



**BRAINWARE UNIVERSITY**

*398, Ramkrishnapur Road, Barasat, North 24 Parganas, Kolkata - 700 125*

**Laboratory Assignment Submission**

**Session – 2024 - 2025**

**Name of the Department:-**

**Programme Name: -**

**Semester / Year:-**

**Course Code: -**

**Course Name: -**

**Name of the Student:-**

**Roll No :-**

**Registration No :-**

**Student Code : -**

# INDEX

SL.No.	Topic	Page No.	Exp. Date	Submission date	Signature	Remarks
1	Literary Management System	04	30/01/25			
2	Schedule all the activities and sub-activities of using the PERT/CPM charts.	05-07	17/04/25			
3	Use case diagram of library management system	08	06/02/25			
4	risk mitigation plan	09-10	02/05/25			
5	Library Management System	11	30/01/25			
6	Sequence Diagram	12-13	22/02/25			
7	Activity Diagram	14	27/02/25			
8	SRS Document (Library Management System)	15-17	13/03/25			
9	Project plan for Library Management	18	20/03/25			
10	Test cases (ATM Management)	19-21	20/03/25			
11	Test cases for Banking Applications	22-23	03/04/25			

12	Library Management System to be automated Developer work	24-26	03/04/25			
13	Project of Rohini Infotech Pvt Ltd	27-29	27/03/25			
14	Intelligent GST Accounting Software	30-32	26/04/25			
15	Know Solution Software Development	33-35	02/05/25			
16	Data flow diagram- [Level - 0] [Level - 1] [Level - 2]	36 37 38	06/03/25			

# Assignment-1

Write Draw the work break down structure of Literary Management System.

## Description-

- **Attributes:**

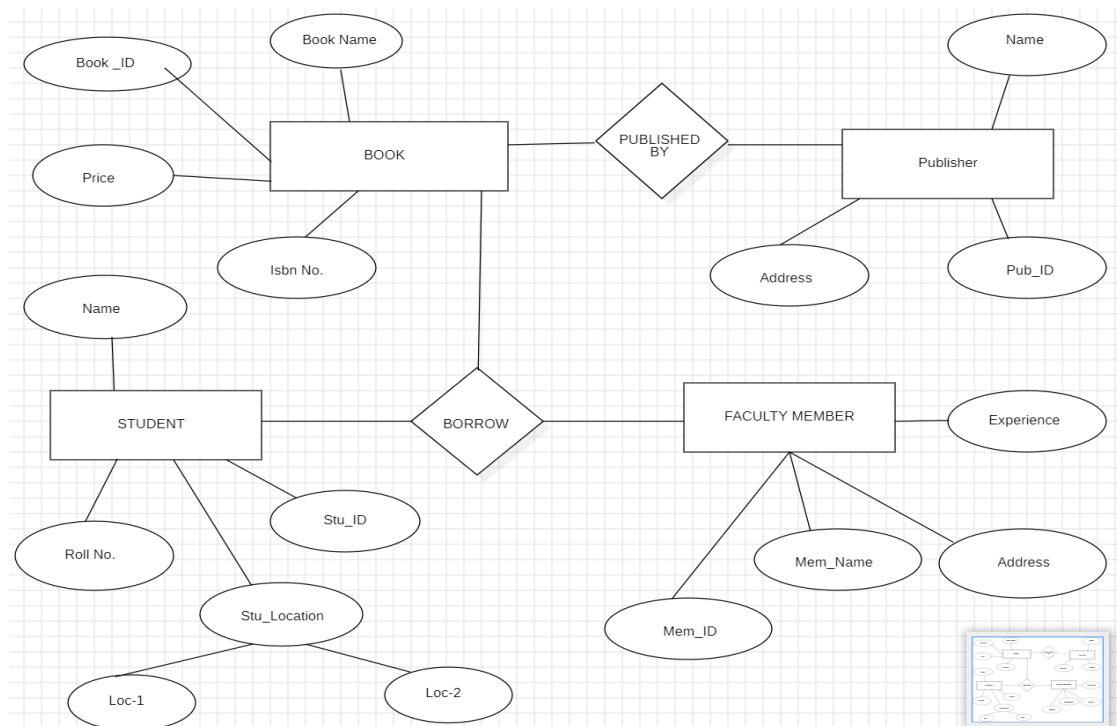
Attributes are characteristics or properties that describe entities like books, members, or staff.

These might include book titles, author names, ISBNs, member IDs, or staff roles, among other details.

- **Entity:**

entities" refer to the key objects or concepts that the system needs to track and manage. These entities typically form the foundation of the system's database design. Here's a breakdown of common entities found in a library management system:

## Output-



## Assignment-2

Schedule all the activities and sub-activities of using the PERT/CPM charts.

Event	Duration(Weeks)	Earliest Start	Earliest Finish	Latest Start	Latest Finish	Total Float
1-2	4	0	4	0	4	0
2-3	0	4	4	4	4	0
3-4	7	4	11	4	11	0
1-3	3	0	3	1	4	1
2-4	6	4	10	5	11	1

### Procedure:

#### 1. Introduction

Basically, CPM (Critical Path Method) and PERT (Programme Evaluation Review Technique) are project management techniques, which have been created out of the need of Western industrial and military establishments to plan, schedule and control complex projects.

#### Planning, Scheduling & Control

Planning, Scheduling (or organizing) and Control are considered to be basic Managerial functions, and CPM/PERT has been rightfully accorded due importance in the literature on Operations Research and Quantitative Analysis. Far more than the technical benefits, it was found that PERT/CPM provided a focus around which managers could brain-storm and put their ideas together. It proved to be a great communication medium by which thinkers and planners at one level could communicate their ideas, their doubts and fears to another level. Most important, it became a useful tool for evaluating the performance of individuals and teams.

#### The Framework for PERT and CPM

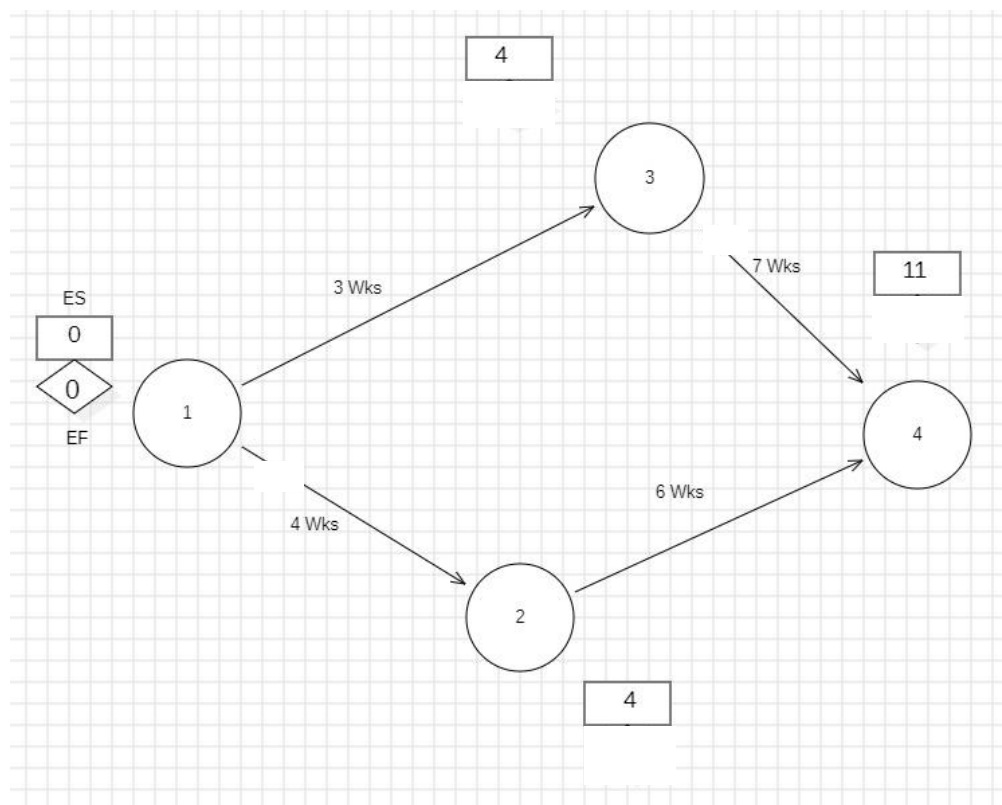
**Essentially, there are six steps which are common to both the techniques. The procedure is listed below:**

1. Define the Project and all of its significant activities or tasks. The Project (made up of several tasks) should have only a single start activity and a single finish activity.
2. Develop the relationships among the activities. Decide which activities must precede and which must follow others.
3. Draw the "Network" connecting all the activities. Each Activity should have unique event numbers. Dummy arrows are used where required to avoid giving the same numbering to two activities.
4. Assign time and/or cost estimates to each activity.

5. Compute the longest time path through the network. This is called the critical path.
6. Use the Network to help plan, schedule, and monitor and control the project.

The Key Concept used by CPM/PERT is that a small set of activities, which make up the longest path through the activity network control the entire project. If these "critical" activities could be identified and assigned to responsible persons, management resources could be optimally used by concentrating on the few activities which determine the fate of the entire project.

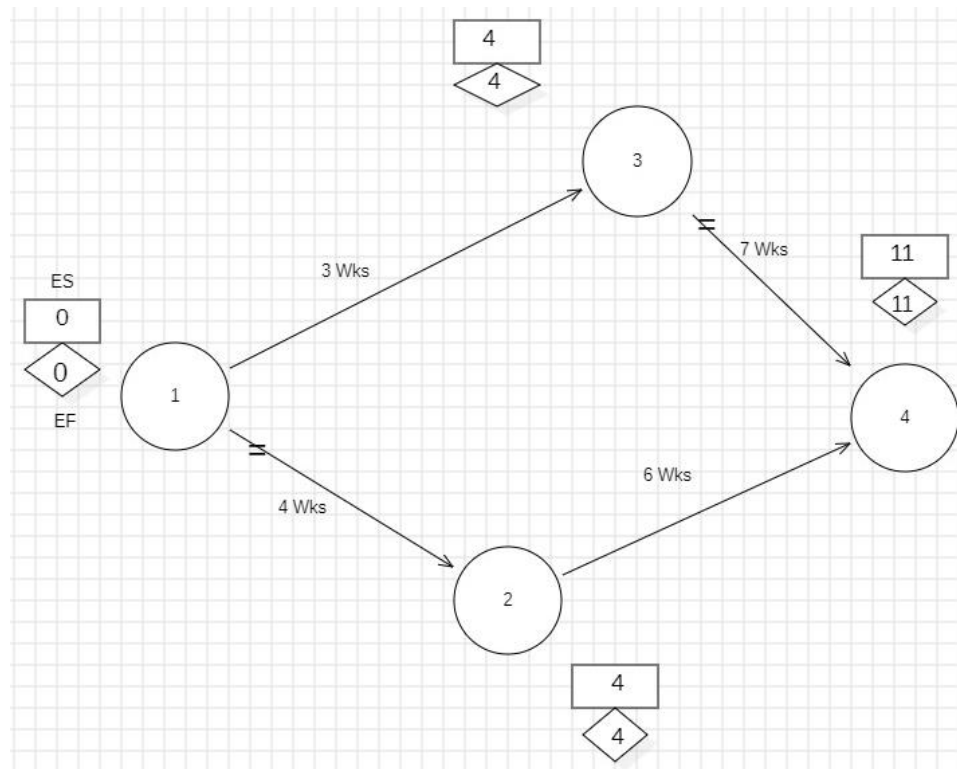
Non-critical activities can be preplanned, rescheduled and resources for them can be reallocated flexibly, without affecting the whole project.



