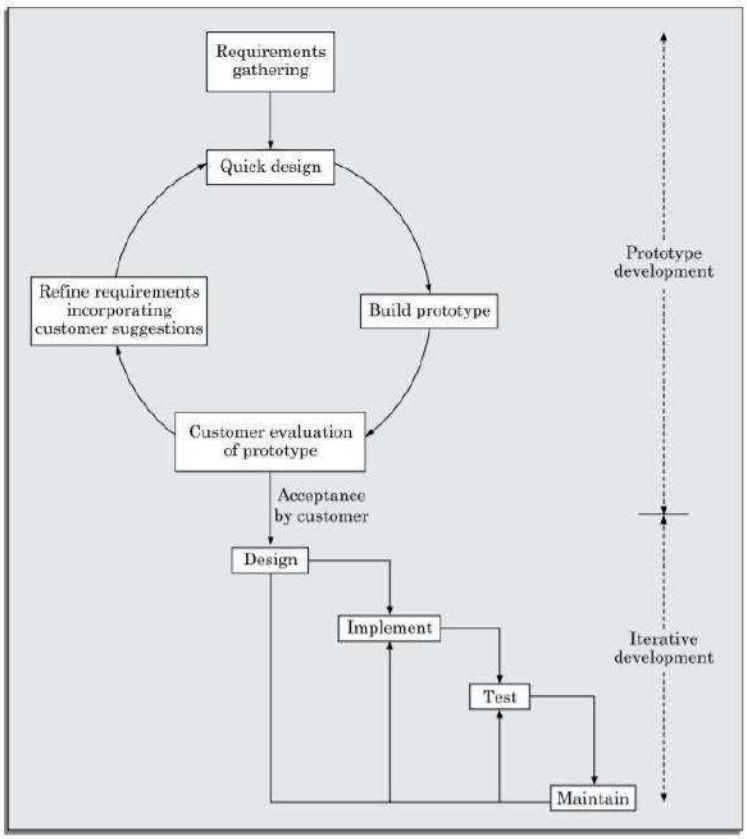


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly.Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module**–In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input:admin details
   Output: unique profile, granted level of access

2. **Complain Module**–In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output:filed FIR, updated database

3. **Crime Module**–In this module, the details of the crime are maintained and chargesheet is filed.

   Input:complaint details, FIR details
   Output:filed chargesheet

4. **Court Module**–In this module, the case is taken to the court and legal action is taken accordingly.

   Input:complaint details, FIR details, chargesheet details
   Output:legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

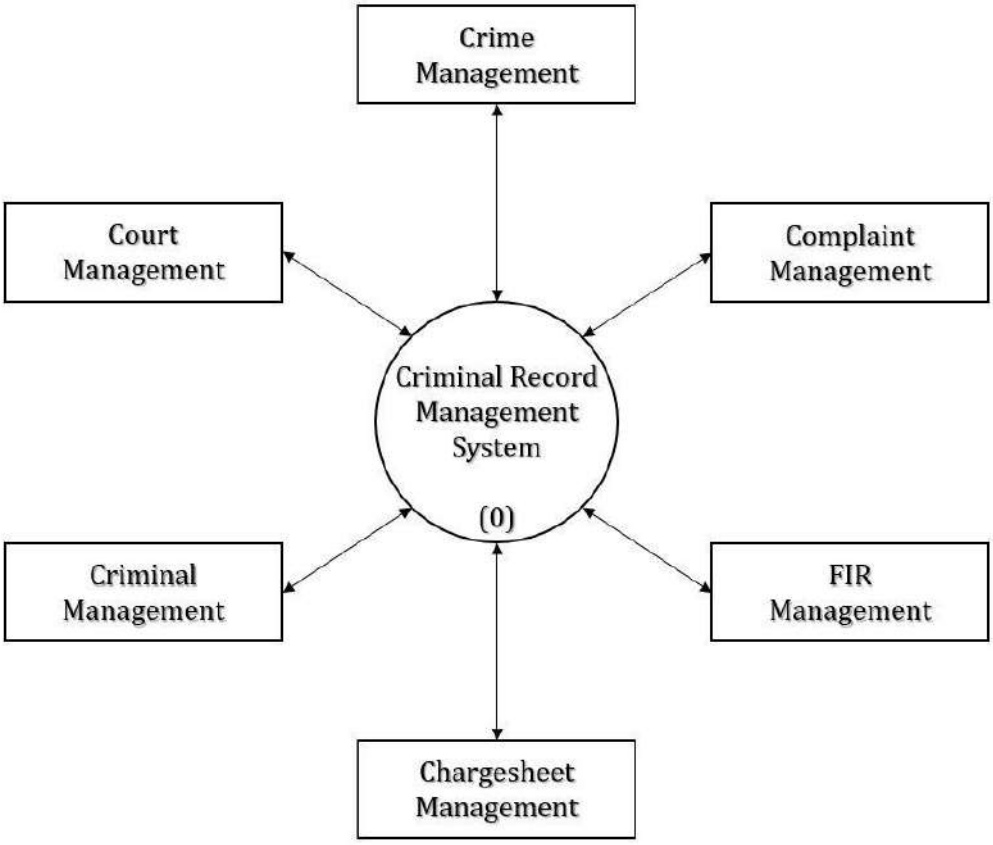


Fig – DFD level zero

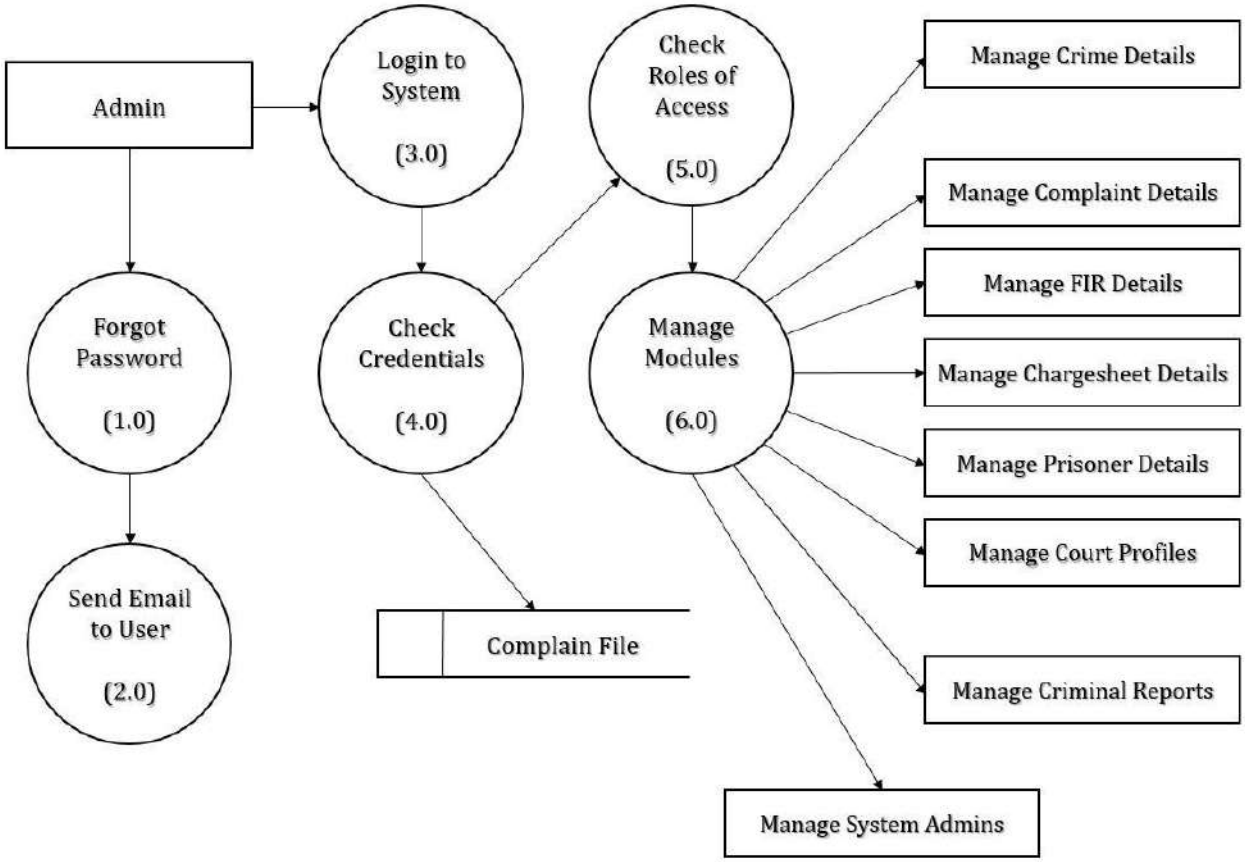
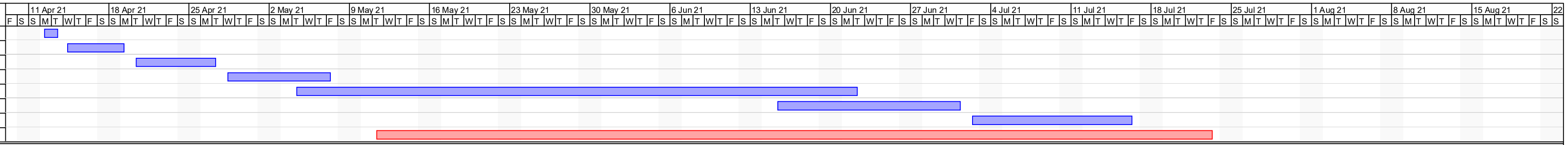


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Criminal Record Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

EXAMINATION AND RESULT COMPUTATION SYSTEM

Problem Statement –Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model –Spiral development model

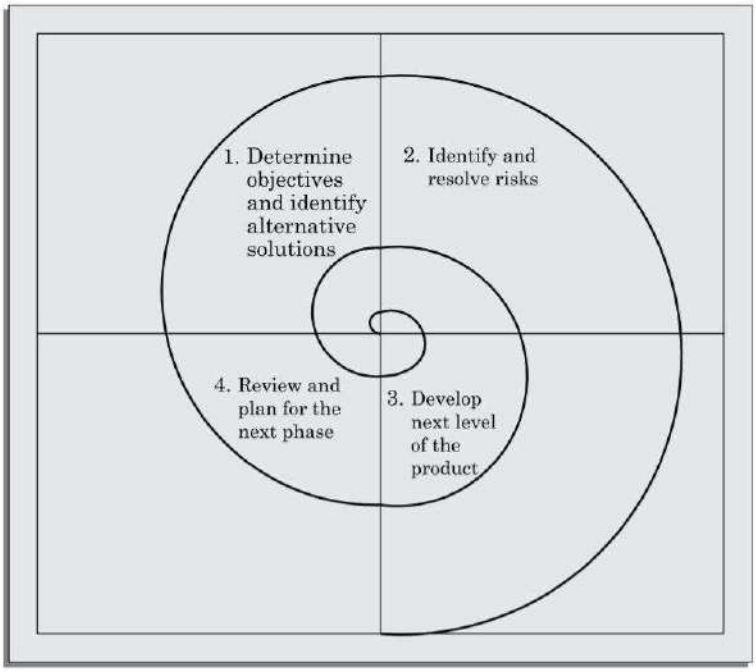


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module**–In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input:student details
   Output: unique profile

2. **Timetable Module**–In this module, the details of branch, course, class and subject ismaintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input:branchdetails, coursedetails, classdetails, subject details
   Output:updated database, created timetable

3. **Examination Module**–In this module,the examination takes place and the results are computed.

   Input:student details, timetable details
   Output:computed results, updated database

4. **Report Module**–In this module,timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input:guest details, room number, token number, food items consumed
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

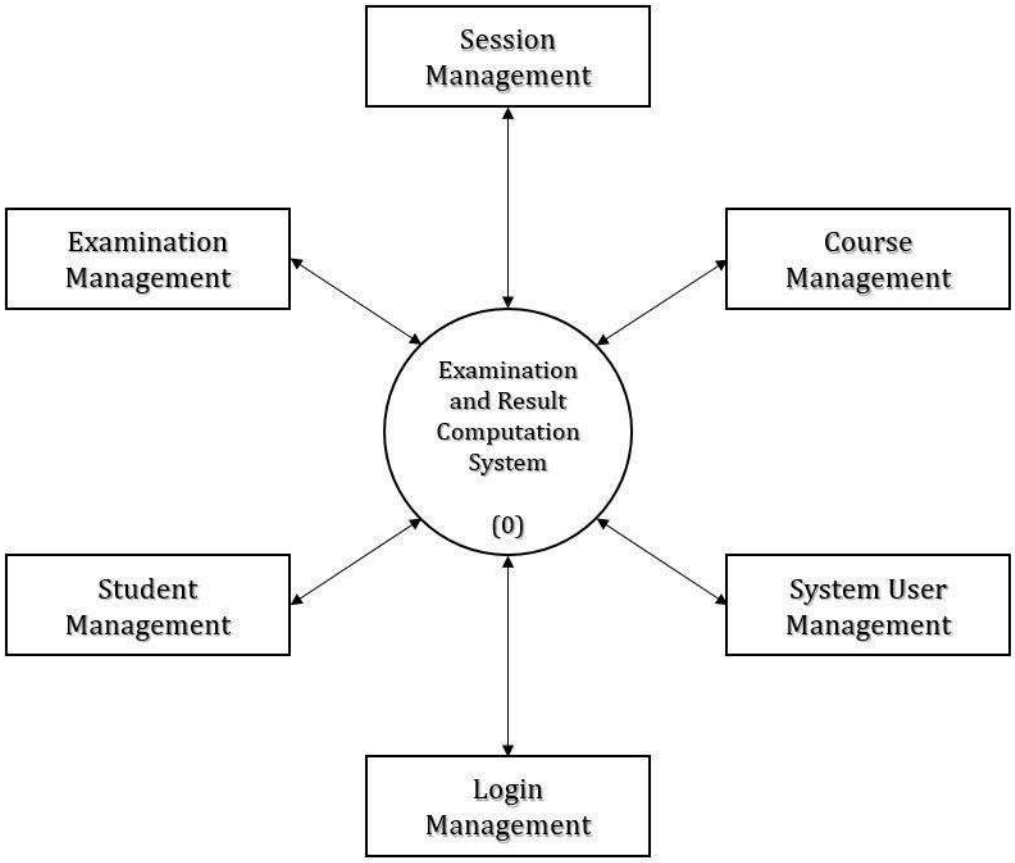


Fig – DFD level zero

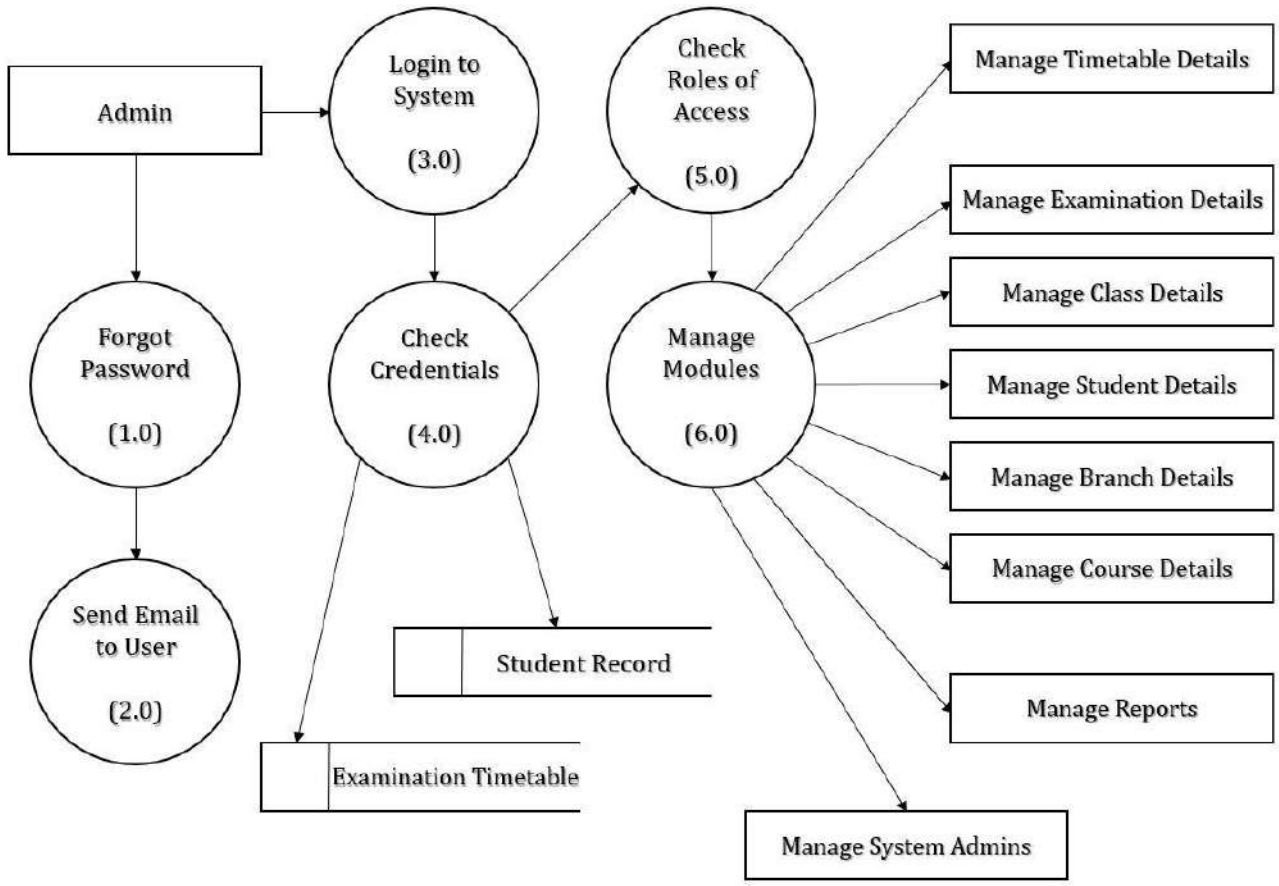
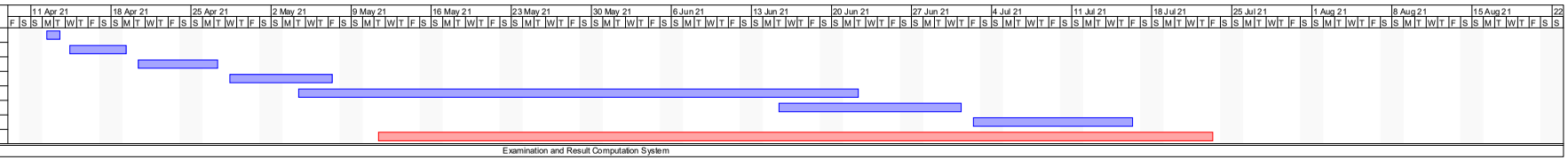


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing**– It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing**– It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing**– It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

PARKING ALLOCATION SYATEM

Problem Statement –Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model –Spiral development model

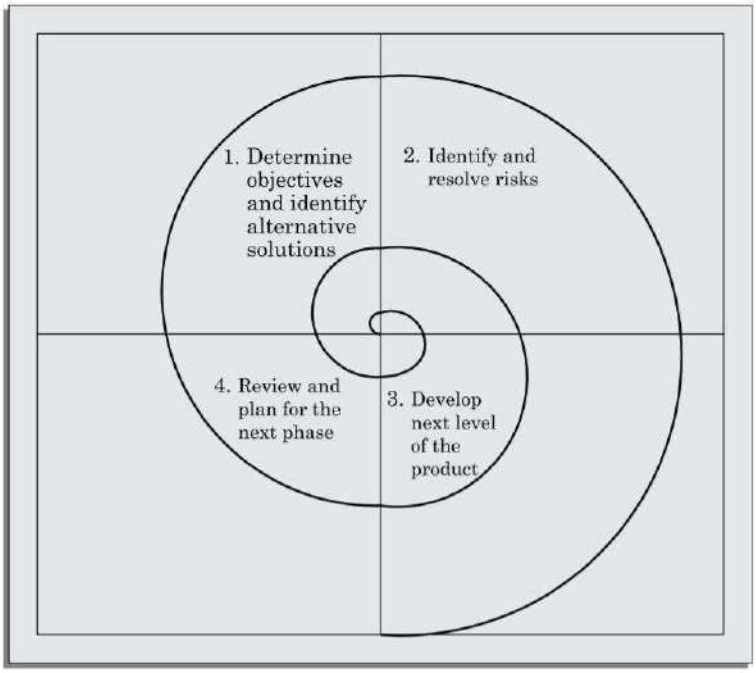


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module**–In this module, a customeris registered and the profile is maintained.

   Input:customerdetails
   Output: unique profile

2. **Vehicle Module**–In this module, the customer's vehicle is registered.

   Input:customer details, vehicle details
   Output:updated database

3. **Parking Module**–In this module,parking slot is allotted and fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input:customer details, vehicle details, duration
   Output:updated database

4. **Report Module**–In this module,parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input:parking slot details, vehicle details, parking fee details, duration details
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

Fig – DFD level zero

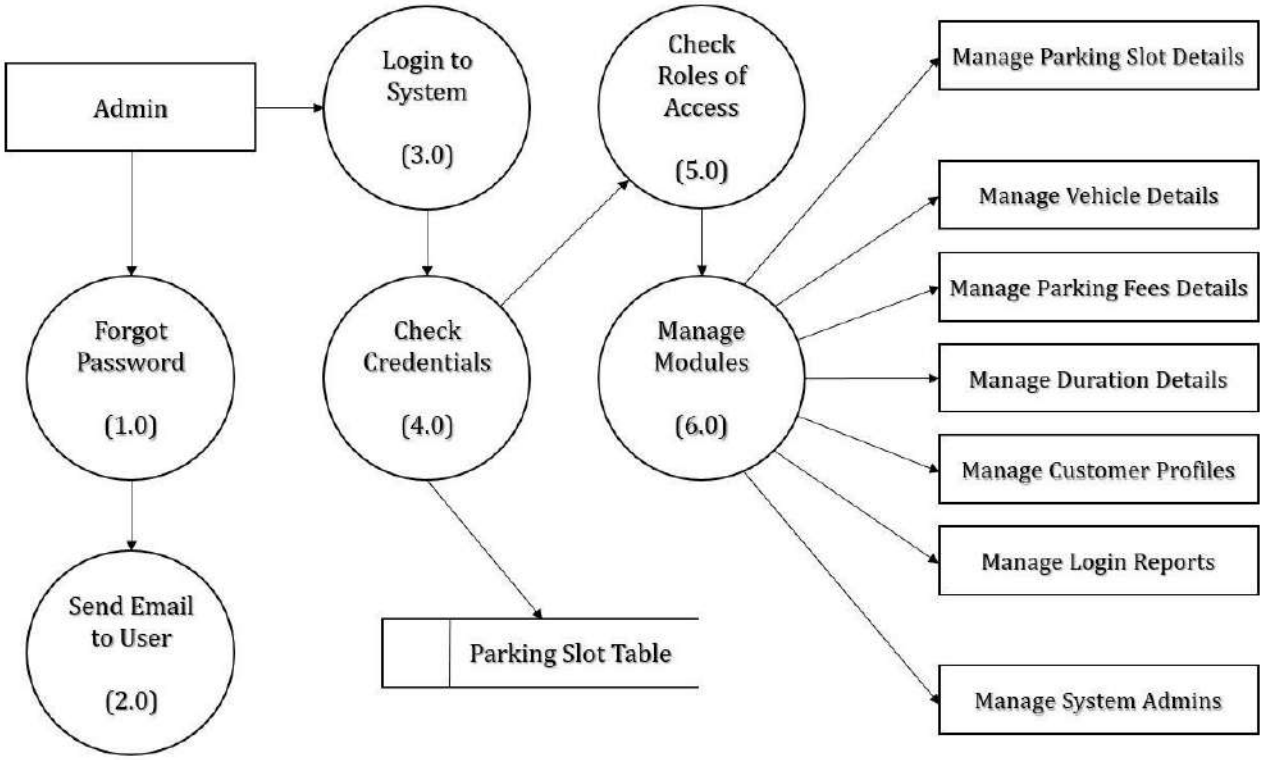
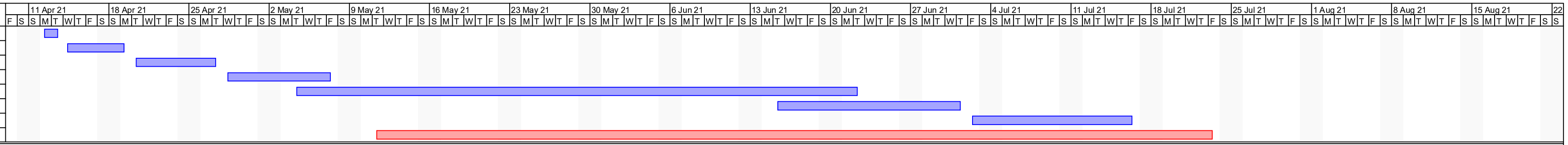


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Parking Allocation System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing–** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing–** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing–** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement –It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model –Spiral development model

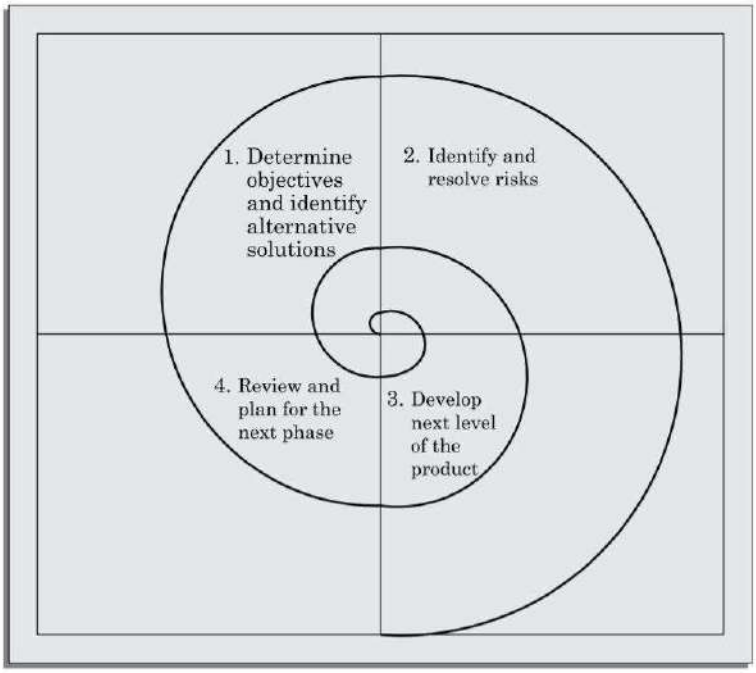


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module**–In this module, a customeris registered and the profile is maintained.

   Input:customerdetails
   Output: unique profile

2. **Stock Module**–In this module, the details of stocks are maintained.
   A database is maintained for keeping the stockinformation.

   Input:stockdetails
   Output:updated database

3. **Bill Module**–In this module,all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input:customer details, stock details
   Output:generated bills, updated database

4. **Report Module**–In this module,store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input:store details, product details, retail price details, bill details, billing counter details, customer details
   Output:generated reports

Design Engineering

Data Flow Diagrams (DFDs)

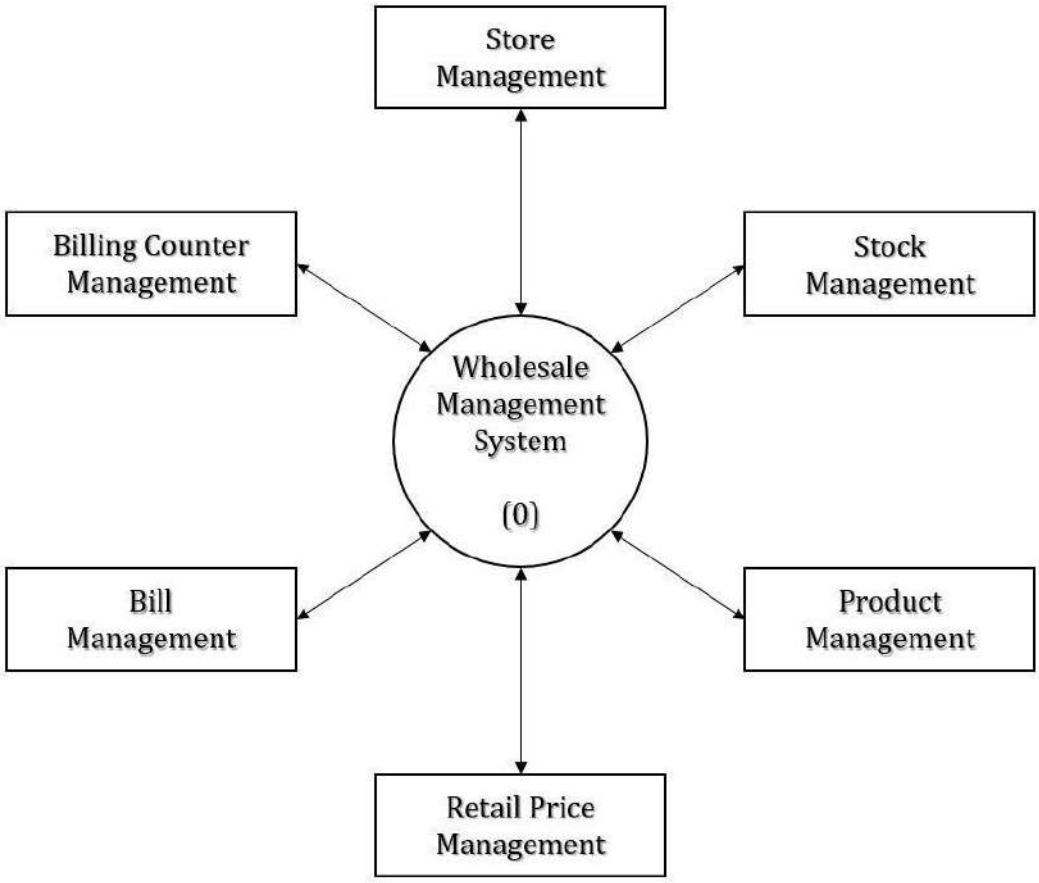


Fig – DFD level zero

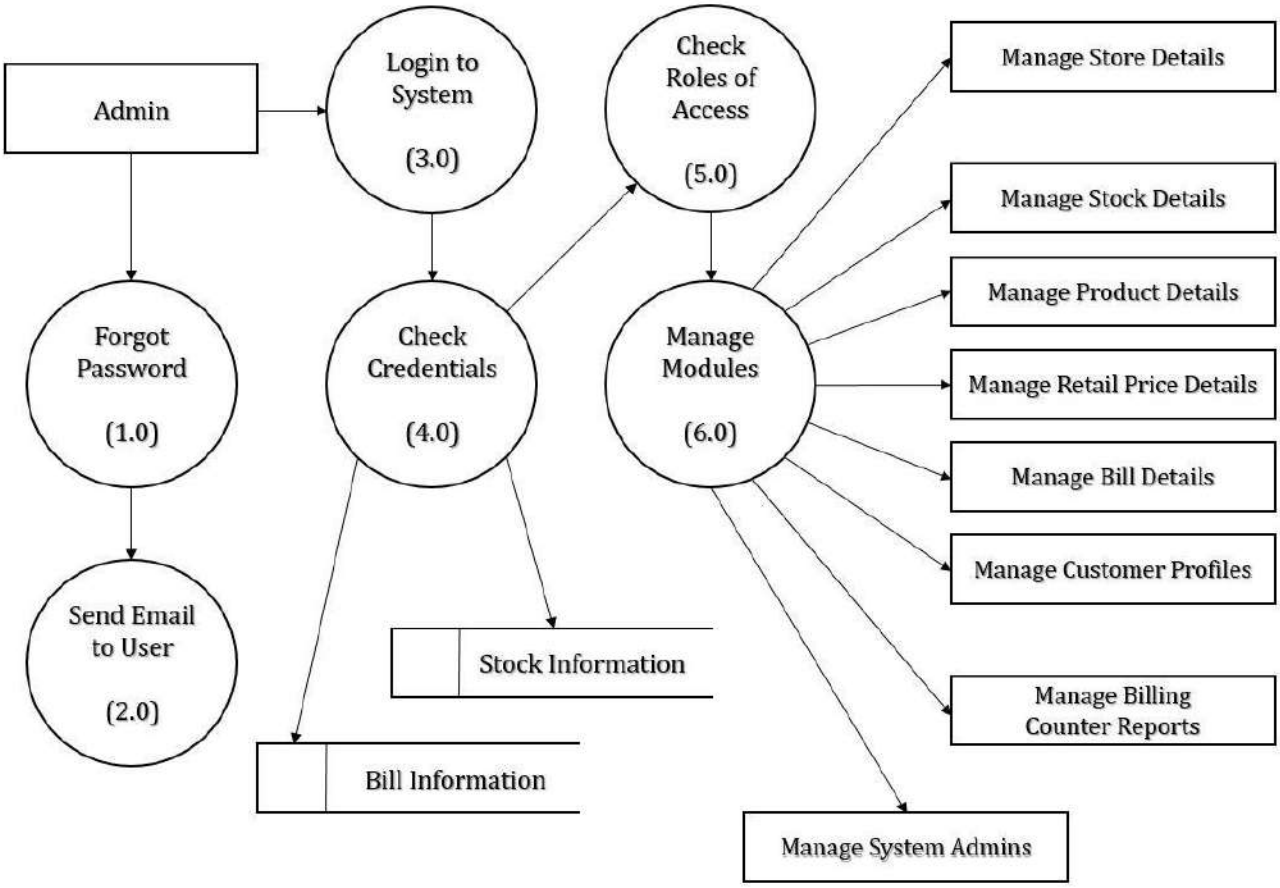
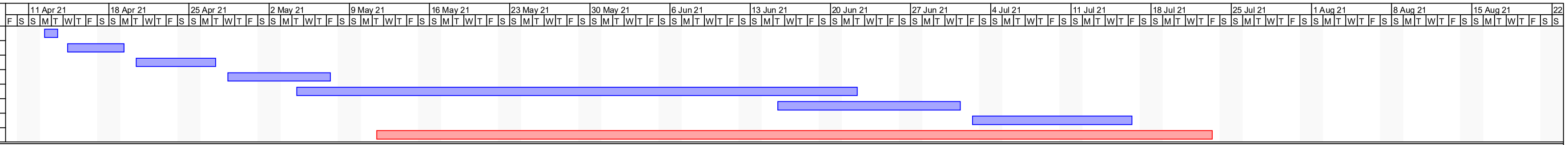


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Wholesale Management System

Risk Analysis

Risk Assessment–Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r– the probability of the risk becoming true

s– the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing–Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing– During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing–** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing–** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing–** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance–The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance**– Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance**– Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance**– Porting the software to work in a new environment.

EUCALYPTUS

Case Study

Name: Srijeet Ghosh
Dept.: Computer Science
Semester: V
Subject: Cloud Computing (DSE3)
Roll: 725
Reg.No.: A01-1152-117-019-2019

Supervisor's Name: Dr. Chayan Halder

CONTENTS

What is Cloud Computing?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end
users. Following are the working models of cloud Computing:

- ➢ Deployment Models
- ➢ Service Models

What are the types of Deployment Model?

Public Cloud

Public clouds are owned and operated by third-party cloud service providers, which deliver their computing resources, like servers and storage, over the Internet. Microsoft Azure is an example of a public cloud. With a public cloud, all hardware, software, and other supporting infrastructure is owned and managed by the cloud
provider. You access these services and manage your account using a web browser.

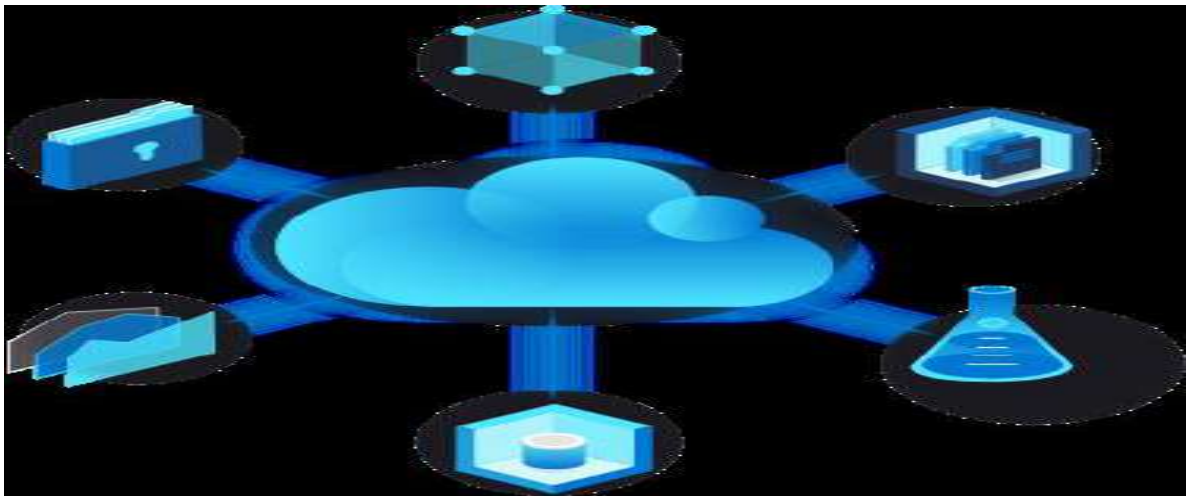
Private Cloud

Private cloud refers to cloud computing resources used exclusively by a single business or organization. A private cloud can be physically located on the company's on-site data centre. Some companies also pay third-party service providers to

host their private cloud. A private cloud is one in which the services and infrastructure are maintained on a private network.

Hybrid Cloud

Hybrid clouds combine public and private clouds, bound together by technology that allows data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives your business greater flexibility, more deployment options, and helps optimize your existing infrastructure, security, and compliance.



What are the types of Service Model?

Infrastructure as a service (IaaS)

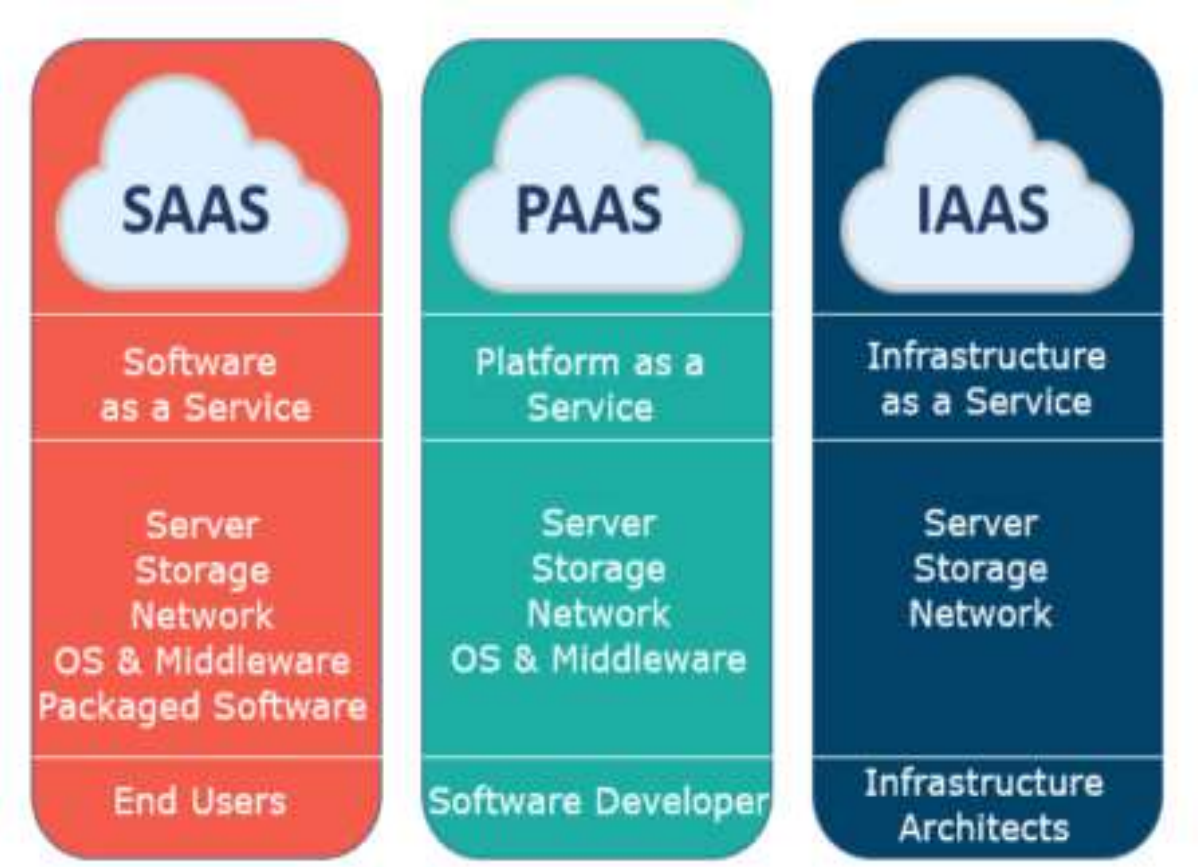
The most basic category of cloud computing services. With IaaS, you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

Platform as a service (PaaS)

Platform as a service refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps.

Software as a service (SaaS)

Software as a service is a method for delivering software applications over the Internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching.



What is Eucalyptus?

Eucalyptus in cloud computing is an open-source software platform for carrying out IaaS or Infrastructure-as-a-Service in a hybrid cloud computing or private cloud computing environment.

Eucalyptus in cloud computing pools together existing virtualised framework to make cloud resources for storage as a service, network as a service and infrastructure as a service. Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems is short known as Eucalyptus in cloud computing.

Eucalyptus in cloud computing frameworks declared a conventional concurrence with AWS or Amazon Web Services in March 2012, permitting overseers to move cases between an Amazon Elastic Compute Cloud and the Eucalyptus private cloud to make a hybrid cloud. The organisation

additionally permits Eucalyptus to work with Amazon’s product groups to create interesting Amazon Web Services viable highlights.

It tends to be effortlessly sent in existing IT frameworks to appreciate the advantages of both eucalyptus private cloud and eucalyptus public cloud models.

History of Eucalyptus

Improvement on Eucalyptus started as an examination project at US-based Rice University in the year 2003. In the year 2009, an organisation named Eucalyptus Systems was framed to market Eucalyptus software. Afterwards, in the year 2012, the firm went into a concurrence with Amazon Web Services for keeping up similarity and Application Programming Interface support. In the year 2014, it was procured by Hewlett-Packard or HP, which unexpectedly has its own cloud contributions under the HPE eucalyptus. The Helion portfolio has an assortment of cloud-related items, which incorporates HP's own kind of OpenStack called HP Helion OpenStack. Presently, Eucalyptus is a piece of the HPE portfolio and is known as HPE Helion Eucalyptus.

Eucalyptus architecture

Eucalyptus CLIs can oversee both Amazon Web Services and their own private occasions. Clients can undoubtedly relocate cases from Eucalyptus to Amazon Elastic Cloud. Network, storage, and compute are overseen by the virtualisation layer. Occurrences are isolated by hardware virtualisation. The following wording is utilised by Eucalyptus architecture in cloud computing.

1. Images: Any software application is work in, configuration, module software or framework software packaged and conveyed in the Eucalyptus cloud is known as a Eucalyptus Machine Image.

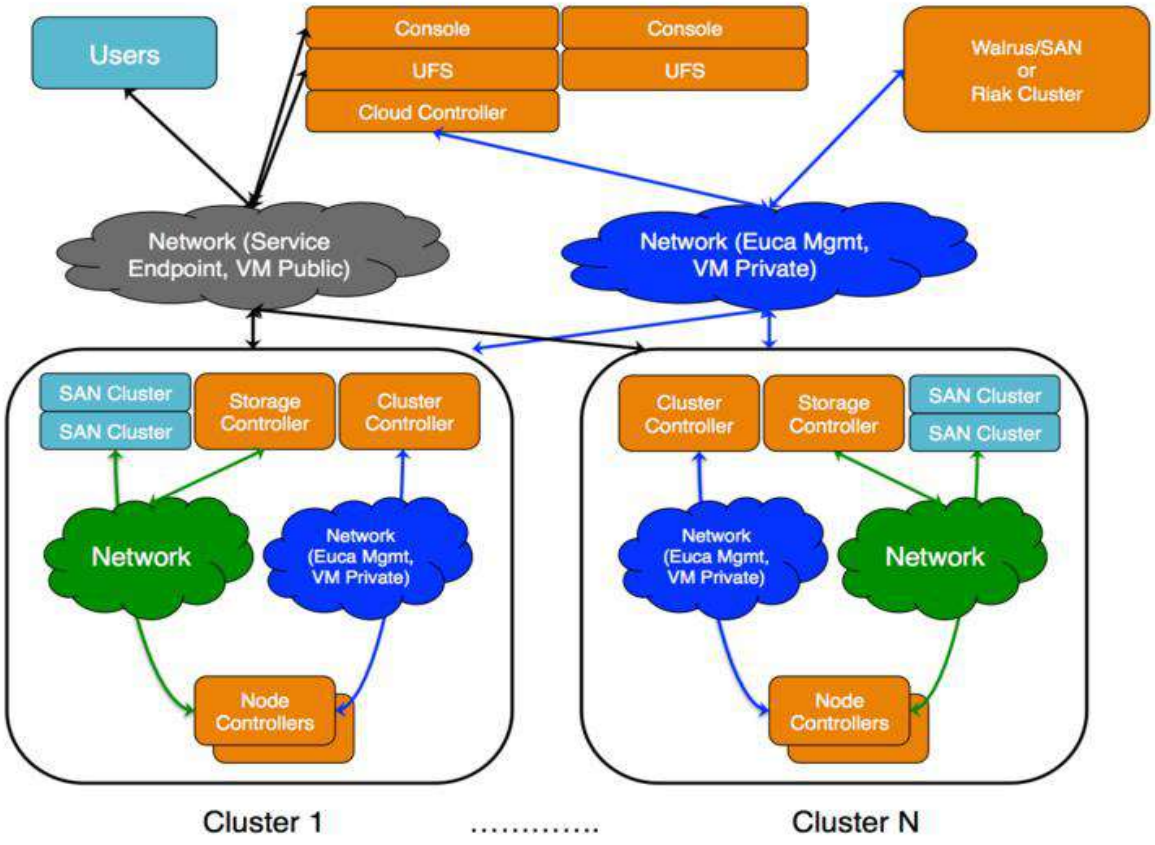
2. Instances: When we run the picture and utilise it, it turns into an instance.

3. Networking: The Eucalyptus network is to partitioned into three modes: Static mode, System mode, and Managed mode.

4. Access control: It is utilised to give limitation to clients.

5. Eucalyptus elastic block storage: It gives block-level storage volumes to connect to an instance.

6. Auto-scaling and load adjusting: It is utilised to make or obliterate cases or administrations dependent on necessities.



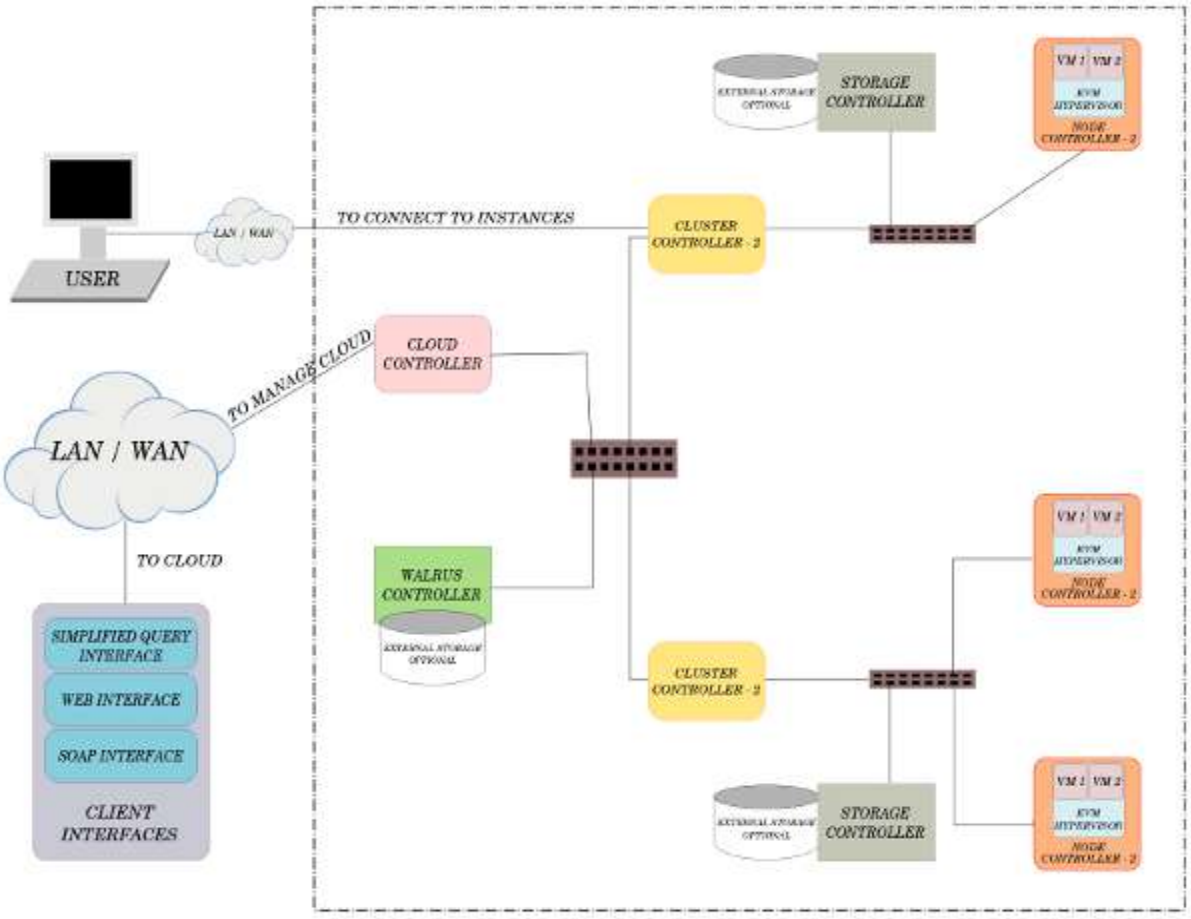
The advantages of the Eucalyptus cloud

The benefits of Eucalyptus in cloud computing are:

1. Eucalyptus can be utilised to benefit both the eucalyptus private cloud and the eucalyptus public cloud.
2. Clients can run Amazon or Eucalyptus machine pictures as examples on both clouds.
3. It isn’t extremely mainstream on the lookout yet is a solid contender to Cloud Stack and OpenStack.
4. It has 100% Application Programming Interface similarity with all the Amazon Web Services.
5. Eucalyptus can be utilised with DevOps apparatuses like Chef and Puppet.

What is the use of eucalyptus in cloud computing?

It is utilised to assemble hybrid, public and private clouds. It can likewise deliver your own data centre into a private cloud and permit you to stretch out the usefulness to numerous different organisations.



EUCALYPTUS BASED CLOUD

Conclusion

Eucalyptus in cloud computing is open-source programming that carries out an AWS viable cloud, which is financially savvy, secure and flexible. It tends to be effectively sent in existing IT frameworks to appreciate both private and public cloud models' advantages.

References

↞https://docs.eucalyptus.cloud/eucalyptus/4.4.4/install-guide/euca_architecture.html

↞https://www.jigsawacademy.com/blogs/cloud-computing/eucalyptus-in-cloud-computing/amp/

↞https://www.wikipedia.org

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Prabal Roy
Roll Number: 727
Exam roll number: 2021141304
Registration number: A01-1152-117- 020-2019
Semester: V

Department: Computer Science

Supervisor: Prof. Manas Pal

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a “Customer Number” (CN). A customer can present ‘CN’ to customer staff when he makes purchases. In this case, the value of his purchase is credited against its ‘CN’. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

REQUIREMENT ANALYSIS (SRS) :

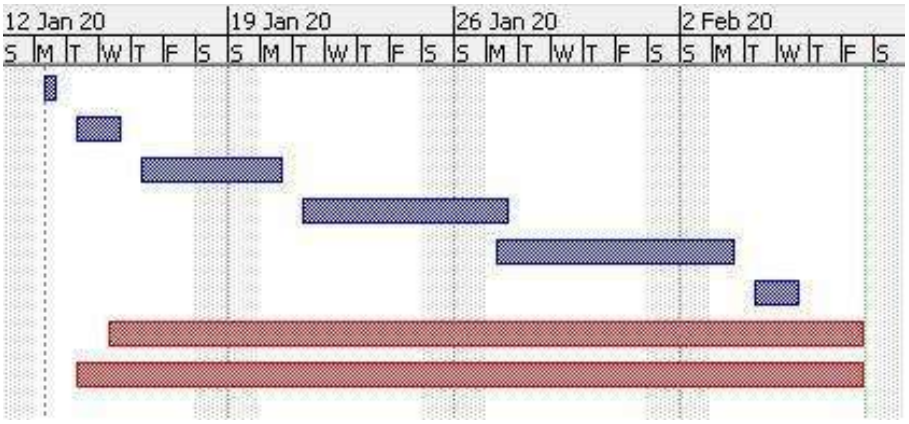
As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account’s purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh

the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



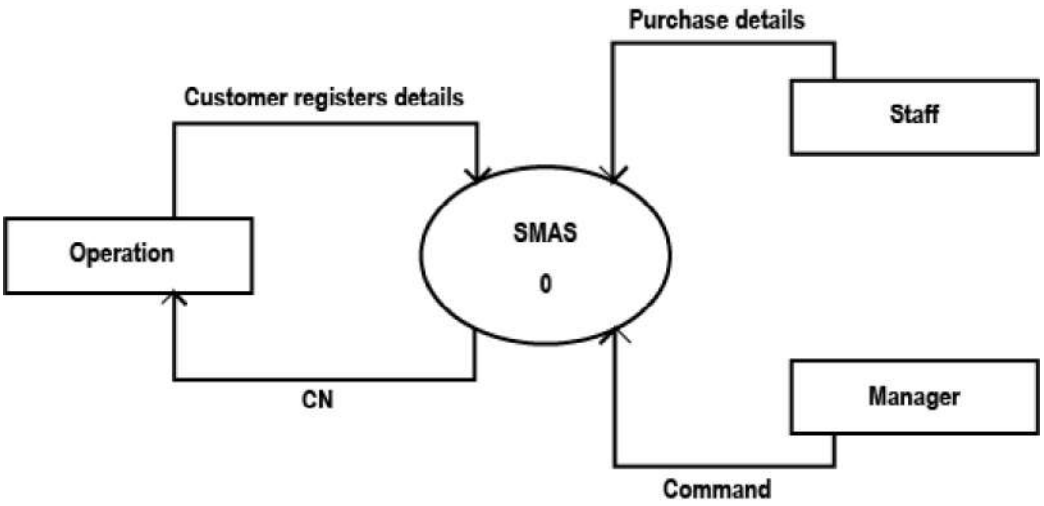
- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.

○ Risk leverage =

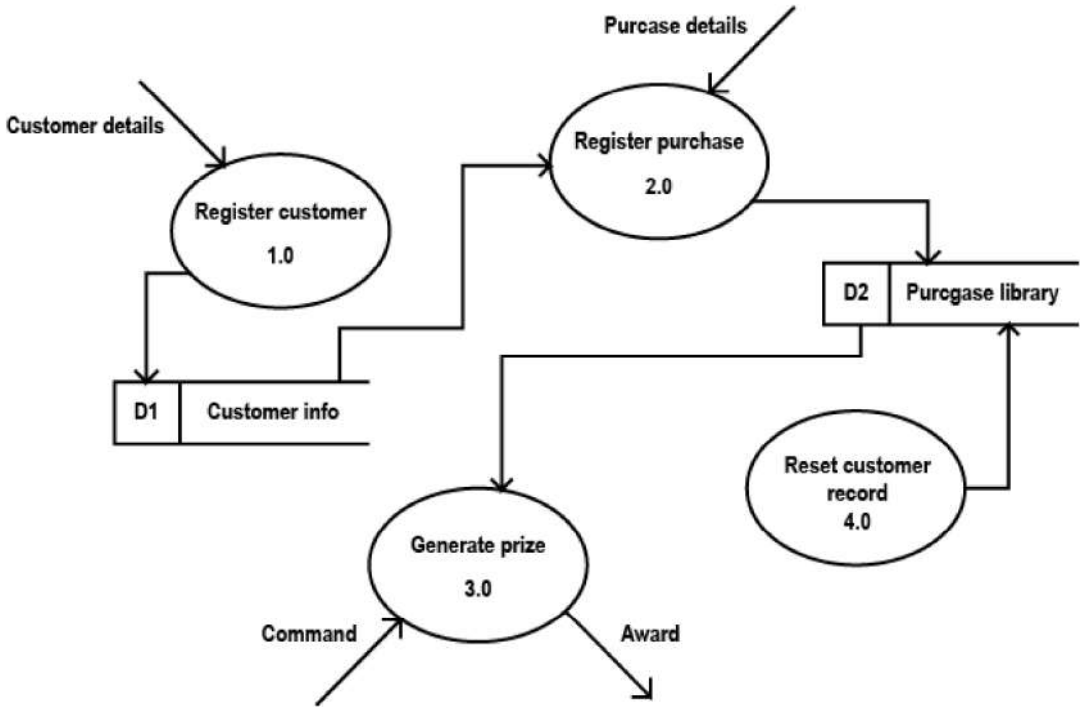
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

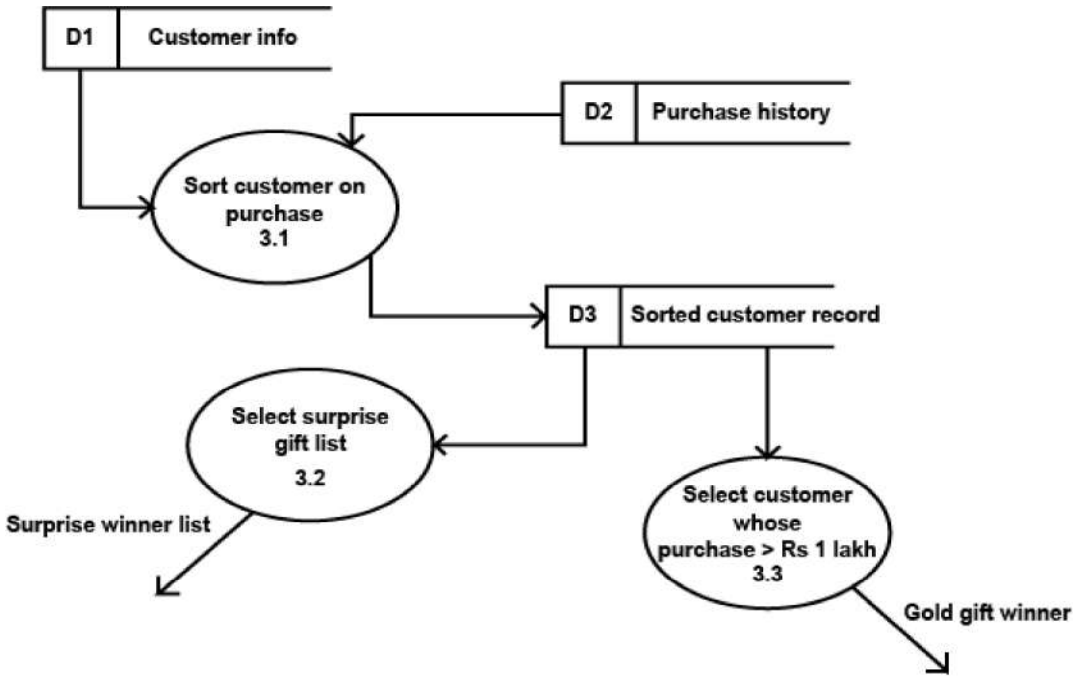
1. Level 0 DFD :



3. Level 1 DFD :



4. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

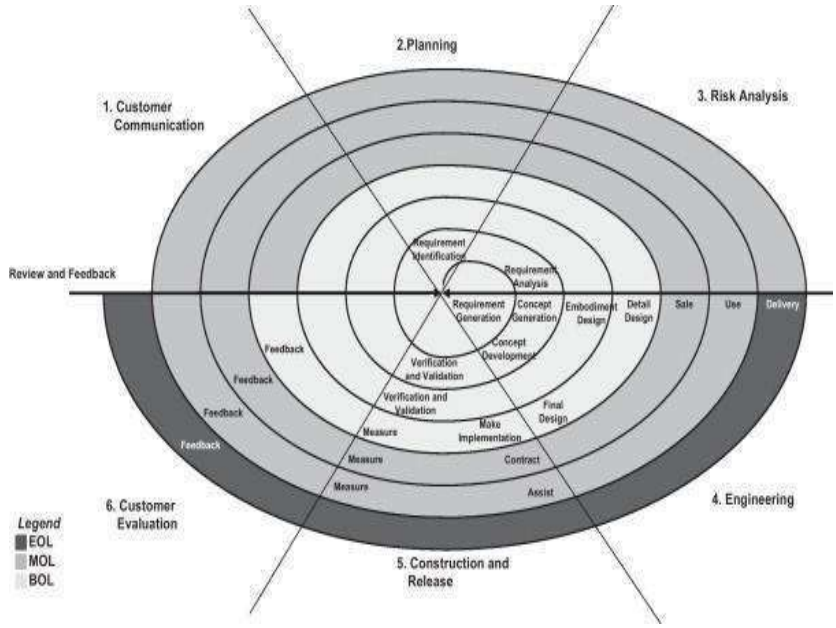
PROJECT 2

Route Management System

It identifies internal datastores of payment, bus route,

customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule.It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. ***Goal of the Project:***

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. ***Functional Requirements:***

As per the requirements of the software from client end. By analysis we realise that we need five modules: A.

Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Customer Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. Schedule Module:

This module maintains timings of the bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

---

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

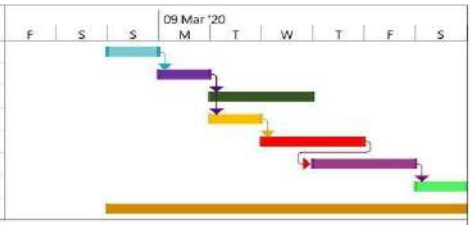
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 08-03-20	Sun 08-03-20	
2	Customer Communication	1 day	Mon 09-03-20	Mon 09-03-20	1
3	Planning	2 days	Tue 10-03-20	Wed 11-03-20	2
4	Risk Analysis	1 day	Tue 10-03-20	Tue 10-03-20	2
5	Engineering	2 days	Wed 11-03-20	Thu 12-03-20	4
6	Construction and Release	2 days	Thu 12-03-20	Fri 13-03-20	5
7	Customer Evaluation	1 day	Sat 14-03-20	Sat 14-03-20	6
8	Write User Manual	7 days	Sun 08-03-20	Sat 14-03-20	



Project: Project8
Date: Sat 14-03-20

Task, Split, Milestone, Summary, Project Summary, Inactive Task, Inactive Milestone
Inactive Summary, Manual Task, Duration-only, Manual Summary Rollup, Manual Summary, Start-only, Finish-only
External Tasks, External Milestone, Deadline, Progress, Manual Progress

b. Project Resources

a. **ManPower Resources:**
   i. **Project Manager**
   ii. **System Analyst**
   iii. **Designer** iv. **Developer** v. **Tester**
vi. **Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above**
   ii. Ram: 4Gb or above
   iii. **Java Development Kit -**
   13 iv. **Java Runtime**
   Environment v. **Sublime**
   Text 3 (Text Editor)
   vi. **Ms - Project**
   vii. **Creately** viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. **Likelihood of a risk coming through**
B. **Consequence of the problem associated with that risk**

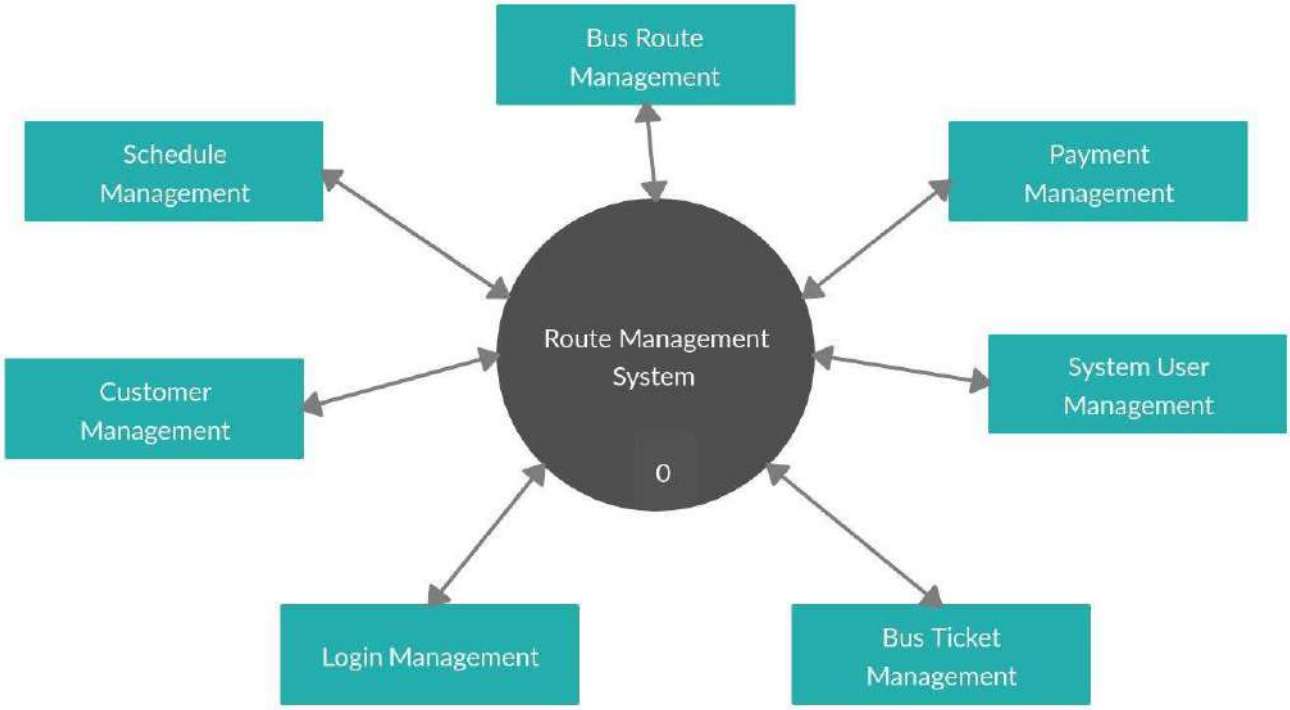
Priority of each risk is multiplication of above two

Risk Leverage =

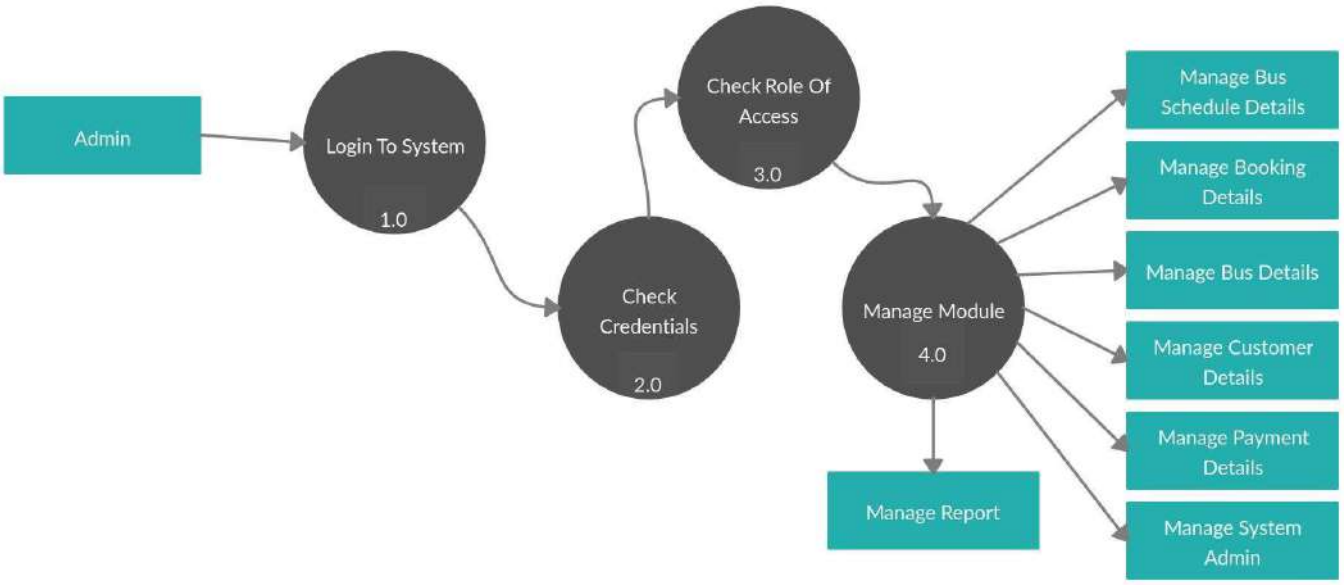
$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduct}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. **Architectural Design**



2. **Component Level Design**
   - **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing: Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance
4. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

    ○ Doctor module : This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.

    ○ Report module : This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.

    ○ Diagnostic module : This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.

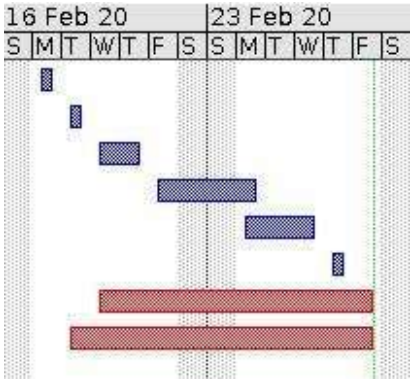
    ○ Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

    ○ Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

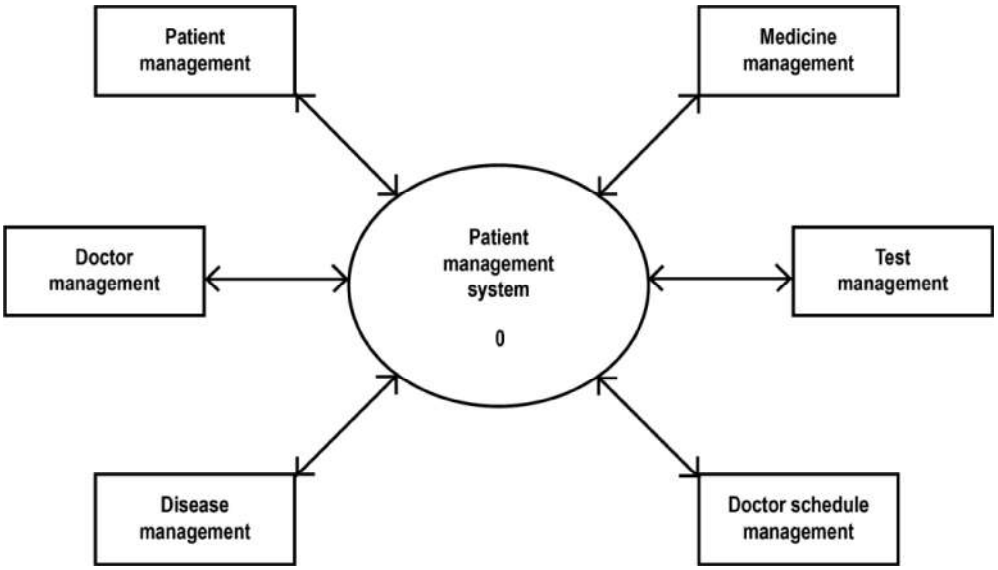
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

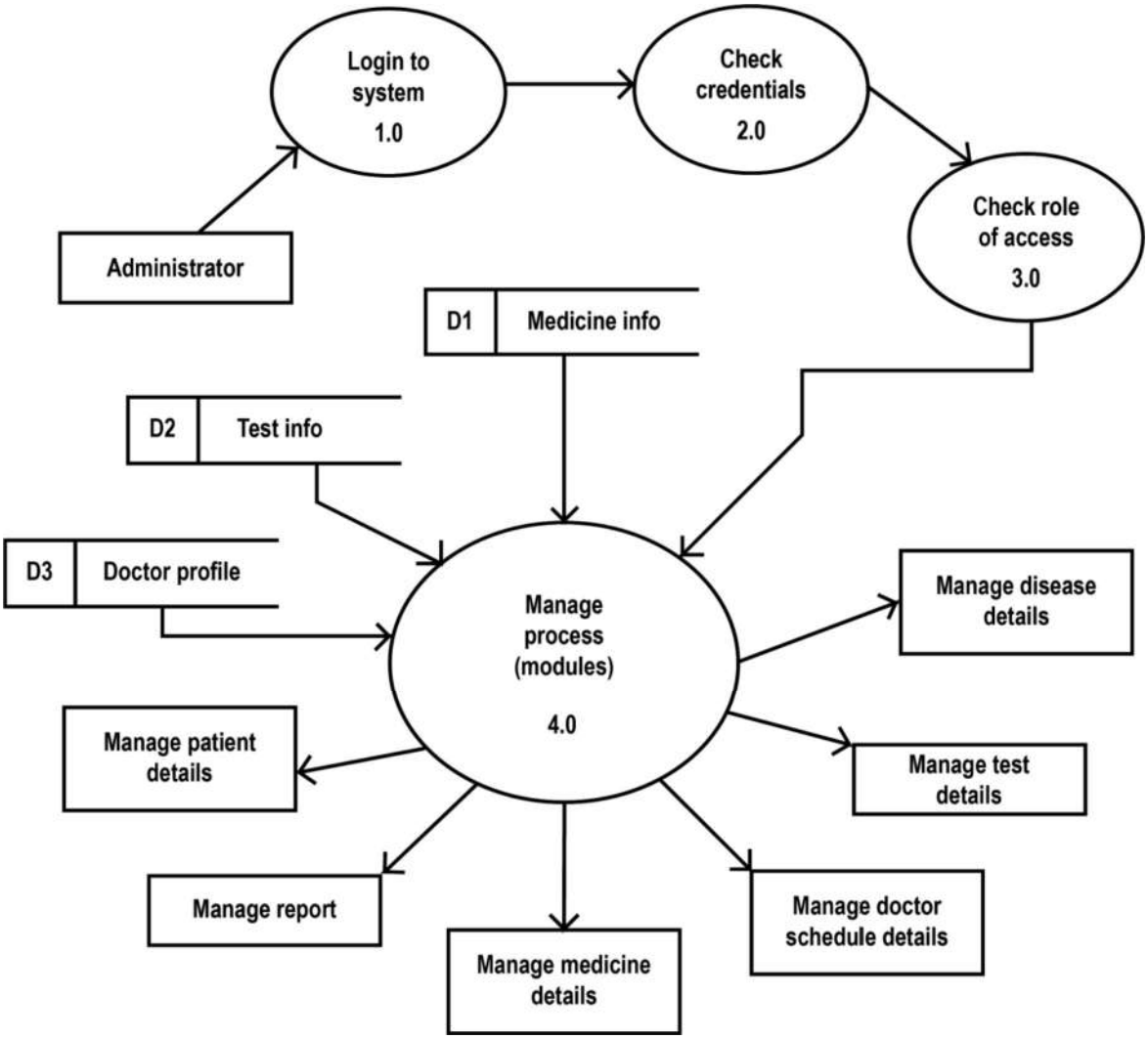
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
        - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.

- Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. Guest module : In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. Stay module : In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

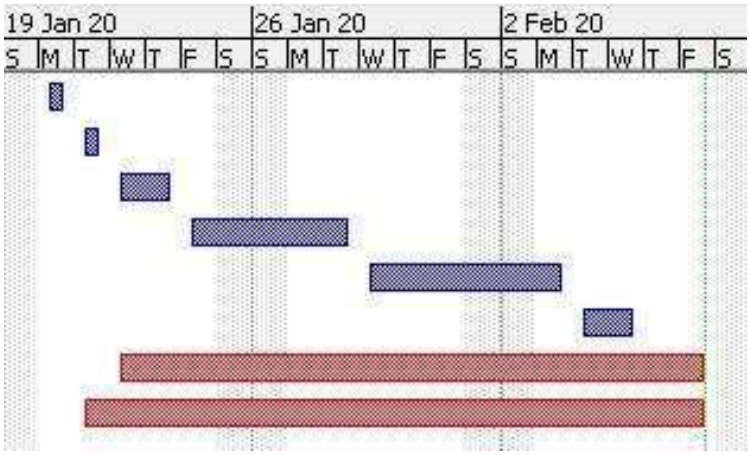
3. Catering module : In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.

4. Bill generation module : At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2		Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3		Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4		Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5		Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



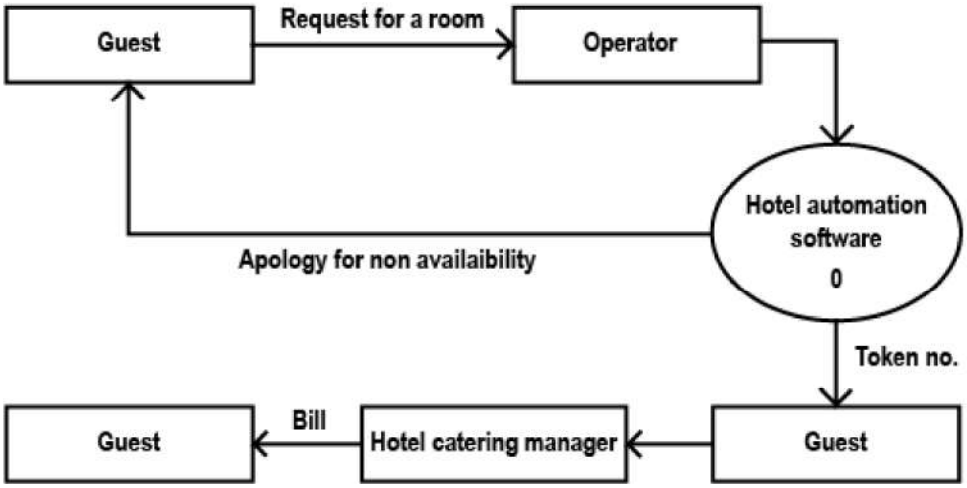
- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

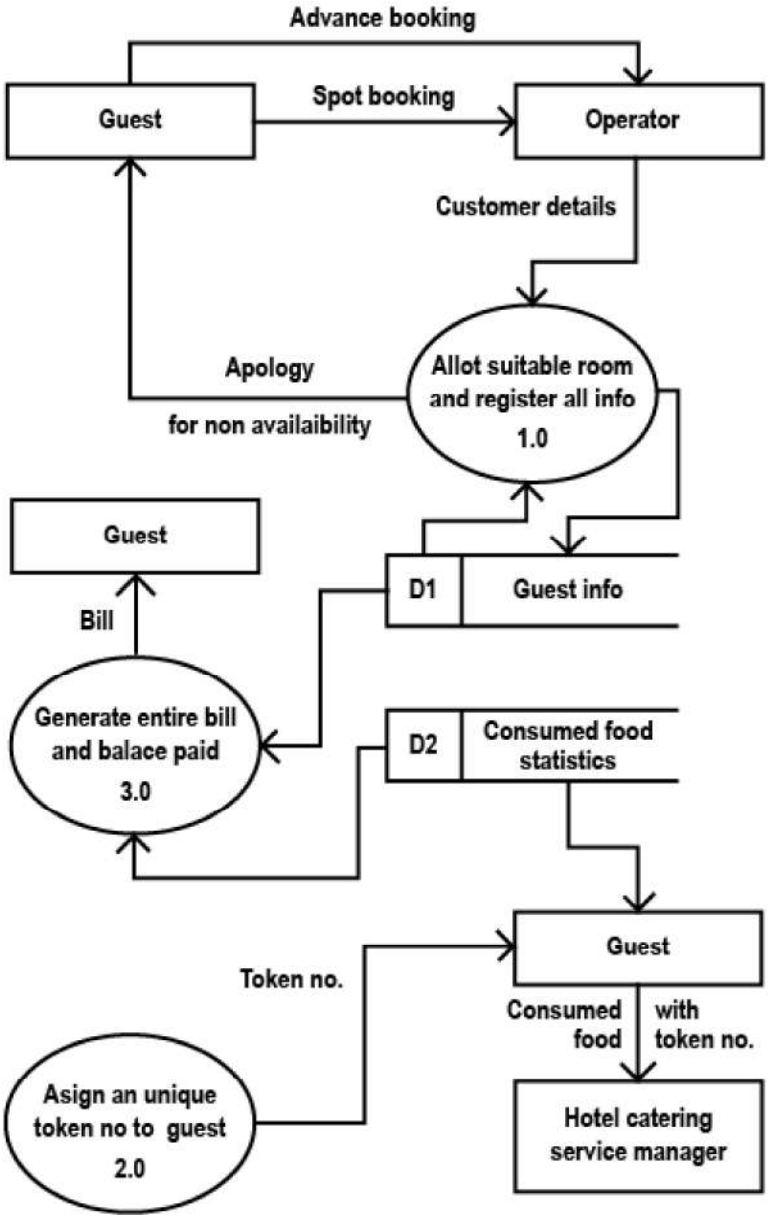
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.

    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Road maintenance is the performance of adaptive, preventive and corrective maintenance for a given

road in a locality. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

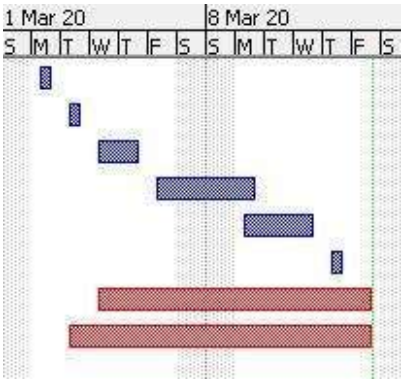
- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
    - Criminal module : This module accesses data from the database of criminal record details and manages it properly.
    - FIR module : This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
    - Chargesheet module : This module accesses data from the database of chargesheet record details and manages it properly.
    - Court module : The main function of this module is to manage the court profiles.

- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
    - Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



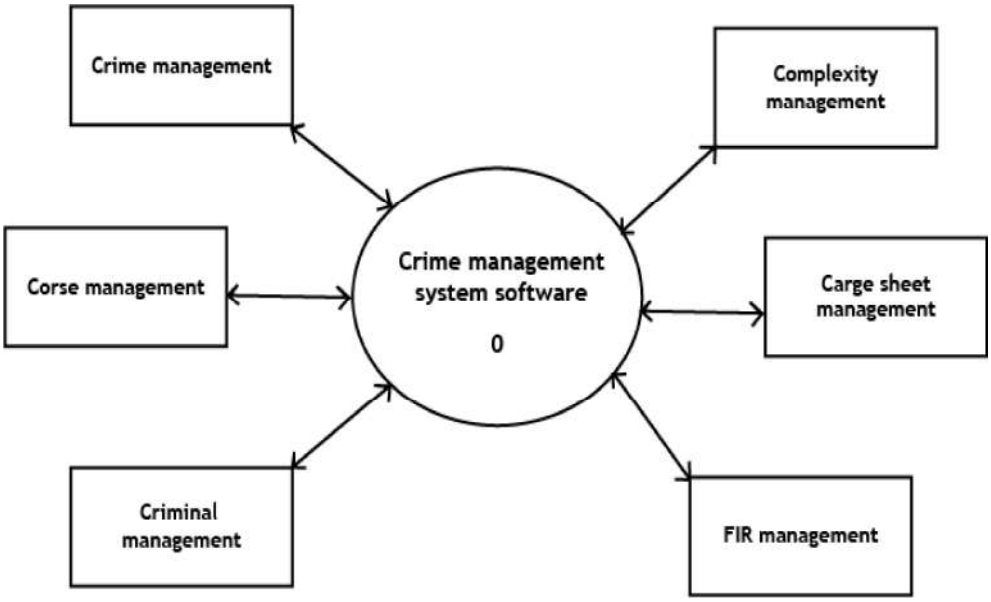
- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
    - Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{}$$

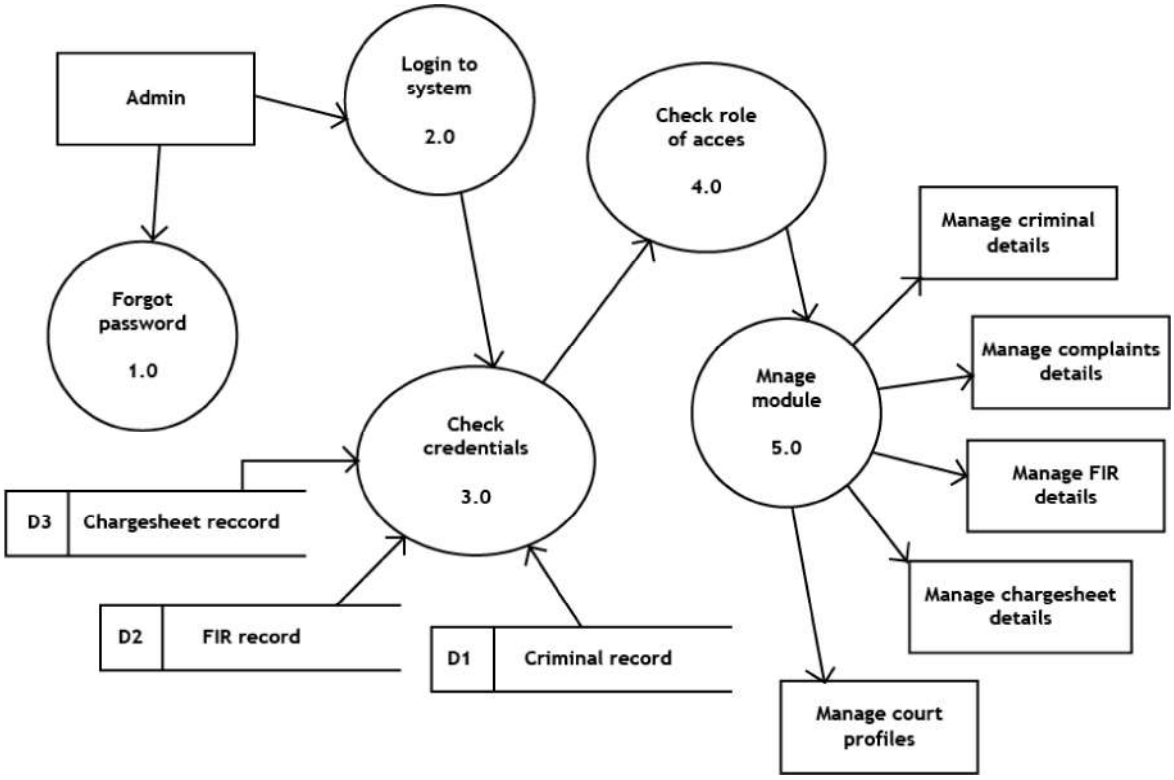
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT :

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

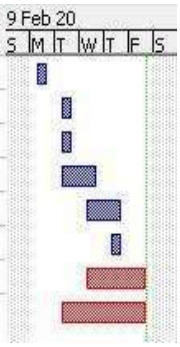
- Goal of the project : The aim and objectives for students' examination
  Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements : The three modules needed are.
  - Login and recovery module : This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
  - Credentials module : This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
  - Administrator module : This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.
- Non-Functional Requirements :
  - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

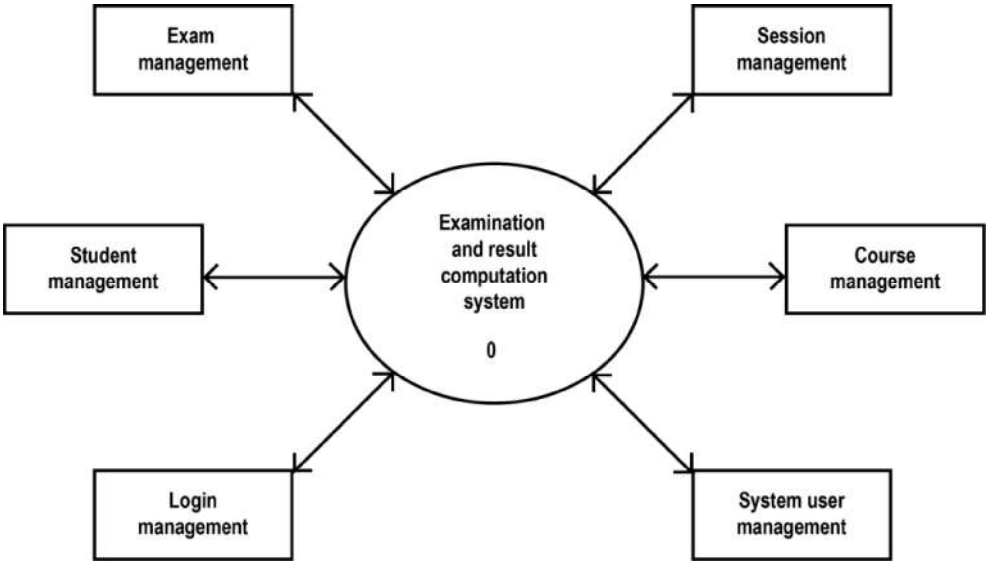


- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
    - Risk leverage =

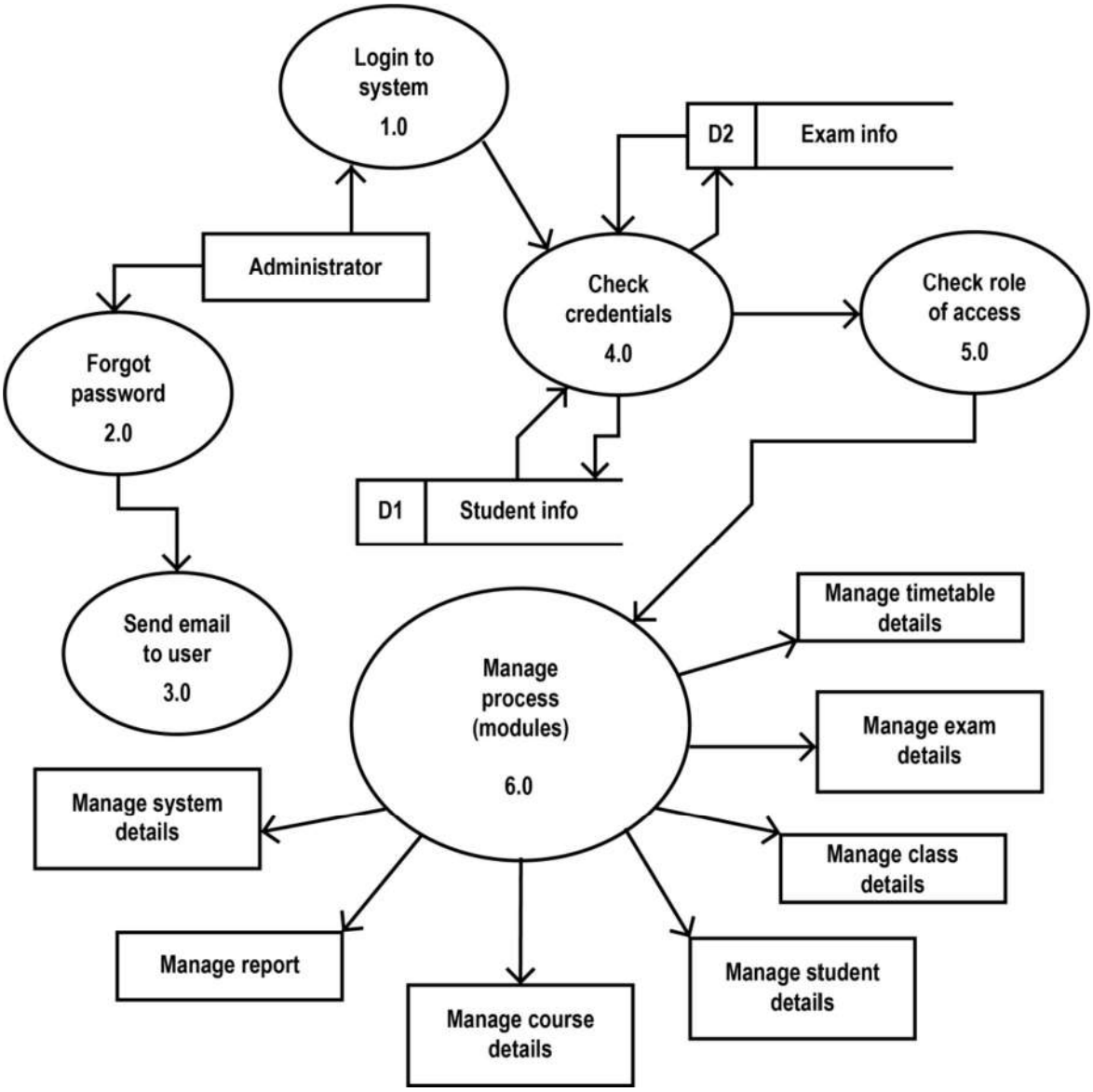
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :


Exam management
Session management
Examination and result computation system
0
Student management
Course management
Login management
System user management

2. Level 1 DFD :


Login to system
1.0
D2
Exam info
Administrator
Check credentials
4.0
Check role of access
5.0
Forgot password
2.0
D1
Student info
Send email to user
3.0
Manage process (modules)
6.0
Manage timetable details
Manage exam details
Manage system details
Manage class details
Manage report
Manage course details
Manage student details

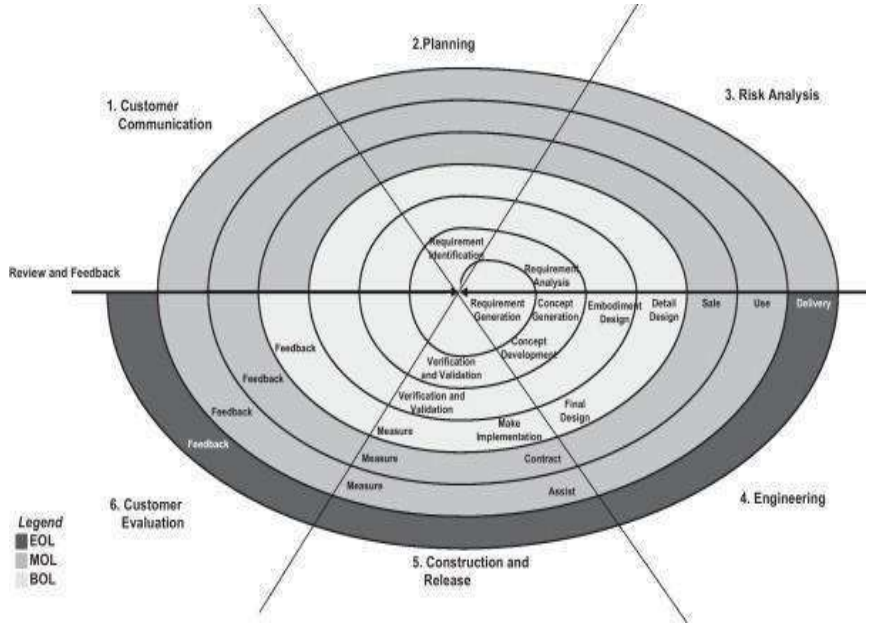
Project 7

Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - ★ **Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
  - ★ **Spiral Model consists of Risk Analysis task region.**
  - ★ **In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications** ***1. Goal of the Project:***

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc which was a hefty work.

2. Functional Requirements:

As per the requirements of the software from client end.

By analysis we realise that we need three modules:

A. Admin Module:

This module asks username and password as input and if the

input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.
Input: Answer to security question
Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. Parking Module:

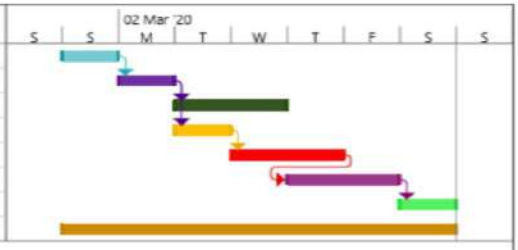
This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.
B. Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.
C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



Project: Project7
Date: Fri 13-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

a. Project Scheduling

b. Project Resources

a. ManPower Resources:

i. Project Manager

ii. System Analyst

iii. Designer iv. Developer v. Tester

vi. Document Writer

b. Hardware-Software Resources

i. Processor: Intel i3 4th gen or above

ii. Ram: 4Gb or above

iii. Java Development Kit - 13 iv. Java Runtime Environment v. Sublime Text 3 (Text Editor)

vi. **Ms - Project**

vii. **Creately** viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk

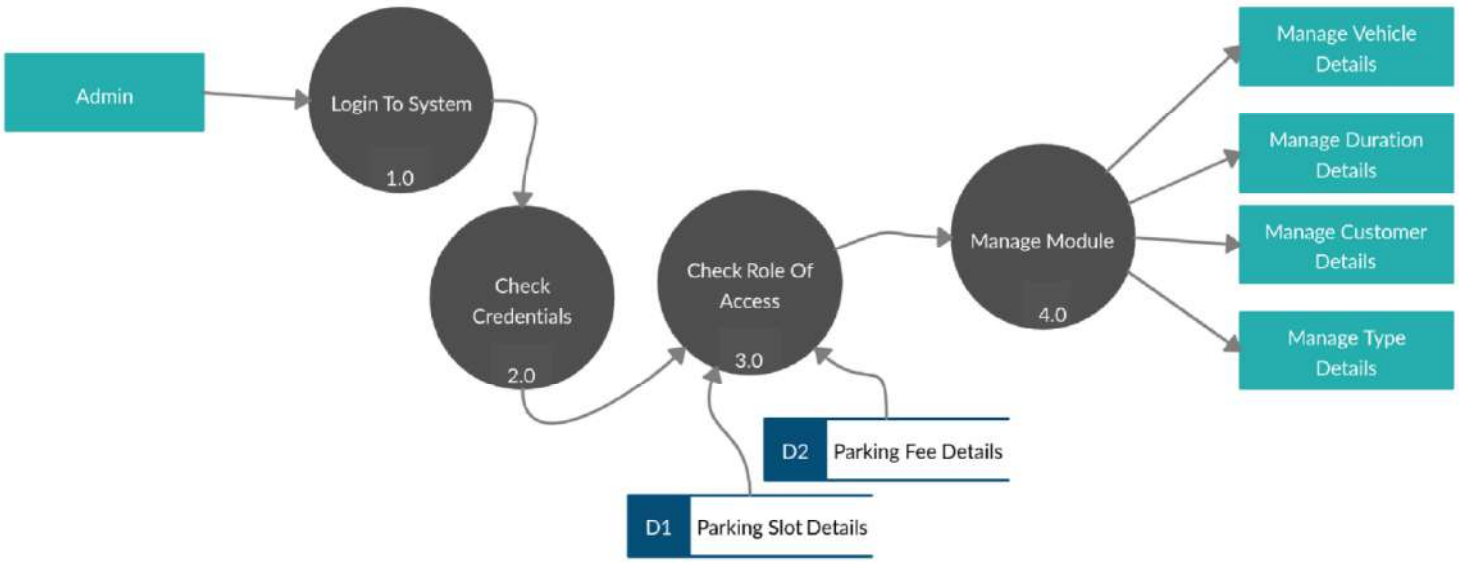
Priority of each risk is multiplication of above two

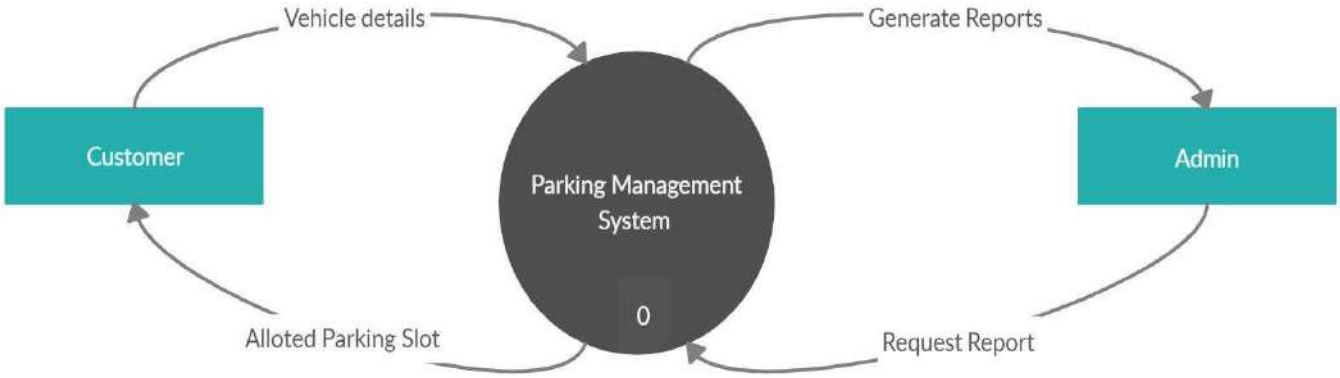
Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. Architectural Design





2. **Component Level Design**

   - **Level 1 DFD**

TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1.Unit Testing:

Each module is tested in isolation **2. System Testing:**

The modules are integrated and again tested. This time this testing is done in three parts

A. Alpha testing(Verification) : done by Development team
B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

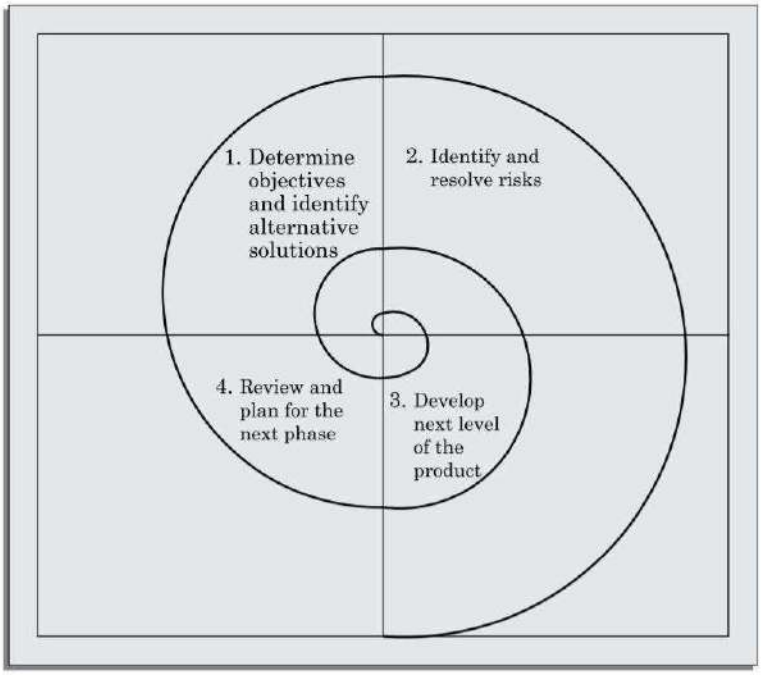


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Customer Module – In this module, a customer is registered and the profile is Maintained.