At Event 3, we have to evaluate two predecessor activities – Activity 1-3 and Activity 2-3, both of which are predecessor activities. Activity 1-3 gives us an Earliest Start of 3 weeks at Event 3. However, Activity 2-3 also has to be completed before Event 3 can begin. Along this route, the Earliest Start would be  $4+0=4$ . The rule is to take the longer (bigger) of the two Earliest Starts. So the Earliest Start at event 3 is 4.

Similarly, at Event 4, we find we have to evaluate two predecessor activities – Activity 2-4 and Activity 3-4. Along Activity 2-4, the Earliest Start at Event 4 would be 10 wks, but along Activity 3-4, the Earliest Start at Event 4 would be 11 wks. Since 11 wks is larger than 10 wks, we select it as the Earliest Start at Event 4. We have now found the longest path through the Network. It will take 11 weeks along activities 1-2, 2-3 and 3-4. This is the Critical Path.

**The Backward Pass – Latest Finish Time Rule** To make the Backward Pass, we begin at the sink or the final event and work backwards to the first event.



**Tabulation & Analysis of Activities** We are now ready to tabulate the various events and calculate the Earliest and Latest Start and Finish times. We are also now ready to compute the SLACK or TOTAL FLOAT, which is defined as the difference between the Latest Start and Earliest Start.

Event	Duration(Weeks)	Earliest Start	Earliest Finish	Latest Start	Latest Finish	Total Float
1-2	4	0	4	0	4	0
2-3	0	4	4	4	4	0
3-4	7	4	11	4	11	0
1-3	3	0	3	1	4	1
2-4	6	4	10	5	11	1

The Earliest Start is the value in the rectangle near the tail of each activity

- ☐ The Earliest Finish is = Earliest Start + Duration
- ☐ The Latest Finish is the value in the diamond at the head of each activity
- ☐ The Latest Start is = Latest Finish – Duration

There are two important types of Float or Slack. These are Total Float and Free Float.

TOTAL FLOAT is the spare time available when all preceding activities occur at the earliest possible times and all succeeding activities occur at the latest possible times. Total Float = Latest Start – Earliest Start Activities with zero Total float are on the Critical Path.

# Assignment-3

## Use Case Diagram of Library Management System.

### Description-

A use case diagram for a Library Management System visually represents the interactions between users (actors) and the system's functionalities (use cases), illustrating how users interact with the library system to achieve specific goals.

### Actors:

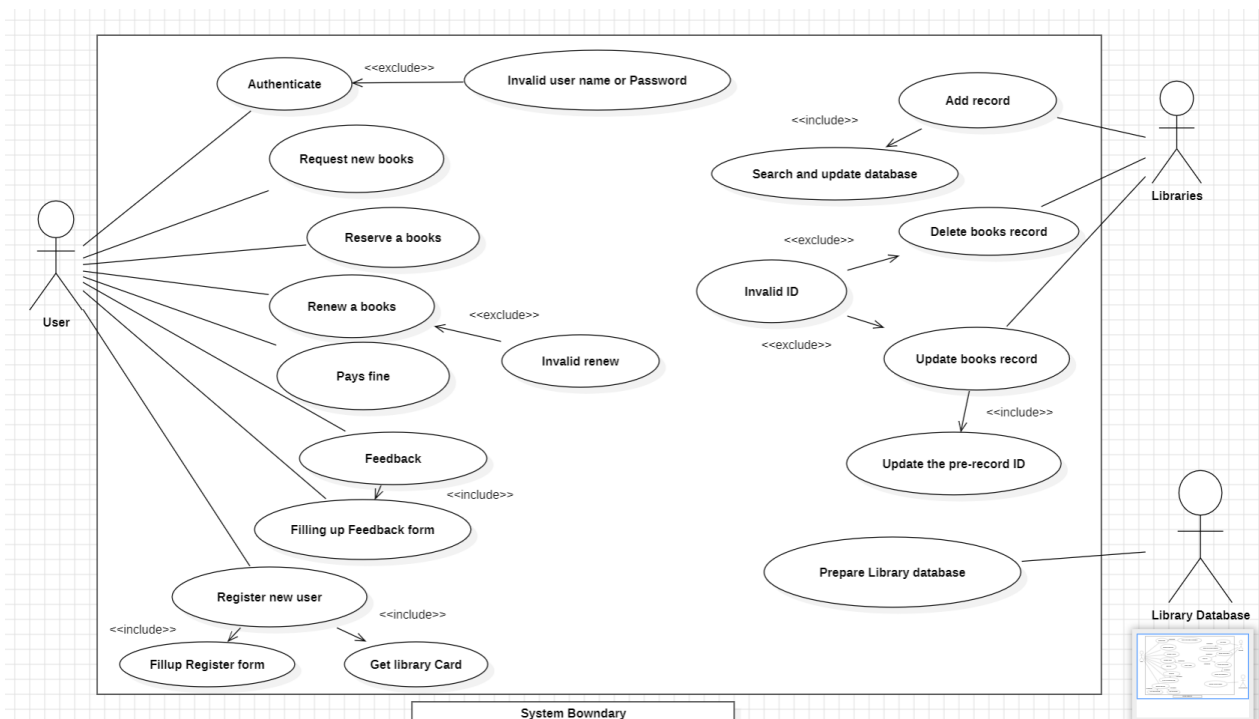
- Patron/Member: Represents any user who borrows books, reserves books, or interacts with the library system as a member.
- Librarian: Represents the library staff who manage books, members, and the system.
- System: Represents the Library Management System itself.

### Use Cases:

#### Patron Use Cases:

- Search for Books: Allows patrons to find books in the library catalog.
- Borrow Books: Allows patrons to check out books.
- Return Books: Allows patrons to return borrowed books.

### Output-

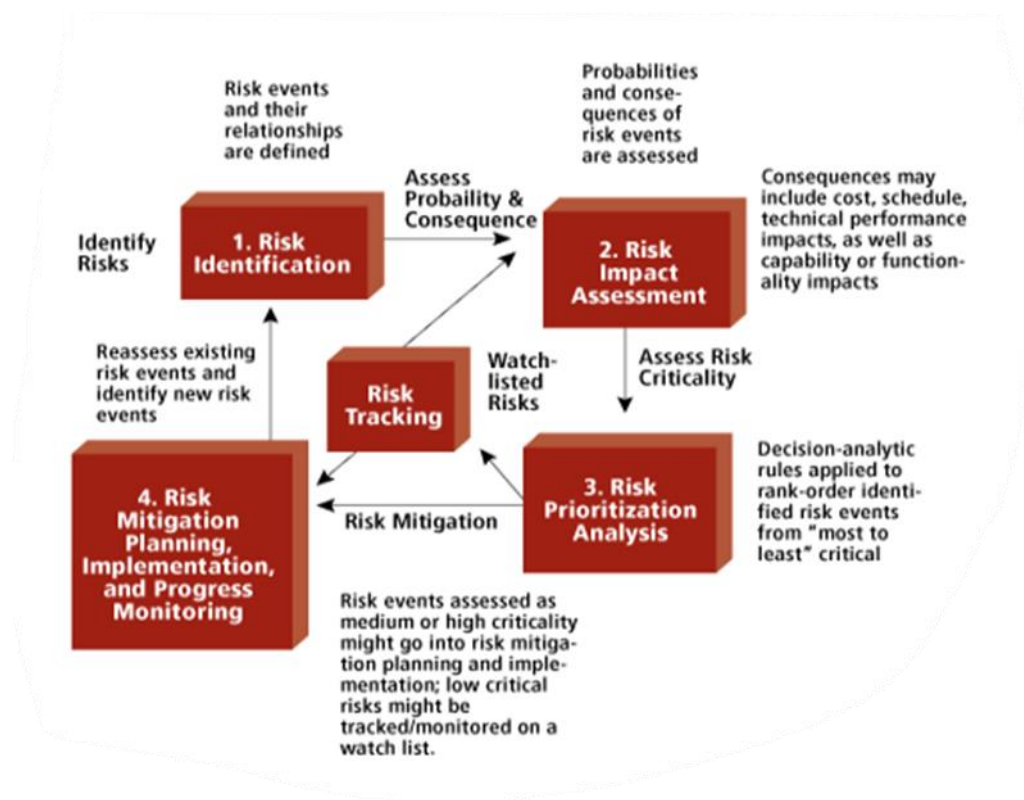




# Assignment-4

**Identify and analyze all the possible risks and its risk mitigation plan for the system to be automated.**

**Procedure:** Definition: Risk mitigation planning is the process of developing options and actions to enhance opportunities and reduce threats to project objectives. Risk mitigation implementation is the process of executing risk mitigation actions. Risk mitigation progress monitoring includes tracking identified risks, identifying new risks, and evaluating risk process effectiveness throughout the project.



**Example:**

## Evolution of Healthcare Enterprise Risk Management (ERM)

**ERM encompasses eight risk domains:**

1. Operational
2. Clinical & Patient Safety
3. Strategic
4. Financial
5. Human Capital
6. Legal & Regulatory
7. Technological

## 8. Environmental- and Infrastructure-Based Hazards.

### **Create a Healthcare Risk Management Plan:**

**There are some fundamental components that belong in all healthcare risk management plans:**

**Education & Training:** Risk management plans need to detail employee training requirements which should include new employee orientation, ongoing and in-service training, annual review and competency validation, and event-specific training.

**Patient & Family Grievances:** To promote patient satisfaction and reduce the likelihood of litigation, procedures for documenting and responding to patient and family complaints should be described in the Risk Management Plan. Response times, staff responsibilities, and prescribed actions need to be articulated and communicated.

**Purpose, Goals, & Metrics:** Risk management plans should clearly define the purpose and benefits of the healthcare risk management plan. Specific goals to reduce liability claims, sentinel events, near misses, and the overall cost of the organization's risk should also be well articulated. Additionally, reporting on quantifiable and actionable data should be detailed and mandated by the plan.

**Communication Plan:** While it is critical that the healthcare risk management team promote open and spontaneous dialogue, information about how to communicate about risk and with whom should be provided in the healthcare risk management plan. Next steps and follow-up activities should be documented. It is essential as well that the plan detail reporting requirements to departments and C-Suite personnel. Furthermore, the plan should promote a safe, "no-blame" culture and should include anonymous reporting capabilities.

**Contingency Plans:** Risk management plans also need to include contingency preparation for adverse system-wide failures and catastrophic situations such as malfunctioning EHR systems, security breaches, and cyber-attacks. The plan needs to include emergency preparedness for things like disease outbreaks, long-term power loss, and terror attacks or mass shootings. **Reporting Protocols:** Every healthcare organization must have a quick and easy-to-use, system for documenting, classifying, and tracking possible risks and adverse events. These systems must include protocols for mandatory reporting.

**Response & Mitigation:** Plans for healthcare risk must also include collaborative systems for responding to reported risks and events including acute response, follow-up, reporting, and repeat failure prevention.

# Assignment-5

## Class Diagram for library management system.

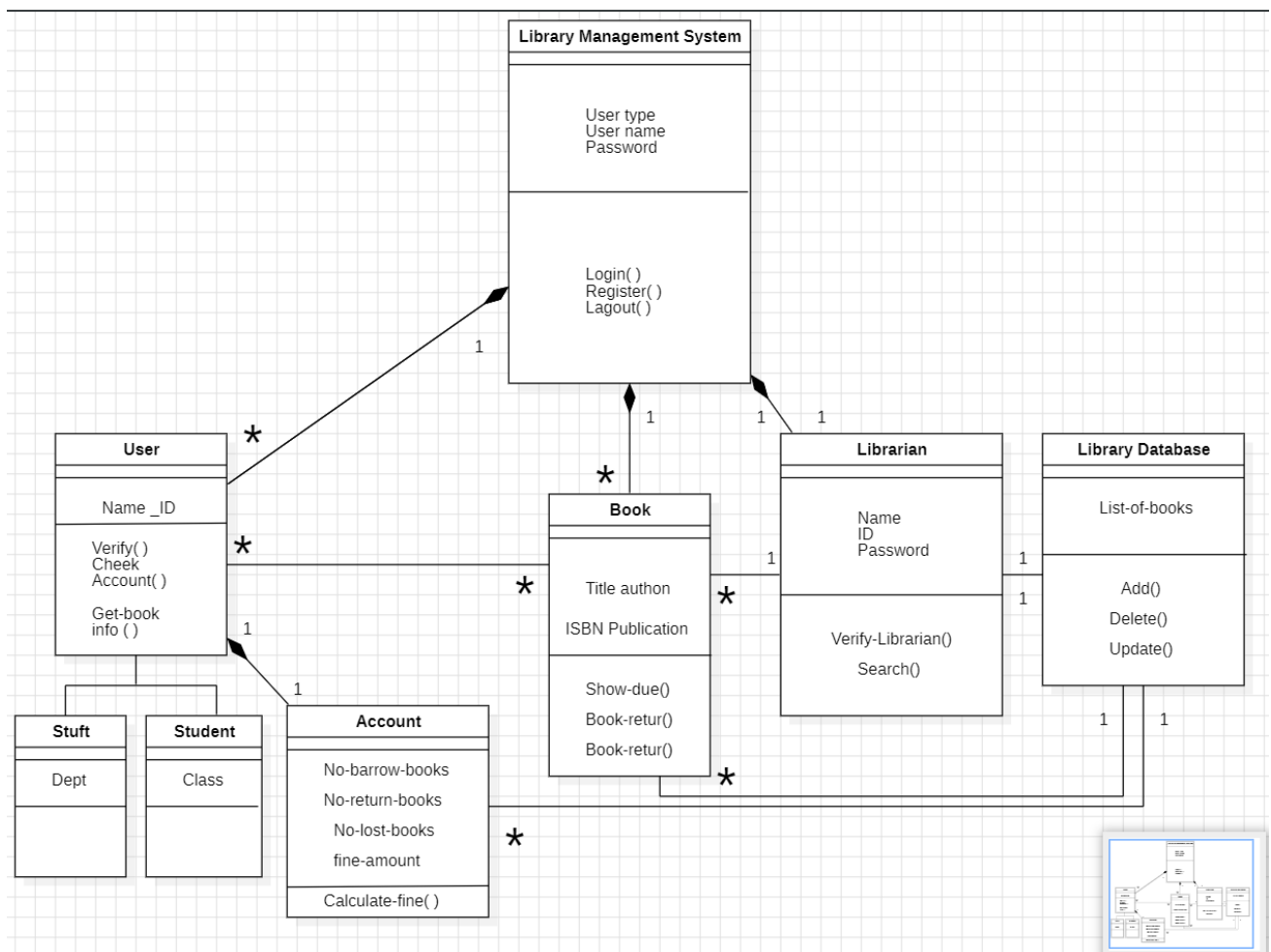
### Description-

Library management system class diagrams describe the structure of a library system. The diagram does this by showing each class and its attributes, methods, and relationships. Our library management system coordinates all the functions of a library. It manages how members search for and check out books.

### Relationships-

The “Library Management System” is superclass that consists of various classes. These classes are User, Book, and Librarian as shown in diagram. Further, for “Account” class, “User” is a superclass. All of these, share a relationship and these relationships are known as aggregate relationships.

### Output-



# Assignment-6

**Draw a Sequence diagram for online ticket reservation.**

## **Description-**

- **User Opens Website:**
  - The user initiates the process by opening the online ticket reservation website in their web browser.
- **Request Homepage:**
  - The web browser sends a request to the web server for the homepage.
- **Retrieve Event Data:**
  - The web server queries the database to retrieve information about available events.
- **Display Event List:**
  - The web server sends the homepage along with the event list to the web browser.
  - The browser displays this information to the user.
- **Event and Ticket Selection:**
  - The user selects an event and specifies the number of tickets they want.
- **Submit Request:**
  - The web browser sends the event and ticket request to the web server.
- **Check Availability:**
  - The web server checks the database for ticket availability.
- **Availability Check and Seat Selection:**
  - If tickets are available, the website will display a seat selection page. The user then selects their seats and confirms.
- **Seat Reservation:**
  - The web browser submits the seat selection to the web server, which updates the database, reserving the seats.

- **Payment:**

- The web server displays a payment page, and the user enters their payment details.
- The web browser submits the payment information to the web server.

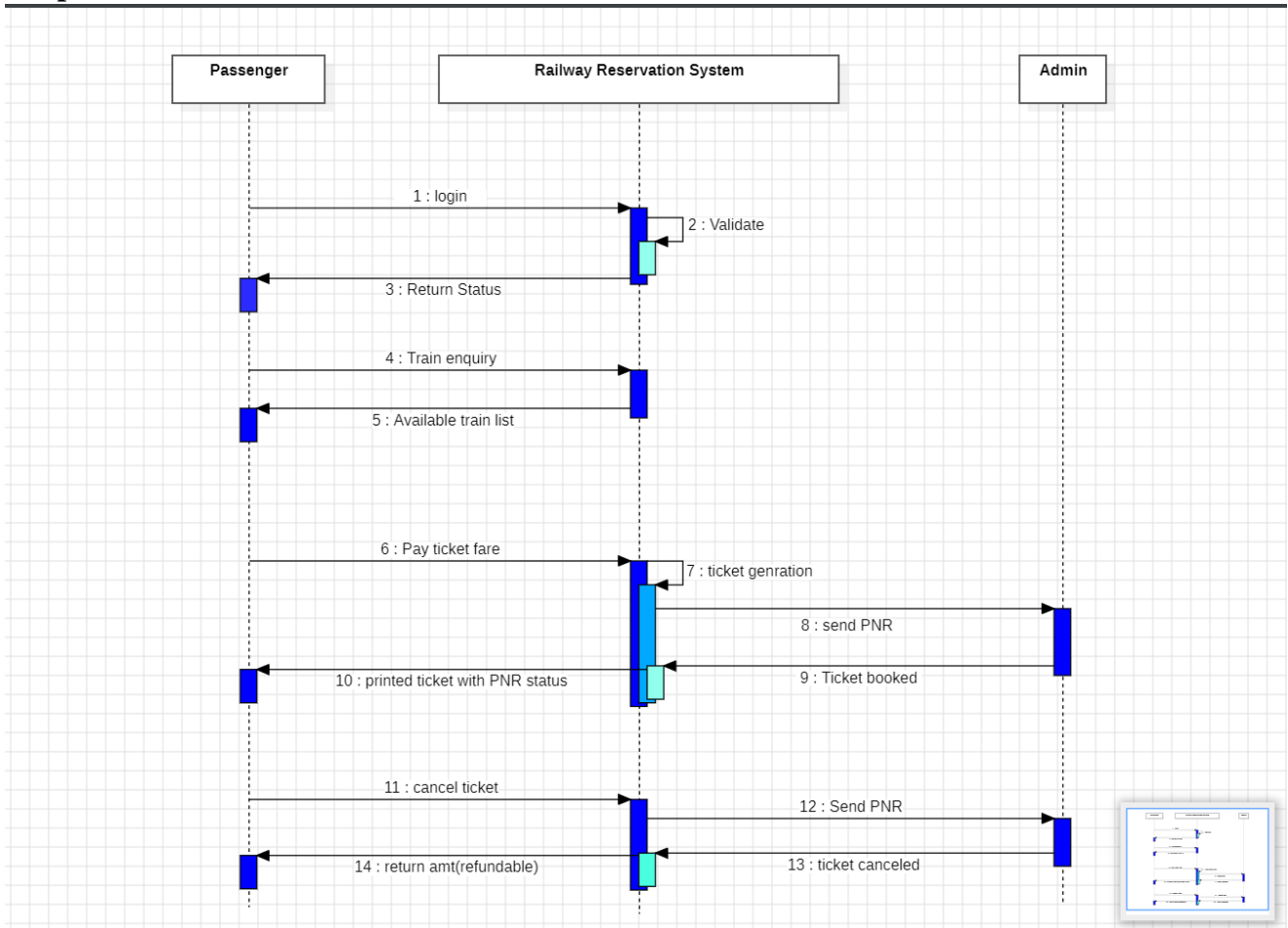
- **Payment Processing and Ticket Confirmation:**

- The web server processes the payment and updates the ticket status in the database.
- The web server sends a ticket confirmation and e-ticket to the web browser.
- The browser displays the confirmation and e-ticket to the user.