Input: customer details
Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained. A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

Report Module – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

Input: store details, product details, retail price details, bill details, billing counter details, customer details
Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

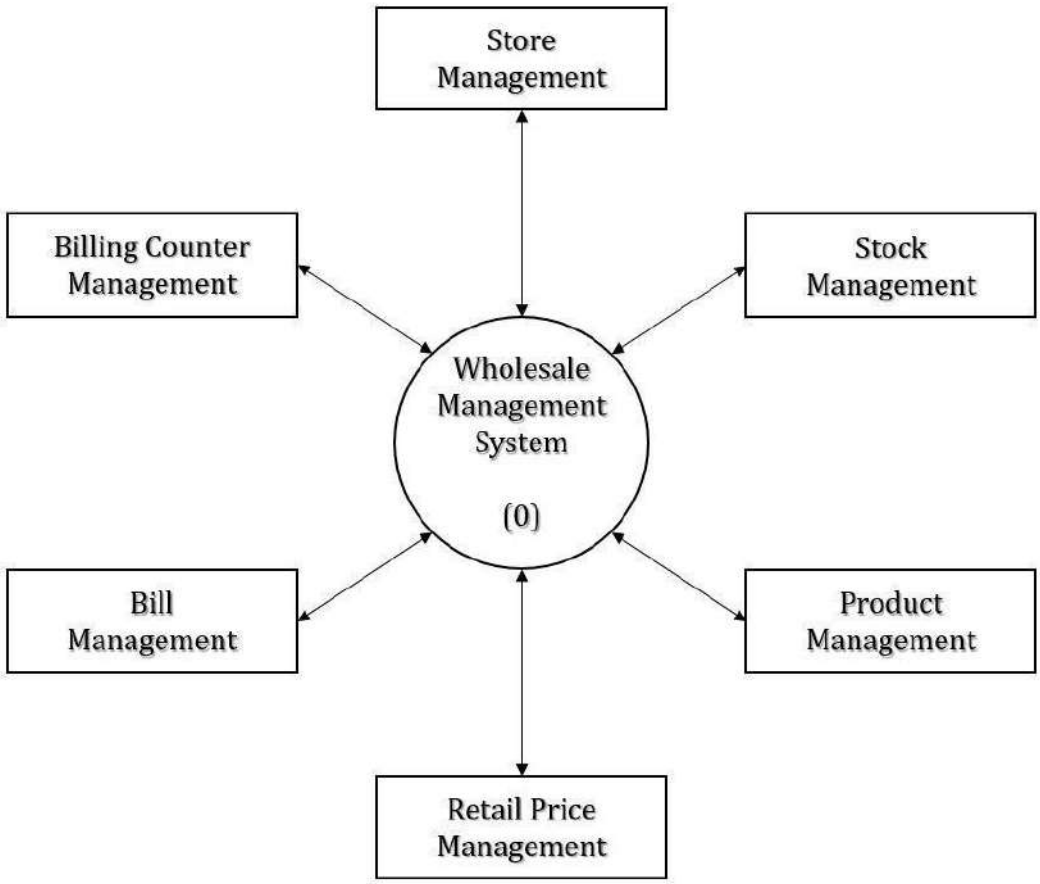


Fig – DFD level zero

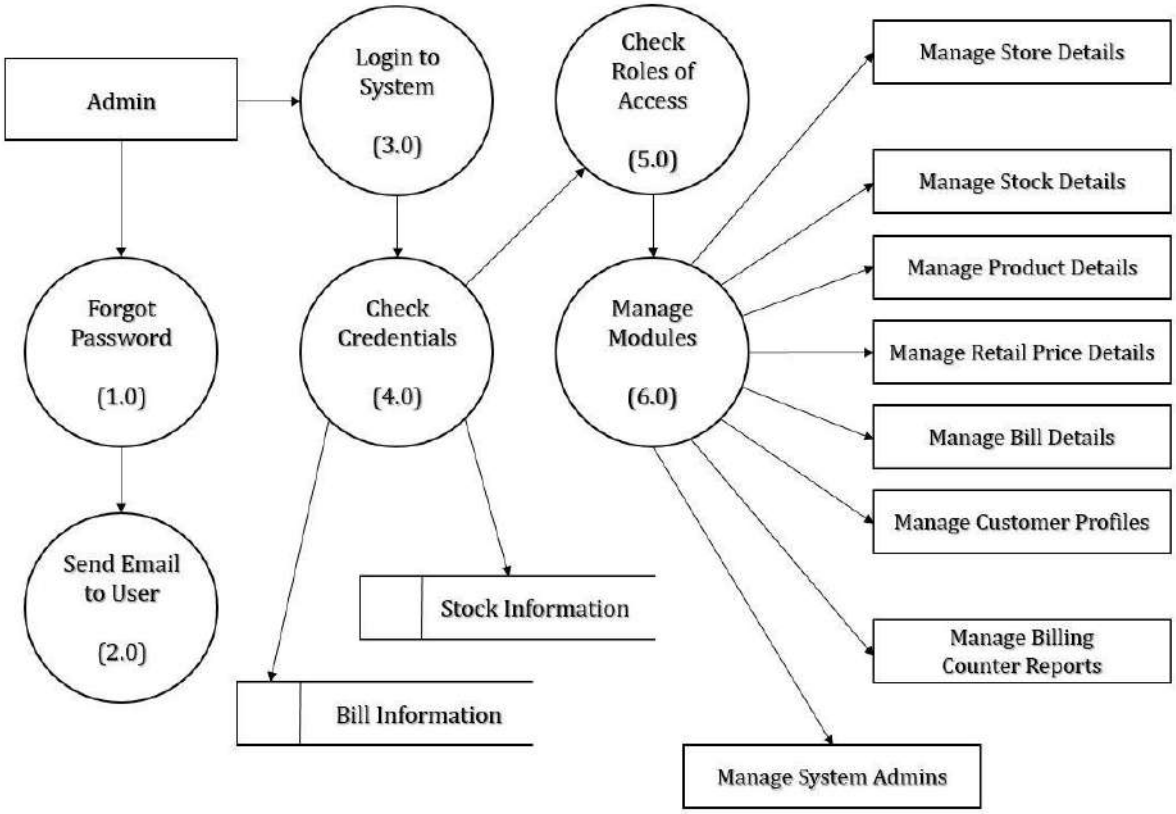
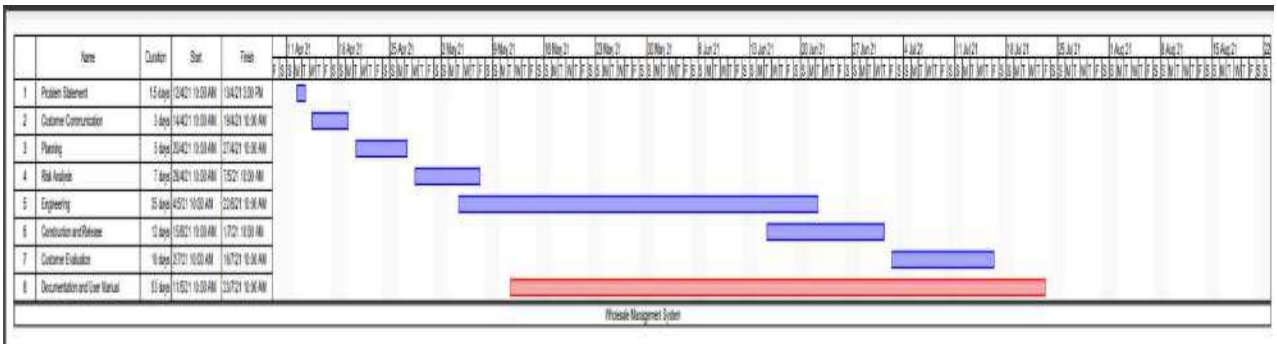


Fig – DFD level one

Project Management

Project Schedule –



Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

The likelihood of a risk coming true (**r**)

The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

risk leverage= risk exposure before reduction-risk exposure after reductioncost of reduction

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

α (Alpha) Testing – It is the system testing performed by the development team in a controlled environment.

ii. **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii. **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i. **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii. **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii. **Adaptive maintenance –** Porting the software to work in a new environment.

Name: DEBENDU BARMAN

Department: Computer Science

Semester: V

Subject: Cloud Computing (DSE III)

Roll: 728

Reg: A01-1122-117-021-2019

Case Study on

AWS Elastic

Cloud Computing

Date Created: 4-01-2021

Supervisor's Name: BB SIR

Contents

What is Cloud Computing?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end users. Following are the working models of cloud computing:

- Deployment Model
- Service Model

Deployment Models

Public Cloud

Public cloud is an IT model where on-demand computing services and infrastructure are managed by a third-party provider and shared with multiple organizations using the public internet. Public cloud makes computing resources available to anyone for purchase. Multiple users typically share the use of a public cloud.

Private Cloud

Private cloud provides a high level of security and privacy to data through firewalls and internal hosting. It also ensures that operational and sensitive data are not accessible to third-party providers. HP Data Centers, Microsoft, Elastra-private cloud, and Ubuntu are the example of a private cloud.

Hybrid Cloud

Hybrid cloud refers to a mixed computing, storage, and services environment made up of on-premises infrastructure, private cloud services, and a public cloud—such as Amazon Web Services (AWS) or Microsoft Azure—with orchestration among the various platforms.

Service Models

Infrastructure as a Service (IaaS)

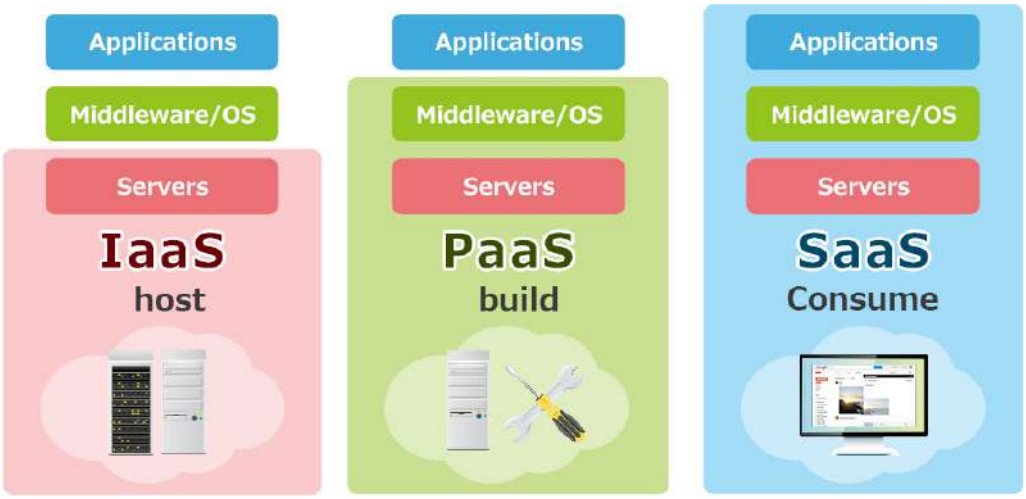
Infrastructure as a service are online services that provide high-level APIs used to dereference various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc.

Platform as a Service (PaaS)

Platform as a service (PaaS) is a cloud computing model where a third-party provider delivers hardware and software tools to users over the internet. As a result, PaaS frees developers from having to install in-house hardware and software to develop or run a new application. For e.g. Google App Engine

Software as a Service (SaaS)

Software as a service (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.



What is Amazon EC2?

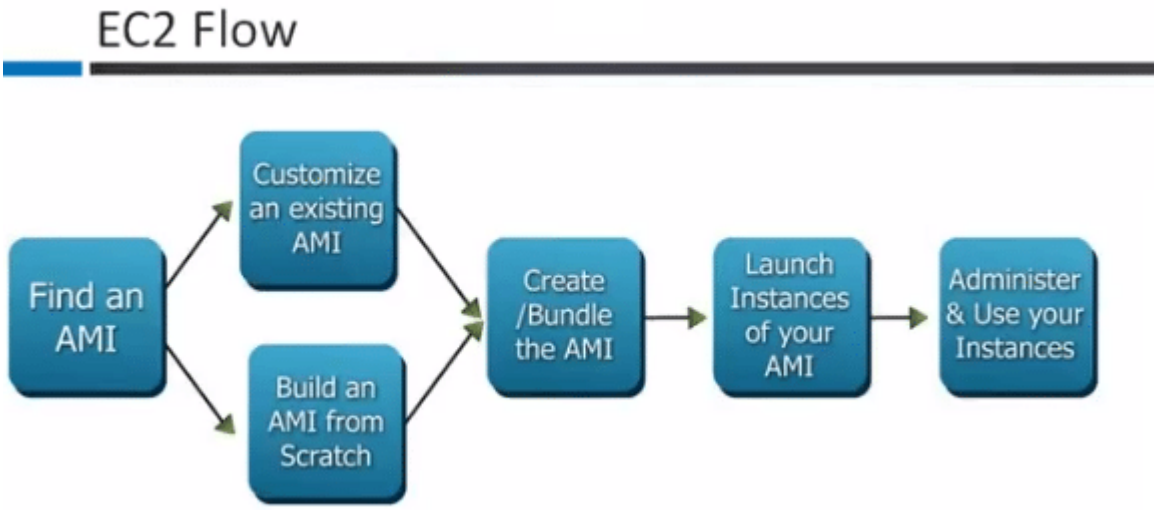
Amazon Elastic Compute Cloud (**EC2**) is a part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), that allows users to rent virtual computers on which to run their own computer applications. EC2 encourages scalable deployment of applications by providing a web service through which a user can boot an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired. A user can create, launch, and terminate server-instances as needed, paying by the second for active servers – hence the term "elastic". EC2 provides users with control over the geographical location of instances that allows for latency optimization and high levels of redundancy.[2] In November 2010, Amazon switched its own retail website platform to EC2 and AWS.

How AWS EC2 works?

It’s pretty simple to get up and running with Amazon EC2. You have a choice of pre-configured, templated Amazon Machine Images (AMI) to use for a quick launch. Or, if you prefer, you can create your own AMI that contains all of your libraries, data, applications and relevant configuration settings.

Amazon EC2 allows you to customize settings by configuring security and network access. After you do that, you determine your AMI instances and whether or not you want to run in multiple locations. You’ll also want to decide if you want static IP endpoints. You can do all of this in the AWS Console with minimum friction.

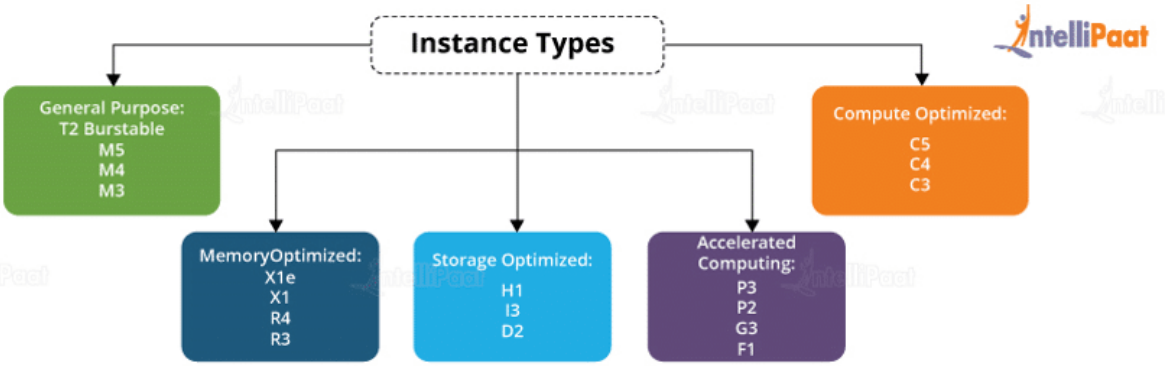
Once you’re ready to get started, you’ll boot, terminate and monitor as many instances as needed. You can do this through a web service or a variety of available management tools. And you’ll pay only for the resources you use.



What are the services Provided by Amazon web service EC2?

Amazon Elastic Compute Cloud (Amazon EC2) offers the broadest and deepest compute platform, with over 475 instances and choice of the latest processor, storage, networking, operating system, and purchase model to help you best match the needs of your workload. We are the first major cloud provider that supports Intel, AMD, and Arm processors, the only cloud with on-demand EC2 Mac instances, and the only cloud with 400 Gbps Ethernet networking. We offer the best price performance for machine learning training, as well as the lowest cost per inference instances in the cloud. More SAP, high performance computing (HPC), ML, and Windows workloads run on AWS than any other cloud.

.



Advantages of AWS EC2

1. ELASTIC WEB-SCALE COMPUTING

Amazon EC2 enables you to increase or decrease capacity within minutes, not hours or days. You can commission one, hundreds or even thousands of server instances simultaneously. Of course, because this is all controlled with web service APIs, your application can automatically scale itself up and down depending on its needs.

2. COMPLETELY CONTROLLED

You have complete control of your instances. You have root access to each one, and you can interact with them as you would any machine. You can stop your instance while retaining the data on your boot partition and then subsequently restart the same instance using web service APIs. Instances can be rebooted remotely using web service APIs. You also have access to console output of your instances.

3. FLEXIBLE CLOUD HOSTING SERVICES

You have the choice of multiple instance types, operating systems, and software packages. Amazon EC2 allows you to select a configuration of memory, CPU, instance storage, and the boot partition size that is optimal for your choice of operating system and application. For example, your choice of operating systems includes numerous Linux distributions, and Microsoft Windows Server.

4. DESIGNED FOR USE WITH OTHER AMAZON WEB SERVICES

Amazon EC2 works in conjunction with Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS) and Amazon Simple Queue Service (Amazon SQS) to provide a complete solution for computing, query processing and storage across a wide range of applications.

5. RELIABLE

Amazon EC2 offers a highly reliable environment where replacement instances can be rapidly and predictably commissioned. The service runs within Amazon’s proven network infrastructure and datacenters.

6. SECURE

Amazon EC2 works in conjunction with Amazon VPC to provide security and robust networking functionality for your compute resources. Your compute instances are located in a Virtual Private Cloud (VPC) with an IP range that you specify. You decide which instances are exposed to the Internet and which remain private.

7. INEXPENSIVE

Amazon EC2 passes on to you the financial benefits of Amazon’s scale. You pay a very low rate for the compute capacity you actually consume.

8. EASY TO START

Quickly get started with Amazon EC2 by visiting the Amazon Web Services Management Console to choose preconfigured software on Amazon Machine Images (AMIs). You can quickly deploy this software to EC2 via the EC2 console.

Disadvantages of AWS EC2

1. Instance types are rigid, must get entirely bigger instances even if just interested in more CPU or RAM.

2. Expensive at on-demand rates if elasticity is not needed (or expensive upfront payment if not using server for entire purchase length).

3. VM performance can be highly variable due to size, placement, and other uses on the same physical hardware.

4. No IPv6 support yet (as of 2015).

5. No VPN access to internal network (need custom software or DirectConnect service).

6. Networking is not as flexible as other providers or colo (no multicast support, low control over interfaces and bindings). Cross-region communication is more complex to setup.

Conclusion

EC2 simply means Elastic Compute Cloud .It is one of service that is provided by the AWS.

Moreover it serve as a Infrastructure as a Service (Iaas), because the AWS provides the server, networking facilities I.e. Infrastructure.

For ex:- If we want to build a house then first of all we need a basic structure of house…later on we will decide how to design our house on various aspects wall painting, interior design.

So the basic structure of house serves as an Infrastructure…

Similarly EC2 can be assumed to be similar that of house structure…

Later on it provides facility of adding Software's , apps , database or anything else means in EC2 we have control of operating system..(I.e. we can design our house and can make the modifications as per our choice).

In EC2 each server is known as instance,

Step 1:- We launch a EC2 server (instance)

Step 2:- Secondly we connect to EC2 server through remote protocols such as SSH, etc..

Step 3:- Now as soon we r connected to instance we got interface of the instance launched (GUI in case of windows server, click in case Linux distribution)

IT FOLLOWS ON- DEMAND PRICING -

On-Demand Instances let you pay for compute capacity by the hour or second (minimum of 60 seconds) with no long-term commitments. This frees you from the costs and complexities of planning, purchasing, and maintaining hardware and transforms what are commonly large fixed costs into much smaller variable costs.

REFERENCES

1. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html
2. https://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud
3. Mastering Cloud Computing Foundations and Applications Programming by Rajkumar Buyya, Christian Vecchiola, S. Thamar Selvi .

PROJECT ON RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Arkojeet Bera
College Roll No : 729
Examination Roll No : 2021141306
Paper Code: CMSADSE3
Semester: V
Department: Computer Science
Supervisor: Dr. Chayan Halder

Contents:

RETAIL SHOPPING MANAGEMENT SOFTWARE

Problem Statement – A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply his/her residential address, mobile number and driving license. Each customer who registers for this scheme is assigned a unique customer number (CN) by the computer. A customer can present CN to the checkout staff when any purchase is made. In this case, the value of this purchase is credited against the CN. At the end of each year, the supermarket intends to award surprise gifts to ten customers who make the highest total purchase over the year. Also it intends to award a 22 karat gold coin to every customer whose purchase exceeds INR 10000. The entries against the CN are reset on the last day of every year after the prize winners' list is generated.

Process Model – Prototyping model

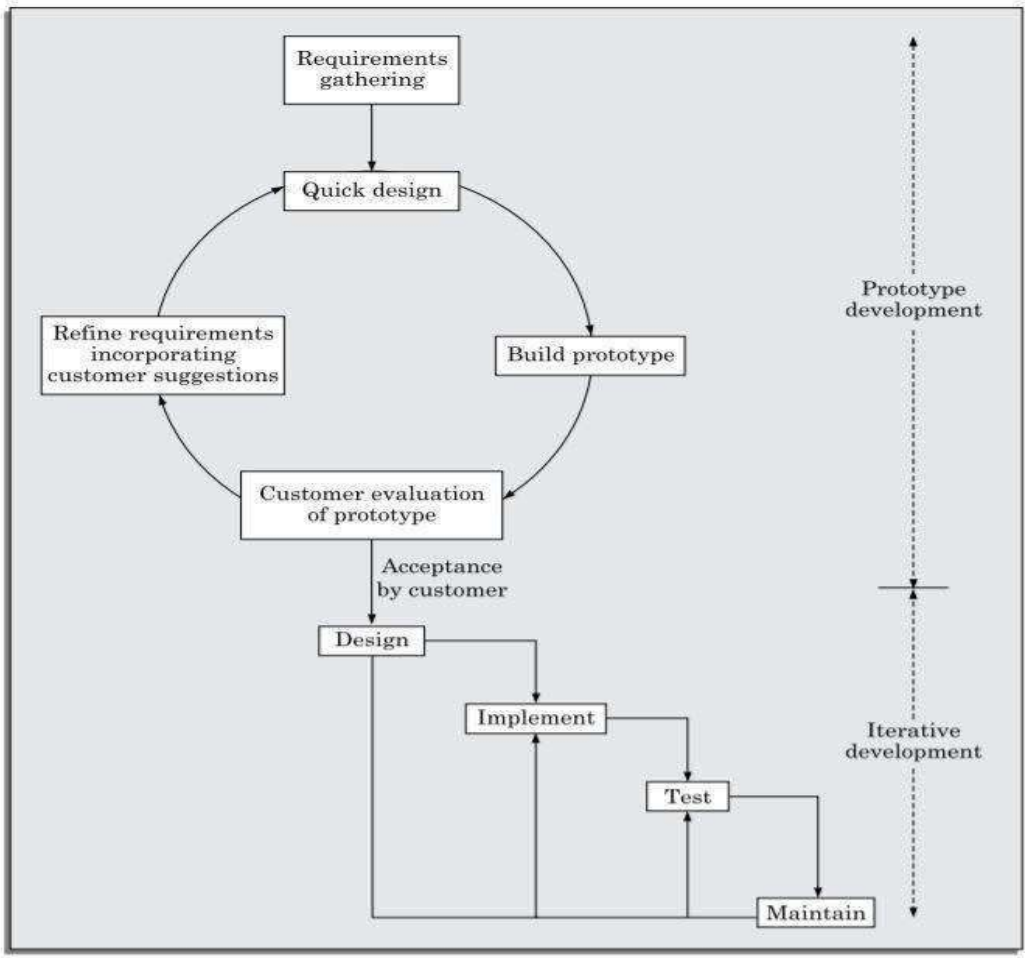


Fig :– Prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) Meeting of the developers and the customers to define the overall objectives of the software
2) Quick design leads to construction of prototype
3) Evolution of prototype by customer
4) Iteration occurs as the prototype is turned to the needs of the customers
5) The code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and assigned a unique customer number (CN).
   A database is maintained for keeping the customer details.

   Input: customer details
   Output: unique CN

2. **Purchase Module –** In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.

   Input: purchase made by the customer, CN
   Output: updated database

3. **Promotional Module –** In this module, customers are selected for giving out the prizes and the prize winners' list is generated.
   The entries against the CN are reset after the generation of the prize winners' list.

   Input: purchase history, CN
   Output: prize winners' list

Design Engineering

Data Flow Diagrams (DFDs)

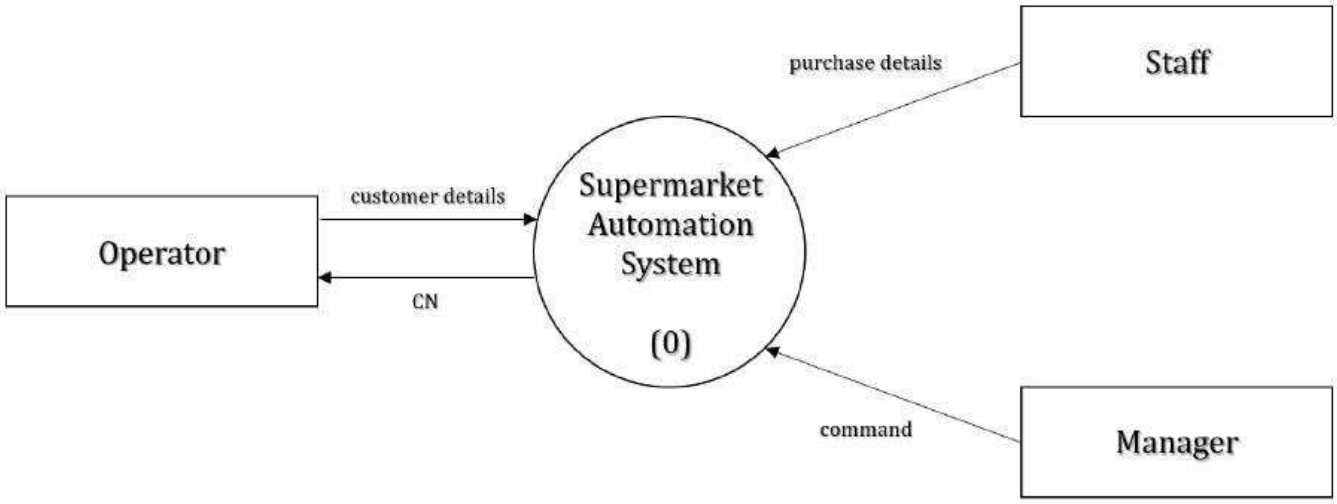


Fig :– DFD level zero

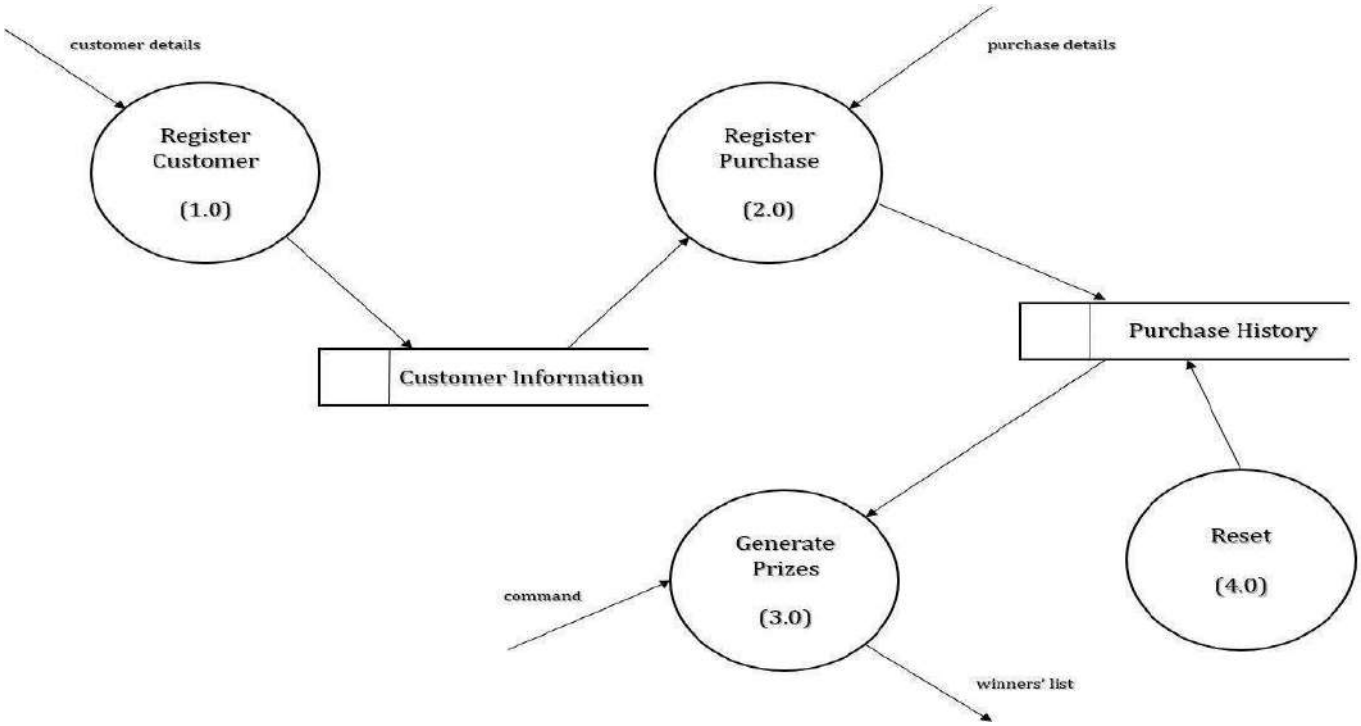


Fig :– DFD level one

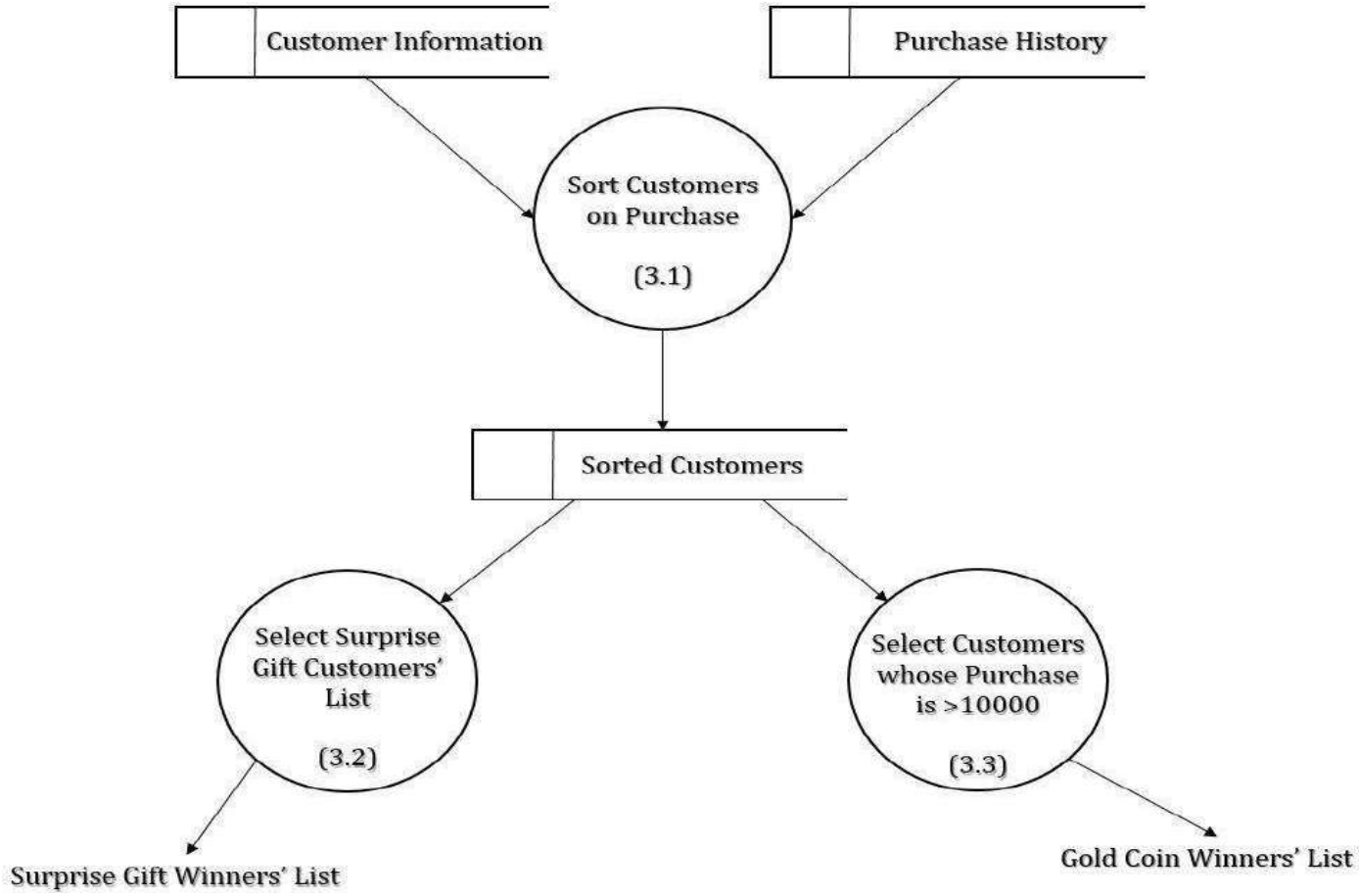
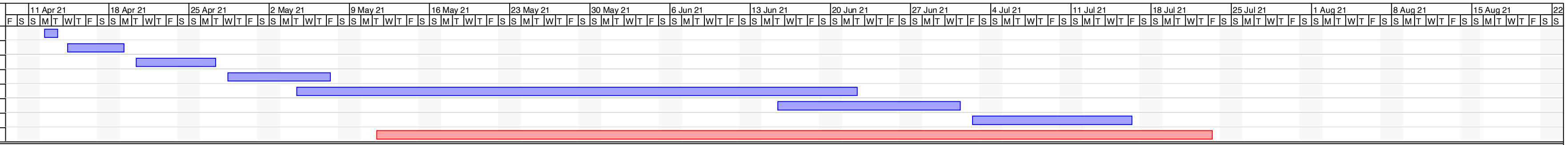


Fig :– DFD level two of Process 3.0

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Retail Shopping Management Software

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by a friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Route Management System (Dtc Route Information)

Problem statement – It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates reports of all bookings, all tickets, bus schedules and all payments.

Process model – Incremental model

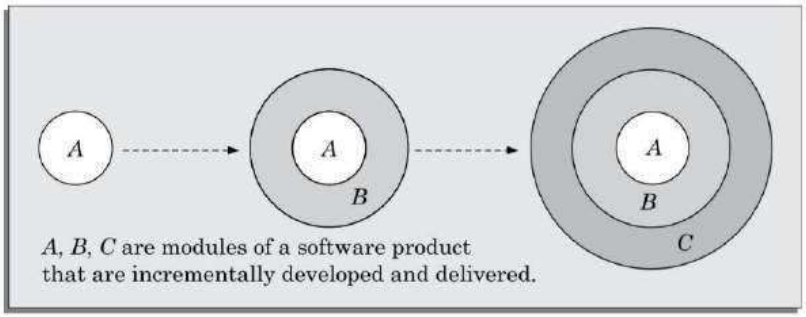


Fig – Incremental model of software development

This life cycle model is also referred to as successive versions of the model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by an iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Bus Ticketing Module –** In this module, any booking made by the customer is registered and the ticket is generated.

   Input: customer details
   Output: generated ticket

3. **Information Module –** In this module, the bus details of the booking made by the customer is maintained.
   A database is maintained for keeping the information.

   Input: customer details, ticket details
   Output: updated database

4. **Report Module –** In this module, the report of all bookings, tickets, bus schedules and payments is generated.

   Input: booking details, ticket details, bus details, payment details
   Output: generated report

Design Engineering

Data Flow Diagrams (DFDs)

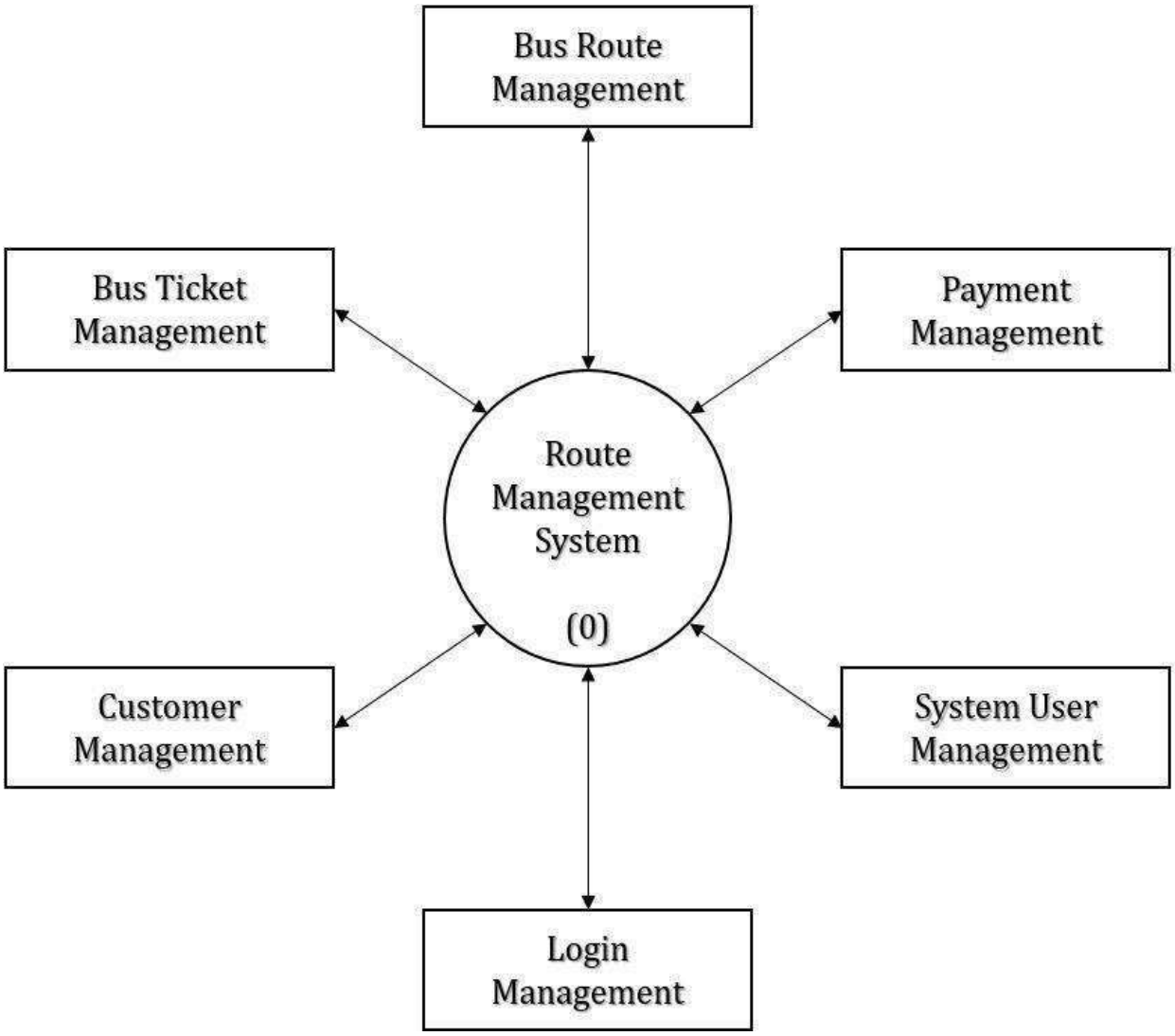


Fig – DFD level zero

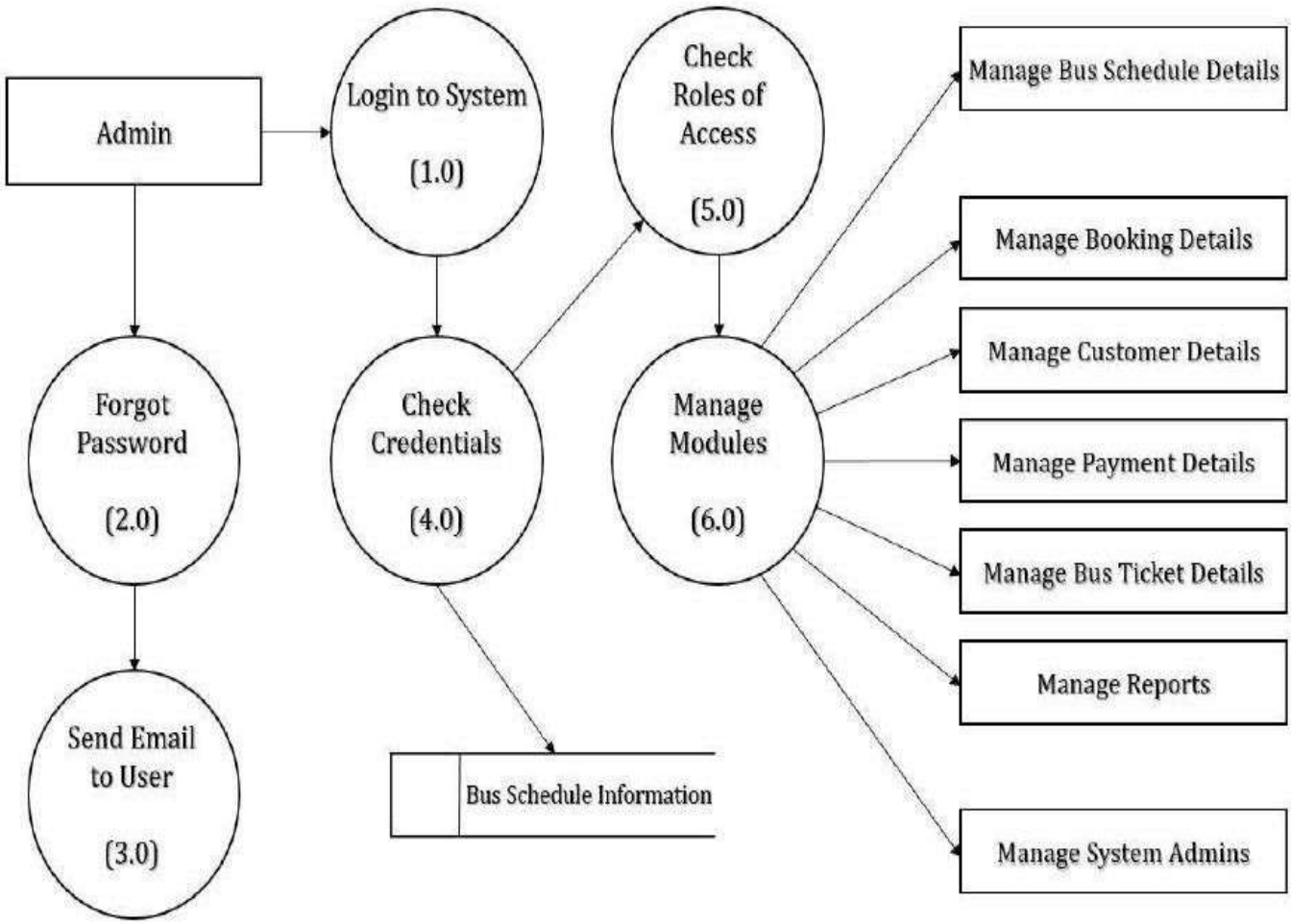
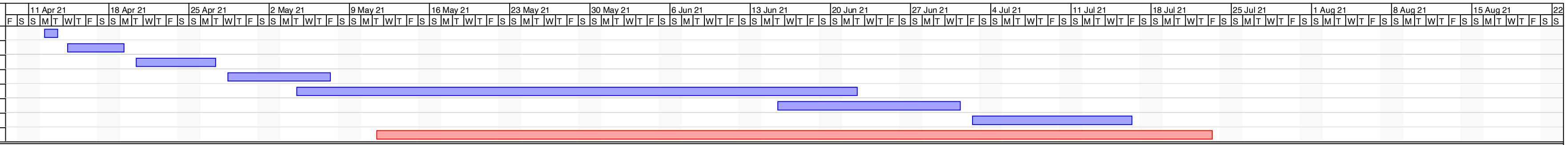


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Route Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $\mathbf{p = r \times s}$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by a friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Patient Appointment And Prescription Management System

Problem Statement – Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model – Incremental development model

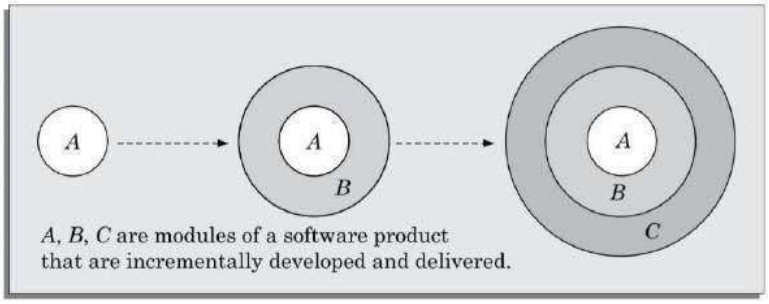


Fig – incremental model of software development

This life cycle model is also referred to as successive versions. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by an iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Software Requirement Analysis

1. **Patient Module –** In this module, a patient is registered and the profile is maintained.

   Input: patient details
   Output: unique profile

2. **Doctor Module –** In this module, the details of doctors are maintained.

   Input: patient details
   Output: updated database

3. **Diagnosis Module –** In this module, the tests are performed and the disease is diagnosed.
   A database is maintained for keeping the patient files.

   Input: patient details, doctor details, test details
   Output: diagnosed disease, updated database

4. **Report Module –** In this module, patient records, medicine and drug records, test records are processed and reports are generated.

   Input: patient details, doctor details, diagnosis details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

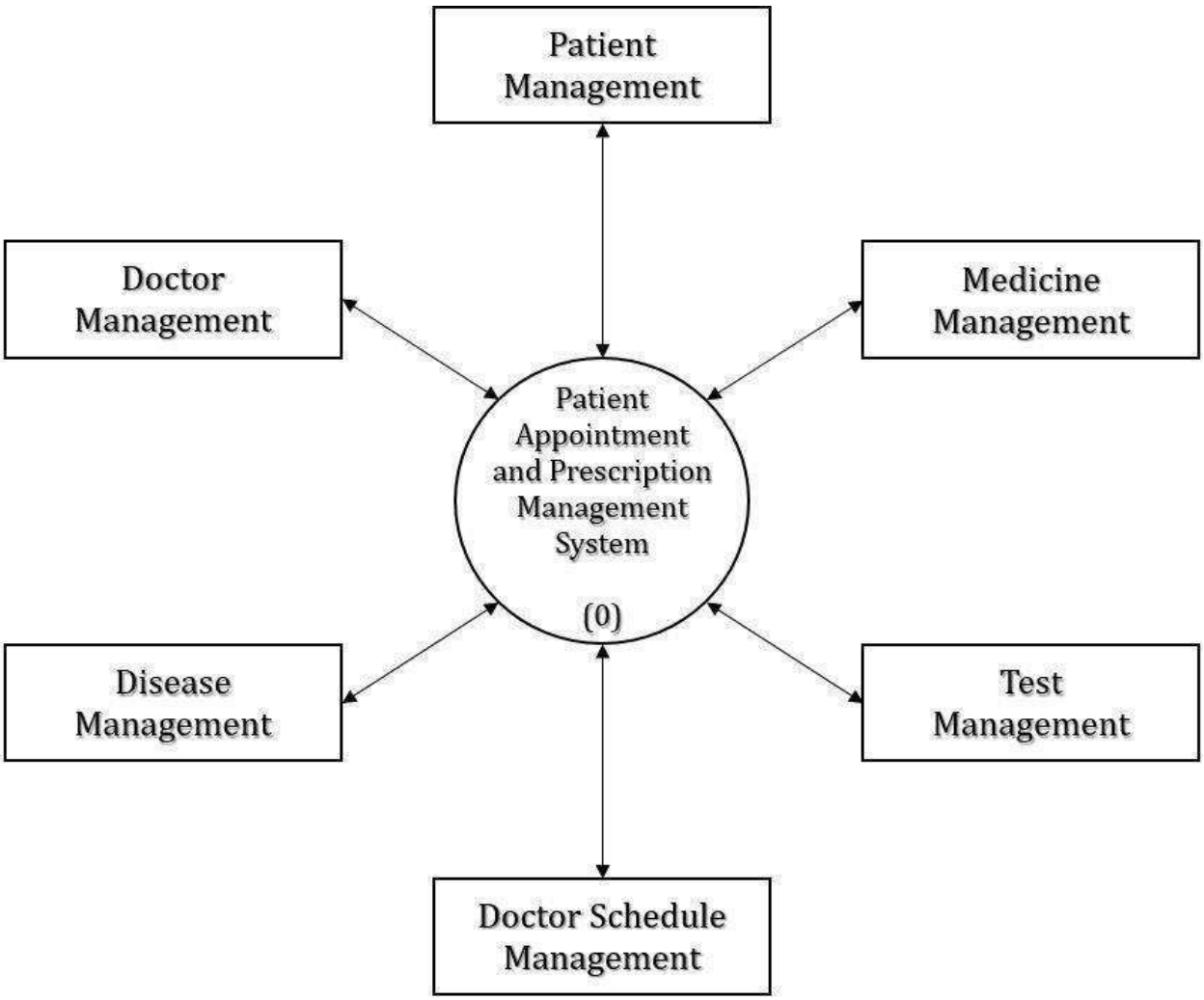


Fig – DFD level zero

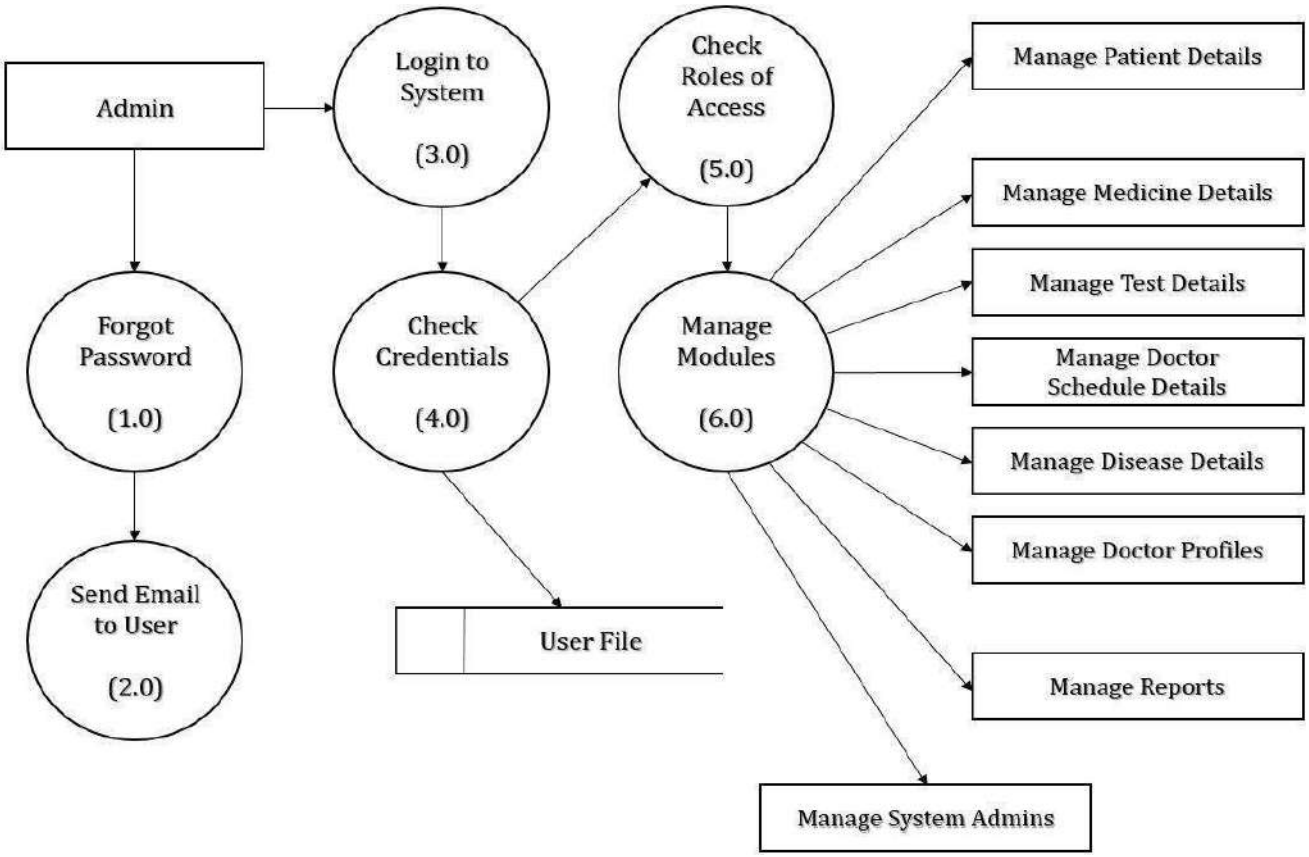
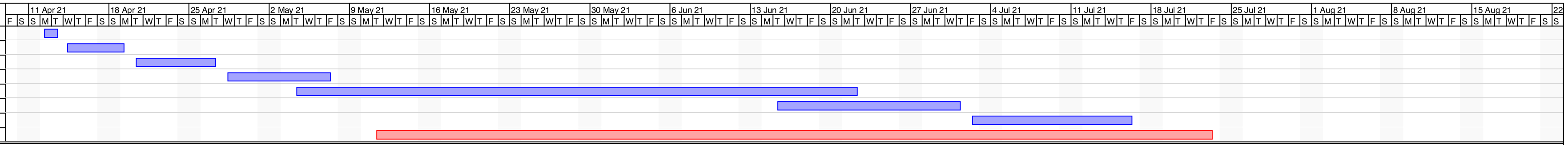


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Patient Appointment and Prescription Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Hotel Automation Management System

Problem Statement – Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model – Spiral development model

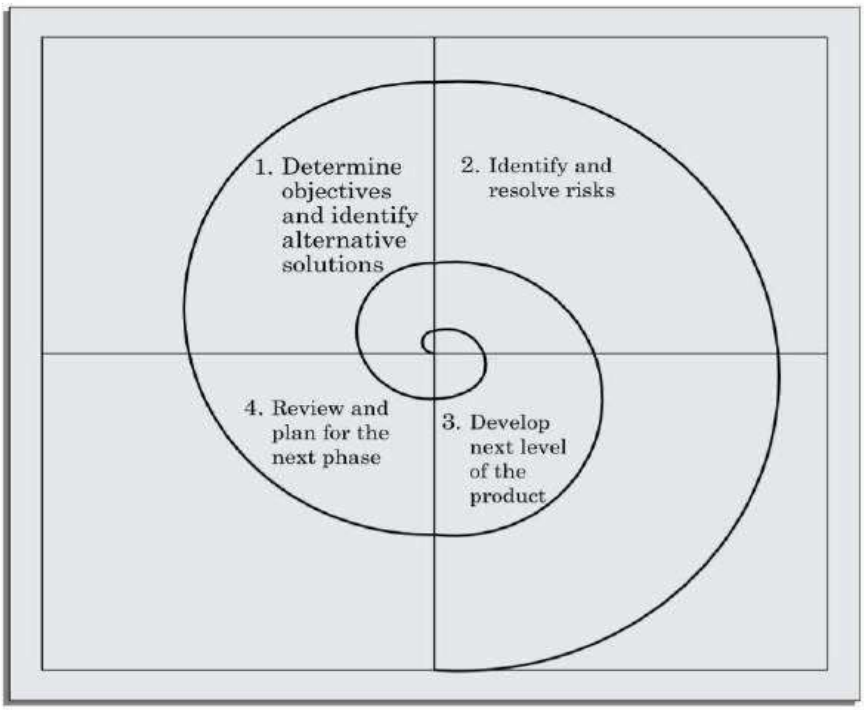


Fig – Spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived from the framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Guest Module –** In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number is also assigned. Else an apology message is generated.
   A database is maintained for keeping the information.

   Input: guest details
   Output: unique profile, updated database

2. **Catering Module –** In this module, the details of food items consumed by the guest are maintained.
   A database is maintained for keeping the information.

   Input: guest details, room number, token number
   Output: updated database

3. **Bill Module –** In this module, the total bill is generated when the guest prepares to check out.

   Input: guest details, room number, token number, food items consumed
   Output: generated bill

Design Engineering

Data Flow Diagrams (DFDs)

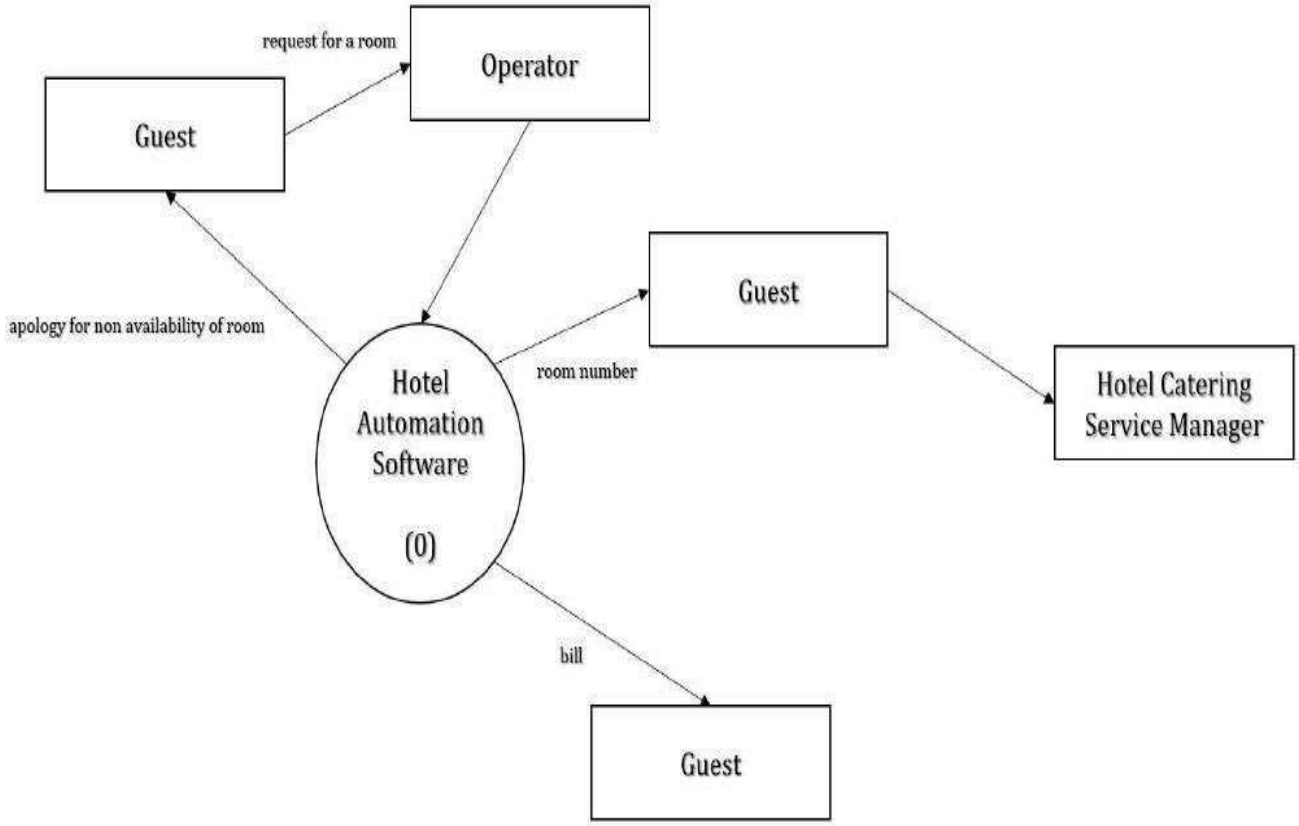


Fig – DFD level zero

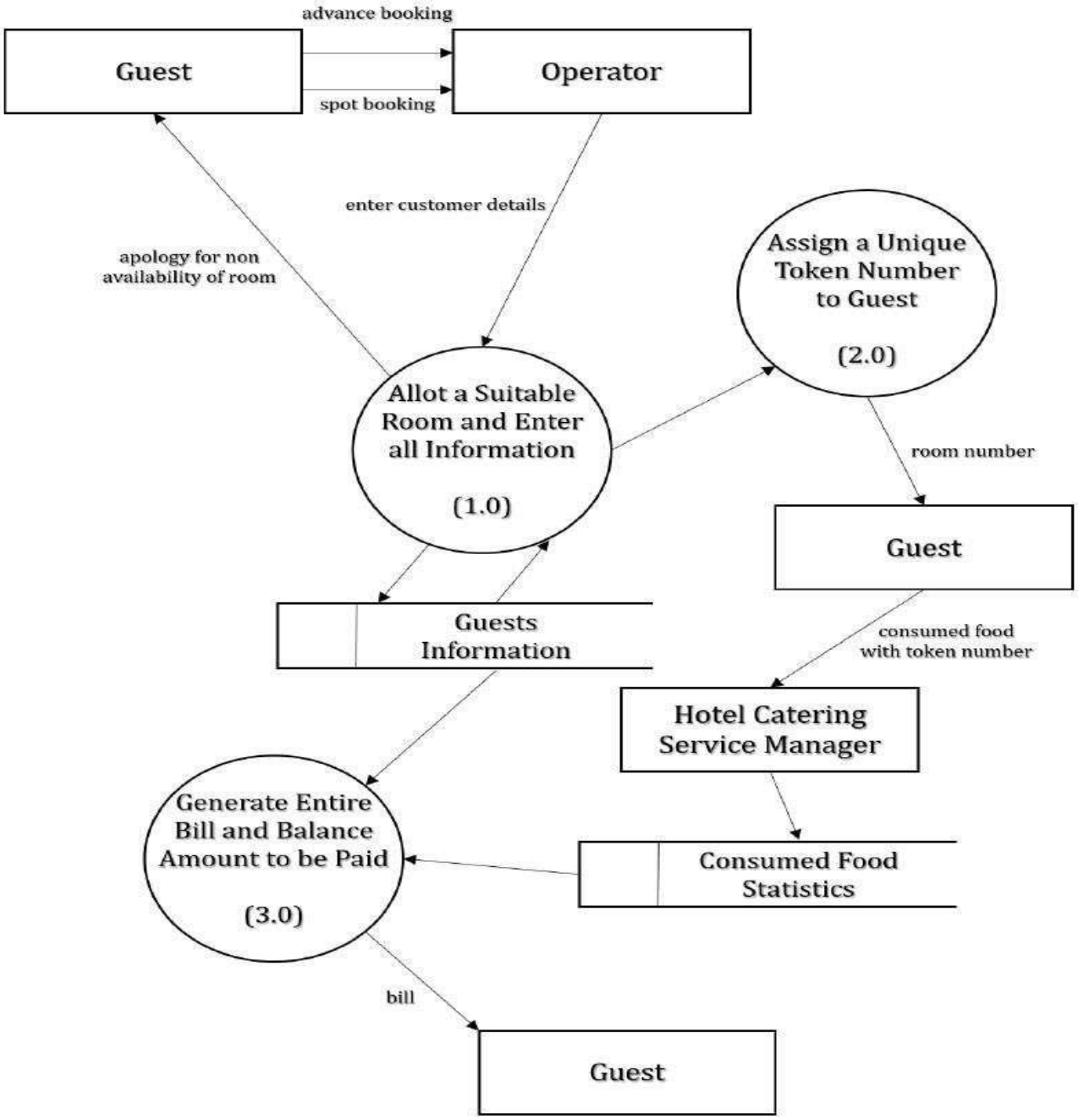
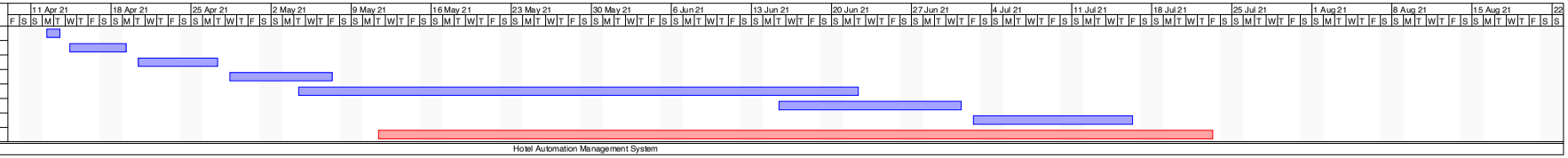


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Hotel Automation Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Criminal Record Management System

Problem Statement – Crime file management system identifies internal data stores of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model – Prototyping model

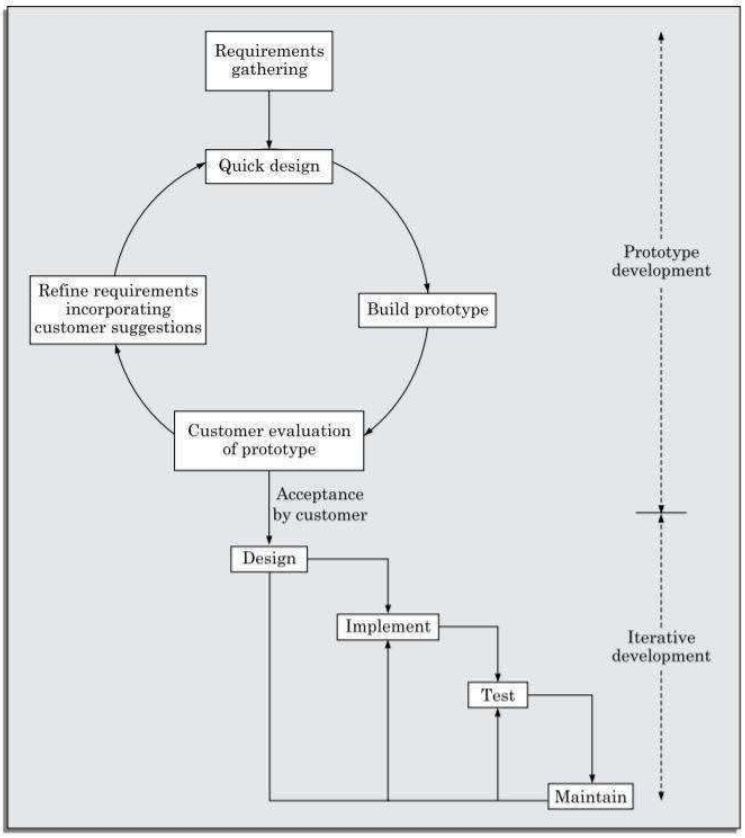


Fig – prototyping model of software development

This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1) Meeting of the developers and the customers to define the overall objectives of the software
2) Quick design leads to construction of prototype
3) Evolution of prototype by customer
4) Iteration occurs as the prototype is turned to the needs of the customers

the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Software Requirement Analysis

1. **Admin Module –** In this module, an admin is registered and the profile is maintained. The level of access is granted according to the designation.

   Input: admin details
   Output: unique profile, granted level of access

2. **Complaint Module –** In this module, the complaint is registered and FIR is filed.
   A database is maintained for keeping the complaint details.

   Input: complaint details
   Output: filed FIR, updated database

3. **Crime Module –** In this module, the details of the crime are maintained and a chargesheet is filed.

   Input: complaint details, FIR details
   Output: filed chargesheet

4. **Court Module –** In this module, the case is taken to the court and legal action is taken accordingly.

   Input: complaint details, FIR details, chargesheet details
   Output: legal action, updated database

Design Engineering

Data Flow Diagrams (DFDs)

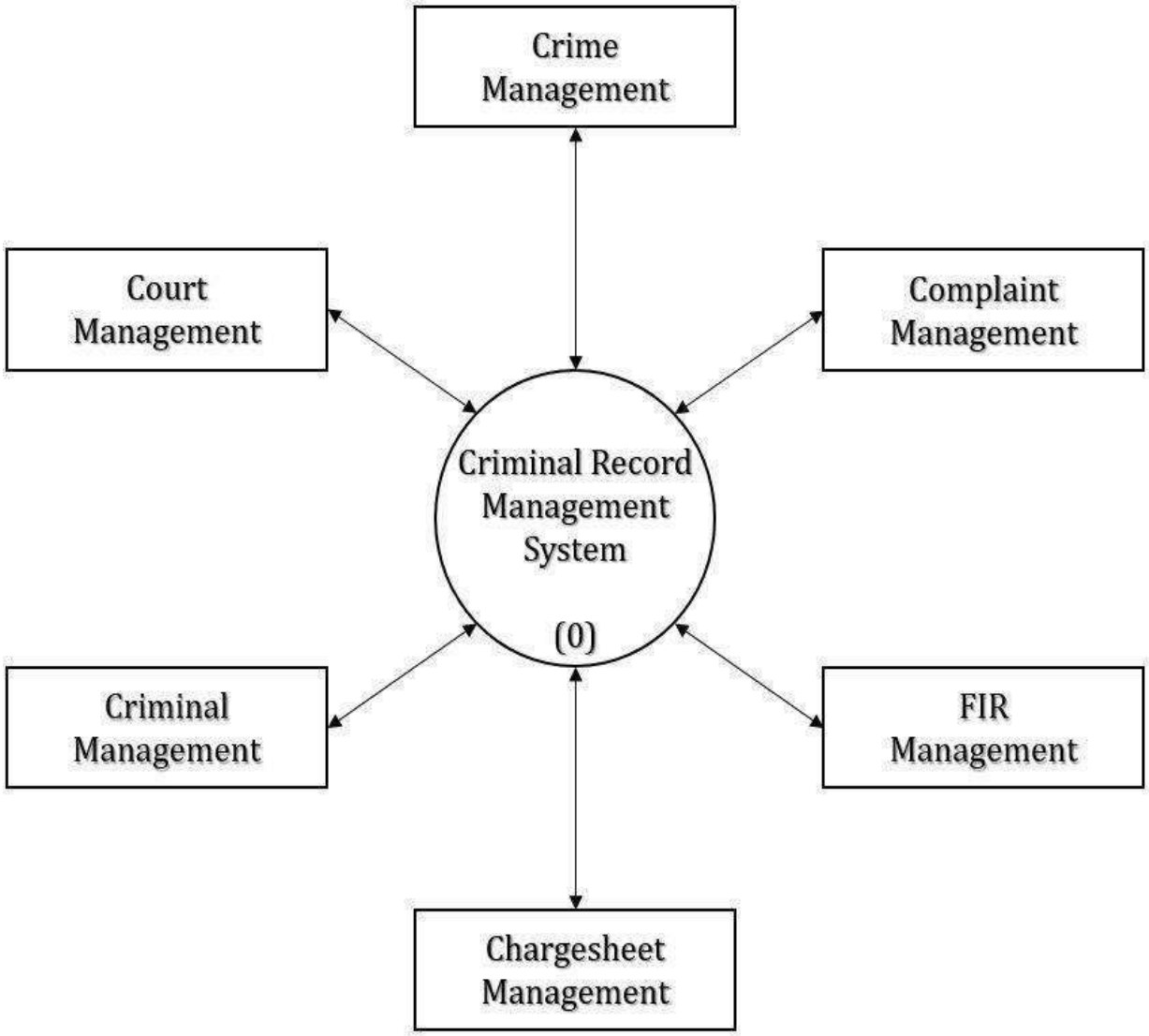


Fig – DFD level zero

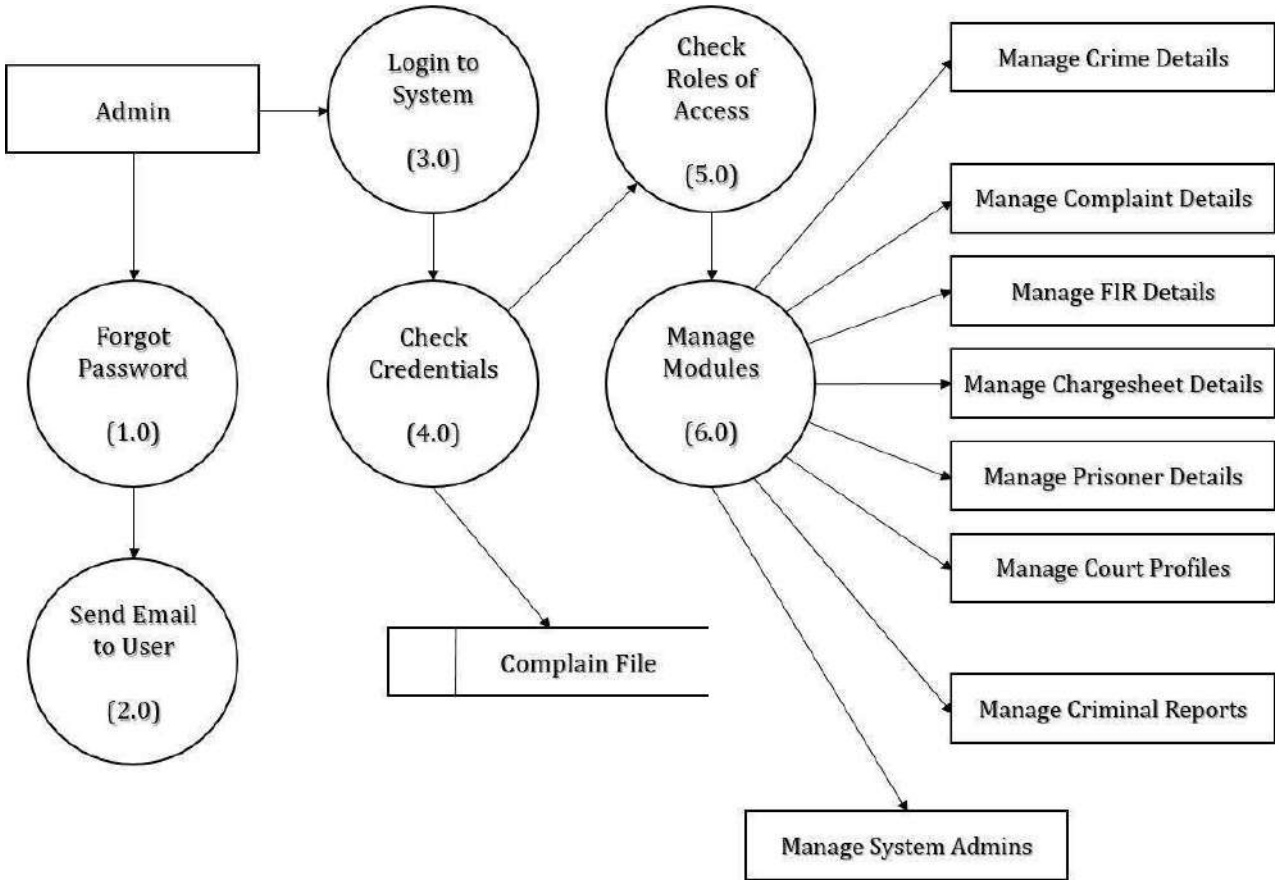
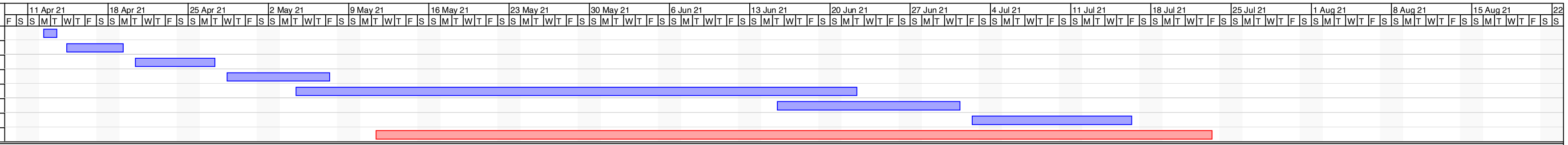


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Criminal Record Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by a friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Examination And Result Computation System

Problem Statement – Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generating reports of the same.

Process Model – Spiral development model

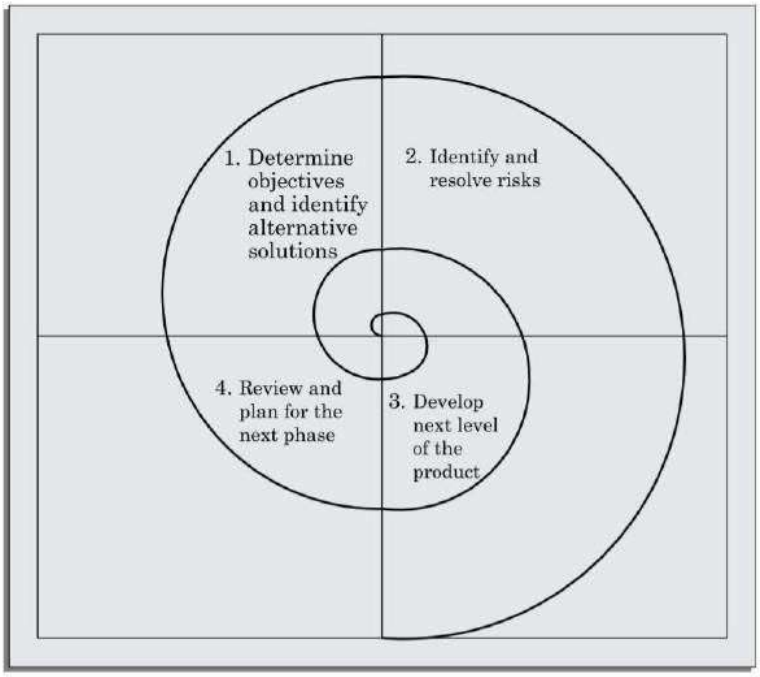


Fig – Spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived from the framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Student Module –** In this module, a student is registered and the profile is maintained.
   A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. **Timetable Module –** In this module, the details of branch, course, class and subject is maintained and the timetable is created.
   A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. **Examination Module –** In this module, the examination takes place and the results are computed.

   Input: student details, timetable details
   Output: computed results, updated database

4. **Report Module –** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

   Input: guest details, room number, token number, food items consumed
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

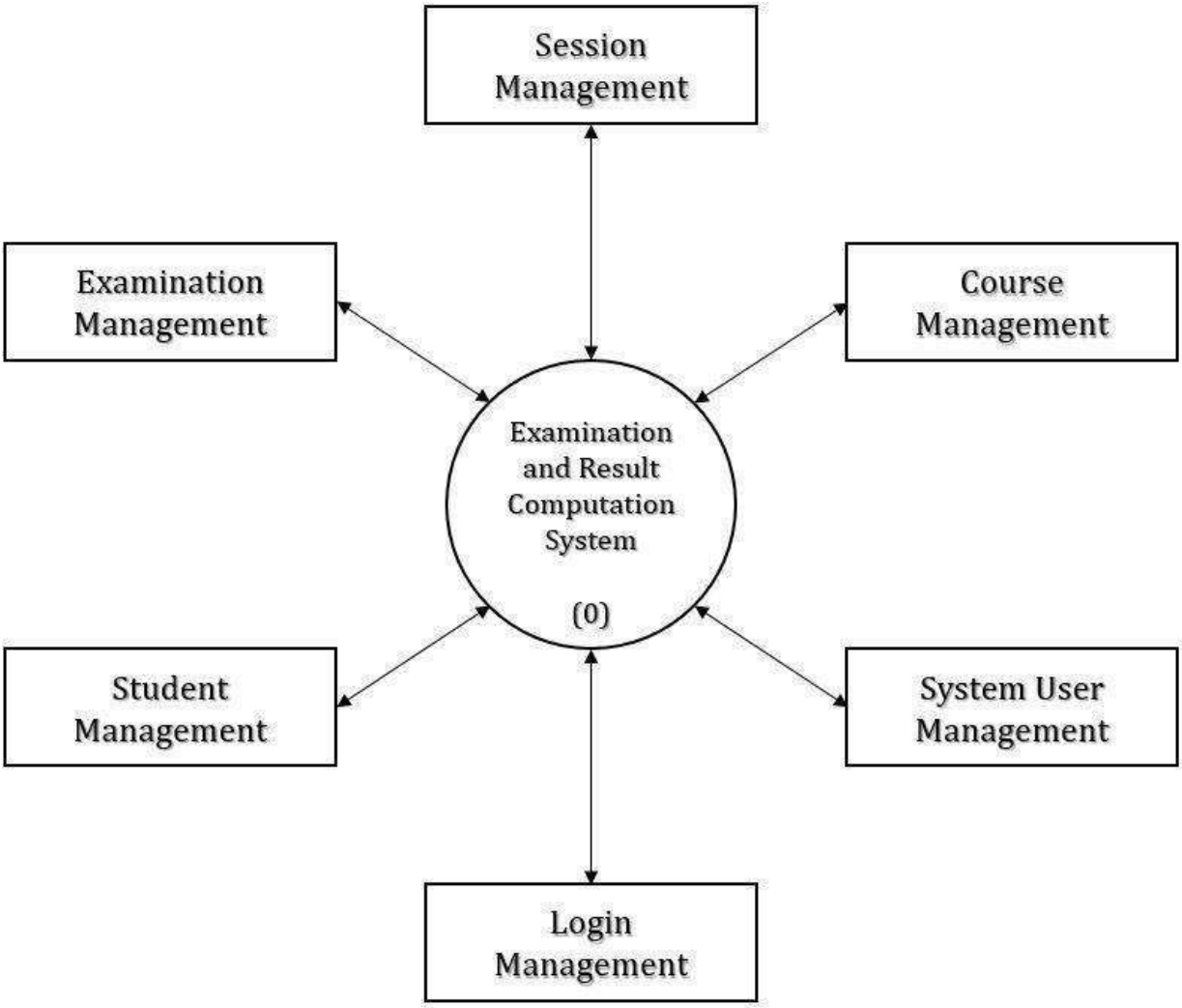


Fig – DFD level zero

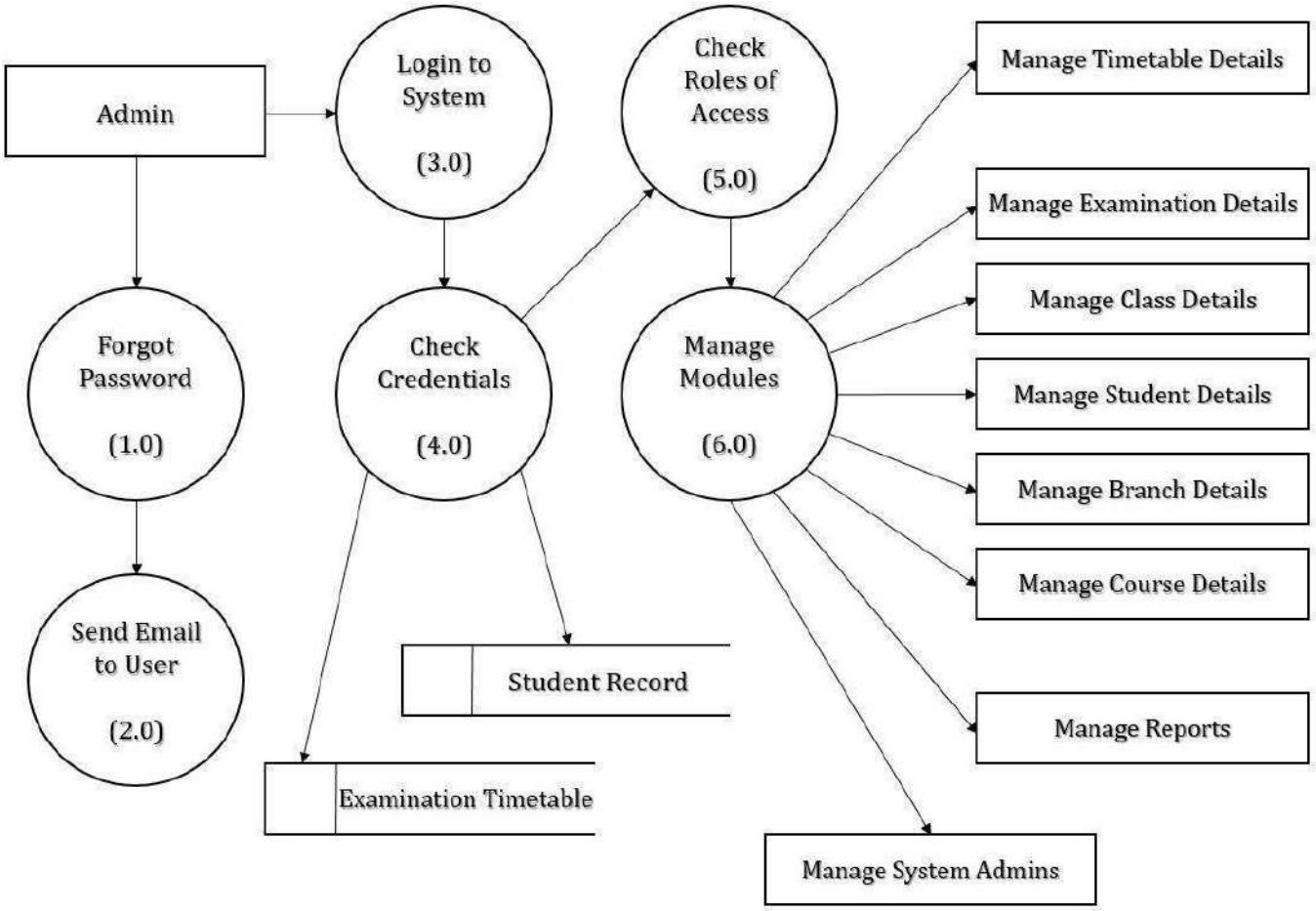
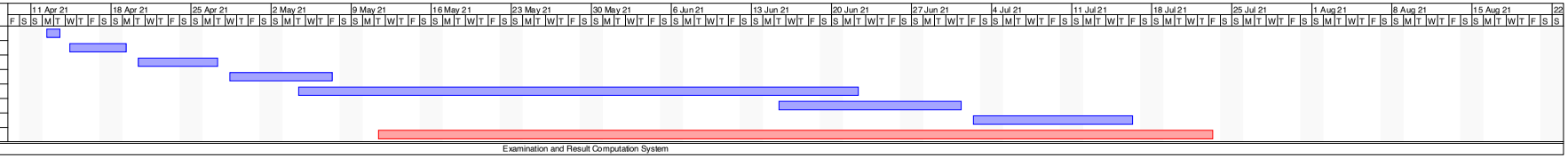


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Examination and Result Computation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Parking Allocation System

Problem Statement – Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of the parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model – Spiral development model

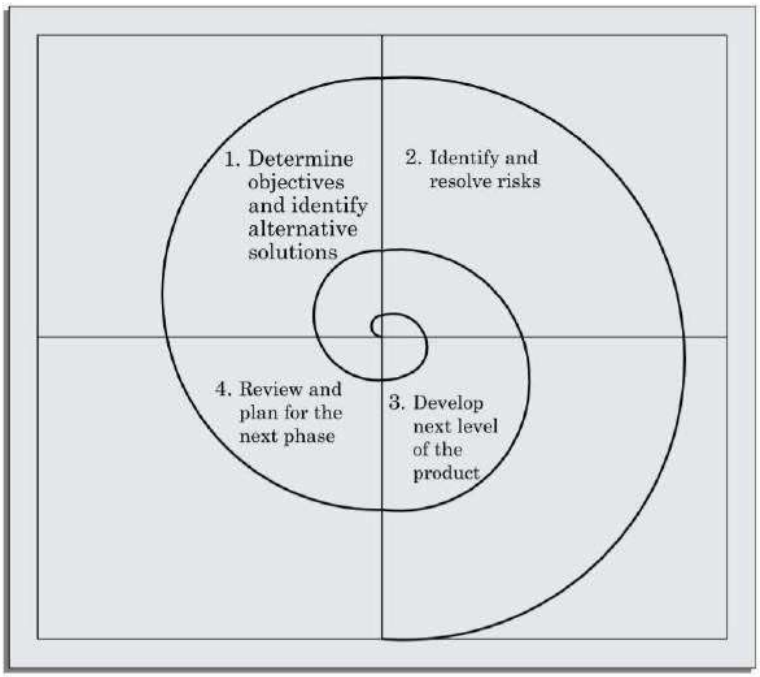


Fig – Spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived from the framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Vehicle Module –** In this module, the customer's vehicle is registered.

   Input: customer details, vehicle details
   Output: updated database

3. **Parking Module –** In this module, a parking slot is allotted and a fee is charged accordingly.
   A database is maintained for keeping the parking information.

   Input: customer details, vehicle details, duration
   Output: updated database

4. **Report Module –** In this module, parking slot, vehicle, parking fee and duration records are processed and reports are generated.

   Input: parking slot details, vehicle details, parking fee details, duration details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

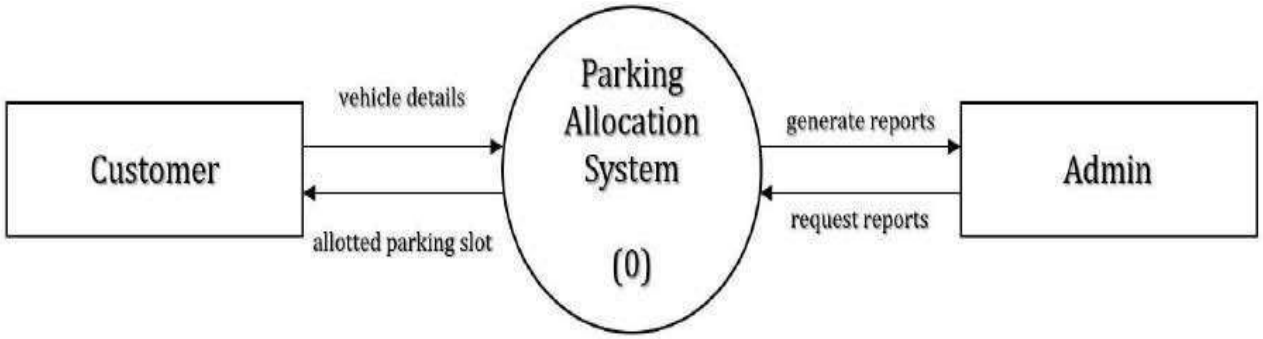


Fig – DFD level zero

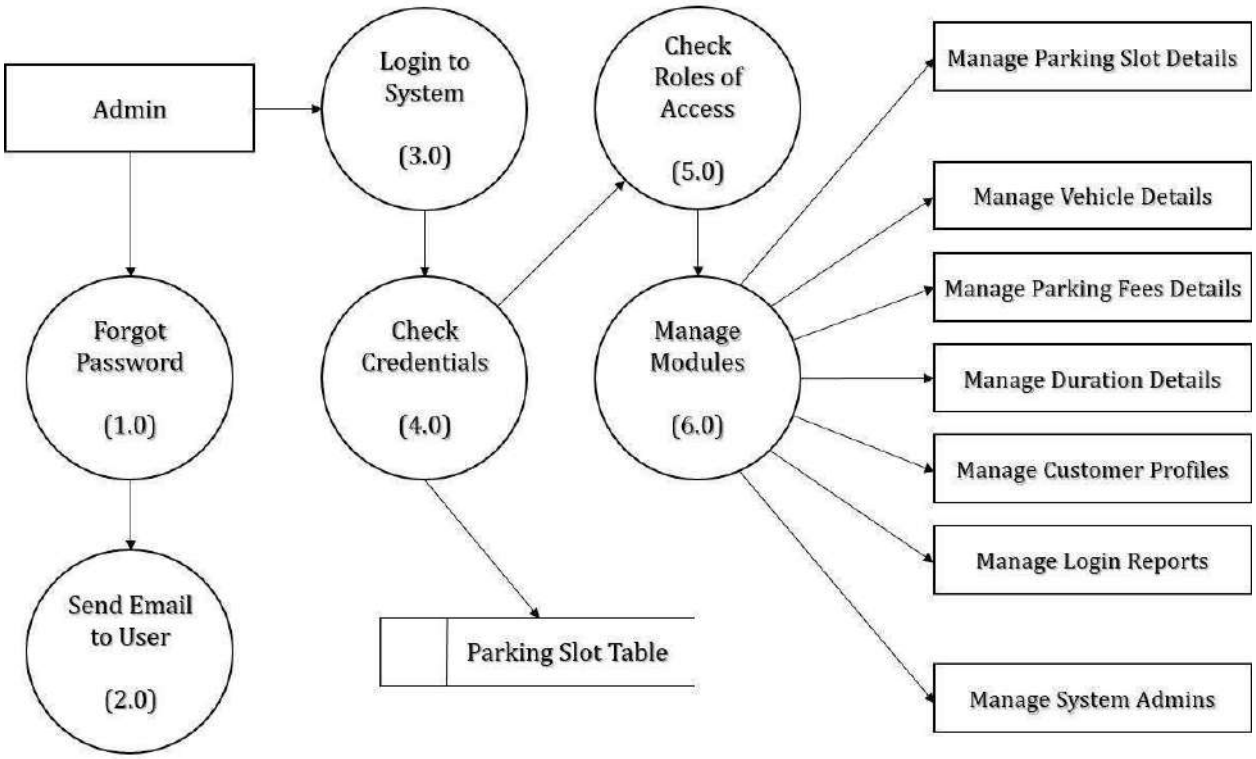
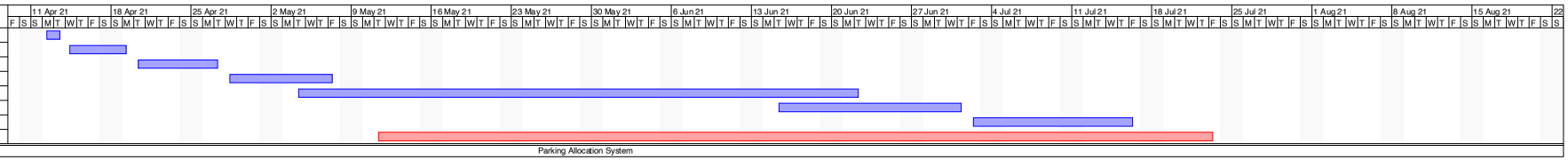


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Parking Allocation System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Wholesale Management System

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

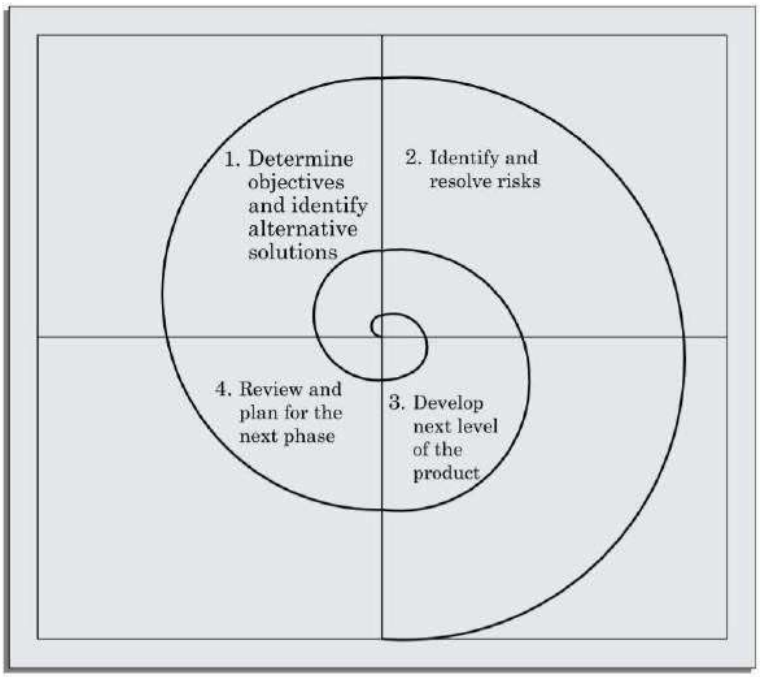


Fig – Spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived from the framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

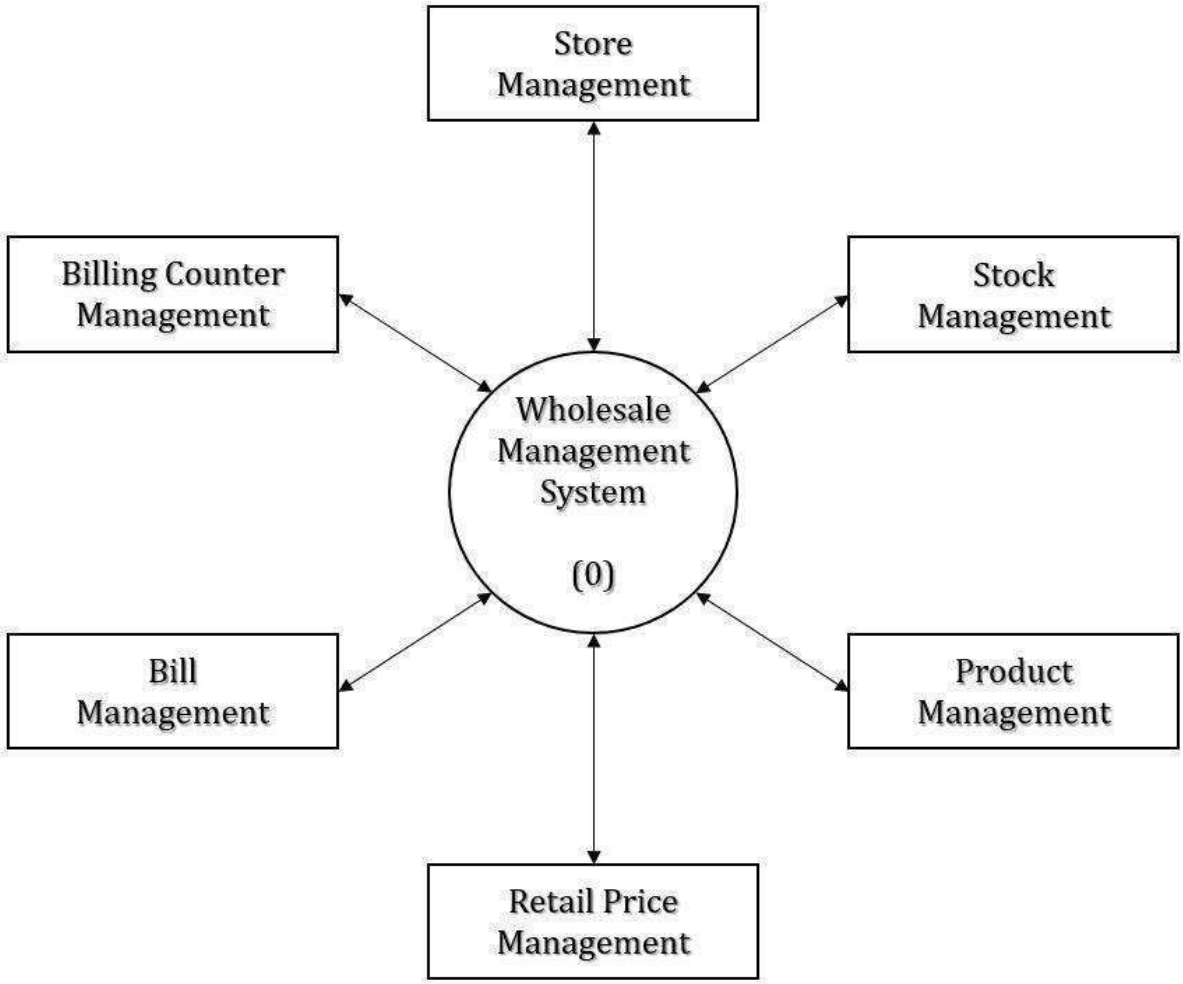


Fig – DFD level zero

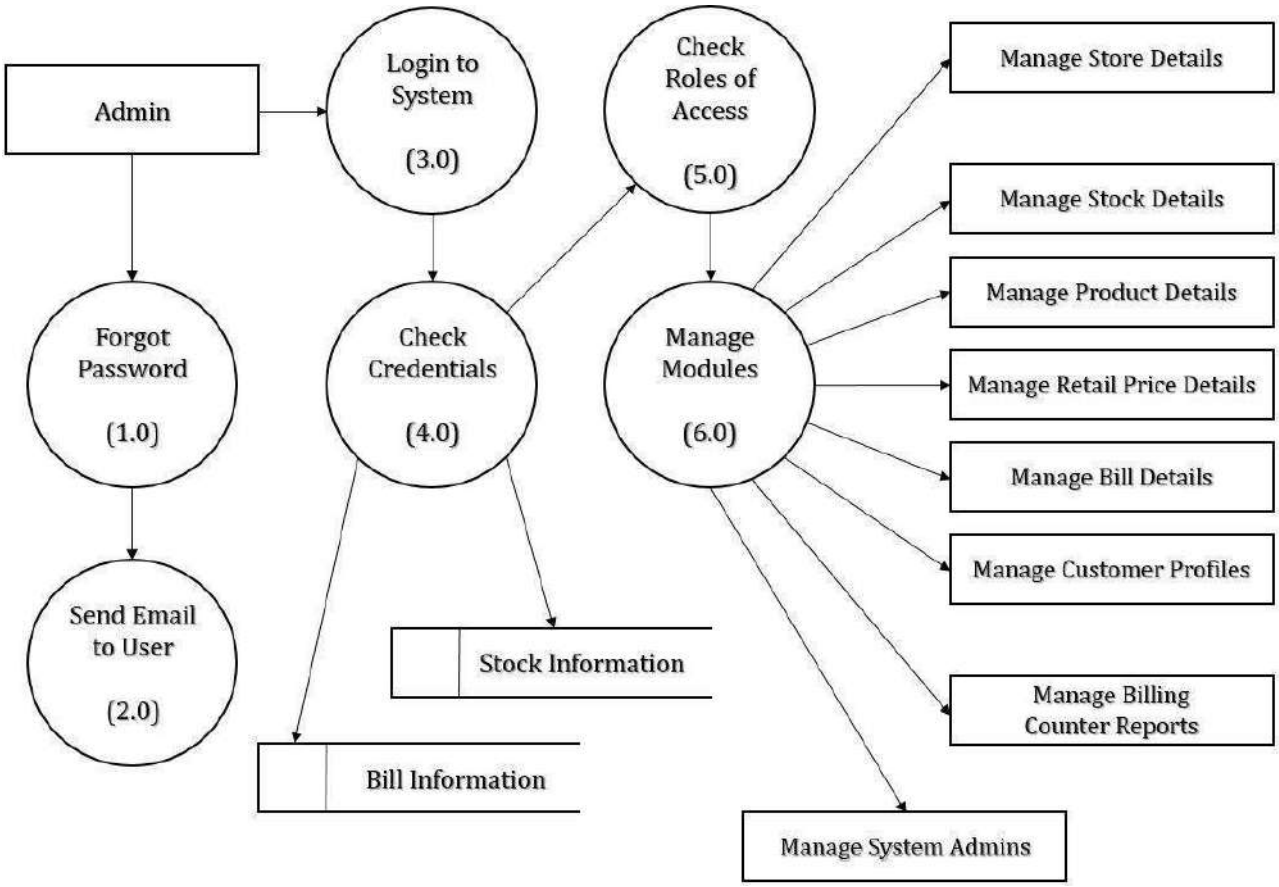
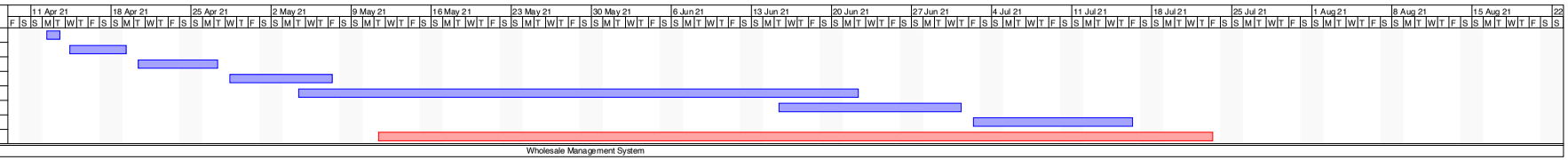


Fig – DFD level one

Project Management

Project Schedule – The file has been attached.

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Wholesale Management System

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as **p = r × s**

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

PROJECT ON RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Jaychandra De
College Roll No : 730
Examination Roll No : 2021141307
Semester: V
Department: Computer Science
Supervisor: Dr. Chayan Halder

Project .: 1
Organised Retail Shopping Management Software

Problem Statement : A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number, driving license number. Each customer who registers for this scheme is assigned the customer number (CN) by the computer. A customer can present his CN to the checkout staff when he makes his purchase. In this case the value of this purchase is credited again in its CN. At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh. The entries against the CN are reset on the last day of each year after the prize winner's lists are generated.

Process Model : Supermarket management software needs to change the information in a basis of regular interval. Every year the number of customers, the purchase history and other necessary information have to be changed. So the software needs to be added more new functionalities. A supermarket exists in a competitive environment, where it acts as a value-added intermediary between geographically dispersed supplier companies and the scattered individual customers who eventually buy their products. The supermarket management involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This incremental model combines the elements of the waterfall model with the iterative philosophy of prototyping. That is basic requirements are addressed, but many supplementary features (some known, others unknown) remain undelivered. The core product is used by the customer (or undergoes detailed review). As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. So for a supermarket management software system the **Incremental model** is the most suitable model.

Requirement Analysis: Basically three modules are used in a supermarket management system. Consumer module, Purchase module, Promotional module. On basis of these three modules supermarket management software can be designed.

- **Consumer Module :**
  A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number. When a customer registers his name in the system he will be provided with a

unique Customer number (CN). To maintain the record of the customer and their CN a database is maintained.

- **Input:** Name, residence number, telephone number, driving licenses all other necessaries.
- **Output:** Customer number (CN).

➢ **Purchase Module :**

When a customer made his purchase his name, bought product, and buying time is recorded. To maintain the purchase history of every customer a separate database is maintained. According to that the amount is deducted from the buyer.

- **Input:** Customer name, bought product, time
- **Output:** Purchase history database

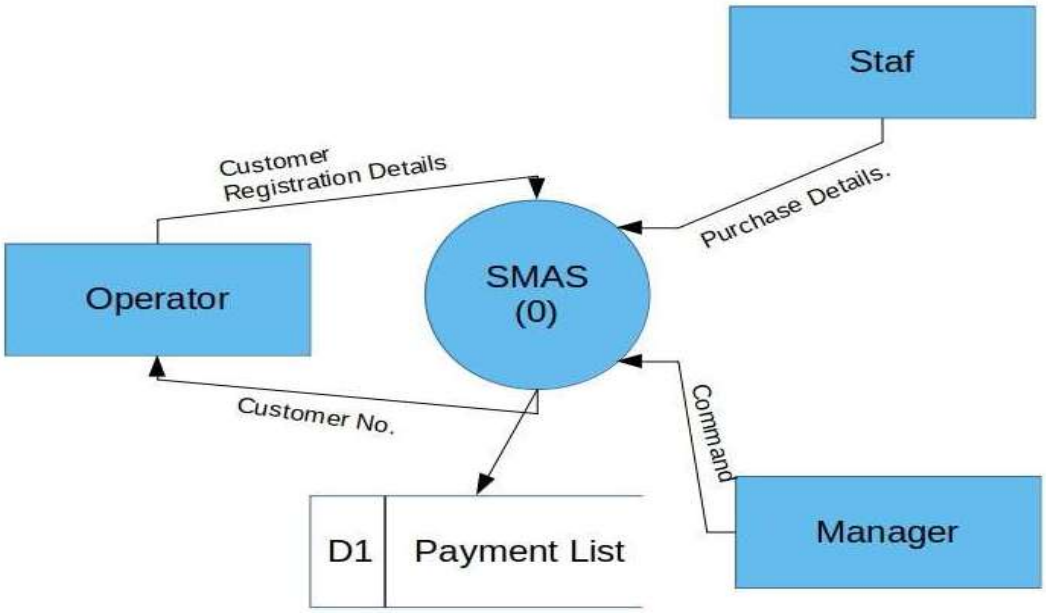
➢ **Promotional module :**

At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh. The entries against the CN are reset on the last day of each year after the prize winner's lists are generated. After lists are created the all history is reset.

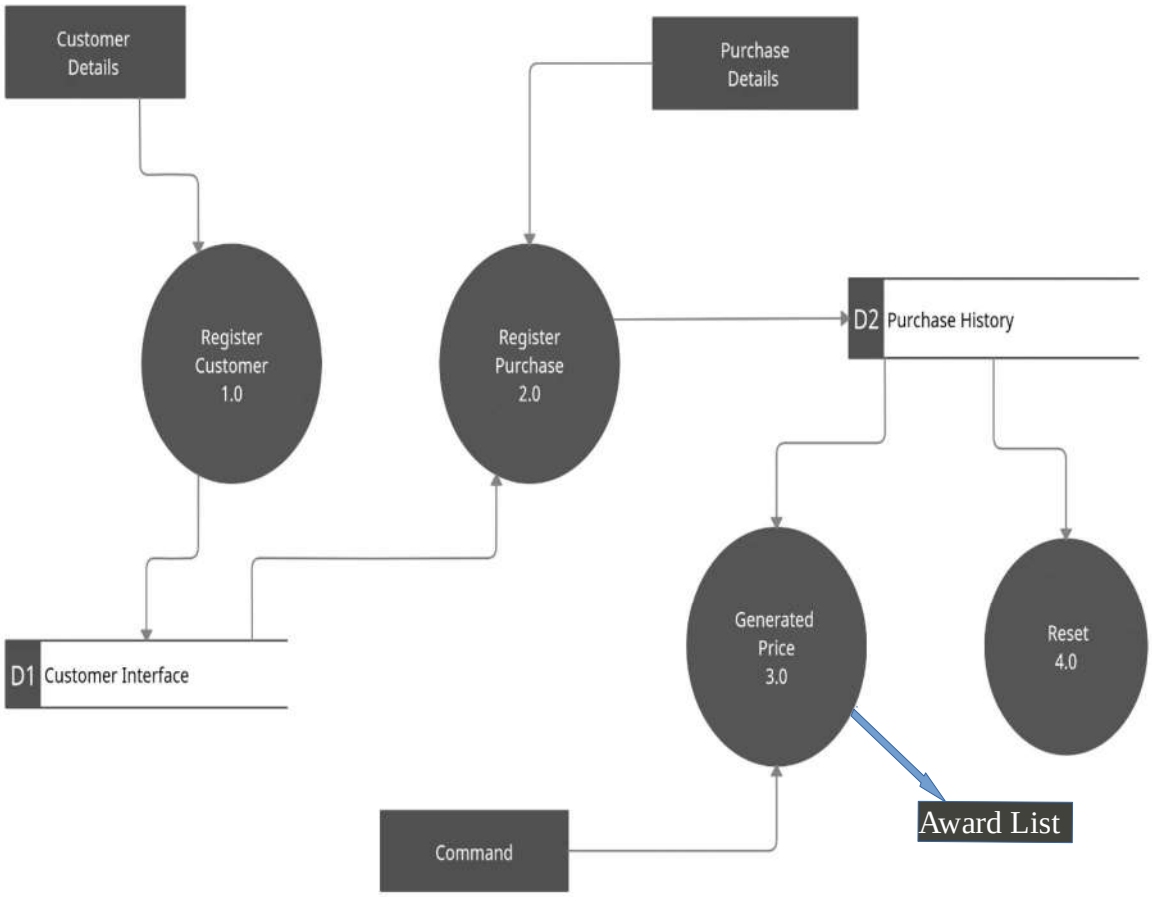
- **Input:** Purchase history according to the customer number
- **Output:** Surprise Gift Winner List, Gold coin Winner List

◆ **Data Flow Diagrams :**

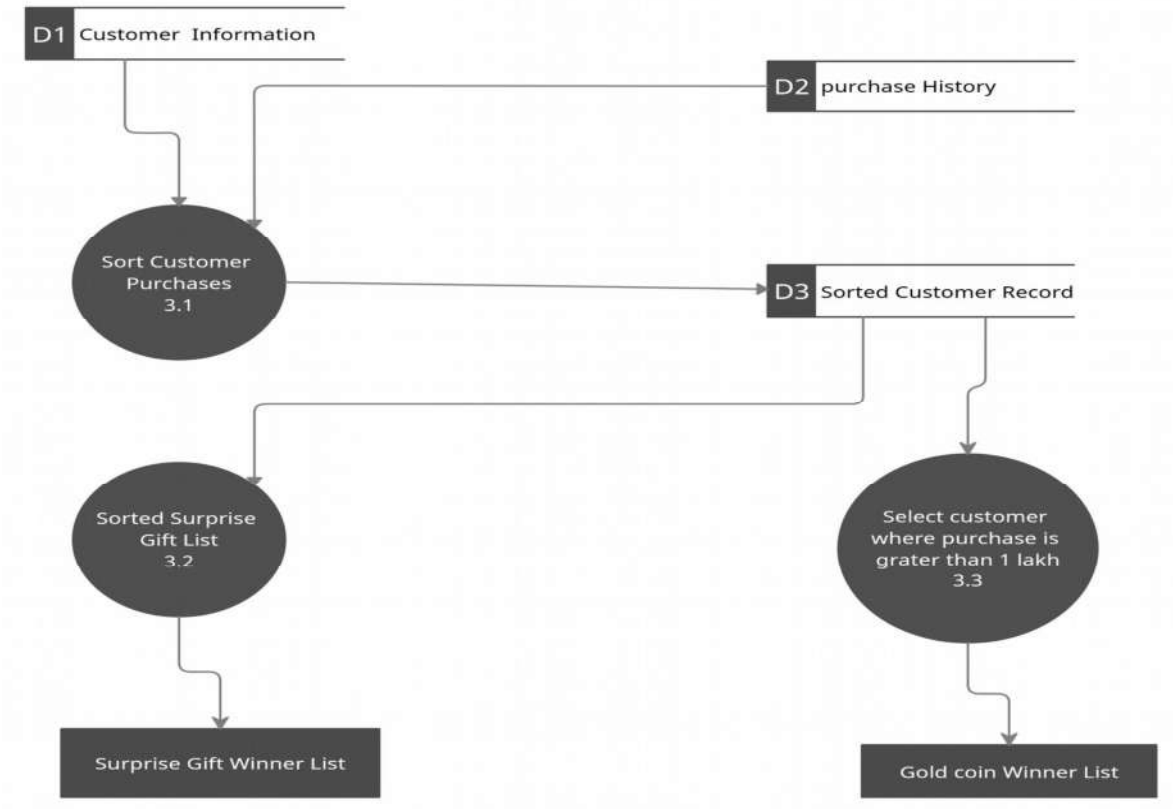
➢ **Zero Level DFD(Context Diagram) :**



- **Level One DFD :**



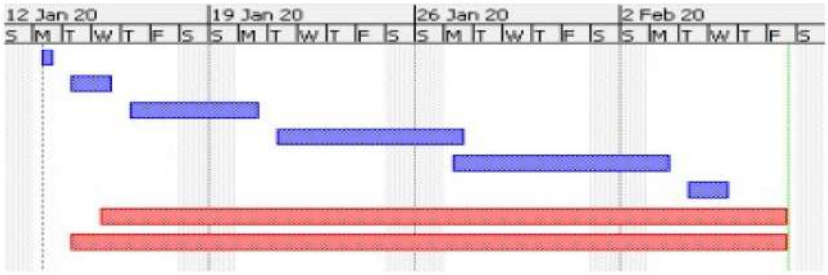
- **Level Two Of Process 3 :**



➢ **Project Management :**

◆ **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



➢ **Project Resources :**

- **Human Resources :**
  - Project manager
  - System analyst
  - Developer
  - Designer
  - Tester
  - Document creator

- **Software Resource :**
  - MS Word
  - MS Project
  - C (For forehand programming)
  - SQL (For backhand programming)

- **Hardware Resource :**
  - Processor: Intel() Core(TM) i5-4310U 2.60Ghz
  - Ram : 8.00 GB

Risk Management plan :

Risk factor = (Risk exposure before reduction - Risk exposure after reduction) / Cost of reduction

No such risk is there in the supermarket management system as it uses incremental model.

Testing:

Unit testing : Each component of the design is implemented as aprogram module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing : During the integration and system testing phase, the modules are integrated in planned manner. System testing consists of three different kinds of testing activities.

1. **α(Alpha) Testing:** It is the system testing performed by the development team in a controlled environment.
2. **β(Beta) Testing:** It is the system testing performed by friendly set of customers in an uncontrolled environment.
3. **Acceptance Testing:** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivery product.

Maintenance: Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 2
Route Management System (DTC Route Infomation)

Problem statement: It is an online information about the bus routes and their frequencies and fares. It shows all the functionalities of the bus booking and ticketing system. It also identifies internal data stores of payments, bus routes, customers, tickets, ticket counters that must be present in order for the bus ticket system to do its job and shows the flow of data among various paths of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing, it generates report of all bookings, all tickets, bus schedules and all payments.

Process Model:

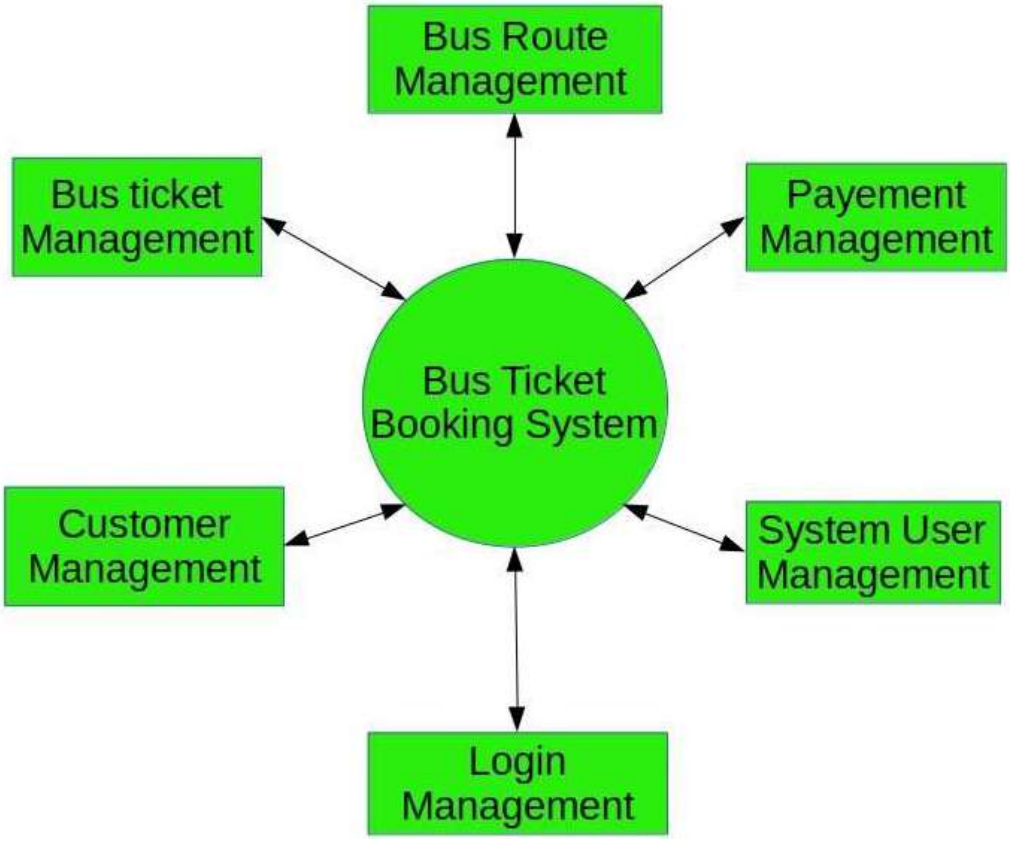
- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental model for the following reasons:
- Incremental model is used because the route infomation can vary from time to time, road condition, frequencies of passenger etc.
- Back tracking is possible in incemental model.

- **Requirement Analysis (SRS) :** As per the requirements of the transport route from client end. By analysis we realise that we need three modules.

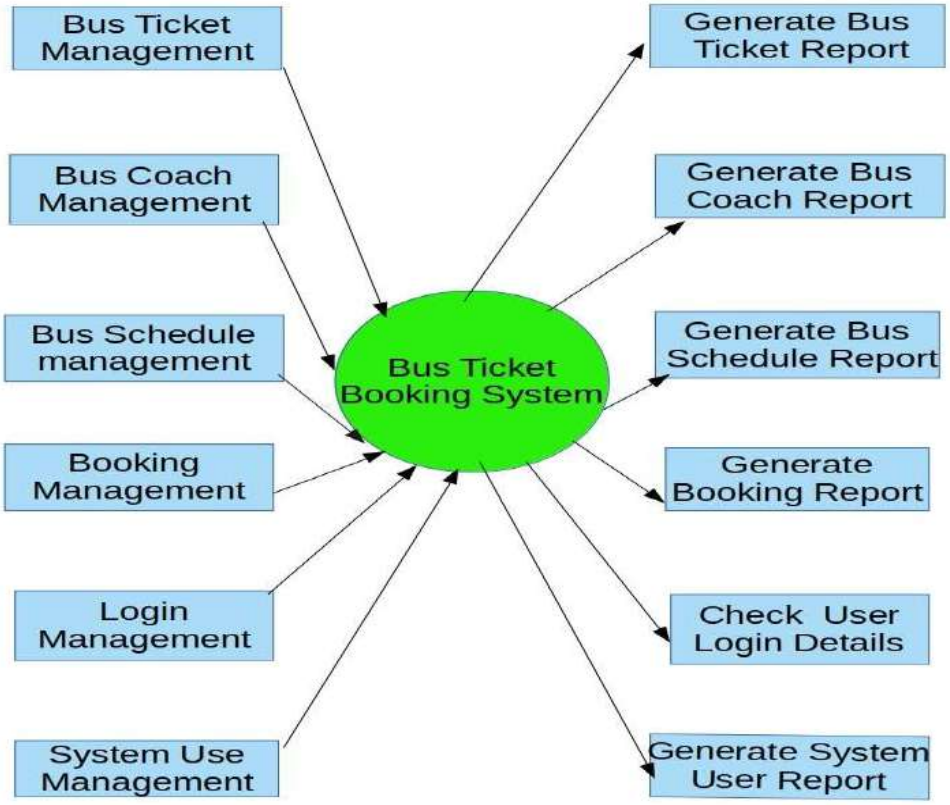
- **Passenger module :** In this module we add a function which will take the type of seat position required by the passenger as input and search if such seat is available for bording or not. If available, passenger details are taken and stored in the database. A unique token no. is also generated as output for passenger tichet identification. If the required seat is not available, an apology message is given output by software.
- **Ticket module :** At the time of ticket generation of a passenger, a function takes token no. as input and searches the lodging database and catering database for the expenses done by passenger during travelling in the bus. All the expenses and details related to that are given output as a bill. After bill payment, the record for that passenger is cleared.
- **Service module :** In this module we add a function for updating expenses made by passenger in the bus while travelling. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of travelling days by passenger accordingly charges.

- **Data Flow Diagrams:**

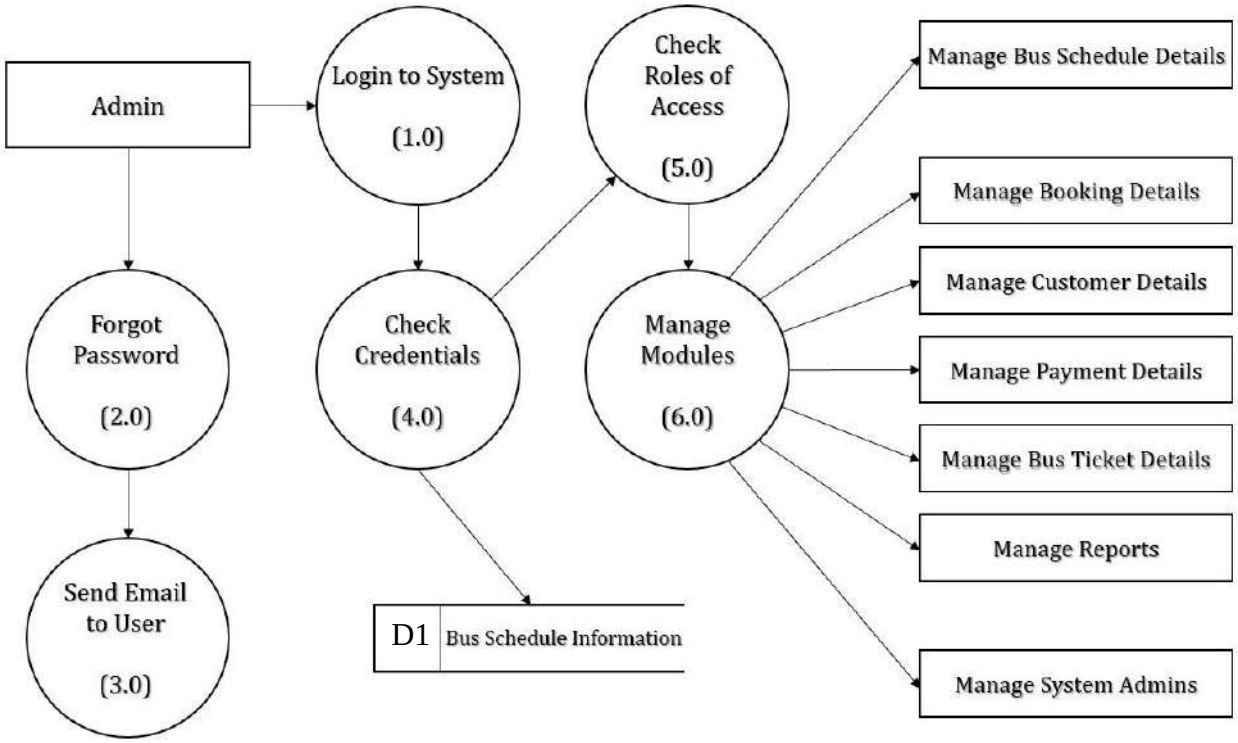
Level 0 DFD:



Level 1 DFD:



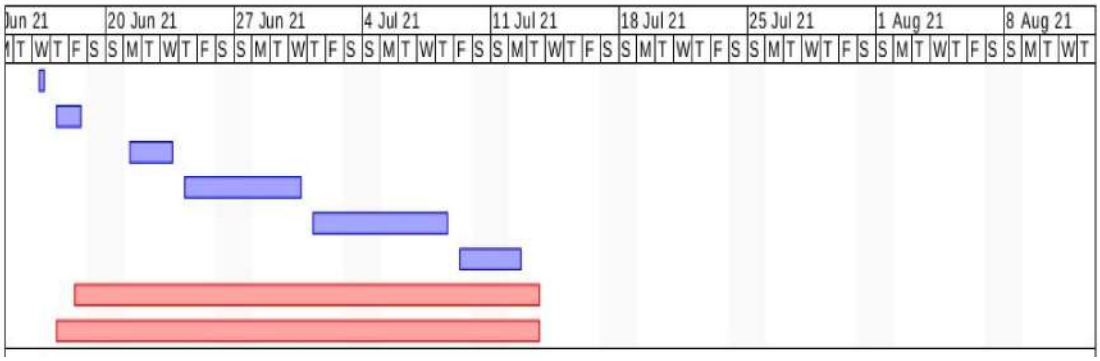
Level 2 DFD:



Project Management :

- **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	0.75 days	16/06/21, 10:00 AM	16/06/21, 5:00 PM
2		Feasibility Study	2 days	17/06/21, 8:00 AM	18/06/21, 5:00 PM
3		Requirement Analysis	3 days	19/06/21, 8:00 AM	23/06/21, 5:00 PM
4		Design	5 days	24/06/21, 8:00 AM	30/06/21, 5:00 PM
5		Coding & Unit Testing	6 days	01/07/21, 8:00 AM	08/07/21, 5:00 PM
6		Integration & System Testing	2 days	09/07/21, 8:00 AM	12/07/21, 5:00 PM
7		Maintainance	18 days	18/06/21, 8:00 AM	13/07/21, 5:00 PM
8		User Manual	19 days	17/06/21, 8:00 AM	13/07/21, 5:00 PM



b) **Risk Analysis:**

Risk management plan: Risks are measured by 2 parameters.

- Likelihood of a risk coming through (Probability of occurrence).
- Consequence of the problem associated with that risk.

Priority of risk is the multiplication of above two.

Risk leverage = (Risk exposure before reduction - Risk exposure before reduction) / (Cost of reduction).

Testing :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
  - **Alpha testing(Verification) :** It is done by the development team.
  - **Beta testing (Validation) :** It is done by a special group of friendly customers in an uncontrolled environment.
  - **Acceptance testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance :

Maintenance stands for all modifications and updations done after the delivery of software products. Route maintenance is the performance of adaptive, preventive and corrective maintenance for a given transport facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 3
Patient Appointment and Prescription Management System

Problem statement : Patient management system identifies internal data stores of reports, doctors, diseases, tests that must be present in order for the patient system to do its job. Each data store shows the flow of data among various paths of patients, tests, doctors, reports, diseases of the system. Patient records, medicine and drug records, test records are processed and reports are generated.

Process Model : Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects. It is similar to the incremental model with more emphasis placed on risk analysis. The spiral model has normally 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction begining at the core. Subsequent passes around the spiral model might be used to develope a prototype and then progressive sophisticated version of prototypes.

For developing this software, we will use Spiral Model for the following reasons:

- Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
- Spiral Model consists of risk analysis task region.
- In this model, we systematically prepare prototype and release for customer feedback.
- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Additional functionalities can be added at a later date.

- **Requirement Analysis:** Basically three modules are used in a patient appointment and prescription management system system. Users Registration Module, Profile Module, Doctor list Module, View Prescription Module, Upcoming Appointments Module. On basis of these five modules patient appointment and prescription management system can be designed.

- **Users Registration Module:** Registering the user by the details that includes Name, Username, Email Address, Password, Residential Address, Phone Number, Gender etc.
- **Profile Module:** Users can edit and update their profile including their password.
- **Doctor list Module:** Users can browse throw the entire list of doctors and select one for making appointment.
- **View Prescription Module:** Users can view, print or delete presciptions sent by doctors.

- **Upcoming Appointments Module:** Users can view/delete upcomming appointments. Past appointment are automatically removed from the list to show only the relevant contents.

◆ **Functional Requirements :**

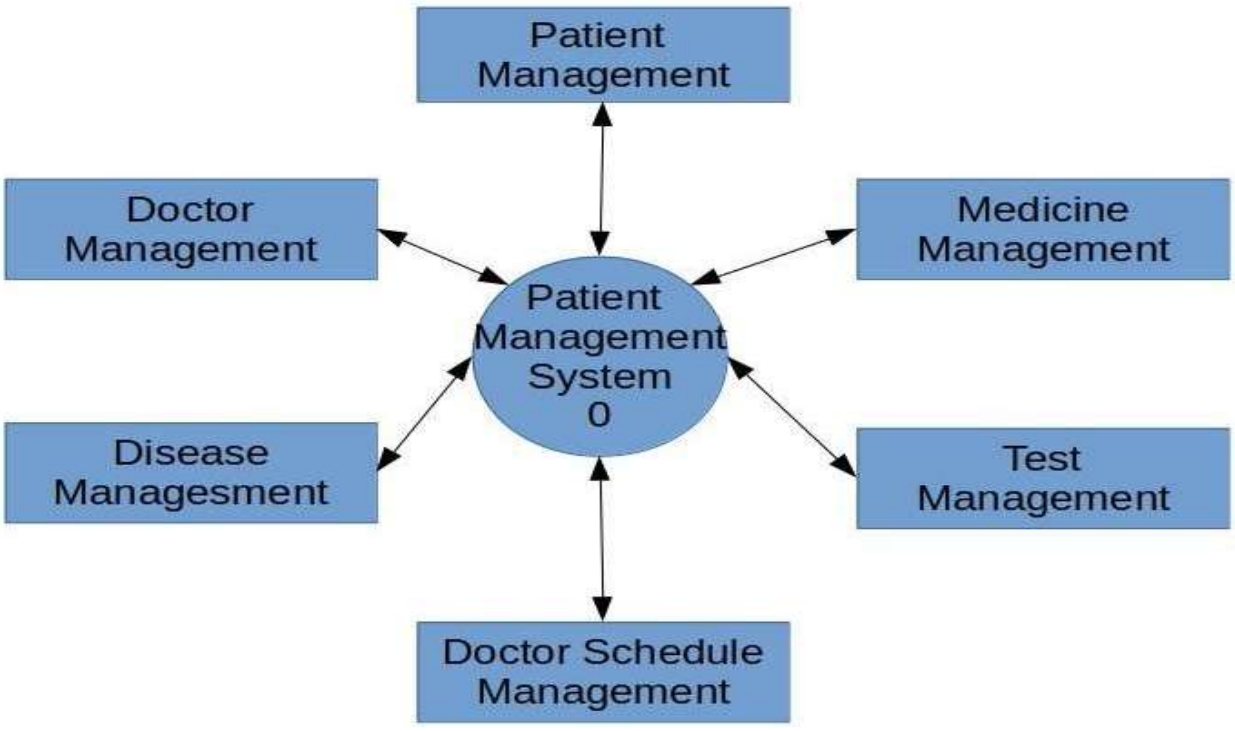
- It must allow input of patients data from the Reception and the Physician.
- It must request username and password for access to data, only after authentication the system will allow access.
- The system to be developed must operate 24 hours a day.
- The system to be developed shall display the correct time of day in compliance with ISO : 9000.
- The system to be developed must display the correct patient name.

◆ **Non Functional Requirements :**

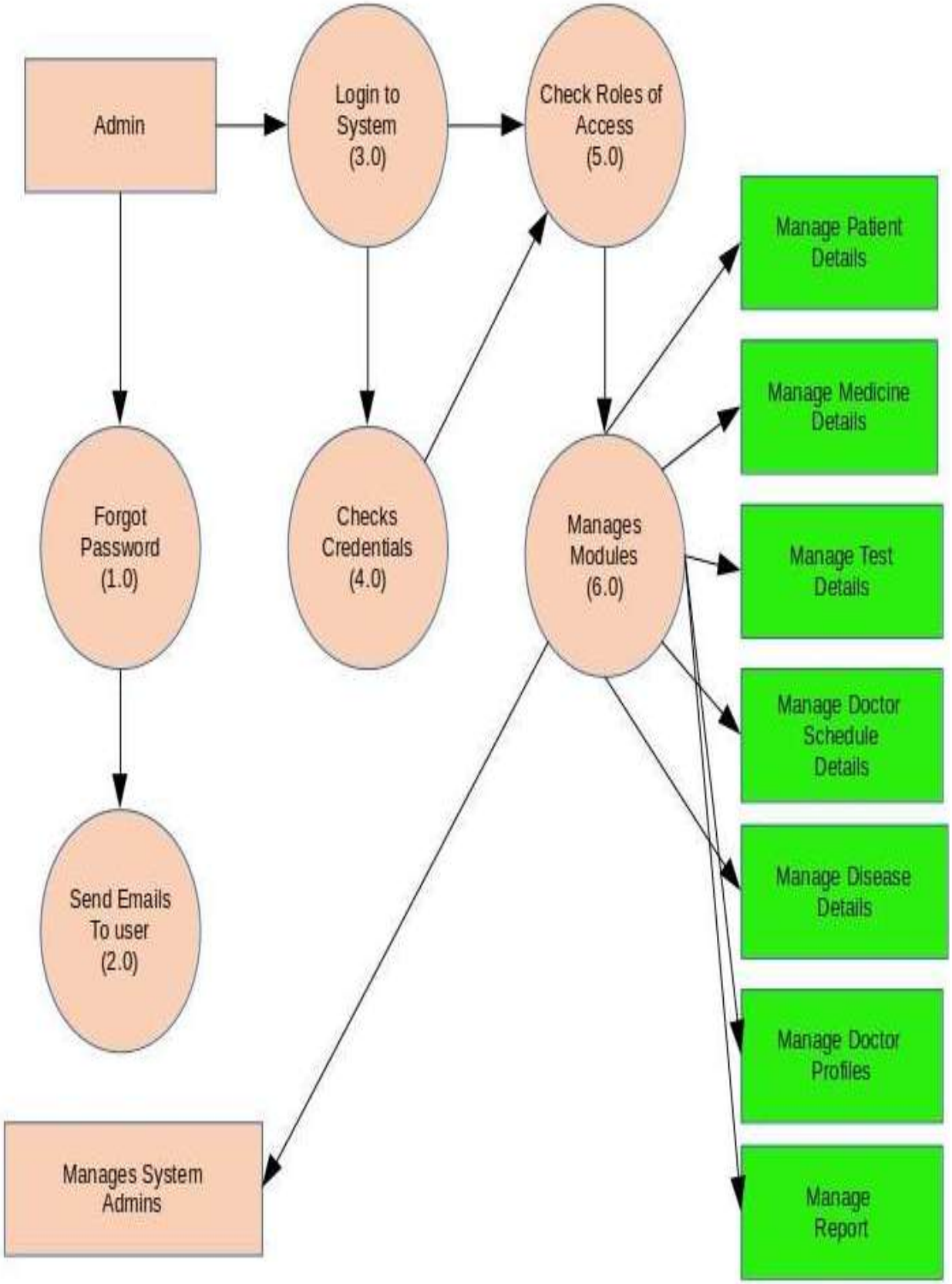
1. **Usability :** The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
2. **Portability :** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
3. **Robustness :** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure ortakes less recovery time.
4. **Availability :** The system shall be available all the time.

a) Data Flow Diagrams:

- **Zero Level DFD (Context Diagram) :**



- **Level One DFD:**

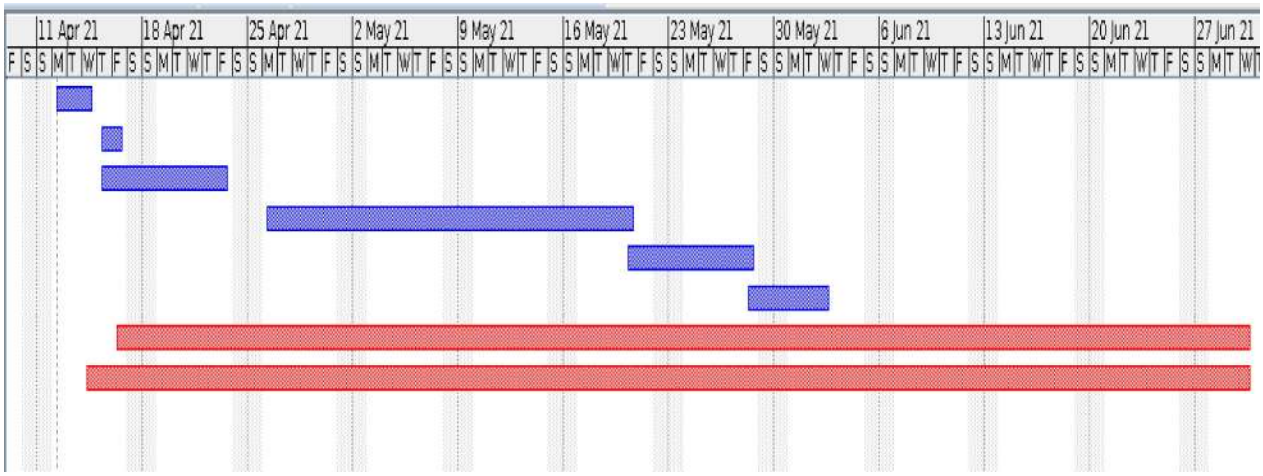


b) Data Dictionary Files : Data dictionary files are patients, presciption, schedule.

◆ **Project Management :**

- **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	3 days?	12/04/21, 8:00 AM	14/04/21, 5:00 PM
2		Feasibility Study	2 days?	15/04/21, 8:00 AM	16/04/21, 5:00 PM
3		Requirement Analysis and Specification	7 days?	15/04/21, 8:00 AM	23/04/21, 5:00 PM
4		Design	19 days?	24/04/21, 8:00 AM	20/05/21, 5:00 PM
5		Coding And Unit Testing	7 days?	20/05/21, 8:00 AM	28/05/21, 5:00 PM
6		Integetion and System Testing	4 days?	28/05/21, 8:00 AM	02/06/21, 5:00 PM
7		Maintenance	54 days?	16/04/21, 8:00 AM	30/06/21, 5:00 PM
8		User Manual	56 days?	14/04/21, 8:00 AM	30/06/21, 5:00 PM



➢ **Risk management plan :** Risks are measured by 2 parameters.

- Likelihood of a risk coming through (Probability of occurrence).
- Consequence of the problem associated with that risk.

◆ Priority of risk is the multiplication of above two.

Risk leverage =
(Risk exposure before reduction - Risk exposure before reduction) /
(Cost of reduction)

➢ **Testing :** Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways.

● **Unit Testing :** Each module is tested in isolation.

● **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.

◆ **Alpha testing(Verification) :** It is done by the development team.

◆ **Beta testing (Validation) :** It is done by a special group of friendly customers in an uncontrolled environment.

◆ **Acceptance testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance : Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 4
Online Hotel Reservation Service System

Problem Statement : Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to check out the hotel, automation software should generate the entire bill for the customer and balance amount payable.

Process Model:

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons:

  ○ Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
  ○ Spiral Model consists of risk analysis task region.
  ○ In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS) :

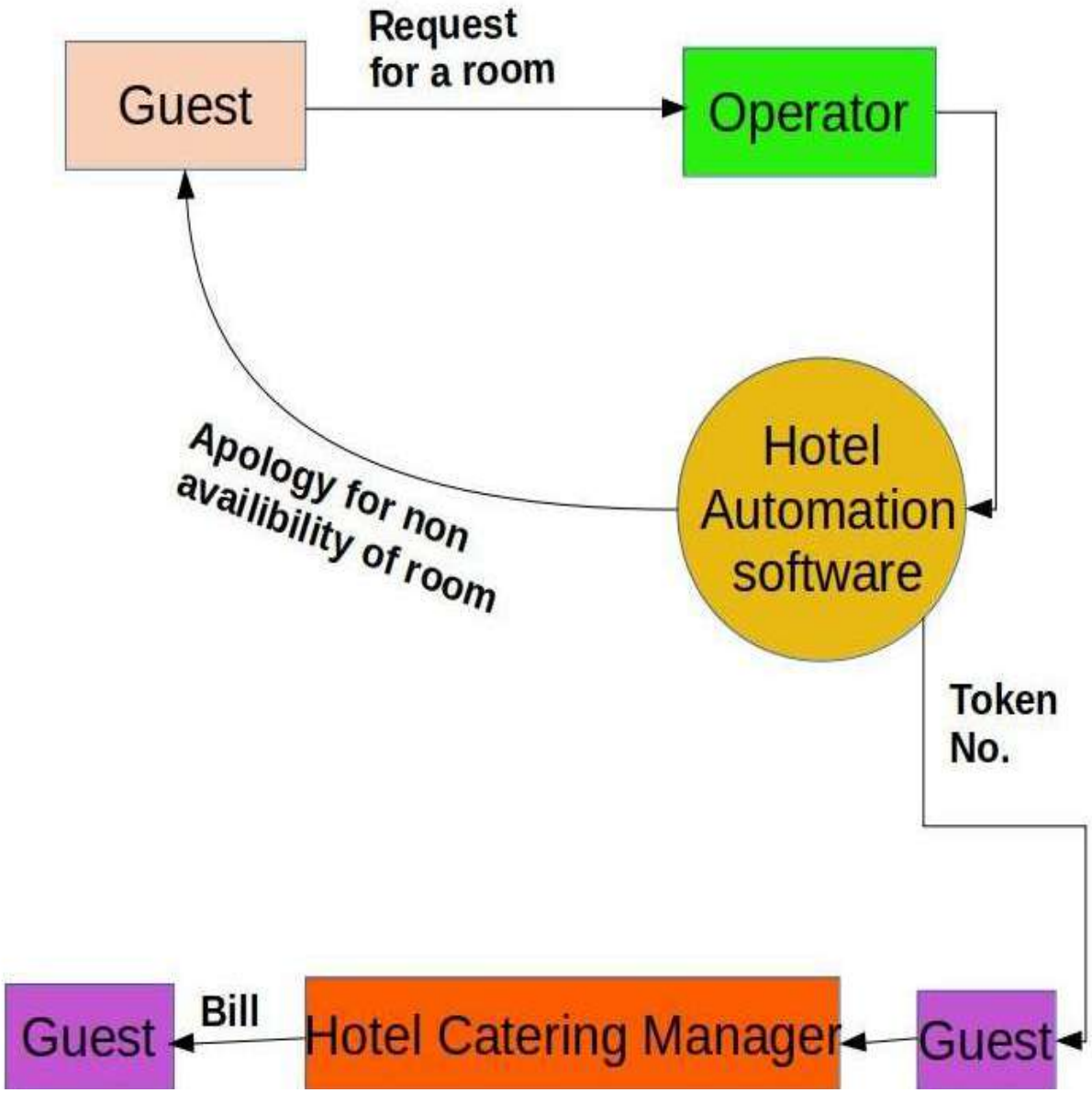
As per the requirements of the software from client end. By analysis we realise that we need three modules.

- **Guest module :** In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.
- **Stay module :** In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

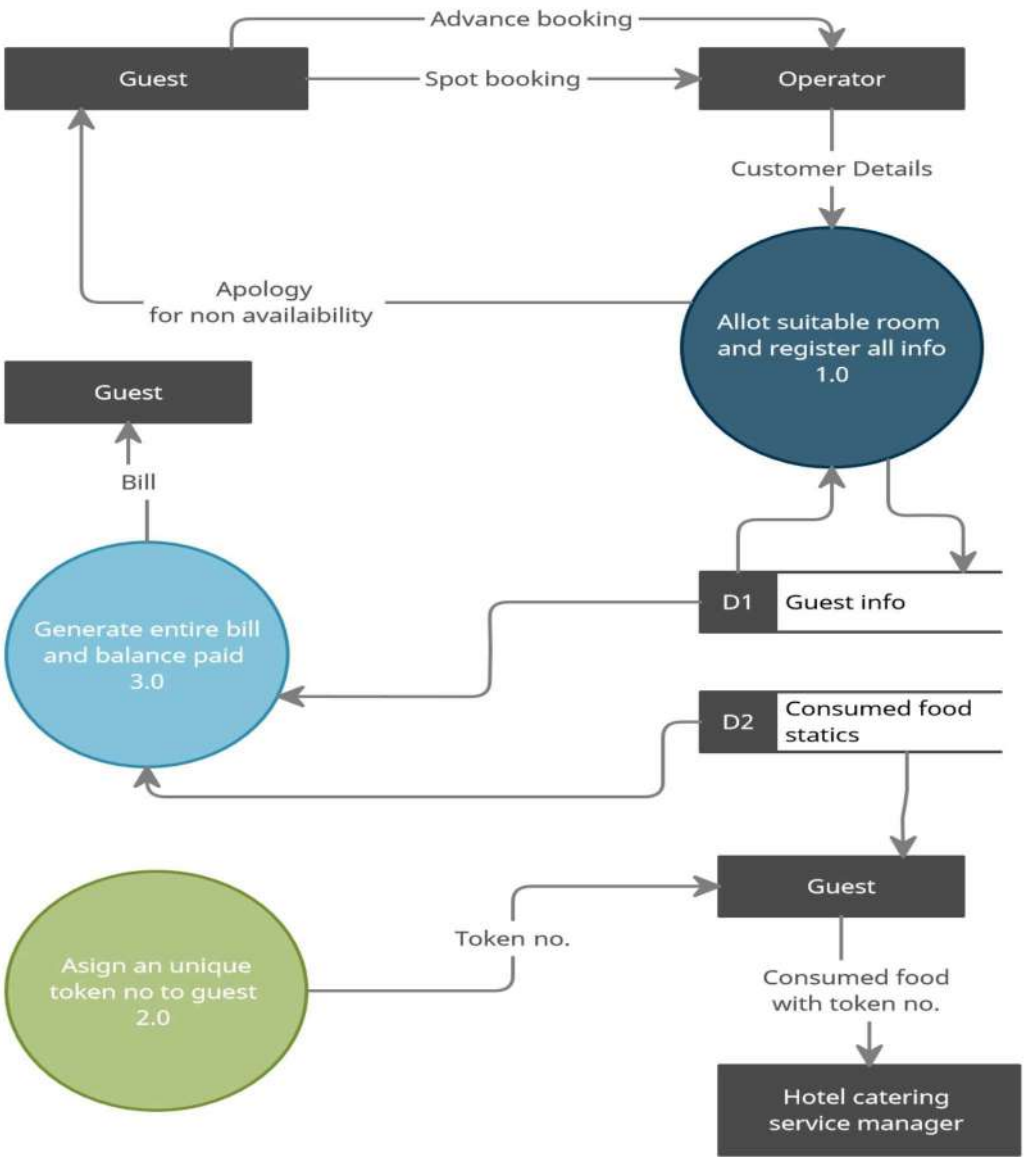
- **Catering module :** In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no.. Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.
- **Bill generation module :** At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

- **Data Flow Diagrams:**

Level 0 DFD:



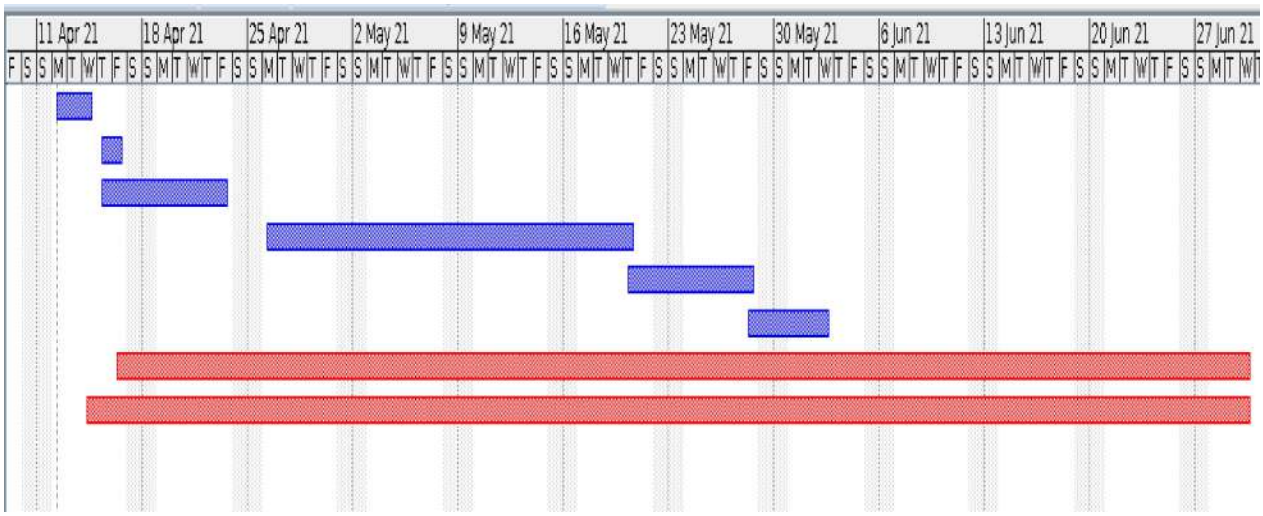
Level 1 DFD :



PROJECT MANAGEMENT :

- **Project Scheduling**

		Name	Duration	Start	Finish
1		Problem Statement	3 days?	12/04/21, 8:00 AM	14/04/21, 5:00 PM
2		Feasibility Study	2 days?	15/04/21, 8:00 AM	16/04/21, 5:00 PM
3		Requirement Analysis and Specification	7 days?	15/04/21, 8:00 AM	23/04/21, 5:00 PM
4		Design	19 days?	24/04/21, 8:00 AM	20/05/21, 5:00 PM
5		Coding And Unit Testing	7 days?	20/05/21, 8:00 AM	28/05/21, 5:00 PM
6		Integetion and System Testing	4 days?	28/05/21, 8:00 AM	02/06/21, 5:00 PM
7		Maintenance	54 days?	16/04/21, 8:00 AM	30/06/21, 5:00 PM
8		User Manual	56 days?	14/04/21, 8:00 AM	30/06/21, 5:00 PM



- **Project Resources :**
    - **Hardware Resources :** Several computer machines.
    - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, tester etc.
- **Risk management plan :** Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.

■ Priority of risk is the multiplication of above two.

- **Risk leverage =**
  (Risk exposure before reduction - Risk exposure before reduction)/ (Cost of reduction)

Testing :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways.

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
    - **Alpha testing(Verification) :** It is done by the development team.
    - **Beta testing (Validation) :** It is done by a special group of friendly customers in an uncontrolled environment.
    - **Acceptance testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 5
Criminal Record Management System

Problem Statement : Crime file management system identifies internal data store of court, criminals, prisoners, chargesheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court, of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model : Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects. For developing this software, we will use prototyping for the following reasons :

- Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
- Requires low up-front commitment.
- The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

Requirement Analysis (SRS) :
As per the requirements of the software from client end by analysis we realise that we need three modules.

- **Goal of the project :** The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.
- **Functional Requirements :** The five modules needed are.

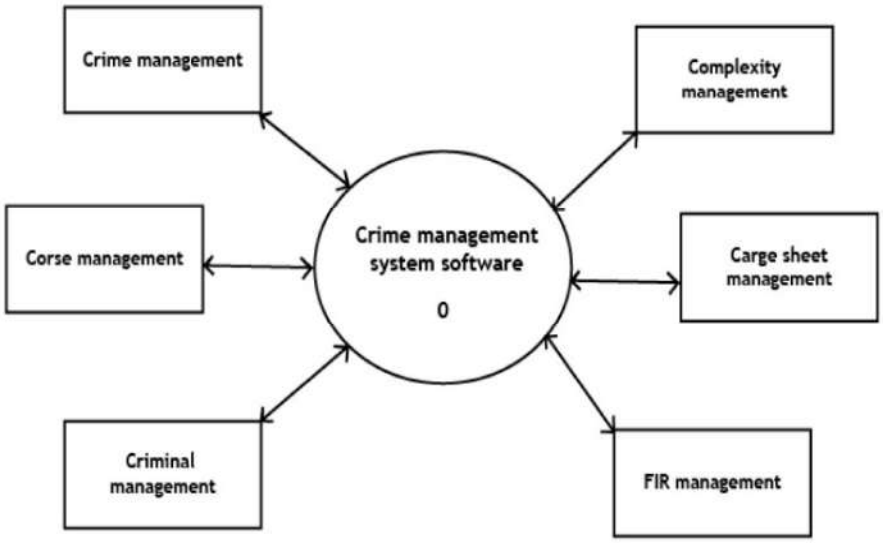
- **Administrator module :** This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
- **Criminal module:** This module accesses data from the database of criminal record details and manages it properly.
- **FIR module :** This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
- **Chargesheet module :** This module accesses data from the database of chargesheet record details and manages it properly.
- **Court module :** The main function of this module is to manage the court profiles.

- **Non-Functional Requirements :**

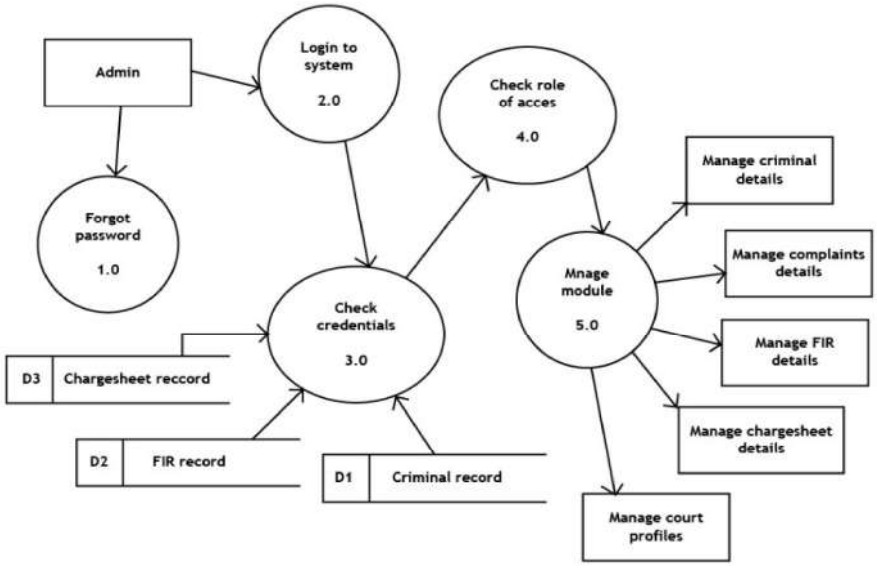
- **Usability :** The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
- **Portability :** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
- **Robustness :** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

➢ **Data flow diagrams :**

1. Level 0 DFD :



2. Level 1 DFD :



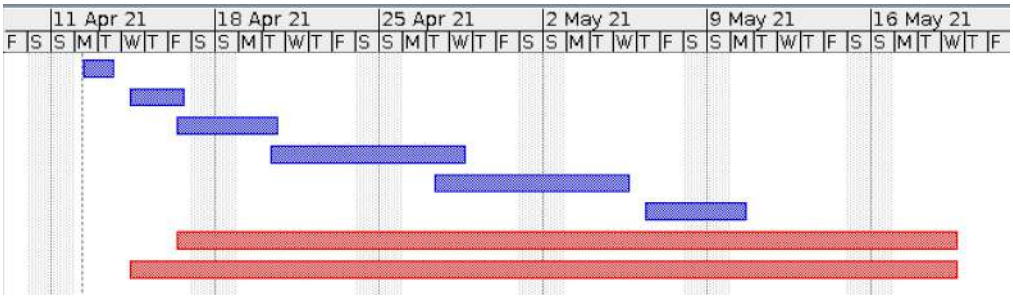
◆ **Data dictionary :** Data dictionary files are one file related to admin(include jailers, police officers, CBI officers), crime and one file related to complain (includes FIR, GD, chargesheet), criminal, court.

◆ **Project Management :**

● **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	1.75 days?	12/04/21, 10:00 AM	13/04/21, 5:00 PM
2		Feasibility study	2.75 days?	14/04/21, 10:00 AM	16/04/21, 5:00 PM
3		Requirement analysis a	2.75 days?	16/04/21, 10:00 AM	20/04/21, 5:00 PM
4		Design	6.75 days?	20/04/21, 10:00 AM	28/04/21, 5:00 PM
5		Coding and unit testin	6.75 days?	27/04/21, 10:00 AM	05/05/21, 5:00 PM
6		Integation and system	2.75 days?	06/05/21, 10:00 AM	10/05/21, 5:00 PM
7		Maintenance	23.75 da...	16/04/21, 10:00 AM	19/05/21, 5:00 PM
8		User manual	25.75 da...	14/04/21, 10:00 AM	19/05/21, 5:00 PM

● **Gantt Chart:**



■ **Risk management plan :** Risks are measured by 2 parameters.

1. Likelihood of a risk coming through (Probability of occurrence).
2. Consequence of the problem associated with that risk.

◆ Priority of risk is the multiplication of above two.

Risk leverage =
(Risk exposure before reduction - Risk exposure before reduction) /
(Cost of reduction)

➢ **Testing :** Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways.

● **Unit Testing :** Each module is tested in isolation.

● **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.

◆ **Alpha testing(Verification) :** It is done by the development team.

◆ **Beta testing (Validation) :** It is done by a special group of friendly customers in an uncontrolled environment.

◆ **Acceptance testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance : Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 6
Examination And Result Computation System (Session Management System)

Problem statement : Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are - processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model : Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
For developing this software, we will use Spiral Model for the following reasons:

- Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
- Spiral Model consists of risk analysis task region.

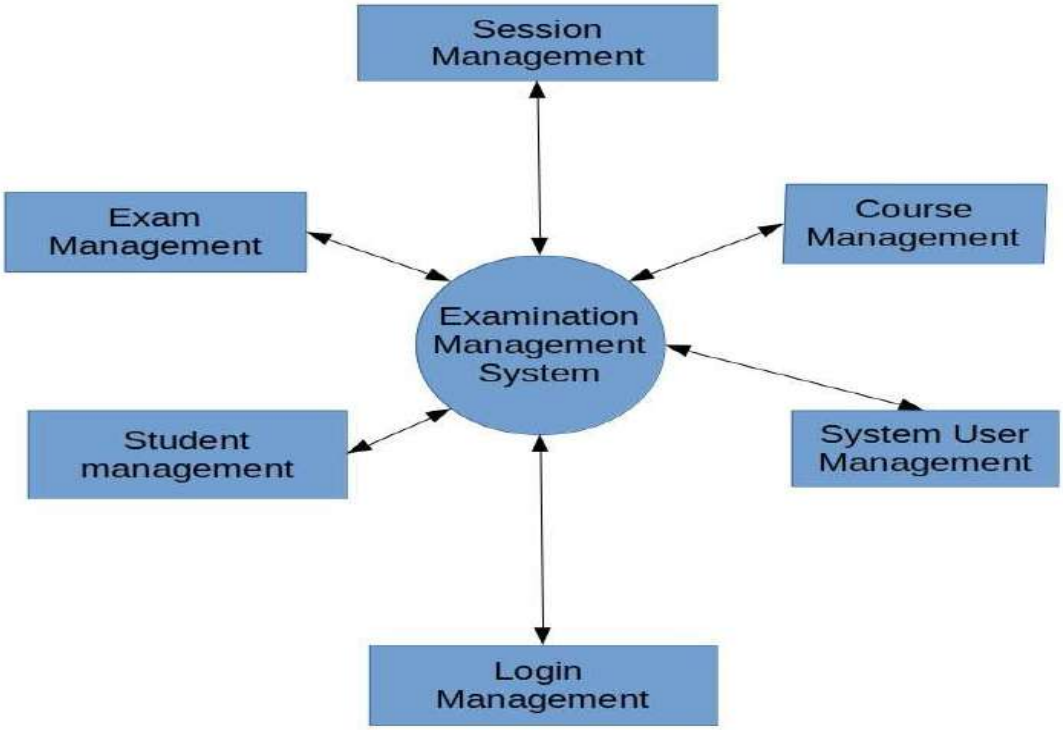
In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS) : Basically three modules are used in a examination management system. Course module, Student module, Exam Module. On basis of these three modules examination management software can be designed.

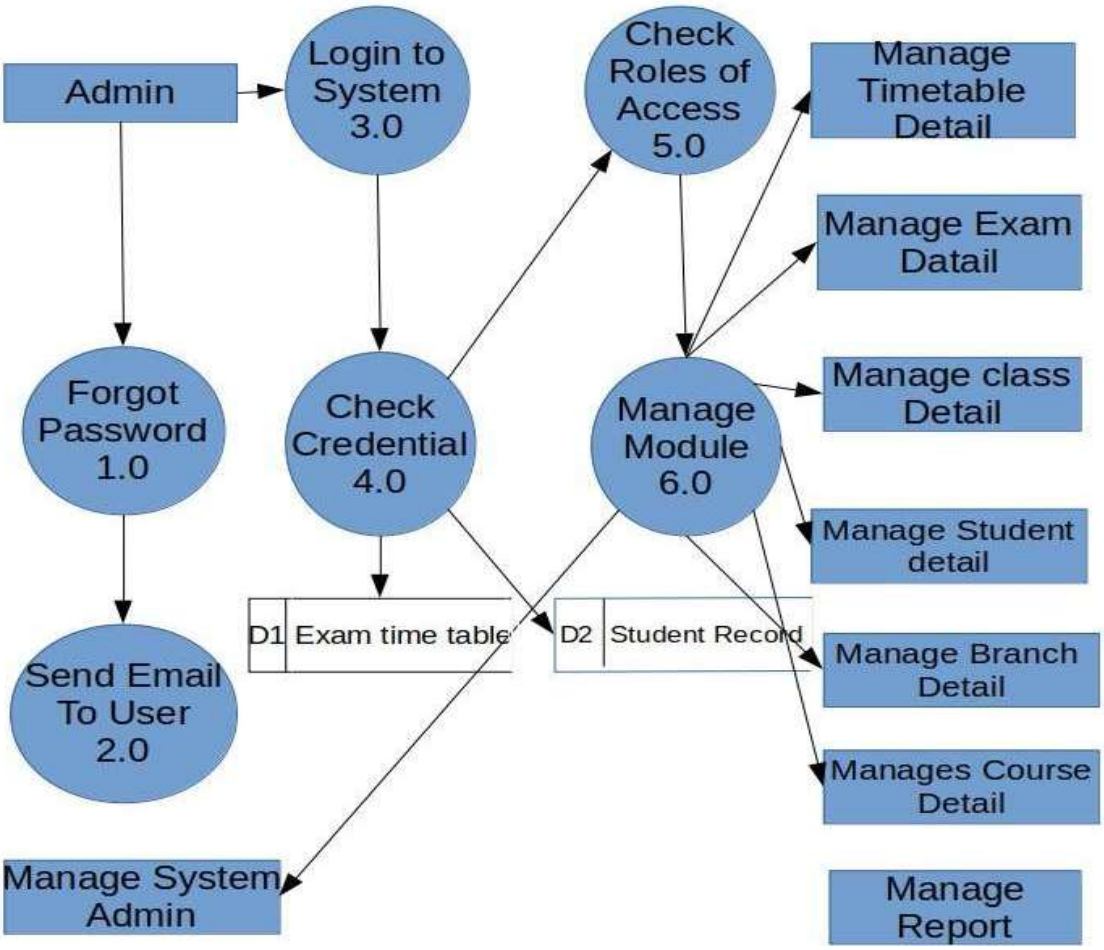
- **Course Module :** In this module the course structure database is designed. Various course information are stored like course start date, end date, duration of the course etc..
  - **Input:** Student name, date of birth, address, marksheet and all other necessary detail.
  - **Output:** Course title.
- **Student Module :** In this module student record is stored and processed. For instance student marks obtained, promotion, semester detail etc..
  Input: Student name.
  Output: Performance detail.
- **Exam Module :** In this module examination related information is stored. For instance exam time, examination question pattern, result etc.,.
  - **Input:** Time table
  - **Output:** Examination.

- **Data Flow Diagrams:**

◆ **Level 0 DFD :**



◆ **Level 1 DFD :**

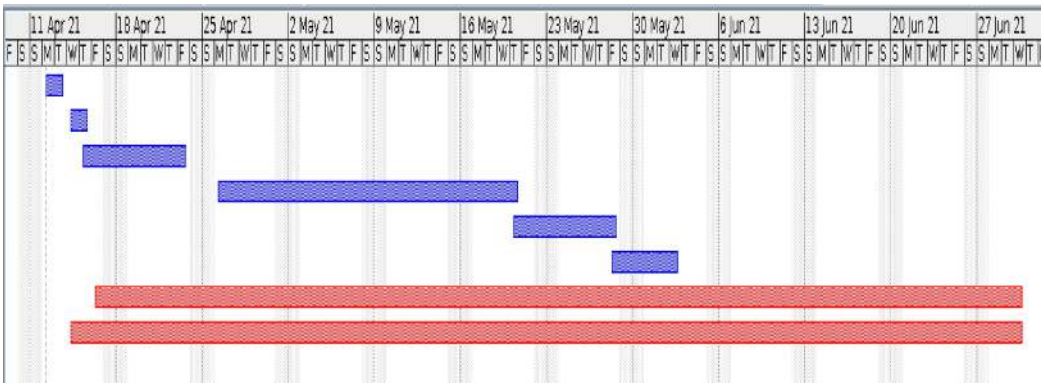


➢ **Project Management :**

◆ **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	2 days?	12/04/21, 8:00 AM	13/04/21, 5:00 PM
2		Feasibility Study	2 days?	14/04/21, 8:00 AM	15/04/21, 5:00 PM
3		Requirement analysis and Specification	7 days?	15/04/21, 8:00 AM	23/04/21, 5:00 PM
4		Design	19 days?	24/04/21, 8:00 AM	20/05/21, 5:00 PM
5		Coding And unit testing	7 days?	20/05/21, 8:00 AM	28/05/21, 5:00 PM
6		Integetion and System testing	4 days?	28/05/21, 8:00 AM	02/06/21, 5:00 PM
7		Maintenance	54 days?	16/04/21, 8:00 AM	30/06/21, 5:00 PM
8		User Manual	56 days?	14/04/21, 8:00 AM	30/06/21, 5:00 PM

◆ **Gantt chart:**



Risk Management plan :
Risk factor = (Risk exposure before reduction - Risk exposure after reduction) / Cost of reduction

Testing :

Unit Testing : Each component of the design is implemented as aprogram module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing : During the integration and system testing phase, the modules are integrated in planned manner. System testing consists of three different kinds of testing activities.

i. **α(Alpha) Testing :** It is the system testing performed by the development team in a controlled environment.
ii. **β(Beta) Testing :** It is the system testing performed by friendly set of customers in an uncontrolled environment.
iii. **Acceptance Testing :** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivery product.

Maintenance : Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project .: 7
Parking Allocation/Management System

Problem statement : Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in orderfor the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionality of parking system - processing of parking slots, vehicle records, parking fees, duration and generates the report of the same.

Process Model : Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
For developing this software, we will use Spiral Model for the following reasons:

- Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
- Spiral Model consists of risk analysis task region.

In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis : As per the requirements of the parking site from client end. By analysis we realise that we need three modules.

- **Data Record:**
- **Staff Records :** It helps to provide details of staff that uses the Vehicle parking management System. It provides the descriptions of staffs like:-Staff first, middle and last name-Address-Contact Number-Gender.
- **User Records :** This record helps for the authorization for using VehicleParking Management System. It Provides the Username and Password for the User(staff). It also includes the level of authority that means it separates the normal usersand administrator.
- **Vehicle Records :** This most important record which focuses in our VehicleParking Management System. It stores the essential Vehicle records like:- Vehicle Number-Vehicle Type-Vehicle Entry Time-Vehicle Exit Time.

- **Reports :**
- **Vehicle Parking Detail :** This report is very essential in this system. Thisreport provides a brief summary of vehicle activities. It shows the overall Entry and Exittime. It shows the User at time of Entry and Exit .It also provides the facility forexamining the total vehicle details according to date wise.
- **Transaction Detail :** This report will show the Transaction between thecustomer and the System. . It shows the cost of the vehicle after using the facility ofparking. It will show the number of transaction by date wise. It will also have User attime of the Transaction.

2. **User Requirement :**

i. Need for an application that makes communicating easy and comfortable.
ii. An application that enables user to park a vehicle with safe and secure.
iii. Need for an application that is easy to use and widely available and hence a webapplication
iv. Handling all functions done with organization in a computerized manner.
v. Allowing the user to park the vehicle directly.

3. Functional Requirement :

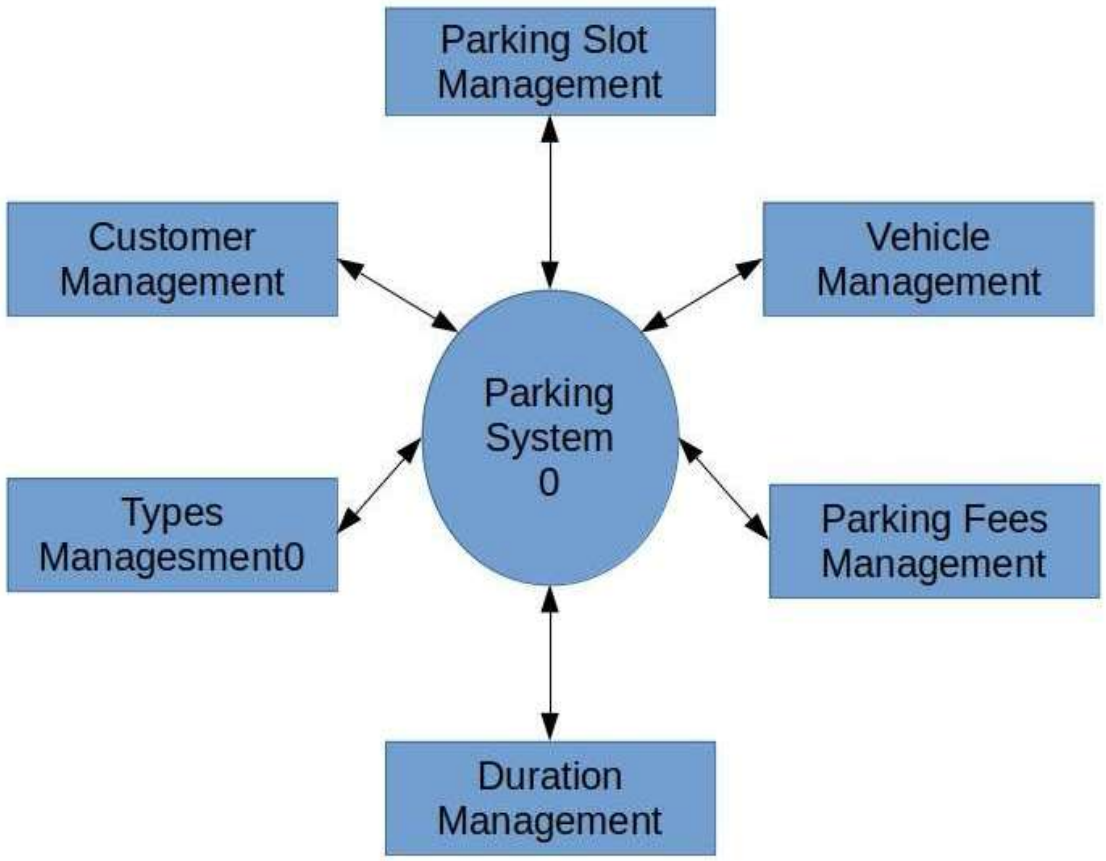
i. Admin need to enter all details for registration.
ii. Admin need to insert all details about customer and vehicle.
iii. Admin need to save all the details of customer and vehicle.
iv. Admin can retrieve the details of customer.
v. Admin must generate a report for payment.

4. Non-functional Requirement :

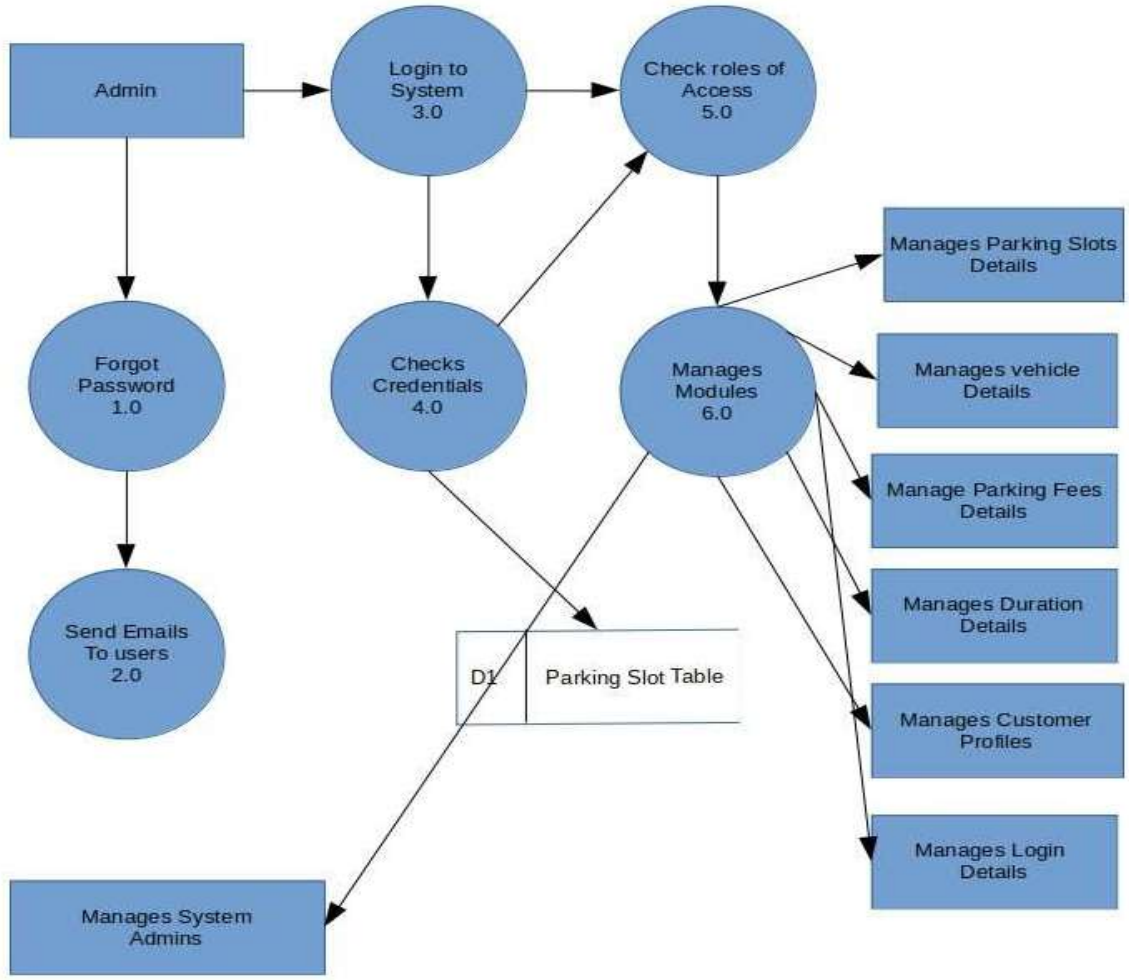
- **Usability :** These website has appropriate user interface and adequate informationto guide the user in order to use the website.
- **Portability:** The website is portable as it is online website running across the net.
- **Flexibility:** It is very flexible.
- **Security :** This website provide user and authentication so that only the legitimateuser are allowed to use the website.
- **Maintainability :** These website is capable to secure the data and easily retrievethe data.
- **Scalability :** These system can further modified in future.

◆ **Data Flow Diagram (DFD)**

• **Level 0 DFD:**



- **Level 1 DFD:**

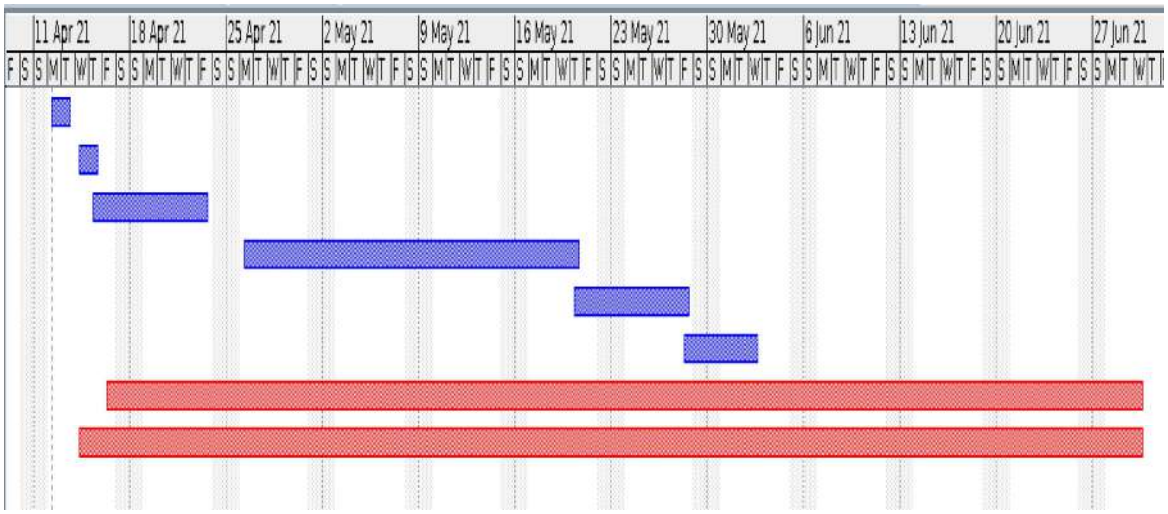


- **Data Dictionary Files :** Customer, Vehicle, Parking Slot (Includes Vehicle Type, Parking Fees and Duration).

  - **Project Management :**
  - **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	2 days?	12/04/21, 8:00 AM	13/04/21, 5:00 PM
2		Feasibility Study	2 days?	14/04/21, 8:00 AM	15/04/21, 5:00 PM
3		Requirement analysis and Specification	7 days?	15/04/21, 8:00 AM	23/04/21, 5:00 PM
4		Design	19 days?	24/04/21, 8:00 AM	20/05/21, 5:00 PM
5		Coding And unit testing	7 days?	20/05/21, 8:00 AM	28/05/21, 5:00 PM
6		Integetion and System testing	4 days?	28/05/21, 8:00 AM	02/06/21, 5:00 PM
7		Maintenance	54 days?	16/04/21, 8:00 AM	30/06/21, 5:00 PM
8		User Manual	56 days?	14/04/21, 8:00 AM	30/06/21, 5:00 PM

  - **Gantt Chart :**



- **Risk Analysis:**

◆ **Risk management plan:** Risks are measured by 2 parameters.

- Likelihood of a risk coming through (Probability of occurrence).
- Consequence of the problem associated with that risk.

◆ Priority of risk is the multiplication of above two.

Risk leverage = (Risk exposure before reduction - Risk exposure before reduction) / (Cost of reduction).

Testing :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.

a) **Alpha testing(Verification) :** It is done by the development team.

b) **Beta testing (Validation):** It is done by a special group of friendly customers in an uncontrolled environment.

c) **Acceptance testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance : The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

- **Corrective maintenance :** Correcting errors that were not discovered during the product development phase.
- **Perfective maintenance :** Enhancing the functionalities of the system according to the customer's requirements.
- **Adaptive maintenance :** Porting the software to work in a new environment.

Project .: 8
Wholesale Management System

Problem Statement : It provide all the functionality of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and generate report of the same.

1) Process Model : The 'V' model is a recognised standard for development of IT systems. The 'V' model for software development can be considered as an extention of the waterfall model. It give more emphasis on testing.There are three level of the 'V' model.

a) Software lifecycle process model
b) Methods to be used
c) Tools requirements

For developing this software, we will use 'V' Model for the following reasons:

- This is a highly-disiciplined model and phases are completed one at a time.
- Simple and easy to understand and use.
- Works well for smaller projects where requirement are very well understood.
- Easy to manage due to the rigidity of the model. Each phases has specific deliverarable and a review process.

2) Requirement Analysis (SRS) : The goal for the application is to manage the wholesale management function of the organization. Once it is automated all the functions can be effectively managed and the organization can achieve the competitive advantage. Business requirement are discussed with the following additional details:

- Helps to search the specific product and remaining stock.
- Details information about the product sales and purchase.
- Brief Information of the organization todays status in terms of news, number of present inventory as per the date entered.

As per the requirements of the software from client end. By analysis we realise that we need three modules.

- **Store Module :** It contains necessary items like products details, quantity, quality etc,.
- **Product Module :** It contains products information like product name, manufacturing date, expiry date etc,.
- **Stock Module :** It maintains a database of current product quantity, that will be renewed every months.

- **Functional Requirements :**

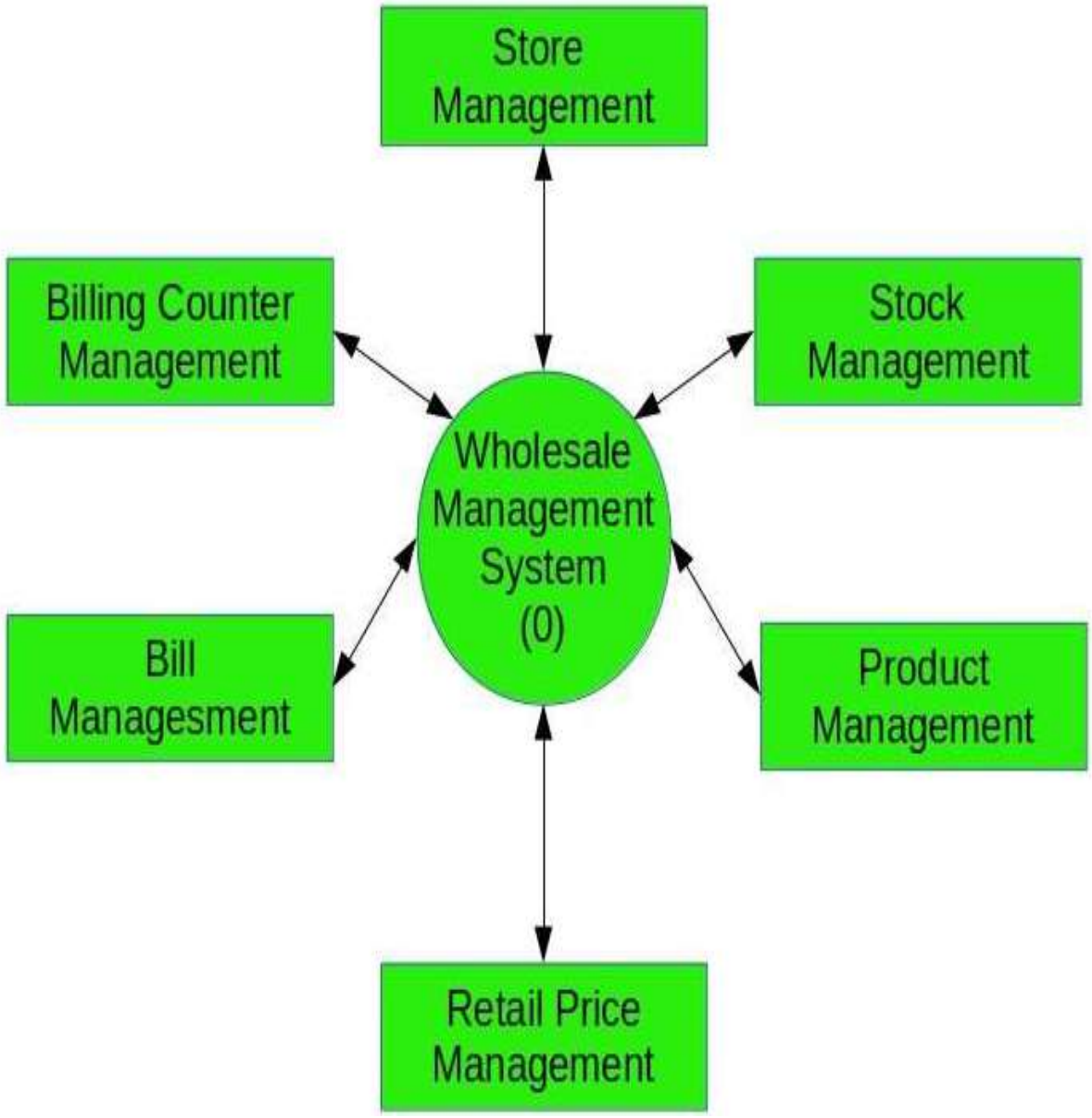
- System shall have a form to accept the customer details.
- System shall display transaction details.
- System should provide facility for change in address/name.
- System should automatically generate the bill.

- **Non Functional Requirements :**

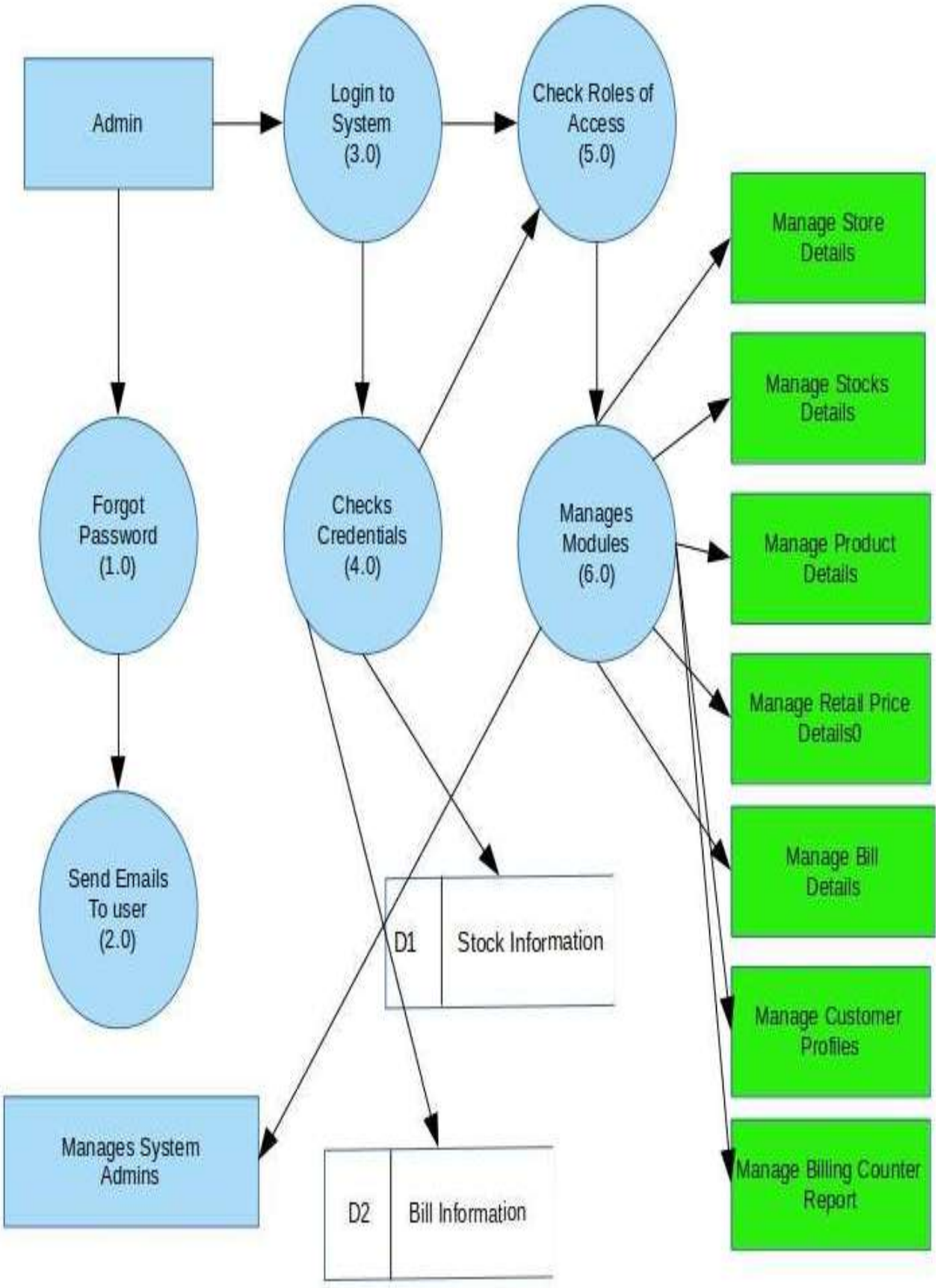
- All user manuals should be provided in the necessary format.
- Application should support 5 simultaneous users.
- There will be backup procedure to maintain records.
- Transaction should be complete wirthin $1/5^{th}$ of a second.

Data Flow Diagrams (DFD) :

- **Level One DFD :**

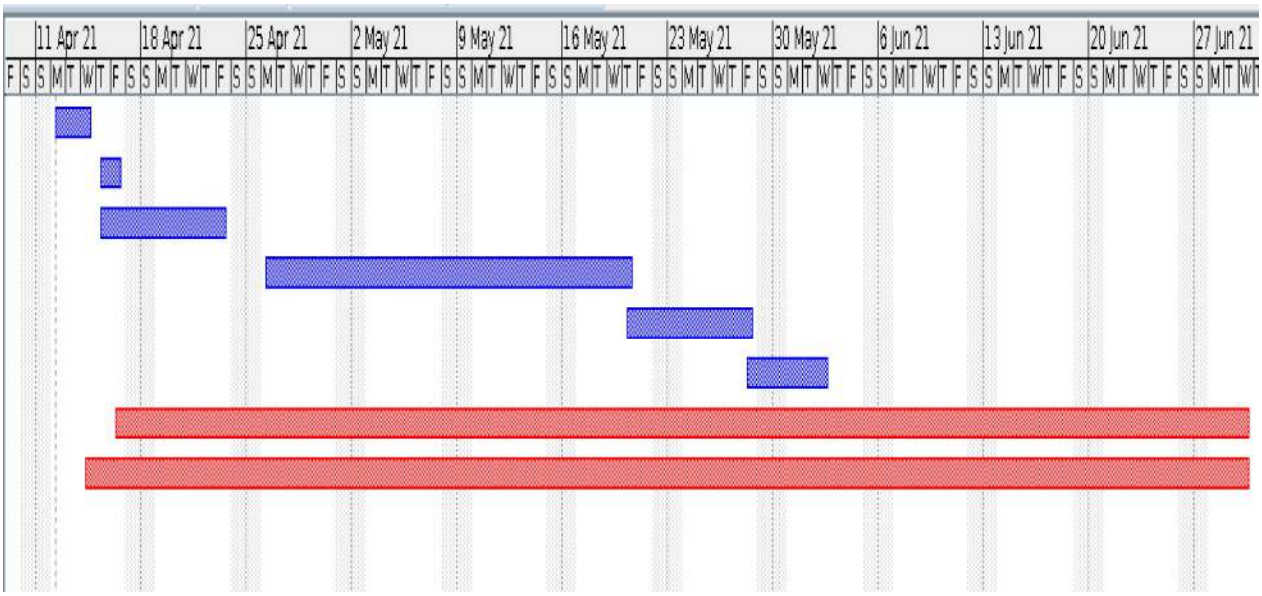


- **Level Two DFD :**



- **Project Management :**
  - **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem Statement	3 days?	12/04/21, 8:00 AM	14/04/21, 5:00 PM
2		Feasibility Study	2 days?	15/04/21, 8:00 AM	16/04/21, 5:00 PM
3		Requirement Analysis and Specification	7 days?	15/04/21, 8:00 AM	23/04/21, 5:00 PM
4		Design	19 days?	24/04/21, 8:00 AM	20/05/21, 5:00 PM
5		Coding And Unit Testing	7 days?	20/05/21, 8:00 AM	28/05/21, 5:00 PM
6		Integetion and System Testing	4 days?	28/05/21, 8:00 AM	02/06/21, 5:00 PM
7		Maintenance	54 days?	16/04/21, 8:00 AM	30/06/21, 5:00 PM
8		User Manual	56 days?	14/04/21, 8:00 AM	30/06/21, 5:00 PM



- **Risk Management Plan :** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
- Priority of risk is the multiplication of above two.

Risk leverage =
(Risk exposure before reduction - Risk exposure before reduction) /
(Cost of reduction)

Testing : Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways.

- **Unit Testing :** Each module is tested in isolation.
- **System Testing :** The modules are integrated and again tested. This time this testing is done in three parts.
  - **Alpha testing(Verification) :** It is done by the development team.

- **Beta testing (Validation) :** It is done by a special group of friendly customers in an uncontrolled environment.
- **Acceptance Testing :** It is performed by customers after product delivery to determine whether to accept or reject software.

Maintenance : The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.
Maintenance involves three kinds of activities.

- **Corrective maintenance :** Correcting errors that were not discovered during the product development phase.
- **Perfective maintenance :** Enhancing the functionalities of the system according to the customer's requirements.
- **Adaptive maintenance :** Porting the software to work in a new environment.

RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Debarpito Sinha
College Roll No : 733
Examination Roll No : 2021141308
Paper Code: CMSADSE3
Semester: V
Department: Computer Science
Supervisor: Dr. Biswajit Biswas

Project 1:

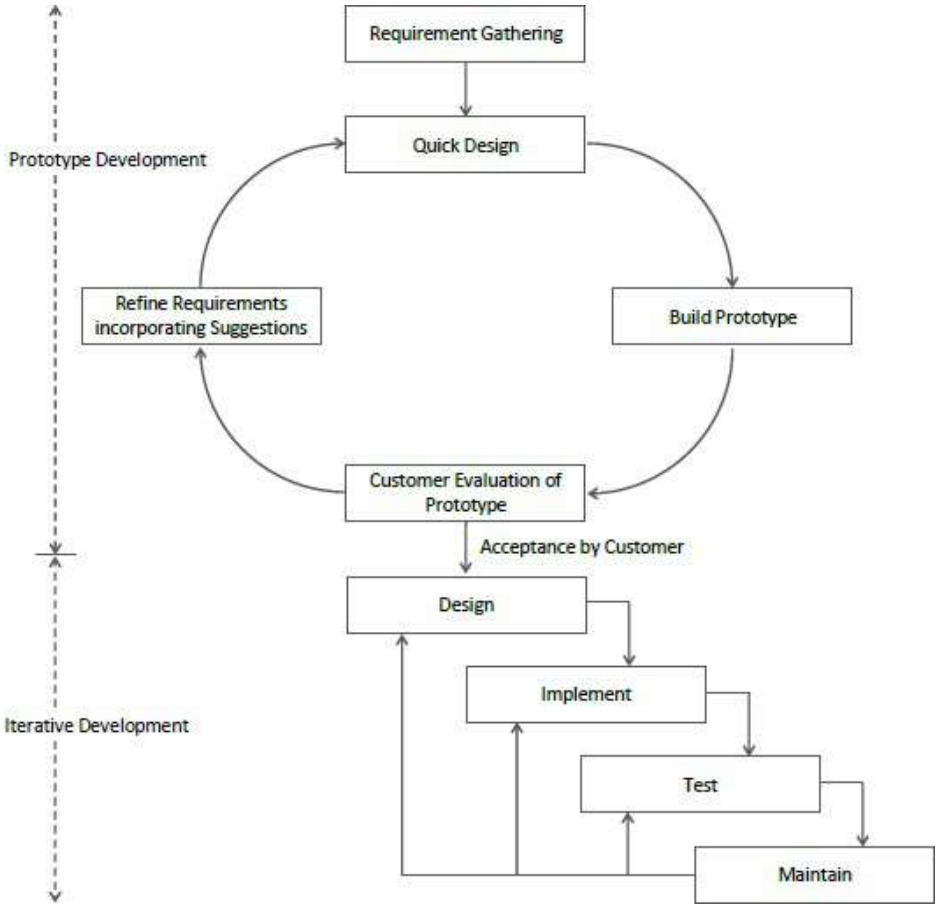
Organised Retail Shop Management System

Problem Statement:

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a "Customer Number" (CN). A customer can present 'CN' to customer staff when he makes purchases. In this case, the value of his purchase is credited against its 'CN'. At the end of the year, the supermarket intends to award surprise gifts to 10customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

Process Model:

Prototyping Model



Thismodelallowsallorpartofasystemtobeconstructedquickly.Goalofprototyping is to reduce the chance of uncertainty. The phases include –

1) meeting of the developers and the customers to define the overall objectives of the software
2) quick design leads to construction of prototype
3) evolution of prototype by customer
4) iteration occurs as the prototype is turned to the needs of the customers
5) the code for prototype is thrown away – the experience gathered from developing the prototypehelpsindevelopingtheactualsystem

Requirement Analysis (SRS):

As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account's purchasehistory. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

Functional Requirements

Asperthe requirements of the software from client end and analysis we realise that we need five modules:

1. **Customer Module:** In this module, a customer is registered and assigned a unique customer number(CN). A database is maintained for keeping the customer details.
   Input: Customer details
   Output: Unique CN

2. **Purchase Module:** In this module, any purchase made by the customer is credited against the CN.
   A database is maintained for keeping the purchase history.
   Input: Purchase made by the customer, CN
   Output: Updated database

3. **Promotional Module:** In this module, customers are selected for giving out the prizes and the prize winners' list is generated. The entries against the CN are reset after the generation of the prize winners' list.
   Input: Purchase history, CN
   Output: Prizewinners'list

Non-Functional Requirements

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

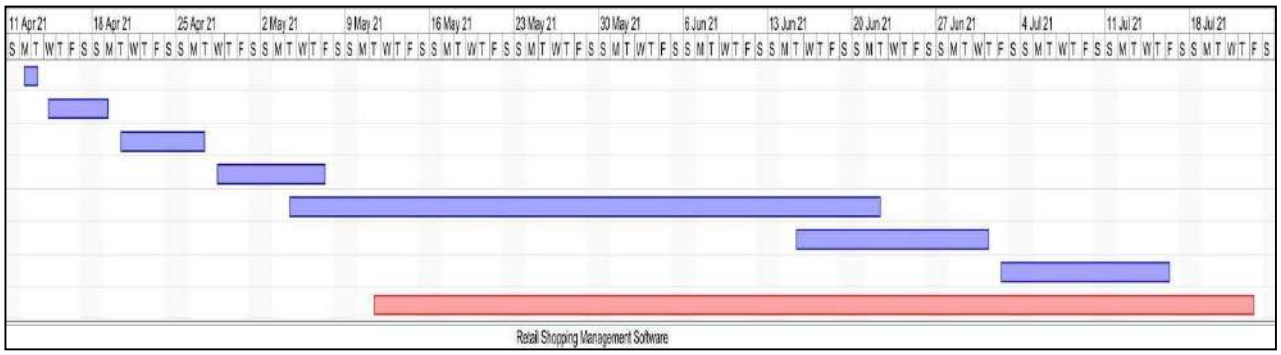
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

Project Scheduling:

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Project Resources:

1. **Hardware Resources:** Several computer machines.
2. **Human Resources:** Manpower resources like project manager, designer, analyst, programmer, tester etc.