- **Unavailable Tickets:**

- If tickets are unavailable, the web server sends a "Tickets Unavailable" message to the web browser.
- The browser displays the message to the user.

### Output-



# Assignment-7

## Draw a Activity Diagram of Library Management System.

### Description-

An activity diagram for a Library Management System (LMS) visually represents the flow of activities and actions within the system, showing how users (like librarians and members) interact to manage books, members, and other library resources.

- **Activities:**

Represented by rounded rectangles, these describe the actions or tasks performed within the system.

- **Actions:**

Represented by rectangles, these are the individual steps within an activity.

- **Control Flow:**

Arrows indicate the sequence of activities and actions.

- **Decision Points:**

Diamond shapes represent decision points where the flow of activities can diverge based on certain conditions.

- **Start Node:**

A solid black circle indicates the starting point of an activity diagram.

- **End Node:**

A solid black circle within a larger white circle indicates the end of an activity diagram.

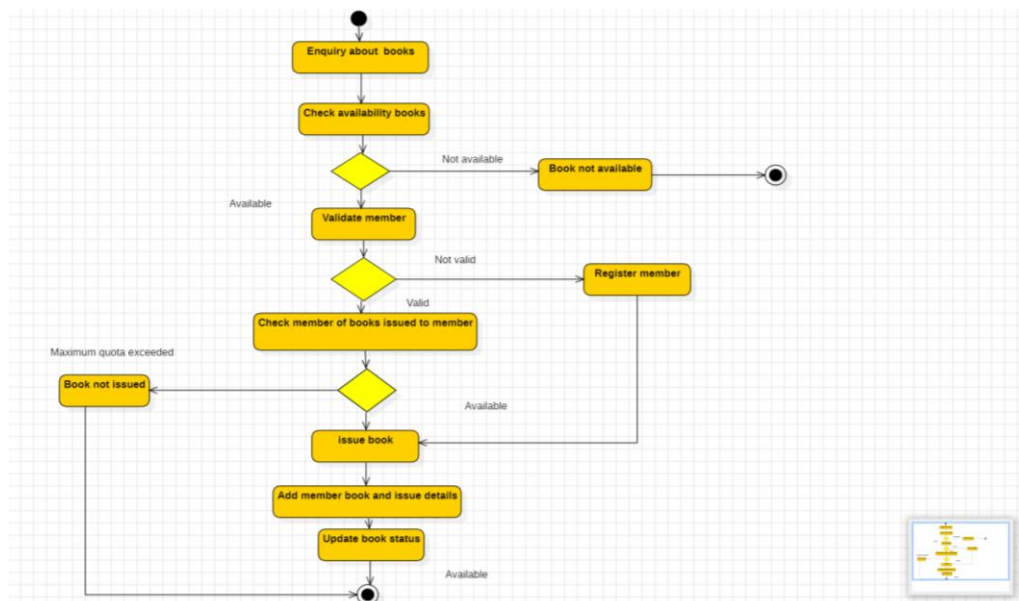
- **Parallel Activities:**

Activities that can occur simultaneously are represented using parallel lines.

- **Loop:**

A loop indicates that an activity is repeated until a certain condition is met.

### Output-



# Assignment-8

**Define the functional and non-functional requirements of Library Management System by using use cases and document in SRS document.**

## 1. Introduction

### 1.1 Purpose

The purpose of this document is to define the functional and non-functional requirements for the Library Management System (LMS). This system is designed to manage the operations of a library, including book cataloging, user management, and transaction handling.

### 1.2 Scope

The LMS will allow librarians to manage books, track borrowing and returns, and manage library members. Users will be able to search for books, check availability, and borrow or return books. The system aims to streamline the library's workflow and improve efficiency.

### 1.3 Definitions, Acronyms, and Abbreviations

- LMS: Library Management System
- UI: User Interface
- DBMS: Database Management System

### 1.4 References

- IEEE 830-1998: Recommended Practice for Software Requirements Specifications

### 1.5 Overview

This document includes system functionalities, use cases, and system constraints.

## 2. Overall Description

### 2.1 Product Perspective

The LMS is a standalone web-based or desktop application designed for educational institutions and public libraries.

### 2.2 Product Functions

- Book management (addition, deletion, updating book details)
- User management (registration, login, user roles)
- Book borrowing and return tracking
- Search functionality for books
- Fine calculation for overdue books

### 2.3 User Characteristics

- Librarians: Manage books and users

- Members: Search, borrow, and return books

## 2.4 Constraints

- The system must be available 24/7.
- The database should support at least 100,000 book records.
- Response time should not exceed 2 seconds for any operation.

## 2.5 Assumptions and Dependencies

- The system assumes internet connectivity for cloud-based operations.
- The database server should be regularly maintained.

## 3. Functional Requirements 3.1 User Authentication

- Users must register and log in to access personalized features.

## 3.2 Book Management

- Librarians can add, update, and delete books.
- Books have attributes: title, author, ISBN, genre, availability status.

## 3.3 Borrow and Return Books

- Users can borrow books (with a limit of 5 books at a time).
- Borrowed books must be returned within the due date.

## 3.4 Fine Calculation

- A late return incurs a daily fine of a predefined amount.

## 3.5 Search and Filtering

- Users can search books by title, author, or genre.

## 4. Non-Functional Requirements 4.1 Performance

- The system should handle 50 concurrent users without performance degradation.

## 4.2 Security

- User data must be encrypted.
- Only librarians can modify book records.

## 4.3 Usability

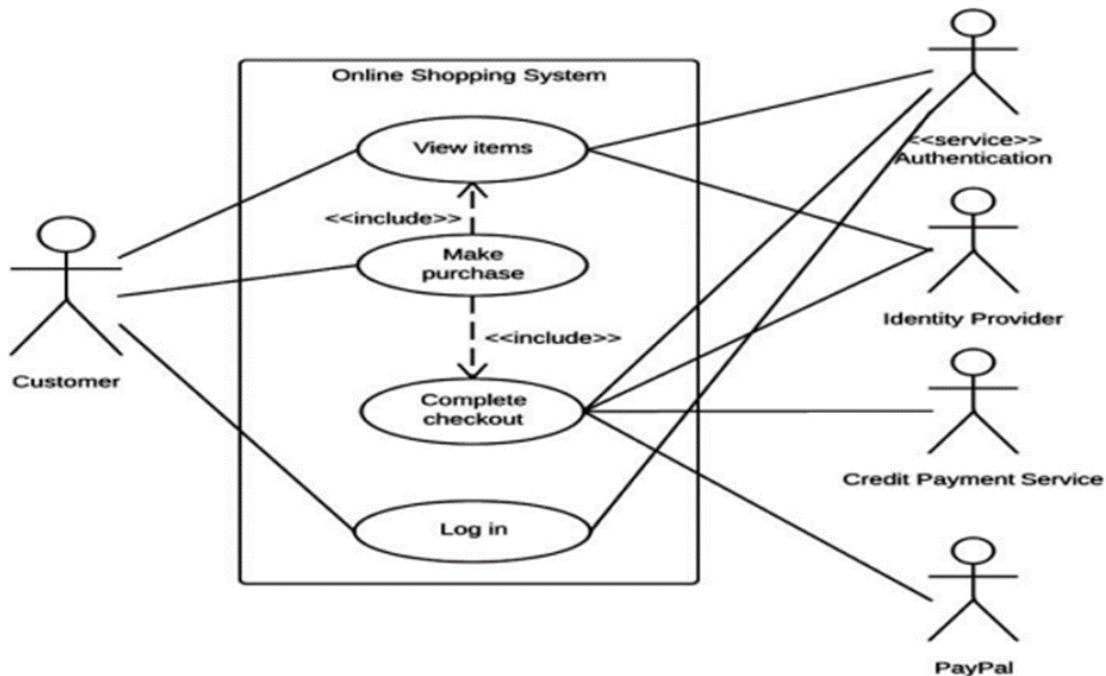
- The UI should be intuitive and accessible.

## 4.4 Availability



- System uptime should be 99.9% per year.

## 5. Use Case Diagram



**6. Conclusion** This SRS provides a detailed overview of the Library Management System's functional and non-functional requirements, ensuring clarity in development and implementation.

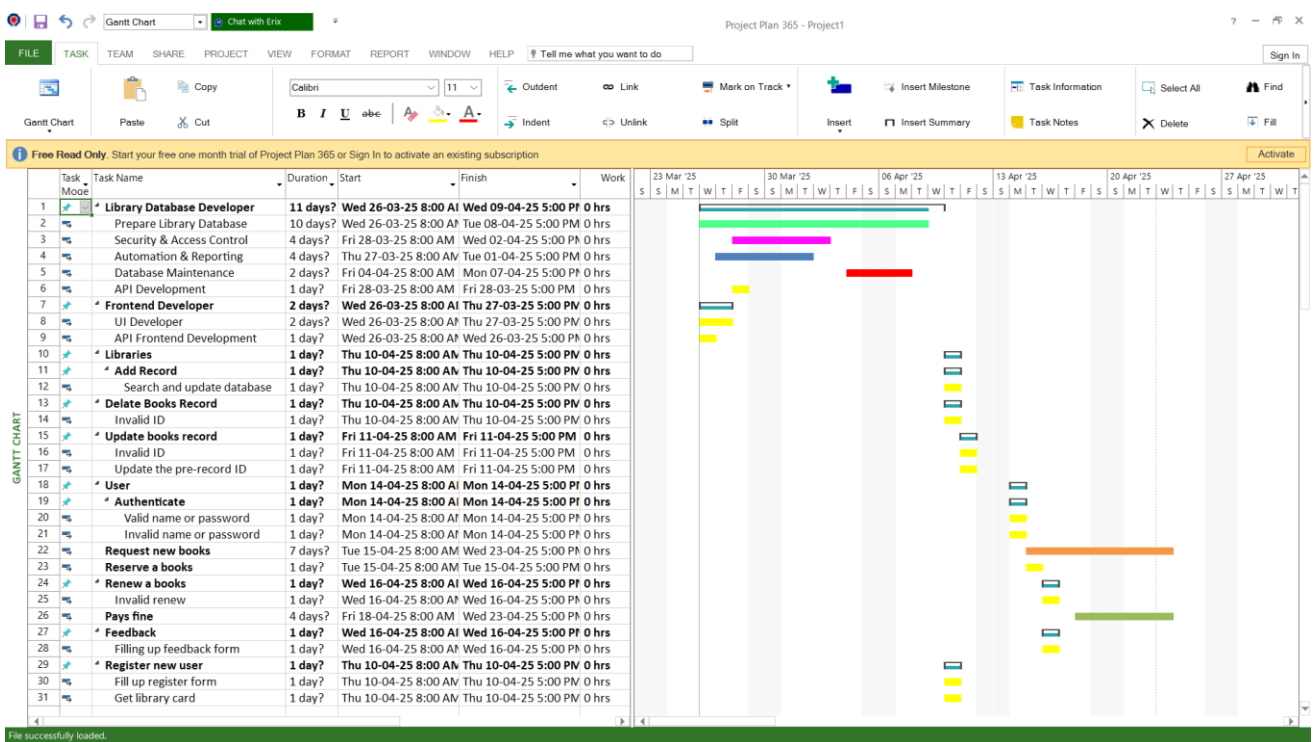
# Assignment-9

**Define Complete Project plan for Library Management System to be automated using Microsoft Project Tool.**

## Purpose-

A library management system means using computers and software to streamline tasks like cataloging, circulation, and acquisitions, improving efficiency and access for both librarians and patrons.