Risk Management:

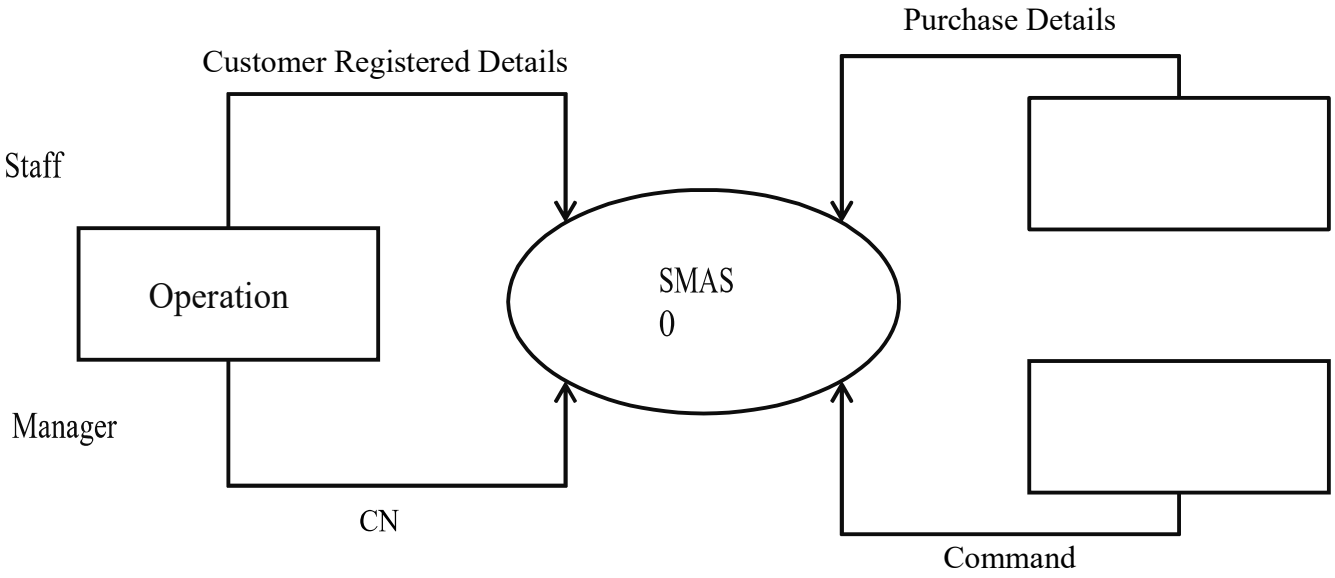
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

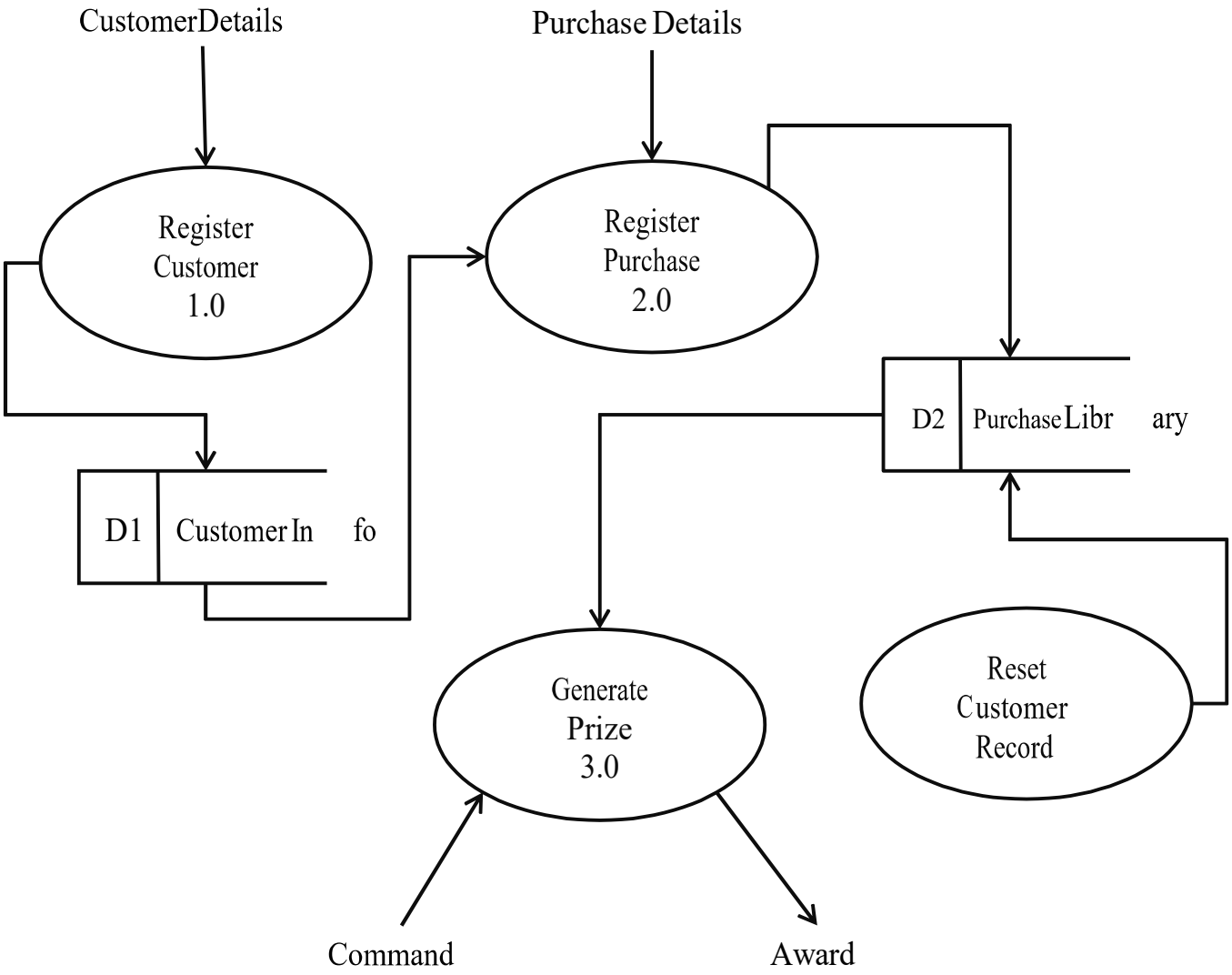
3. Risk leverage = $\frac{\qquad\qquad}{\qquad\qquad}$

Design Engineering:

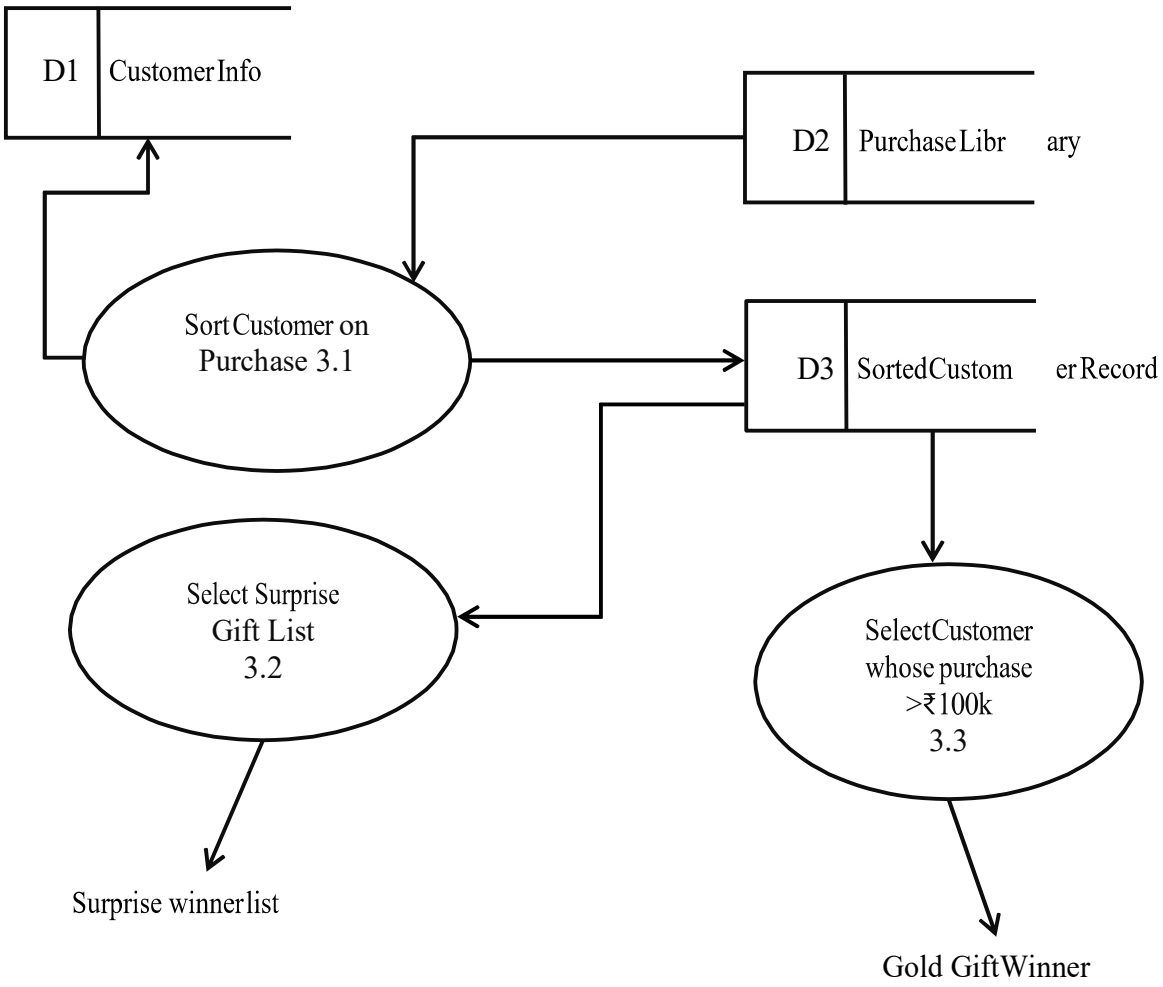
Level 0 DFD: Context Diagram



Level 1 DFD:



Level 2 DFD of Process 3.0:



Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team.
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition

Project 2
Route Management System (DTC)

Problem Statement:

It is the online information about the bus routes and their frequency and fairs. It shows all the functionality of the bus ticket booking system as a whole. It also identifies internal data stores of payment, bus route, customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between various parts of bus schedule, ticket counter, bus route, payment and customer of the system. Finally after processing it generates report of all booking, all ticket, bus schedule and all payment.

Process Model:

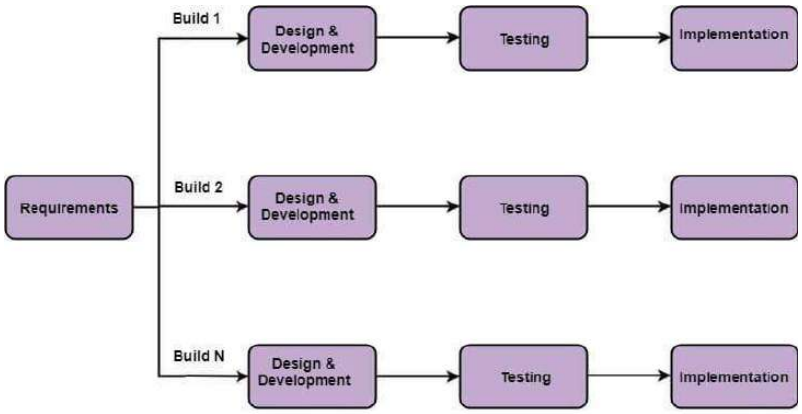


Fig: Incremental Model

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Requirement Analysis (SRS):

1. Goal of Project

The software is made to manage the various data stores related to route, payment, and schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. Functional Requirements

As per the requirements of the software from client end and analysis we realise that we need five modules:

A. **Admin module:** This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
Input: Username and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.
Input: Answer to security question
Output: Link for changing password

B. **Customer Module:** This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. **Schedule Module:** This module maintains timings of the bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. **Route Module:** This module has a function to query about bus routes and eventually know the stops it would cover during the journey. It also has a function which makes drivers know their routes.

E. **Report Module:** At the end of the journey, the admin (driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, and graphical data of passengers.

2. Non - Functional Requirements

A. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

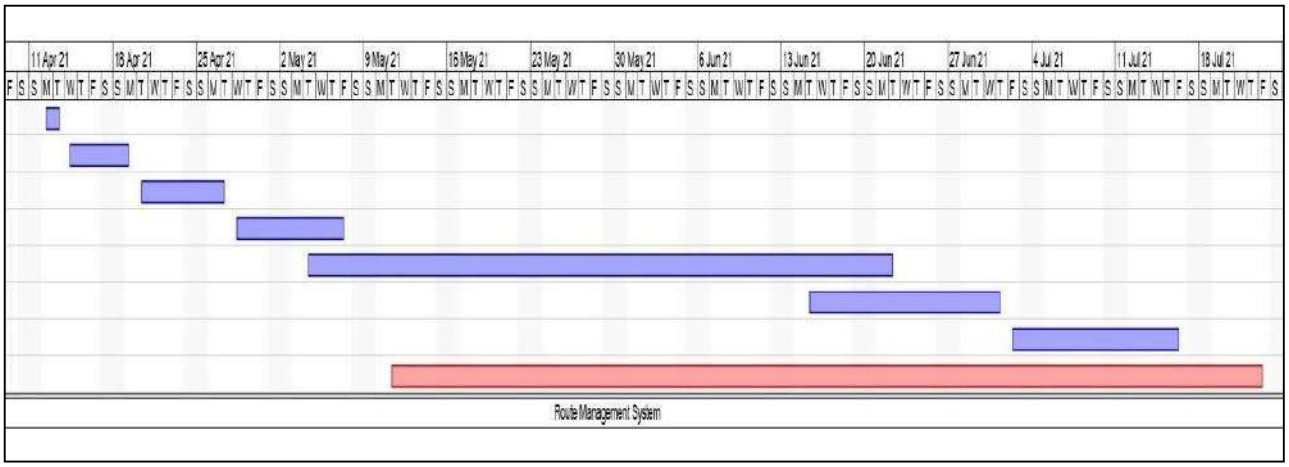
B. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

1. Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2. Project Resources

A. Man Power Resources:

- Project Manager
- System Analyst
- Designer
- Developer
- Tester
- Document Writer

B. Hardware and Software Resources

- Processor: Intel i3 4th gen or above
- Ram: 4Gb or above
- Java Development Kit – 13
- Java Runtime Environment
- Sublime Text 3 (Text Editor)
- Ms –Project
- Creately
- Google Docs

3. Risk Management Plan

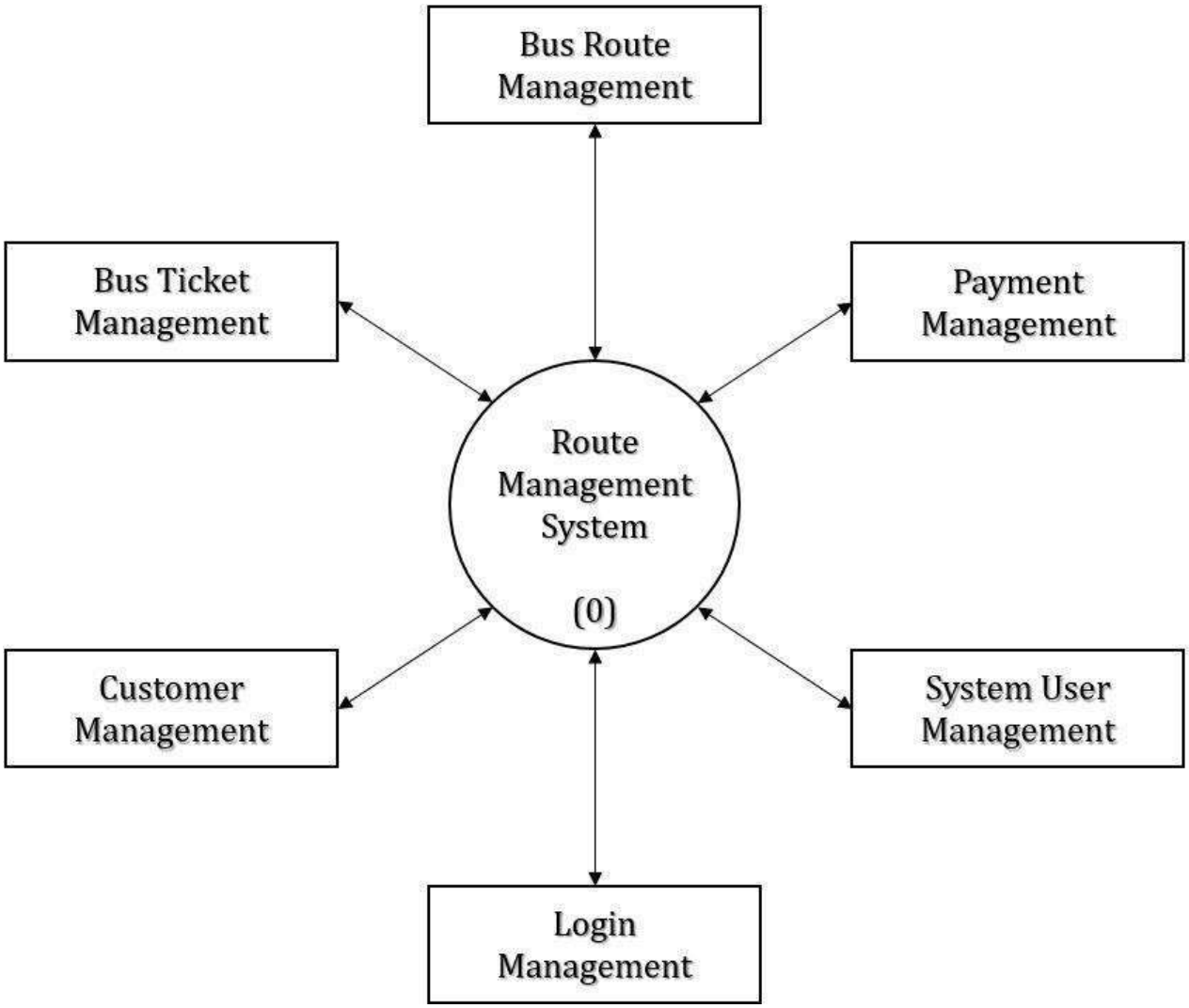
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

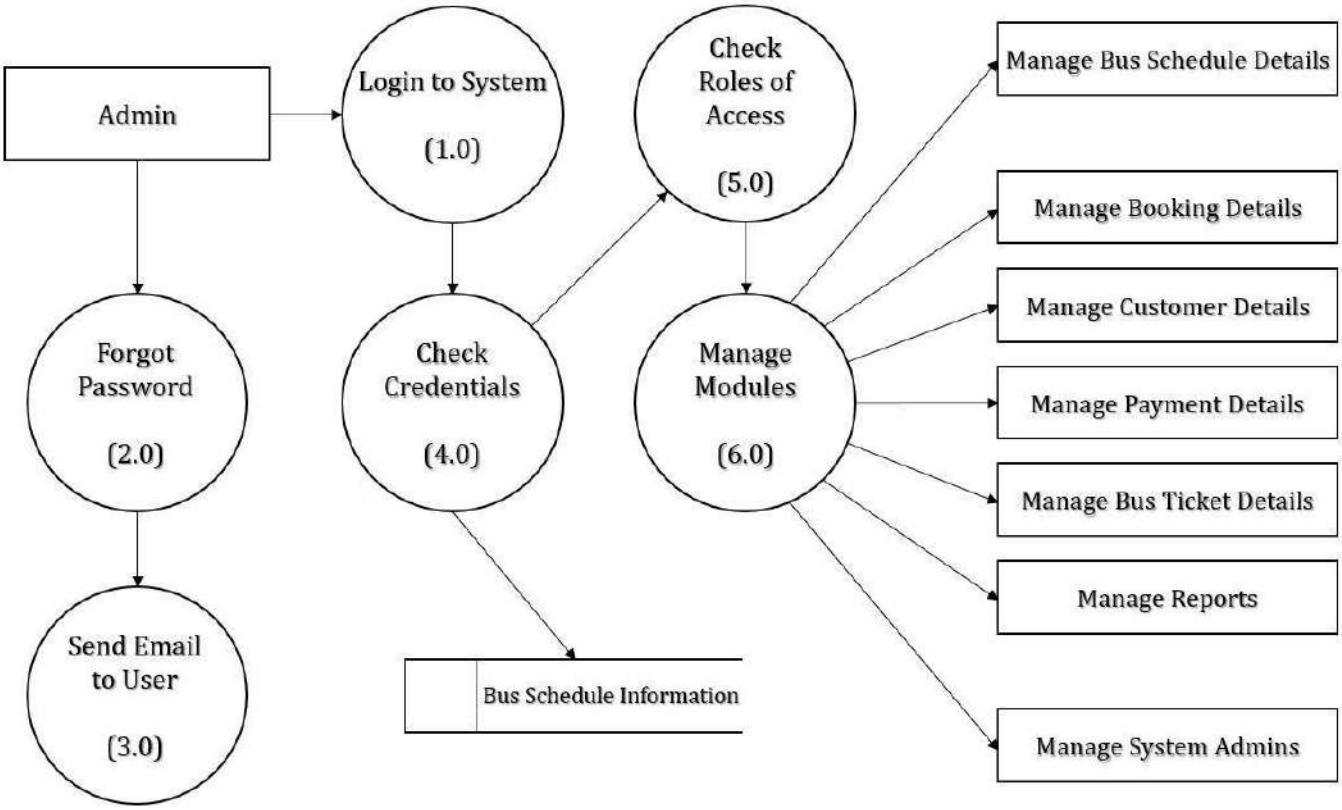
3. Risk leverage = _

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD:



Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project 3:

Patient Appointmentand Prescription Management System

Problem Statement:

Patient management system identifies internal data store of report, doctors, disease, test that must be present in order for the patient system to do its job. It shows the flow of data between various parts of patient, test, doctor, report, disease of the system. Patient records, medicine records, test records are processed and reports are generated.

Process Model:

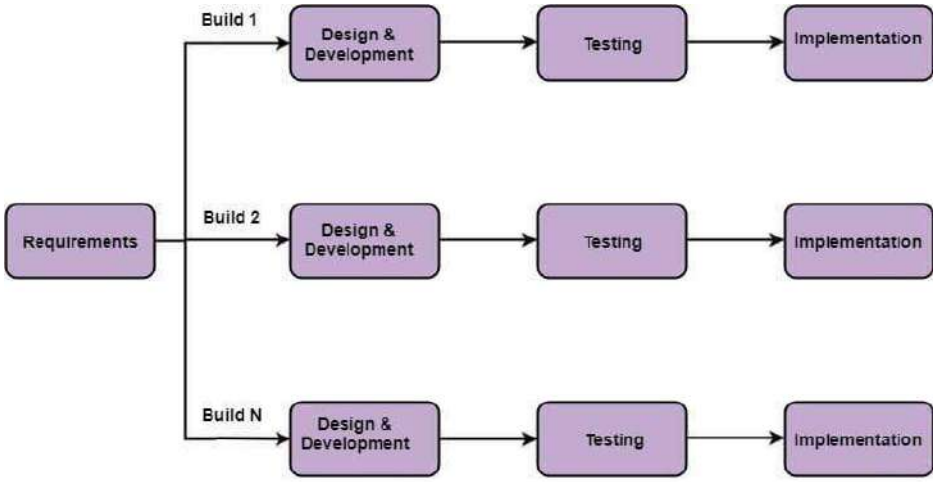


Fig: Incremental Model

This life cycle model is also referred to as successive versions model. In this life cycle model, the software is first broken down into several modules which can be incrementally constructed and delivered. The development team first develops the core modules of the system. This initial product skeleton (A) is refined into increasing levels of capability by adding new functionalities. Each evolutionary version can be developed by iterative waterfall model. The user gets a chance to experiment with the partially developed software much before the complete version of the system is released.

Requirement Analysis (SRS):

1. Goal of Project

The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital staff can also take care of medicines required and accordingly order before stock runs out in hospital.

2. Functional Requirements

As per the requirements of the software from client end and analysis we realise that we need five modules:

A. **Admin module:** This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

B. **Doctor Module:** This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.

C. **Report Module:** This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.

D. **Diagnostic Module:** This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.

2. Non - Functional Requirements

A. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

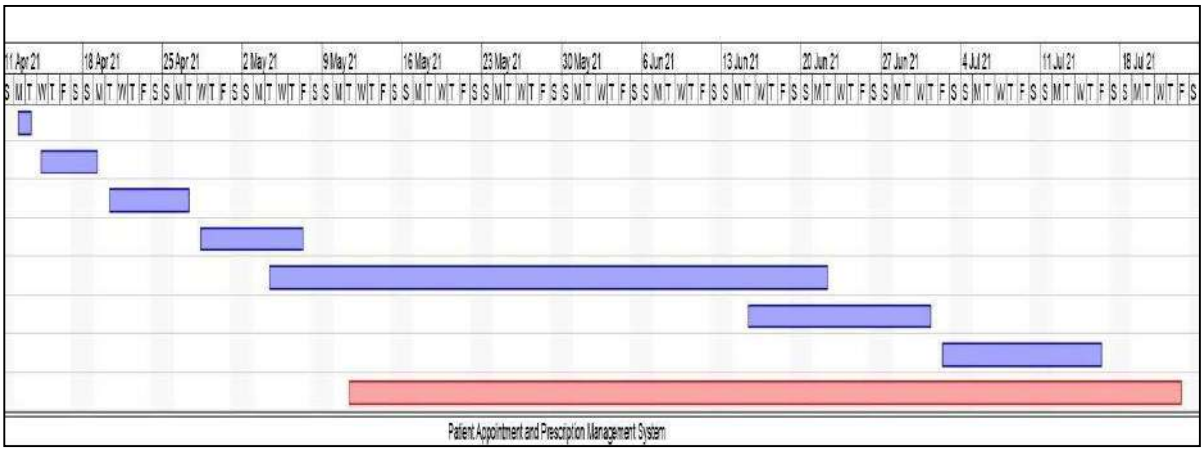
B. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

1. Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2. Project Resources

A. Man Power Resources:

- Project Manager
- System Analyst
- Designer
- Developer
- Tester
- Document Writer

B. Hardware and Software Resources

- Processor: Intel i3 4th gen or above
- Ram: 4Gb or above
- Java Development Kit – 13
- Java Runtime Environment
- Sublime Text 3 (Text Editor)
- Ms –Project
- Creately
- Google Docs

3. Risk Management Plan

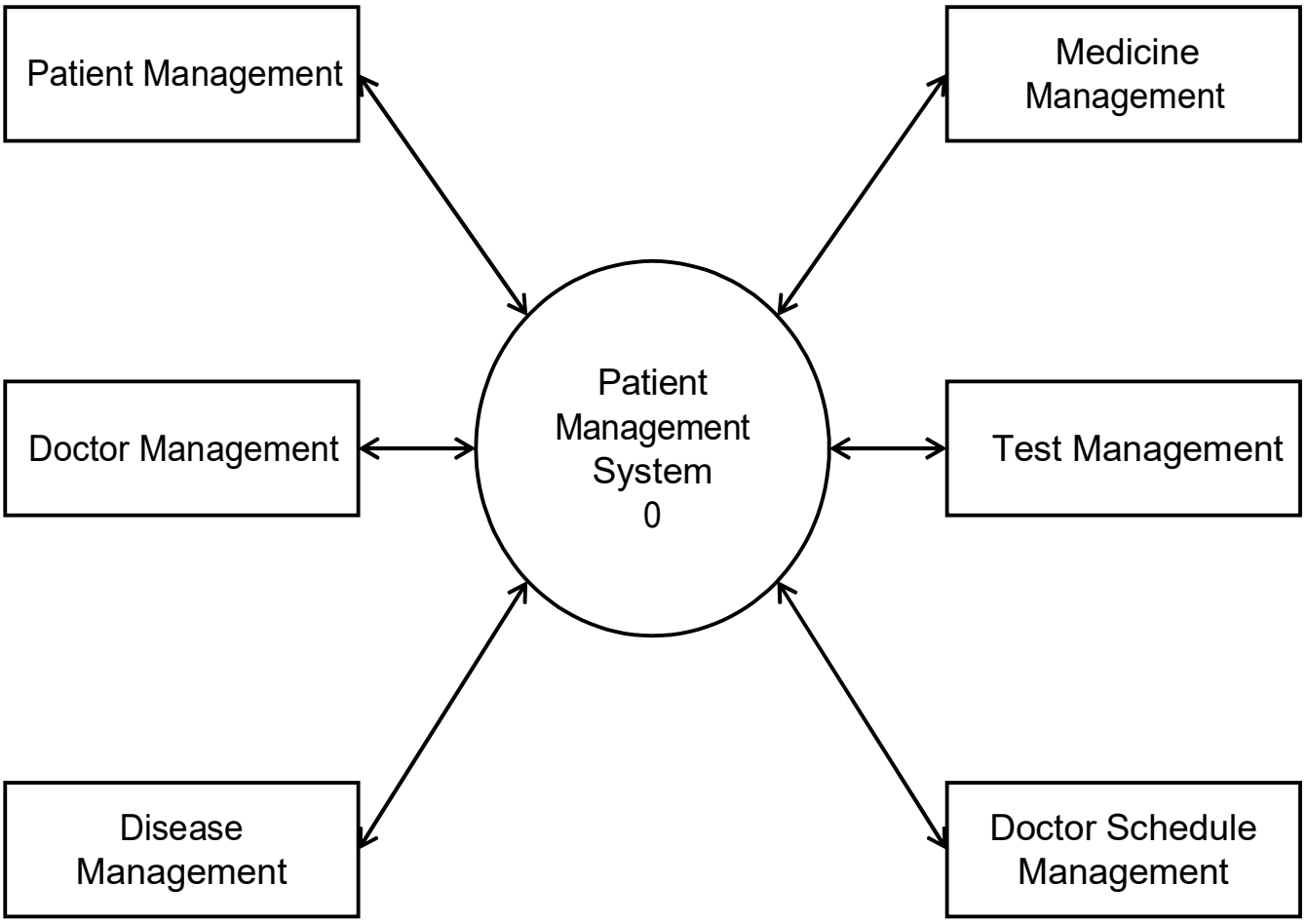
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

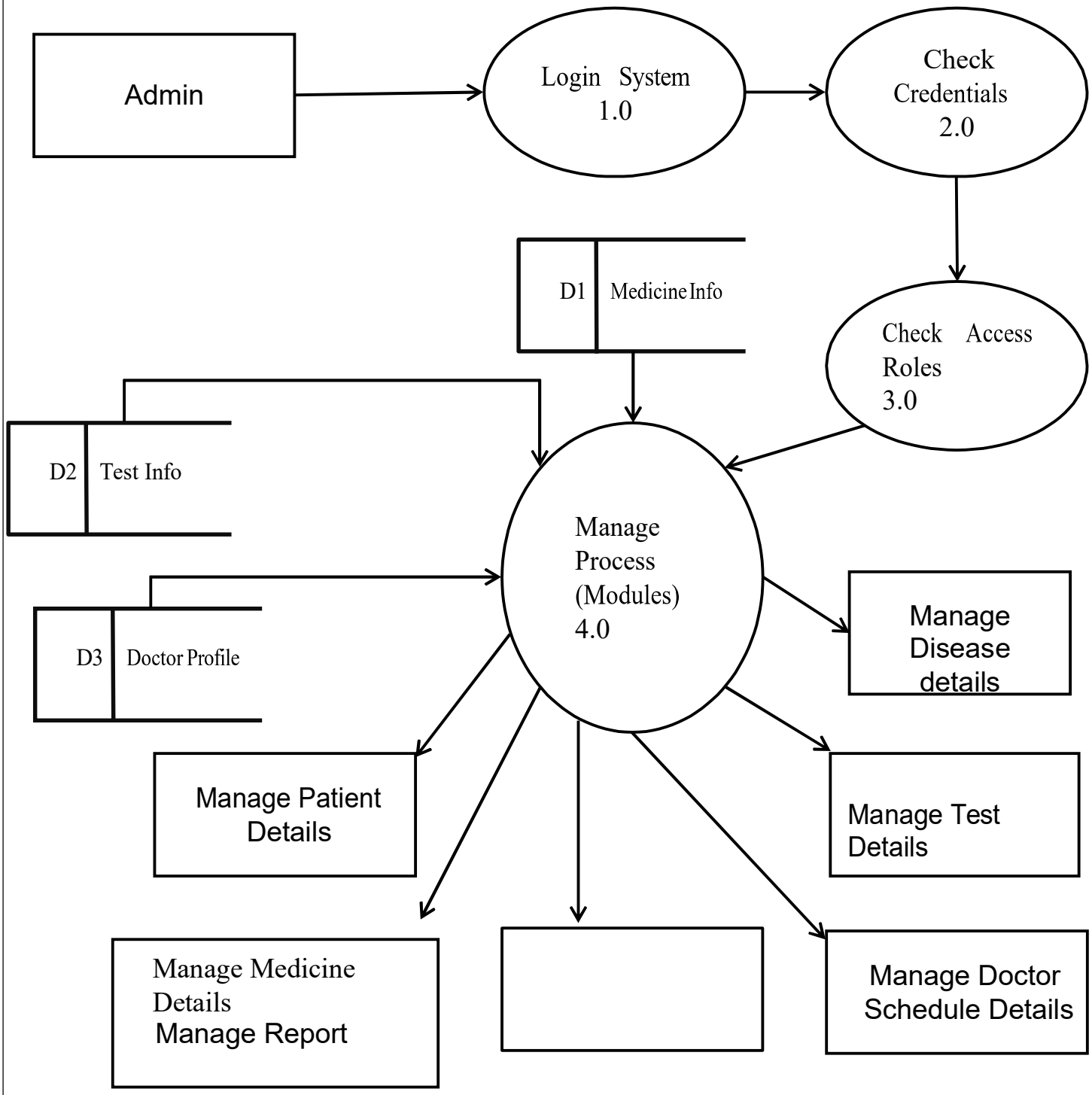
Risk leverage = ¯_____

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD


Admin
Login System
1.0
Check
Credentials
2.0
D1
Medicine Info
Check Access
Roles
3.0
D2
Test Info
Manage
Process
(Modules)
4.0
D3
Doctor Profile
Manage
Disease
details
Manage Patient
Details
Manage Test
Details
Manage Medicine
Details
Manage Report
Manage Doctor
Schedule Details

Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

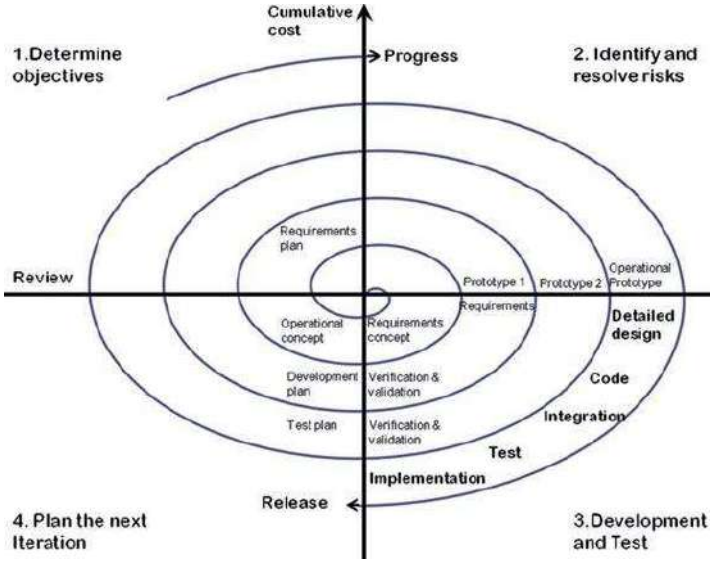
Project 4:

Hotel Automation Management System

Problem Statement:

Guests can reserve hotel rooms in advance or on the spot depending on the availability of rooms. The operator would enter data pertaining to guests such as their arrival time, advance pay, approximate duration of stay, purpose and type of room required i.e. AC, non AC, deluxe, single/double occupancy. The computer should register this data and depending on the availability of a suitable room, it should provisionally allot a room number to the guest and assign a unique token number. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering service manager would input the food items and their quantity as and when consumed by the guest along with the token number and the corresponding date and time. It can be assumed that different food items have different codes and hence catering service manager need not enter the name of the food items. When the customer prepares to check out, the hotel automation software generates the bill along with the taxes.

Process Model:



1. Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
2. For developing this software, we will use Spiral Model for the following reason:
    a. Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
    b. Spiral Model consists of Risk Analysis task region.
    c. In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS):

1. Functional Requirements

As per the requirements of the software from client end and analysis we realise that we need five modules:

A. **Guest Module:** In this module, a guest is registered and the profile is maintained. If a suitable room is available, a room number and unique token number are also assigned. Else an apology message is generated. A database is maintained for keeping the information.

Input: Guest Details
Output: Unique Profile, Updated Database

B. **Catering Module:** In this module, the details of food items consumed by the guest are maintained. A database is maintained for keeping the information.

Input: Guest Details, Room number, Token number Output: Updated Database

C. **Bill Module:** In this module, the total bill is generated when the guest prepares to check out.

Input: Guest Details, Room number, Token number, Food Output: Generated Bill

2. Non - Functional Requirements

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

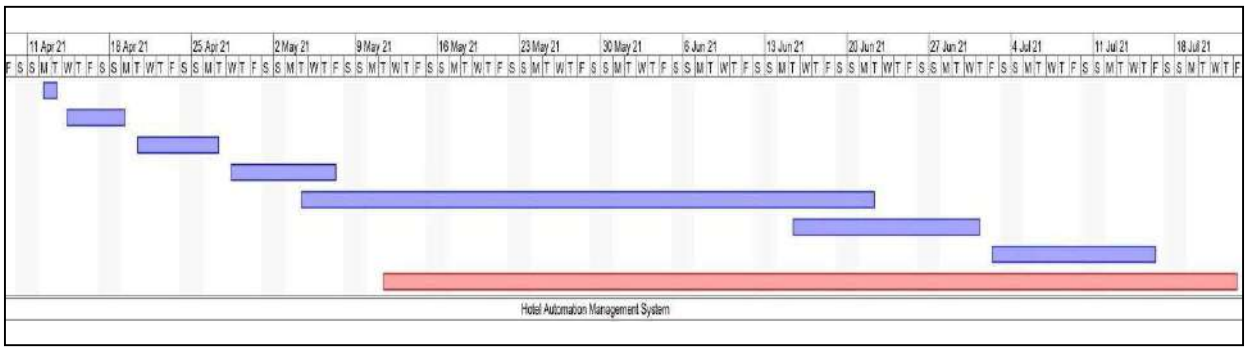
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

1. Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2. Project Resources

A. Man Power Resources:

- Project Manager
- System Analyst
- Designer
- Developer
- Tester
- Document Writer

B. Hardware and Software Resources

- Processor: Intel i3 4th gen or above
- Ram: 4Gb or above
- Java Development Kit – 13
- Java Runtime Environment
- Sublime Text 3 (Text Editor)
- Ms –Project
- Creately
- Google Docs

3. Risk Management Plan

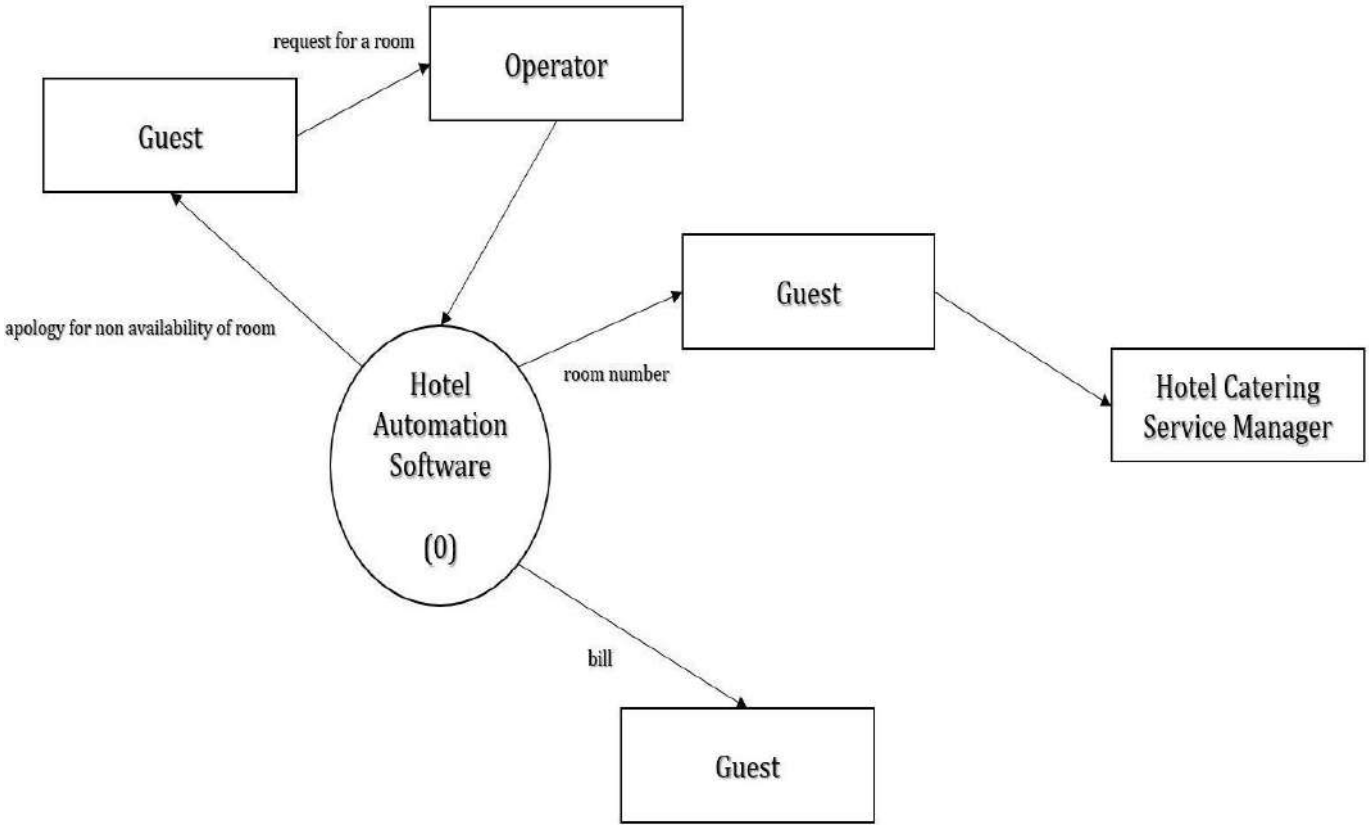
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

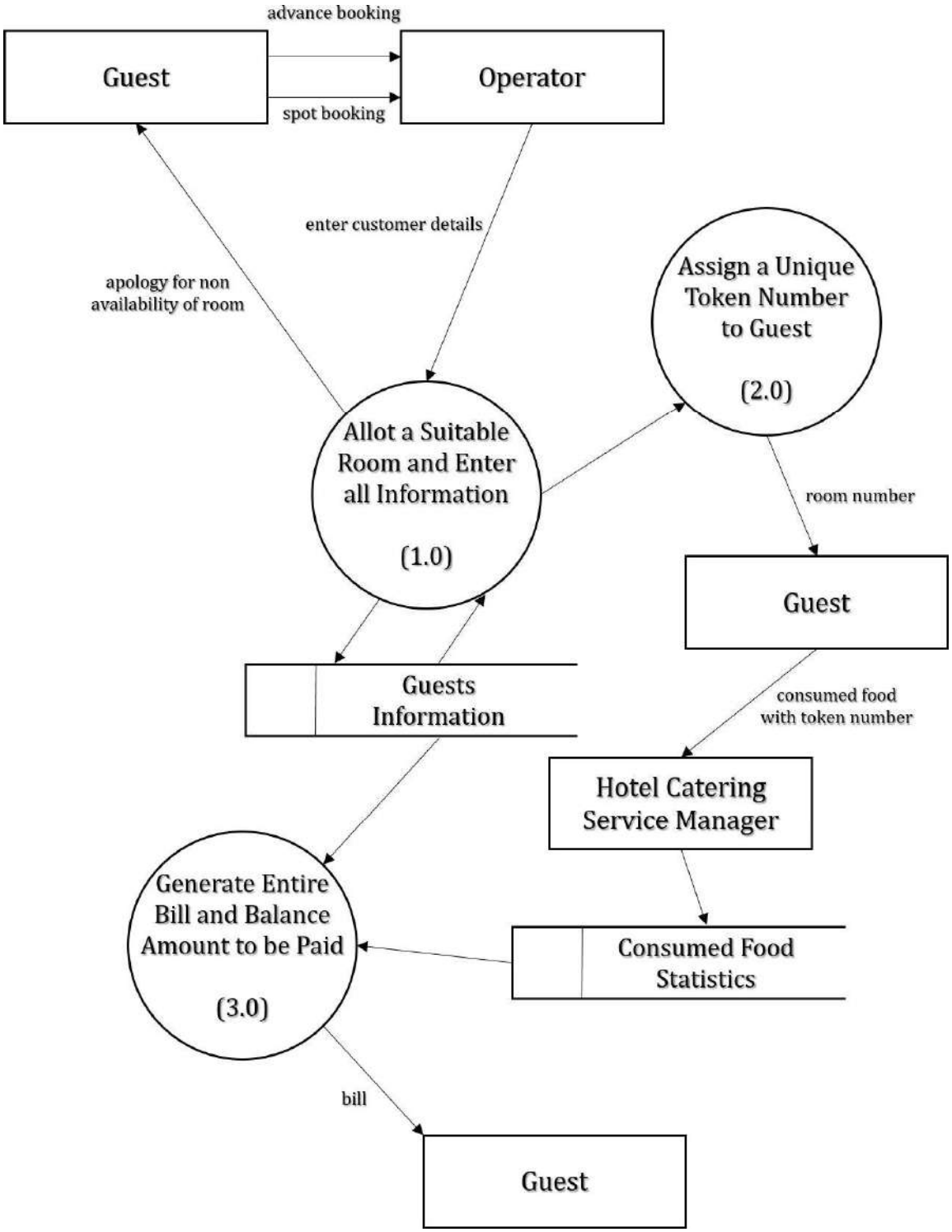
Risk leverage = ¯

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD


advance booking
Guest
Operator
spot booking
enter customer details
apology for non availability of room
Assign a Unique Token Number to Guest
(2.0)
Allot a Suitable Room and Enter all Information
(1.0)
room number
Guest
Guests Information
consumed food with token number
Hotel Catering Service Manager
Generate Entire Bill and Balance Amount to be Paid
(3.0)
Consumed Food Statistics
bill
Guest

Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project 5:

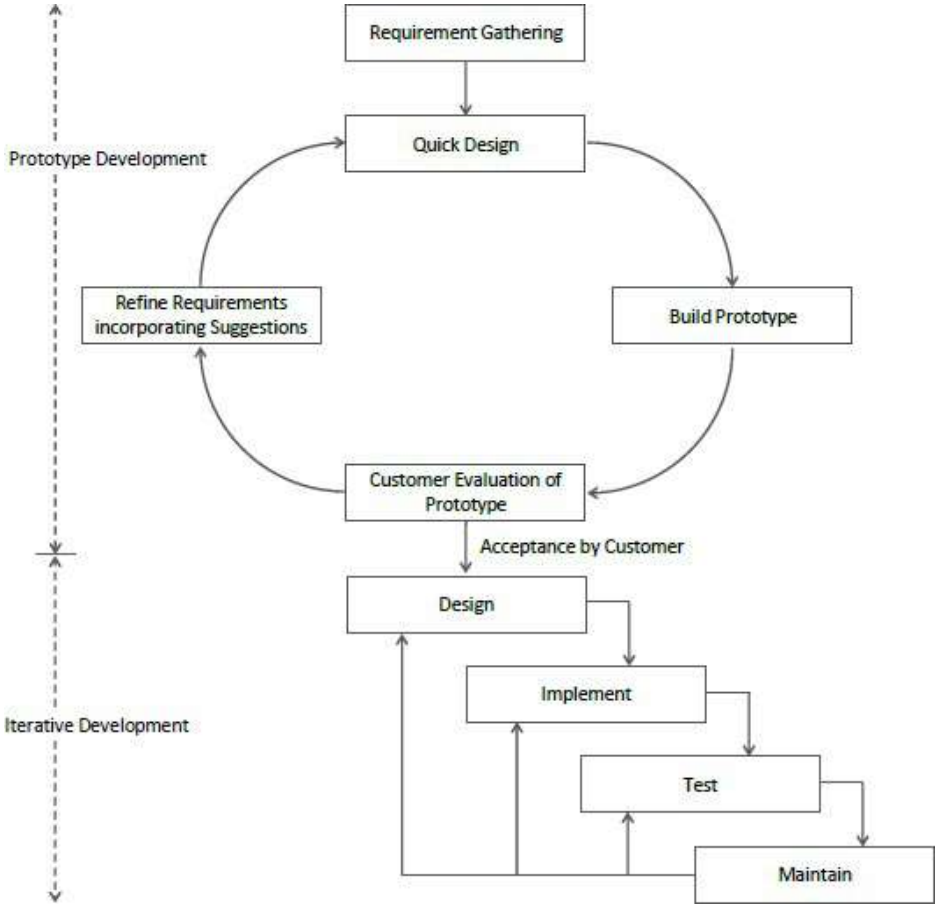
Criminal Record Management System

Problem Statement:

Crime file management system identifies internal data stores of court, criminals, prisoners, charge sheet, FIR that must be present in order for the crime system to do its job. Each shows the flow of data among the various paths of crime, FIR, criminals, court of the system. A criminal record management system is implemented for jailers, police officers and CBI officers.

Process Model:

Prototyping Model



This model allows all or part of a system to be constructed quickly. Goal of prototyping is to reduce the chance of uncertainty. The phases include –

1. meeting of the developers and the customers to define the overall objectives of the software
2. quick design leads to construction of prototype
3. evolution of prototype by customer
4. iteration occurs as the prototype is turned to the needs of the customers
    a. the code for prototype is thrown away – the experience gathered from developing the prototype helps in developing the actual system

Requirement Analysis

1. Functional Requirements

As per the requirements of the software from client end and analysis we realise that we need four modules:

A. Admin Module: In this module, an admin is registered and the profile is maintained. The level of access is granted according to designation.

Input: Admin Details
Output: Unique Profile, Granted Level of Access

B. Complaint Module: In this module, complain is registered and FIR is filed. A database is maintained to keep the complaint details.

Input: Complaint Details
Output: Filed FIR, Updated Database

C. Crime Module: In this module, the details of crime are maintained and charge sheet is filed.

Input: Complaint Details, FIR Details
Output: Filed Charge sheet

D. Court Module: In this module, the case is taken to the court and legal action is taken accordingly.

Input: Complaint Details, FIR Details, Charge sheet Details Output: Legal Action, Updated Database

2. Non - Functional Requirements

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

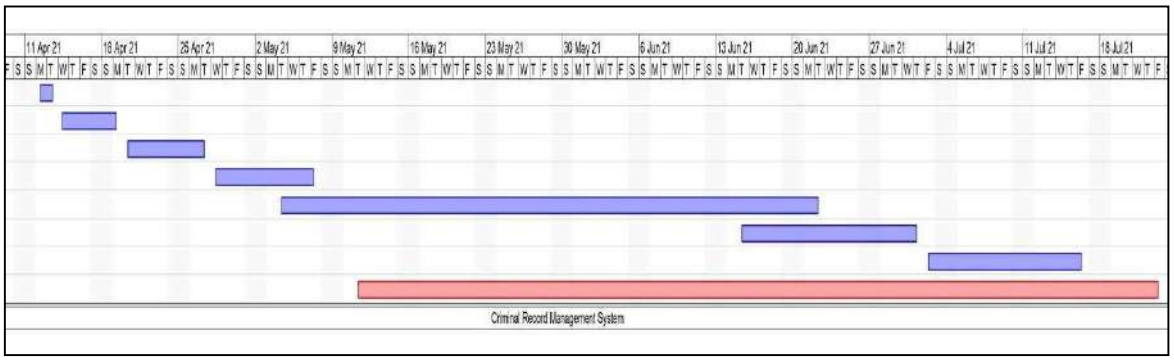
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Feasibility Study	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Requirement Analysis and Specifications	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Design	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Coding and Unit Testing	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Integration and System Testing	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Maintenance	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



Project Resources:

1. **Hardware Resources:** Several computer machines.
2. **Human Resources:** Manpower Resources like Project Manager, Designer, Analyst, Programmer, Tester etc.

3. Risk Management Plan

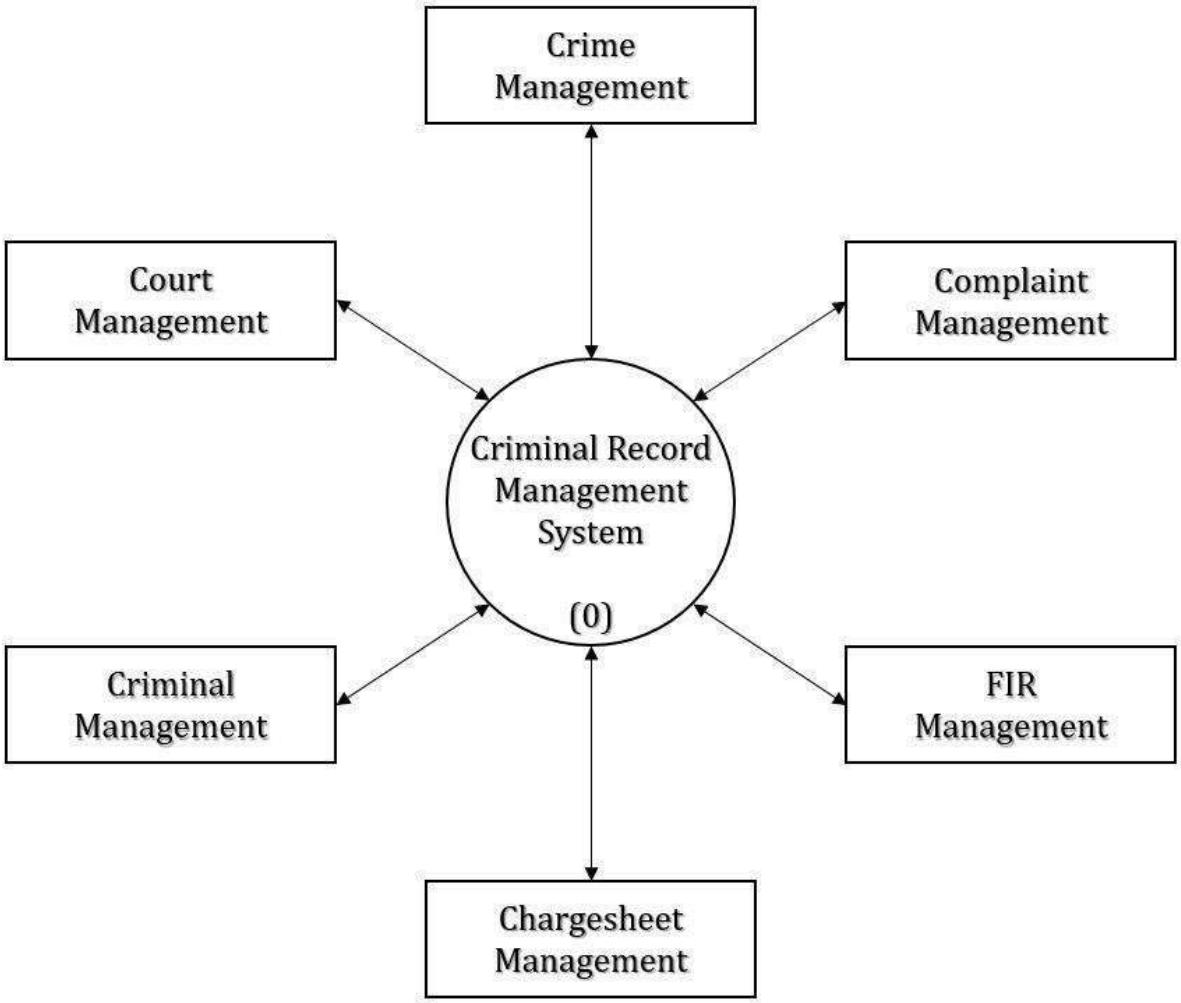
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

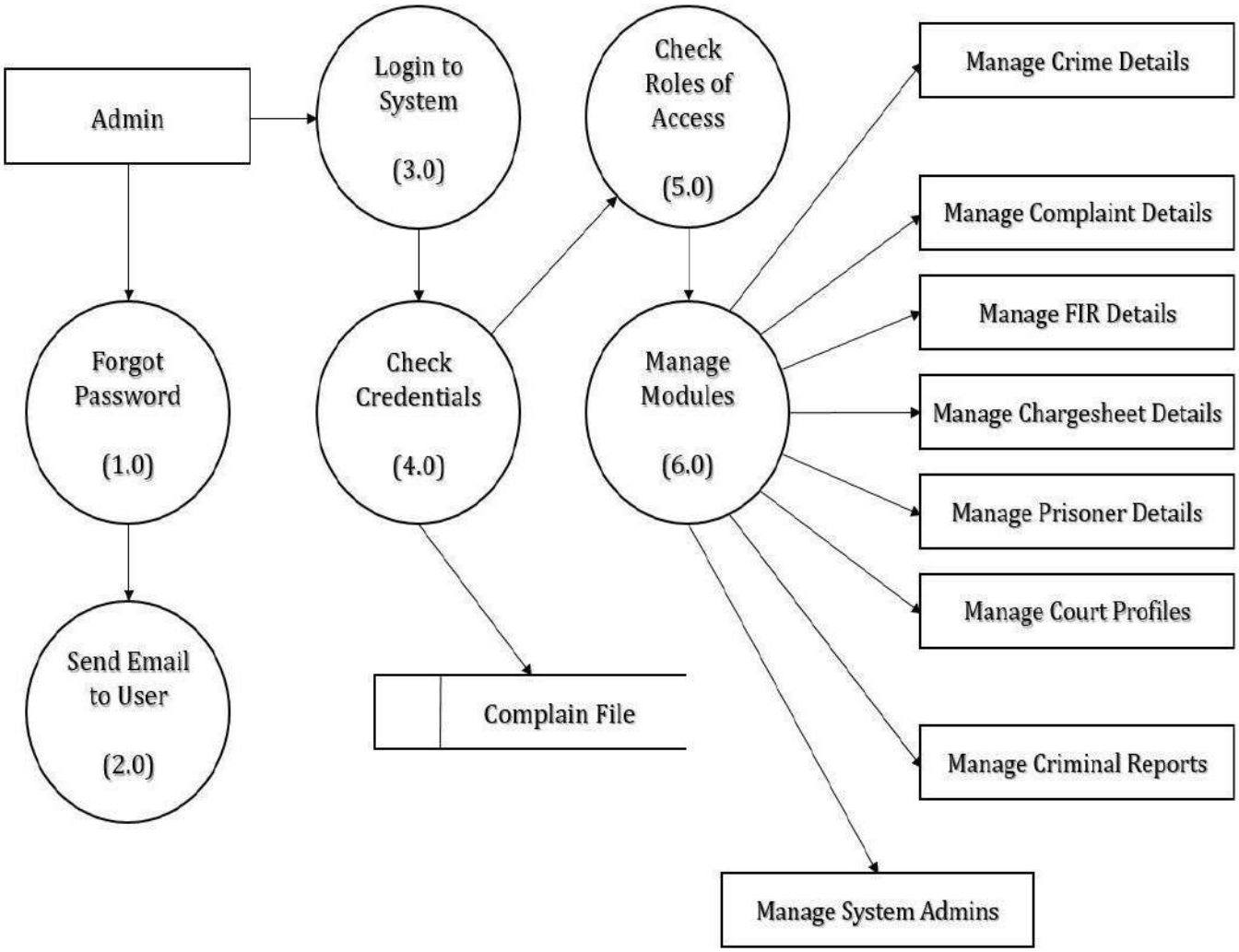
Risk leverage = ¯_____

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD


Admin
Login to System
(3.0)
Check Roles of Access
(5.0)
Manage Crime Details
Manage Complaint Details
Manage FIR Details
Manage Chargesheet Details
Manage Prisoner Details
Manage Court Profiles
Manage Criminal Reports
Forgot Password
(1.0)
Check Credentials
(4.0)
Manage Modules
(6.0)
Send Email to User
(2.0)
Complain File
Manage System Admins

Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

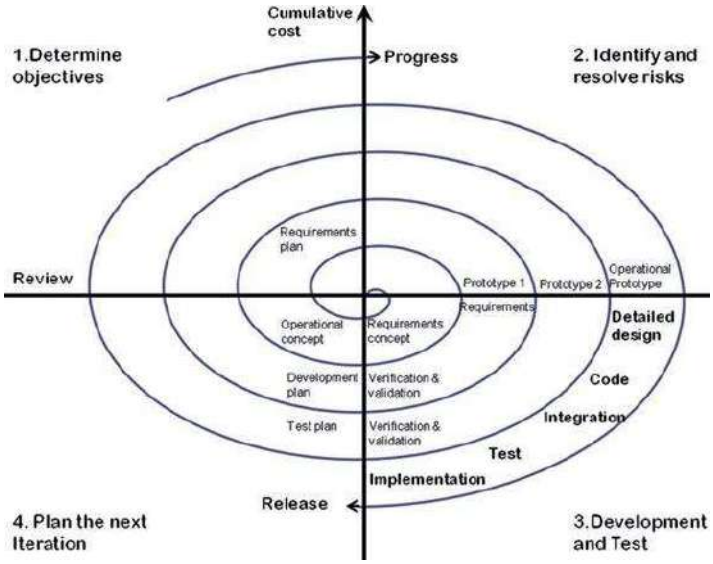
It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project 6:
Examination and Result Computation System

Problem Statement:

Examination management system identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. Each shows the flow of data among the various parts of timetable, class, course, subject, branch of the system. The main functionalities of the examination are processing timetable records, examination records, class records, student records, branch records, course records, subject records and generate report of the same.

Process Model:



1. Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
2. For developing this software, we will use Spiral Model for the following reason:
    a. Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
    b. Spiral Model consists of Risk Analysis task region.
    c. In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS):

Functional Requirements:

As per the requirements of the software from client end and analysis we realise that we need four modules:

A. **Student Module:** In this module, a student is registered and the profile is maintained. A database is maintained for keeping the student information.

   Input: Student Details
   Output: Unique Profile

B. **Timetable Module:** In this module, the details of branch, course, class and subject are maintained and the timetable is created. A database is maintained for keeping the timetable information.

   Input: Branch details, Course details, Class Details, Subject Details Output: Updated database, Created timetable

C. **Examination Module:** In this module, the examination takes place and the results are computed.

   Input: Student details, Timetable details Output: Computed results, Updated database

D. **Report Module:** In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated

   Input: Exam details, Student details, timetable details, Branch details, Course details
   Output: Generated Reports

Non-Functional Requirements

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

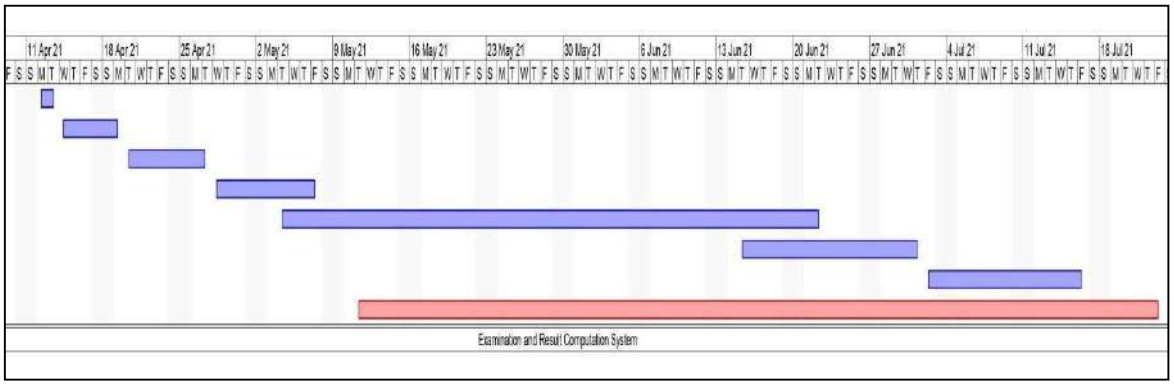
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2.Project Resources:

1. **Hardware Resources:** Several computer machines.
2. **Human Resources:** Manpower Resources like Project Manager, Designer, Analyst, Programmer, Tester etc.

3. Risk Management Plan

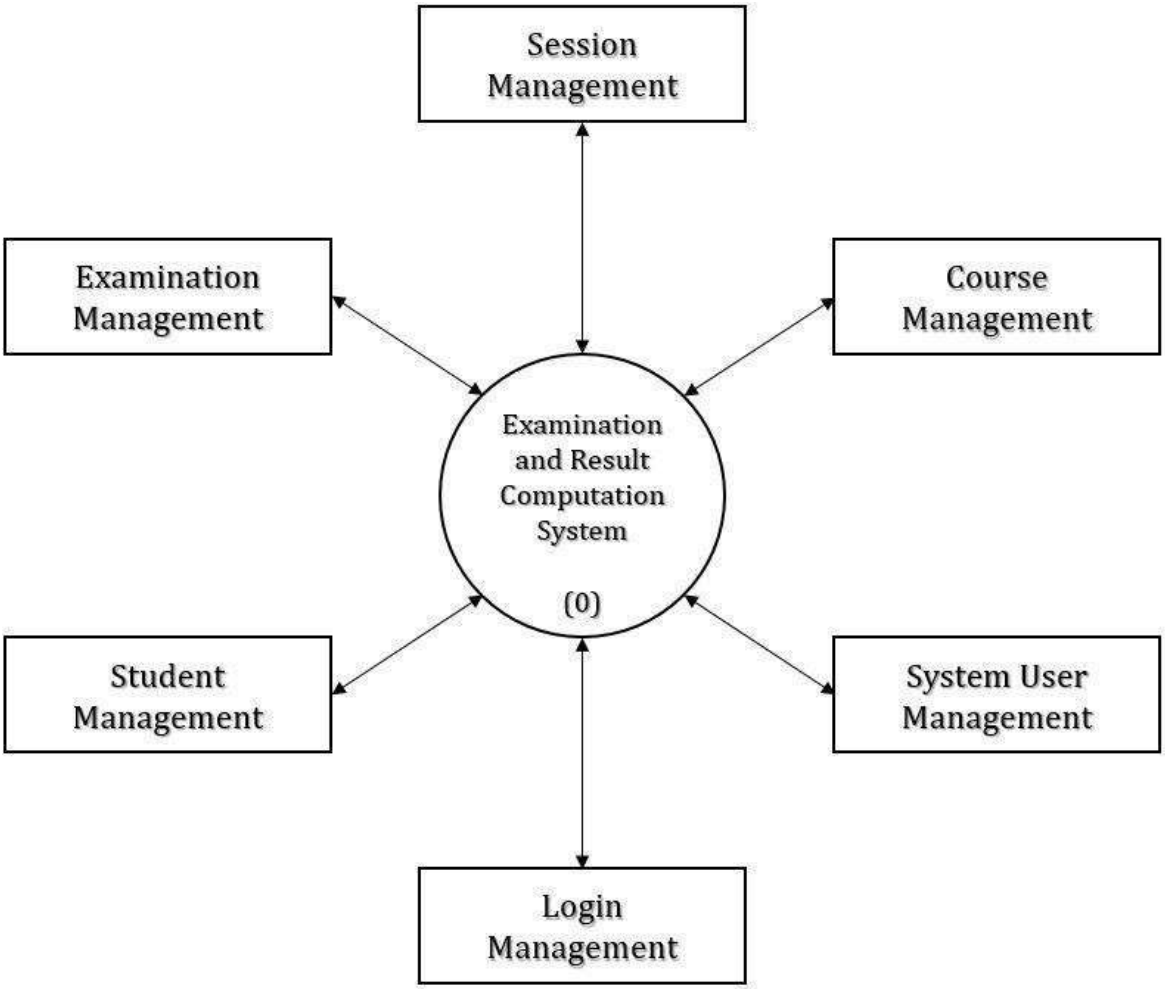
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

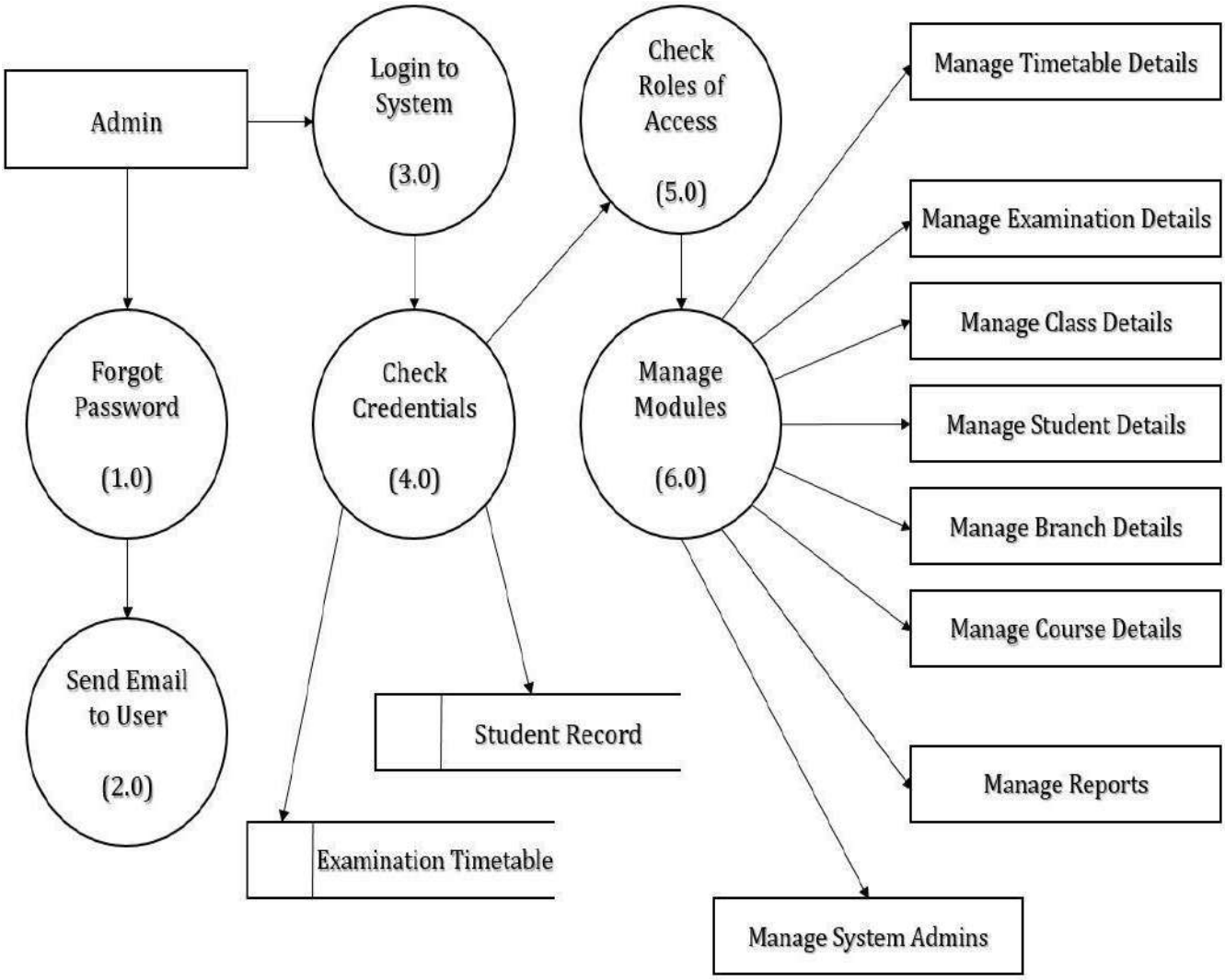
Risk leverage = ¯

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD


Admin
Login to System (3.0)
Check Roles of Access (5.0)
Manage Timetable Details
Manage Examination Details
Manage Class Details
Manage Student Details
Manage Branch Details
Manage Course Details
Manage Reports
Forgot Password (1.0)
Check Credentials (4.0)
Manage Modules (6.0)
Send Email to User (2.0)
Student Record
Examination Timetable
Manage System Admins

Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

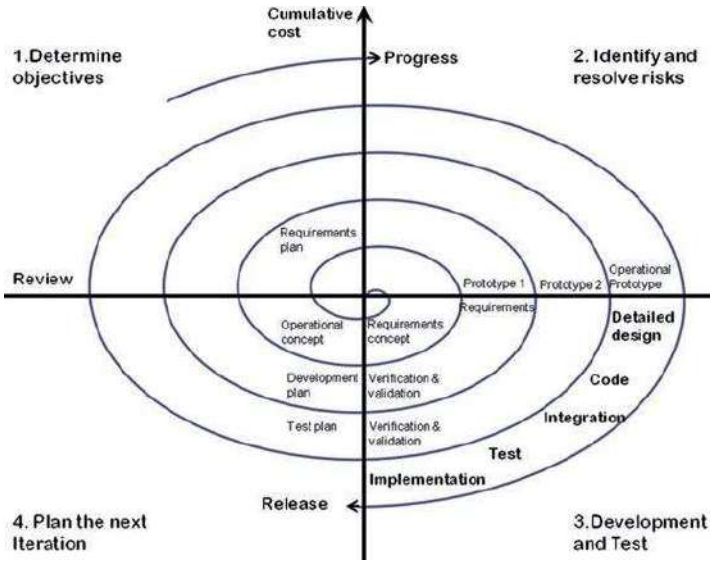
It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project 7:
Parking Allocation System

Problem Statement:

Parking management system identifies internal data stores of login, customers, duration and parking fees that must be present in order for the parking system to do its job. It shows the flow of data between the various paths of parking slots and parking fees. The functionalities of parking system are processing of parking slots, vehicle records, parking fees, duration and generating the report of the same.

Process Model:



1. Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
2. For developing this software, we will use Spiral Model for the following reason:
    a. Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
    b. Spiral Model consists of Risk Analysis task region.
    c. In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS):

Functional Requirements

Asperthe requirements of the software from client end and analysis we realise that we need four modules:

A. **Customer Module:** In this module, a customer is registered and the profile is maintained.

   Input: Customer Details
   Output: Unique Profile

B. **Vehicle Module:** In this module, the customer's vehicle is registered.

   Input: Customer details, Vehicle details
   Output: Updated database

C. **Parking Module:** In this module, parking slot is allotted and fee is charged accordingly. A database is maintained for keeping the vehicle information.

   Input: Customer details, Vehicle details, Duration Output: Updated database

D. **Report Module:** In this module, parking slots, vehicle, parking fee and duration records are processed and reports are generated.

   Input: Parking slot details, Vehicle Details, Fee details, Duration Output: Generated Reports

Non - Functional Requirements:

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

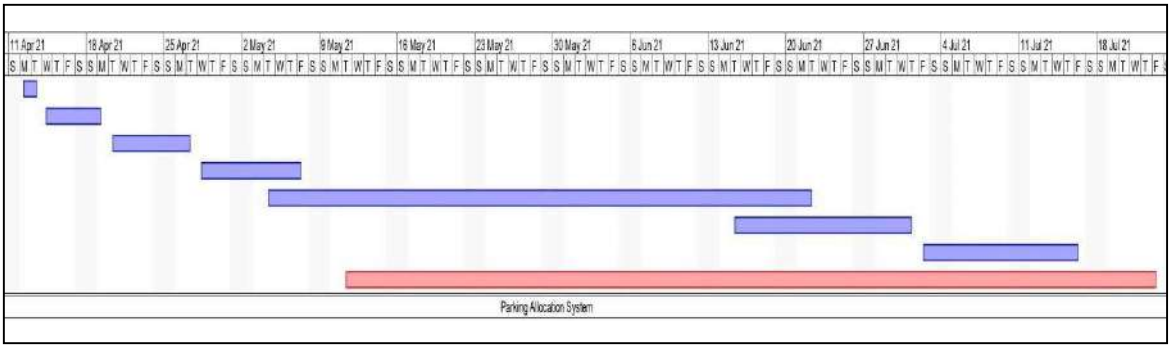
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2.Project Resources:

1. **Hardware Resources:** Several computer machines.
2. **Human Resources:** Manpower Resources like Project Manager, Designer, Analyst, Programmer, Tester etc.

3. Risk Management Plan

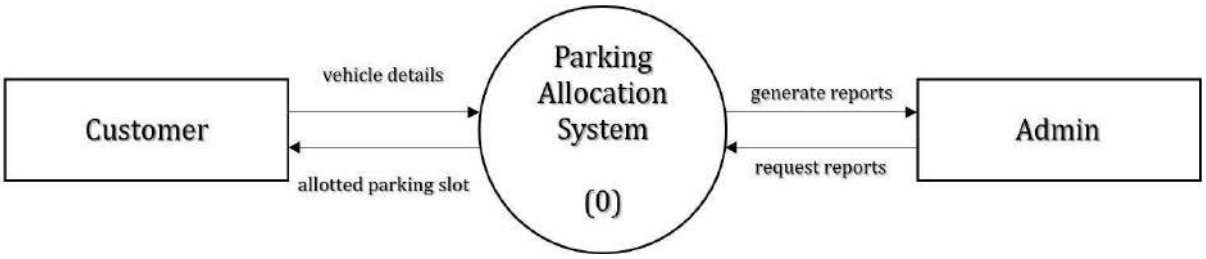
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

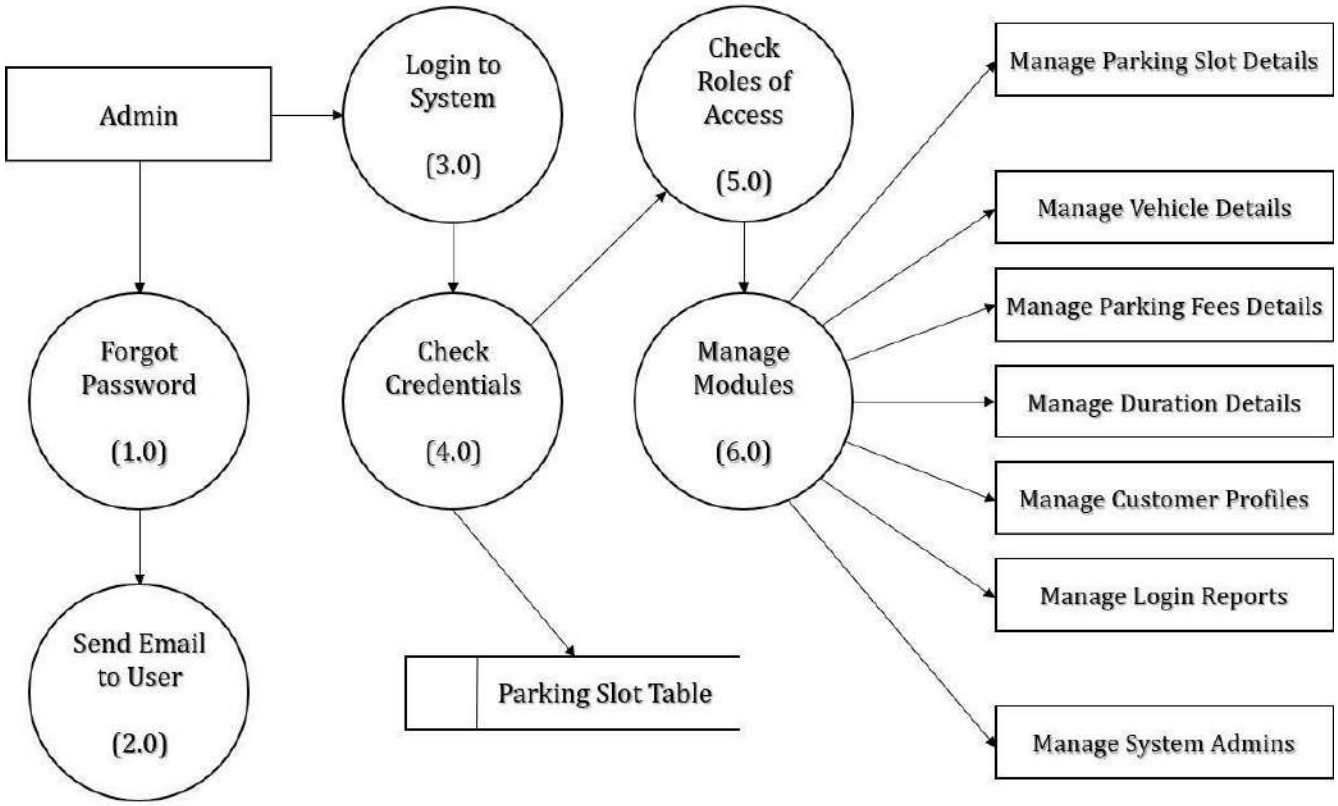
Risk leverage = $\frac{}{}$

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD



Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

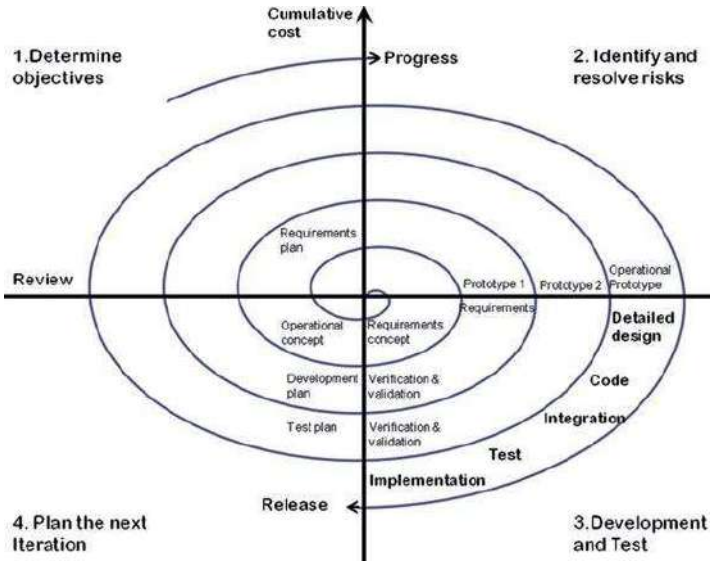
It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project 8:
Wholesale Management System

Problem Statement:

It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model:



1. Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
2. For developing this software, we will use Spiral Model for the following reason:
    a. Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
    b. Spiral Model consists of Risk Analysis task region.
    c. In this model, we systematically prepare prototype and release for customer feedback.

Requirement Analysis (SRS):

Functional Requirements:

Asperthe requirements of the software from client end and analysis we realise that we need four modules:

A. **Customer Module:** In this module, a customer is registered and the profile is maintained.

   Input: Customer Details
   Output: Unique Profile

B. **Stock Module:** In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: Stock details Output:
   Updated database

C. **Bill Module:** In this module, all bills are maintained. A database is maintained for keeping the bill information.

   Input: Customer details, Stock details Output:
   Generated bills, Updated database

D. **Report Module:** In this module, product details, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: Store details, Product details, Retail price, Billing counter details, Customer details
   Output: Generated Reports

2.NonFunctional Requirements

1. **Usability:** The software should have a user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

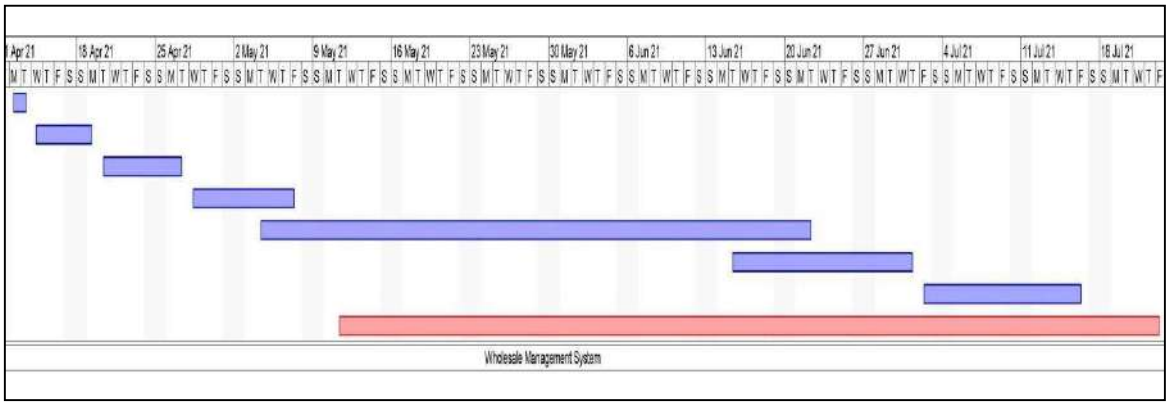
2. **Portability:** The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

3. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

Project Management:

Project Scheduling

	Name	Duration	Start	Finish
1	Problem Statement	1.5 days	12/4/21 10:00 AM	13/4/21 3:00 PM
2	Customer Communication	3 days	14/4/21 10:00 AM	19/4/21 10:00 AM
3	Planning	5 days	20/4/21 10:00 AM	27/4/21 10:00 AM
4	Risk Analysis	7 days	28/4/21 10:00 AM	7/5/21 10:00 AM
5	Engineering	35 days	4/5/21 10:00 AM	22/6/21 10:00 AM
6	Construction and Release	12 days	15/6/21 10:00 AM	1/7/21 10:00 AM
7	Customer Evaluation	10 days	2/7/21 10:00 AM	16/7/21 10:00 AM
8	Documentation and User Manual	53 days	11/5/21 10:00 AM	23/7/21 10:00 AM



2.Project Resources:

1. **Hardware Resources:** Several computer machines.
2. **Human Resources:** Manpower Resources like Project Manager, Designer, Analyst, Programmer, Tester etc.

3. Risk Management Plan

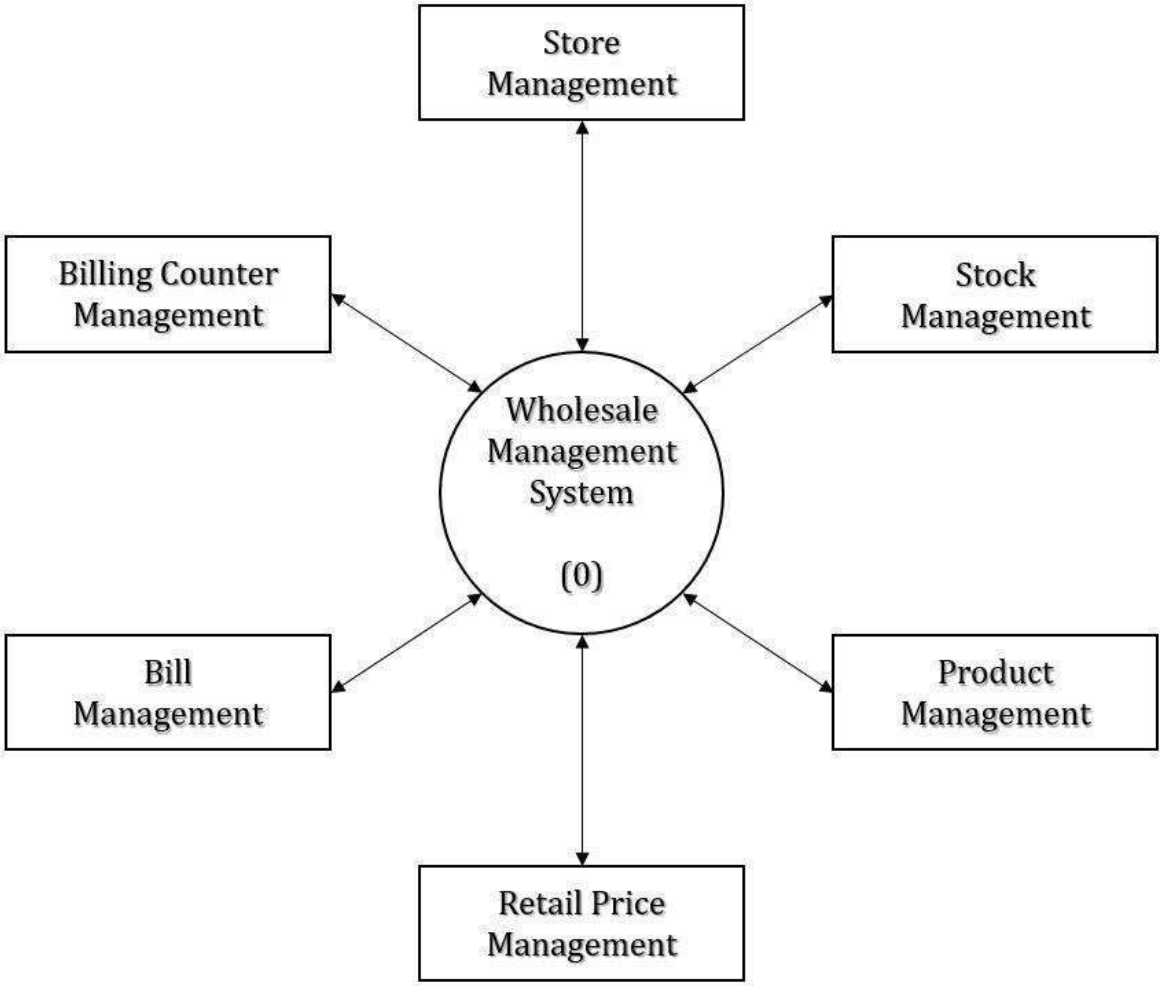
Risks are measured by 2 parameters:

1. Likelihood of a risk coming through (Probability of occurrence)
2. Consequence of the problem associated with that risk.
    a. Priority of risk is the multiplication of above two.

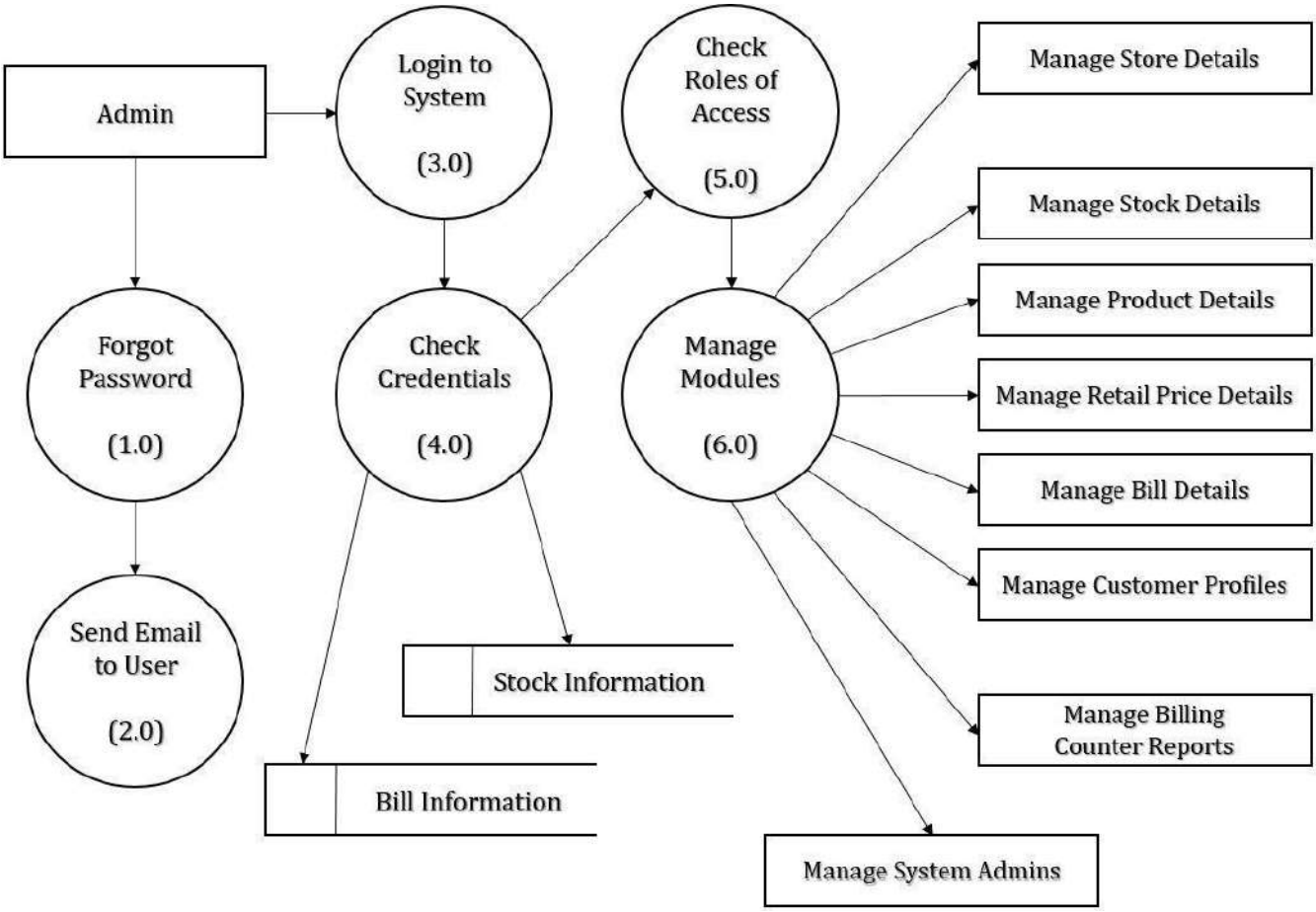
Risk leverage = $\frac{}{}$

Design Engineering:

Level 0 DFD: Context Diagram



Level 1 DFD


Admin
Login to System (3.0)
Check Roles of Access (5.0)
Manage Store Details
Manage Stock Details
Manage Product Details
Manage Retail Price Details
Manage Bill Details
Manage Customer Profiles
Manage Billing Counter Reports
Forgot Password (1.0)
Check Credentials (4.0)
Manage Modules (6.0)
Send Email to User (2.0)
Stock Information
Bill Information
Manage System Admins

Testing:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

Each module is tested in isolation.

2. System Testing:

The modules are integrated and again tested. This time this testing is done in three parts:

A. **Alpha testing (Verification):** Done by Development team
B. **Beta testing (Validation):** Done by a special group of friendly customers in an uncontrolled environment.
C. **Acceptance testing:** Performed by customers after product delivery to determine whether to accept or reject software.

3. Integration testing:

Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

Maintenance:

Maintenance stands for all modifications and updating done after the delivery of a software product. Types of maintenance done in this retail software are:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Name: Sudipta Das
College Roll: 1705
Registration No: A01-1112-117-005-2018
Exam Roll no: 2021141309
Semester: V
Dept: Comp. Sc.