## Project Tool-



## Conclusion-

Automating a library management system (LMS) offers significant benefits, streamlining operations, improving efficiency, and enhancing user experience.

# Assignment-10

## Test Cases for Functionalities of ATM system.

### Purpose:

This document describes the software requirements and specification (SRS) for an automated teller machine (ATM) network. The document is intended for the customer and the developer (designers, testers, maintainers). The reader is assumed to have basic knowledge of banking accounts and account services. Knowledge and understanding of Unified Modeling Language (UML) diagrams is also required.

### Scope:

The software supports a computerized banking network called \_Bank24,. The network enables customers to complete simple bank account services via automated teller machines (ATMs) that may be located off premise and that need not be owned and operated by the customer's bank. The ATM identifies a customer by a cash card and password. It collects information about a simple account transaction (e.g., deposit, withdrawal, transfer, bill payment), communicates the transaction information to the customer's bank, and dispenses cash to the customer. The banks provide their own software for their own computers. The \_Bank24,, software requires appropriate record keeping and security provisions. The software must handle concurrent accesses to the same account correctly.

### Intended Audience:

The intended audience of this SRS consists of:

- ☐ Software designers
- ☐ Systems engineers
- ☐ Software developers
- ☐ Software testers
- ☐ Customers

### The actors of the system are:

1. User
2. ATM Machine
3. Bank

### Product Perspective:

An automated teller machine (ATM) is a computerized telecommunications device that provides the customers of a financial institution with access to financial transactions in a public space without the need for a human clerk or bank teller. On most modern ATMs, the customer is identified by inserting a plastic ATM card with a magnetic stripe or a plastic smartcard with a chip, that contains a unique card number and some security information, such as an expiration date or CVC (CVV). Security is provided by the customer entering a personal identification number (PIN).

### Product functions:

Using an ATM, customers can access their bank accounts in order to make cash withdrawals (or credit card cash advances) and check their account balances.

### The functions of the system are:

1. Login
2. Get Balance Information
3. Withdraw Cash
4. Transfer Funds

### Operating Environments:

The hardware, software and technology used should have following specifications:

- ☐ Ability to read the ATM card.

- ☐ Ability to count the currency notes.
- ☐ Touch screen for convenience.
- ☐ Keypad (in case touchpad fails)
- ☐ Continuous power supply.
- ☐ Ability to connect to bank's network.
- ☐ Ability to validate user.

#### **Design/implementation constraints:**

##### **Login:**

##### **Validate Bank Card**

- ☐ Validate for Card Expiration Date
- ☐ Validate that the card's expiration date is later than today's date
- ☐ If card is expired, prompt error message "Card is expired"

##### **Validate for Stolen or Lost Card**

- ☐ Validate that the card is not reported lost or stolen.
- ☐ If card is lost, prompt error message, "Card has been reported lost".
- ☐ If card is stolen, prompt error message, "Card has been reported stolen".

##### **Validate for Disabled Card**

- ☐ Validate that the card is not disabled
- ☐ If card is disabled, prompt error message, "Card has been disabled as of <expiration date>"
- ☐ Validate for Locked Account
- ☐ Validate that the account is not locked
- ☐ If account is locked, prompt error message "Account is locked"

##### **Validate PIN**

- ☐ Validate that the password is not blank
- ☐ If PIN is blank, prompt error message "Please provide PIN"
- ☐ Validate that the password entered matches the password on file
- ☐ If password does not match, prompt error message "Password is Incorrect"

##### **Lock Account**

- ☐ If number of consecutive unsuccessful logins exceeds three attempts, lock account

##### **Maintain Consecutive Unsuccessful Login Counter**

Increment Login Counter

For every consecutive Login attempt, increment logic counter by 1.

Reset login counter to 0 after login is successful.

Get Balance Information

Withdraw Cash

Transfer Funds

##### **Assumptions and Dependencies:**

- ☐ Hardware never fails
- ☐ ATM casing is impenetrable
- ☐ Limited number of transactions per day (sufficient paper for receipts)
- ☐ Limited amount of money withdrawn per day (sufficient money)

##### **External Interface Requirements**

##### **User interfaces**

The customer user interface should be intuitive, such that 99.9% of all new ATM users are able to complete their banking transactions without any assistance.

##### **Hardware interfaces**

The hardware should have following specifications:

- ☐ Ability to read the ATM card
- ☐ Ability to count the currency notes

- ☐ Touch screen for convenience
- ☐ Keypad (in case touchpad fails)
- ☐ Continuous power supply
- ☐ Ability to connect to bank's network
- ☐ Ability to take input from user
- ☐ Ability to validate user

#### **Software interfaces**

The software interfaces are specific to the target banking software systems. At present, two known banking systems will participate in the ATM network.

- ☐ State Bank
- ☐ Indian Overseas Bank

#### **Safety requirements:**

Must be safe kept in physical aspects, say in a cabin

- ☐ Must be bolted to floor to prevent any kind of theft
- ☐ Must have an emergency phone outside the cabin
- ☐ There must be an emergency phone just outside the cabin
- ☐ The cabin door must have an ATM card swipe slot
- ☐ The cabin door will always be locked, which will open only when user swipes his/her ATM card in the slot & is validated as genuine

#### **Security requirements:**

- ☐ Users accessibility is censured in all the ways
- ☐ Users are advised to change their PIN on first use
- ☐ Users are advised not to tell their PIN to anyone
- ☐ The maximum number of attempts to enter PIN will be three

#### **Some of the possible Bugs on ATM machine?**

1. Successful insertion of ATM card
2. Unsuccessful operation due to insert card in wrong angle
3. Unsuccessful operation due to invalid account Ex: other bank card or time expired card
4. successful entry of PIN number
5. un successful operation due to enter wrong PIN number 3times
6. successful selection of language
7. successful selection of account type
8. unsuccessful operation due to invalid account type
9. successful selection of withdraw operation
10. successful selection of amount to be withdrawal
11. successful withdraw operation
12. unsuccessful withdraw operation due to wrong denominations
13. unsuccessful withdraw operation due to amount is greater than day limit
14. unsuccessful withdraw operation due to lack of money in ATM
15. unsuccessful withdraw operation due to amount is greater than possible balance
16. unsuccessful withdraw operation due to transactions is greater than day limit
17. unsuccessful withdraw operation due to click cancel after insert card
18. unsuccessful withdraw operation due to click cancel after insert card & pin number
19. unsuccessful withdraw operation due to click cancel after insert card, pin number & language
20. unsuccessful withdraw operation due to click cancel after insert card, pin number, language & account type
21. unsuccessful withdrawal operation due to click cancel after insert card, pin number, language, account type & withdrawal operation
22. unsuccessful withdrawal operation due to click cancel after insert card, pin number, language, account type, withdrawal operation & amount to be withdraw.

# Assignment-11

## Test plan document for Library Management System.

Testing a banking application requires a comprehensive approach covering functional, non-functional, security, and performance aspects. Here are some test cases categorized accordingly:

### 1. Functional Test Cases:-

#### Account Management-

- Verify that a user can successfully create a new bank account.
- Verify that a user can log in with valid credentials.
- Verify that invalid login attempts are rejected.
- Verify that users can update their profile details.
- Verify that password reset functionality works correctly.

#### Transactions & Payments-

- Verify that users can deposit money into their accounts.
- Verify that users can withdraw money within allowed limits.
- Verify that users can transfer funds between accounts.
- Verify that transactions reflect correctly in the account statement.
- Verify that overdraft transactions are blocked if the account has insufficient funds.

#### Loan & Credit Services-

- Verify that users can apply for loans online.
- Verify that EMI calculations are correct.
- Verify that credit card payments are processed successfully.

### 2. Security Test Cases:

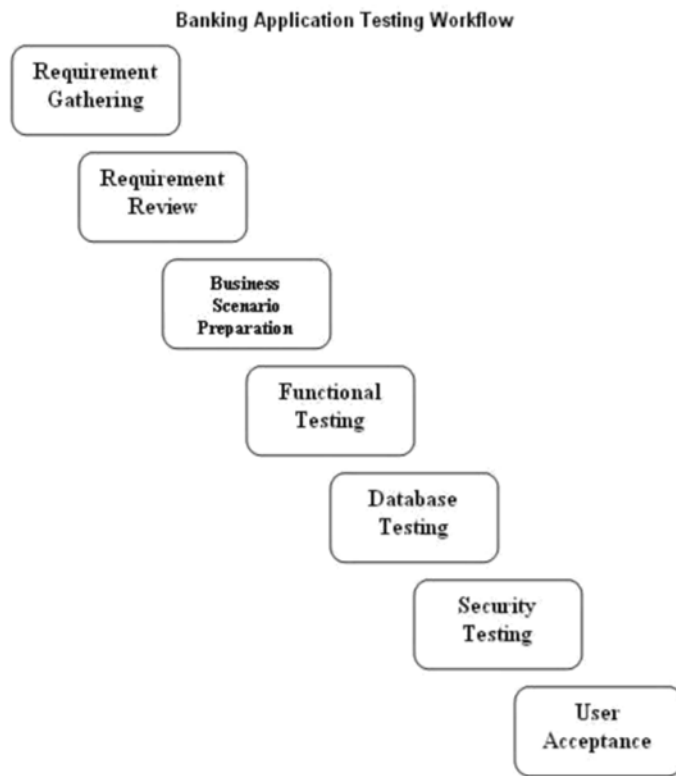
- Verify that sensitive data (passwords, PINs) are encrypted.
- Verify that session timeouts occur after a period of inactivity.
- Verify that the application enforces multi-factor authentication (MFA).
- Verify that failed login attempts lock the account after multiple tries.
- Verify SQL injection, XSS, and other security vulnerabilities.

### 3. Performance Test Cases :

- Verify the response time for account login and transactions.
- Verify system behavior under peak load conditions.
- Verify that concurrent users can perform banking operations without performance degradation.

### 4. Usability & Accessibility Test Cases:

- Verify that the application UI is user-friendly and easy to navigate.
- Verify that the application supports screen readers for visually impaired users.
- Verify that all banking operations are accessible via mobile devices.



## **5. Conclusion:**








Testing a banking application requires a detailed and strategic approach to ensure reliability, accuracy, security, and performance. By covering a wide range of test scenarios—including functional, security, performance, and usability tests—we can validate that the application meets both user and regulatory requirements. Thorough testing not only ensures a seamless and secure experience for users but also strengthens trust in the banking system. A well-structured testing process ultimately contributes to the success and stability of the banking application in a highly sensitive and competitive domain.

# Assignment-12

## Test plan document for Library Management System.

### Objective:

To develop and implement an automated Library Management System that improves efficiency in book management, borrowing, returns, cataloging, and reporting.

-  Developer Responsibilities in LMS Automation
-  System Setup & Environment Configuration (Week 1-2)
- Set up development environment (e.g., Visual Studio, VS Code, SQL Server, .NET, Java, etc.)
- Install necessary frameworks & libraries (e.g., ASP.NET, React.js, Node.js, Bootstrap)
- Configure version control system (e.g., GitHub, GitLab, Azure DevOps)
- Database setup: Install Microsoft SQL Server and create an initial schema
-  Backend Development (Week 3-6)
-  Tech Stack: .NET (C#), Java Spring Boot, Node.js (Express)
-  Database: SQL Server, MySQL, PostgreSQL
- Tasks:
- ☒ Develop user authentication module (Admin, Librarian, Students)
- ☒ Implement role-based access control (RBAC)
- ☒ Design & develop RESTful APIs for:
- Book cataloging (Add, Update, Delete, Search)
- Borrowing & returning books
- Fine calculation for overdue books
- User management
- ☒ Create a notification system (email alerts, SMS for due dates)
- ☒ Implement logging & error handling
- Milestones:
- ☒ User authentication and role-based access working
- ☒ Core backend API services functional
-  Database Development (Week 5-7)
-  DBMS: Microsoft SQL Server / MySQL / PostgreSQL
- Tasks:
- ☒ Design Entity-Relationship Diagram (ERD)
- ☒ Create tables for:
- Users (Admin, Librarians, Members)
- Books (Title, Author, ISBN, Copies Available, etc.)
- Borrowing transactions (Book ID, User ID, Issue & Return Dates)






- Fines (Late return charges)
  - ✓ Implement stored procedures for issuing/returning books
  - ✓ Optimize queries for performance
- Milestones:
- ✓ Database schema finalized & tested
- 4 Frontend Development (Week 6-9)
  - ☺ Tech Stack: React.js, Angular, Vue.js, Bootstrap, Tailwind CSS
  - Tasks:
    - ✓ Develop UI for:
      - User login and registration
      - Book catalog search (filters, sorting)
      - Borrow & return book forms
      - Admin dashboard for book & user management
    - ✓ Integrate backend APIs with frontend
    - ✓ Implement form validation
    - ✓ Develop responsive UI (mobile-friendly)
  - Milestones:
  - ✓ Fully functional book catalog & user dashboard
- 5 Testing & Debugging (Week 10-11)
  - ✂ Tools: Jest, Selenium, Postman
  - Tasks:
    - ✓ Unit testing for backend functions
    - ✓ API testing using Postman
    - ✓ UI testing with Selenium
    - ✓ Fix bugs & optimize performance
  - Milestones:
  - ✓ 95%+ test case pass rate
- 6 Deployment & Go-Live (Week 12)
  - ☁ Cloud: AWS, Azure, Google Cloud
  - ✂ CI/CD Tools: GitHub Actions, Jenkins
  - Tasks:
    - ✓ Deploy backend to cloud
    - ✓ Deploy frontend to hosting service
    - ✓ Conduct final security checks
    - ✓ Handover to users & provide developer documentation
  - Milestones:
  - ✓ LMS successfully deployed

## ✓ Conclusion

The **Database Developer's role** in automating a Library Management System is critical to ensuring that the entire system functions efficiently, securely, and with high performance. From initial **requirement analysis and ERD design**, through to **schema implementation, stored procedures, query optimization, and data security**, every step lays the foundation for a reliable LMS.

By following best practices in **SQL development**, implementing **robust relationships and constraints**, and preparing for **real-world scenarios** like overdue fines and limited book availability, the database ensures seamless data flow and integrity across all library operations.

In short:

-  **Structured Data = Better Decisions**
-  **Secure Data = Trustworthy System**
-  **Optimized Data = Faster Performance**

With a well-designed and properly implemented database, the LMS can offer a smooth, scalable, and user-friendly experience for administrators, librarians, and students alike.

## Assignment-13

Assume a Project of Rohini Infotech Pvt Ltd will start on 7th August 2023. The fiscal year of the manufacturing company starts on 1st April. The company works by a standard time from 9.00AM to 2.00PM (pre-lunch) and 3:00 PM to 6:00 PM (post-lunch). The working hour for Saturday is 9:00AM to 2:00PM.

The Project consists of following tasks:

ID	Tasks	Duration (Days)	Predecessor
A	High level analysis	3	.....
B	Selection of hardware platform	4	A
C	Installation and commissioning of hardware	2	B
D	Detailed analysis of core modules	5	B
E	Detailed analysis of supporting modules	1	C
F	Programming of core modules	3	D
G	Programming of supporting modules	6	D
H	Quality assurance of core modules	4	F
I	Quality assurance of supporting modules	4	G
J	Core module training	2	D,H
K	Development and QA of accounting reporting	1	I
L	Development and QA of management reporting	1	J
M	Development of Management Information System	1	K
N	Detailed training	4	I,J,K,M

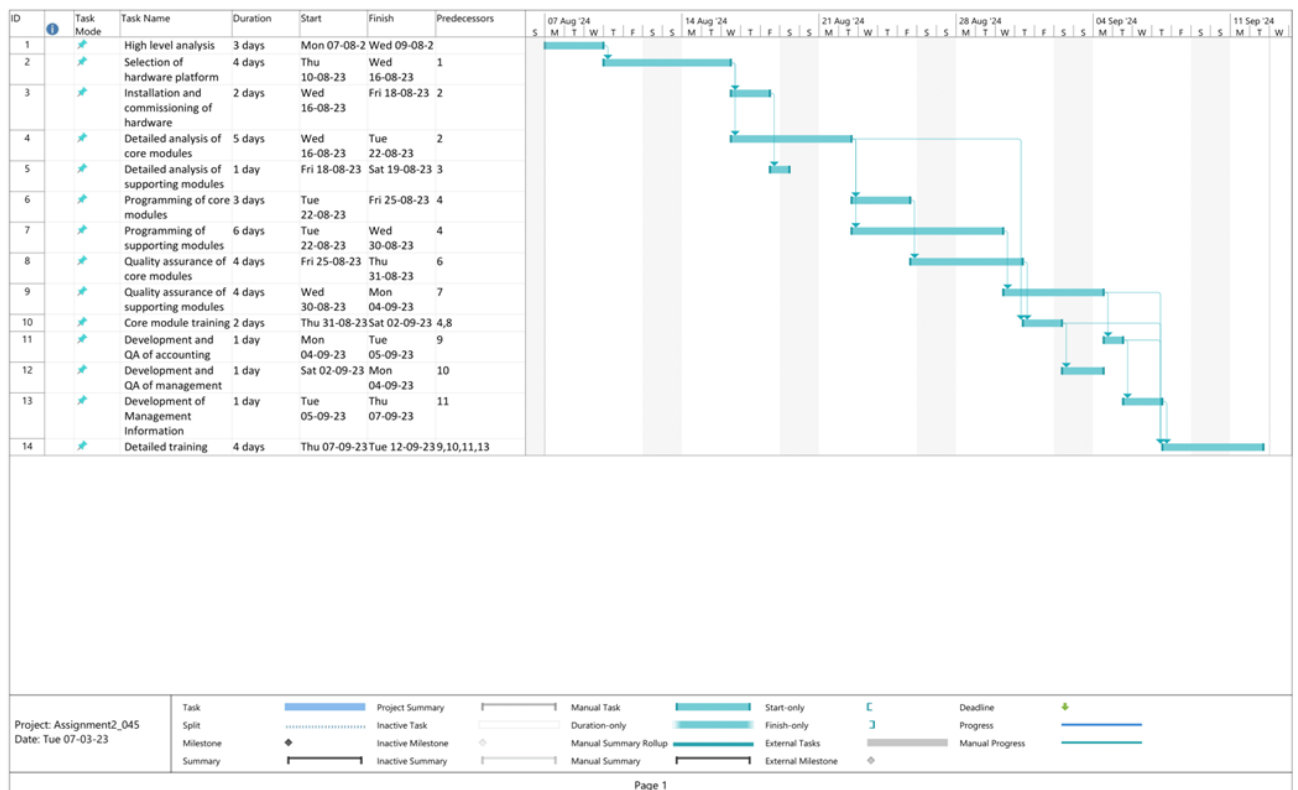
Each employee gets the following statutory holidays:

1<sup>st</sup> January (New Year's Day );23<sup>rd</sup> January(Birthday of Netaji ); 26<sup>th</sup> January(Republic Day & Shree Panchami); 8<sup>th</sup> March(Doljatra); 7<sup>th</sup> April(Good Friday); 15<sup>th</sup> April( Bengali New YearDay); 21<sup>st</sup> April (Id-Ul-Fitar); 1<sup>st</sup> May(May Day); 28<sup>th</sup> June (Id-Uz- Zoha); 29<sup>th</sup> July (Maharaum); 15<sup>th</sup> August(Independence Day);6<sup>th</sup> September (Janmashtami);2<sup>nd</sup> October(Gandhi's Birthday); 14<sup>th</sup> October (Mahalaya); 20<sup>th</sup> October to 28<sup>th</sup> October (Durga Puja); 12<sup>th</sup> November (Kalipuja & Birthday of Guru Nanak); 15<sup>th</sup> November (bhatridwitiya); 25<sup>th</sup> December(Christmas Day).