Supervisor: Prof. Manas Pal

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a “Customer Number” (CN). A customer can present ‘CN’ to customer staff when he makes purchases. In this case, the value of his purchase is credited against its ‘CN’. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

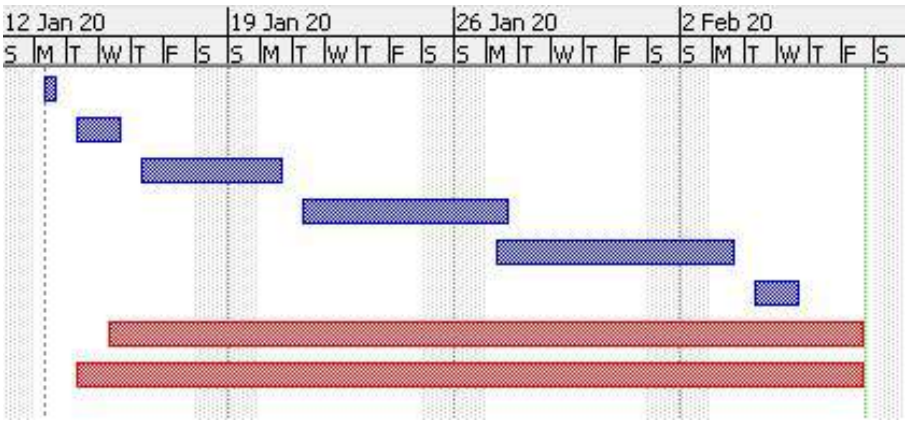
REQUIREMENT ANALYSIS (SRS) :

As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account's purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- ### Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM

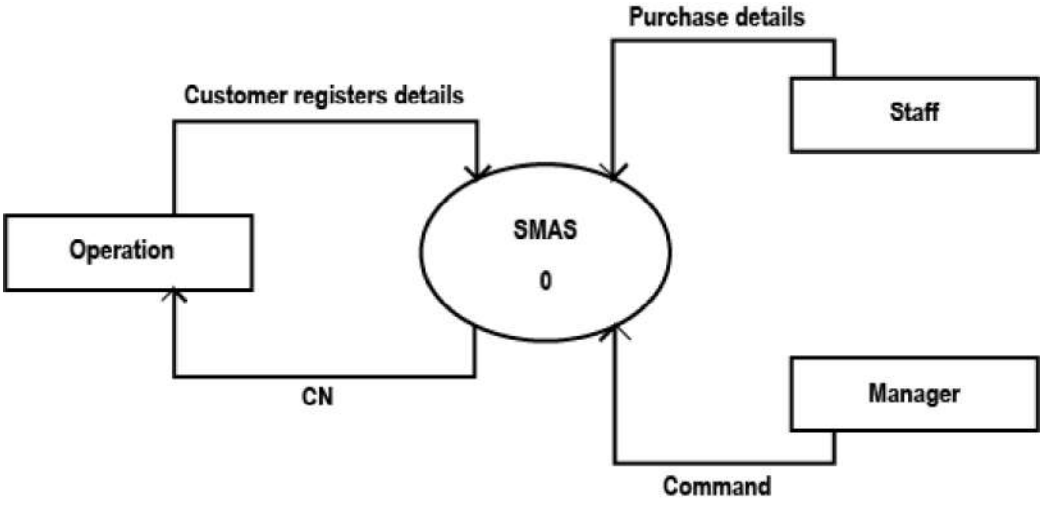


- **Project Resources :**
    - **Hardware Resources :** Several computer machines.
    - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, tester etc.
- **Risk management plan :** Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
        - Priority of risk is the multiplication of above two.
    - Risk leverage =

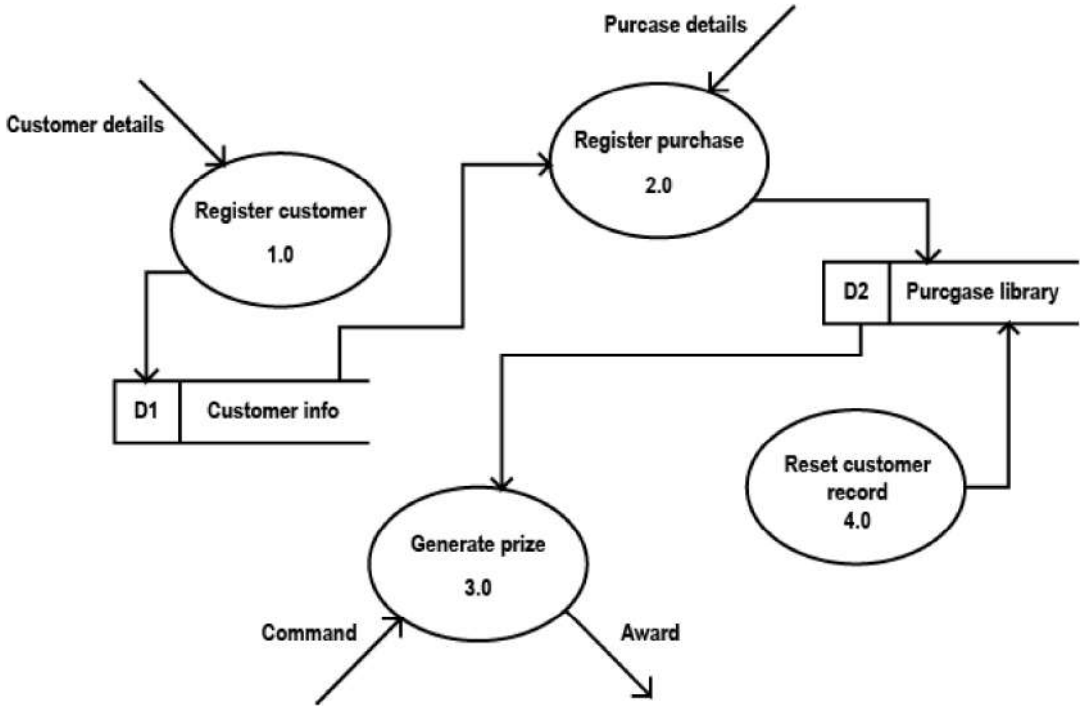
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

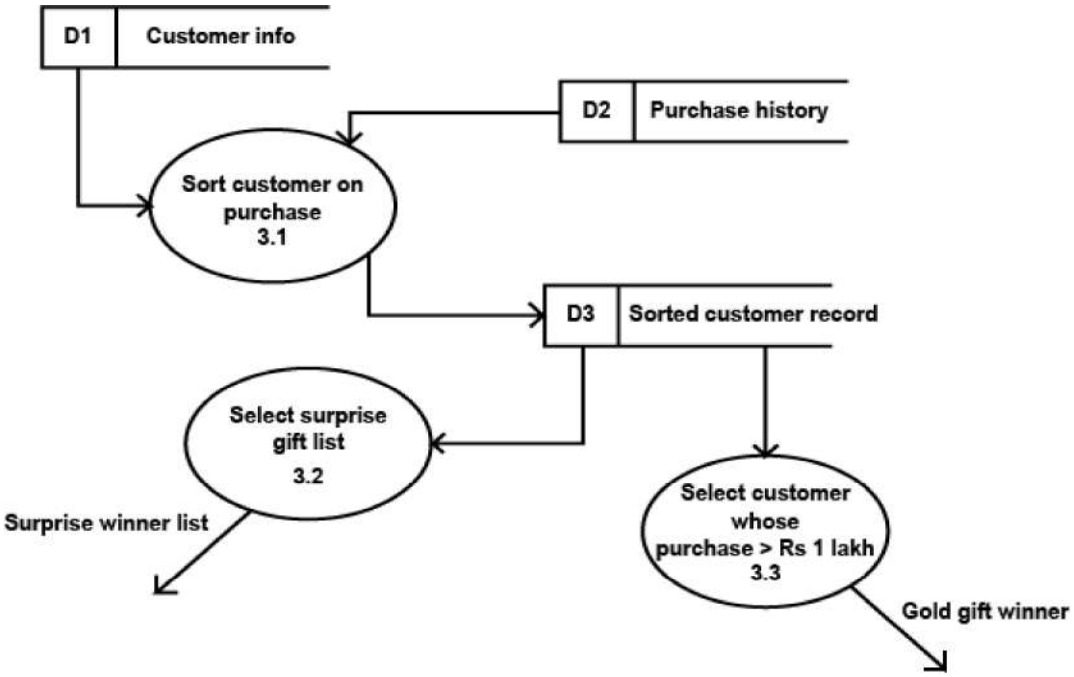
1. **Level 0 DFD :**



2. Level 1 DFD :



3. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

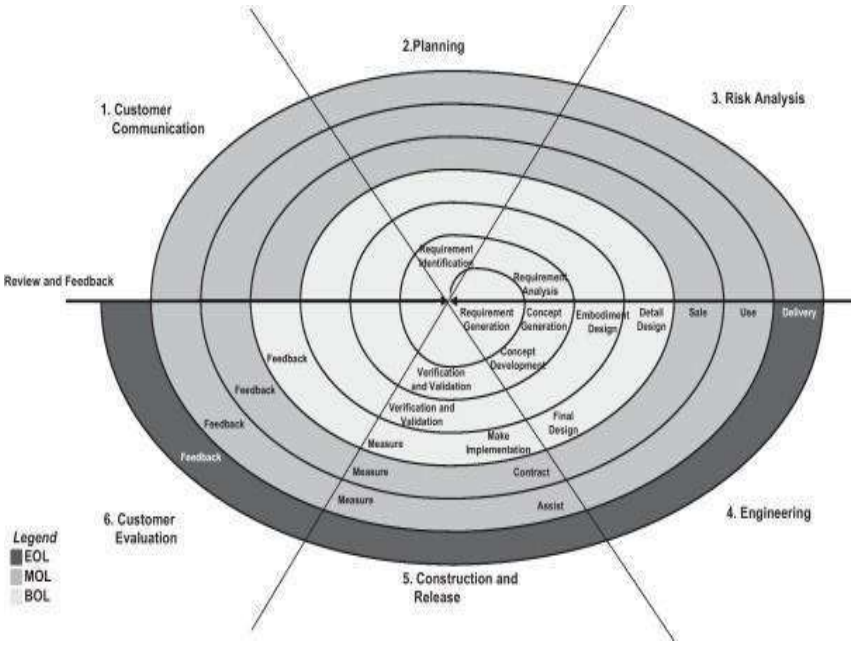
Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project 2: Route Management System

PROBLEM STATEMENT

It identifies internaldatastores of payment,bus route, customer,ticket,ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule.It also shows the details of ticket counter,bus route,payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:
  - ★ Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
  - ★ Spiral Model consists of Risk Analysis task region.
  - ★ In this model, we systematically prepare prototype and release for customer feedback.
- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. ***Goal of the Project:***

   The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. ***Functional Requirements:***

   As per the requirements of the software from client end. By analysis we realise that we need five modules:

   A. Admin Module:

   This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

   Input: UserName and Password

   This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

   Input: Answer to security question
   Output: Link for changing password

   B. Customer Module:

   This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

   C. Schedule Module:

   This module maintains timings of the bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

   D. Route Module:

   This module has a function to query about bus routes and eventually know the stops it would cover during its journey . It also has a function which makes drivers know their route.

   E. Report Module:

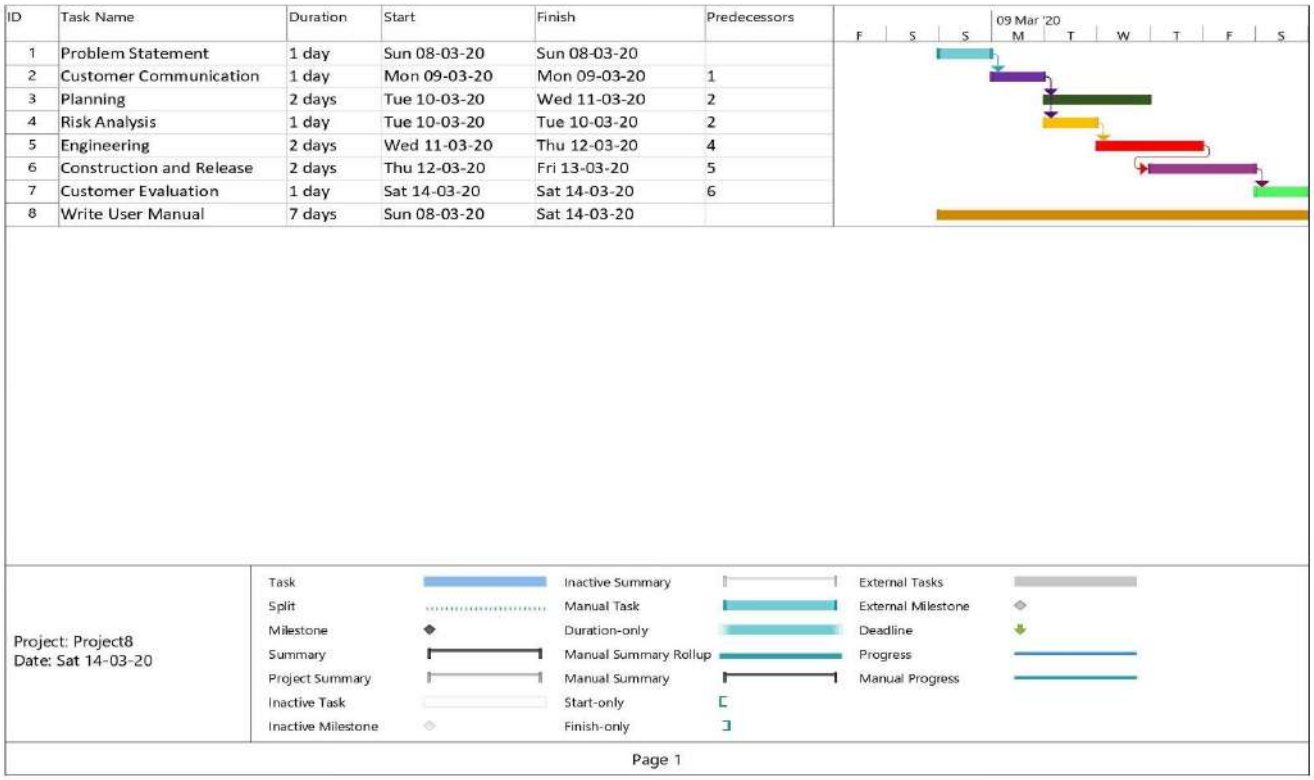
   At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

3. ***Non-Functional Requirements:***

   A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.
   B. Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
   C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a. **Project Scheduling**



ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 08-03-20	Sun 08-03-20	
2	Customer Communication	1 day	Mon 09-03-20	Mon 09-03-20	1
3	Planning	2 days	Tue 10-03-20	Wed 11-03-20	2
4	Risk Analysis	1 day	Tue 10-03-20	Tue 10-03-20	2
5	Engineering	2 days	Wed 11-03-20	Thu 12-03-20	4
6	Construction and Release	2 days	Thu 12-03-20	Fri 13-03-20	5
7	Customer Evaluation	1 day	Sat 14-03-20	Sat 14-03-20	6
8	Write User Manual	7 days	Sun 08-03-20	Sat 14-03-20	

b. **Project Resources**

a. ManPower Resources:

i. **Project Manager**
ii. **System Analyst**
iii. **Designer**
iv. **Developer**
v. **Tester**
vi. **Document Writer**

b. Hardware-Software Resources

i. Processor: Intel i3 4th gen or above
ii. Ram: 4Gb or above
iii. Java Development Kit - 13
iv. **Java Runtime Environment**
v. Sublime Text 3 (Text Editor)
vi. Ms - Project
vii. Creately
viii. Google Docs

c. **Risk Management Plan**

All risks are measured in terms of :

A. Likelihood of a risk coming through
B. Consequence of the problem associated with that risk

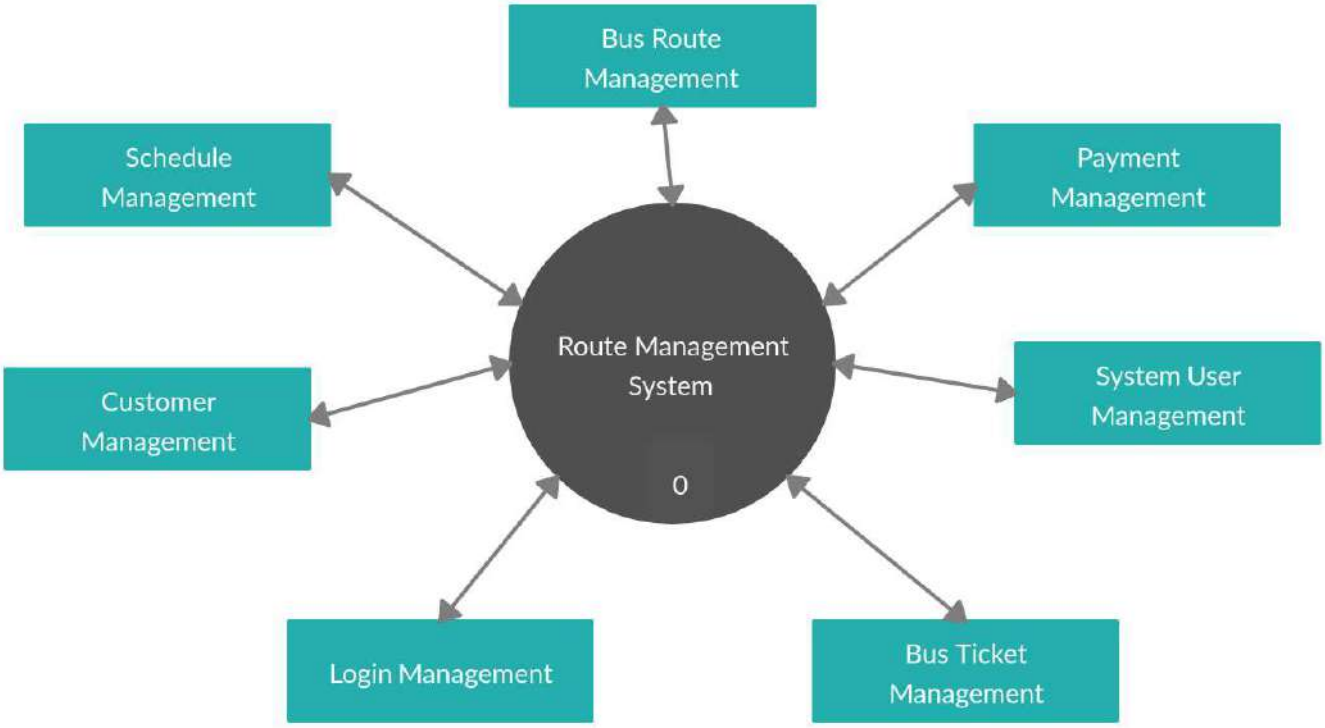
Priority of each risk is multiplication of above two

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

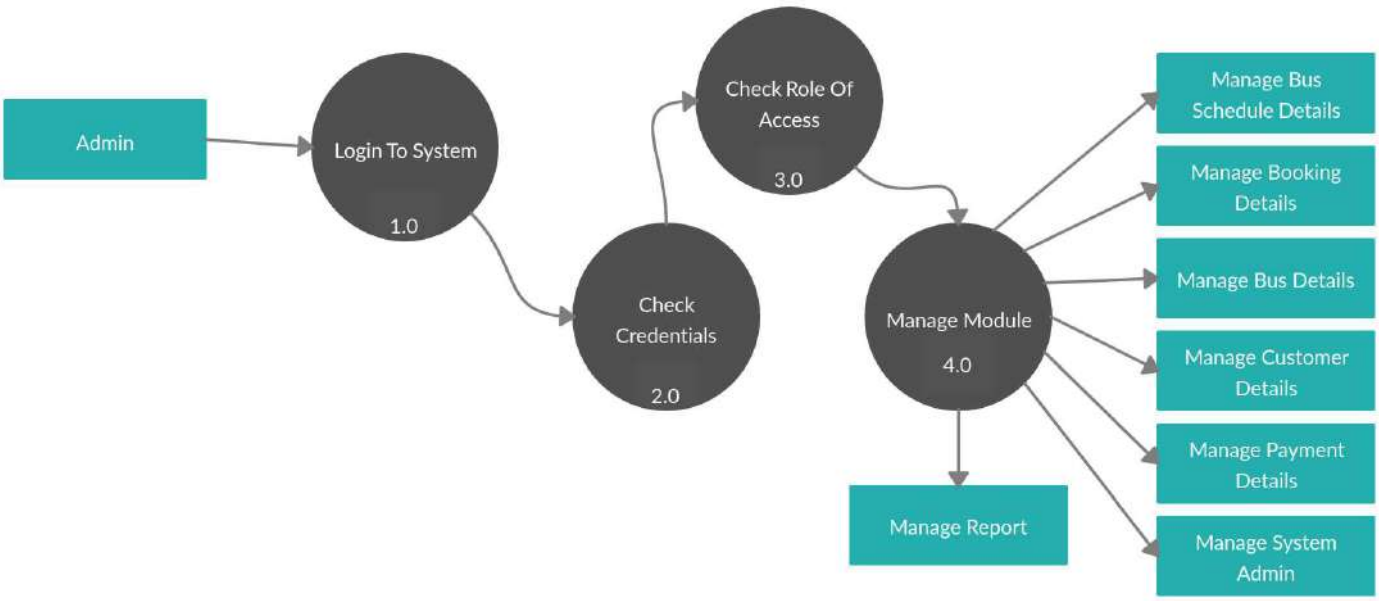
DESIGN ENGINEERING

1. Architectural Design



2. Component Level Design

- **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:
   Each module is tested in isolation
2. System Testing:
   The modules are integrated and again tested. This time this testing
   is done in three parts
   A. Alpha testing(Verification) : done by Development team
   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.
3. Integration Testing:
   Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

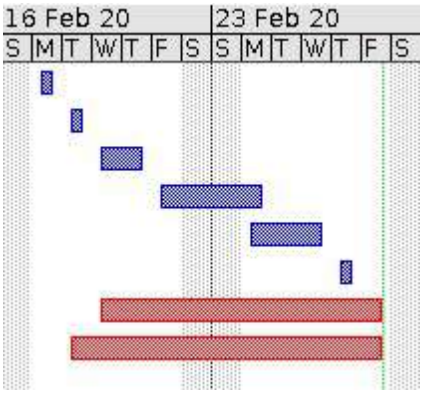
- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
    - Doctor module : This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.
    - Report module : This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.
    - Diagnostic module : This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.

- Non-Functional Requirements:
  - Usability: The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
  - Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



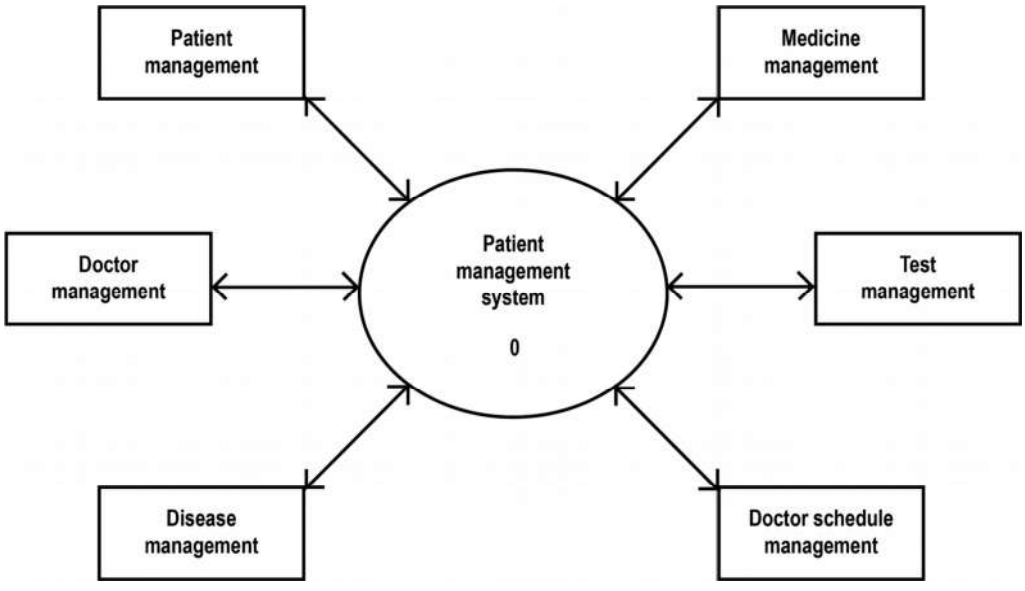
- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan :
  - Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

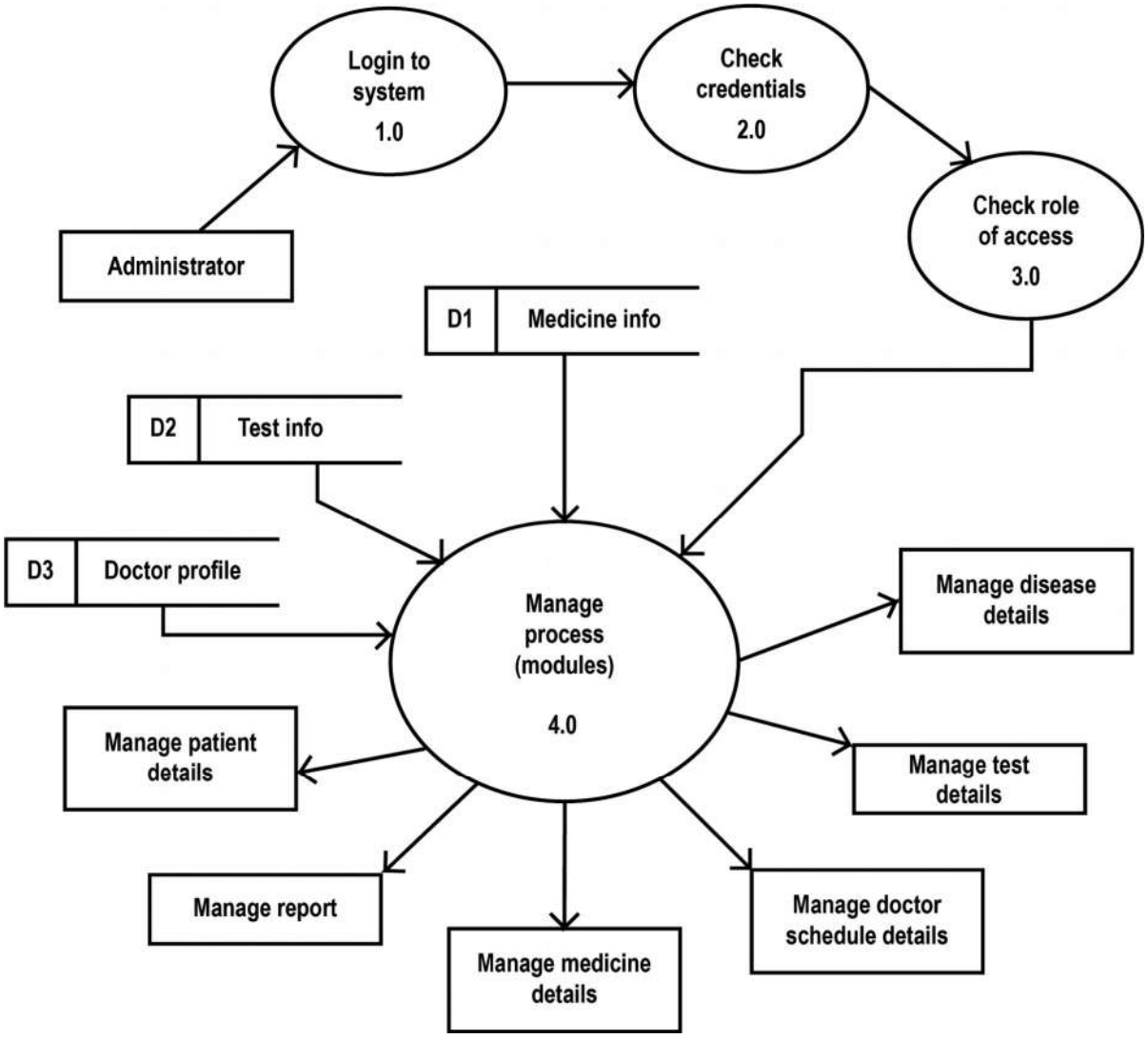
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Login to system
1.0
Check credentials
2.0
Check role of access
3.0
Administrator
D1
Medicine info
D2
Test info
D3
Doctor profile
Manage process (modules)
4.0
Manage disease details
Manage patient details
Manage test details
Manage report
Manage medicine details
Manage doctor schedule details

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. Guest module - In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. Stay module - In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

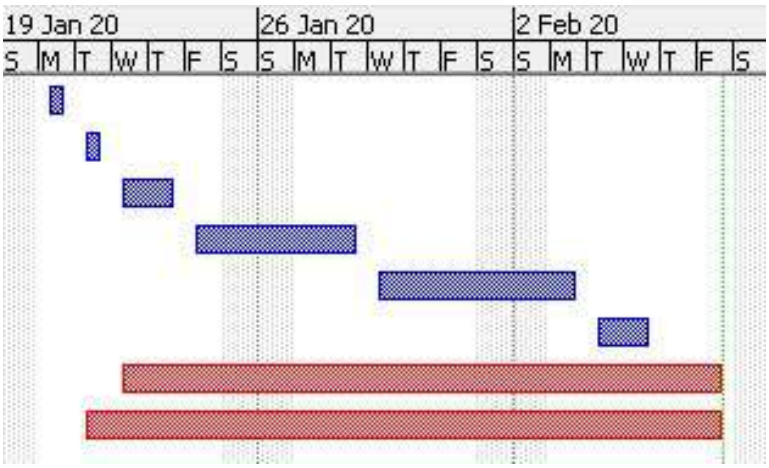
3. Catering module - In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest’s bills by identifying records with that token no.

4. Bill generation module - At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2	Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3	Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4	Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5	Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6	Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7	Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8	User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



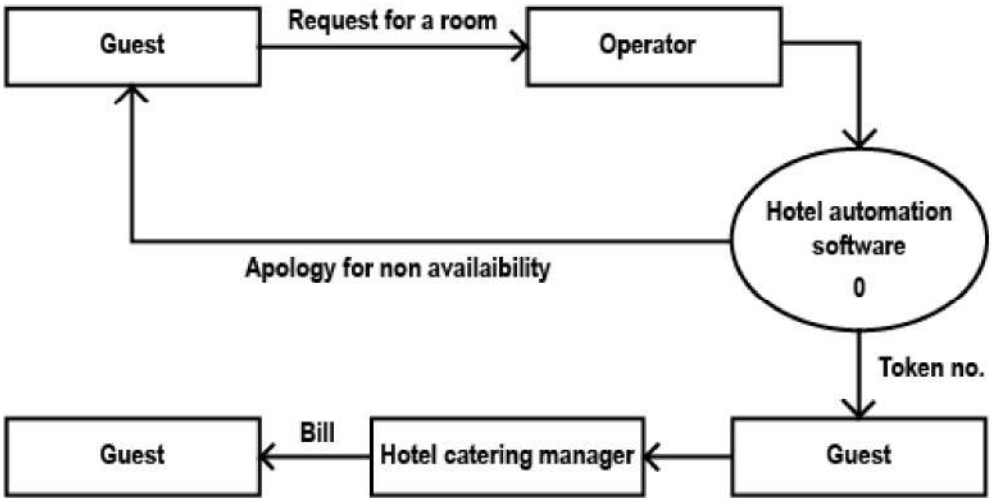
- Project Resources :
  - Hardware Resources: Several computer machines.
  - Human Resources: Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plans
  - Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
      - Priority of risk is the multiplication of above two.
    - Risk leverage =

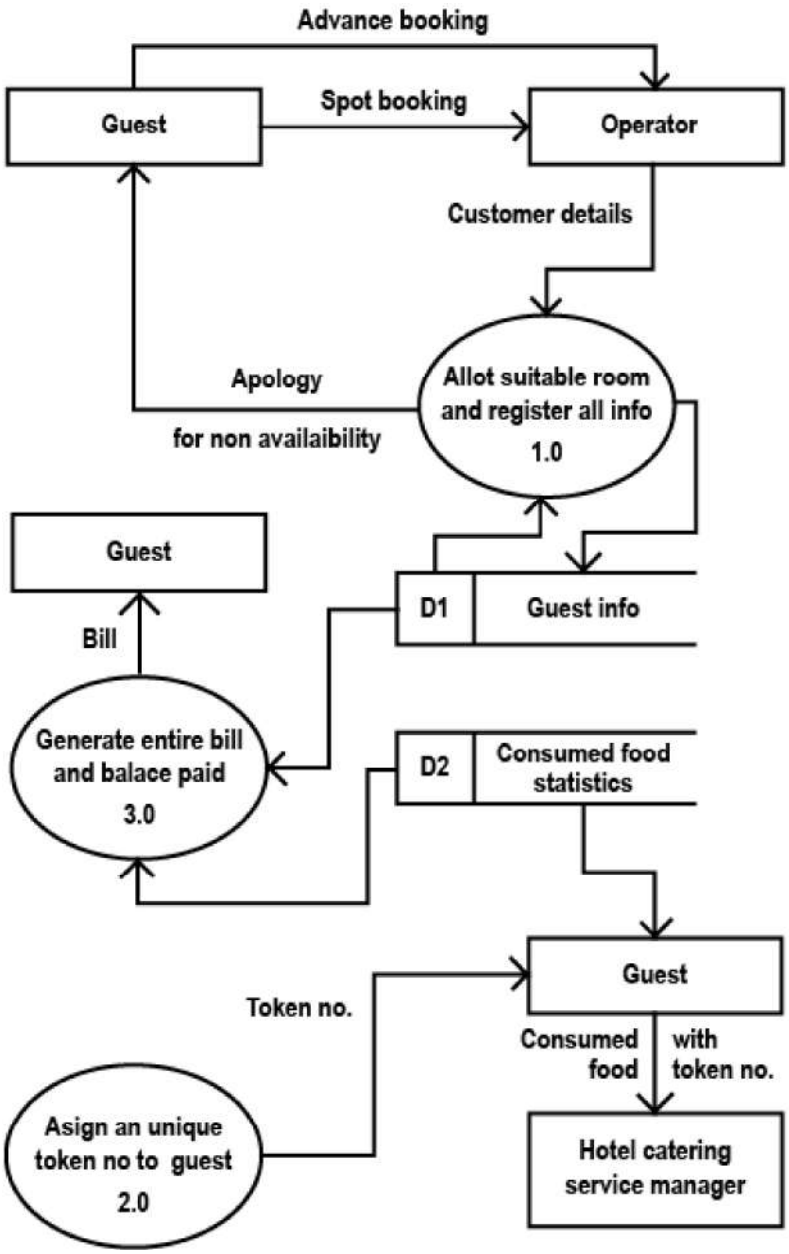
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Advance booking
Guest
Spot booking
Operator
Customer details
Apology
for non availaibility
Allot suitable room
and register all info
1.0
Guest
Bill
D1
Guest info
Generate entire bill
and balace paid
3.0
D2
Consumed food
statistics
Guest
Token no.
Consumed
food
with
token no.
Asign an unique
token no to guest
2.0
Hotel catering
service manager

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Road maintenance is the performance of adaptive, preventive and corrective maintenance for a given road in a locality. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

- Functional requirements : The three modules needed are.
  - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
  - Criminal module : This module accesses data from the database of criminal record details and manages it properly.
  - FIR module : This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
  - Chargesheet module : This module accesses data from the database of chargesheet record details and manages it properly.
  - Court module : The main function of this module is to manage the court profiles.

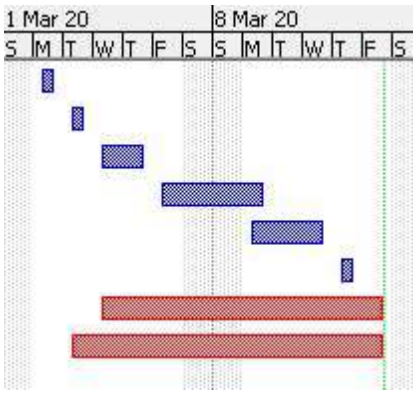
- Non-Functional Requirements :
  - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
  - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

- Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



- Project Resources :
  - Hardware Resources: Several computer machines.
  - Human Resources: Man power resources like project manager, designer, analysist, programmer, tester etc.

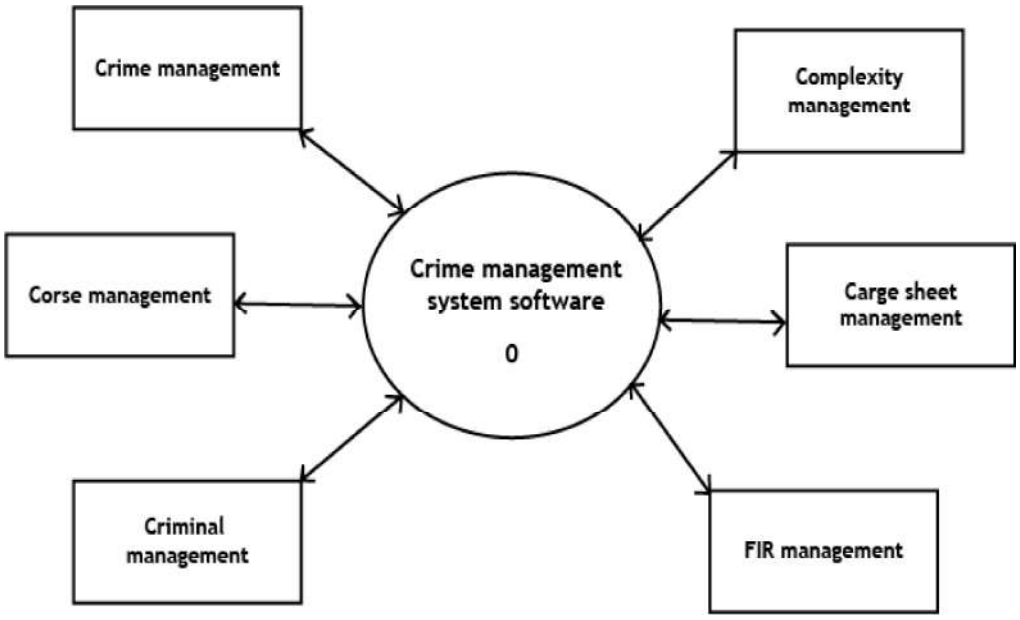
- Risk management plans :
  - Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
      - Priority of risk is the multiplication of above two.
    - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

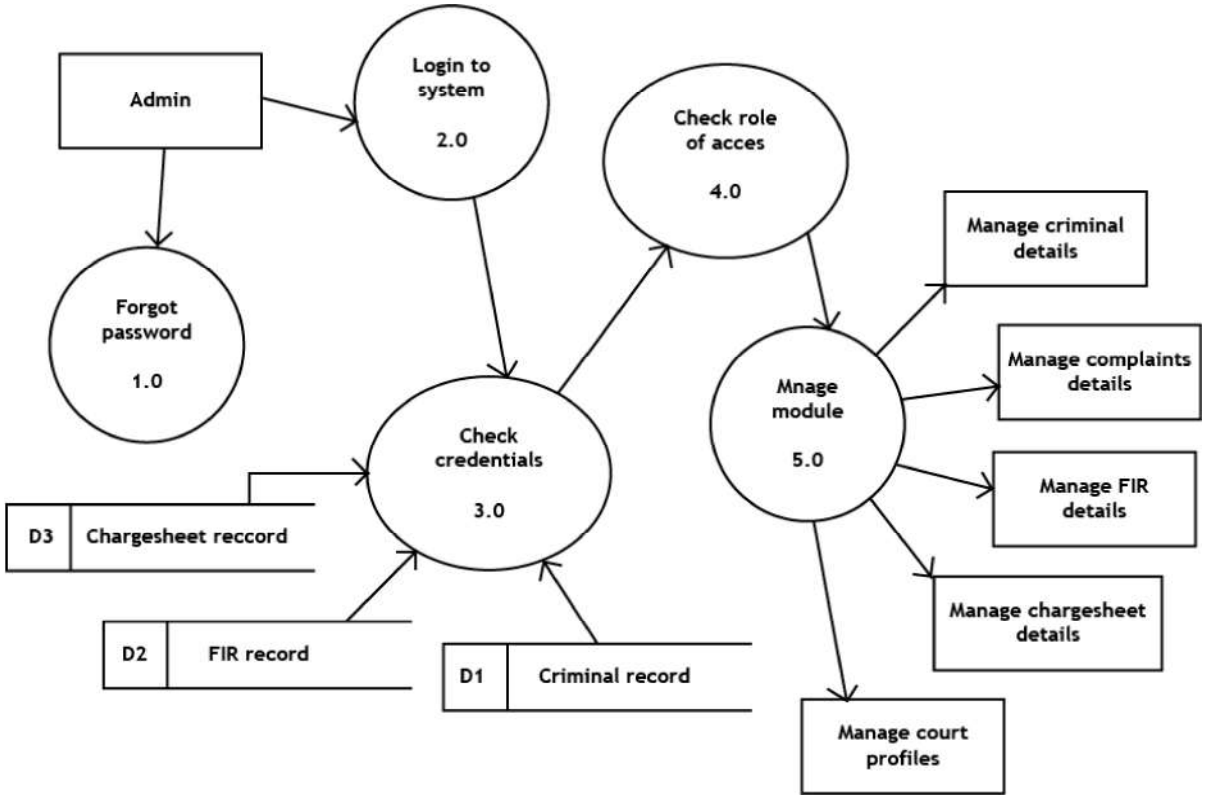
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT :

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project : The aim and objectives for students' examination Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements : The three modules needed are.
    - Login and recovery module : This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
    - Credentials module : This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
    - Administrator module : This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.

- Non-Functional Requirements:
    - Usability: The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

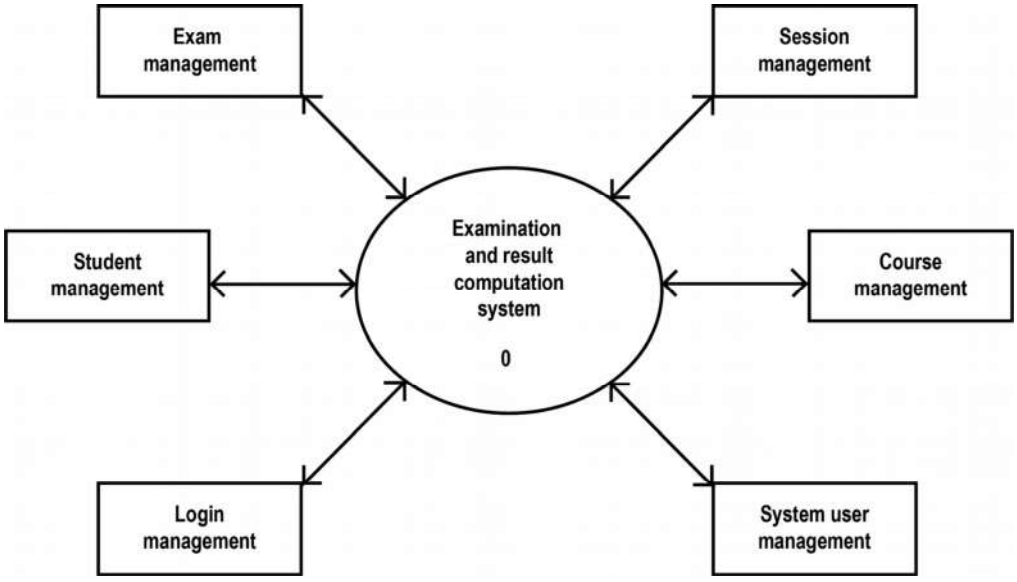
- Project Resources :
    - Hardware Resources: Several computer machines.
    - Human Resources: Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan
  - Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
      - Priority of risk is the multiplication of above two.
    - Risk leverage =

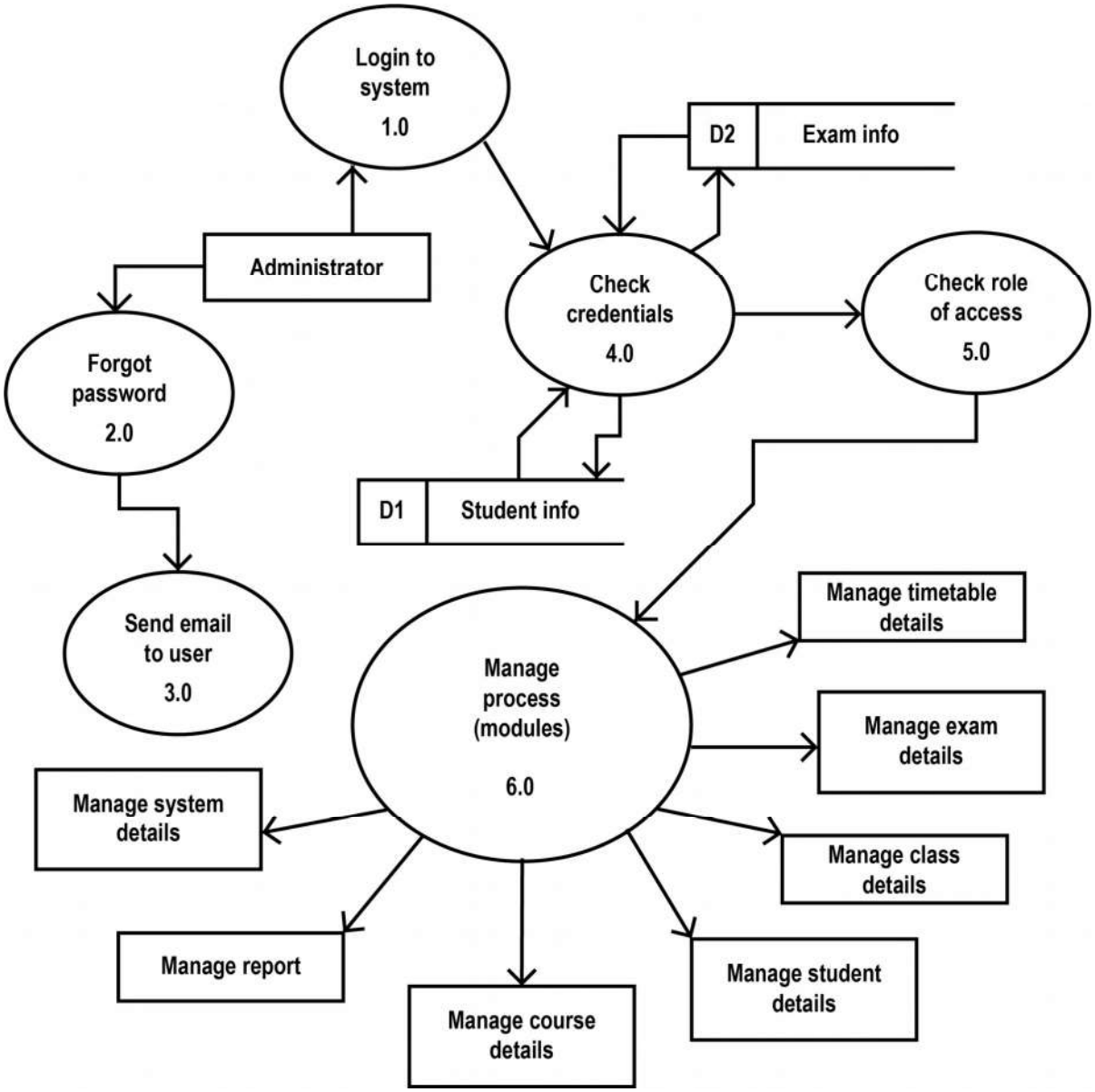
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :

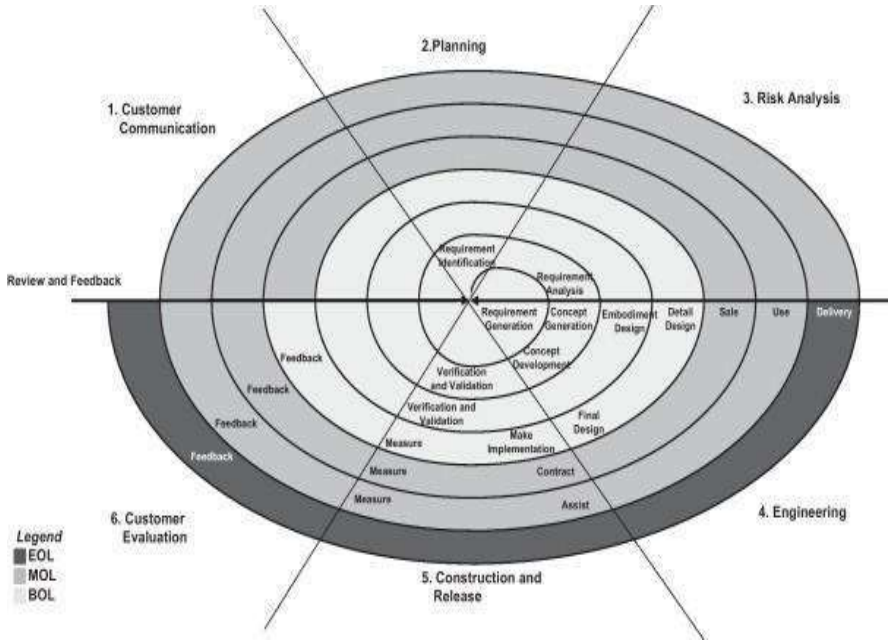

Login to system
1.0
D2
Exam info
Administrator
Check credentials
4.0
Check role of access
5.0
Forgot password
2.0
D1
Student info
Send email to user
3.0
Manage process (modules)
6.0
Manage timetable details
Manage exam details
Manage system details
Manage class details
Manage report
Manage course details
Manage student details

Project 7: Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:
  - ★ Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
  - ★ Spiral Model consists of Risk Analysis task region.
  - ★ In this model, we systematically prepare prototype and release for customer feedback.
- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

➔ Software Requirements Specifications

1. *Goal of the Project:*

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc which was a hefty work.

2. *Functional Requirements:*

As per the requirements of the software from client end.

By analysis we realise that we need three modules:

A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. Parking Module:

This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

3. *Non-Functional Requirements:*

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

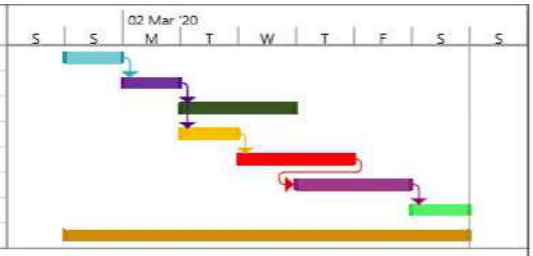
B. Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



Project: Project7
Date: Fri 13-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

b. Project Resources

a. ManPower Resources:
   i. Project Manager
   ii. System Analyst
   iii. Designer
   iv. Developer
   v. Tester
   vi. Document Writer

b. Hardware-Software Resources
   i. Processor: Intel i3 4th gen or above
   ii. Ram: 4Gb or above
   iii. Java Development Kit - 13
   iv. Java Runtime Environment
   v. Sublime Text 3 (Text Editor)
   vi. Ms - Project
   vii. Creately
   viii. Google Docs

c. Risk Management Plan

All risks are measured in terms of :

A. Likelihood of a risk coming through
B. Consequence of the problem associated with that risk

Priority of each risk is multiplication of above two

Risk Leverage =

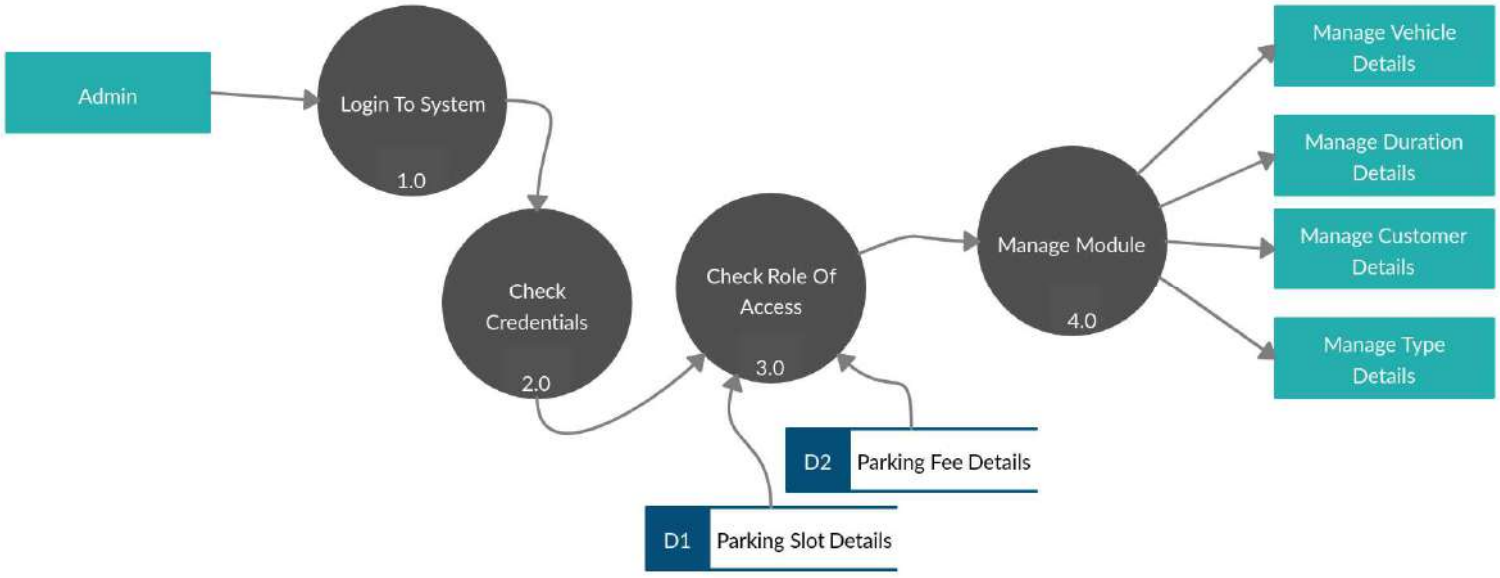
$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

DESIGN ENGINEERING

1. Architectural Design

2. Component Level Design

- **Level 1 DFD**



TESTING

_____Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. **Unit Testing:**
   Each module is tested in isolation
2. **System Testing:**
   The modules are integrated and again tested. This time this testing is done in three parts
   A. Alpha testing(Verification) : done by Development team
   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.
3. **Integration Testing:**
   Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

_____Maintenance stands for all modifications and updations done after the delivery of software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project-8

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

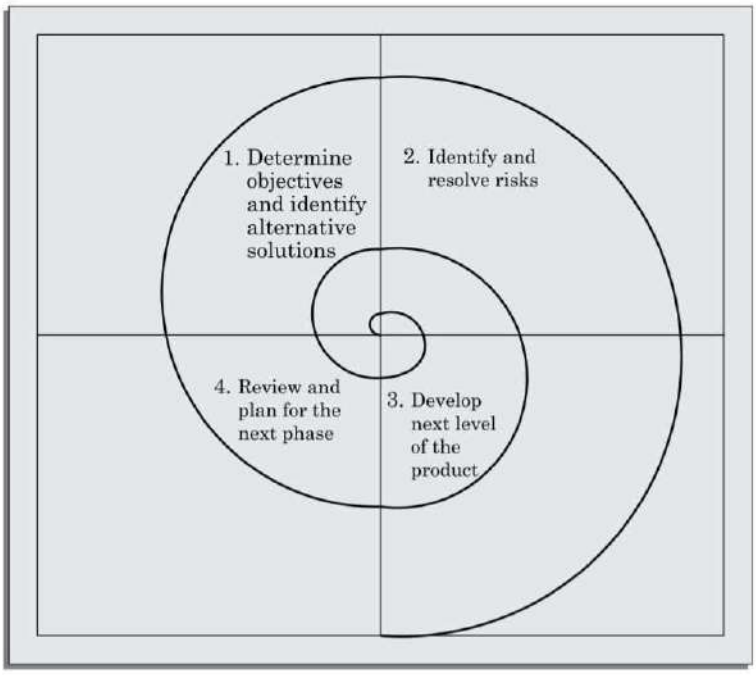


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

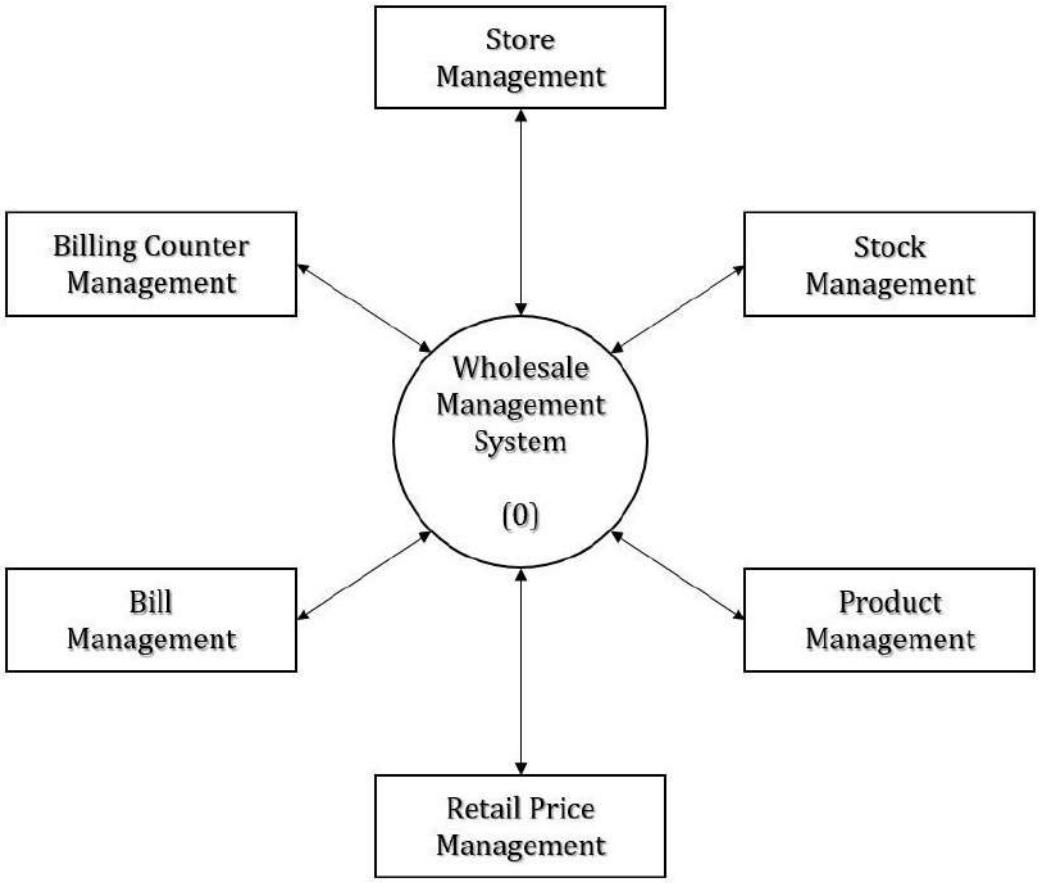


Fig – DFD level zero

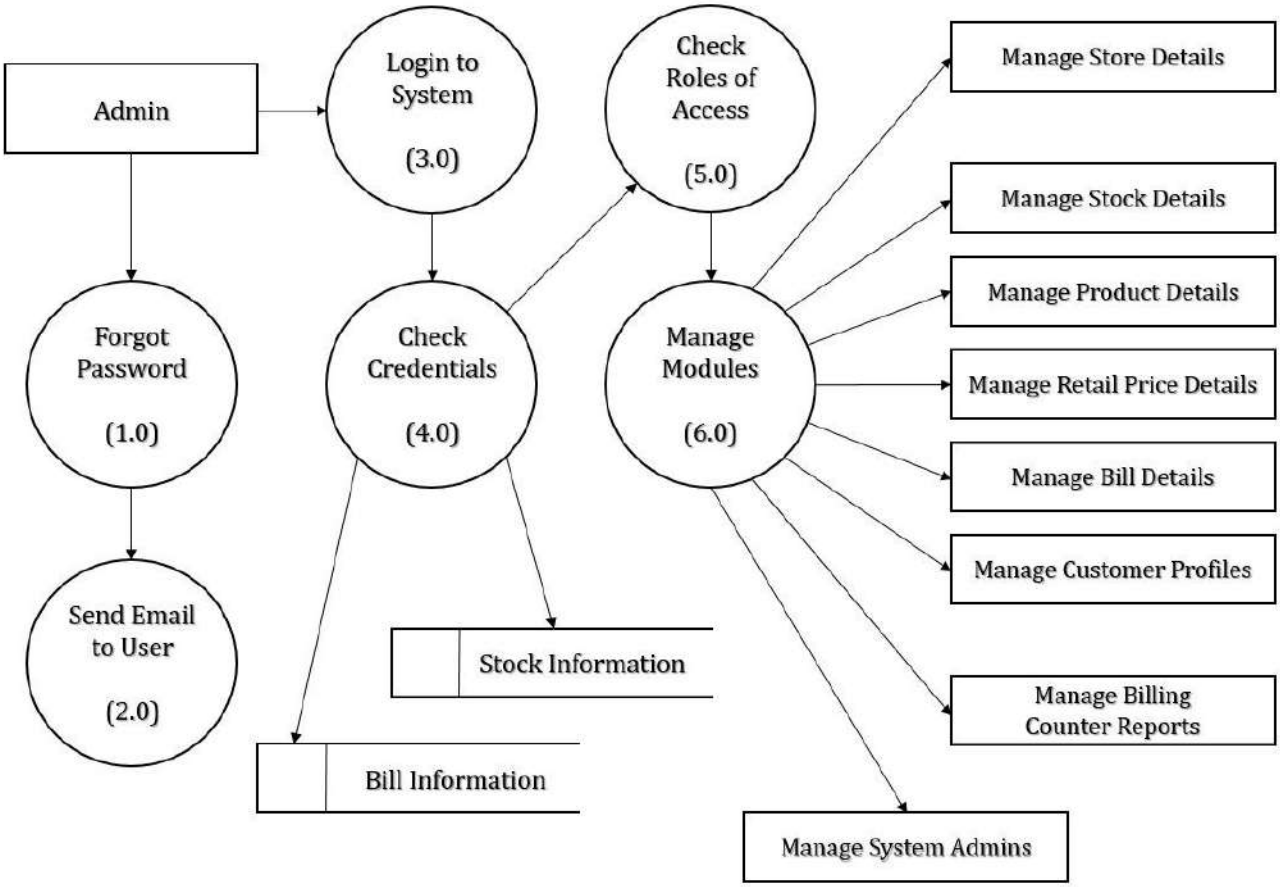


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or

reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Name: Sayan Ghosh
College Roll: 1719
Registration No: A01-1112-117-017-2018
Exam Roll no: 2021141310
Semester: V
Dept: Comp. Sc.

Supervisor: Prof. Manas Pal

PROJECT 1

ORGANIZED RETAIL SHOPPING MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

A supermarket needs to develop the following software to encourage regular customers. For this, the customer needs to supply residence address, contact number, license number etc. Each customer who registers for this scheme is assigned a "Customer Number" (CN). A customer can present 'CN' to customer staff when he makes purchases. In this case, the value of his purchase is credited against its 'CN'. At the end of the year, the supermarket intends to award surprise gifts to 10 customers who make the highest total purchase over the year. Also it intends to award a 22 carat gold coin to every customer whose purchased-amount exceeds 1 lakh. The entries against the CN are reset on the last day of every year, after the prize winner lists are generated.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Incremental Model for the following reasons :
    - This life cycle model is referred to as successive versions model.
    - In this model, the development team first develops the core modules of the system - This initial product skeleton is redefined into increasing levels of capability by adding new functionality.
- Each evolutionary version is developed using iterative waterfall model.

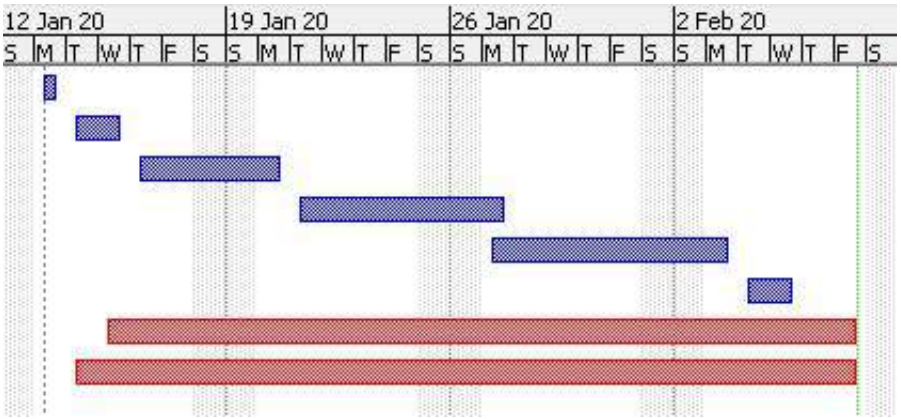
REQUIREMENT ANALYSIS (SRS):

As per the request of the client end, at first we have to create a module for Customer Registration to give every customer a unique customer ID. Then we have to create a module to add the purchase value with respect to the customer to his/her account's purchase history. Then we have to create a module to give a surprise give and gold coin gift to selected candidates at the year end. At last, we have to refresh the customer purchase history after the gift is given to freshly count the purchase history of the customer on the New Year.

PROJECT MANAGEMENT :

- **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM

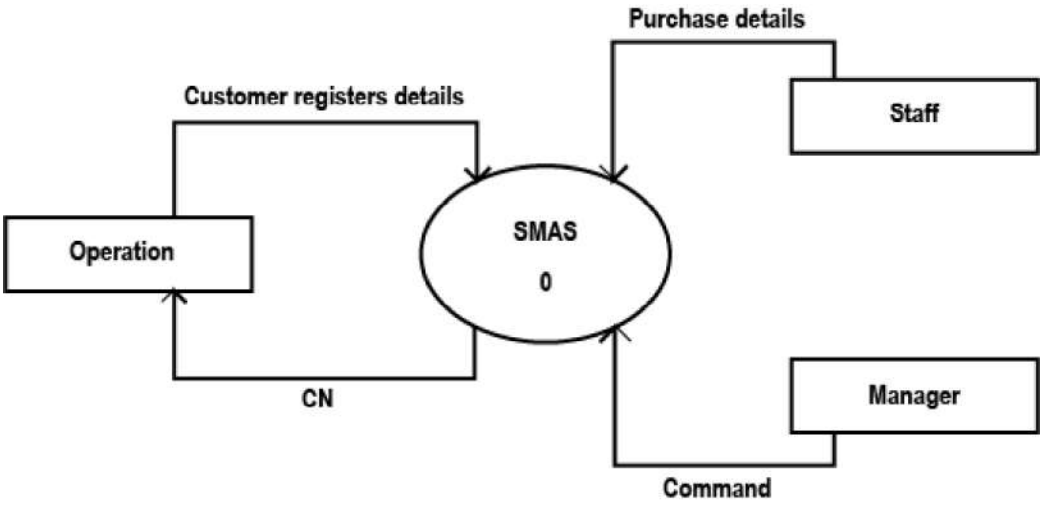


- **Project Resources :**
  - **Hardware Resources :** Several computer machines.
  - **Human Resources :** Manpower resources like project manager, designer, analysist, programmer, testeretc.
- **Risk management plan :** Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

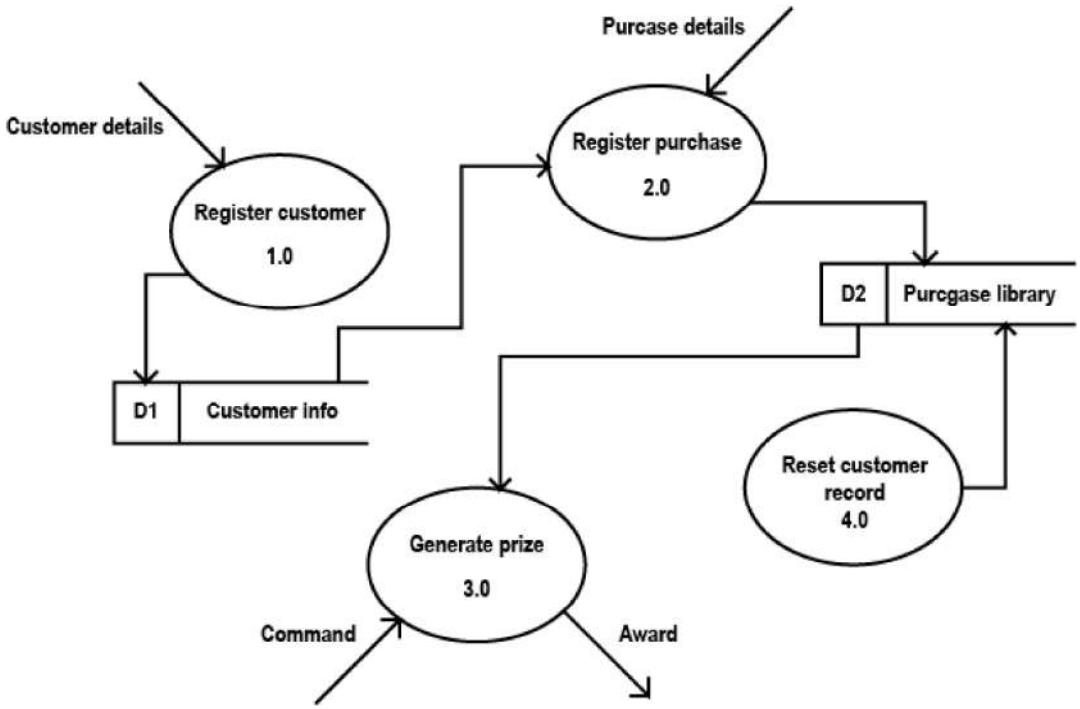
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

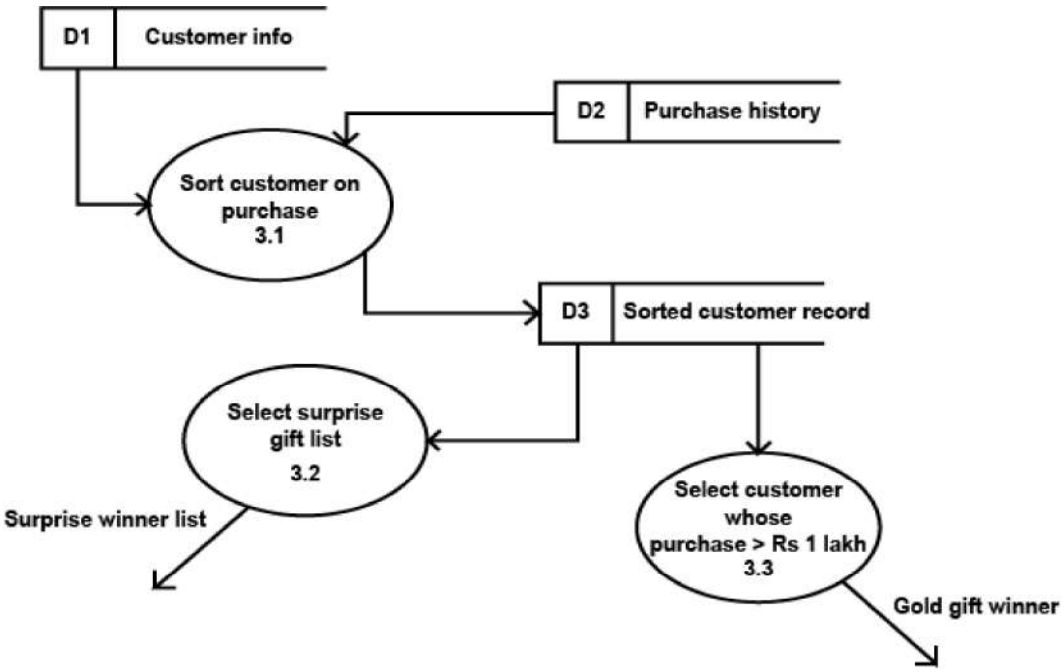
1. **Level 0 DFD :**



2. Level 1 DFD :



3. Level 2 DFD of process 3.0 :



TESTING :

Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

MAINTENANCE :

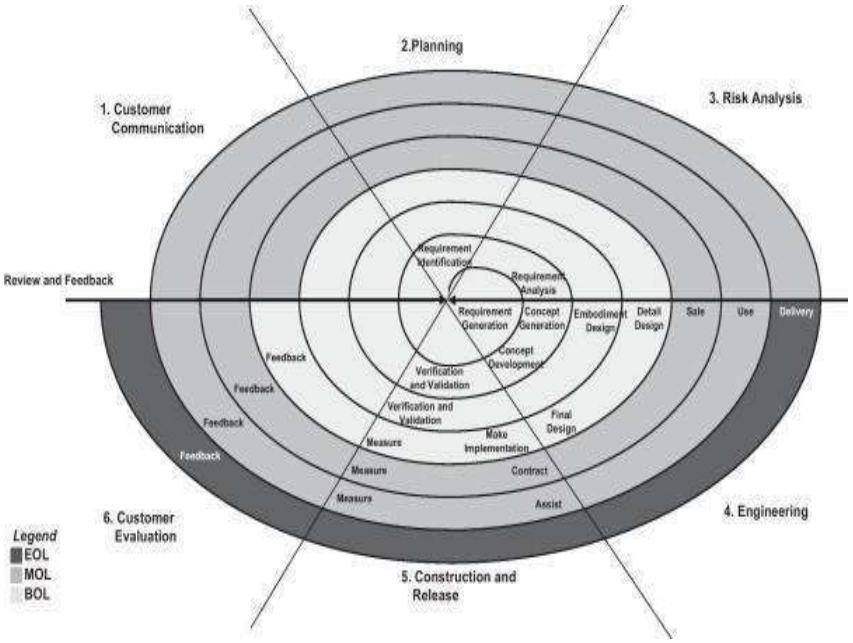
Maintenance stands for all modifications and updations done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project 2: Route Management System

PROBLEM STATEMENT

It identifies internaldatastores of payment,bus route, customer,ticket,ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule.It also shows the details of ticket counter, bus route,payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:
  - ★ Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
  - ★ Spiral Model consists of Risk Analysis task region.
  - ★ In this model, we systematically prepare prototype and release for customer feedback.
- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. ***Goal of the Project:***

   The software is made to manage the various datastores related to route,payment,schedule of bus.This would reduce the hectic job of bus authority to monitor reports.

2. ***Functional Requirements:***

   As per the requirements of the software from client end. By analysis we realise that we need five modules:

   A. **Admin Module:**

   This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.Only one admin can make more admin accounts for them to access data in software.
   Input:UserName and Password

   This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.
   Input:Answer to security question
   Output:Link for changing password

   B. **Customer Module:**

   This module has function to monitor customer details like from and to place,distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

   C. **Schedule Module:**

   This module maintains timings ofthe bus. This includes function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

   D. **Route Module:**

   This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

   E. **Report Module:**

   At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

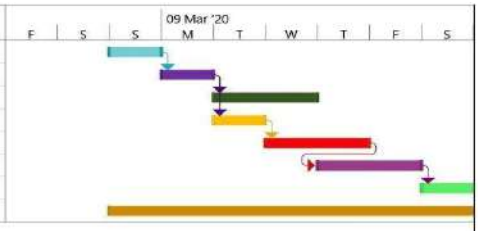
3. ***Non-Functional Requirements:***

   A. **Usability:The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.**
   B. **Portability: The software should have all modules encapsulated in such a manner that uses memory eciently and takes optimal storage for being portable.**
   C. **Robustness:The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.**

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 08-03-20	Sun 08-03-20	
2	Customer Communication	1 day	Mon 09-03-20	Mon 09-03-20	1
3	Planning	2 days	Tue 10-03-20	Wed 11-03-20	2
4	Risk Analysis	1 day	Tue 10-03-20	Tue 10-03-20	2
5	Engineering	2 days	Wed 11-03-20	Thu 12-03-20	4
6	Construction and Release	2 days	Thu 12-03-20	Fri 13-03-20	5
7	Customer Evaluation	1 day	Sat 14-03-20	Sat 14-03-20	6
8	Write User Manual	7 days	Sun 08-03-20	Sat 14-03-20	



Project: Project8
Date: Sat 14-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

b. Project Resources

a. **ManPower Resources**:
   i. **Project Manager**
   ii. **System Analyst**
   iii. **Designer**
   iv. **Developer**
   v. **Tester**
   vi. **Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above**
   ii. **Ram: 4Gb or above**
   iii. **Java Development Kit -13**
   iv. **Java Runtime Environment**
   v. **Sublime Text 3 (Text Editor)**
   vi. **Ms - Project**
   vii. **Creately**
   viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. **Likelihood of a risk coming through**
B. **Consequence of the problem associated with that risk**

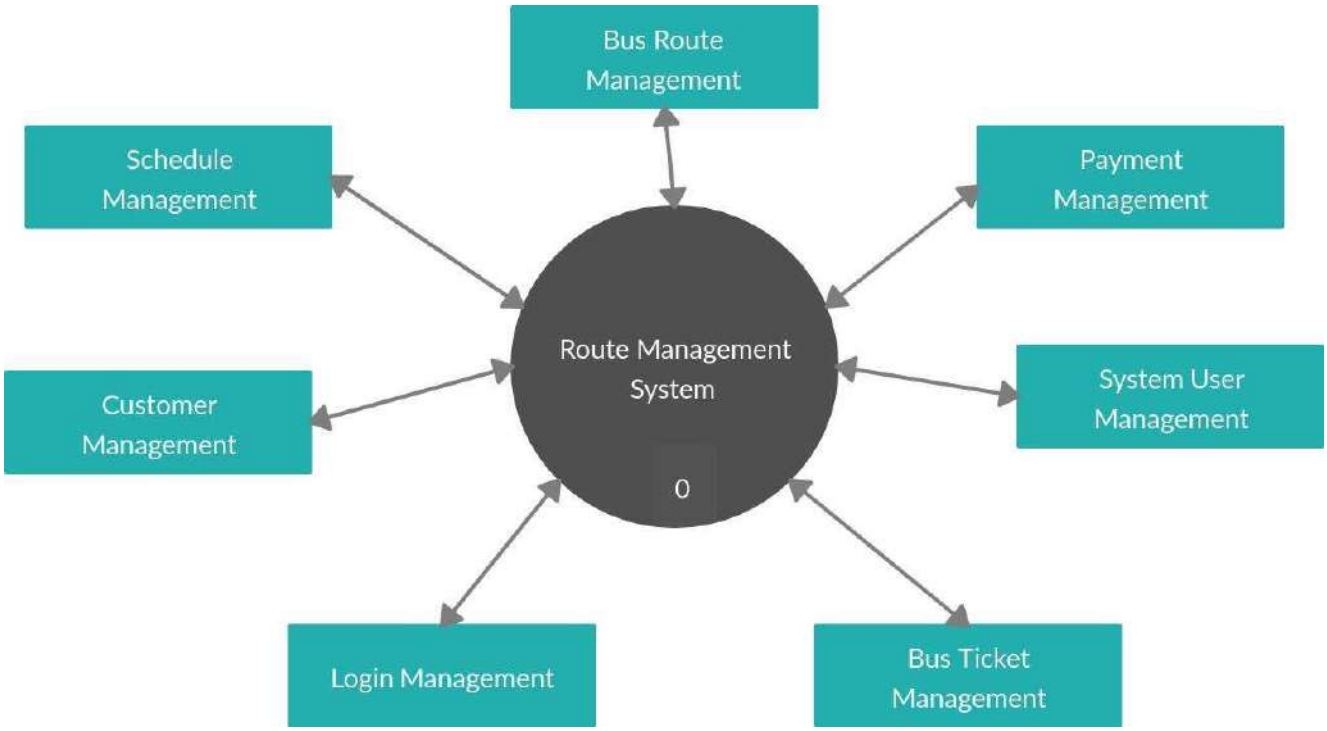
Priority of each risk is multiplication of above two

Risk Leverage =

$$\frac{Risk\ exposure\ before\ Reduction - Risk\ exposure\ after\ Reduction}{Cost\ of\ Reduction}$$

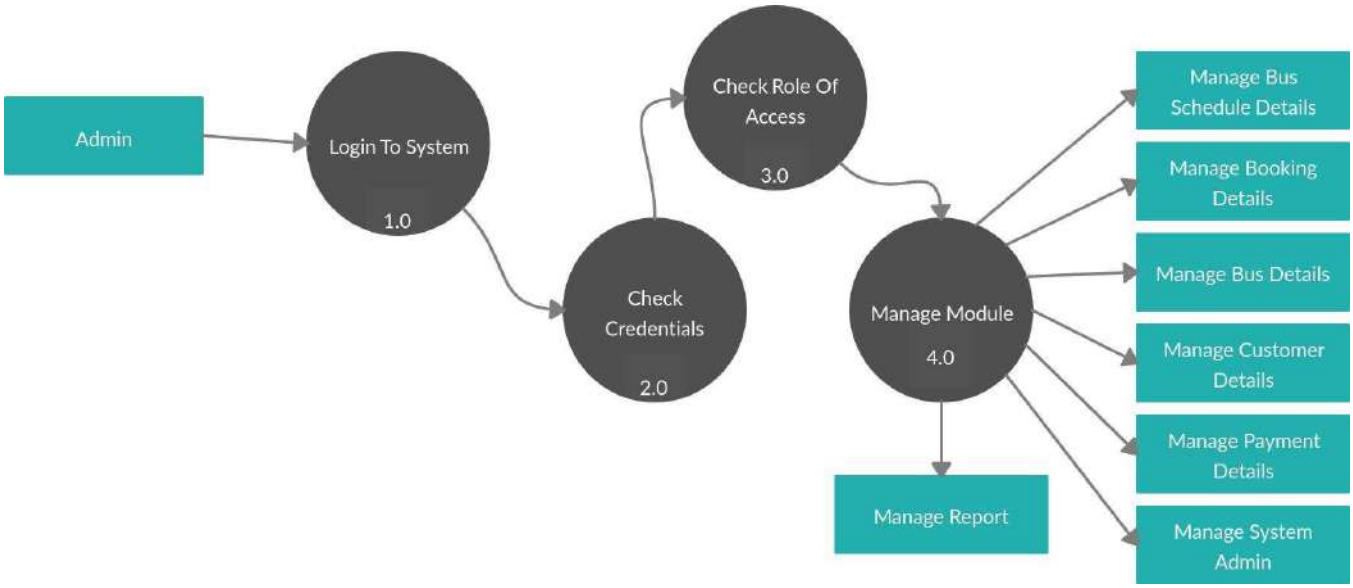
DESIGN ENGINEERING

1. Architectural Design



2 Component Level Design

- **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. **Unit Testing:**

   Each module is tested in isolation

2. **System Testing:**

   The modules are integrated and again tested. This time this testing is done in three parts

   A. Alpha testing(Verification) : done by Development team
   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. **Integration Testing:**

   Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software.
Admin can edit, elete, view records of patients, test, disease and manage all the details of medicines, doctor schedule.
Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project: The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

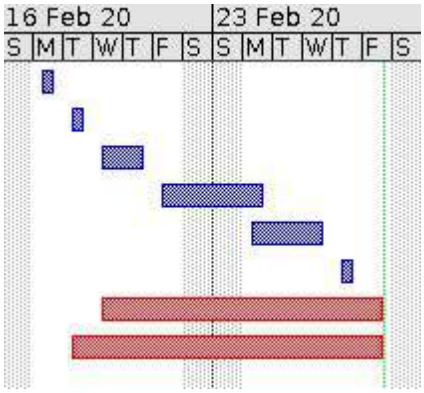
- Functional requirements: The three modules needed are.
    - Administrator module: This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.
    - Doctor module: This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.
    - Report module: This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.
    - Diagnostic module: This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.

- Non-Functional Require:ments
    - UsabilityT:he software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - PortabilityT:he software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.
    - RobustnessT:he software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

		Name	Duration	Start	Finish
1		Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2		Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3		Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4		Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5		Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6		Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7		Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8		User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



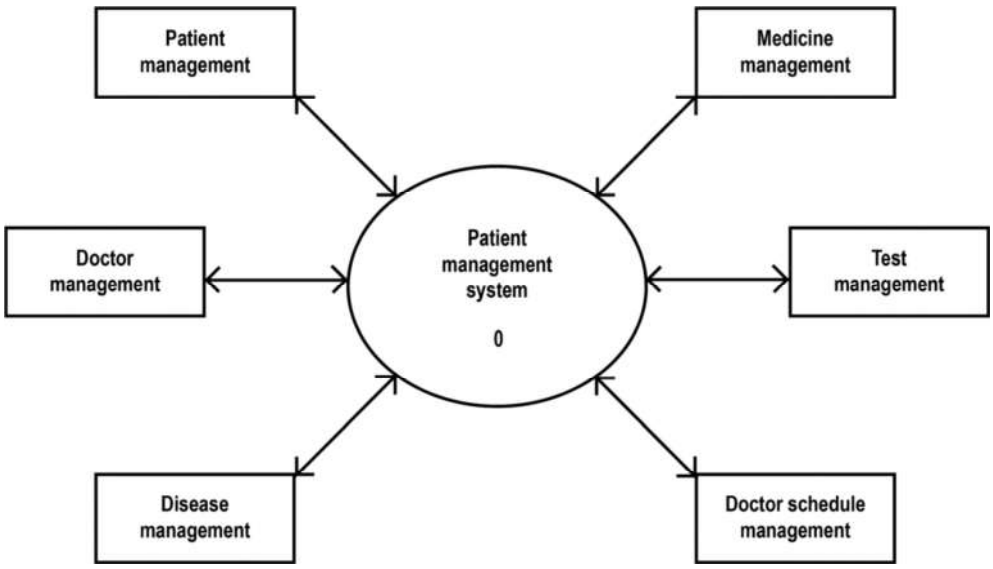
- Project Resources :
    - Hardware Resources : Several computer machines.
    - Human Resources : Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan :
    - Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
        - Priority of risk is the multiplication of above two.
    - Risk leverage =

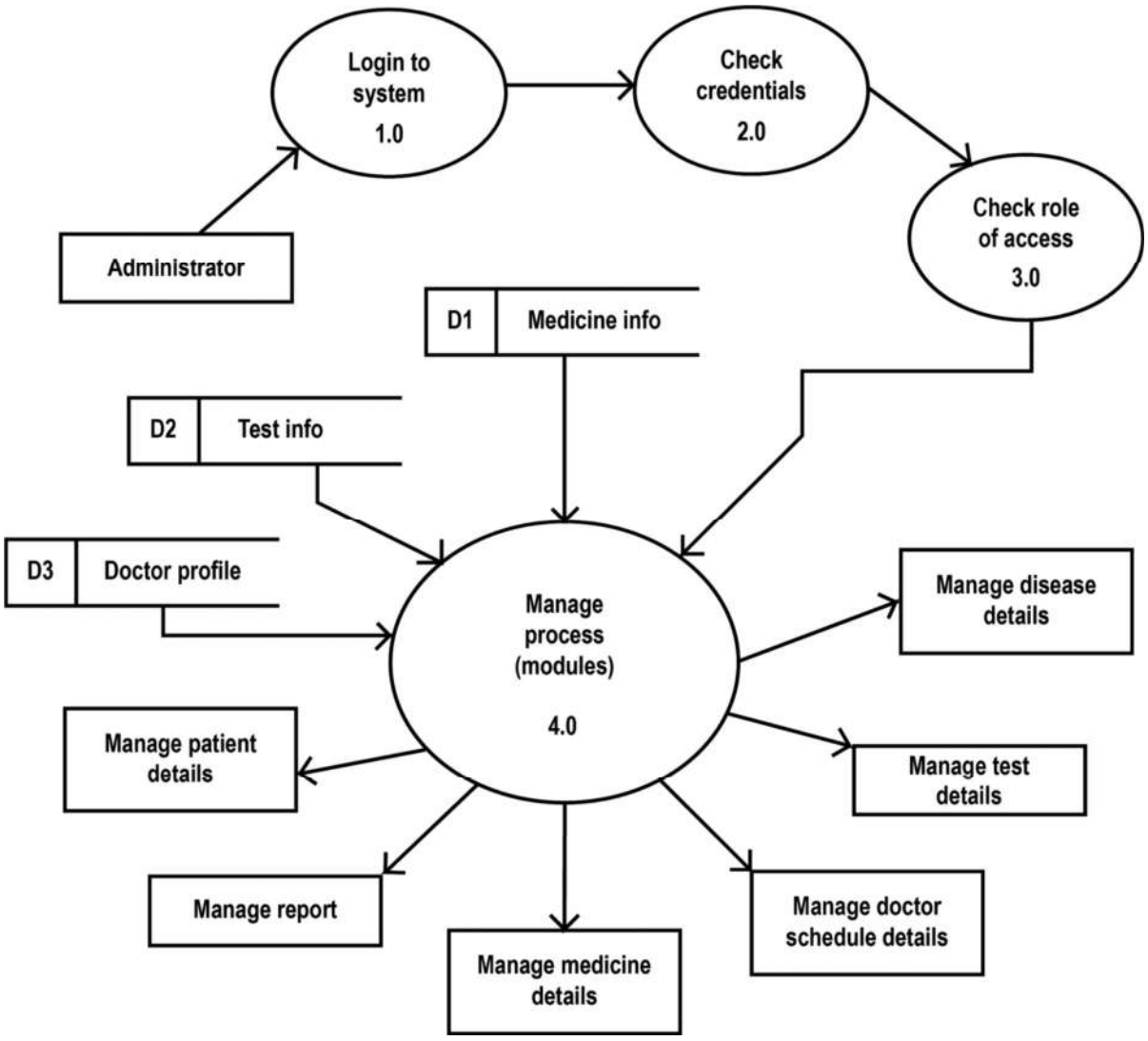
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Login to system
1.0
Check credentials
2.0
Check role of access
3.0
Administrator
D1
Medicine info
D2
Test info
D3
Doctor profile
Manage process (modules)
4.0
Manage disease details
Manage patient details
Manage test details
Manage report
Manage medicine details
Manage doctor schedule details

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification): It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT :

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to checkout the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. Guest module: In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.

2. Stay module: In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

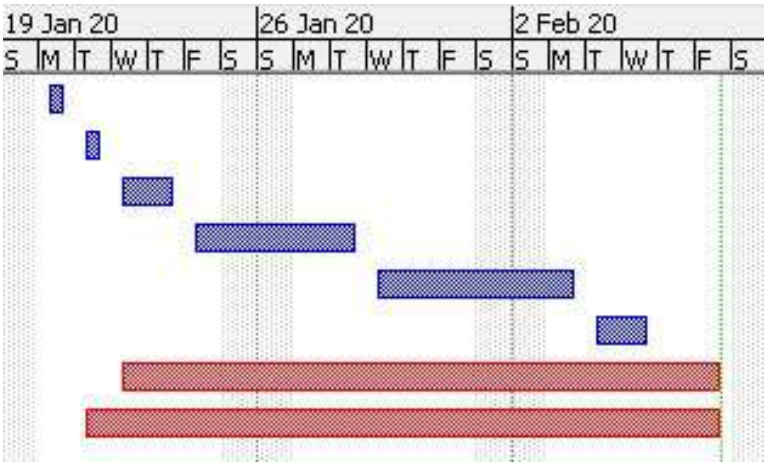
3. Catering module: In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no. . Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.

4. Bill generation module: At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2	Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3	Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4	Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5	Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6	Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7	Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8	User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



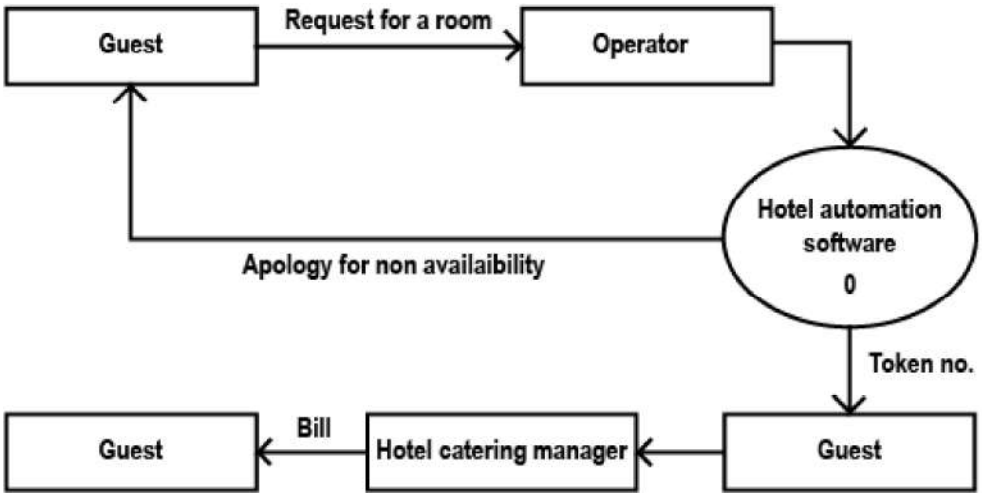
- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plans:
  - Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk.
    - Priority of risk is the multiplication of above two.
  - Risk leverage =

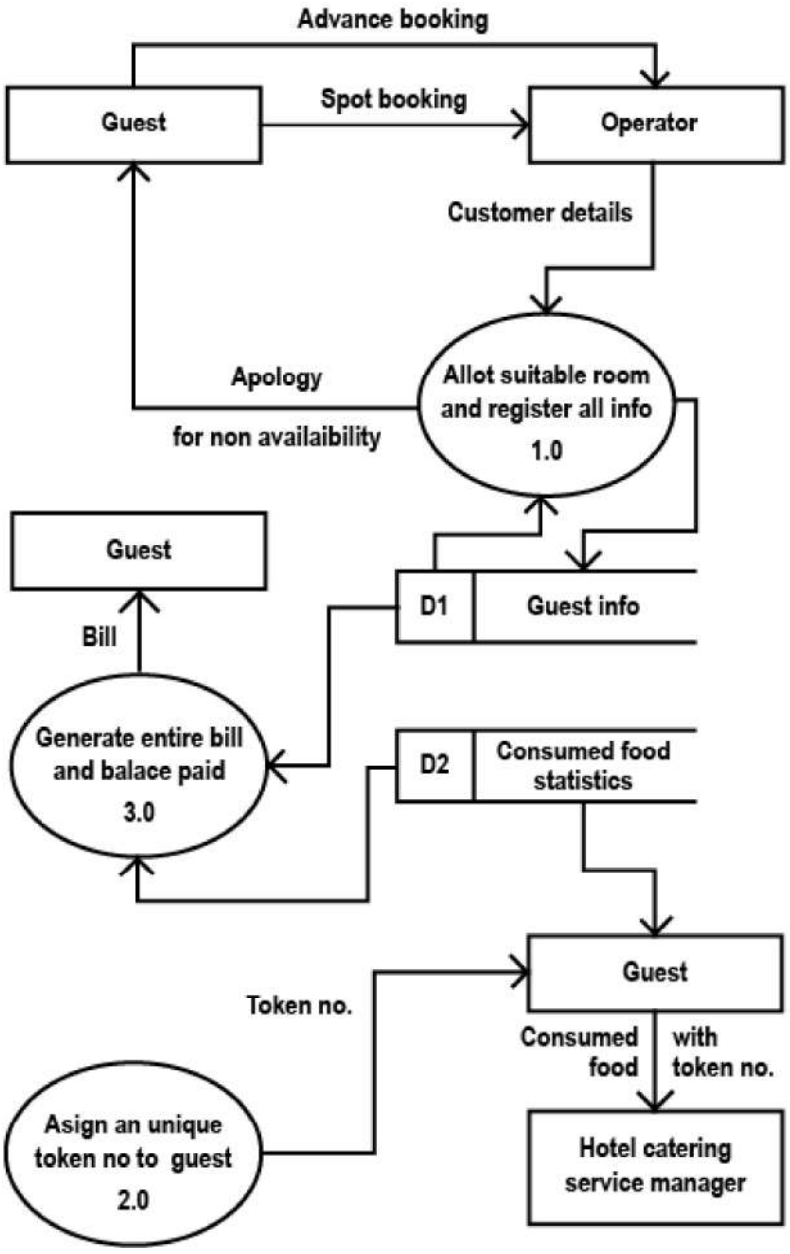
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Advance booking
Guest
Spot booking
Operator
Customer details
Apology
for non availaibility
Allot suitable room
and register all info
1.0
Guest
D1
Guest info
Bill
Generate entire bill
and balace paid
3.0
D2
Consumed food
statistics
Guest
Token no.
Consumed
food
with
token no.
Asign an unique
token no to guest
2.0
Hotel catering
service manager

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification): It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an
item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification): It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Road maintenance is the performance of adaptive, preventive and corrective maintenance for a given road in a locality. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT :

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, chargesheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
    - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project: The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

- Functional requirements: The three modules needed are.
    - Administrator module: This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.
    - Criminal module: This module accesses data from the database of criminal record details and manages it properly.
    - FIR module: This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.
    - Chargesheet module: This module accesses data from the database of chargesheet record details and manages it properly.
    - Court module: The main function of this module is to manage the court profiles.

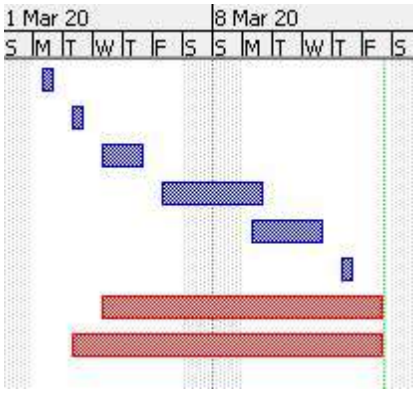
- Non-Functional Requirements:
    - Usability: The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

- Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



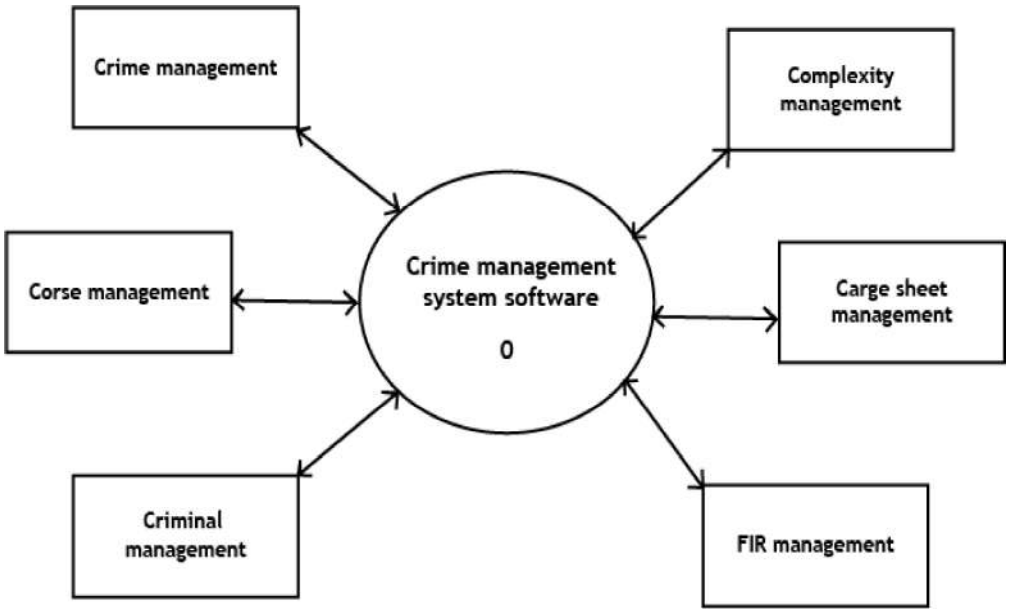
- Project Resources :
  - Hardware Resources: Several computer machines.
  - Human Resources: Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan :Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
        - Priority of risk is the multiplication of above two.
    - Risk leverage =

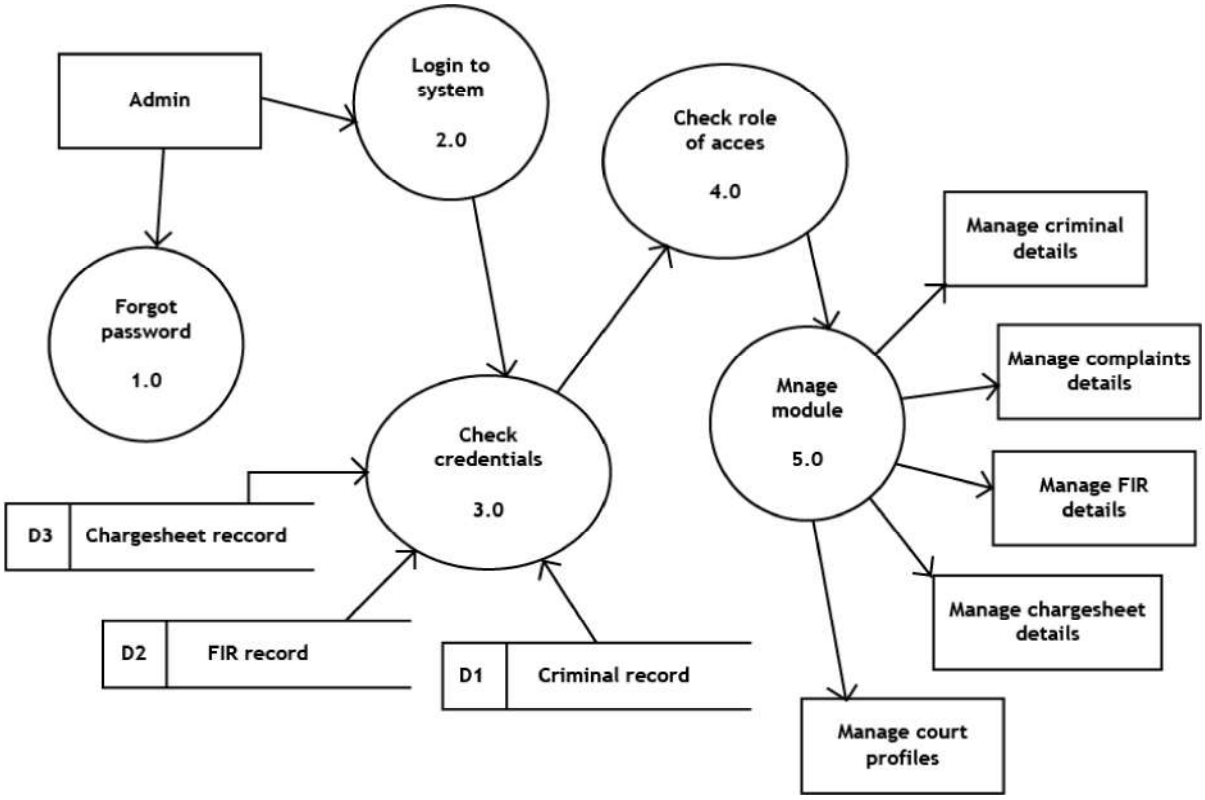
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :



TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification): It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT :

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realise that we need three modules.

- Goal of the project: The aim and objectives for students' examination Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements: The three modules needed are.
    - Login and recovery module: This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
    - Credentials module: This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
    - Administrator module: This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting
      and viewing the record of all time tables. It also tracks the detailed information of result computation of students.