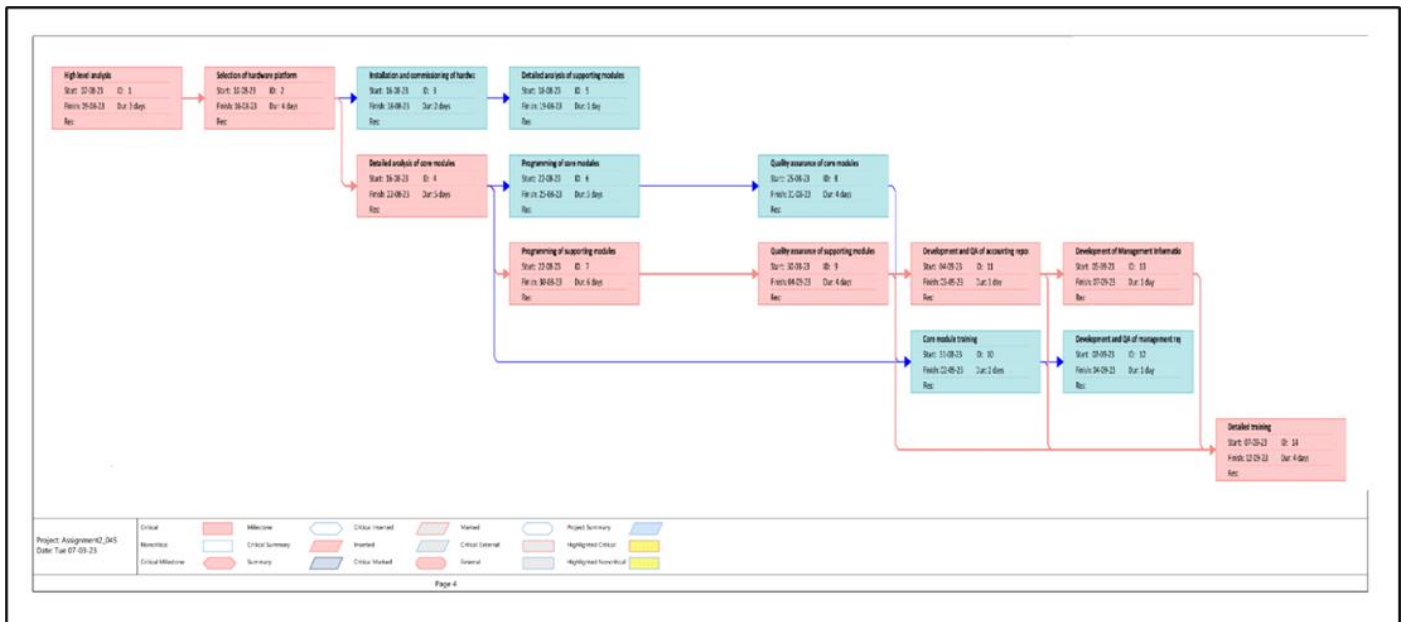
Change the 'Standard (Project Calendar)' calendar to a customized one (e.g. Roll\_CompanyName) in order to reflect these company-wide working and non-working times as well.

- 1) Draw the Gantt chart view.
- 2) Draw the Network Diagram and indicate the Critical path and indicate it in Red.
- 3) Find also the Minimum time of completion for the Project.

## Project Tools-



## Network Diagram-



The minimum time of completion the project: -

High Level Analysis + Selection of Hardware Platform + Detailed analysis of core Modules +  
Programming of Supporting Modules + Quality assurance of supporting modules +  
Development and QA of accounting reporting + Development of Management Information +  
Detailed training

$$= 3 + 4 + 5 + 6 + 4 + 1 + 1 + 4 = 28 \text{ days}$$

## Assignment-14

Assume a Project of Intelligent GST Accounting Software will start on 2<sup>nd</sup> September 2023. The fiscal year of the manufacturing company starts on 1st April. The company works by a standard time from 9.30AM to 1.00PM (pre-lunch) and 2:30 PM to 6:00 PM (post-lunch). The working hour for Saturday is 8:00AM to 1:00PM.

The 2nd and 4th Saturday of the month are the Bi-Weekly holiday.

The Project consists of following tasks:

ID	Tasks	Duration (Days)	Predecessor
A	High level analysis	3	.....
B	Selection of hardware platform	3	A
C	Installation and commissioning of hardware	2	B
D	Detailed analysis of core modules	5	B
E	Detailed analysis of supporting modules	2	C
F	Programming of core modules	3	D, C
G	Programming of supporting modules	5	D
H	Quality assurance of core modules	4	F
I	Quality assurance of supporting modules	3	G
J	Core module training	2	D,H
K	Development and QA of accounting reporting	1	I
L	Development and QA of management reporting	2	J
M	Development of Management Information System	2	K
N	Detailed training	4	I,J,K

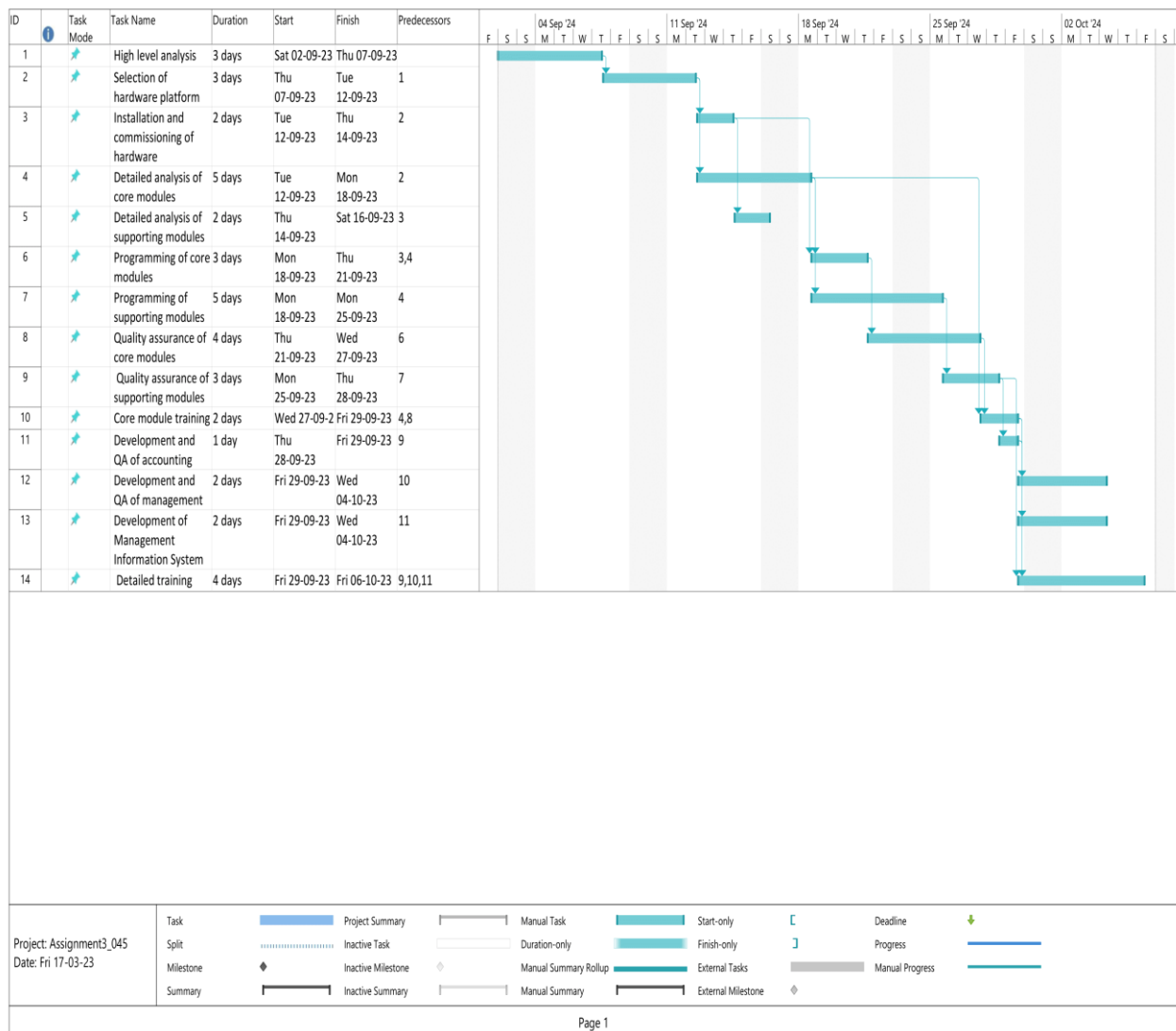
Each employee gets the following statutory holidays:

1<sup>st</sup> January (New Year's Day ); 23<sup>rd</sup> January (Birthday of Netaji ); 26<sup>th</sup> January (Republic Day & Shree Panchami); 8<sup>th</sup> March (Doljatra); 7<sup>th</sup> April (Good Friday); 15<sup>th</sup> April ( Bengali New Year Day); 21<sup>st</sup> April (Id-Ul-Fitar); 1<sup>st</sup> May (May Day); 28<sup>th</sup> June (Id-Uz- Zoha); 29<sup>th</sup> July (Maharajm); 15<sup>th</sup> August (Independence Day); 6<sup>th</sup> September (Janmashtami); 2<sup>nd</sup> October (Gandhi's Birthday); 14<sup>th</sup> October (Mahalaya); 20<sup>th</sup> October to 28<sup>th</sup> October (Durga Puja); 12<sup>th</sup> November (Kalipuja & Birthday of Guru Nanak); 15<sup>th</sup> November (bhatridwitiya); 25<sup>th</sup> December (Christmas Day).

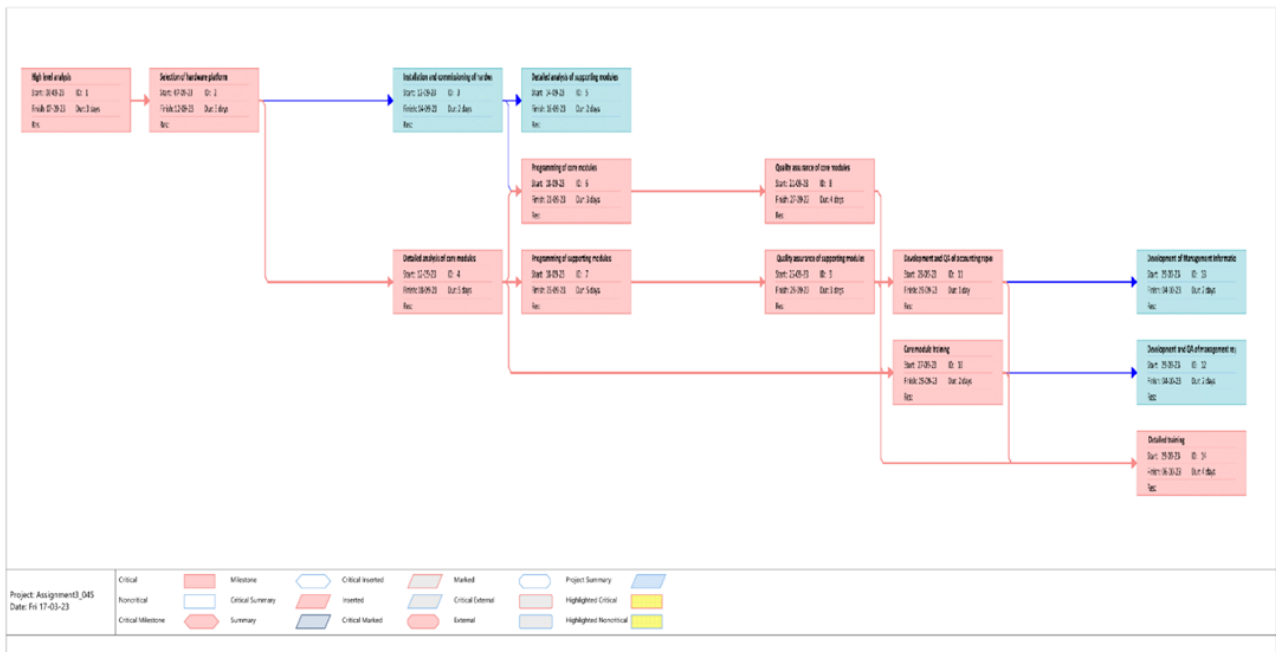
Change the 'Standard (Project Calendar)' calendar to a customized one (e.g. Roll\_CompanyName) in order to reflect these company-wide working and non-working times as well.