- Non-Functional Require:ments
    - UsabilityT:he software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - PortabilityT:he software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

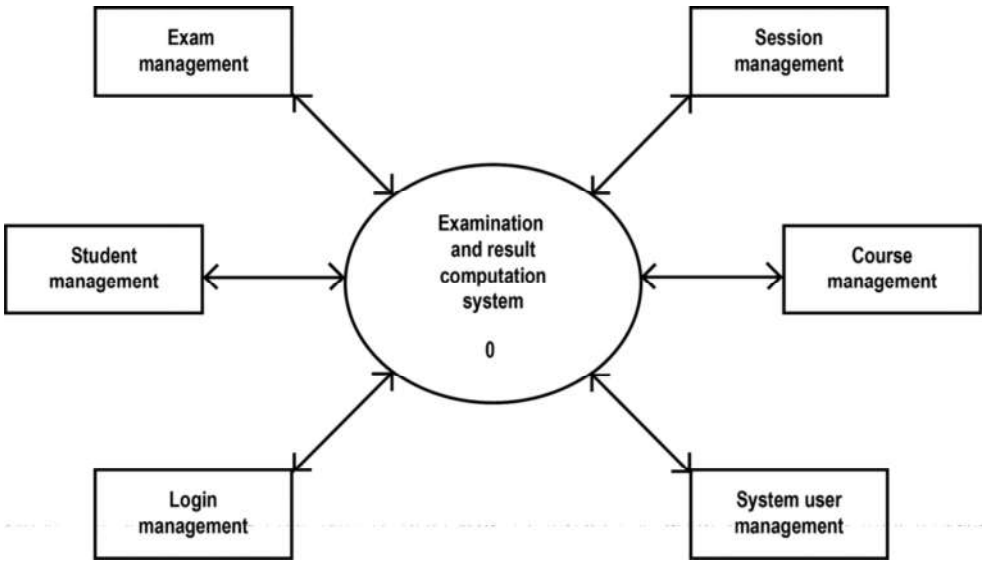
- Project Resources :
    - Hardware Resources:Several computer machines.
    - Human Resources:Man power resources like project manager, designer, analysist, programmer, tester etc.

- Risk management plan
  - Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
      - Priority of risk is the multiplication of above two.
    - Risk leverage =

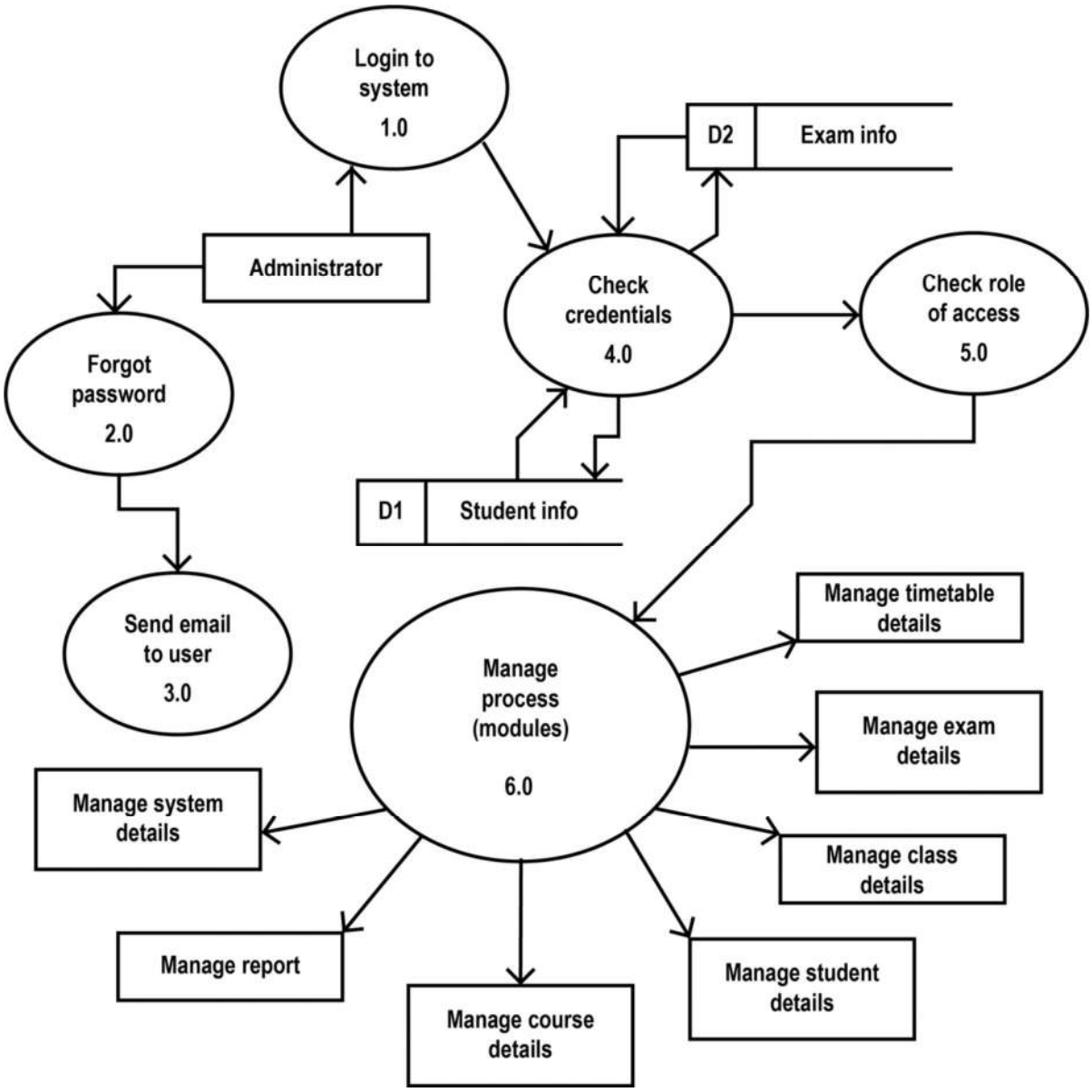
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :

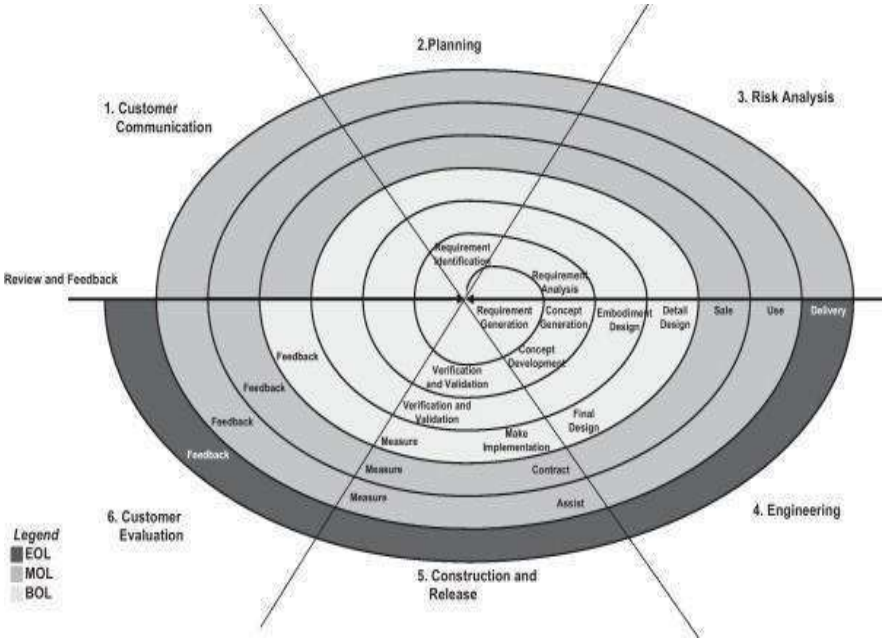

Login to system
1.0
D2
Exam info
Administrator
Check credentials
4.0
Check role of access
5.0
Forgot password
2.0
D1
Student info
Send email to user
3.0
Manage process (modules)
6.0
Manage timetable details
Manage exam details
Manage class details
Manage student details
Manage course details
Manage report
Manage system details

Project 7: Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**
  - **★ Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
  - **★ Spiral Model consists of Risk Analysis task region.**
  - **★ In this model, we systematically prepare prototype and release for customer feedback.**
- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

→Software Requirements Specifications

1. *Goal of the Project:*

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc which was a hefty work.

2. *Functional Requirements:*

As per the requirements of the software from client end.

By analysis we realise that we need three modules:

A. **Admin Module:**

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. **Vehicle Module:**

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. **Parking Module:**

This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

3. *Non-Functional Requirements:*

A. **Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.**

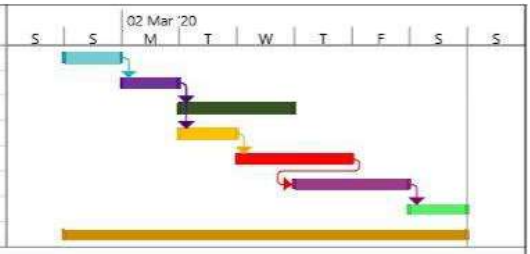
B. **Portability: The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.**

C. **Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.**

PROJECT MANAGEMENT

a. Project Scheduling

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



Project: Project7
Date: Fri 13-03-20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 1

b. Project Resources

a. **ManPower Resources:**
   i. **Project Manager**
   ii. **System Analyst**
   iii. **Designer**
   iv. **Developer**
   v. **Tester**
   vi. **Document Writer**

b. **Hardware-Software Resources**
   i. **Processor: Intel i3 4th gen or above**
   ii. **Ram: 4Gb or above**
   iii. **Java Development Kit -13**
   iv. **Java Runtime Environment**
   v. **Sublime Text 3 (Text Editor)**
   vi. **Ms - Project**
   vii. **Creately**
   viii. **Google Docs**

c. Risk Management Plan

All risks are measured in terms of :

A. Likelihood of a risk coming through
B. Consequence of the problem associated with that risk

Priority of each risk is multiplication of above two

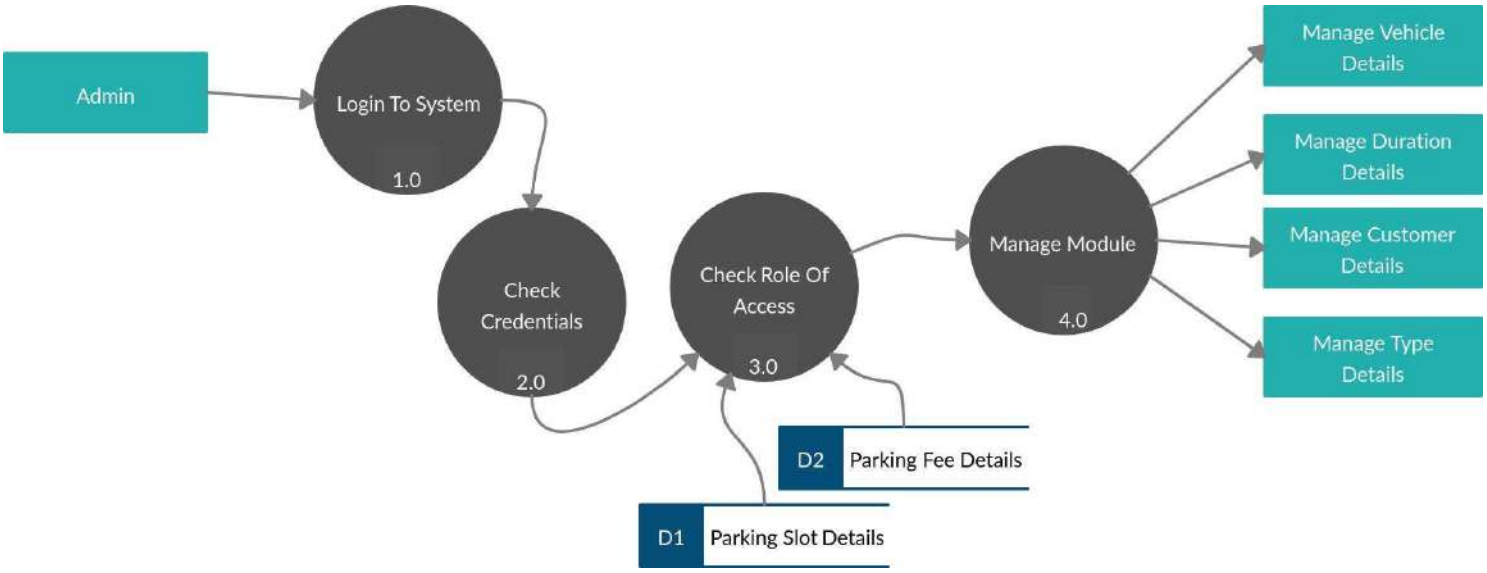
$$\text{Risk Leverage} = \frac{\textit{Risk exposure before Reduction} - \textit{Risk exposure after Reduction}}{\textit{Cost of Reduction}}$$

DESIGN ENGINEERING

1. Architectural Design

2. Component Level Design

- **Level 1 DFD**



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. **Unit Testing:**
   Each module is tested in isolation
2. **System Testing:**
   The modules are integrated and again tested. This time this testing is done in three parts
   A. Alpha testing(Verification) : done by Development team
   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.
3. **Integration Testing:**
   Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

Project-8

WHOLESALE MANAGEMENT SYSTEM

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

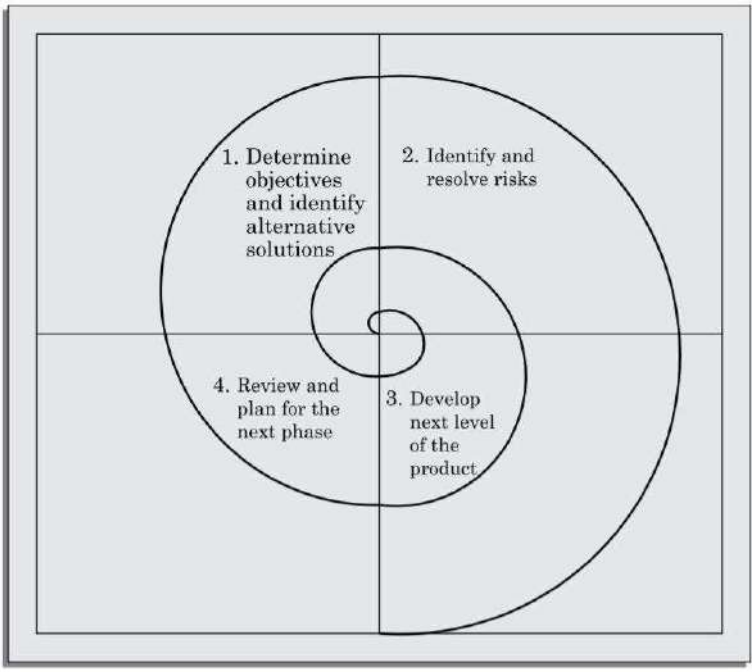


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module** – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module** – In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module** – In this module, all the bills are maintained. A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module** – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

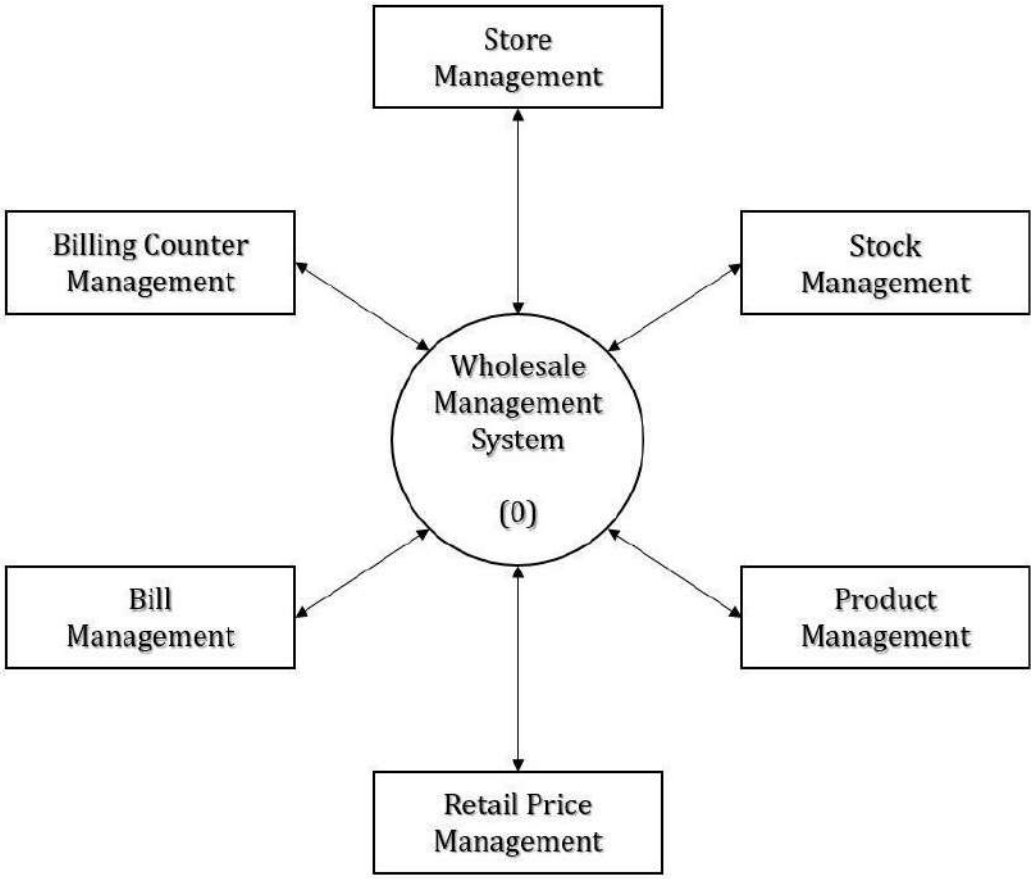


Fig – DFD level zero

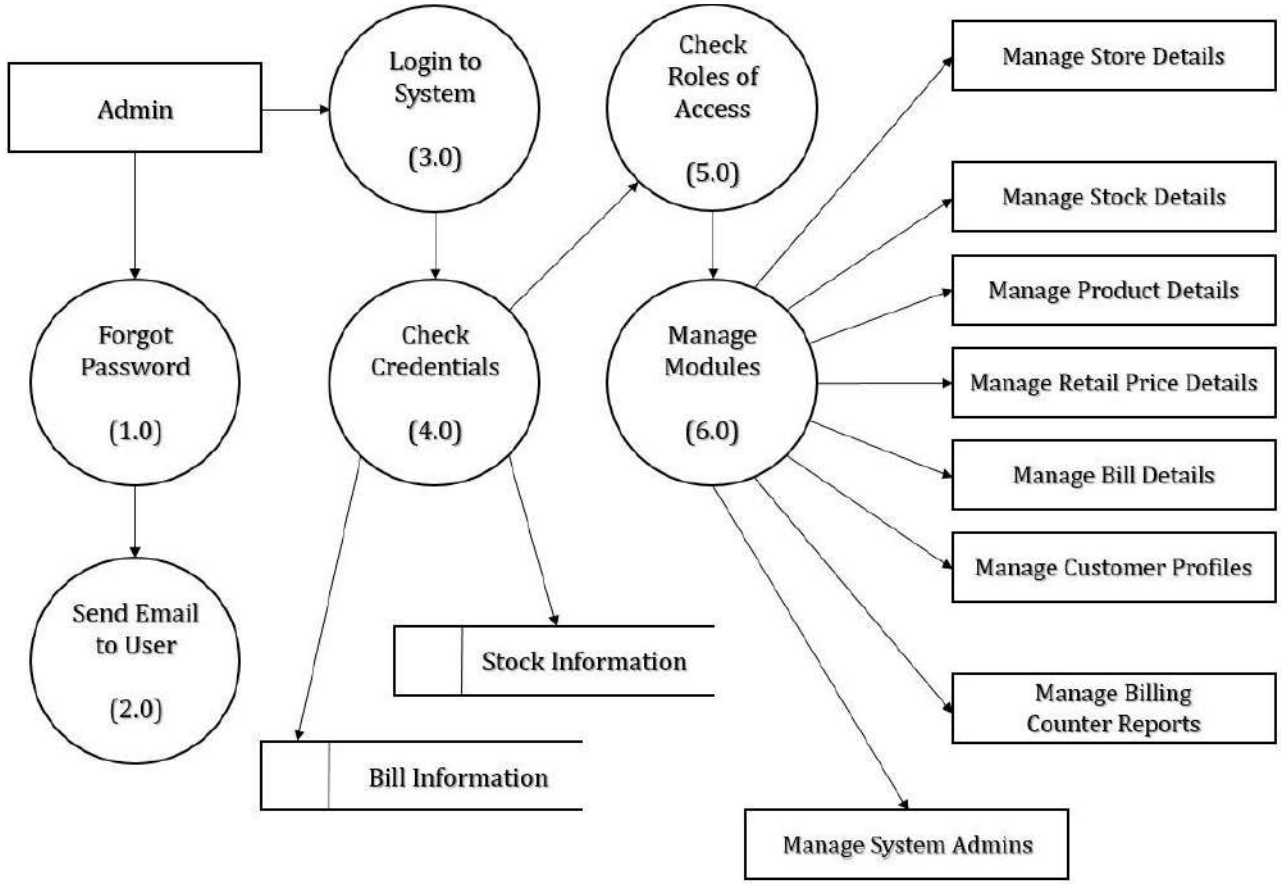


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p** – priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

- **α (Alpha) Testing** – It is the system testing performed by the development team in a controlled environment.
- **β (Beta) Testing** – It is the system testing performed by friendly set of customers in an uncontrolled environment.
- **Acceptance Testing** – It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or

reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance** – Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance** – Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance** – Porting the software to work in a new environment.

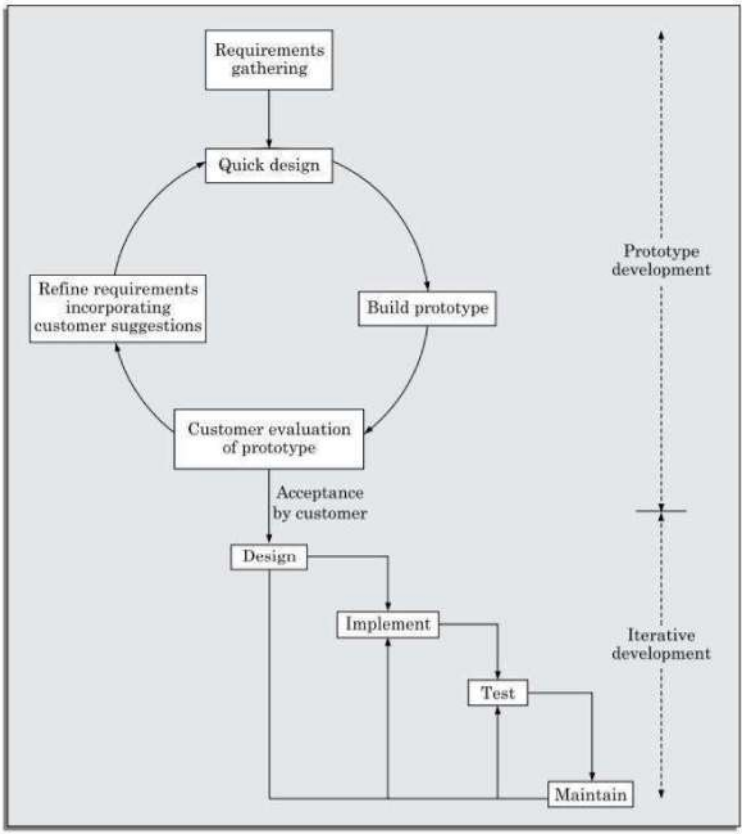
RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: Koustav Bal
College Roll No.: 1 730
Registration No.: A01-1112-117-026-2018
Examination Roll No.: 2021141311
Paper Code: CMSADSE3
Semester: V
Department: Computer Science
Supervisor: Dr. Biswajit Biswas

RETAILSHOPPINGMANAGEMENTSOFTWARE

ProblemStatement–Asupermarketneedstodevelopthefollowingsoftwareto encourageregularcustomers.Forthis,thecustomerneedstosupplyhis/her residentialaddress,mobilenumberanddrivinglicense.Eachcustomerwho registersforthisschemeisassignedauniquecustomernumber(CN)bythe computer.AcustomercanpresentCNtothecheckoutstaffwhenanypurchase ismade.Inthiscase,thevalueofthispurchaseiscreditedagainsttheCN.Atthe endofeachyear,thesupermarketintendstoawardsurprisegiftstoten customerswhomakethehighesttotalpurchaseovertheyear.Alsoitintendsto awarda22karatgoldcointoeverycustomerwhosepurchaseexceedsINR 10000.TheentriesagainsttheCNareresetonthelastdayofeveryyearafterthe prizewinners'listisgenerated.

ProcessModel–Prototypingmodel



Fig–prototypingmodelofsoftwaredevelopment
Thismodelallowsallorpartofasystemtobeconstructedquickly.

Goalofprototypingistoreducethechanceofuncertainty.Thephases include–

1)meetingofthedevelopersandthecustomerstodefinetheoverall objectivesofthesoftware
2)quickdesignleadstoconstructionofprototype
3)evolutionofprototypebycustomer
4)iterationoccursastheprototypeisturnedtotheneedsofthecustomers
5)thecodeforprototypeisthrownaway–theexperiencegathered fromdevelopingtheprototypehelpsindevelopingtheactual system

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredand assignedauniquecustomernumber(CN). Adatabaseismaintainedforkeepingthecustomerdetails.

Input:customerdetails
Output:uniqueCN

2.PurchaseModule–Inthismodule,anypurchasemadebythe customeriscreditedagainsttheCN. Adatabaseismaintainedforkeepingthepurchasehistory.

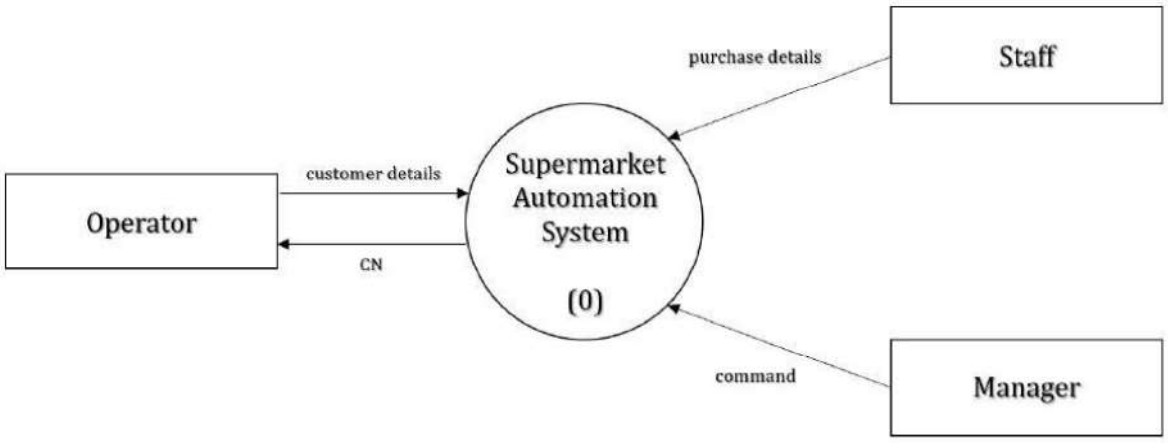
Input:purchasemadebythecustomer,CN
Output:updateddatabase

3.PromotionalModule–Inthismodule,customersareselectedforgiving outtheprizesandtheprizewinners'listisgenerated. TheentriesagainsttheCNareresetafterthegenerationoftheprize winners' list.

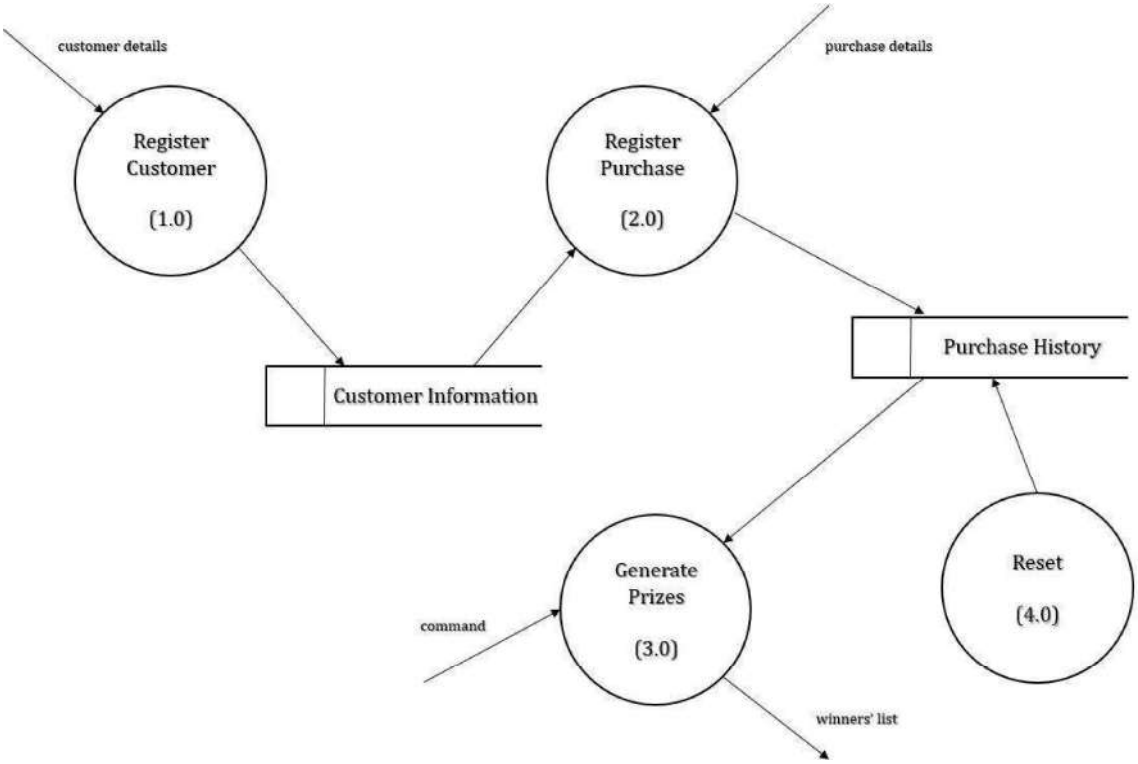
Input:purchasehistory,CN
Output:prizewinners'list

DesignEngineering

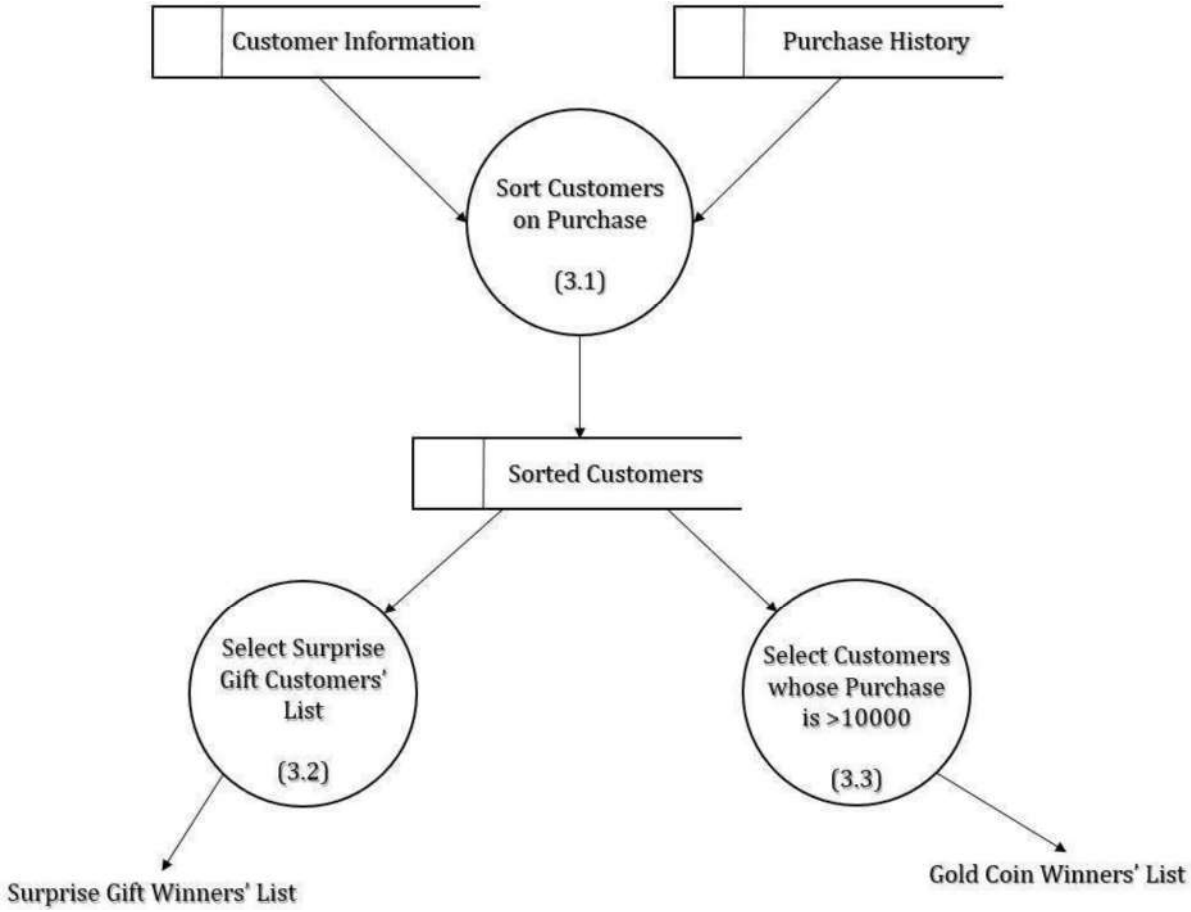
DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



Fig–DFDlevelone

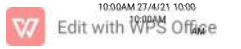


Fig–DFDleveltwoofProcess3.0

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	
2	FeasibilityStudy	1	0:00AM	
3	RequirementAnalysisandSpecifications	3days	101:40/04/A	2M1
4	Design	5days	20/4/21	
5	CodingandUnitTesting	1	0:00AM	



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s

wherep–prioritywithwhichtheriskmustbehandled

r–theprobabilityoftheriskbecomingtrue

s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtestingphase, themodulesareintegratedinaplannedmanner.Systemtestingconsistsofthree differentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

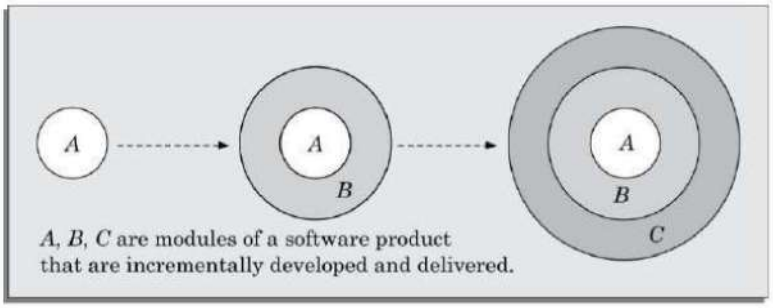
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

ROUTEMANAGEMENTSYSTEM(DTCROUTEINFORMATION)

Problemstatement–Itisanonlineinformationaboutthebusroutesandtheir frequenciesandfares.Itshowsallthefunctionalitiesofthebusbookingand ticketingsystem.Italsoidentifiesinternaldatastoresofpayments,busroutes, customers,tickets,ticketcountersthatmustbepresentinorderforthebus ticketsystemtodoitsjobandshowstheflowofdataamongvariouspathsof busschedule,ticketcounter,busroute,paymentandcustomerofthesystem. Finallyafterprocessing,itgeneratesreportofallbookings,alltickets,bus schedulesandallpayments.

Processmodel–Incrementalmodel



Fig–incrementalmodelofsoftwaredevelopment

Thislifecyclemodelisalsoreferredtoassuccessiveversionsmodel.Inthislife cyclemodel,thesoftwareisfirstbrokendownintoseveralmoduleswhichcanbe incrementallyconstructedanddelivered.Thedevelopmentteamfirstdevelops thecoremodulesofthesystem.Thisinitialproductskeleton(A)isrefinedinto increasinglevelsofcapabilitybyaddingnewfunctionalities.Eachevolutionary versioncanbedevelopedbyiterativewaterfallmodel.Theusergetsachanceto experimentwiththepartiallydevelopedsoftwaremuchbeforethecomplete versionofthesystemisreleased.

SoftwareRequirementAnalysis

1. CustomerModule–Inthismodule,acustomerisregisteredandthe profileismaintained.

   Input:customerdetails
   Output:uniqueprofile

2.BusTicketingModule–Inthismodule,anybookingmadebythe customerisregisteredandtheticketisgenerated.

Input:customerdetails
Output:generatedticket

3.InformationModule–Inthismodule,thebusdetailsofthebookingmade bythecustomerismaintained.
Adatabaseismaintainedforkeepingtheinformation.

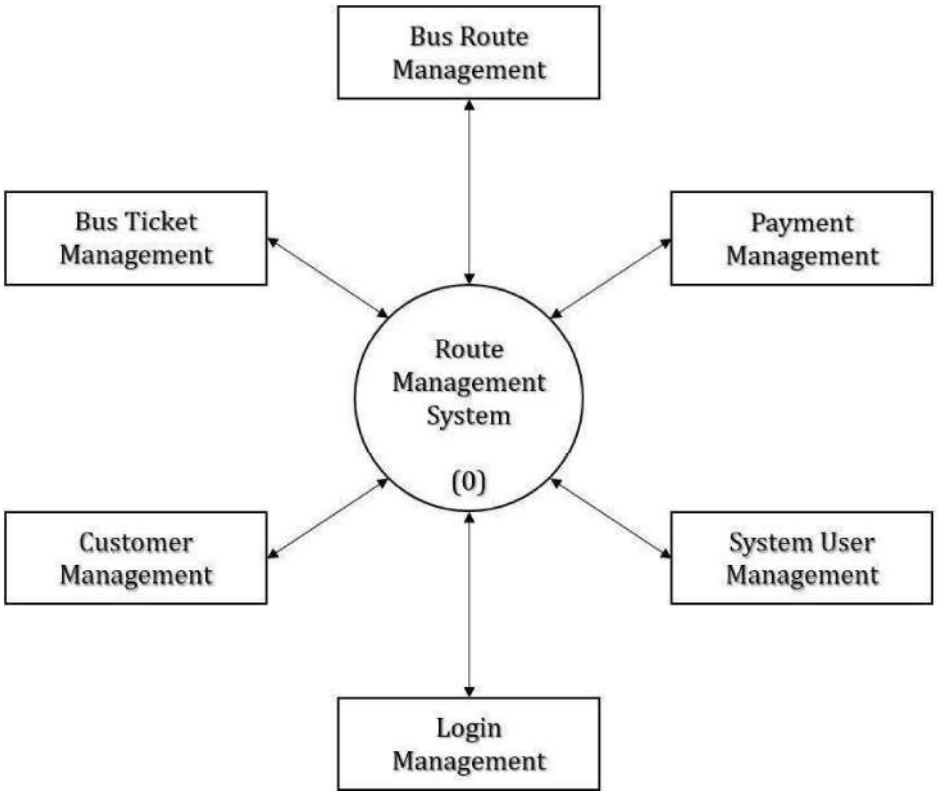
Input:customerdetails,ticketdetails
Output:updateddatabase

4.ReportModule–Inthismodule,thereportofallbookings,tickets, busschedulesandpaymentsisgenerated.

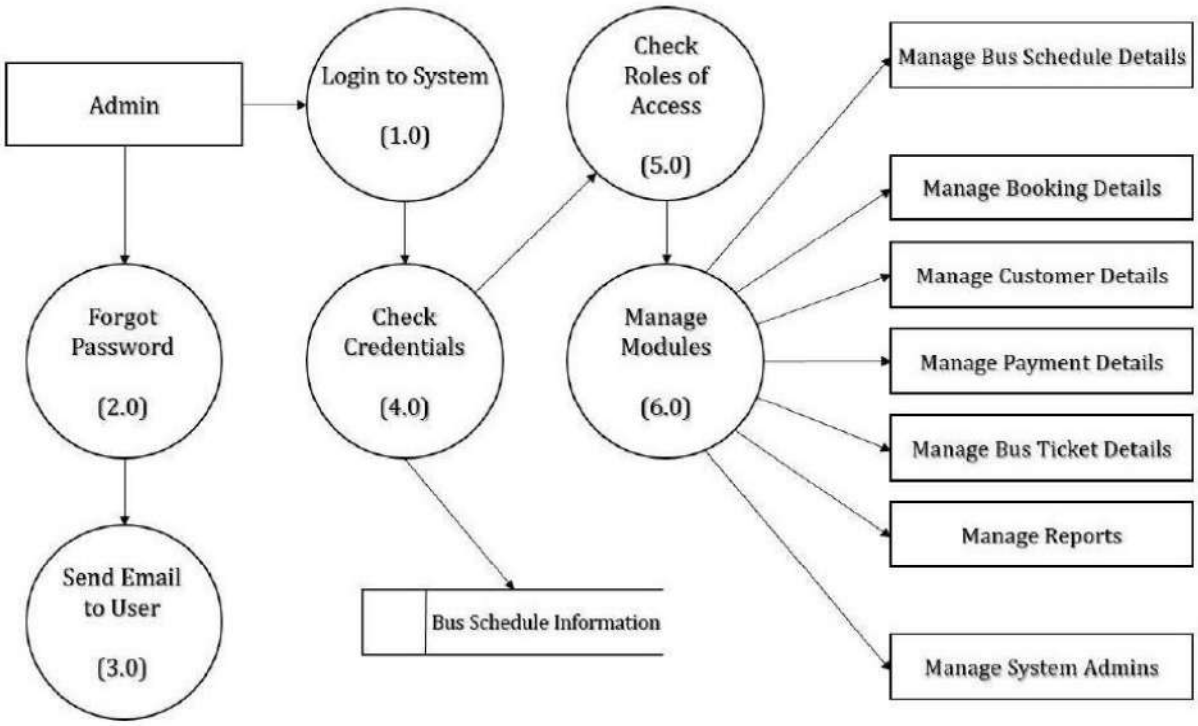
Input:bookingdetails,ticketdetails,busdetails,paymentdetails
Output:generatedreport

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero

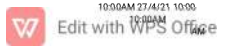


Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	
2	FeasibilityStudy	1	0:00AM	
3	RequirementAnalysisandSpecifications	3days	101:40/04/A	2M1
4	Design	5days	20/4/21	
5	CodingandUnitTesting	1	0:00AM	



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

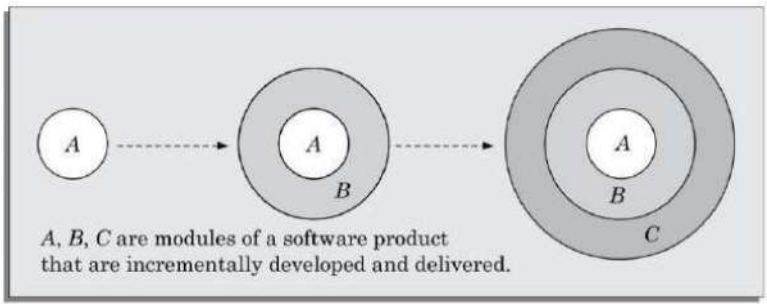
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthe systemaccordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

PATIENTAPPOINTMENTAND PRESCRIPTION MANAGEMENTSYSTEM

ProblemStatement–Patientmanagementsystemidentifiesinternaldatastores ofreports,doctors,diseases,teststhatmustbepresentinorderforthepatient systemtodoitsjob.Eachdatastoreshowstheflowofdataamongvariouspaths ofpatients,tests,doctors,reports,diseasesofthesystem.Patientrecords, medicineanddrugrecords,testrecordsareprocessedandreportsaregenerated.

ProcessModel–Incrementaldevelopmentmodel



Fig–incrementalmodelofsoftwaredevelopment

Thislifecyclemodelisalsoreferredtoassuccessiveversionsmodel.Inthislife cyclemodel,thesoftwareisfirstbrokendownintoseveralmoduleswhichcanbe incrementallyconstructedanddelivered.Thedevelopmentteamfirstdevelops thecoremodulesofthesystem.Thisinitialproductskeleton(A)isrefinedinto increasinglevelsofcapabilitybyaddingnewfunctionalities.Eachevolutionary versioncanbedevelopedbyiterativewaterfallmodel.Theusergetsachanceto experimentwiththepartiallydevelopedsoftwaremuchbeforethecomplete versionofthesystemisreleased.

SoftwareRequirementAnalysis

1. PatientModule–Inthismodule,apatientisregisteredandtheprofile ismaintained.

   Input:patientdetails
   Output:uniqueprofile

2. DoctorModule–Inthismodule,thedetailsofdoctorsaremaintained.

Input:patientdetails
Output:updateddatabase

3.DiagnosisModule–Inthismodule,thetestsareperformedandthe diseaseisdiagnosed.
Adatabaseismaintainedforkeepingthepatientfiles.

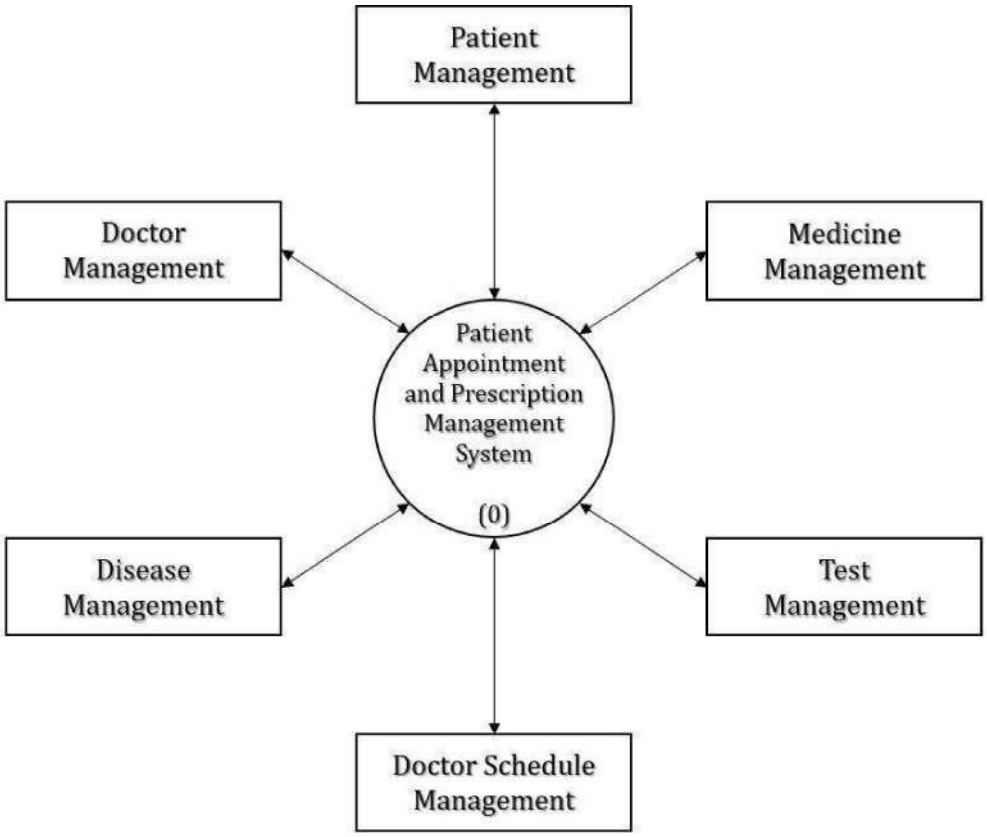
Input:patientdetails,doctordetails,testdetails
Output:diagnoseddisease,updateddatabase

4.ReportModule–Inthismodule,patientrecords,medicineanddrug records,testrecordsareprocessedandreportsaregenerated.

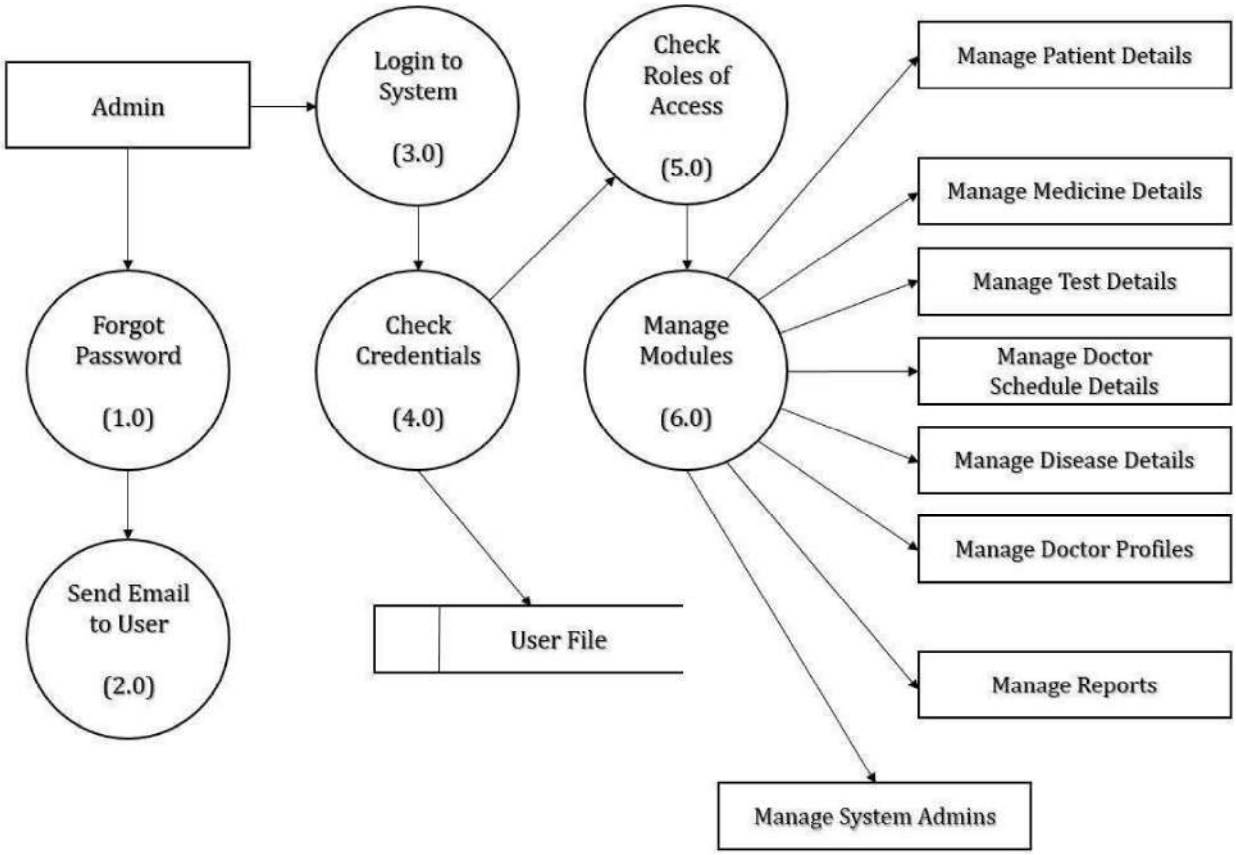
Input:patientdetails,doctordetails,diagnosisdetails
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero

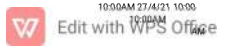


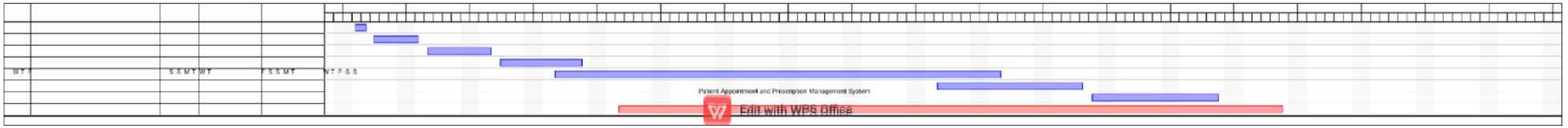
Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	
2	FeasibilityStudy	1	0:00AM	
3	RequirementAnalysisandSpecifications	3days	101:40/04/A	2M1
4	Design	5days	20/4/21	
5	CodingandUnitTesting	1	0:00AM	





RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

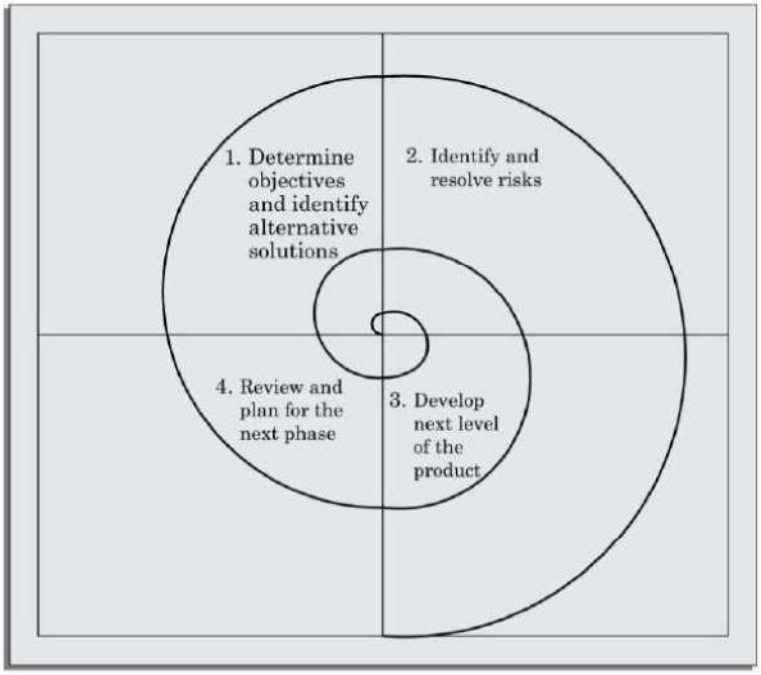
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

HOTELAUTOMATIONMANAGEMENTSYSTEM

ProblemStatement–Guestscanreservehotelroomsinadvanceoronthespot dependingontheavailabilityofrooms.Theoperatorwouldenterdatapertaining toguestssuchastheirarrivaltime,advancepay,approximatedurationofstay, purposeandtypeofroomrequiredi.e.AC,nonAC,deluxe,single/double occupancy.Thecomputershouldregisterthisdataanddependingonthe availabilityofasuitableroom,itshouldprovisionallyallotaroomnumbertothe guestandassignauniquetokennumber.Iftheguestcannotbeaccommodated, thecomputergeneratesanapologymessage.Thehotelcateringservice managerwouldinputthefooditemsandtheirquantityasandwhenconsumed bytheguestalongwiththetokennumberandthecorrespondingdateandtime. Itcanbeassumedthatdifferentfooditemshavedifferentcodesandhence cateringservicemanagerneednotenterthenameofthefooditems.Whenthe customerpreparestocheckout,thehotelautomationsoftwaregeneratesthebill alongwiththetaxes.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment
Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the

softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1. GuestModule–Inthismodule,aguestisregisteredandtheprofileis maintained.Ifasuitableroomisavailable,aroomnumberandunique tokennumberisalsoassigned.Elseanapologymessageisgenerated. Adatabaseismaintainedforkeepingtheinformation.

   Input:guestdetails
   Output:uniqueprofile,updateddatabase

2. CateringModule–Inthismodule,thedetailsoffooditemsconsumedby theguestismaintained.
   Adatabaseismaintainedforkeepingtheinformation.

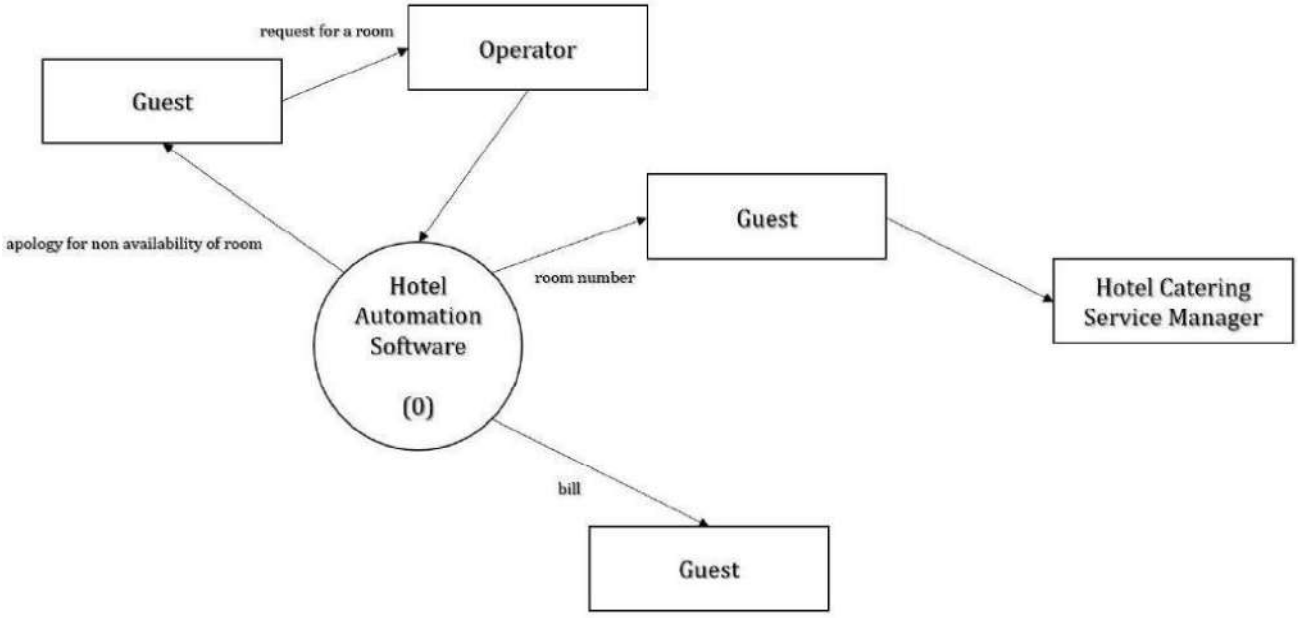
   Input:guestdetails,roomnumber,tokennumber
   Output:updateddatabase

3. BillModule–Inthismodule,thetotalbillisgeneratedwhenthe guestpreparestocheckout.

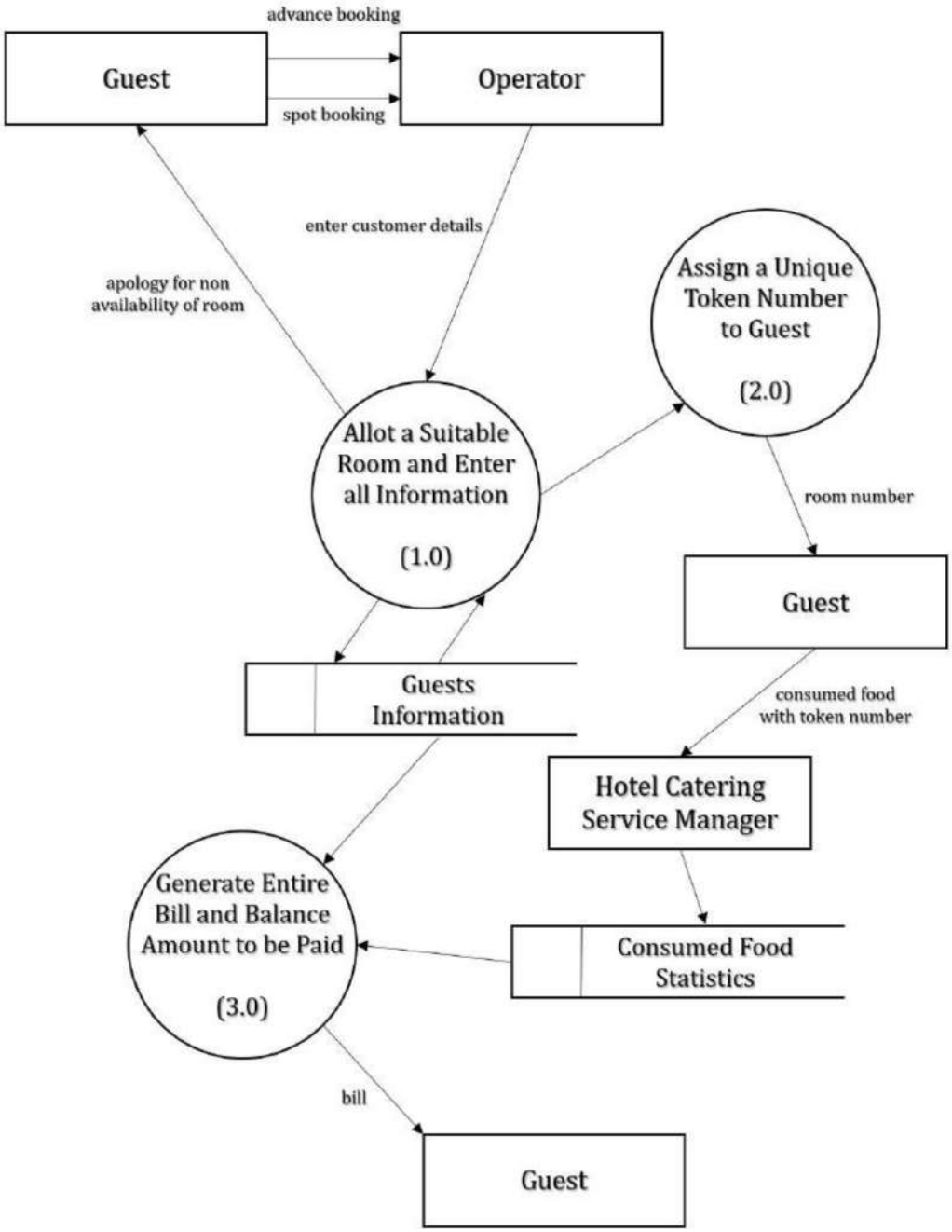
   Input:guestdetails,roomnumber,tokennumber,fooditemsconsumed
   Output:generatedbill

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



Fig–DFDlevelone ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	7/5/21 10:00AM
2	CustomerCommunication	1	0:00AM	23/7/ 21 22/6/21 10:00 10:00AM
3	Planning	3days	101:40	0/4/A2M1
4	RiskAnalysis	5days	20/4/21	12days 13/4/21 10:00AM
5	Engineering	1	0:00AM	15/6/21 3:00PM

11Apr21 18Apr21 25Apr21 2May21 9May21 16May21 23May21 30May21 6Jun21 13Jun21 20Jun21 27Jun21 4Jul21 1

1/7/2 10:00AM

FSSFSSMT MT WT WTF SSMT WTF SSMT FSS FSSMT WT MTW T WTF SSM T WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS WT

10:00 AM

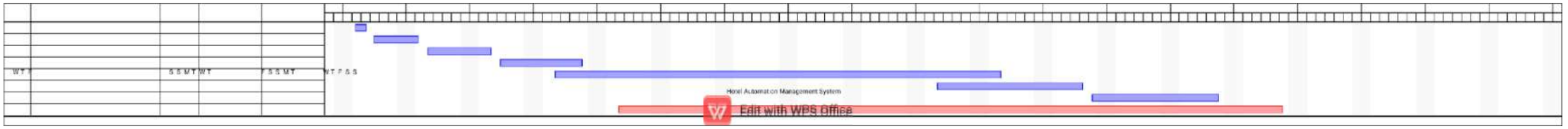
10days 2/7/21 10:00AM

19/4/21 10:00AM

53days 11/5/21 10:00AM

1_6_/7/2_1

27/4/21 10:00 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

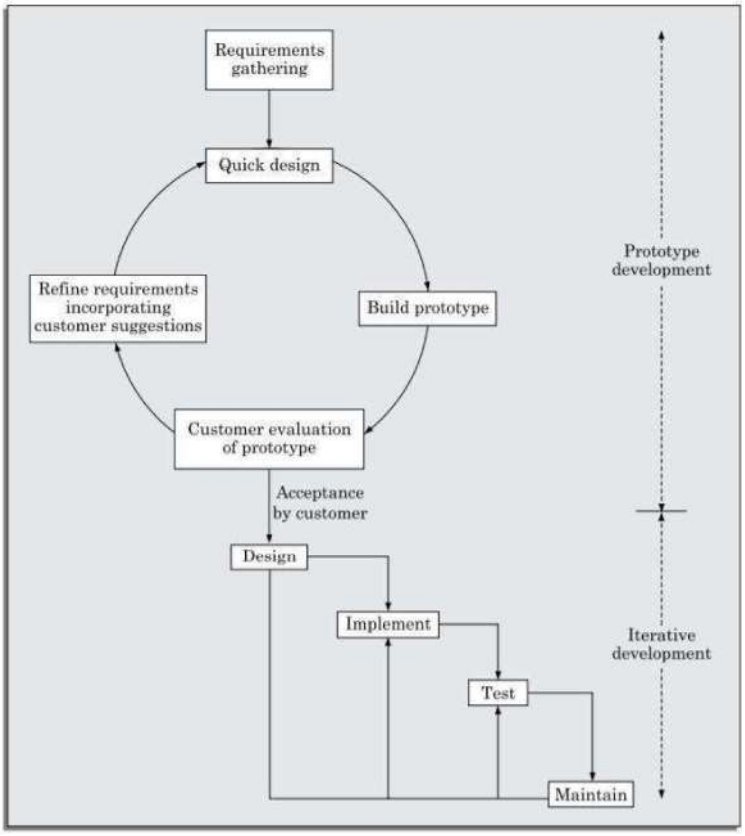
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

CRIMINALRECORDMANAGEMENT SYSTEM

ProblemStatement–Crimefilemanagementsystemidentifiesinternaldata storesofcourt,criminals,prisoners,chargesheet,FIRthatmustbepresentin orderforthecrimesystemtodoitsjob.Eachshowstheflowofdataamongthe variouspathsofcrime,FIR,criminals,courtofthesystem.Acriminalrecord managementsystemisimplementedforjailers,policeofficersandCBIofficers.

ProcessModel–Prototypingmodel



Fig–prototypingmodelofsoftwaredevelopment
Thismodelallowsallorpartofasystemtobeconstructedquickly.
Goalofprototypingistoreducethechanceofuncertainty.Thephases include–

1)meetingofthedevelopersandthecustomerstodefinetheoverall objectivesofthesoftware
2)quickdesignleadstoconstructionofprototype
3)evolutionofprototypebycustomer
4)iterationoccursastheprototypeisturnedtotheneedsofthecustomers

thecodeforprototypeisthrownaway–theexperiencegatheredfrom

developingtheprototypehelpsindevelopingtheactualsystem

SoftwareRequirementAnalysis

1. AdminModule – Inthismodule,anadminisregisteredandtheprofile ismaintained.Thelevelofaccessisgrantedaccordingtothe designation.

   Input:admindetails
   Output:uniqueprofile,grantedlevelofaccess

2. ComplainModule – Inthismodule,thecomplaintisregisteredandFIR isfiled. Adatabaseismaintainedforkeepingthecomplaintdetails.

   Input:complaintdetails
   Output:filedFIR,updateddatabase

3. CrimeModule – Inthismodule,thedetailsofthecrimearemaintained andchargesheetisfiled.

   Input:complaintdetails,FIRdetails
   Output:filedchargesheet

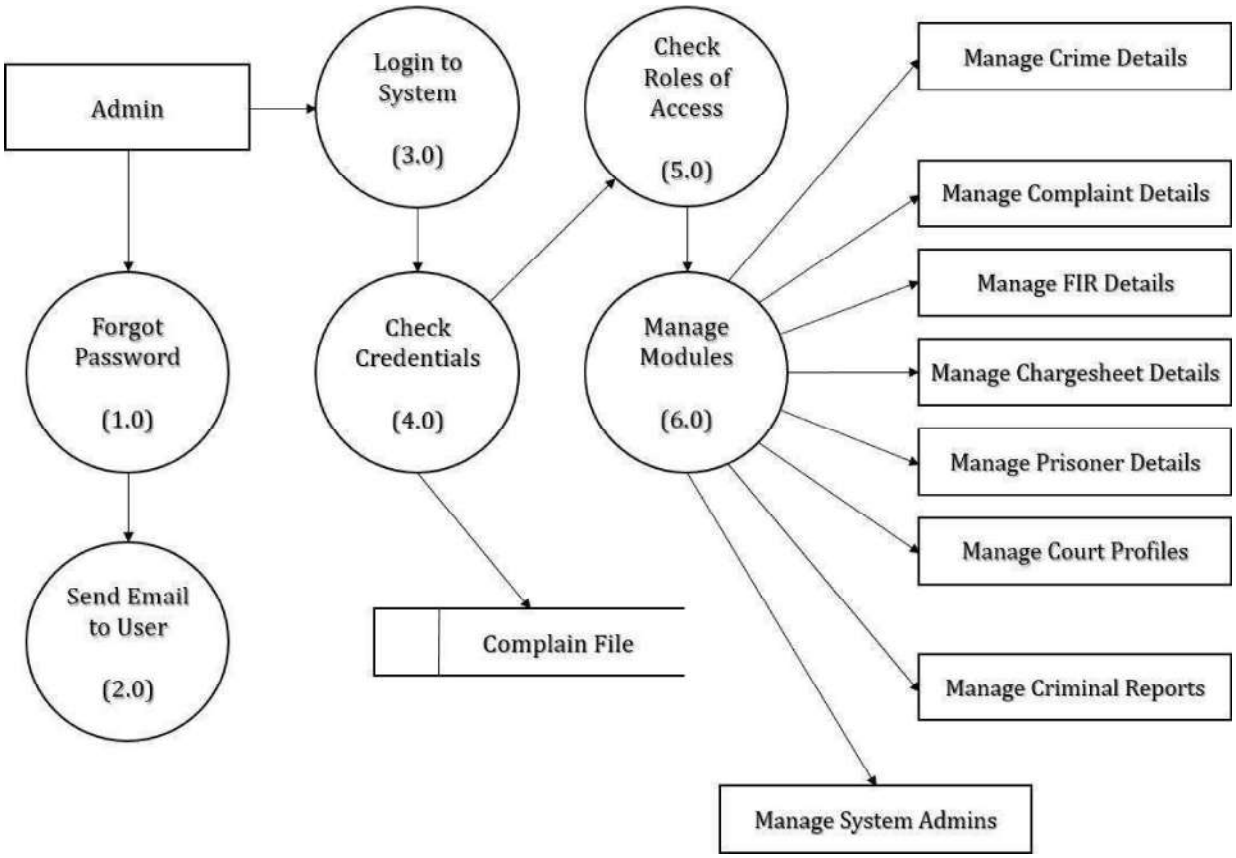
4. CourtModule – Inthismodule,thecaseistakentothecourtandlegal actionistakenaccordingly.

   Input:complaintdetails,FIRdetails,chargesheetdetails
   Output:legalaction,updateddatabase

DesignEngineering

DataFlowDiagrams(DFDs)

Fig–DFDlevelzero

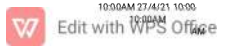


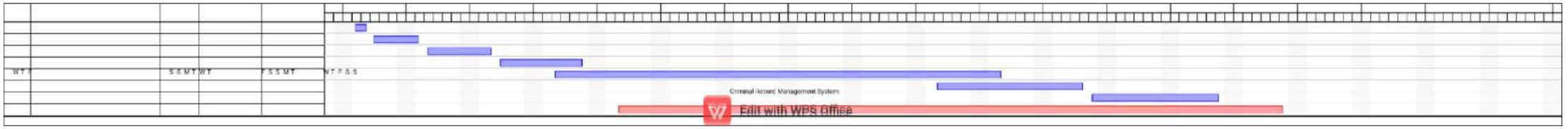
Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	
2	FeasibilityStudy	1	0:00AM	
3	RequirementAnalysisandSpecifications	3days	101:40/04/A	2M1
4	Design	5days	20/4/21	
5	CodingandUnitTesting	1	0:00AM	





RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

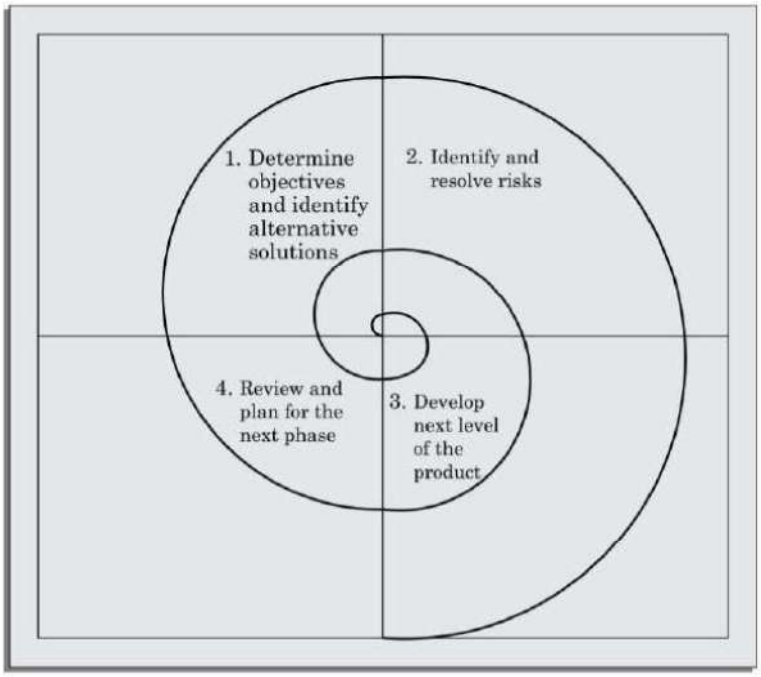
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

EXAMINATIONANDRESULTCOMPUTATIONSYSTEM

ProblemStatement–Examinationmanagementsystemidentifiesinternaldata storesofsubject,course,branch,student,classthatmustbepresentinorderfor theexaminationsystemtodoitsjob.Eachshowstheflowofdataamongthe variouspartsoftimetable,class,course,subject,branchofthesystem.Themain functionalitiesoftheexaminationareprocessingtimetablerecords,examination records,classrecords,studentrecords,branchrecords,courserecords,subject recordsandgeneratereportofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectofwaterfall model.Itisderivedintoframeworkofactivitiesalsocalledtaskregions.

Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1. StudentModule – In this module, a student is registered and the profile is maintained. A database is maintained for keeping the student information.

   Input: student details
   Output: unique profile

2. TimetableModule – In this module, the details of branch, course, class and subject is maintained and the timetable is created. A database is maintained for keeping the timetable information.

   Input: branch details, course details, class details, subject details
   Output: updated database, created timetable

3. ExaminationModule – In this module, the examination takes place and the results are computed.

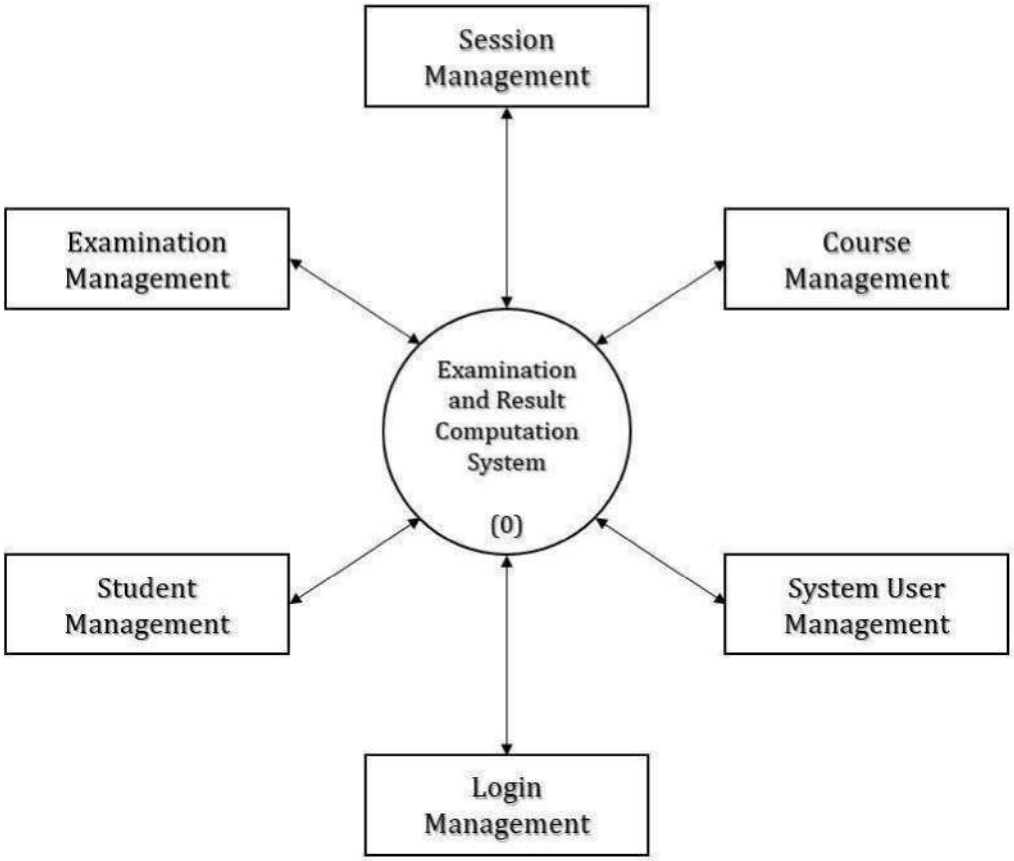
   Input: student details, timetable details
   Output: computed results, updated database

4. ReportModule – In this module, timetable, examination, class, student, branch, course and subject records are processed and reports are generated.

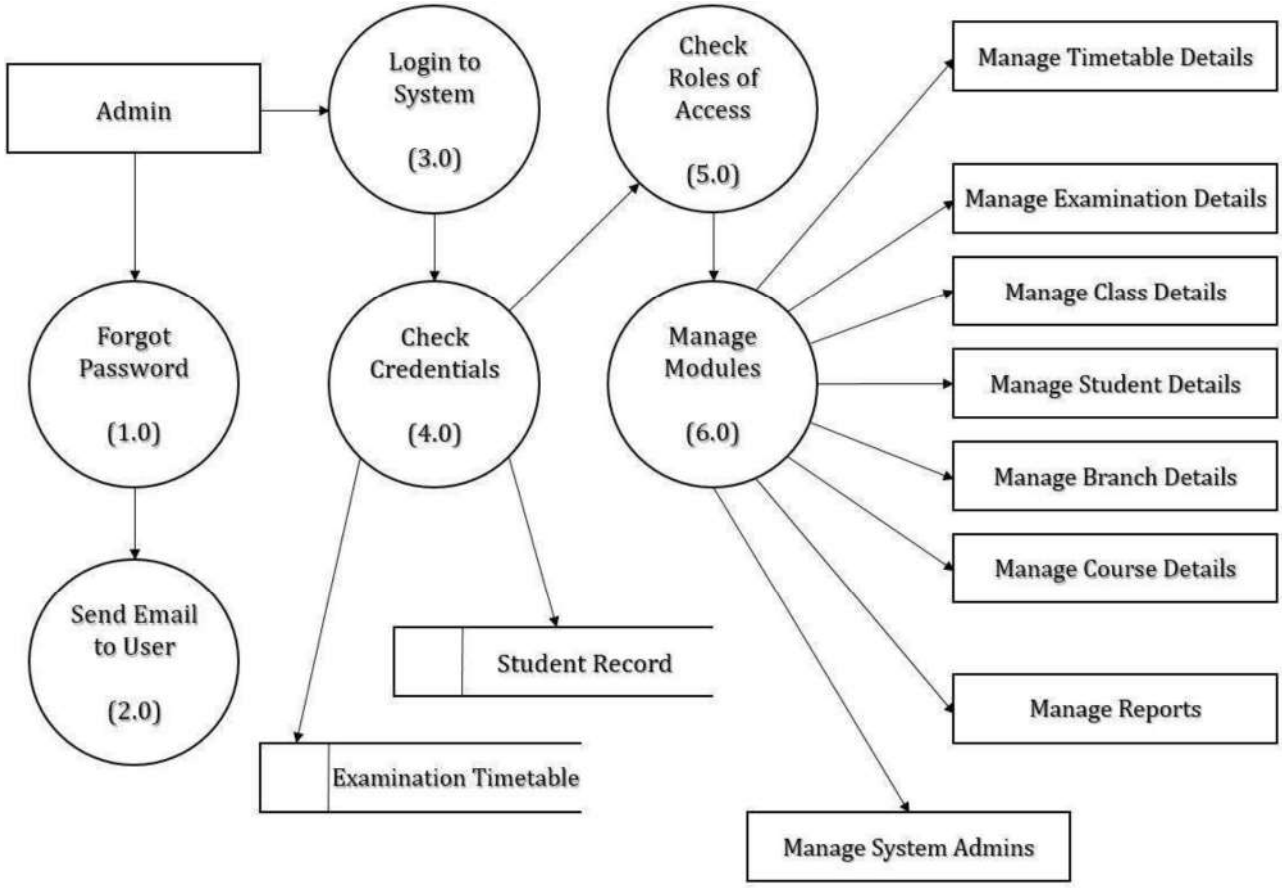
   Input: guest details, room number, token number, food items consumed
   Output: generated reports

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	7/5/21 10:00AM
2	CustomerCommunication	1	0:00AM	23/7/ 21 22/6/21 10:00 10:00AM
3	Planning	3days	101:40	0/4/A2M1
4	RiskAnalysis	5days	20/4/21	12days 13/4/21 10:00AM
5	Engineering	1	0:00AM	15/6/21 3:00PM

11Apr21 18Apr21 25Apr21 2May21 9May21 16May21 23May21 30May21 6Jun21 13Jun21 20Jun21 27Jun21 4Jul21 1

1/7/2 10:00AM

FSSFSSMT MT WT WTF SSMT WTF SSMT FSS FSSMT WT MTW T WTF SSM T WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS WT

10:00 AM

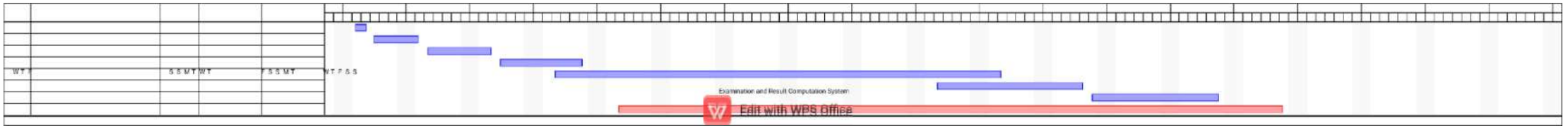
10days 2/7/21 10:00AM

19/4/21 10:00AM

53days 11/5/21 10:00AM

1_6_/7/2_1

27/4/21 10:00 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

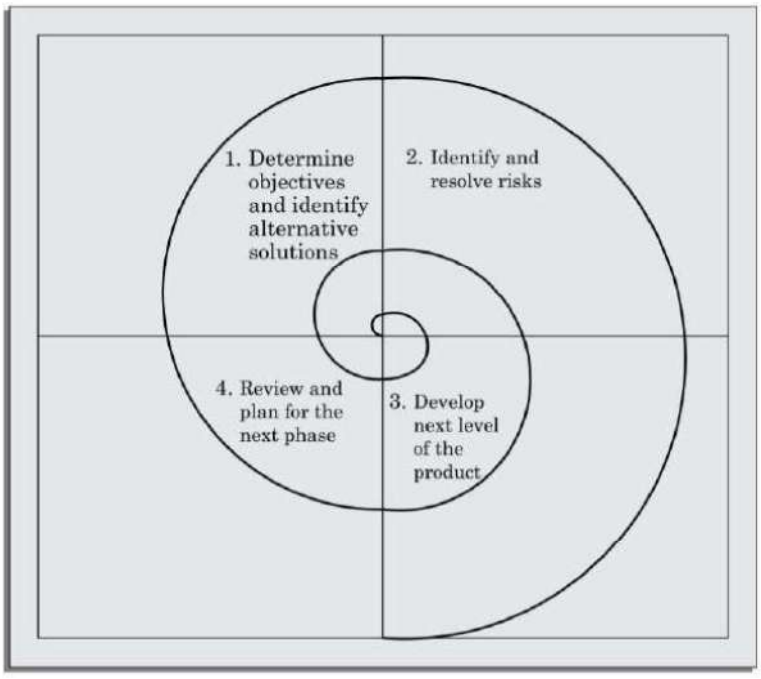
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

PARKINGALLOCATION SYATEM

ProblemStatement–Parkingmanagementsystemidentifiesinternaldatastores oflogin,customers,durationandparkingfeesthatmustbepresentinorderfor theparkingsystemtodoitsjob.Itshowstheflowofdatabetweenthevarious pathsofparkingslotsandparkingfees.Thefunctionalitiesofparkingsystemare processingofparkingslots,vehiclerecords,parkingfees,durationandgenerating thereportofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredandthe profileismaintained.

Input:customerdetails Output:uniqueprofile

2.VehicleModule–Inthismodule,thecustomer'svehicleisregistered.

Input:customerdetails,vehicledetails
Output:updateddatabase

3.ParkingModule–Inthismodule,parkingslotisallottedandfeeis chargedaccordingly.
Adatabaseismaintainedforkeepingtheparkinginformation.

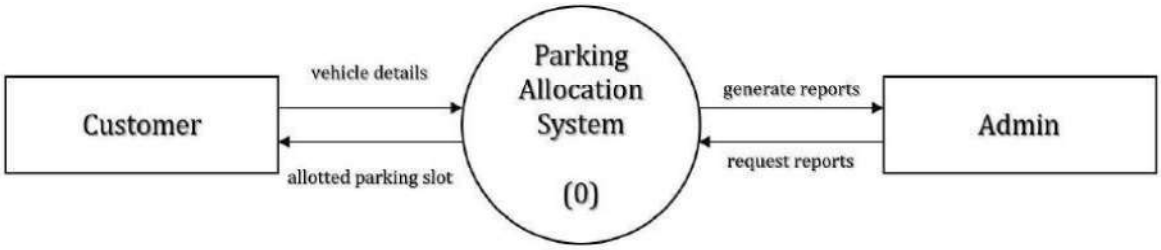
Input:customerdetails,vehicledetails,duration
Output:updateddatabase

4.ReportModule–Inthismodule,parkingslot,vehicle,parkingfee anddurationrecordsareprocessedandreportsaregenerated.

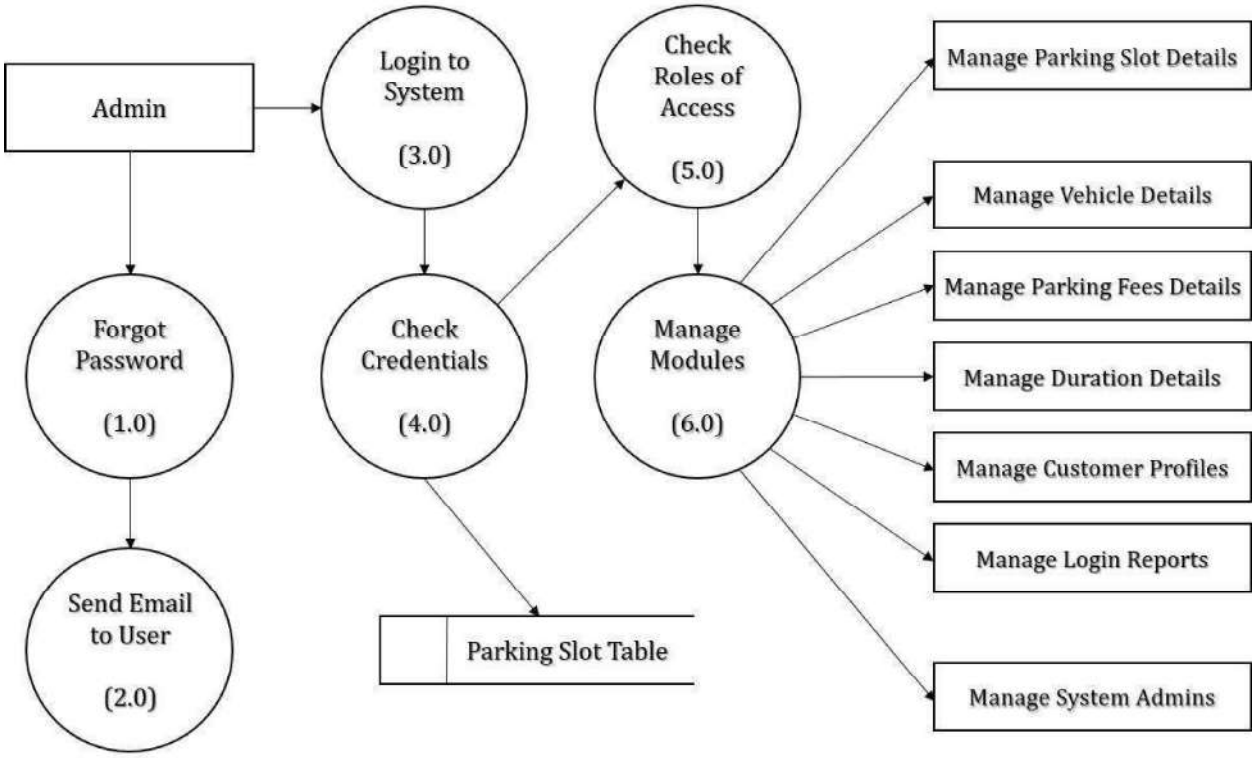
Input:parkingslotdetails,vehicledetails,parkingfeedetails,durationdetails
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	7/5/21 10:00AM
2	CustomerCommunication	1	0:00AM	23/7/ 21 22/6/21 10:00 10:00AM
3	Planning	3days	101:40	0/4/A2M1
4	RiskAnalysis	5days	20/4/21	12days 13/4/21 10:00AM
5	Engineering	1	0:00AM	15/6/21 3:00PM

11Apr21 18Apr21 25Apr21 2May21 9May21 16May21 23May21 30May21 6Jun21 13Jun21 20Jun21 27Jun21 4Jul21 1

1/7/2 10:00AM

FSSFSSMT MT WT WTF SSMT WTF SSMT FSS FSSMT WT MTW T WTF SSM T WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS WT

10:00 AM

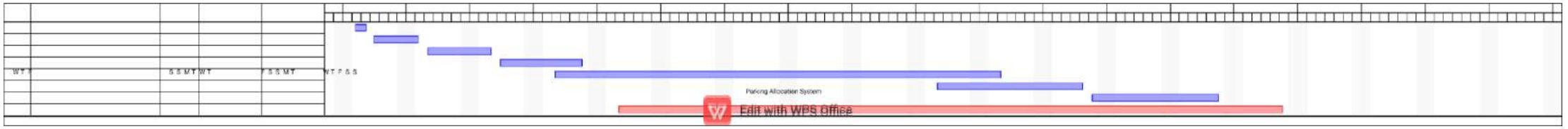
10days 2/7/21 10:00AM

19/4/21 10:00AM

53days 11/5/21 10:00AM

1_6_/7/2_1

27/4/21 10:00 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

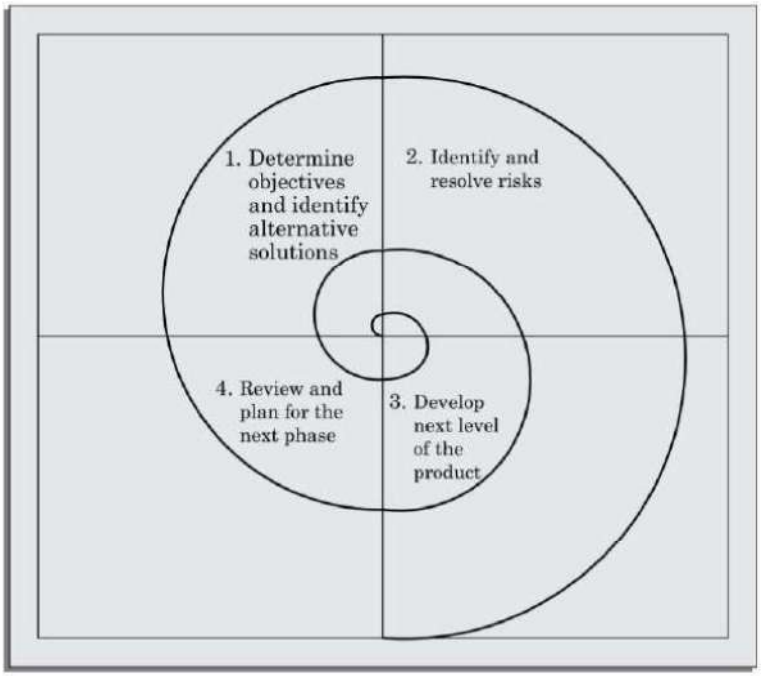
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

WHOLESALEMANAGEMENTSYSTEM

ProblemStatement–Itprovidesallthefunctionalitiesoftheretailstore managementsystemasawhole.Itidentifiesinternaldatastoresofcustomer, billingcounter,bill,retailprice,productthatmustbepresentinorderfortheretail storesystemtodoitsjob.Storerecords,productrecords,retailpricerecords,bill records,billingcounterrecordsandcustomerrecordsareprocessedandreports aregeneratedofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1. CustomerModule – In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. StockModule – In this module, the details of stocks are maintained. A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. BillModule – In this module, all the bills are maintained. A database is maintained for keeping the bill information.

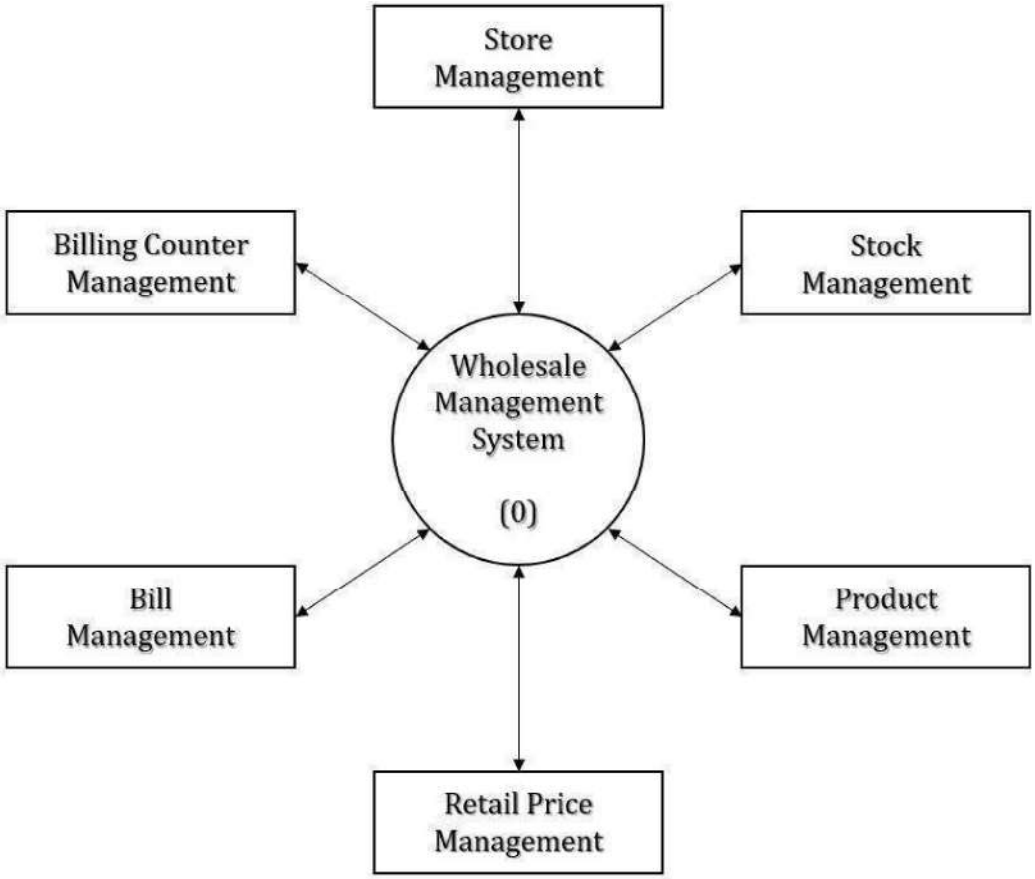
   Input: customer details, stock details
   Output: generated bills, updated database

4. ReportModule – In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

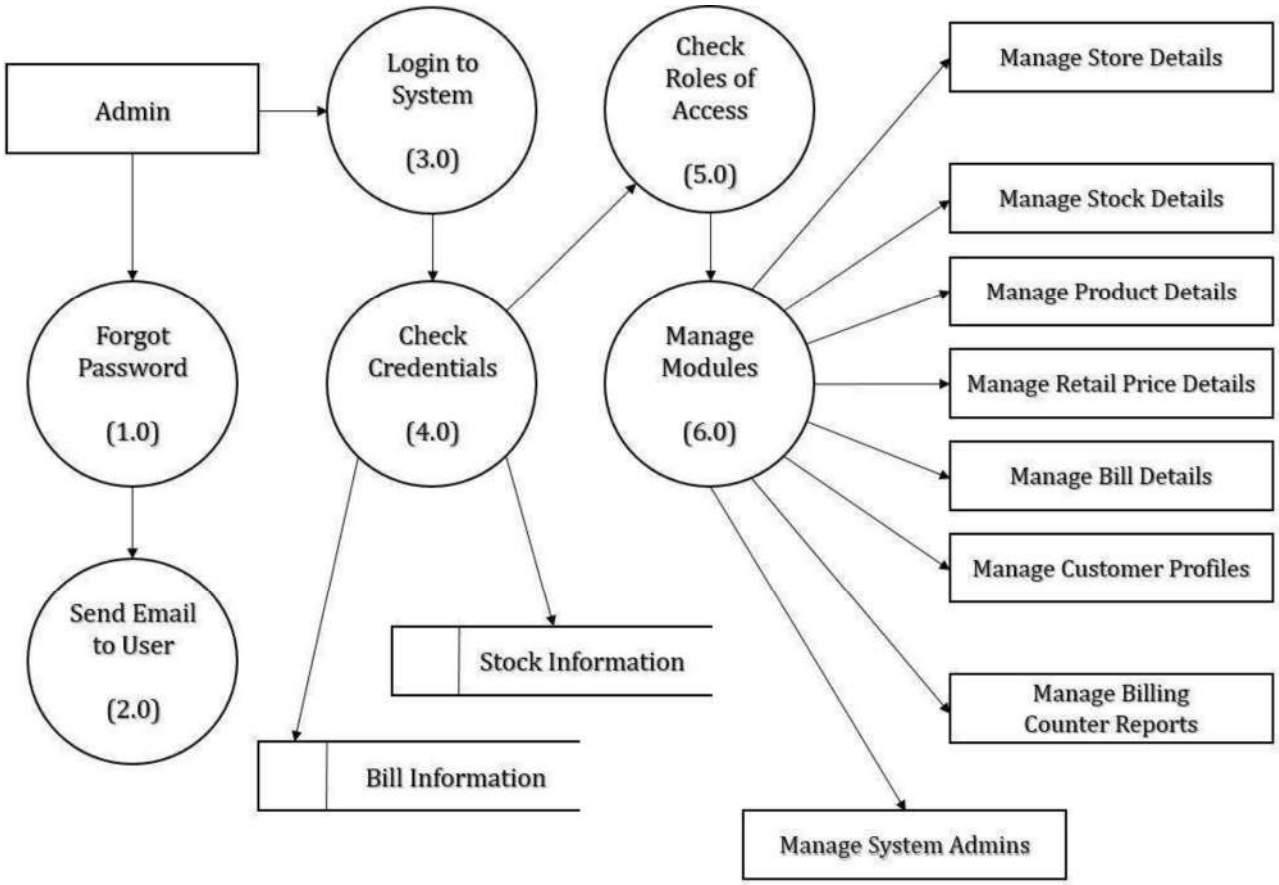
   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

DesignEngineering

DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.

	Name	Duration	Start	Finish
1	ProblemStatement	1.5days	12/4/21	7/5/21 10:00AM
2	CustomerCommunication	1	0:00AM	23/7/ 21 22/6/21 10:00 10:00AM
3	Planning	3days	101:40	0/4/A2M1
4	RiskAnalysis	5days	20/4/21	12days 13/4/21 10:00AM
5	Engineering	1	0:00AM	15/6/21 3:00PM

11Apr21 18Apr21 25Apr21 2May21 9May21 16May21 23May21 30May21 6Jun21 13Jun21 20Jun21 27Jun21 4Jul21 1

1/7/2 10:00AM

FSSFSSMT MT WT WTF SSMT WTF SSMT FSS FSSMT WT MTW T WTF SSM T WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS WT

10:00 AM

10days 2/7/21 10:00AM

19/4/21 10:00AM

53days 11/5/21 10:00AM

1_6_/7/2_1

27/4/21 10:00 10:00AM

RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s
wherep–prioritywithwhichtheriskmustbehandled
r–theprobabilityoftheriskbecomingtrue
s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthe systemaccordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

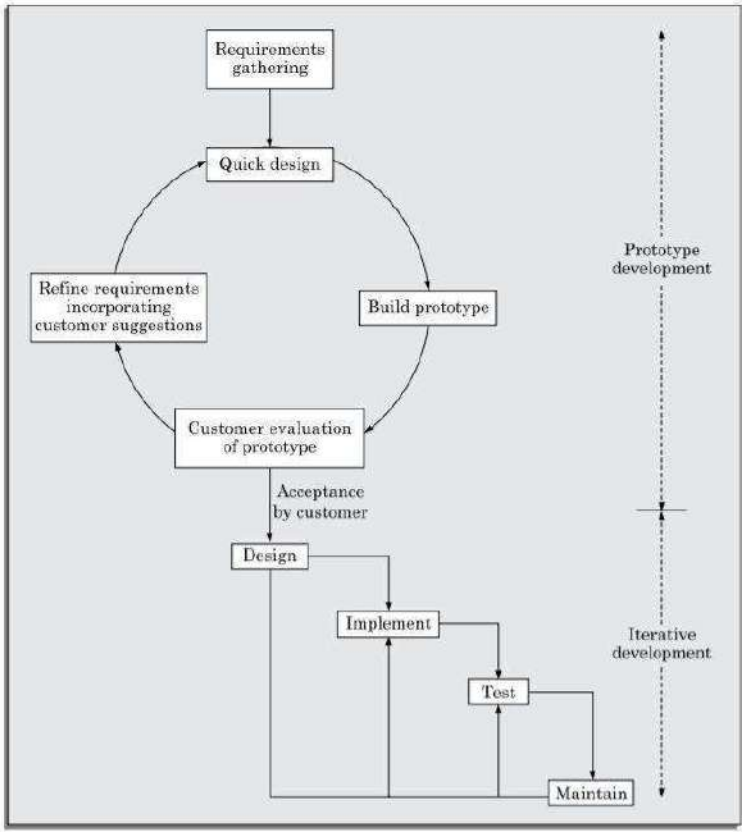
RETAIL SHOPPING MANAGEMENT SOFTWARE

Name: SAYAN NATH
College Roll No.: 1 732
Registration No.: A01-1112-117-027-2018
Examination Roll No.: 2021141312
Paper Code: CMSADSE3
Semester: V
Department: Computer Science
Supervisor: Prof. Manas Pal

RETAILSHOPPINGMANAGEMENTSOFTWARE

ProblemStatement–Asupermarketneedstodevelopthefollowingsoftwareto encourageregularcustomers.Forthis,thecustomerneedstosupplyhis/her residentialaddress,mobilenumberanddrivinglicense.Eachcustomerwho registersforthisschemeisassignedauniquecustomernumber(CN)bythe computer.AcustomercanpresentCNtothecheckoutstaffwhenanypurchase ismade.Inthiscase,thevalueofthispurchaseiscreditedagainsttheCN.Atthe endofeachyear,thesupermarketintendstoawardsurprisegiftstoten customerswhomakethehighesttotalpurchaseovertheyear.Alsoitintendsto awarda22karatgoldcointoeverycustomerwhosepurchaseexceedsINR 10000.TheentriesagainsttheCNareresetonthelastdayofeveryyearafterthe prizewinners'listisgenerated.

ProcessModel–Prototypingmodel



Fig–prototypingmodelofsoftwaredevelopment
Thismodelallowsallorpartofasystemtobeconstructedquickly.
Goalofprototypingistoreducethechanceofuncertainty.Thephases include–

1)meetingofthedevelopersandthecustomerstodefinetheoverall objectivesofthesoftware
2)quickdesignleadstoconstructionofprototype
3)evolutionofprototypebycustomer
4)iterationoccursastheprototypeisturnedtotheneedsofthecustomers
5)thecodeforprototypeisthrownaway–theexperiencegathered fromdevelopingtheprototypehelpsindevelopingtheactual system

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredand assignedauniquecustomernumber(CN).
Adatabaseismaintainedforkeepingthecustomerdetails.

Input:customerdetails
Output:uniqueCN

2.PurchaseModule–Inthismodule,anypurchasemadebythe customeriscreditedagainsttheCN.
Adatabaseismaintainedforkeepingthepurchasehistory.

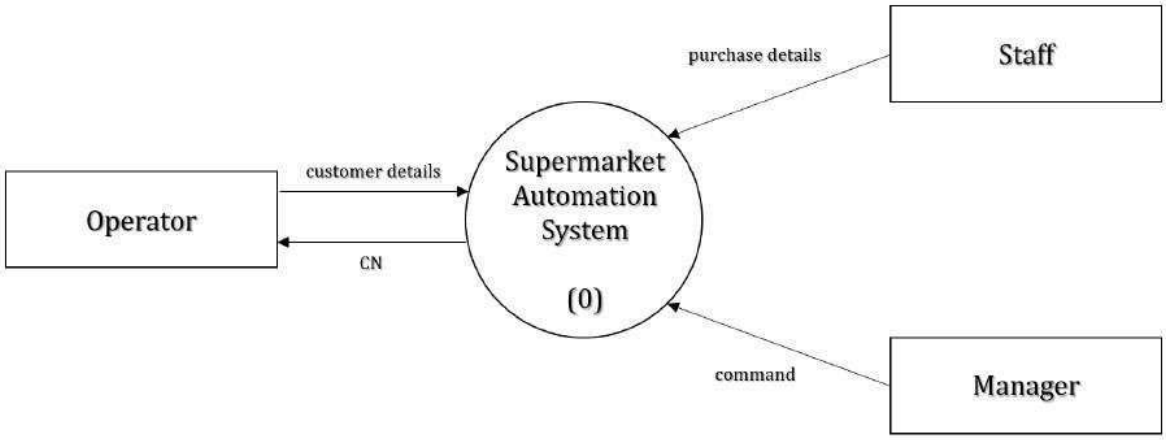
Input:purchasemadebythecustomer,CN
Output:updateddatabase

3.PromotionalModule–Inthismodule,customersareselectedforgiving outtheprizesandtheprizewinners'listisgenerated.
TheentriesagainsttheCNareresetafterthegenerationoftheprize winners' list.

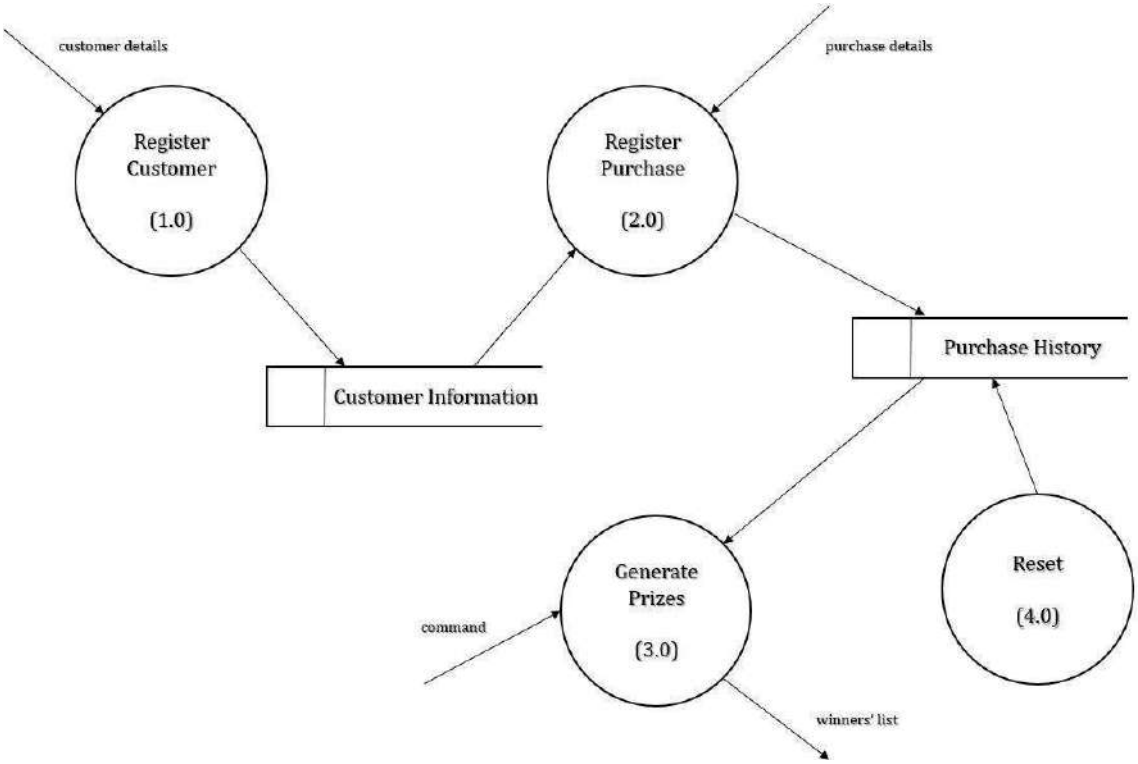
Input:purchasehistory,CN
Output:prizewinners'list

DesignEngineering

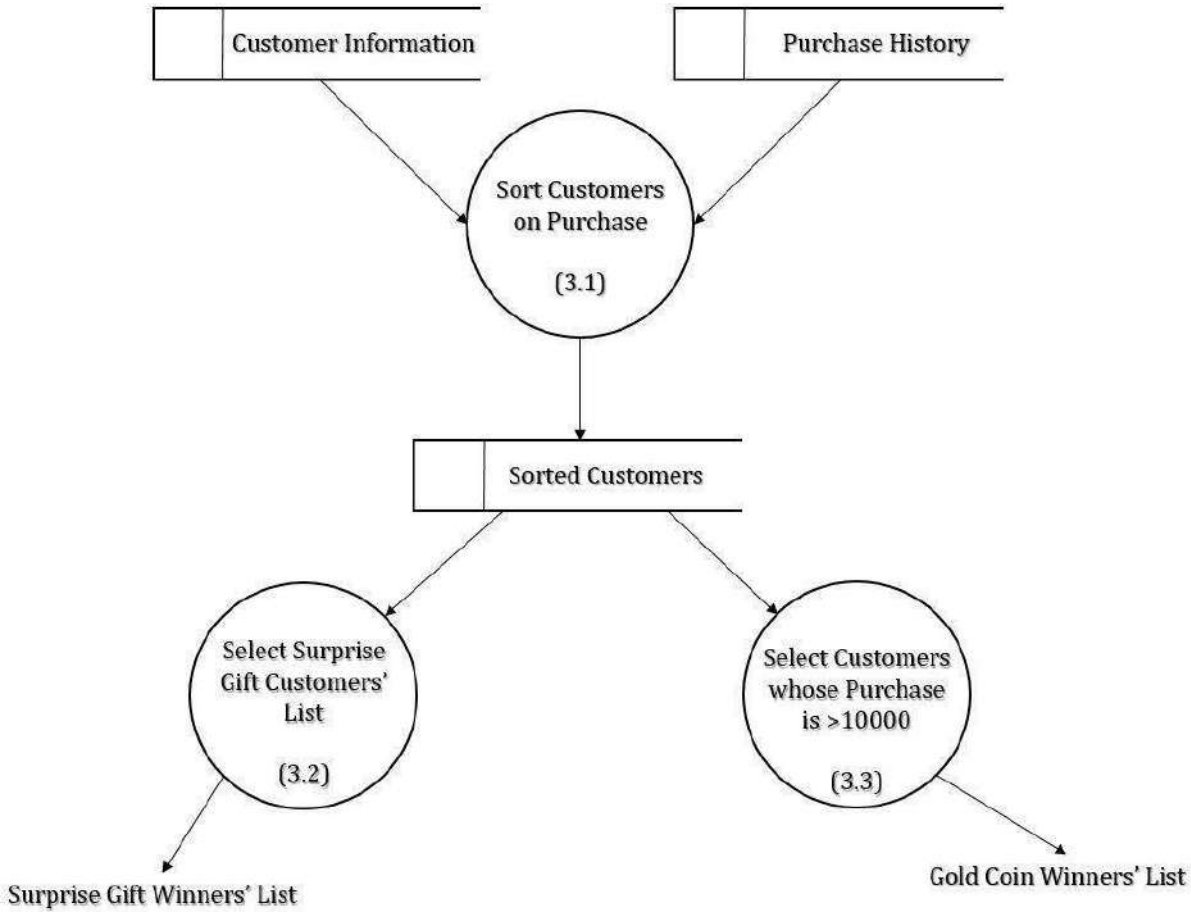
DataFlowDiagrams(DFDs)



Fig–DFDlevelzero



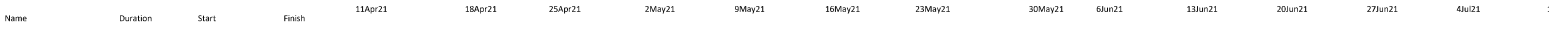
Fig–DFDlevelone

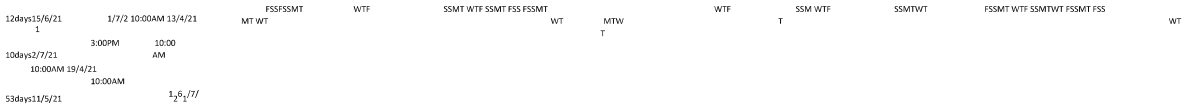


Fig–DFDleveltwoofProcess3.0

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21

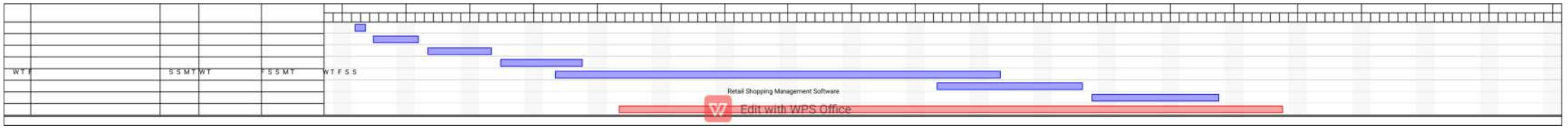

FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
MT WT WT MTW WT
T
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21 1 6 /7/
2 1

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM


1 ProblemStatement 1.5days12/4/21
2 FeasibilityStudy 10:00AM
3 RequirementAnalysisandSpecifications 3days101:40/04/A2M1
18Jul21
25Jul21
1Aug21
8Aug21
15Aug21
22
WTF SSM FSSMT TWT
WTF SSM WTF
T
SSMTW FSSMT T

4	Design	5days	20/4/21 10:00AM
5	CodingandUnitTesting	7days	28/4/21 10:00AM
6	IntegrationandSystemTesting	35days	4/5/21 10:00AM
7	Maintenance		
8	DocumentationandUserManual		



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtestingphase, themodulesareintegratedinaplannedmanner.Systemtestingconsistsofthree differentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

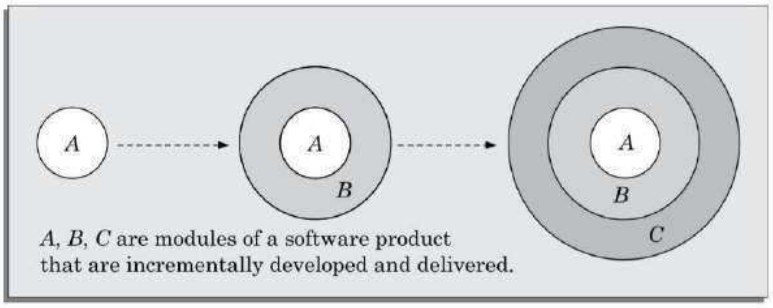
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

ROUTEMANAGEMENTSYSTEM(DTCROUTEINFORMATION)

Problemstatement–Itisanonlineinformationaboutthebusroutesandtheir frequenciesandfares.Itshowsallthefunctionalitiesofthebusbookingand ticketingsystem.Italsoidentifiesinternaldatastoresofpayments,busroutes, customers,tickets,ticketcountersthatmustbepresentinorderforthebus ticketsystemtodoitsjobandshowstheflowofdataamongvariouspathsof busschedule,ticketcounter,busroute,paymentandcustomerofthesystem. Finallyafterprocessing,itgeneratesreportofallbookings,alltickets,bus schedulesandallpayments.

Processmodel–Incrementalmodel



Fig–incrementalmodelofsoftwaredevelopment

Thislifecyclemodelisalsoreferredtoassuccessiveversionsmodel.Inthislife cyclemodel,thesoftwareisfirstbrokendownintoseveralmoduleswhichcanbe incrementallyconstructedanddelivered.Thedevelopmentteamfirstdevelops thecoremodulesofthesystem.Thisinitialproductskeleton(A)isrefinedinto increasinglevelsofcapabilitybyaddingnewfunctionalities.Eachevolutionary versioncanbedevelopedbyiterativewaterfallmodel.Theusergetsachanceto experimentwiththepartiallydevelopedsoftwaremuchbeforethecomplete versionofthesystemisreleased.

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredandthe profileismaintained.

Input:customerdetails
Output:uniqueprofile

2.BusTicketingModule–Inthismodule,anybookingmadebythe customerisregisteredandtheticketisgenerated.

Input:customerdetails
Output:generatedticket

3.InformationModule–Inthismodule,thebusdetailsofthebookingmade bythecustomerismaintained.
Adatabaseismaintainedforkeepingtheinformation.

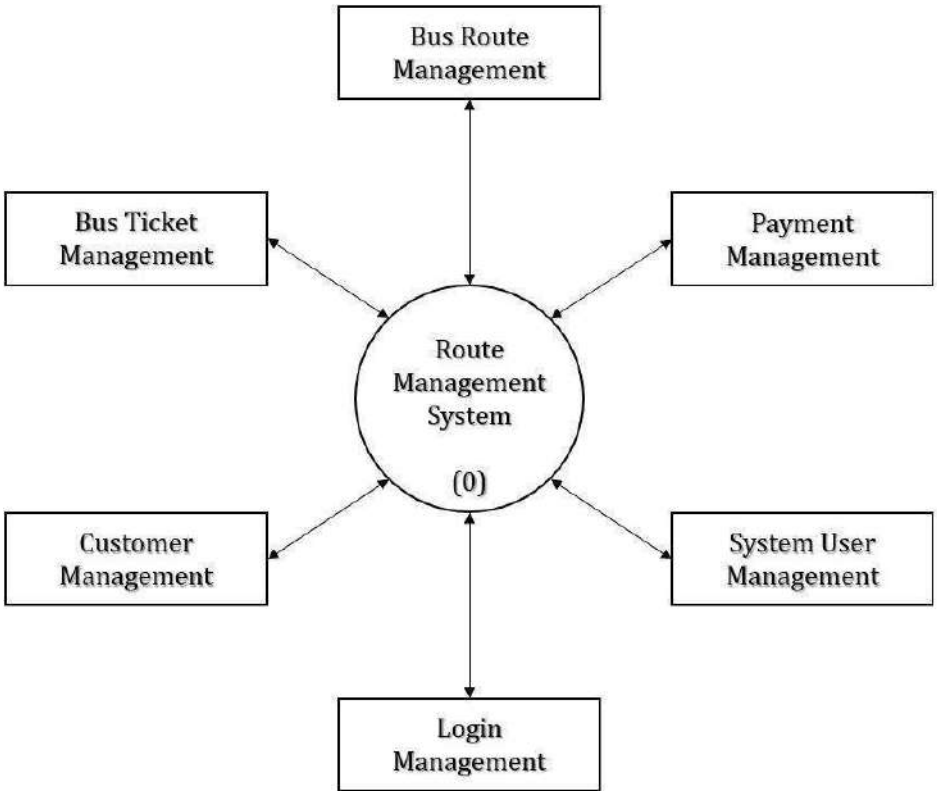
Input:customerdetails,ticketdetails
Output:updateddatabase

4.ReportModule–Inthismodule,thereportofallbookings,tickets, busschedulesandpaymentsisgenerated.

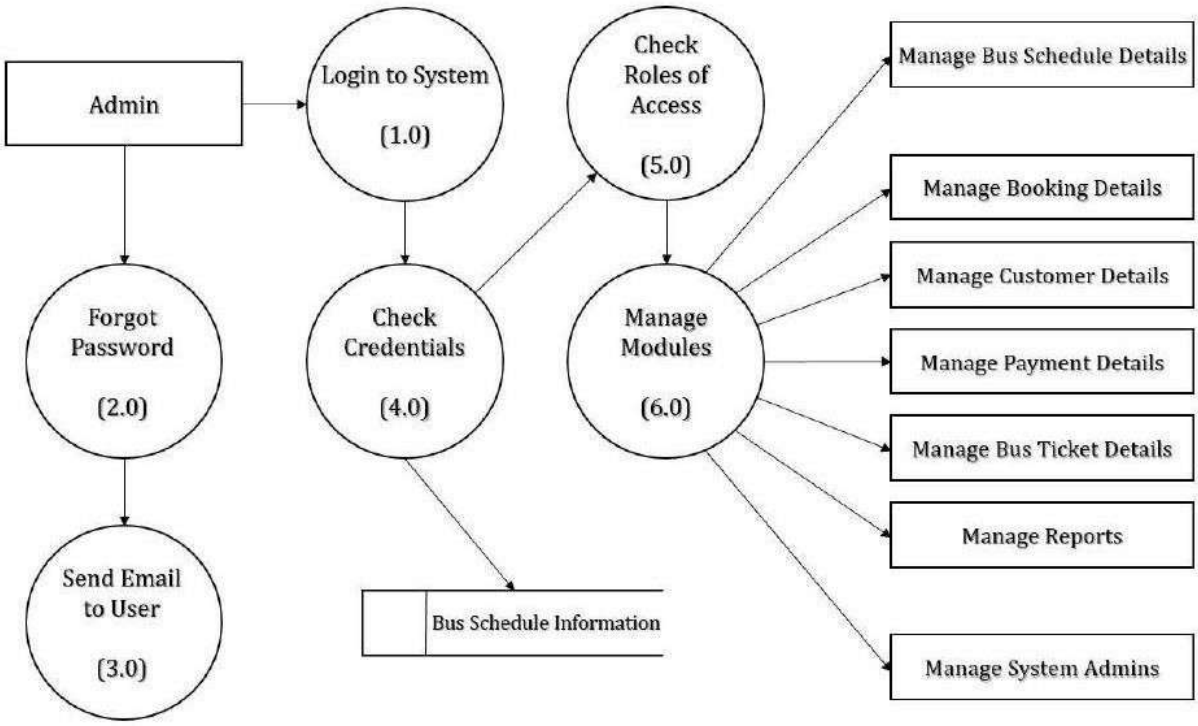
Input:bookingdetails,ticketdetails,busdetails,paymentdetails
Output:generatedreport

DesignEngineering

DataFlowDiagrams(DFDs)



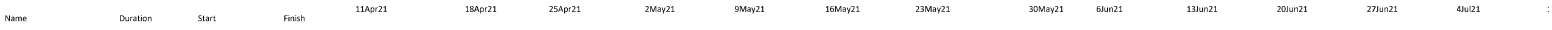
Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21

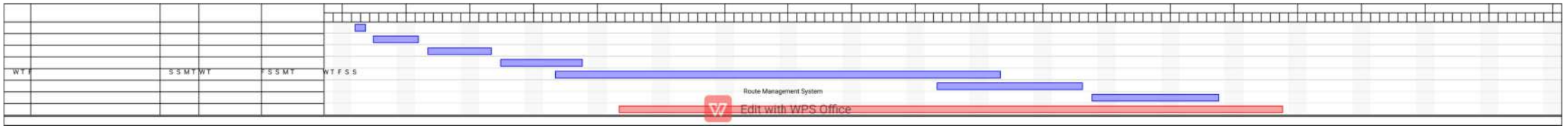

FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
MT WT WT MTW WT
T
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21 1 6 2 1 /7/

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM


1 ProblemStatement 1.5days12/4/21
2 FeasibilityStudy 10:00AM
3 RequirementAnalysisandSpecifications 3days101:40/04/A2M1
18Jul21
25Jul21
1Aug21
8Aug21
15Aug21
22
WTF SSM FSSMT TWT
WTF SSM WTF
T
SSMTW FSSMT T

4	Design	5days	20/4/21 10:00AM
5	CodingandUnitTesting	7days	28/4/21 10:00AM
6	IntegrationandSystemTesting	35days	4/5/21 10:00AM
7	Maintenance		
8	DocumentationandUserManual		



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

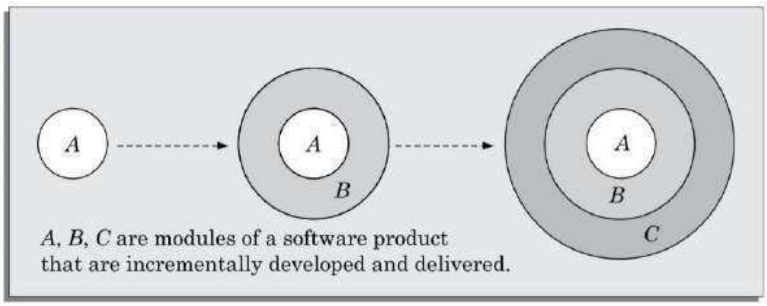
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthe systemaccordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

PATIENTAPPOINTMENTAND PRESCRIPTION MANAGEMENTSYSTEM

ProblemStatement–Patientmanagementsystemidentifiesinternaldatastores ofreports,doctors,diseases,teststhatmustbepresentinorderforthepatient systemtodoitsjob.Eachdatastoreshowstheflowofdataamongvariouspaths ofpatients,tests,doctors,reports,diseasesofthesystem.Patientrecords, medicineanddrugrecords,testrecordsareprocessedandreportsaregenerated.

ProcessModel–Incrementaldevelopmentmodel



Fig–incrementalmodelofsoftwaredevelopment

Thislifecyclemodelisalsoreferredtoassuccessiveversionsmodel.Inthislife cyclemodel,thesoftwareisfirstbrokendownintoseveralmoduleswhichcanbe incrementallyconstructedanddelivered.Thedevelopmentteamfirstdevelops thecoremodulesofthesystem.Thisinitialproductskeleton(A)isrefinedinto increasinglevelsofcapabilitybyaddingnewfunctionalities.Eachevolutionary versioncanbedevelopedbyiterativewaterfallmodel.Theusergetsachanceto experimentwiththepartiallydevelopedsoftwaremuchbeforethecomplete versionofthesystemisreleased.

SoftwareRequirementAnalysis

1. PatientModule–Inthismodule,apatientisregisteredandtheprofile ismaintained.

   Input:patientdetails
   Output:uniqueprofile

2. DoctorModule–Inthismodule,thedetailsofdoctorsaremaintained.

Input:patientdetails
Output:updateddatabase

3.DiagnosisModule–Inthismodule,thetestsareperformedandthe diseaseisdiagnosed. Adatabaseismaintainedforkeepingthepatientfiles.

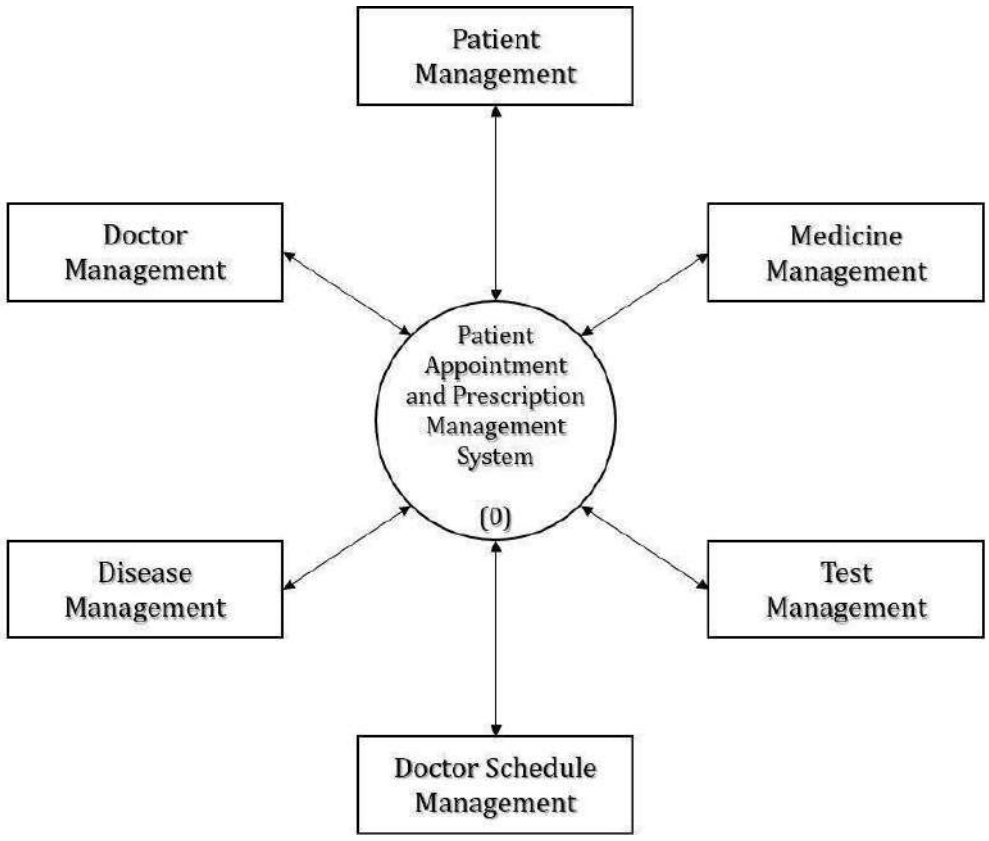
Input:patientdetails,doctordetails,testdetails
Output:diagnoseddisease,updateddatabase

4.ReportModule–Inthismodule,patientrecords,medicineanddrug records,testrecordsareprocessedandreportsaregenerated.

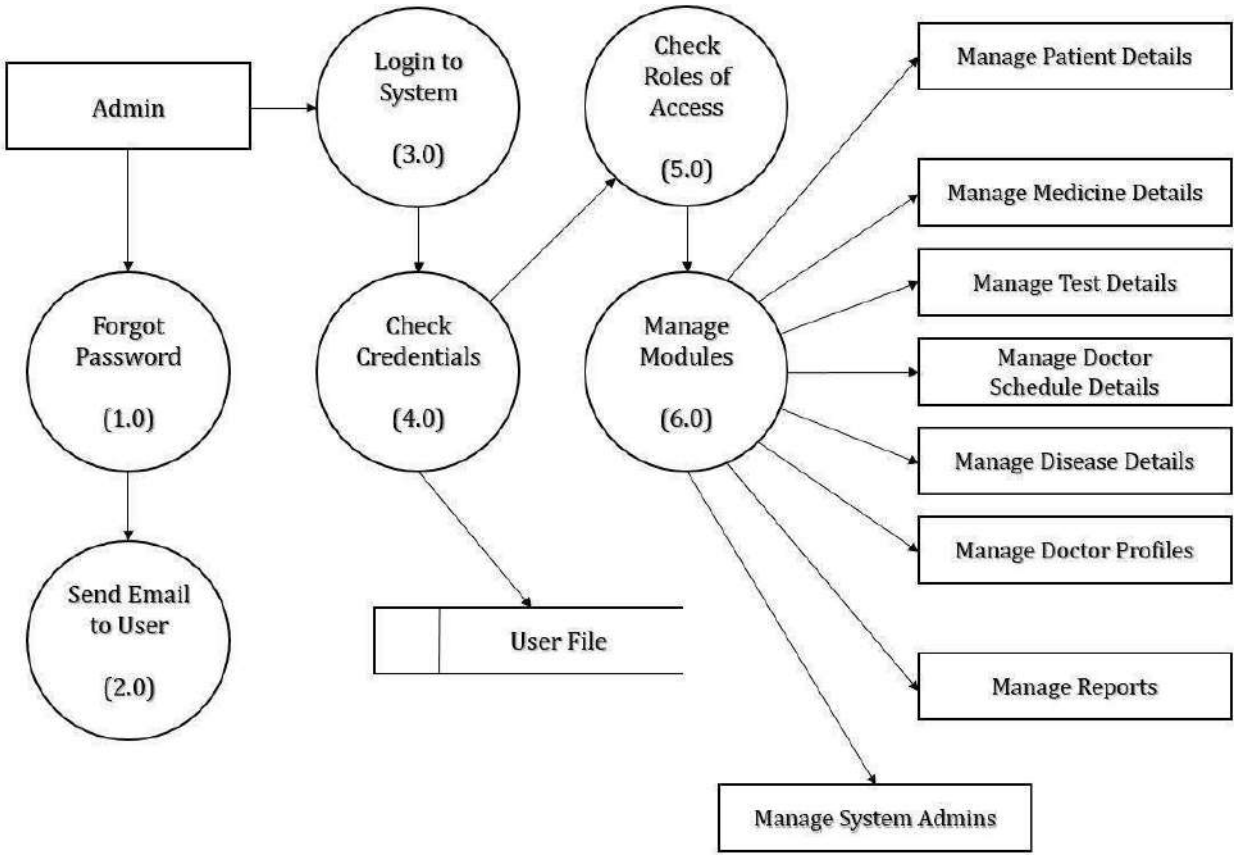
Input:patientdetails,doctordetails,diagnosisdetails
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



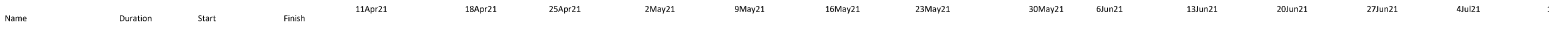
Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21

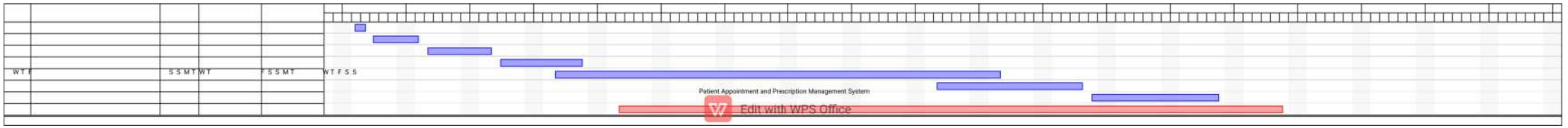

FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
MT WT WT MTW WT
T
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21 1 6 2 1 /7/

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM


1 ProblemStatement 1.5days12/4/21
2 FeasibilityStudy 10:00AM
3 RequirementAnalysisandSpecifications 3days101:40/04/A2M1
18Jul21
25Jul21
1Aug21
8Aug21
15Aug21
22
WTF SSM FSSMT TWT
WTF SSM WTF
T
SSMTW FSSMT T

4	Design	5days	20/4/21 10:00AM
5	CodingandUnitTesting	7days	28/4/21 10:00AM
6	IntegrationandSystemTesting	35days	4/5/21 10:00AM
7	Maintenance		
8	DocumentationandUserManual		



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

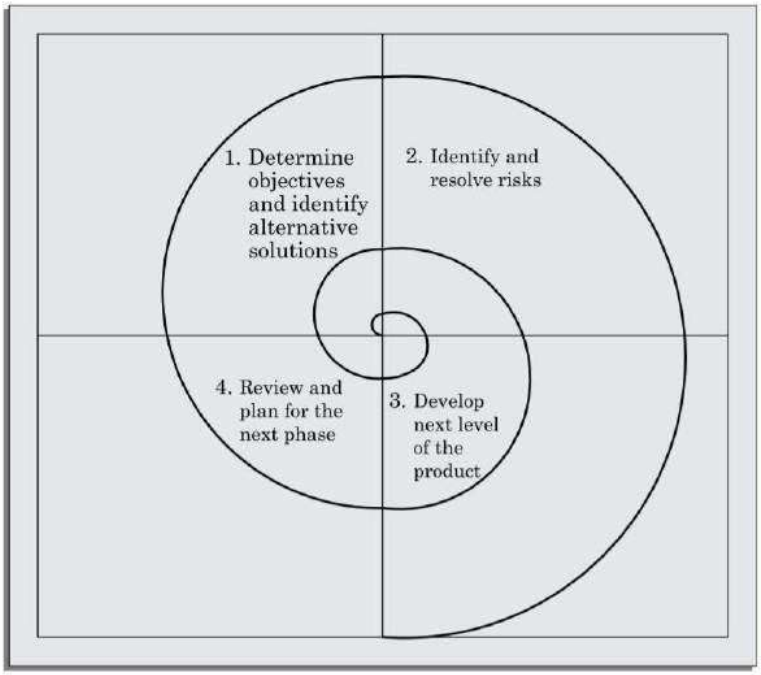
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

HOTELAUTOMATIONMANAGEMENTSYSTEM

ProblemStatement–Guestscanreservehotelroomsinadvanceoronthespot dependingontheavailabilityofrooms.Theoperatorwouldenterdatapertaining toguestssuchastheirarrivaltime,advancepay,approximatedurationofstay, purposeandtypeofroomrequiredi.e.AC,nonAC,deluxe,single/double occupancy.Thecomputershouldregisterthisdataanddependingonthe availabilityofasuitableroom,itshouldprovisionallyallotaroomnumbertothe guestandassignauniquetokennumber.Iftheguestcannotbeaccommodated, thecomputergeneratesanapologymessage.Thehotelcateringservice managerwouldinputthefooditemsandtheirquantityasandwhenconsumed bytheguestalongwiththetokennumberandthecorrespondingdateandtime. Itcanbeassumedthatdifferentfooditemshavedifferentcodesandhence cateringservicemanagerneednotenterthenameofthefooditems.Whenthe customerpreparestocheckout,thehotelautomationsoftwaregeneratesthebill alongwiththetaxes.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the

softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1.GuestModule–Inthismodule,aguestisregisteredandtheprofileis maintained.Ifasuitableroomisavailable,aroomnumberandunique tokennumberisalsoassigned.Elseanapologymessageisgenerated. Adatabaseismaintainedforkeepingtheinformation.

Input:guestdetails
Output:uniqueprofile,updateddatabase

2.CateringModule–Inthismodule,thedetailsoffooditemsconsumedby theguestismaintained.
Adatabaseismaintainedforkeepingtheinformation.

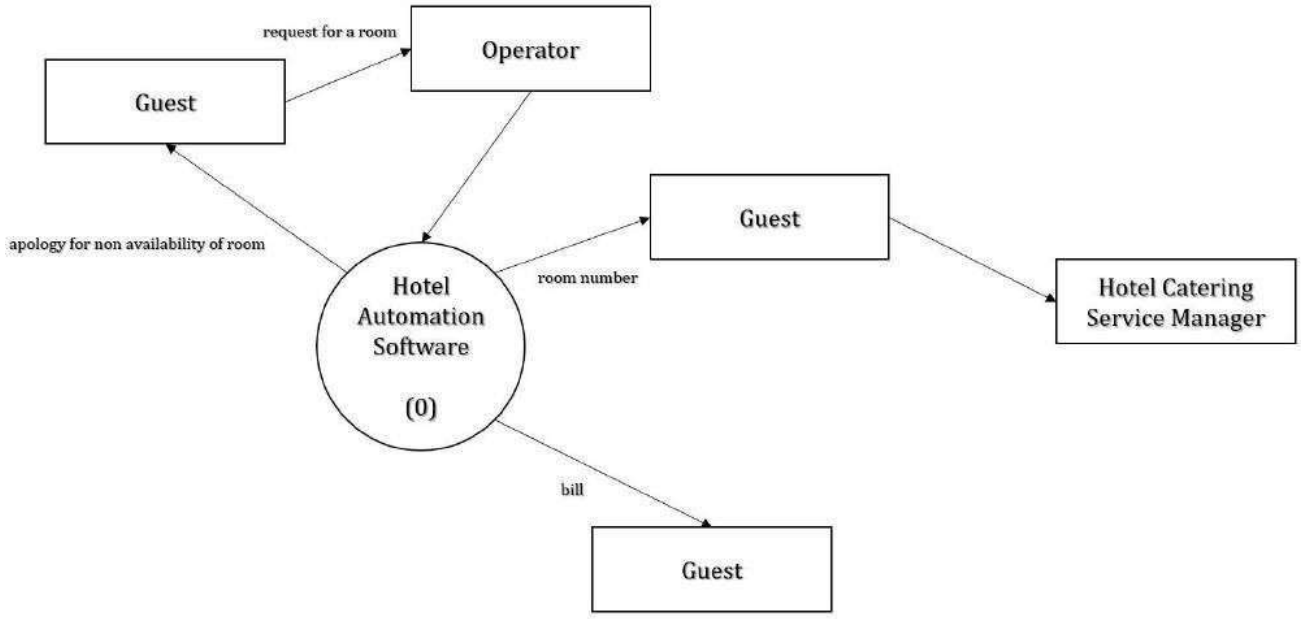
Input:guestdetails,roomnumber,tokennumber
Output:updateddatabase

3.BillModule–Inthismodule,thetotalbillisgeneratedwhenthe guestpreparestocheckout.

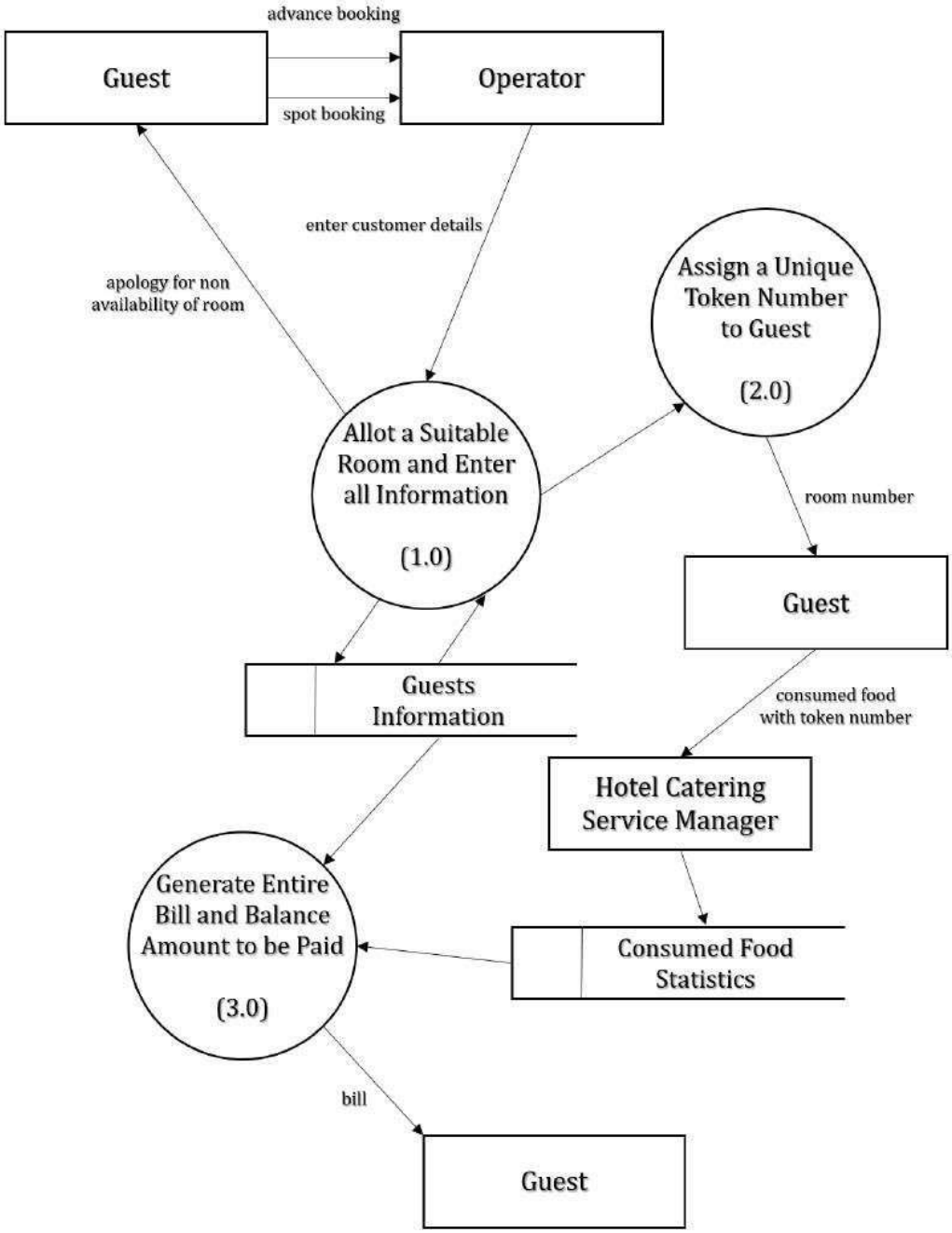
Input:guestdetails,roomnumber,tokennumber,fooditemsconsumed
Output:generatedbill

DesignEngineering

DataFlowDiagrams(DFDs)

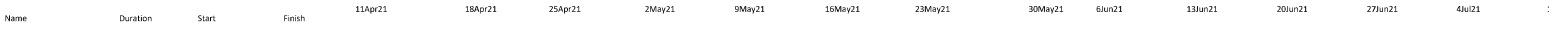


Fig–DFDlevelzero



Fig–DFDlevelone ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21


FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21 1 6 /7/
2 1

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM

1 ProblemStatement 1.5days12/4/21 18Jul21 25Jul21 1Aug21 8Aug21 15Aug21 22

2 CustomerCommunication 10:00AM

3 Planning 3days101:40/04/A2M1 WTF SSM WTF

WTF SSM FSSMT TWT T SSMTW FSSMT T

4	RiskAnalysis	5days	20/4/21 10:00AM	
5	Engineering	10:00AM		
6	ConstructionandRelease		7days	28/4/21 10:00AM
7	CustomerEvaluation			
8	DocumentationandUserManual		35days	4/5/21 10:00AM

RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

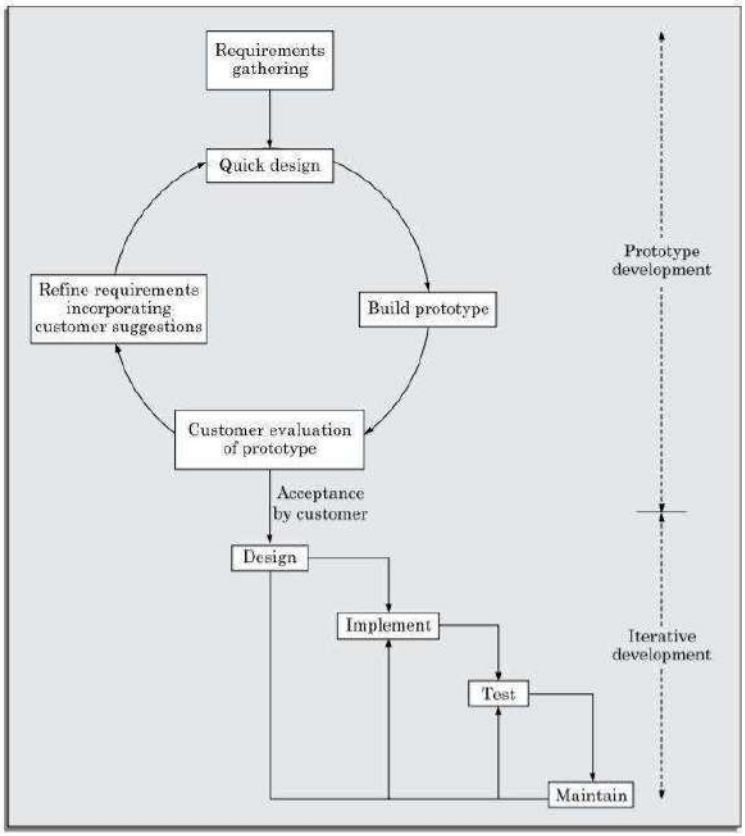
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

CRIMINALRECORDMANAGEMENT SYSTEM

ProblemStatement–Crimefilemanagementsystemidentifiesinternaldata storesofcourt,criminals,prisoners,chargesheet,FIRthatmustbepresentin orderforthecrimesystemtodoitsjob.Eachshowstheflowofdataamongthe variouspathsofcrime,FIR,criminals,courtofthesystem.Acriminalrecord managementsystemisimplementedforjailers,policeofficersandCBIofficers.

ProcessModel–Prototypingmodel



Fig–prototypingmodelofsoftwaredevelopment
Thismodelallowsallorpartofasystemtobeconstructedquickly.
Goalofprototypingistoreducethechanceofuncertainty.Thephases include–

1)meetingofthedevelopersandthecustomerstodefinetheoverall objectivesofthesoftware
2)quickdesignleadstoconstructionofprototype
3)evolutionofprototypebycustomer

4)iterationoccursastheprototypeisturnedtotheneedsofthecustomers thecodeforprototypeisthrownaway–theexperiencegatheredfrom developingtheprototypehelpsindevelopingtheactualsystem

SoftwareRequirementAnalysis

1.AdminModule–Inthismodule,anadminisregisteredandtheprofile ismaintained.Thelevelofaccessisgrantedaccordingtothe designation.

Input:admindetails
Output:uniqueprofile,grantedlevelofaccess

2.ComplainModule–Inthismodule,thecomplaintisregisteredandFIR isfiled. Adatabaseismaintainedforkeepingthecomplaintdetails.

Input:complaintdetails
Output:filedFIR,updateddatabase

3.CrimeModule–Inthismodule,thedetailsofthecrimearemaintained andchargesheetisfiled.

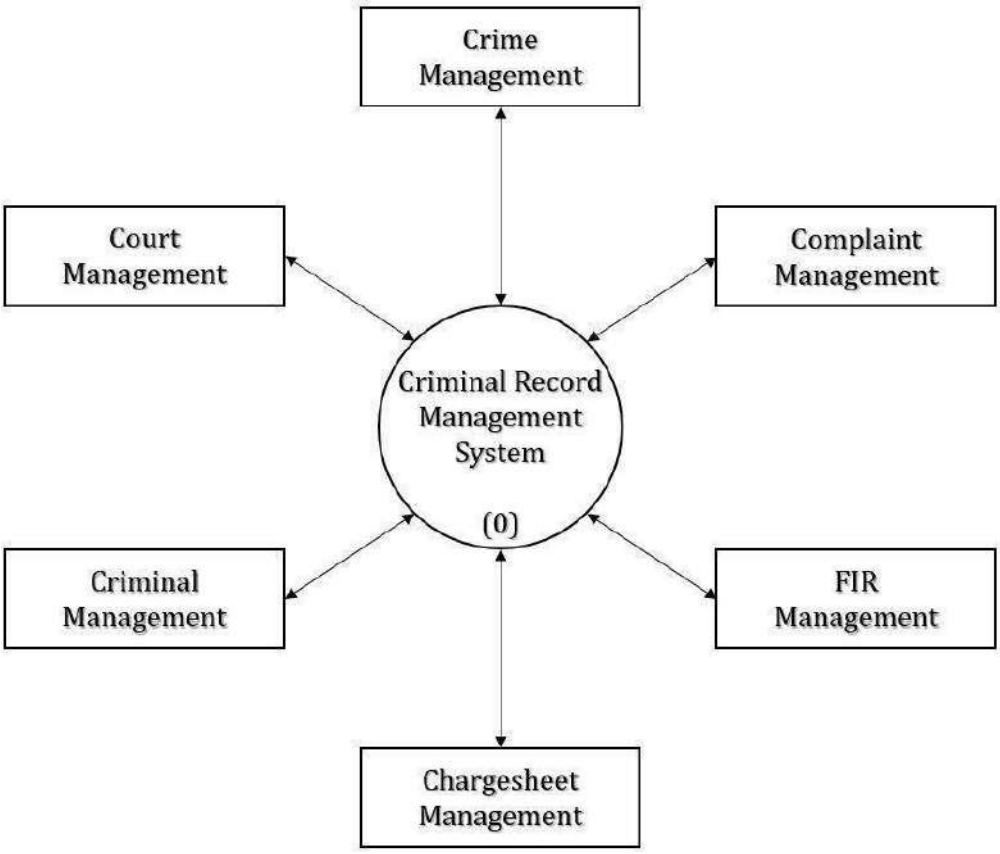
Input:complaintdetails,FIRdetails
Output:filedchargesheet

4.CourtModule–Inthismodule,thecaseistakentothecourtandlegal actionistakenaccordingly.

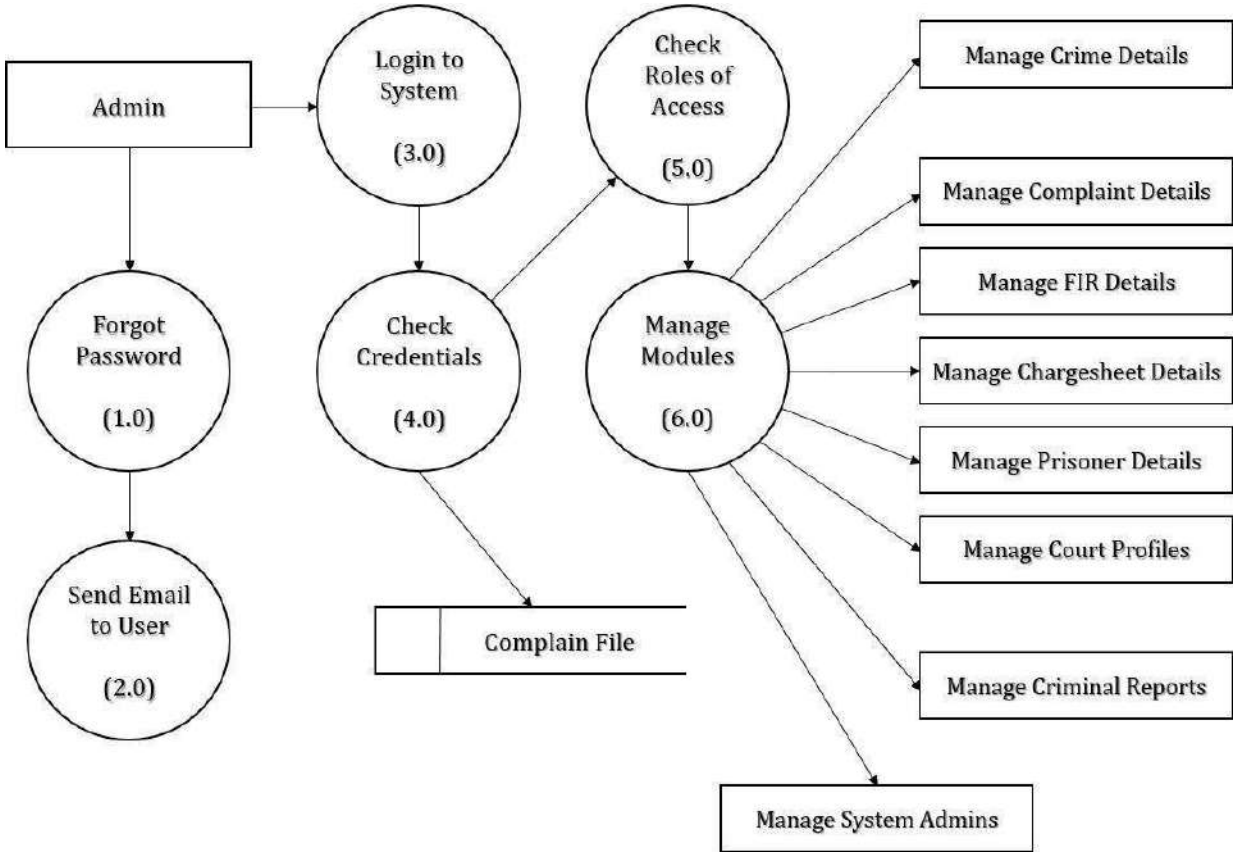
Input:complaintdetails,FIRdetails,chargesheetdetails
Output:legalaction,updateddatabase

DesignEngineering

DataFlowDiagrams(DFDs)



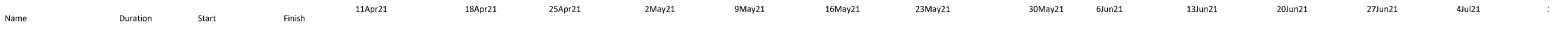
Fig–DFDlevelzero

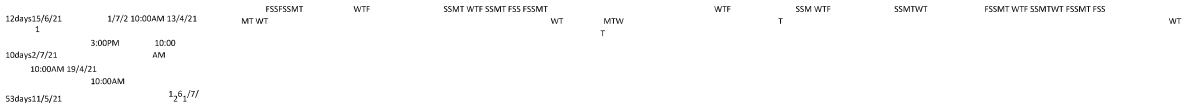


Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21

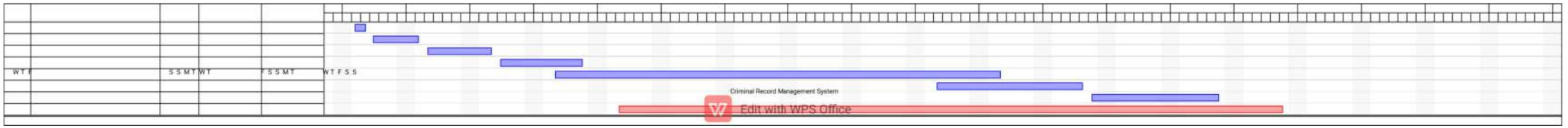

FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
MT WT WT MTW WT
T
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21 1 6 /7/
2 1

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM


1 ProblemStatement 1.5days12/4/21
2 FeasibilityStudy 10:00AM
3 RequirementAnalysisandSpecifications 3days101:40/04/A2M1
18Jul21
25Jul21
1Aug21
8Aug21
15Aug21
22
WTF SSM FSSMT TWT
WTF SSM WTF
T
SSMTW FSSMT T

4	Design	5days	20/4/21 10:00AM
5	CodingandUnitTesting	7days	28/4/21 10:00AM
6	IntegrationandSystemTesting	35days	4/5/21 10:00AM
7	Maintenance		
8	DocumentationandUserManual		



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

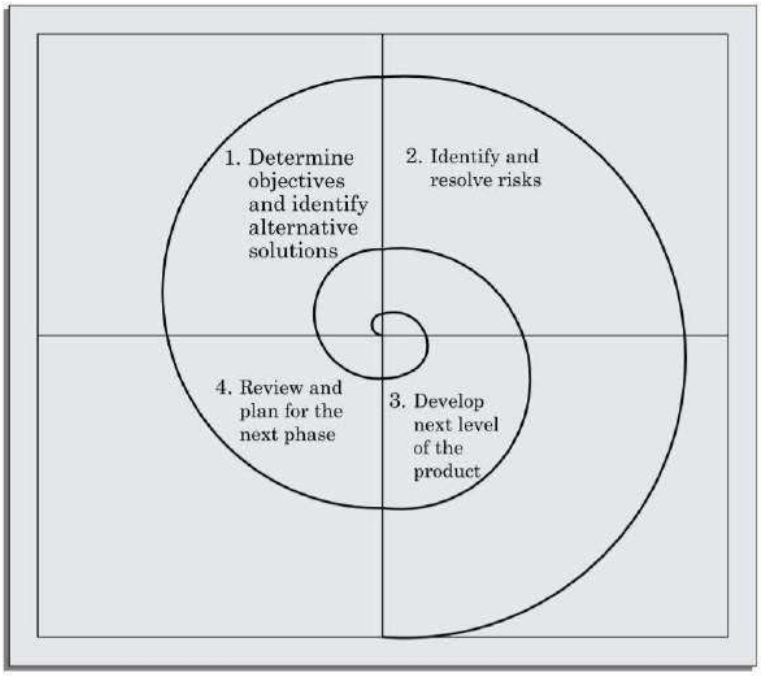
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

EXAMINATIONANDRESULTCOMPUTATIONSYSTEM

ProblemStatement–Examinationmanagementsystemidentifiesinternaldata storesofsubject,course,branch,student,classthatmustbepresentinorderfor theexaminationsystemtodoitsjob.Eachshowstheflowofdataamongthe variouspartsoftimetable,class,course,subject,branchofthesystem.Themain functionalitiesoftheexaminationareprocessingtimetablerecords,examination records,classrecords,studentrecords,branchrecords,courserecords,subject recordsandgeneratereportofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectofwaterfall model.Itisderivedintoframeworkofactivitiesalsocalledtaskregions.

Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1.StudentModule–Inthismodule,astudentisregisteredandtheprofile ismaintained. Adatabaseismaintainedforkeepingthestudentinformation.

Input:studentdetails
Output:uniqueprofile

2.TimetableModule–Inthismodule,thedetailsofbranch,course,class andsubjectismaintainedandthetimetableiscreated. Adatabaseismaintainedforkeepingthetimetableinformation.

Input:branchdetails,coursedetails,classdetails,subjectdetails
Output:updateddatabase,createdtimetable

3.ExaminationModule–Inthismodule,theexaminationtakesplaceand theresultsarecomputed.

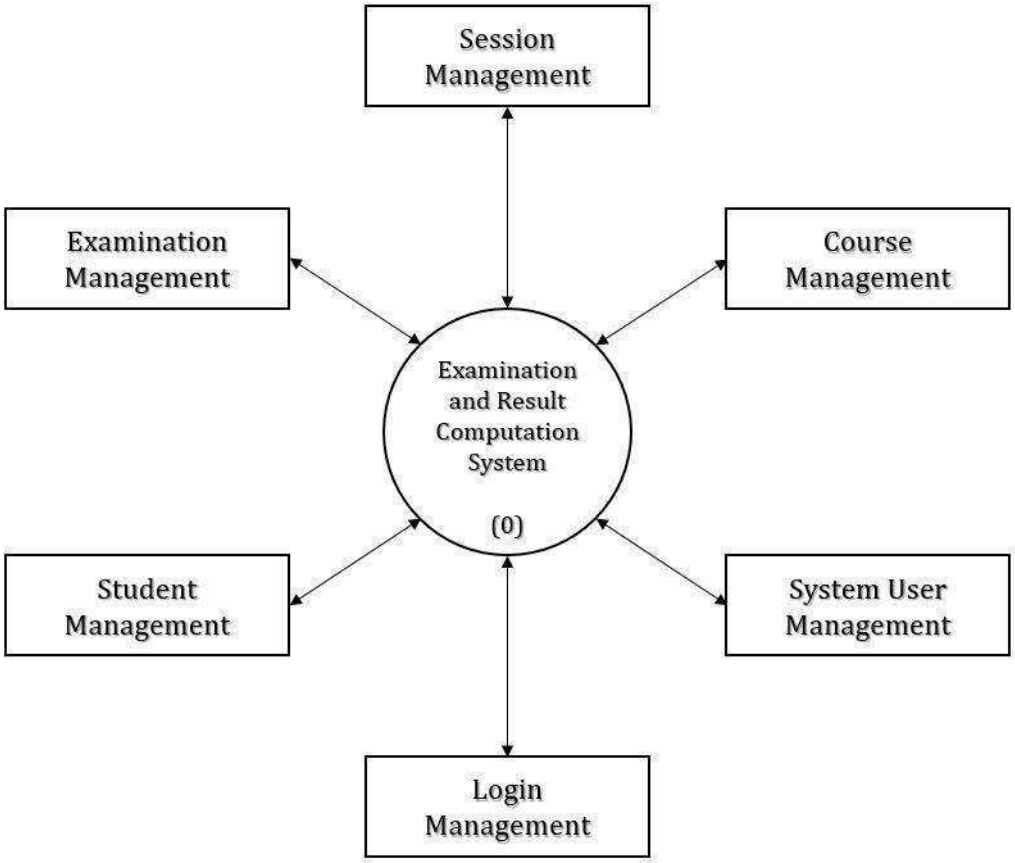
Input:studentdetails,timetabledetails
Output:computedresults,updateddatabase

4.ReportModule–Inthismodule,timetable,examination,class,student, branch,courseandsubjectrecordsareprocessedandreportsare generated.

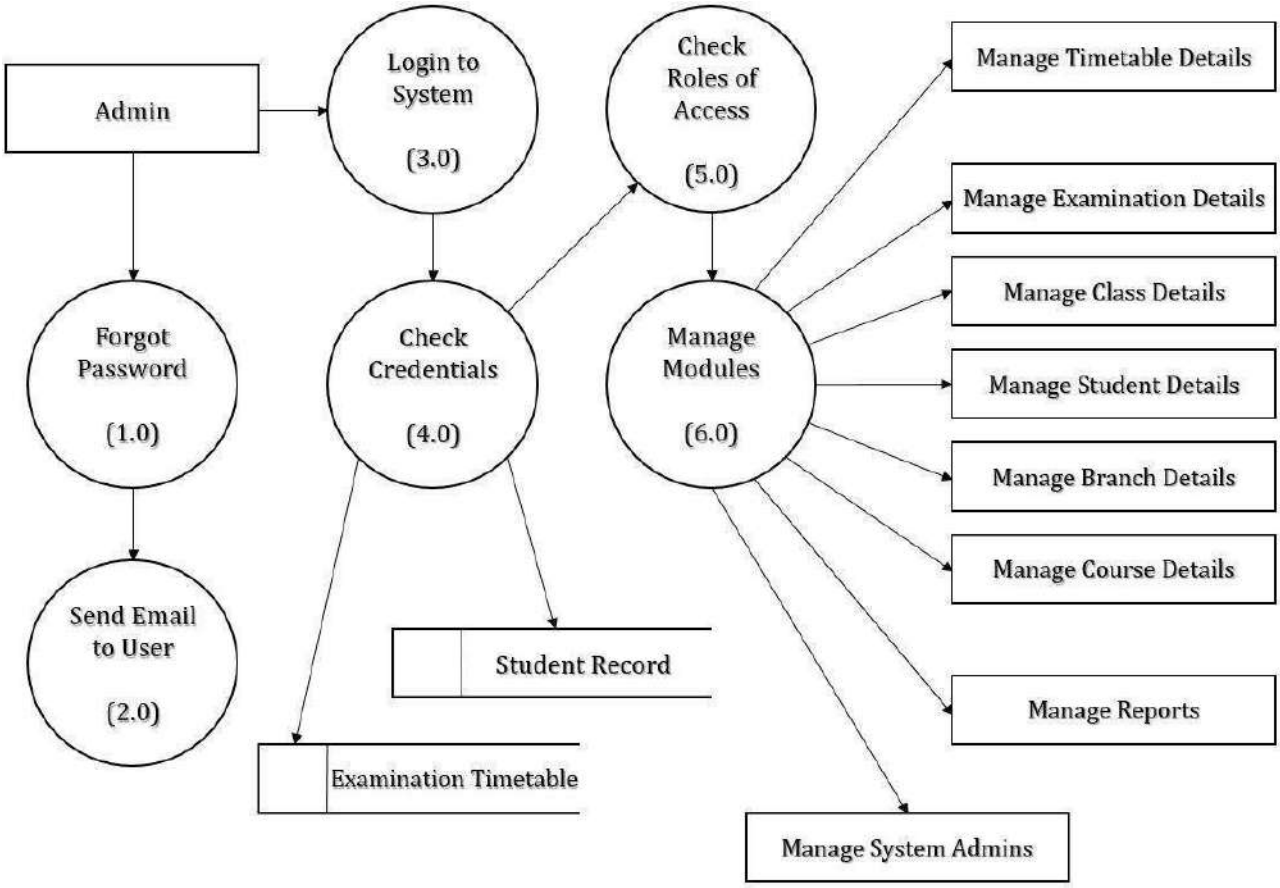
Input:guestdetails,roomnumber,tokennumber,fooditemsconsumed
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



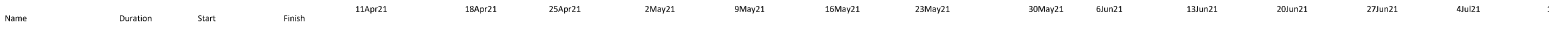
Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21


FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM

1 ProblemStatement 1.5days12/4/21

2 CustomerCommunication 10:00AM

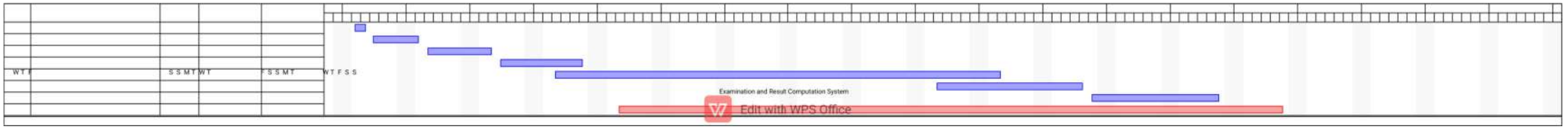
3 Planning 3days101:40/04/A2M1

18Jul21 25Jul21 1Aug21 8Aug21 15Aug21 22

WTF SSM WTF

WTF SSM FSSMT TWT T SSMTW FSSMT T

4	RiskAnalysis	5days	20/4/21 10:00AM	
5	Engineering	10:00AM		
6	ConstructionandRelease		7days	28/4/21 10:00AM
7	CustomerEvaluation			
8	DocumentationandUserManual		35days	4/5/21 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

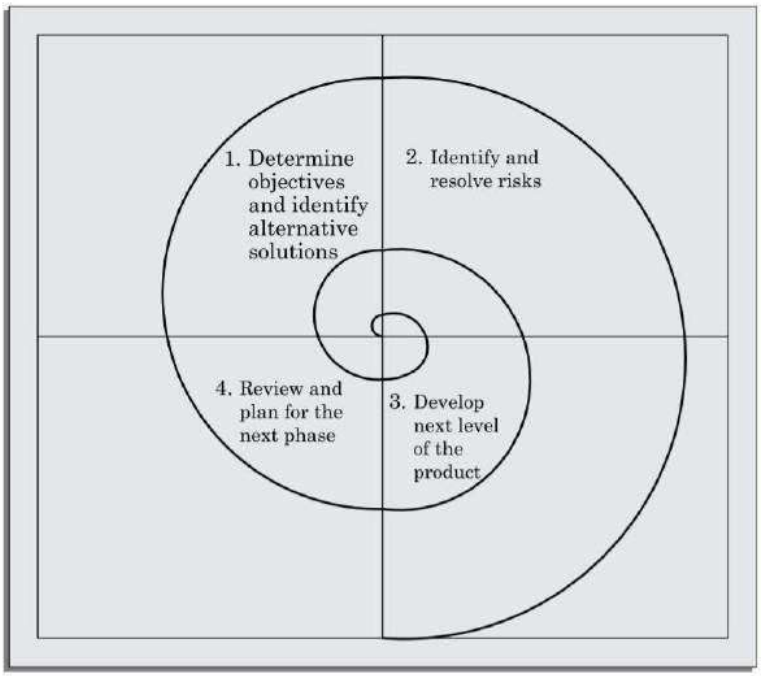
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

PARKINGALLOCATION SYATEM

ProblemStatement–Parkingmanagementsystemidentifiesinternaldatastores oflogin,customers,durationandparkingfeesthatmustbepresentinorderfor theparkingsystemtodoitsjob.Itshowstheflowofdatabetweenthevarious pathsofparkingslotsandparkingfees.Thefunctionalitiesofparkingsystemare processingofparkingslots,vehiclerecords,parkingfees,durationandgenerating thereportofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredandthe profileismaintained.

Input:customerdetails Output:uniqueprofile

2.VehicleModule–Inthismodule,thecustomer'svehicleisregistered.

Input:customerdetails,vehicledetails
Output:updateddatabase

3.ParkingModule–Inthismodule,parkingslotisallottedandfeeis chargedaccordingly. Adatabaseismaintainedforkeepingtheparkinginformation.

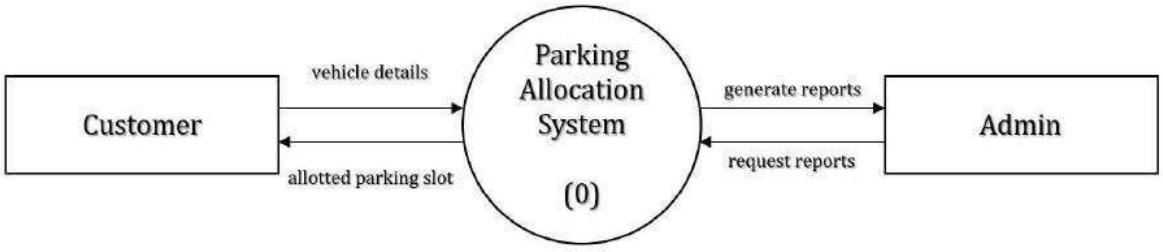
Input:customerdetails,vehicledetails,duration
Output:updateddatabase

4.ReportModule–Inthismodule,parkingslot,vehicle,parkingfee anddurationrecordsareprocessedandreportsaregenerated.

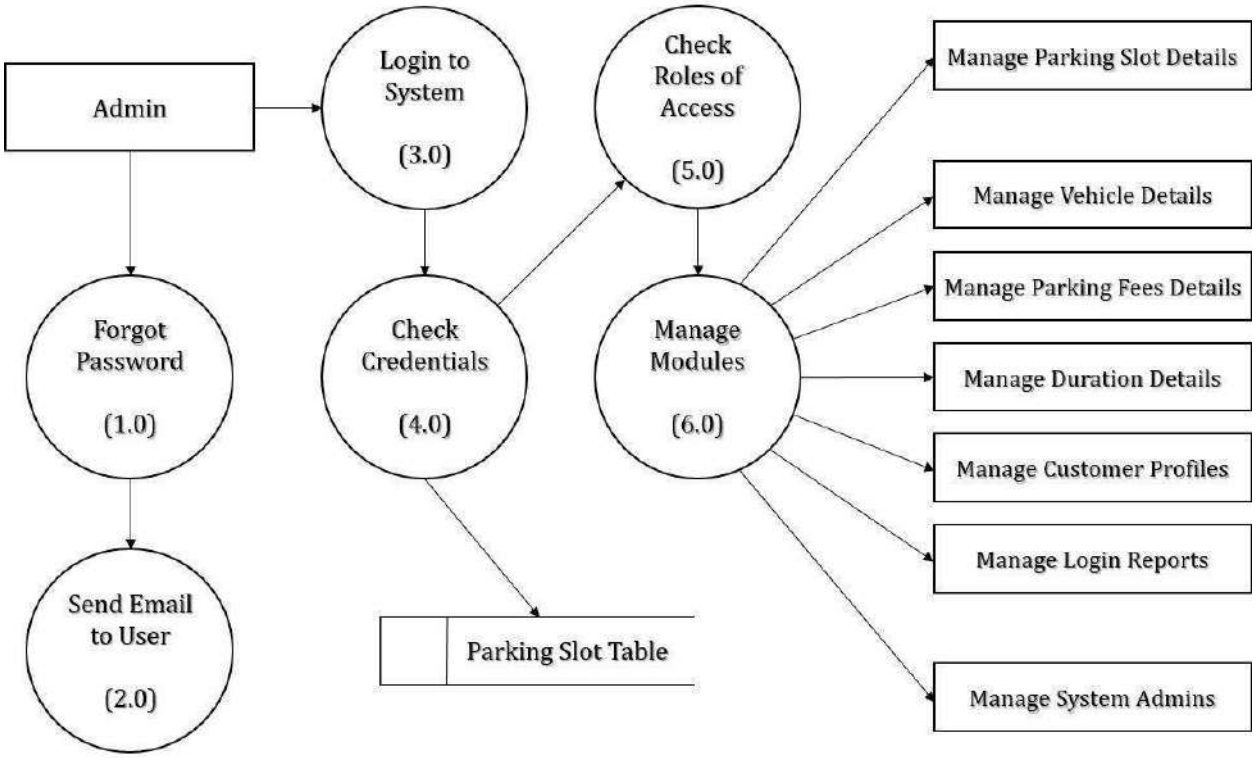
Input:parkingslotdetails,vehicledetails,parkingfeedetails,durationdetails
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



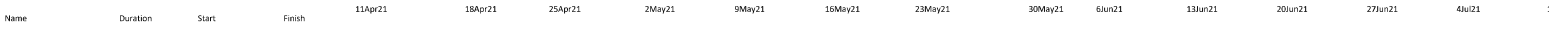
Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21


FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM

1 ProblemStatement 1.5days12/4/21

2 CustomerCommunication 10:00AM

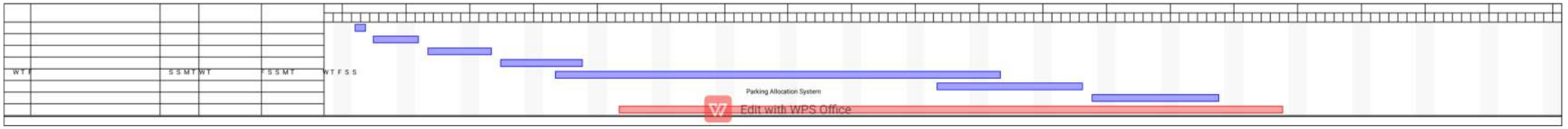
3 Planning 3days101:40/04/A2M1

18Jul21 25Jul21 1Aug21 8Aug21 15Aug21 22

WTF SSM WTF

WTF SSM FSSMT TWT T SSMTW FSSMT T

4	RiskAnalysis	5days	20/4/21 10:00AM	
5	Engineering	10:00AM		
6	ConstructionandRelease		7days	28/4/21 10:00AM
7	CustomerEvaluation			
8	DocumentationandUserManual		35days	4/5/21 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

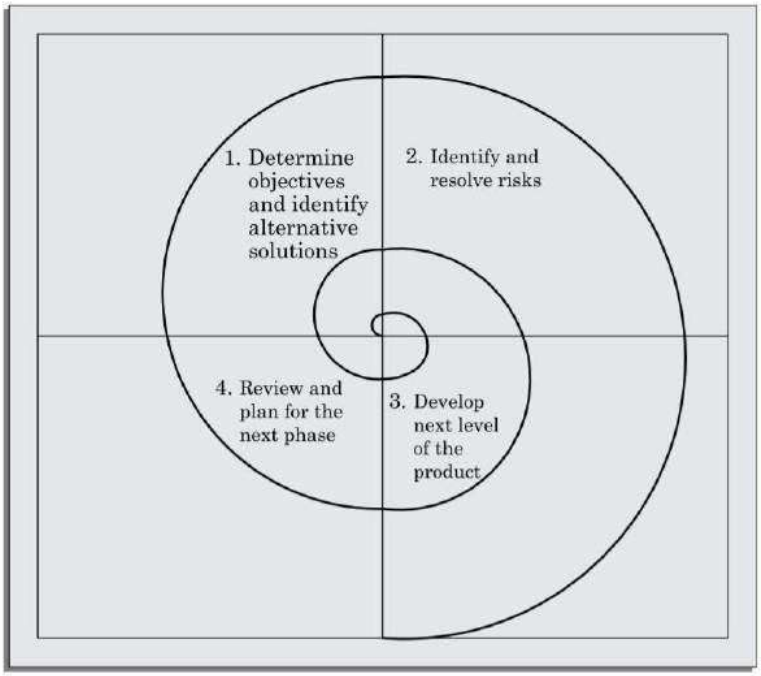
ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthesystem accordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

WHOLESALEMANAGEMENTSYSTEM

ProblemStatement–Itprovidesallthefunctionalitiesoftheretailstore managementsystemasawhole.Itidentifiesinternaldatastoresofcustomer, billingcounter,bill,retailprice,productthatmustbepresentinorderfortheretail storesystemtodoitsjob.Storerecords,productrecords,retailpricerecords,bill records,billingcounterrecordsandcustomerrecordsareprocessedandreports aregeneratedofthesame.

ProcessModel–Spiraldevelopmentmodel



Fig–spiralmodelofsoftwaredevelopment

Itcouplestheiterativenatureofprototypingwiththesystematicaspectof waterfallmodel.Itisderivedintoframeworkofactivitiesalsocalledtaskregions. Normallythereare4to6taskregions.Asthisevolutionaryprocessbegins,the softwareengineeringmovesaroundthespiralinclockwisedirectionbeginningat thecore.Subsequentpassesaroundthespiralmodelmightbeusedtodevelopa prototypeandthenprogressivesophisticatedversionsofprototypes.

SoftwareRequirementAnalysis

1.CustomerModule–Inthismodule,acustomerisregisteredandthe profileismaintained.

Input:customerdetails
Output:uniqueprofile

2.StockModule–Inthismodule,thedetailsofstocksare maintained.Adatabaseismaintainedforkeepingthestock information.

Input:stockdetails
Output:updateddatabase

3.BillModule–Inthismodule,allthebillsare maintained.Adatabaseismaintainedforkeepingthe billinformation.

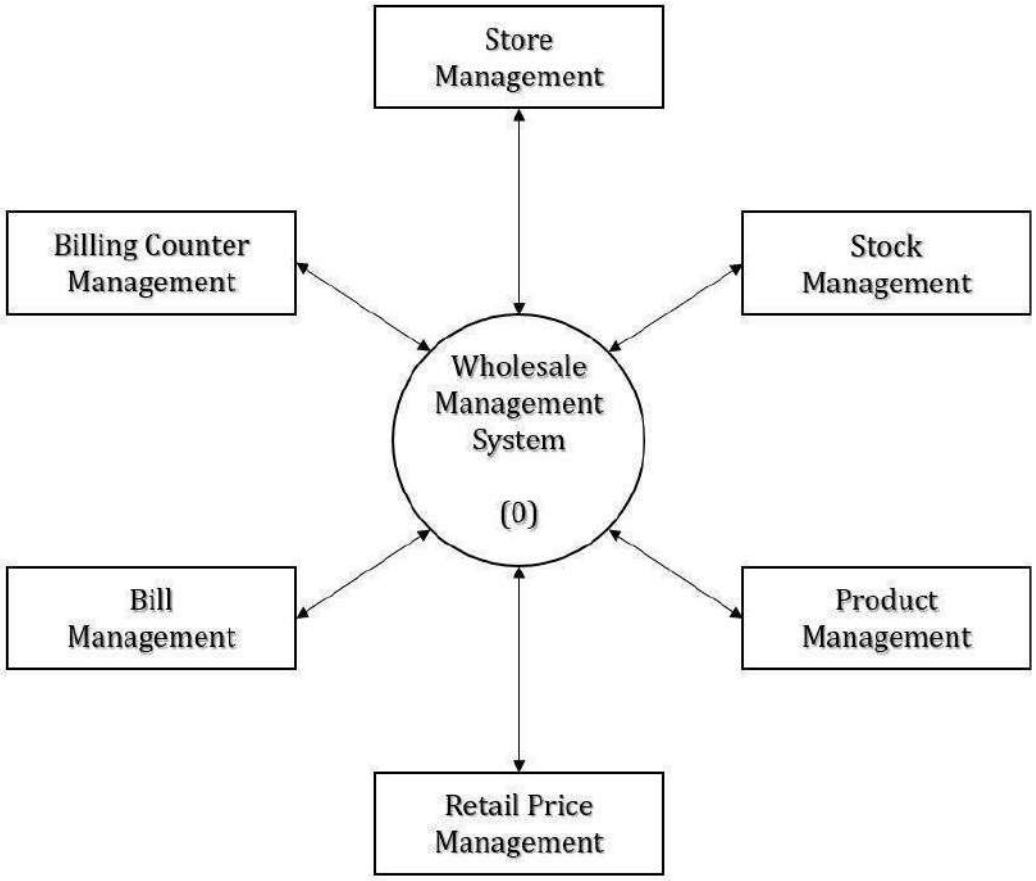
Input:customerdetails,stockdetails
Output:generatedbills,updateddatabase

4.ReportModule–Inthismodule,store,product,retailprice,bill, billingcounterandcustomerrecordsareprocessedandreportsare generated.

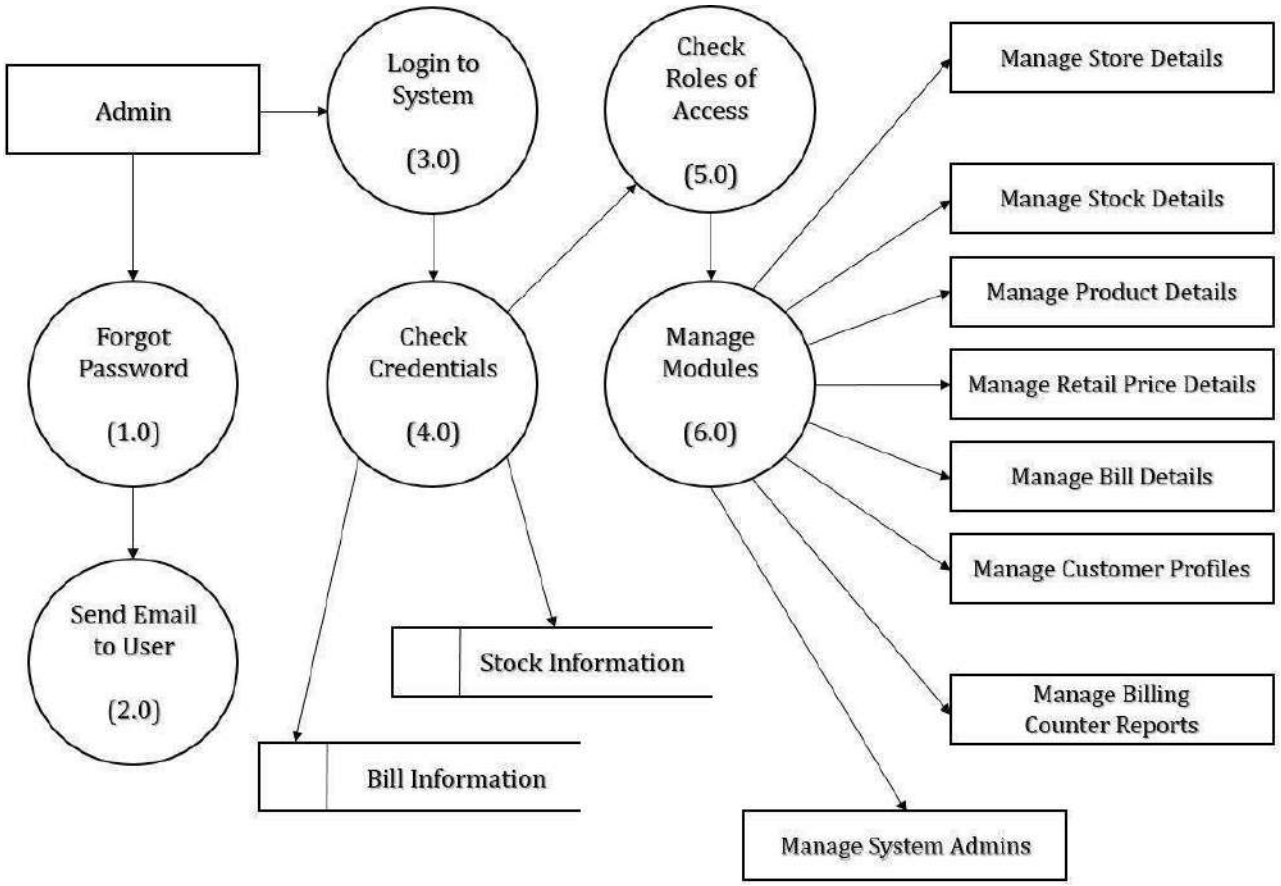
Input:storedetails,productdetails,retailpricedetails,billdetails,billing counterdetails,customerdetails
Output:generatedreports

DesignEngineering

DataFlowDiagrams(DFDs)



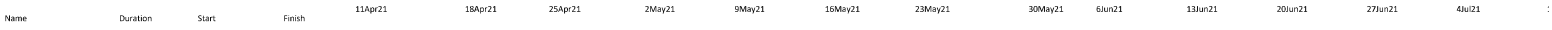
Fig–DFDlevelzero



Fig–DFDlevelone

ProjectManagement

ProjectSchedule–Thefilehasbeenattached.


Name
Duration
Start
Finish
11Apr21
18Apr21
25Apr21
2May21
9May21
16May21
23May21
30May21
6Jun21
13Jun21
20Jun21
27Jun21
4Jul21


FSSFSSMT WTF SSMT WTF SSMT FSS FSSMT WTF SSM WTF SSMTWT FSSMT WTF SSMTWT FSSMT FSS
12days15/6/21 1/7/2 10:00AM 13/4/21
1
3:00PM 10:00
10days2/7/21 AM
10:00AM 19/4/21
10:00AM
53days11/5/21

7/5/21
10:00AM 23/7/ 21
22/6/21 10:00
10:00AM AM

10:00AM 27/4/21 10:00
10:00AM
AM

1 ProblemStatement 1.5days12/4/21

2 CustomerCommunication 10:00AM

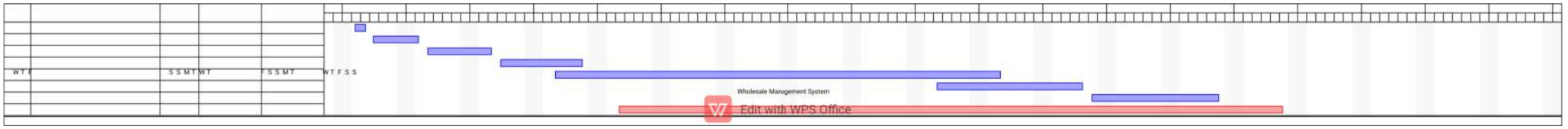
3 Planning 3days101:40/04/A2M1

18Jul21 25Jul21 1Aug21 8Aug21 15Aug21 22

WTF SSM WTF

WTF SSM FSSMT TWT T SSMTW FSSMT T

4	RiskAnalysis	5days	20/4/21 10:00AM	
5	Engineering	10:00AM		
6	ConstructionandRelease		7days	28/4/21 10:00AM
7	CustomerEvaluation			
8	DocumentationandUserManual		35days	4/5/21 10:00AM



RiskAnalysis

RiskAssessment–Eachriskshouldfirstberatedintwoways.

i) Thelikelihoodofariskcomingtrue(r)

ii) Theconsequencesoftheproblemsassociatedwiththerisk(s)

Basedonthesetwofactors,thepriorityofeachriskcanbecomputedasp=r×s wherep–prioritywithwhichtheriskmustbehandled r–theprobabilityoftheriskbecomingtrue s–theseverityofdamagecausedduetoriskbecomingtrue

–

=

Testing

UnitTesting–Eachcomponentofthedesignisimplementedasaprogram module.Eachmoduleisunittestedtodeterminethecorrectworkingofall individualmodules.

IntegrationandSystemTesting–Duringtheintegrationandsystemtesting phase,themodulesareintegratedinaplannedmanner.Systemtestingconsists ofthreedifferentkindsoftestingactivities.

i) α(Alpha)Testing–Itisthesystemtestingperformedby thedevelopmentteaminacontrolledenvironment.

ii) β(Beta)Testing–Itisthesystemtestingperformedbyfriendlyset ofcustomersinanuncontrolledenvironment.

iii) AcceptanceTesting–Itisthesystemtestingperformedbythe customersthemselvesaftertheproductdeliverytodeterminewhether toacceptorrejectthedeliveredproduct.

Maintenance–Therelativeeffortofdevelopmentofatypicalsoftwareproduct toitsmaintenanceeffortisintheratio40:60.

Maintenanceinvolvesthreekindsofactivities.

i) Correctivemaintenance–Correctingerrorsthatwerenot discoveredduringtheproductdevelopmentphase.

ii) Perfectivemaintenance–Enhancingthefunctionalitiesofthe systemaccordingtothecustomer'srequirements.

iii) Adaptivemaintenance–Portingthesoftwaretoworkina newenvironment.

Organized Retail Shopping Management Software

Name: Akhlakh Ahmed Reja
College Roll: 1736
Semester: V
Department: Computer Science

Supervisor: Dr. Chayan Halder

Project -1

Organized Retail Shopping Management Software

Problem Statement:

A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number, driving license number. Each customer who registers for this scheme is assigned the customer number (CN) by the computer. A customer can present his CN to the checkout staff when he makes his purchase. In this case the value of this purchase is credited again in its CN. At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh /-.The entries against the CN are reset on the last day of each year after the prize winner's lists are generated.

Process Model:

Supermarket management software needs to change the information in a basis of regular interval. Every year the number of customers, the purchase history and other necessary information have to be changed. So the software needs to be added more new functionalities. A supermarket exists in a competitive environment, where it acts as a value-added intermediary between geographically dispersed supplier companies and the scattered individual customers who eventually buy their products. The supermarket management involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This incremental model combines the elements of the waterfall model with the iterative philosophy of prototyping. That is basic requirements are addressed, but many supplementary features (some known, others unknown) remain undelivered. The core product is used by the customer (or undergoes detailed review). As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. So for a supermarket management software system the **Incremental model** is the most suitable model.

Requirement Analysis:

Basically three modules are used in a Supermarket management system. Consumer module, Purchase module, Promotional Module. On basis of these three modules supermarket management software can be designed.

- ### Consumer Module:

A supermarket needs to develop the following software to a regular customer. For this the customer needs to supply his residence number, telephone number. When a customer registers his name in the system he will be provided with a unique Customer number (CN).To maintain the record of the customer and their CN a database is maintained.

Input: Name, residence number, telephone number, driving licenses all other necessaries.
Output: Customer number (CN).

- ### Purchase Module:

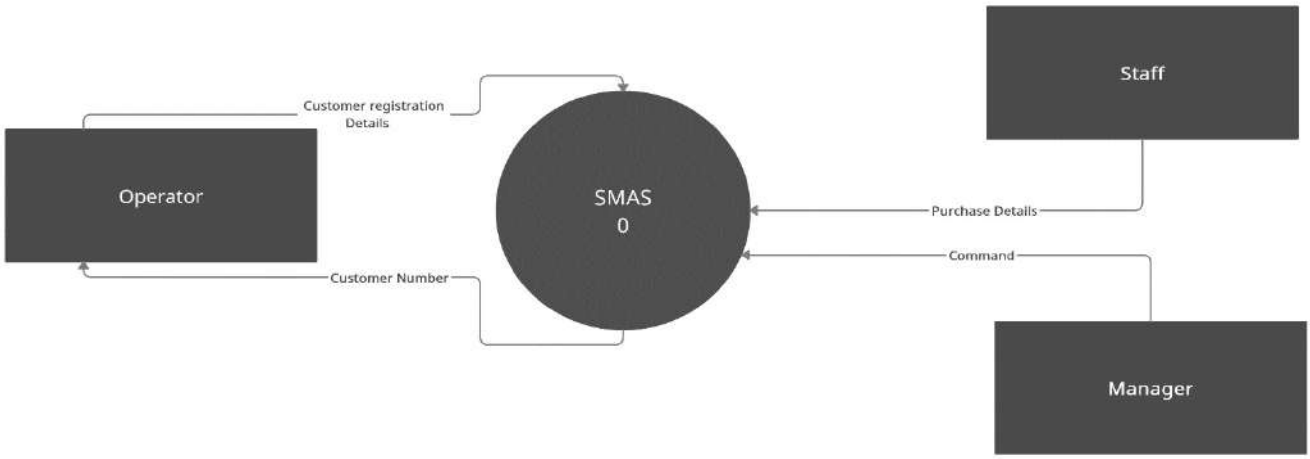
When a customer made his purchase his name, bought product, and buying time is recorded. To maintain the purchase history of every customer a separate database is maintained. According to that the amount is deducted from the buyer.
Input: Customer name, bought product, time
Output: Purchase history database

- ### Promotional module:

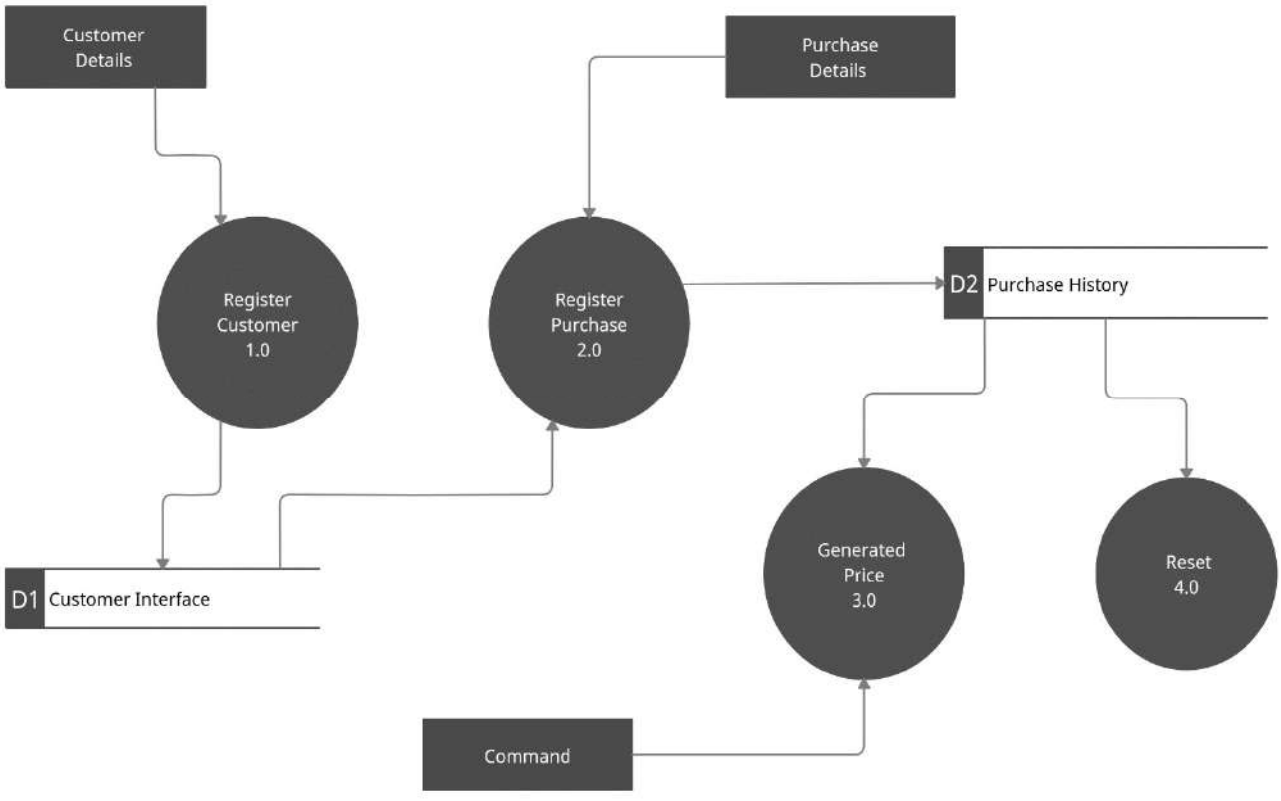
At the end of each year as its 31st December, the supermarket intense to aware surprise gifts to ten customers who made highest total purchase over the year. It also intense to aware a 22 carat gold coin to every customer whose purchase exceeds Rs 1 Lakh /-.The entries against the CN are reset on the last day of each year after the prize winner's lists are generated. After lists are created the all history is reset.
Input: Purchase history according to the customer number
Output: Surprise Gift Winner List, Gold coin Winner List

Design Engineering:

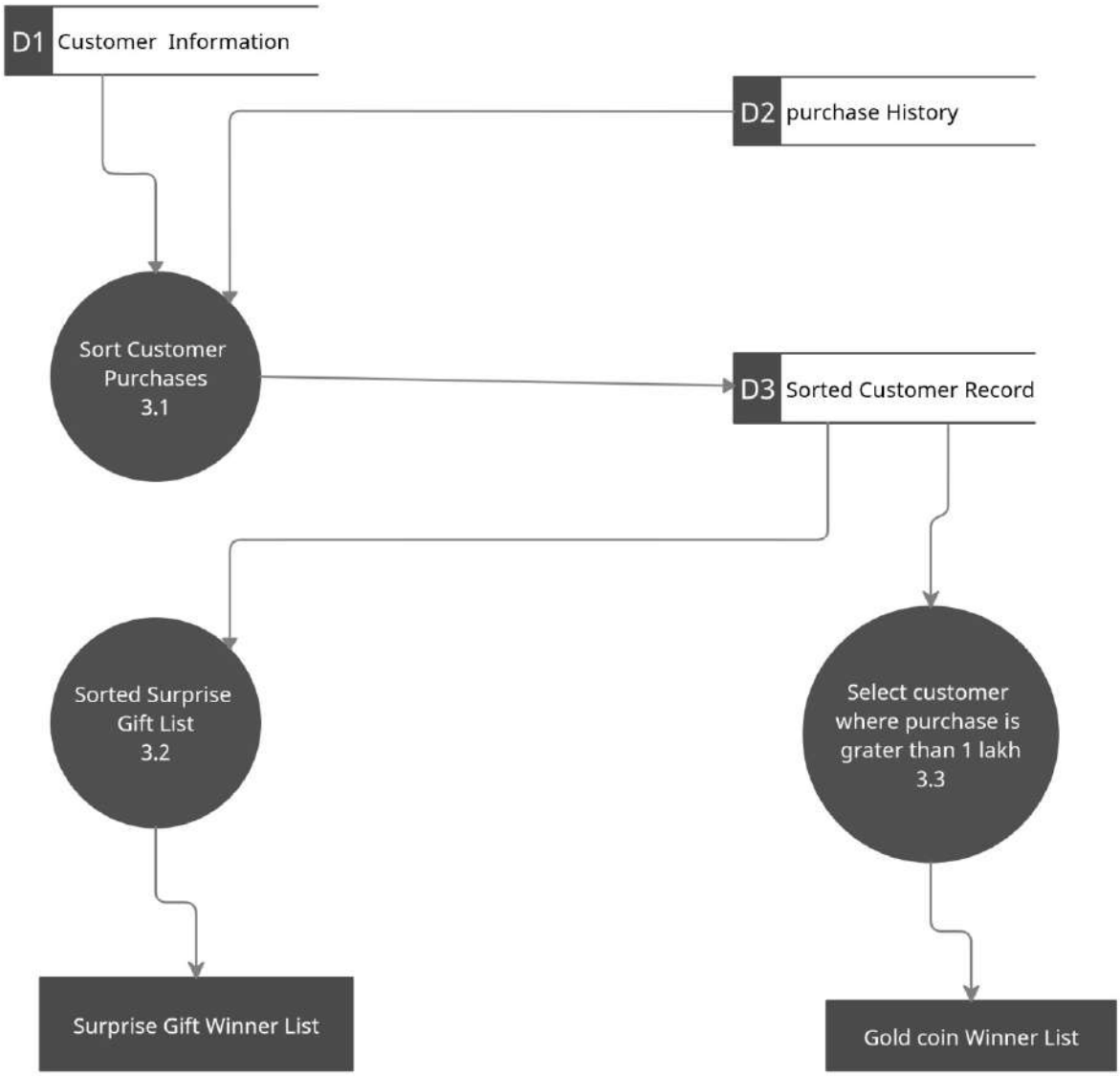
- ### Zero Level DFD(Context Diagram):



- level One DFD:



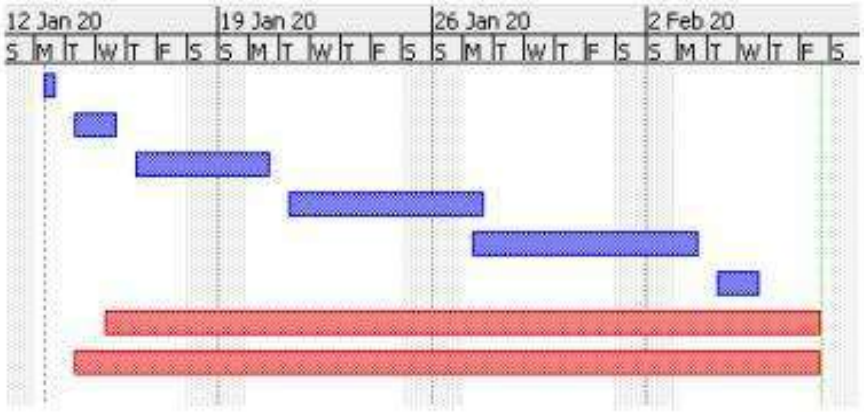
- Level Two of process 3:



Project Management:

Project Scheduling:

		Name	Duration	Start	Finish
1		Problem statement	1 day?	13/1/20 8:00 AM	13/1/20 5:00 PM
2		Feasibility study	2 days?	14/1/20 8:00 AM	15/1/20 5:00 PM
3		Requirement analysis & specification	3 days?	16/1/20 8:00 AM	20/1/20 5:00 PM
4		Design	5 days?	21/1/20 8:00 AM	27/1/20 5:00 PM
5		Coding & unit testing	6 days?	27/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	18 days?	15/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	19 days?	14/1/20 8:00 AM	7/2/20 5:00 PM



- Project Resources:

- **Human Resources :**
  - project manager
  - System analyst
  - Developer
  - Designer
  - Tester
  - Document creator

- **Software Resource :**
  - MS Word
  - MS Project
  - C (For forehand programming)
  - SQL (For backhand programming)

- **Hardware Resource :**
  - Processor: Intel() Core(TM) i5-4310U 2.60Ghz
  - Ram : 12.00 GB

Risk Management plan:

Risk factor =

$$\frac{\text{(Risk exposure before reduction-Risk exposure after reduction)}}{\text{Cost of reduction}}$$

No such risk is there in the supermarket management system as it uses incremental model.

Testing:
Basic Path Testing (White Box Testing), Unit Testing followed by Integration Testing.

Maintenance: Maintenance stands for all modifications and updation done after the delivery of software products. Shop maintenance is the performance of adaptive, preventive and corrective maintenance for a given shop. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

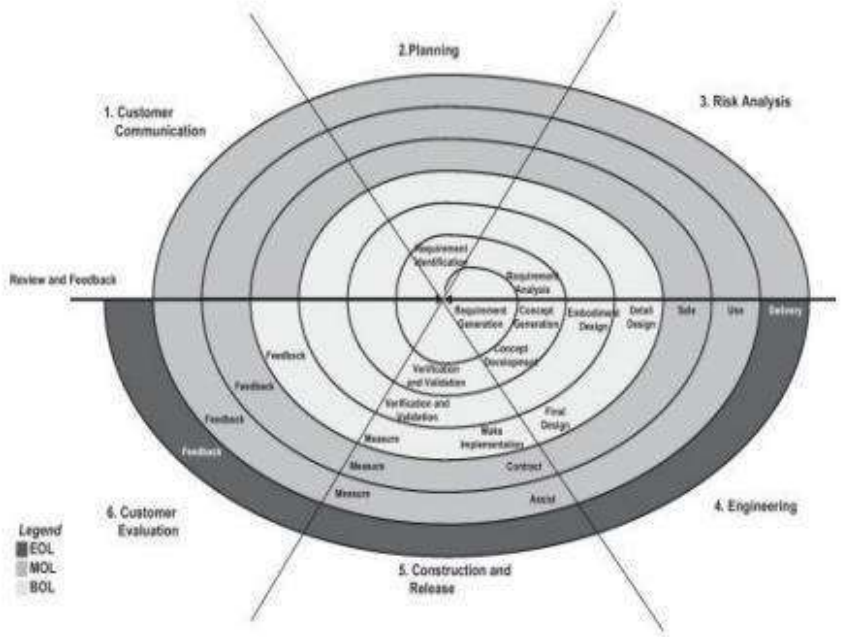
Project 2:

Route Management System

PROBLEM STATEMENT

It identifies internal datastores of payment, bus route, customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule. It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL:



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:

  ★ Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.

  ★ Spiral Model consists of Risk Analysis task region.

  ★ In this model, we systematically prepare prototype and release for customer feedback.

- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

Software Requirements Specifications

1. Goal of the Project:

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. Functional Requirements:

As per the requirements of the software from client end. By analysis we realise that we need five modules:

A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

-This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password.

B. Customer Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. Schedule Module:

This module maintains timings of the bus. This include function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

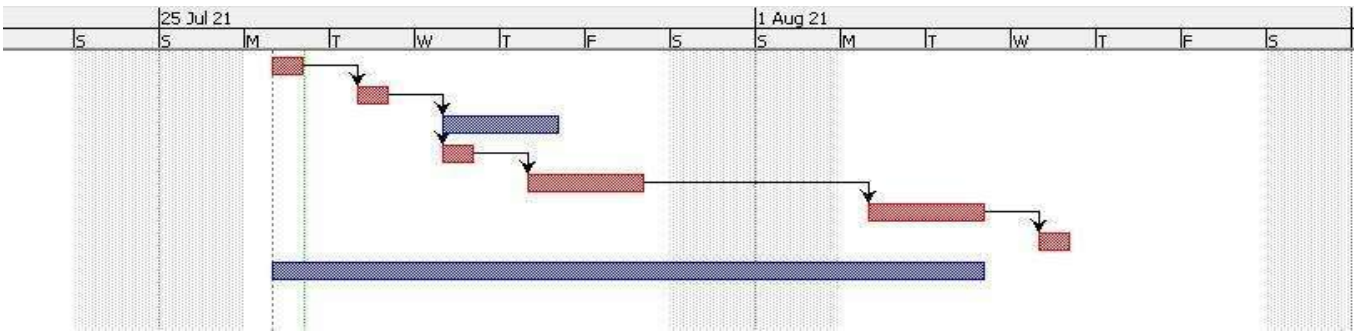
B. Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT:

a) Project Scheduling:-

	Name	Duration	Start	Finish	Predecessors
1	Probleam Statement	1 day	7/26/21 8:00 AM	7/26/21 5:00 PM	
2	Customer Commmunication	1 day	7/27/21 8:00 AM	7/27/21 5:00 PM	1
3	Planning	2 days	7/28/21 8:00 AM	7/29/21 5:00 PM	2
4	Risk Analysis	1 day	7/28/21 8:00 AM	7/28/21 5:00 PM	2
5	Engineering	2 days	7/29/21 8:00 AM	7/30/21 5:00 PM	4
6	Construction and Release	2 days	8/2/21 8:00 AM	8/3/21 5:00 PM	5
7	Consumer Evaluation	1 day?	8/4/21 8:00 AM	8/4/21 5:00 PM	6
8	Write User manual	7 days	7/26/21 8:00 AM	8/3/21 5:00 PM	



b. Project Resources

a. ManPower Resources:
   i. Project Manager
   ii. System Analyst
   iii. Designer
   iv. Developer
   v. Tester
   vi. Document Writer

b. Hardware-Software Resources
   i. Processor: Intel i3 4th gen or above
   ii. Ram: 4Gb or above
   iii. Java Development Kit - 13
   iv. Java Runtime Environment
   v. Sublime Text 3 (Text Editor)
   vi. Ms - Project
   vii. Create.ly
   viii. Google Docs

c. Risk Management Plan

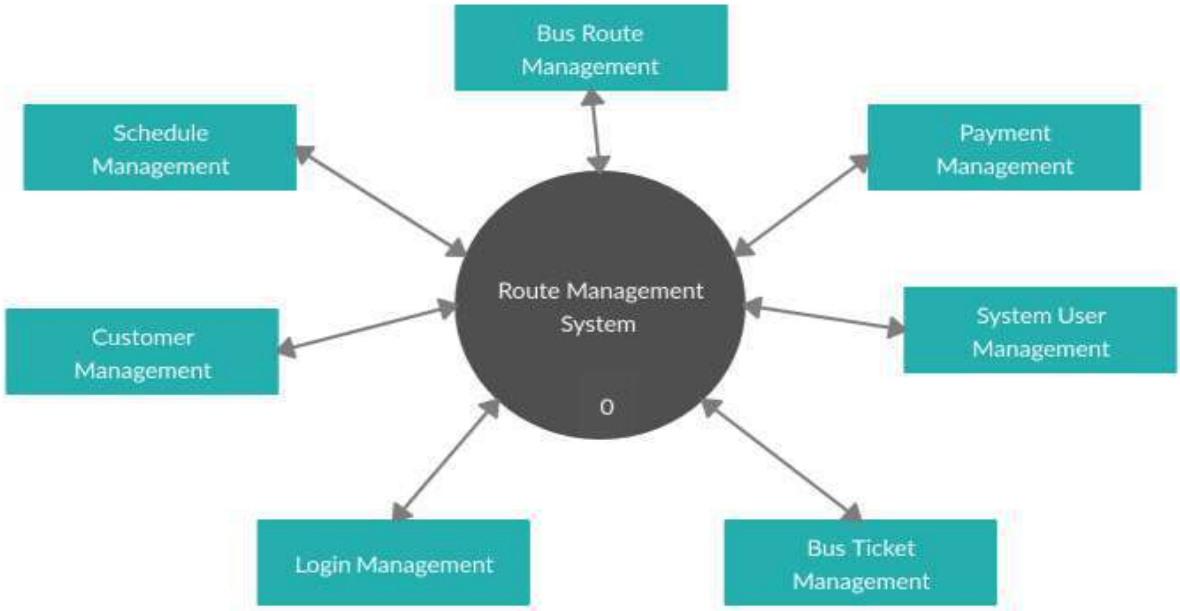
All risks are measured in terms of :

A. Likelihood of a risk coming through
B. Consequence of the problem associated with that risk Priority of each risk is multiplication of above two Risk Leverage =

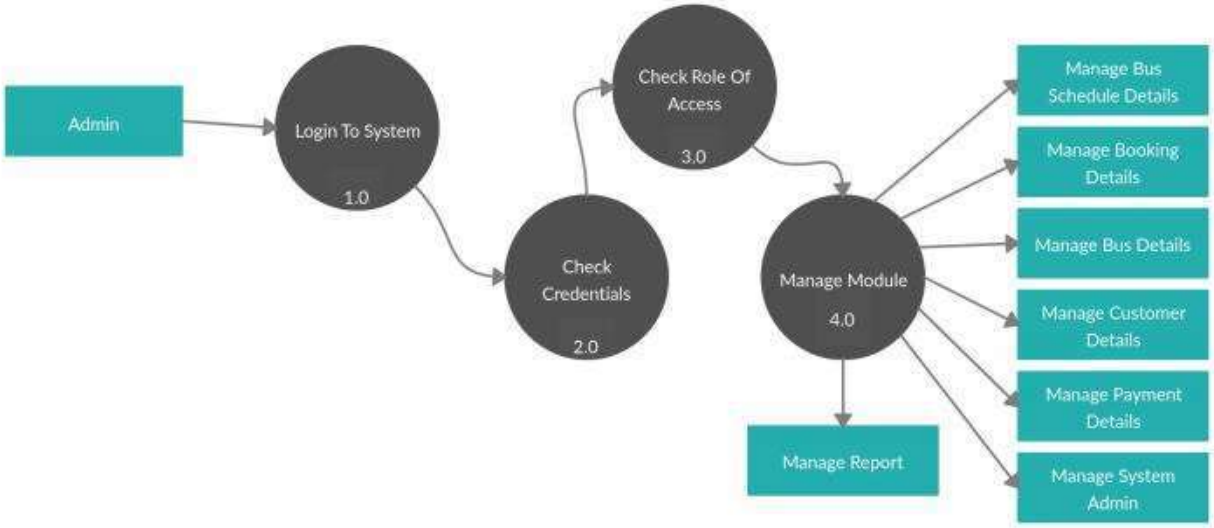
$$\frac{\textit{Risk exposure before Reduction} - \textit{Risk exposure after Reduction}}{\textit{Cost of Reduction}}$$

DESIGN ENGINEERING

1. *Architectural Design*



2. Component Level Design
   Level 1 DFD



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

   Each module is tested in isolation

2. System Testing:

   The modules are integrated and again tested. This time this testing is done in three parts

   A. Alpha testing(Verification) : done by Development team

   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.

   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

   Integration testing is done to verify the functional, performance,and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition.

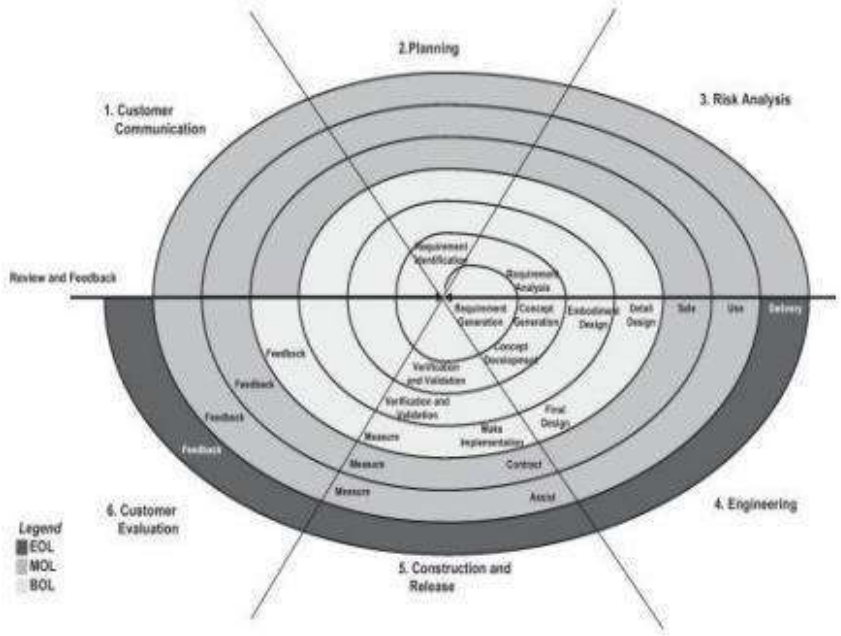
Project 2:

Route Management System

PROBLEM STATEMENT

It identifies internal datastores of payment, bus route, customer, ticket, ticket counter that must be present in order for the bus ticket system to do its job and shows the flow of data between the various parts of bus schedule. It also shows the details of ticket counter, bus route, payment and the customer of the system and finally the generate the report of the same.

PROCESS MODEL:



- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reason:
    - ★ Spiral Model couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into activities called task regions.
    - ★ Spiral Model consists of Risk Analysis task region.
    - ★ In this model, we systematically prepare prototype and release for customer feedback.
- Spiral Process model will be very helpful to create a criminal record management software system.

REQUIREMENT ANALYSIS

Software Requirements Specifications

1. Goal of the Project:

The software is made to manage the various datastores related to route, payment, schedule of bus. This would reduce the hectic job of bus authority to monitor reports.

2. Functional Requirements:

As per the requirements of the software from client end. By analysis we realise that we need five modules:

A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

-This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password.

B. Customer Module:

This module has function to monitor customer details like from and to place, distance between stops and accordingly calculates fare. It also has a function to generate tickets which has the above mentioned details.

C. Schedule Module:

This module maintains timings of the bus. This include function which returns bus timing and number if starting stop and destination stop is given input. It can also produce whole schedule if queried.

D. Route Module:

This module has a function to query about bus routes and eventually know the stops it would cover during its journey .It also has a function which makes drivers know their route.

E. Report Module:

At the end of the journey, the admin(driver or owner) asks a function of this module to generate a total report of the journey which includes duration of journey, tickets sold, graphical data of passengers.

3. Non-Functional Requirements:

A. Usability: The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

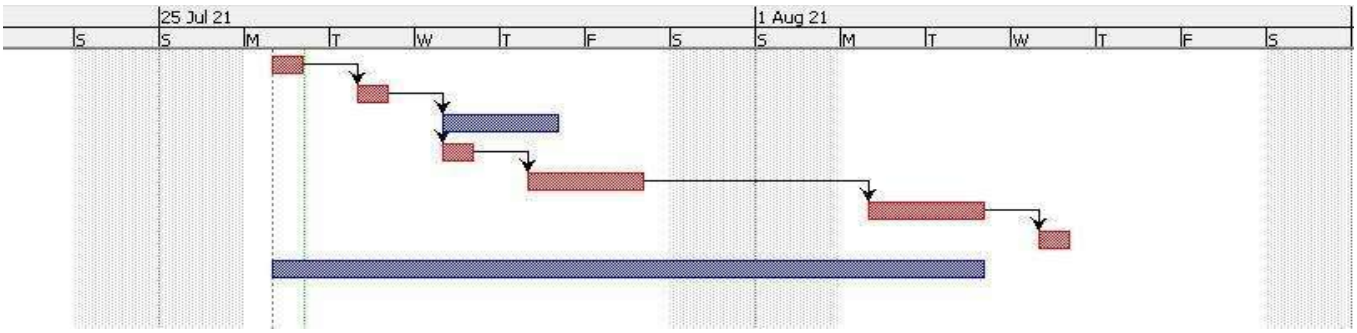
B. Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

C. Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT:

a) Project Scheduling:-

	Name	Duration	Start	Finish	Predecessors
1	Probleam Statement	1 day	7/26/21 8:00 AM	7/26/21 5:00 PM	
2	Customer Commmunication	1 day	7/27/21 8:00 AM	7/27/21 5:00 PM	1
3	Planning	2 days	7/28/21 8:00 AM	7/29/21 5:00 PM	2
4	Risk Analysis	1 day	7/28/21 8:00 AM	7/28/21 5:00 PM	2
5	Engineering	2 days	7/29/21 8:00 AM	7/30/21 5:00 PM	4
6	Construction and Release	2 days	8/2/21 8:00 AM	8/3/21 5:00 PM	5
7	Consumer Evaluation	1 day?	8/4/21 8:00 AM	8/4/21 5:00 PM	6
8	Write User manual	7 days	7/26/21 8:00 AM	8/3/21 5:00 PM	



b. Project Resources

a. ManPower Resources:
   i. Project Manager
   ii. System Analyst
   iii. Designer
   iv. Developer
   v. Tester
   vi. Document Writer

b. Hardware-Software Resources
   i. Processor: Intel i3 4th gen or above
   ii. Ram: 4Gb or above
   iii. Java Development Kit - 13
   iv. Java Runtime Environment
   v. Sublime Text 3 (Text Editor)
   vi. Ms - Project
   vii. Create.ly
   viii. Google Docs

c. Risk Management Plan

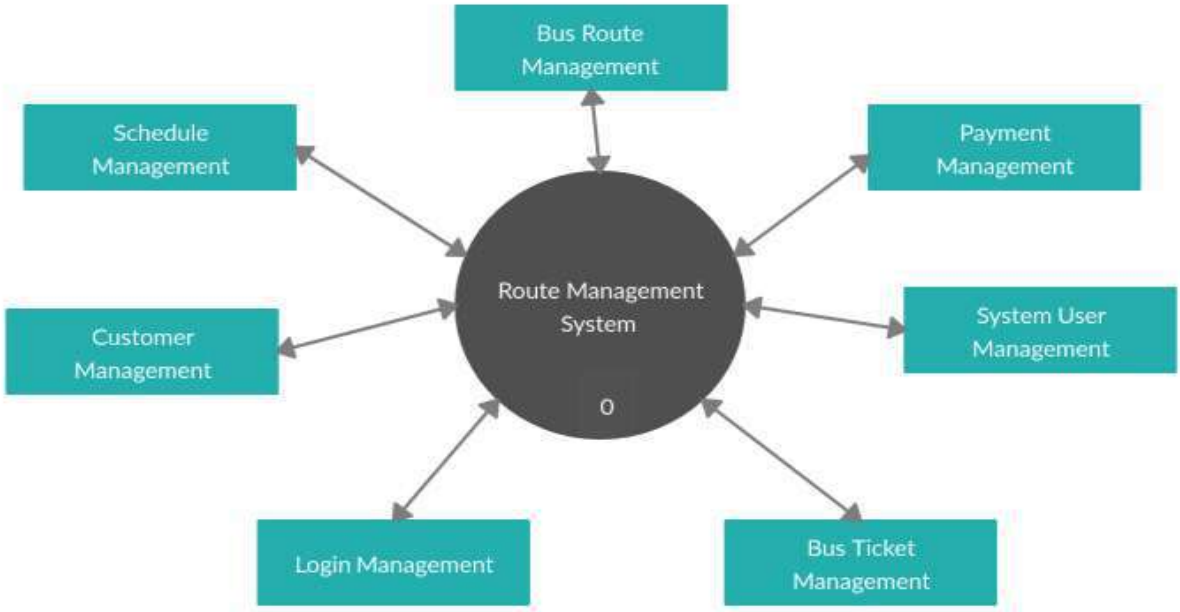
All risks are measured in terms of :

A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk Priority of each risk is multiplication of above two Risk Leverage =

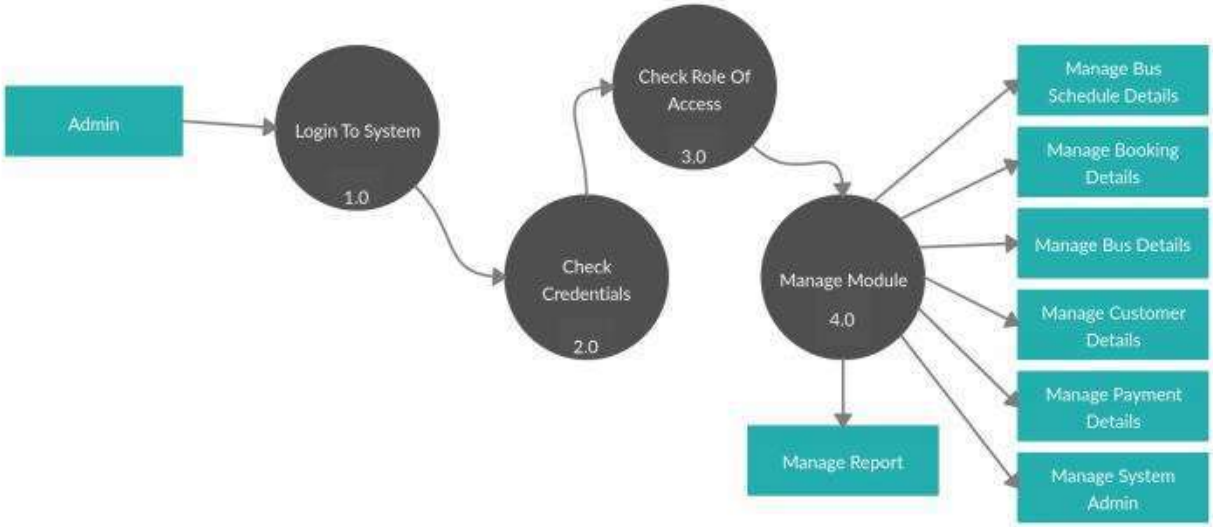
$$\frac{\textit{Risk exposure before Reduction} - \textit{Risk exposure after Reduction}}{\textit{Cost of Reduction}}$$

2. *Architectural Design*



2. Component Level Design

Level 1 DFD



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. Unit Testing:

   Each module is tested in isolation

2. System Testing:

   The modules are integrated and again tested. This time this testing is done in three parts

   A. Alpha testing(Verification) : done by Development team

   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.

   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. Integration Testing:

   Integration testing is done to verify the functional, performance,and reliability between the modules that are integrated.

MAINTENANCE

Maintenance stands for all modifications and updations done after the delivery of a software product. Types of maintenance done in this retail software is:

1. Corrective Maintenance
2. Adaptive Maintenance
3. Perfective Maintenance

It involves a combination of technical and administrative actions carried out to retain an item, equipment, system or machine in order to restore it to an acceptable working condition

PROJECT 3

PATIENT MANAGEMENT SYSTEM SOFTWARE

PROBLEM STATEMENT :

The system identifies internal data stores of report, doctors, disease, doctor schedule, test that must be present in order of the patient system to do its job. Administrator reports to the system and manages all the functionalities of the patient management software. Admin can edit, delete, view records of patients, test, disease and manage all the details of medicines, doctor schedule. Medicines, diseases, doctors can be scheduled by admin and also generates several reports.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realize that we need three modules.

- Goal of the project : The software is made to ease out the process for staff in hospitals by managing information related to patients. It will make it easier for doctors to monitor every particular patient admitted and accordingly prescribe medicines for every patient. Hospital Staff can also take care of medicines required and accordingly order before stock runs out in hospital.

- Functional requirements : The three modules needed are.
    - Administrator module : This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

    ○ Doctor module : This module accesses data from the database of doctor information and consists of functions which can update, insert or delete records from the database. Doctors in the list can also access this database.

    ○ Report module : This module has a function that can access the data from the patient database. Another function can generate reports for a patient or group of patients in a particular ward. All tests done for a particular patient are stored particularly to each patient's record.

    ○ Diagnostic module : This module monitors the medicine-info database. Admin, doctors can view, update, insert or delete medicine records. Doctors can ask for specific medicines to be bought. That request is recorded and the medicines are made available as early as possible.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.

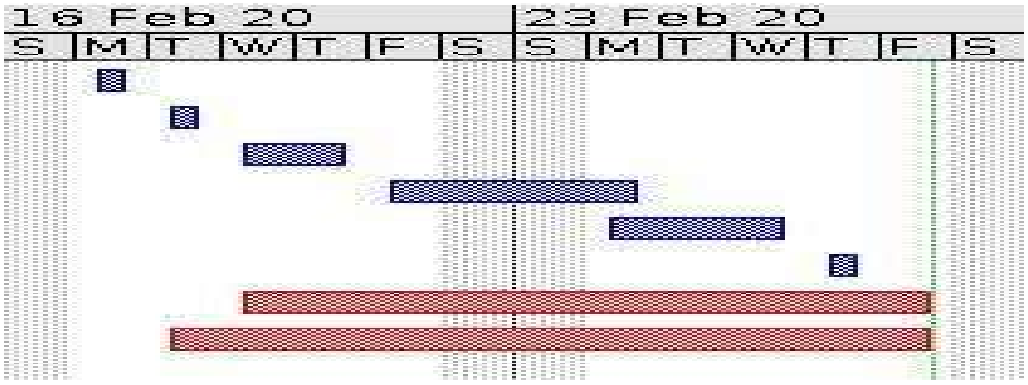
    ○ Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

    ○ Robustness : The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day	2/17/20, 8:00 AM	2/17/20, 5:00 PM
2	Feasibility study	1 day	2/18/20, 8:00 AM	2/18/20, 5:00 PM
3	Requirement analysis & specification	2 days	2/19/20, 8:00 AM	2/20/20, 5:00 PM
4	Design	2 days	2/21/20, 8:00 AM	2/24/20, 5:00 PM
5	Coding & unit testing	3 days	2/24/20, 8:00 AM	2/26/20, 5:00 PM
6	Integration & system testing	1 day	2/27/20, 8:00 AM	2/27/20, 5:00 PM
7	Maintainance	8 days	2/19/20, 8:00 AM	2/28/20, 5:00 PM
8	User manual	9 days	2/18/20, 8:00 AM	2/28/20, 5:00 PM



- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.

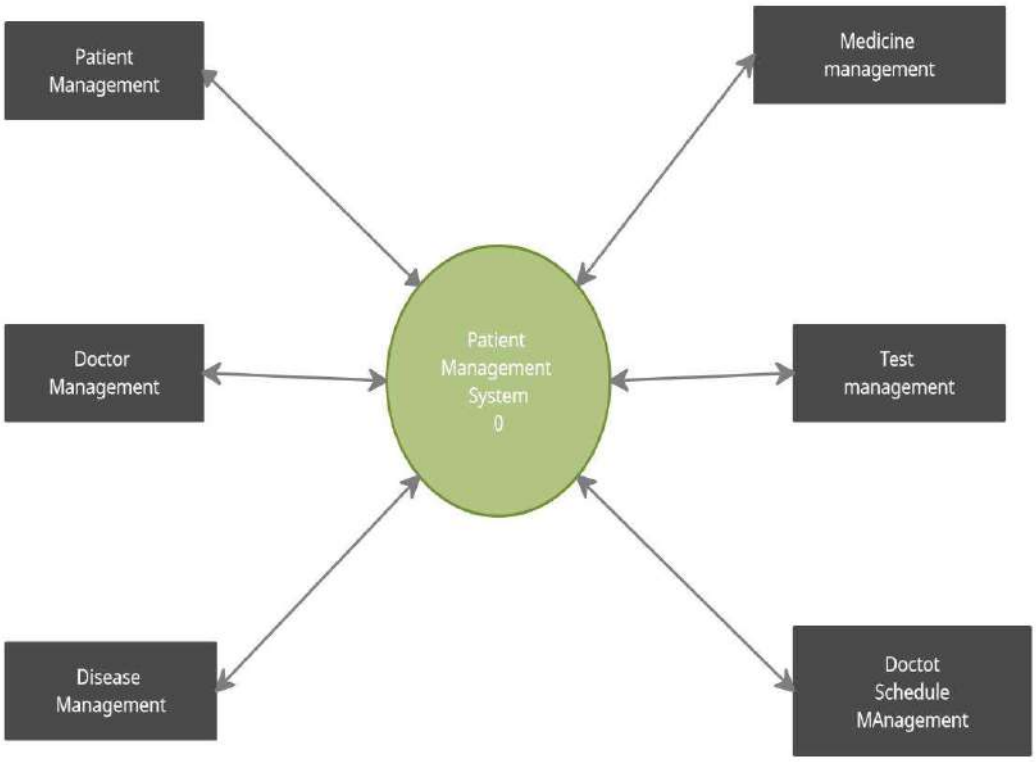
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

(Risk exposure before reduction - Risk exposure before reduction)

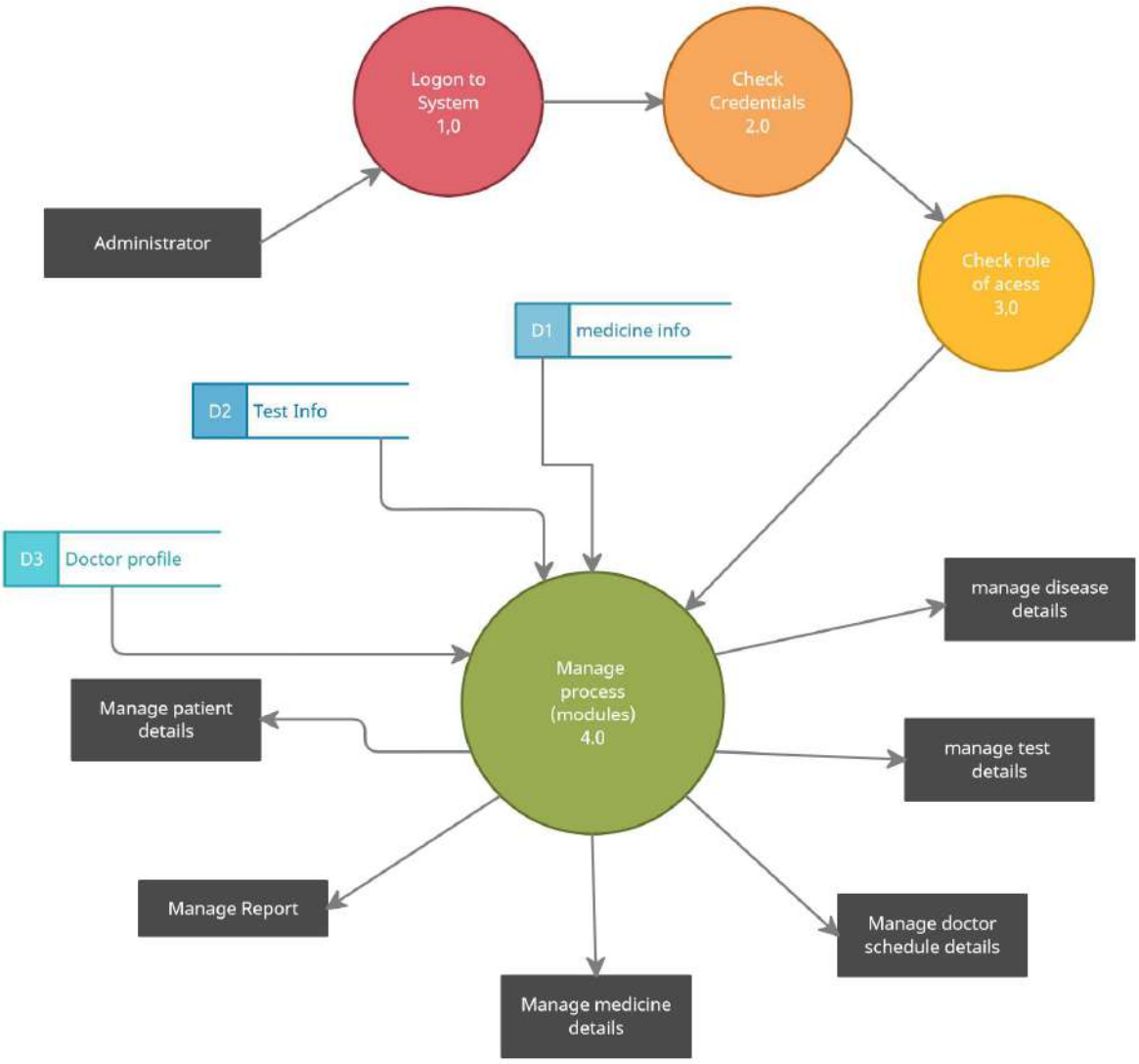
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Logon to System 1,0
Check Credentials 2.0
Administrator
Check role of acess 3,0
D1 medicine info
D2 Test Info
D3 Doctor profile
manage disease details
Manage process (modules) 4.0
Manage patient details
manage test details
Manage Report
Manage doctor schedule details
Manage medicine details

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 4

ONLINE HOTEL BOOKING SYSTEM SOFTWARE

PROBLEM STATEMENT:

Guests can reserve rooms in a hotel in advance or can reserve on the spot. The operator would enter data pertaining to guests such as their arrival time, advance paid, approximate duration of stay and type of room required. The computer should register this data and depending on the availability of the suitable room, the computer should provisionally allot a room number to the guest and a unique token no. to the guest. If the guest cannot be accommodated, the computer generates an apology message. The hotel catering services manager would input the quantity and type of food items consumed by the guest along with the token no. of the guest and the corresponding date and time. When a customer prepares to check out the hotel, automation software should generate the entire bill for the customer and balance amount payable.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end. By analysis we realise that we need three modules.

1. **Guest module:** In this module we add a function which will take the type of room required by the guest as input and search if such room is available for accommodation or not. If available, guest details are taken and stored in the database. A unique token no. is also generated as output for guest bill identification. If the room required by a guest is not available, an apology message is given output by software.
2. **Stay module:** In this module we add a function for updating expenses made by guests in the hotel while staying. It takes a unique token no. and expended amount as input. The function

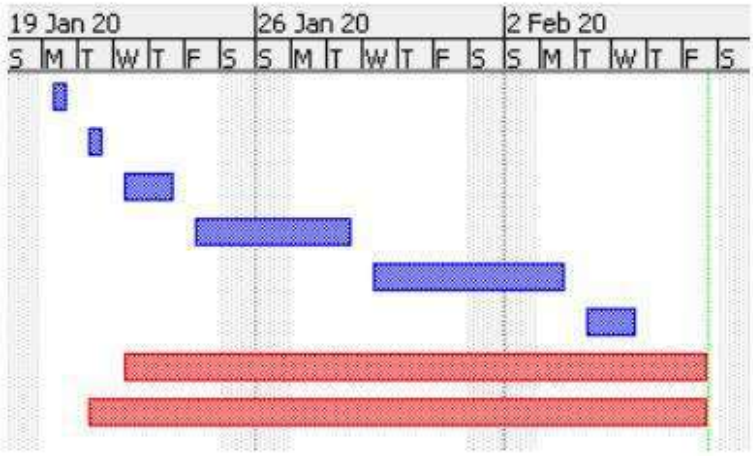
searches record with that token no. and updates the amount expended. Another function keeps track of stayed days by guest accordingly charges for room.

3. **Catering module:** In this module, a function should be made to order food by the hotel catering service manager when told by a guest. The guest tells what he would eat and his unique token no.. Hotel catering Service Manager uses special code for the food to update guest's bills by identifying records with that token no.
4. **Bill generation module:** At the time of checkout of a guest, a function takes token no. as input and searches the lodging database and catering database for the expenses done by guest during staying in hotel. All the expenses and details related to that are given output as a bill. After bill payment, the record for that guest is cleared.

PROJECT MANAGEMENT :

- **Project Scheduling :**

		Name	Duration	Start	Finish
1		Problem statement	1 day?	20/1/20 8:00 AM	20/1/20 5:00 PM
2		Feasibility study	1 day?	21/1/20 8:00 AM	21/1/20 5:00 PM
3		Requirement analysis & specification	2 days?	22/1/20 8:00 AM	23/1/20 5:00 PM
4		Design	3 days?	24/1/20 8:00 AM	28/1/20 5:00 PM
5		Coding & unit testing	4 days?	29/1/20 8:00 AM	3/2/20 5:00 PM
6		Integration & system testing	2 days?	4/2/20 8:00 AM	5/2/20 5:00 PM
7		Maintainance	13 days?	22/1/20 8:00 AM	7/2/20 5:00 PM
8		User manual	14 days?	21/1/20 8:00 AM	7/2/20 5:00 PM



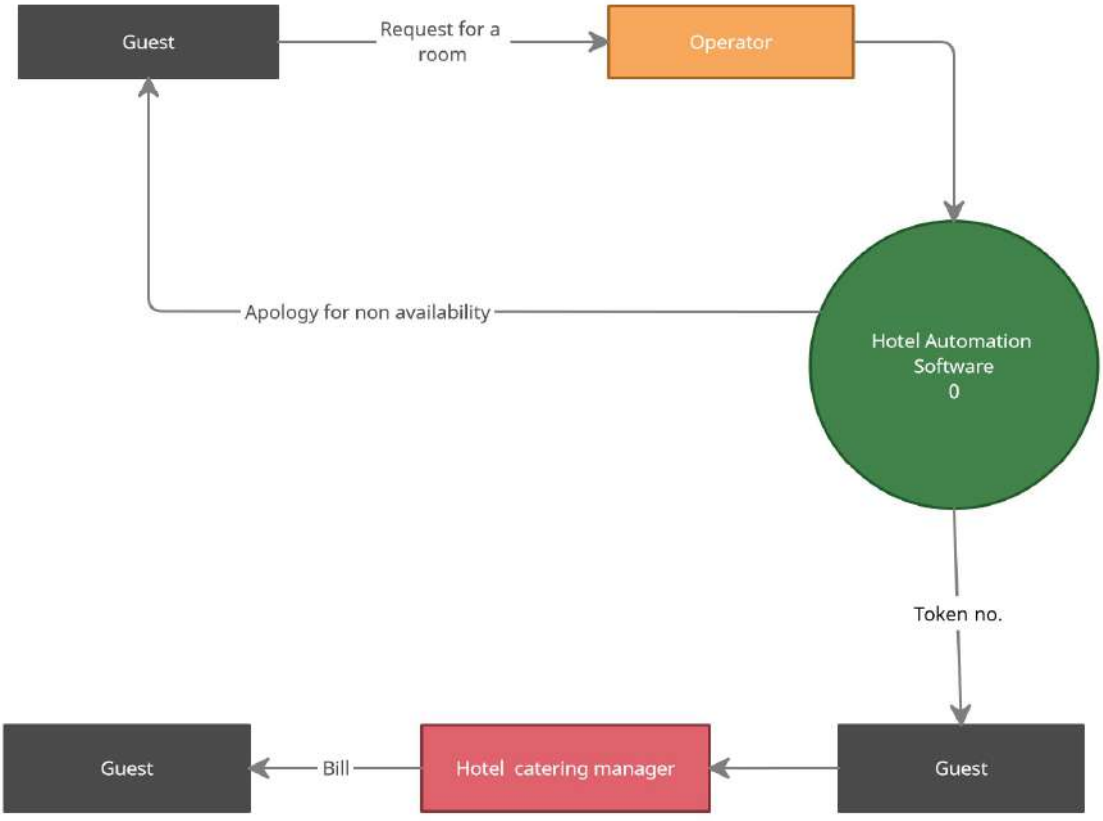
- **Project Resources :**
    - **Hardware Resources:** Several computer machines.
    - **Human Resources:** Manpower resources like project manager, designer, analysist, programmer, tester etc.
- **Risk management plan:** Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk.
        - Priority of risk is the multiplication of above two.

    - Risk leverage =

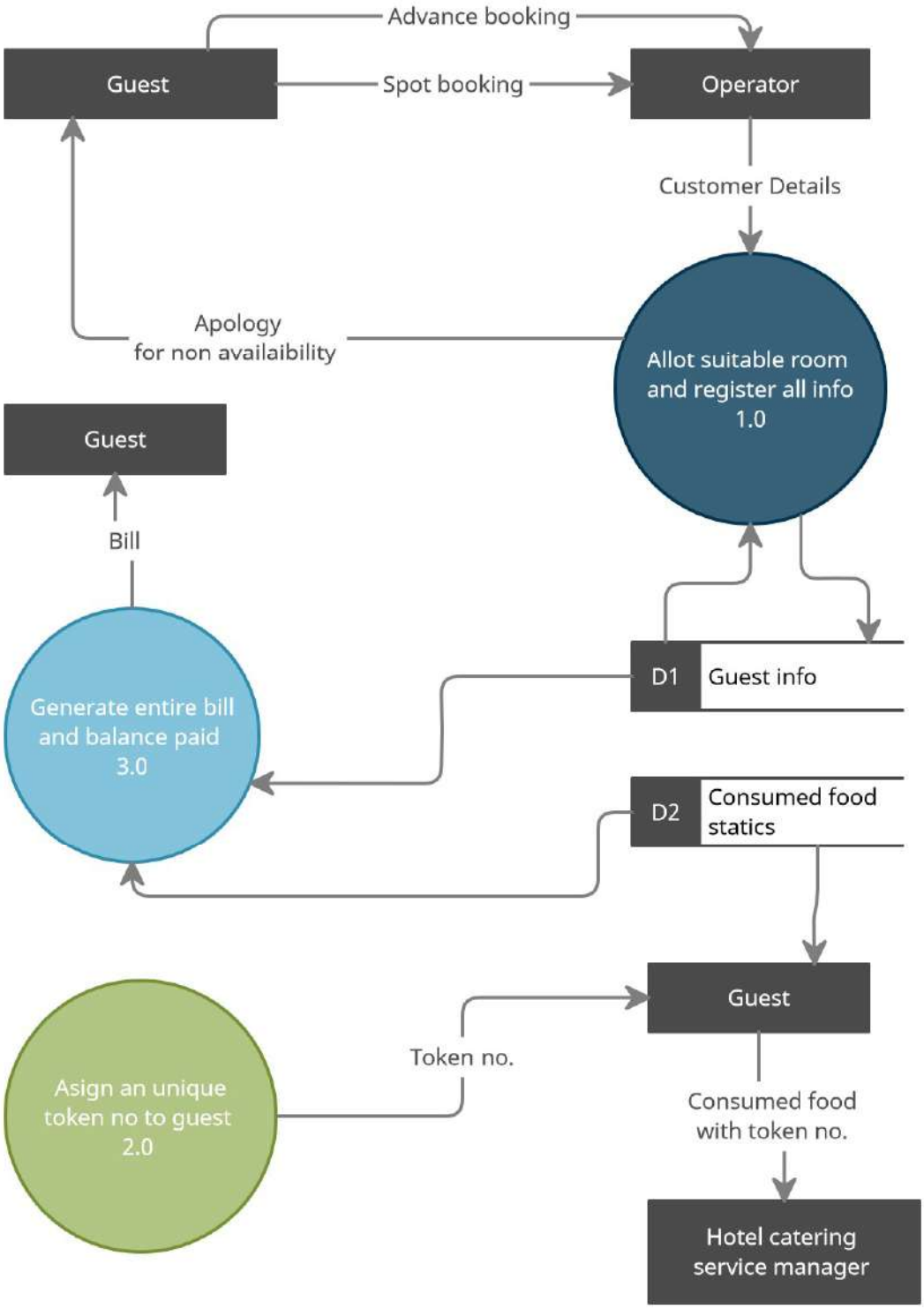
    (Risk exposure before reduction - Risk exposure before reduction)/ (Cost of reduction)

Design Engineering : Context Diagram

1. LEVEL 0 DFD:



Level 1 DFD :


Advance booking
Guest
Spot booking
Operator
Customer Details
Apology
for non availaibility
Allot suitable room
and register all info
1.0
Guest
Bill
Generate entire bill
and balance paid
3.0
D1
Guest info
D2
Consumed food
statics
Guest
Asign an unique
token no to guest
2.0
Token no.
Consumed food
with token no.
Hotel catering
service manager

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested. This time this testing is done in three parts.
    - Alpha testing(Verification) : It is done by the development team.
    - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
    - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. Hotel maintenance is the performance of adaptive, preventive and corrective maintenance for a given hotel facility. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 5

CRIMINAL RECORD MANAGEMENT SOFTWARE

PROBLEM STATEMENT:

It shows how the crime file system is divided into sub-system and each of which deals with data flows to or from external agents. The system identifies the processing of crime records, complaint records, FIR records, charge-sheet records and generates the report of the same. It then processes criminal records and produces to court and fully generates reports to all courts.

PROCESS MODEL:

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons:
    - Appropriate for exploratory projects that are working in an unfamiliar domain or with unproven technical approaches. The iterative nature allows for knowledge gained during early passes to inform subsequent passes.
    - Requires low up-front commitment
        - The spiral model uses the approach of Prototyping Model by building a prototype at the start of each phase as a risk handling technique. Also the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.

REQUIREMENT ANALYSIS (SRS):

As per the requirements of the software from client end by analysis we realize that we need three modules.

- Goal of the project: The software is made to ease out the process for managing crime reports. The main objective of this project is to improve the crime reporting system by using computers and reducing man power, time.

- Functional requirements: The three modules needed are.
    - Administrator module: This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned.

      ○ Criminal module: This module accesses data from the database of criminal record details and manages it properly.

      ○ FIR module: This module accesses data from the database of FIR record details and manages it properly by issuing complaints when needed.

      ○ Charge-sheet module: This module accesses data from the database of charge-sheet record details and manages it properly.

      ○ Court module: The main function of this module is to manage the court profiles.

- Non-Functional Requirements:
    - Usability: The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.

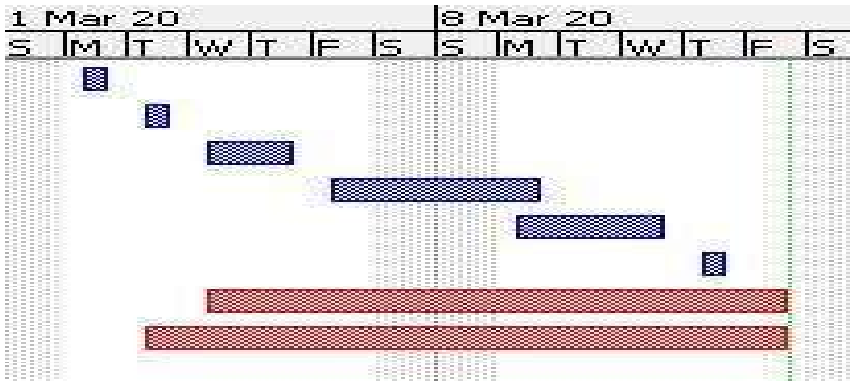
      ○ Portability: The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

      ○ Robustness: The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

PROJECT MANAGEMENT:

- Project Scheduling:
-

	Name	Duration	Start	Finish
1	Problem statement	1 day	3/2/20 8:00 AM	3/2/20 5:00 PM
2	Feasibility study	1 day	3/3/20 8:00 AM	3/3/20 5:00 PM
3	Requirement analysis & specification	2 days	3/4/20 8:00 AM	3/5/20 5:00 PM
4	Design	2 days	3/6/20 8:00 AM	3/9/20 5:00 PM
5	Coding & unit testing	3 days	3/9/20 8:00 AM	3/11/20 5:00 PM
6	Integration & system testing	1 day	3/12/20 8:00 AM	3/12/20 5:00 PM
7	Maintainance	8 days	3/4/20 8:00 AM	3/13/20 5:00 PM
8	User manual	9 days	3/3/20 8:00 AM	3/13/20 5:00 PM



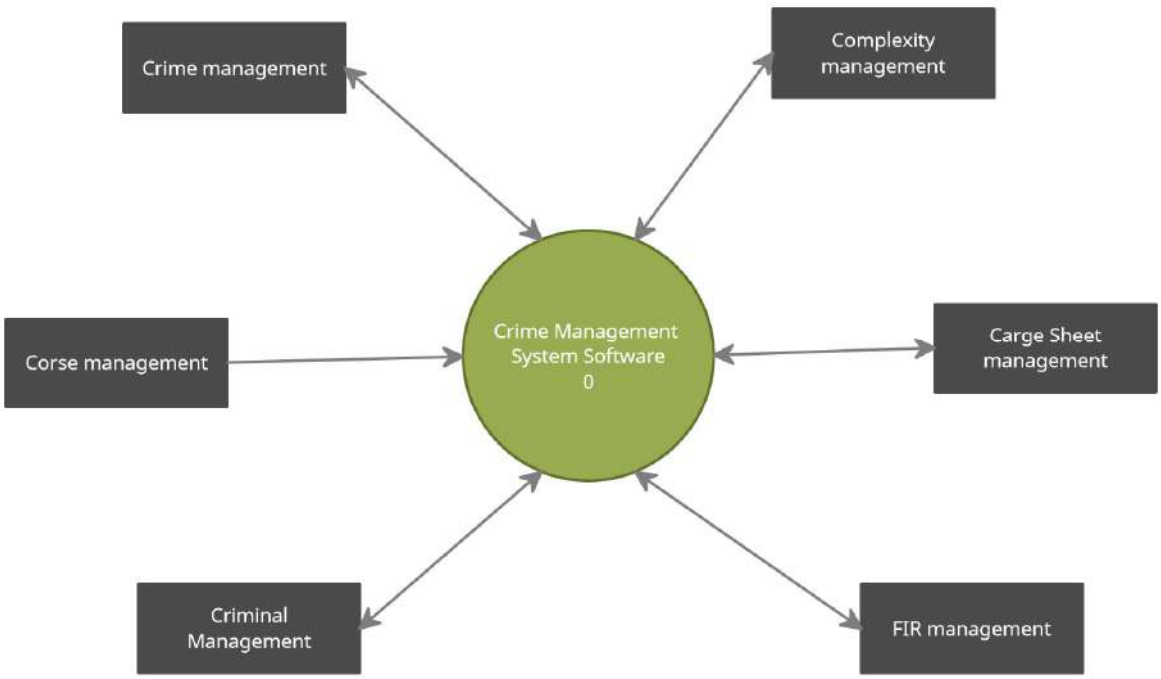
- Project Resources:
    - Hardware Resources: Several computer machines.
    - Human Resources: Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan: Risks are measured by 2 parameters.
    - Likelihood of a risk coming through (Probability of occurrence).
    - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.

- Risk leverage =

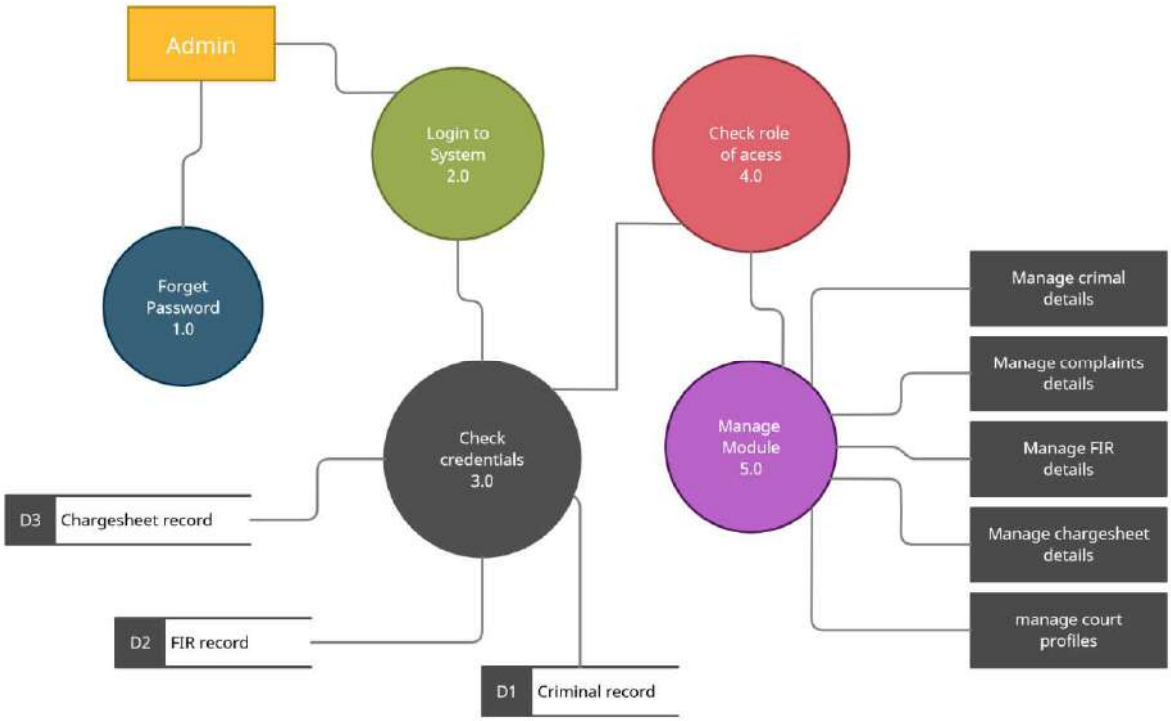
(Risk exposure before reduction - Risk exposure before reduction)
_____
(Cost of reduction)

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Admin
Login to System 2.0
Check role of acess 4.0
Forget Password 1.0
Check credentials 3.0
Manage Module 5.0
Manage crimal details
Manage complaints details
Manage FIR details
Manage chargesheet details
manage court profiles
D3 Chargesheet record
D2 FIR record
D1 Criminal record

TESTING :

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing : Each module is tested in isolation.
- System Testing : The modules are integrated and again tested.

  This time this testing is done in three parts.

  - Alpha testing(Verification) : It is done by the development team.
  - Beta testing (Validation) : It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing : It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE :

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

PROJECT 6

EXAMINATION AND RESULT COMPUTATION SOFTWARE

PROBLEM STATEMENT:

This software identifies internal data stores of subject, course, branch, student, class that must be present in order for the examination system to do its job. It highlights the flow of data between various parts of time table, class, course, branch of the system.

Administrator logins to the system and manages all functionalities like adding, editing, deleting and viewing the record of all time tables. Admin can manage all the details of examination and search the details of examination, it can also track the detailed information of result computation of students.

PROCESS MODEL :

- Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.
- For developing this software, we will use Spiral Model for the following reasons :
    - Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.
    - Spiral Model consists of risk analysis task region.
    - In this model, we systematically prepare prototype and release for customer feedback.

REQUIREMENT ANALYSIS (SRS) :

As per the requirements of the software from client end by analysis we realize that we need three modules.

- Goal of the project : The aim and objectives for students' examination

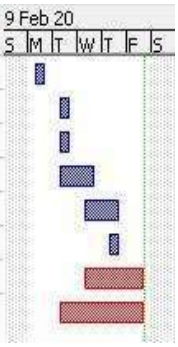
  Result is to study and document all processes involved in the task of generating students examination Result and eliminate errors due to manual processing. It is also to enhance the speed of the results. Finally to keep accurate records of students examination Results in the school and prevent loss of result, which are vital to the exams and records.

- Functional requirements : The three modules needed are.
    - Login and recovery module : This module is created for client-end result accessing. We add a function to take the user id and password as input and gives access to the data. We add another function which takes care of the situation when the user forgets his / her password. This module then sends an email to the user's email id for password recovery.
    - Credentials module : This module checks the credentials and is responsible for storing and accessing data from the data stores of exam and student information.
    - Administrator module : This module is used to monitor all the data and processes. It checks the role of access and manages all the details like course, examination, branch, class, student details. This module also manages the time table details by adding, editing, deleting and viewing the record of all time tables. It also tracks the detailed information of result computation of students.
- Non-Functional Requirements :
    - Usability : The software should have a user friendly interface for ease of access. It ensures more users to get indulged by the interface.
    - Portability : The software should have all modules encapsulated in such a manner that uses memory efficiently and takes optimal storage for being portable.

PROJECT MANAGEMENT :

- Project Scheduling :

	Name	Duration	Start	Finish
1	Problem statement	1 day?	10/2/20 8:00 AM	10/2/20 5:00 PM
2	Feasibility study	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
3	Requirement analysis & specification	1 day?	11/2/20 8:00 AM	11/2/20 5:00 PM
4	Design	2 days?	11/2/20 8:00 AM	12/2/20 5:00 PM
5	Coding & unit testing	2 days?	12/2/20 8:00 AM	13/2/20 5:00 PM
6	Integration & system testing	1 day?	13/2/20 8:00 AM	13/2/20 5:00 PM
7	Maintainance	3 days?	12/2/20 8:00 AM	14/2/20 5:00 PM
8	User manual	4 days?	11/2/20 8:00 AM	14/2/20 5:00 PM

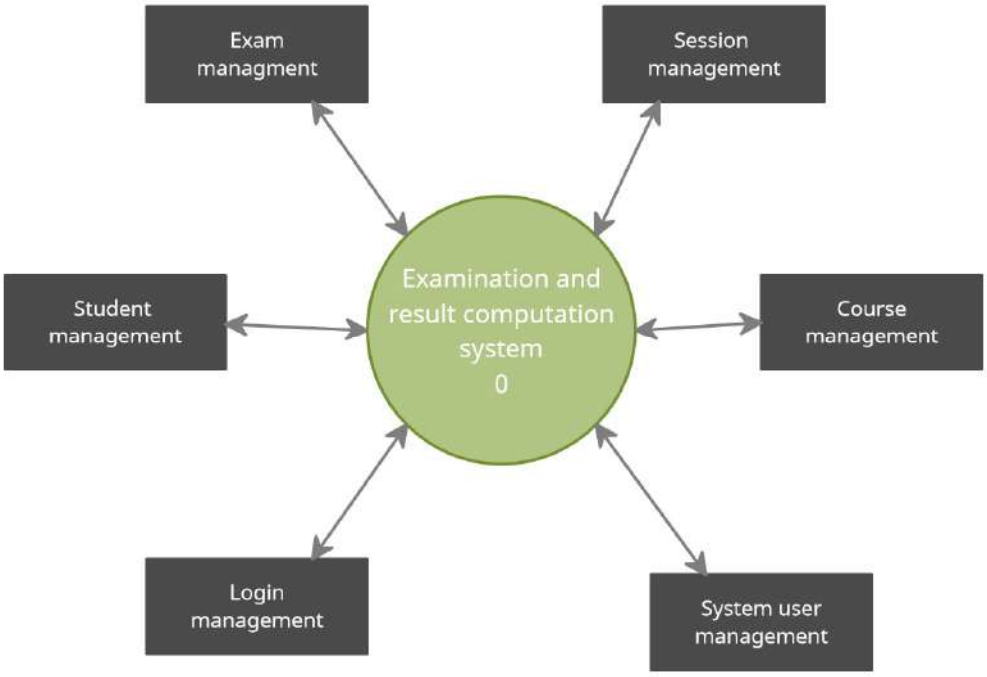


- Project Resources :
  - Hardware Resources : Several computer machines.
  - Human Resources : Manpower resources like project manager, designer, analysist, programmer, tester etc.
- Risk management plan : Risks are measured by 2 parameters.
  - Likelihood of a risk coming through (Probability of occurrence).
  - Consequence of the problem associated with that risk. ■ Priority of risk is the multiplication of above two.
  - Risk leverage =

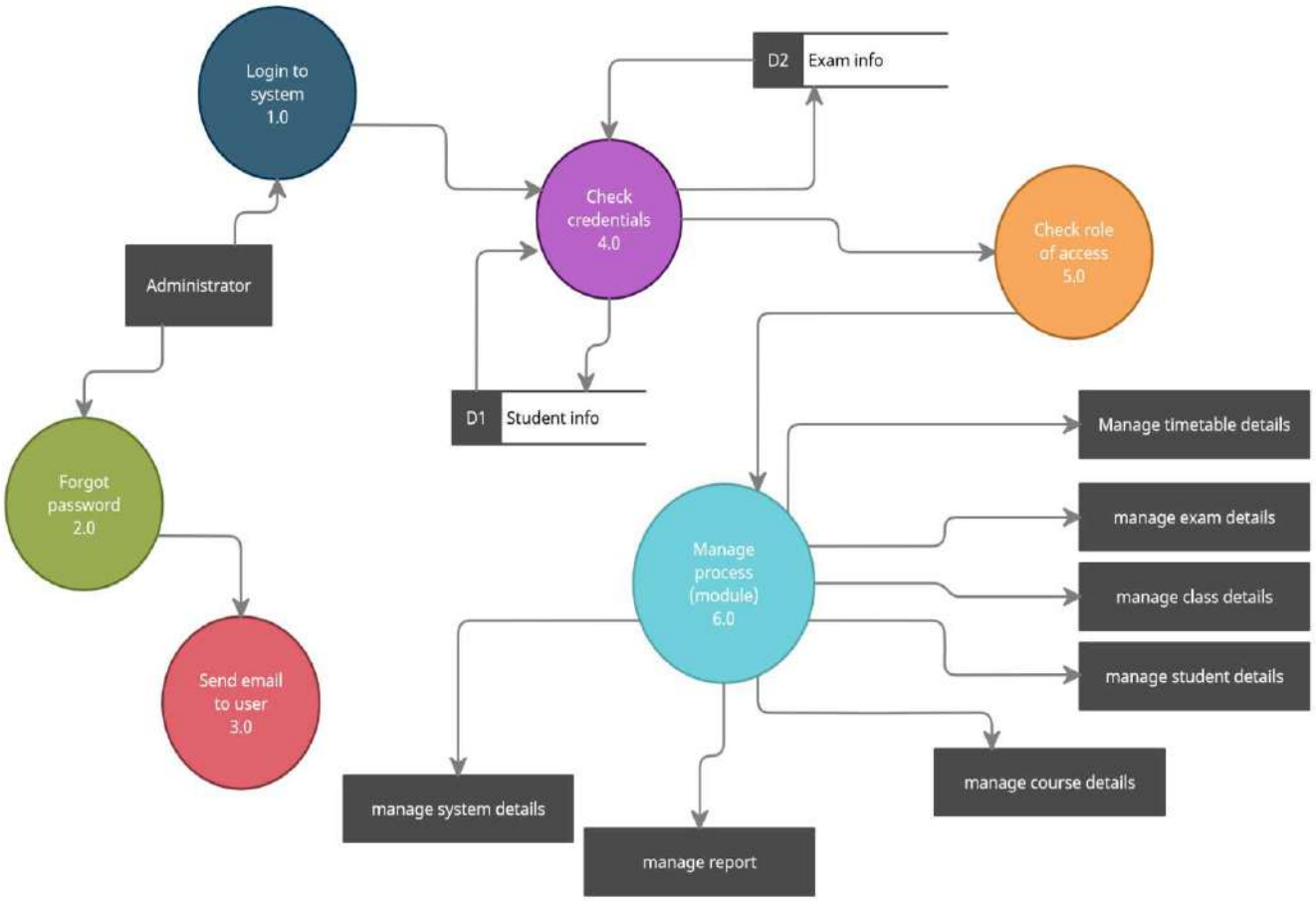
$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING : Context Diagram

1. Level 0 DFD :



2. Level 1 DFD :


Login to system 1.0
D2 Exam info
Check credentials 4.0
Check role of access 5.0
Administrator
D1 Student info
Forgot password 2.0
Send email to user 3.0
Manage process (module) 6.0
Manage timetable details
manage exam details
manage class details
manage student details
manage course details
manage system details
manage report

TESTING:

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways

- Unit Testing: Each module is tested in isolation.
- System Testing: The modules are integrated and again tested.

  This time this testing is done in three parts.

  - Alpha testing (Verification): It is done by the development team.
  - Beta testing (Validation): It is done by a special group of friendly customers in an uncontrolled environment.
  - Acceptance testing: It is performed by customers after product delivery to determine whether to accept or reject software.

MAINTENANCE:

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

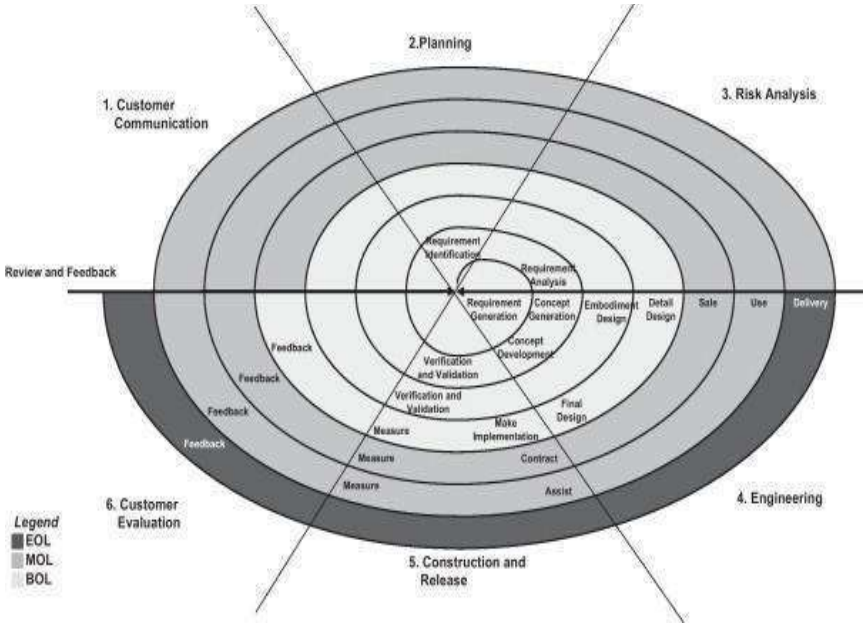
Project 7:

Parking Management System

PROBLEM STATEMENT

Parking Management System identifies internal datastores of login, customer, types, duration, parking fees. It shows the flow of data between the various parts of parking slots, parking fees etc. It processes parking slot records, vehicle records, parking fee record with duration and generates the report of the same.

PROCESS MODEL



- **Software process model is designed as a framework involving a small number of framework activities that are applicable to all software projects.**
- **For developing this software, we will use Spiral Model for the following reason:**

- ★ **Spiral Model couples iterative nature of prototyping with systematic aspect of waterfall model. It is derived into activities called task regions.**
- ★ **Spiral Model consists of Risk Analysis task region.**
- ★ **In this model, we systematically prepare prototype and release for customer feedback.**

- **Spiral Process model will be very helpful to create a criminal record management software system.**

REQUIREMENT ANALYSIS

➔ **Software Requirements Specifications**

1. Goal of the Project:

The software is made to ease out the process for employees managing parking space by managing information related to vehicles. It will make it easier for the person to monitor every vehicle in parking space. The software reduces a big tension of maintaining datastores of login, customer, types, duration parking fees etc. which was a hefty work.

2. Functional Requirements:

As per the requirements of the software from client end.

By analysis we realize that we need three modules:

A. Admin Module:

This module asks username and password as input and if the input is correct gives access to the system. Every admin had his/her role of access assigned. Only one admin can make more admin accounts for them to access data in software.

Input: UserName and Password

This module also ensures if admin forgets password, the software asks security questions. If the answer matches, it mails to the registered Mail ID for a new password.

Input: Answer to security question

Output: Link for changing password

B. Vehicle Module:

This module updates, inserts or deletes crime related information from respective databases. This module includes functions to access data from databases. Admin can see about a customer's vehicle details, duration of parking and accordingly calculate fee to be paid.

C. **Parking Module:**

This module monitors which parking slots are free and which are occupied and accordingly assign to a particular type of vehicle. The function also gives a parking no. to identify where his/her vehicle is parked.

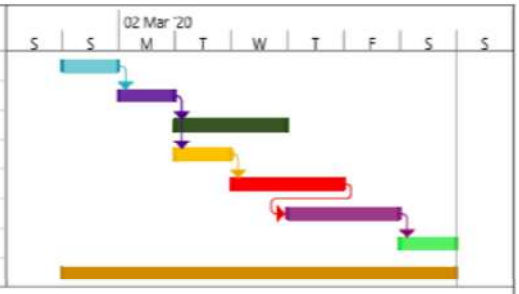
3. ***Non-Functional Requirements:***

A. **Usability:** The software should have an user friendly interface for ease of access. It ensures more customers to get indulged by the interface.

B. **Portability:** The software should have all modules encapsulated in such a manner that uses memory e ciently and takes optimal storage for being portable.

C. **Robustness:** The software should be prepared for possible errors and exceptions in such a way that it has low frequency of failure or takes less recovery time.

ID	Task Name	Duration	Start	Finish	Predecessors
1	Problem Statement	1 day	Sun 01-03-20	Sun 01-03-20	
2	Customer Communication	1 day	Mon 02-03-20	Mon 02-03-20	1
3	Planning	2 days	Tue 03-03-20	Wed 04-03-20	2
4	Risk Analysis	1 day	Tue 03-03-20	Tue 03-03-20	2
5	Engineering	2 days	Wed 04-03-20	Thu 05-03-20	4
6	Construction and Release	2 days	Thu 05-03-20	Fri 06-03-20	5
7	Customer Evaluation	1 day	Sat 07-03-20	Sat 07-03-20	6
8	Write User Manual	7 days	Sun 01-03-20	Sat 07-03-20	



PROJECT MANAGEMENT

a. Project Scheduling

b. Project Resources

a. **ManPower Resources:**

i. Project Manager

ii. System Analyst

iii. Designer iv. Developer v. Tester

vi. **Document Writer**

b. **Hardware-Software Resources**

i. **Processor: Intel i3 4th gen or above**

ii.Ram: 4Gb or above

iii. Java Development Kit 13

iv.Java Runtime Environment

v.Sublime Text 3 (Text Editor)

vi. Ms - Project

vii. Creately viii. Google Docs.

c. **Risk Management Plan:**

All risks are measured in terms of

A. Likelihood of a risk coming through

B. Consequence of the problem associated with that risk Priority of each risk is multiplication of above two.

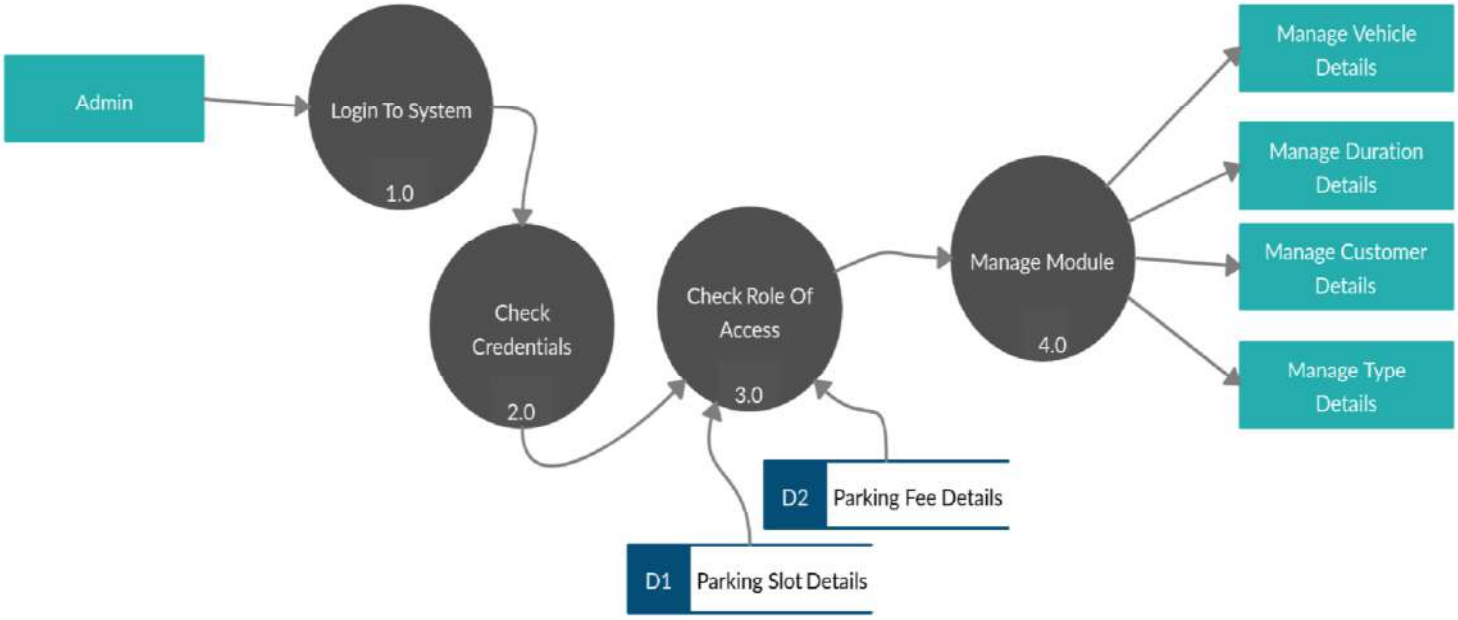
- Risk leverage =

$$\frac{\text{(Risk exposure before reduction - Risk exposure before reduction)}}{\text{(Cost of reduction)}}$$

DESIGN ENGINEERING

Level 1 DFD:

Level-2 DFD



TESTING

Testing is a very essential part before software is released to the clients or in the market. Testing is done in two ways:

1. **Unit Testing:**

   Each module is tested in isolation

2. **System Testing:**

   The modules are integrated and again tested. This time this testing is done in three parts

   A. Alpha testing(Verification) : done by Development team
   B. Beta testing (Validation): done by a special group of friendly customers in an uncontrolled environment.
   C. Acceptance testing: performed by customers after product delivery to determine whether to accept or reject software.

3. **Integration Testing:**

   Integration testing is done to verify the functional, performance, and reliability between the modules that are integrated.

MAINTENANCE:

Maintenance stands for all modifications and updations done after the delivery of software products. It involves a combination of technical and administrative actions carried out to retain an item, equipment, system, plant or machine in order to restore it to an acceptable working condition.

Project 8:

Wholesale Management System

Problem Statement – It provides all the functionalities of the retail store management system as a whole. It identifies internal data stores of customer, billing counter, bill, retail price, product that must be present in order for the retail store system to do its job. Store records, product records, retail price records, bill records, billing counter records and customer records are processed and reports are generated of the same.

Process Model – Spiral development model

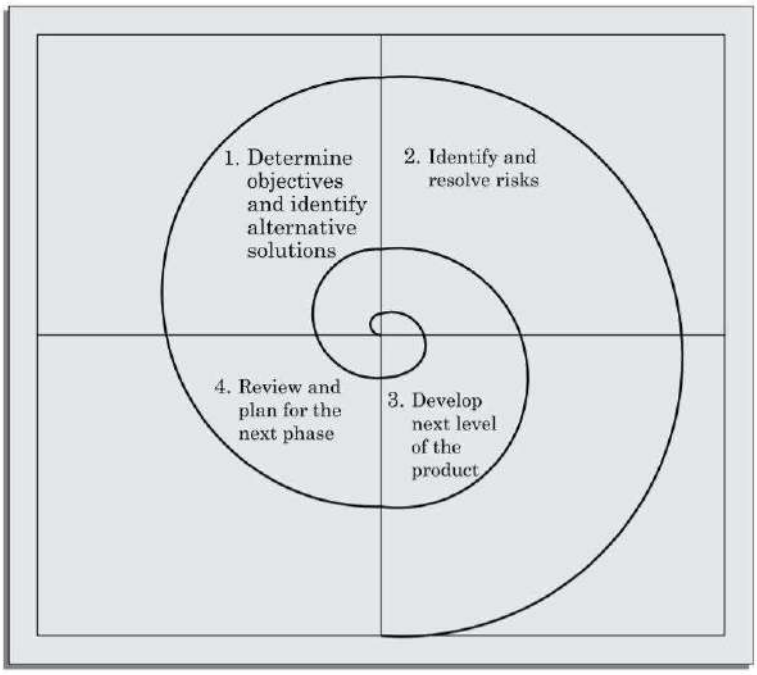


Fig – spiral model of software development

It couples the iterative nature of prototyping with the systematic aspect of waterfall model. It is derived into framework of activities also called task regions.

Normally there are 4 to 6 task regions. As this evolutionary process begins, the software engineering moves around the spiral in clockwise direction beginning at the core. Subsequent passes around the spiral model might be used to develop a prototype and then progressive sophisticated versions of prototypes.

Software Requirement Analysis

1. **Customer Module –** In this module, a customer is registered and the profile is maintained.

   Input: customer details
   Output: unique profile

2. **Stock Module –** In this module, the details of stocks are maintained.
   A database is maintained for keeping the stock information.

   Input: stock details
   Output: updated database

3. **Bill Module –** In this module, all the bills are maintained.
   A database is maintained for keeping the bill information.

   Input: customer details, stock details
   Output: generated bills, updated database

4. **Report Module –** In this module, store, product, retail price, bill, billing counter and customer records are processed and reports are generated.

   Input: store details, product details, retail price details, bill details, billing counter details, customer details
   Output: generated reports

Design Engineering

Data Flow Diagrams (DFDs)

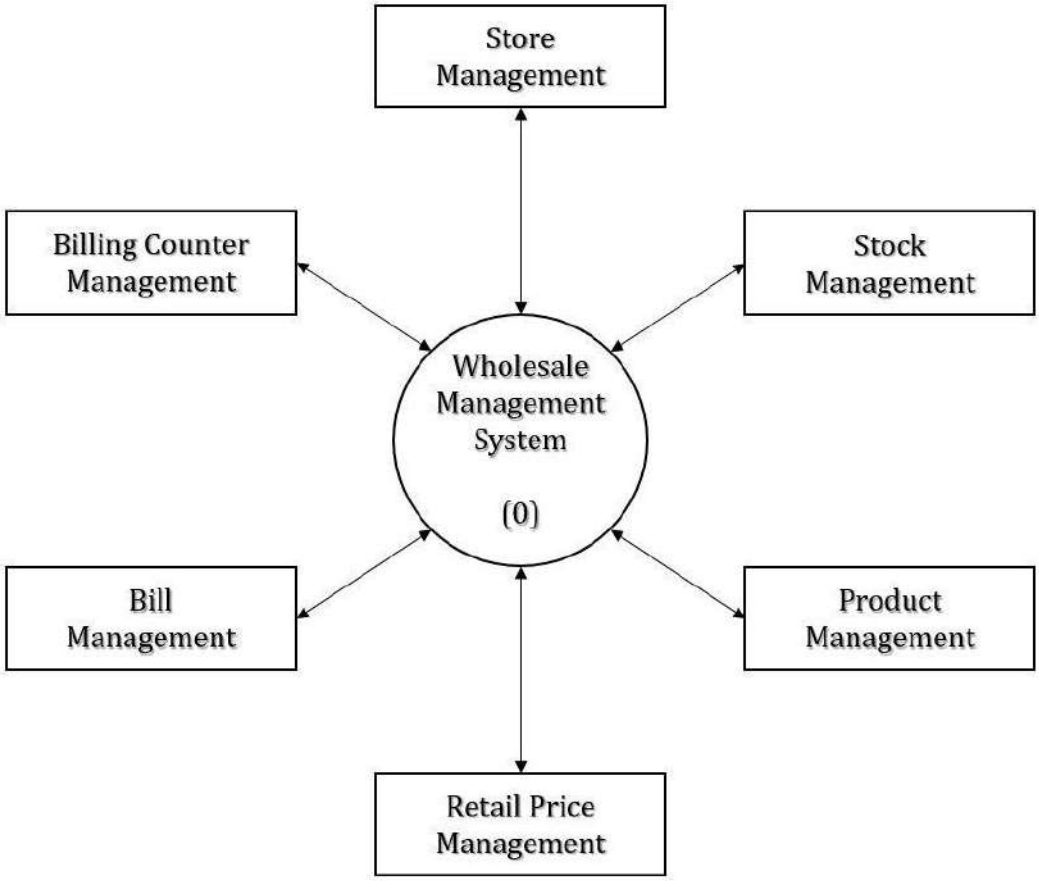


Fig – DFD level zero

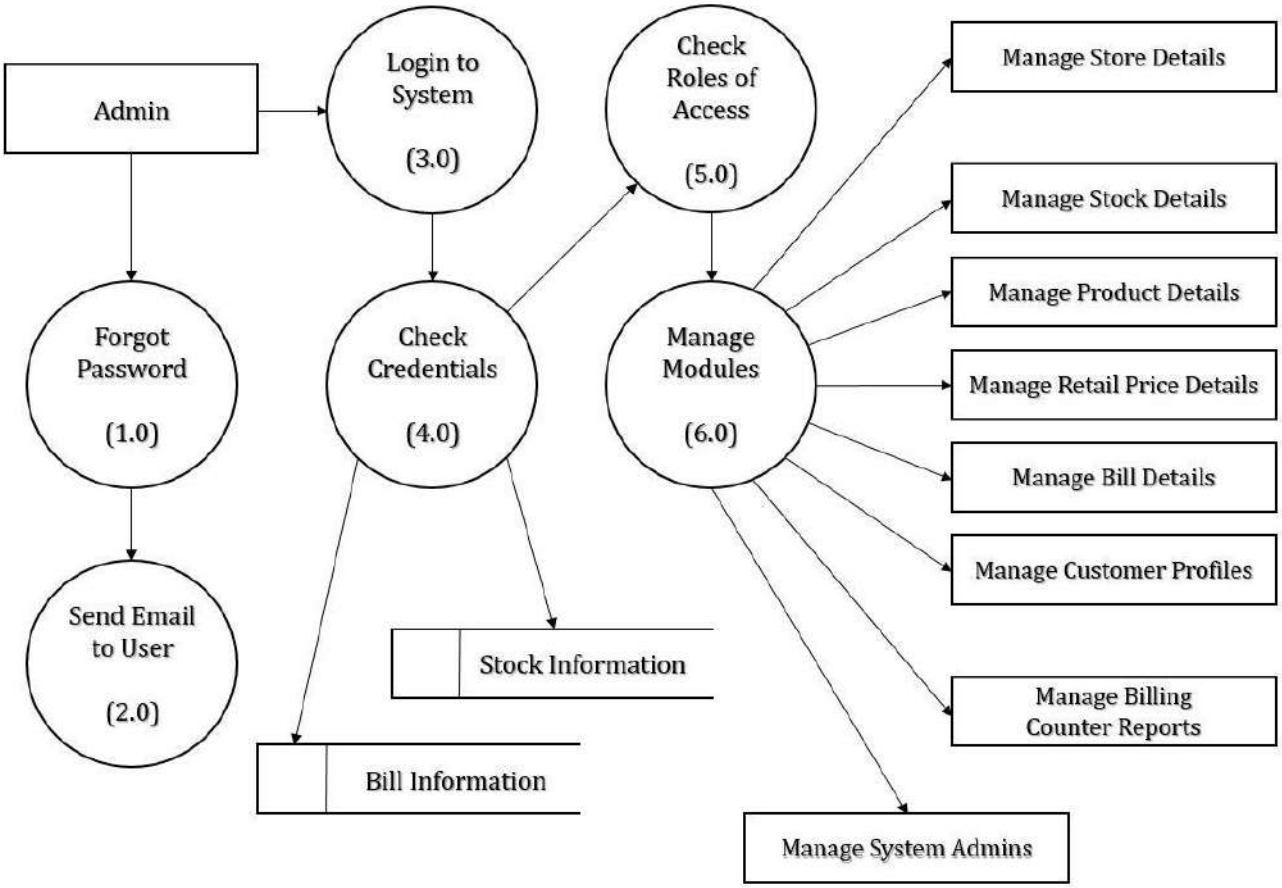


Fig – DFD level one

Project Management

Project Schedule – The file has been attached at the end.

Attached file – GanttWholesale.pdf

Risk Analysis

Risk Assessment – Each risk should first be rated in two ways.

i) The likelihood of a risk coming true (**r**)

ii) The consequences of the problems associated with the risk (**s**)

Based on these two factors, the priority of each risk can be computed as $p = r \times s$

where **p –** priority with which the risk must be handled

r – the probability of the risk becoming true

s – the severity of damage caused due to risk becoming true

$$risk\ leverage = \frac{risk\ exposure\ before\ reduction - risk\ exposure\ after\ reduction}{cost\ of\ reduction}$$

Testing

Unit Testing – Each component of the design is implemented as a program module. Each module is unit tested to determine the correct working of all individual modules.

Integration and System Testing – During the integration and system testing phase, the modules are integrated in a planned manner. System testing consists of three different kinds of testing activities.

i) **α (Alpha) Testing –** It is the system testing performed by the development team in a controlled environment.

ii) **β (Beta) Testing –** It is the system testing performed by friendly set of customers in an uncontrolled environment.

iii) **Acceptance Testing –** It is the system testing performed by the customers themselves after the product delivery to determine whether to accept or reject the delivered product.

Maintenance – The relative effort of development of a typical software product to its maintenance effort is in the ratio 40:60.

Maintenance involves three kinds of activities.

i) **Corrective maintenance –** Correcting errors that were not discovered during the product development phase.

ii) **Perfective maintenance –** Enhancing the functionalities of the system according to the customer's requirements.

iii) **Adaptive maintenance –** Porting the software to work in a new environment.

Name: DEBARPAN GOSWAMI

Department: Computer Science

Semester: V

Subject: Cloud Computing (DSE III)

Roll: 1738

Reg: A01-1112-117-009-2018

Case Study on

AWS Elastic

Cloud Computing

Date Created: 4-01-2021

Supervisor's Name: BB SIR

Contents

What is Cloud Computing?

Cloud computing is the delivery of computing services including servers, storage, databases, networking, software, analytics, and intelligence over the Internet to offer faster innovation, flexible resources, and economies of scale. We can typically pay only for cloud services we use, helping us lower our operating costs, run our infrastructure more efficiently, and scale as our business needs change.

There are certain services and models working behind the scene making the cloud feasible and accessible to end users. Following are the working models of cloud computing:

- Deployment Model
- Service Model

Deployment Models

Public Cloud

Public cloud is an IT model where on-demand computing services and infrastructure are managed by a third-party provider and shared with multiple organizations using the public internet. Public cloud makes computing resources available to anyone for purchase. Multiple users typically share the use of a public cloud.

Private Cloud

Private cloud provides a high level of security and privacy to data through firewalls and internal hosting. It also ensures that operational and sensitive data are not accessible to third-party providers. HP Data Centers, Microsoft, Elastra-private cloud, and Ubuntu are the example of a private cloud.

Hybrid Cloud

Hybrid cloud refers to a mixed computing, storage, and services environment made up of on-premises infrastructure, private cloud services, and a public cloud—such as Amazon Web Services (AWS) or Microsoft Azure—with orchestration among the various platforms.

Service Models

Infrastructure as a Service (IaaS)

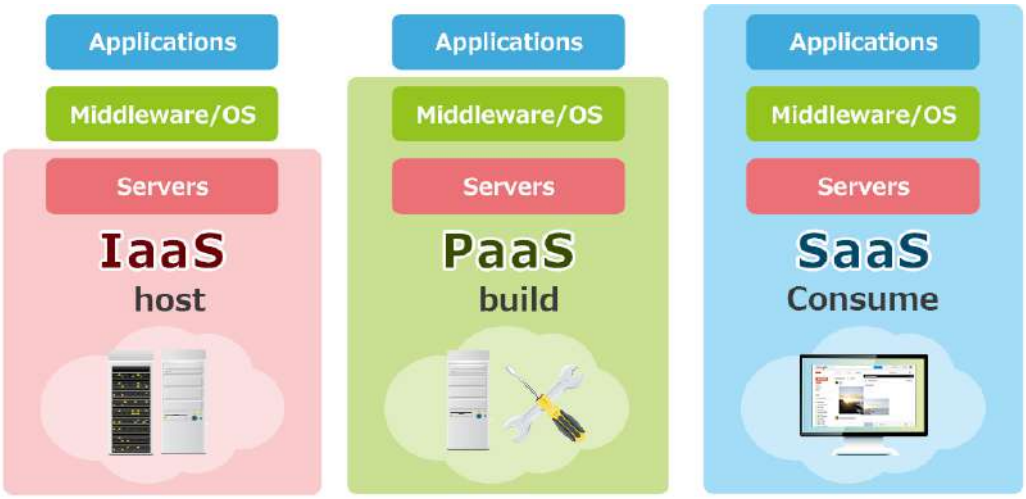
Infrastructure as a service are online services that provide high-level APIs used to dereference various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc.

Platform as a Service (PaaS)

Platform as a service (PaaS) is a cloud computing model where a third-party provider delivers hardware and software tools to users over the internet. As a result, PaaS frees developers from having to install in-house hardware and software to develop or run a new application. For e.g. Google App Engine

Software as a Service (SaaS)

Software as a service (or SaaS) is a way of delivering applications over the Internet—as a service. Instead of installing and maintaining software, you simply access it via the Internet, freeing yourself from complex software and hardware management.



What is Amazon EC2?

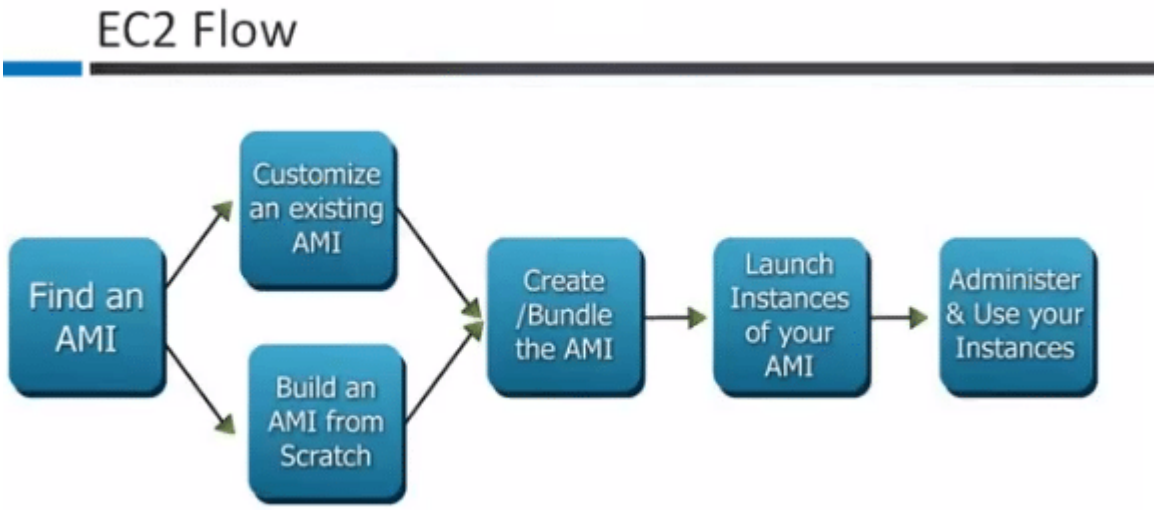
Amazon Elastic Compute Cloud (**EC2**) is a part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), that allows users to rent virtual computers on which to run their own computer applications. EC2 encourages scalable deployment of applications by providing a web service through which a user can boot an Amazon Machine Image (AMI) to configure a virtual machine, which Amazon calls an "instance", containing any software desired. A user can create, launch, and terminate server-instances as needed, paying by the second for active servers – hence the term "elastic". EC2 provides users with control over the geographical location of instances that allows for latency optimization and high levels of redundancy.[2] In November 2010, Amazon switched its own retail website platform to EC2 and AWS.

How AWS EC2 works?

It’s pretty simple to get up and running with Amazon EC2. You have a choice of pre-configured, templated Amazon Machine Images (AMI) to use for a quick launch. Or, if you prefer, you can create your own AMI that contains all of your libraries, data, applications and relevant configuration settings.

Amazon EC2 allows you to customize settings by configuring security and network access. After you do that, you determine your AMI instances and whether or not you want to run in multiple locations. You’ll also want to decide if you want static IP endpoints. You can do all of this in the AWS Console with minimum friction.

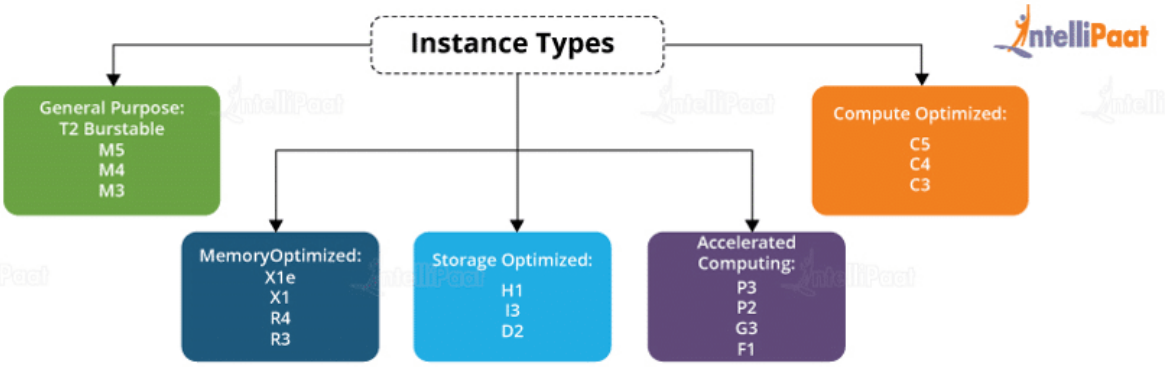
Once you’re ready to get started, you’ll boot, terminate and monitor as many instances as needed. You can do this through a web service or a variety of available management tools. And you’ll pay only for the resources you use.



What are the services Provided by Amazon web service EC2?

Amazon Elastic Compute Cloud (Amazon EC2) offers the broadest and deepest compute platform, with over 475 instances and choice of the latest processor, storage, networking, operating system, and purchase model to help you best match the needs of your workload. We are the first major cloud provider that supports Intel, AMD, and Arm processors, the only cloud with on-demand EC2 Mac instances, and the only cloud with 400 Gbps Ethernet networking. We offer the best price performance for machine learning training, as well as the lowest cost per inference instances in the cloud. More SAP, high performance computing (HPC), ML, and Windows workloads run on AWS than any other cloud.

.



Advantages of AWS EC2

1. ELASTIC WEB-SCALE COMPUTING

Amazon EC2 enables you to increase or decrease capacity within minutes, not hours or days. You can commission one, hundreds or even thousands of server instances simultaneously. Of course, because this is all controlled with web service APIs, your application can automatically scale itself up and down depending on its needs.

2. COMPLETELY CONTROLLED

You have complete control of your instances. You have root access to each one, and you can interact with them as you would any machine. You can stop your instance while retaining the data on your boot partition and then subsequently restart the same instance using web service APIs. Instances can be rebooted remotely using web service APIs. You also have access to console output of your instances.

3. FLEXIBLE CLOUD HOSTING SERVICES

You have the choice of multiple instance types, operating systems, and software packages. Amazon EC2 allows you to select a configuration of memory, CPU, instance storage, and the boot partition size that is optimal for your choice of operating system and application. For example, your choice of operating systems includes numerous Linux distributions, and Microsoft Windows Server.

4. DESIGNED FOR USE WITH OTHER AMAZON WEB SERVICES

Amazon EC2 works in conjunction with Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS) and Amazon Simple Queue Service (Amazon SQS) to provide a complete solution for computing, query processing and storage across a wide range of applications.

5. RELIABLE

Amazon EC2 offers a highly reliable environment where replacement instances can be rapidly and predictably commissioned. The service runs within Amazon’s proven network infrastructure and datacenters.

6. SECURE

Amazon EC2 works in conjunction with Amazon VPC to provide security and robust networking functionality for your compute resources. Your compute instances are located in a Virtual Private Cloud (VPC) with an IP range that you specify. You decide which instances are exposed to the Internet and which remain private.

7. INEXPENSIVE

Amazon EC2 passes on to you the financial benefits of Amazon’s scale. You pay a very low rate for the compute capacity you actually consume.

8. EASY TO START

Quickly get started with Amazon EC2 by visiting the Amazon Web Services Management Console to choose preconfigured software on Amazon Machine Images (AMIs). You can quickly deploy this software to EC2 via the EC2 console.

Disadvantages of AWS EC2

1. Instance types are rigid, must get entirely bigger instances even if just interested in more CPU or RAM.

2. Expensive at on-demand rates if elasticity is not needed (or expensive upfront payment if not using server for entire purchase length).

3. VM performance can be highly variable due to size, placement, and other uses on the same physical hardware.

4. No IPv6 support yet (as of 2015).

5. No VPN access to internal network (need custom software or DirectConnect service).

6. Networking is not as flexible as other providers or colo (no multicast support, low control over interfaces and bindings). Cross-region communication is more complex to setup.

Conclusion

EC2 simple means Elastic Compute Cloud .It is one of service that is provided by the AWS.

Moreover it serve as a Infrastructure as a Service (Iaas), because the AWS provides the server, networking facilities I.e. Infrastructure.

For ex:- If we want to build a house then first of all we need a basic structure of house…later on we will decide how to design our house on various aspects wall painting, interior design.

So the basic structure of house serves as an Infrastructure…

Similarly EC2 can be assumed to be similar that of house structure…

Later on it provides facility of adding Software's , apps , database or anything else means in EC2 we have control of operating system..(I.e. we can design our house and can make the modifications as per our choice).

In EC2 each server is known as instance,

Step 1:- We launch a EC2 server (instance)

Step 2:- Secondly we connect to EC2 server through remote protocols such as SSH, etc..

Step 3:- Now as soon we r connected to instance we got interface of the instance launched (GUI in case of windows server, click in case Linux distribution)

IT FOLLOWS ON- DEMAND PRICING -

On-Demand Instances let you pay for compute capacity by the hour or second (minimum of 60 seconds) with no long-term commitments. This frees you from the costs and complexities of planning, purchasing, and maintaining hardware and transforms what are commonly large fixed costs into much smaller variable costs.

REFERENCES

1. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html
2. https://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud
3. Mastering Cloud Computing Foundations and Applications Programming by Rajkumar Buyya, Christian Vecchiola, S. Thamar Selvi .