1. Draw the Gantt chart view.
2. Draw the Network Diagram and indicate the Critical path and indicate it in Red.
3. Find also the Minimum time of completion for the Project

# Project Tools-



## Network Diagram-



**The minimum time of completion the project: -**

High level analysis + Selection of hardware platform + Detailed analysis of core modules + Programming of core modules + Programming of supporting modules + Quality assurance of core modules+ Quality assurance of supporting modules + Core module training + Development and QA of accounting reporting + Development and QA of management reporting  
 $3 + 3 + 5 + 3 + 5 + 4 + 3 + 2 + 1 + 2 = 31$  days.



## Assignment-15

Assume a Project of KnowSolution Software Development will start on 13<sup>th</sup> February 2023. The fiscal year of the manufacturing company starts on 1st April. The company works by a standard time from 7.30AM to 5:00PM on Monday, 8.00 AM to 5:30PM on Tuesday, 9.00AM to 7.00PM on Wednesday, Thursday and Friday. The working hour for Saturday is 9:00AM to 1:00PM.

The Project consists of following tasks:

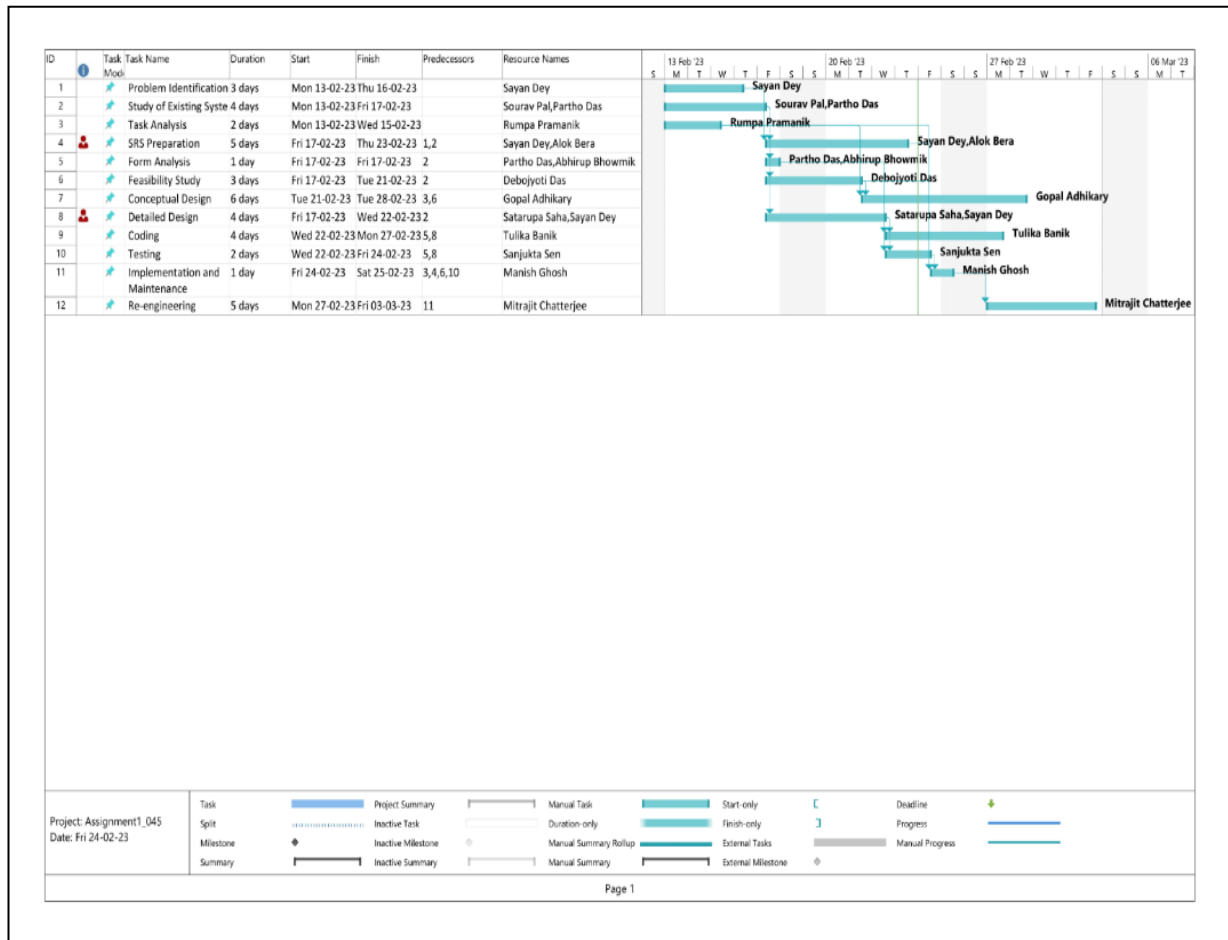
ID	Tasks	Duration (Days)	Predecessor	Resources
A	Problem Identification	3	.....	Sayan Dey
B	Study of Existing System	4	.....	Sourav Pal, Partho Das
C	Task Analysis	2	.....	Rumpa Pramanik
D	SRS Preparation	5	A,B	Sayan Dey, Alok Bera
E	Form Analysis	1	B	Partho Das , Abhirup Bhowmik
F	Feasibility Study	3	B	Debojyoti Das
G	Conceptual Design	6	F,C	Gopal Adhikary
H	Detailed Design	4	B	Satarupa Saha, Sayan Dey
I	Coding	4	E,H	Tulika Banik
J	Testing	2	E,H	Sanjukta Sen
K	Implementation and Maintenance	1	C,D,F,J	Manish Ghosh
L	Re-engineering	5	K	Mitrajit Chatterjee

Each employee gets the following statutory holidays:

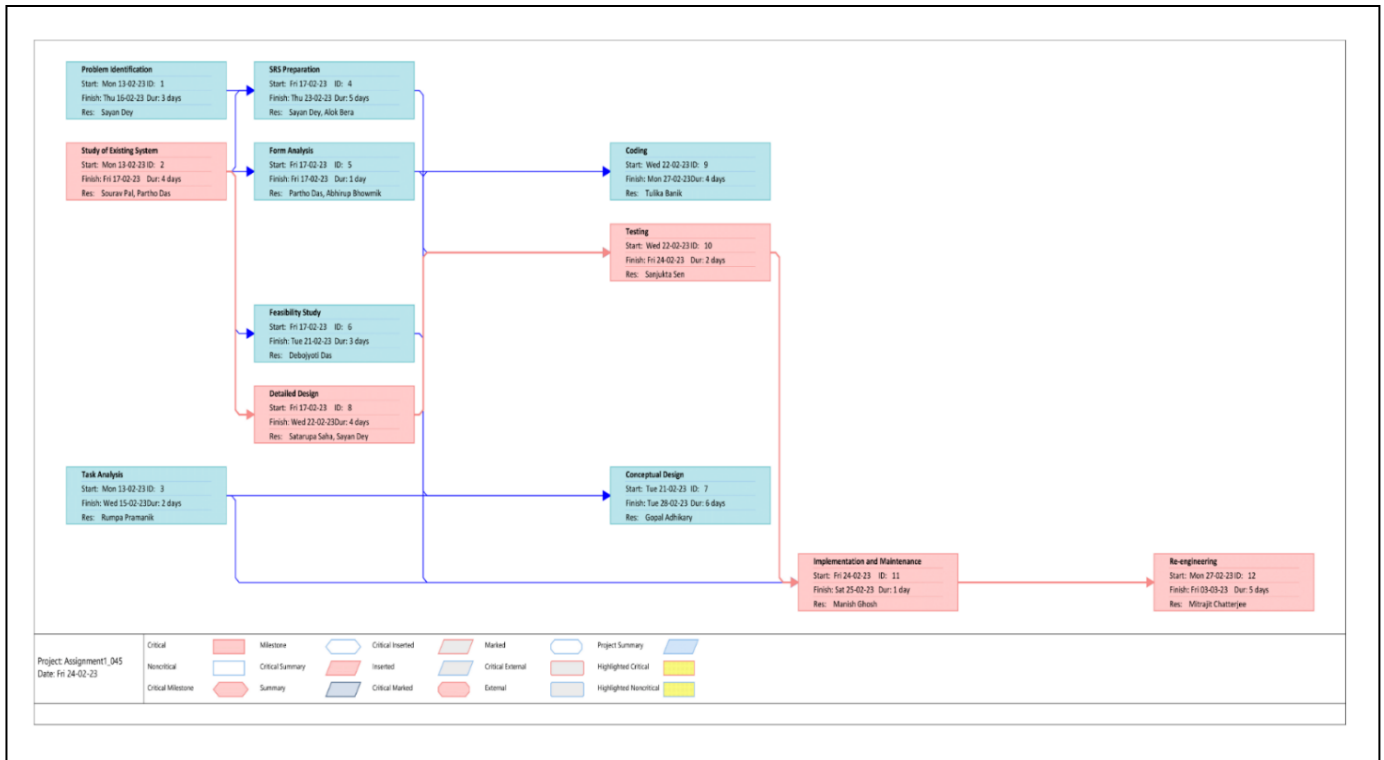
1<sup>st</sup> January(New Year's Day );23<sup>rd</sup> January ( Birthday of Netaji );26<sup>th</sup> January(Republic Day); 1<sup>st</sup> February(Shree Panchami); 13<sup>th</sup> March(Doljatra); 14<sup>th</sup> April(Good Friday);15<sup>th</sup> April( Bengali New Year Day);1<sup>st</sup> May(May Day);26<sup>th</sup> June(Id-Ul-Fitar); 7<sup>th</sup> August(University Foundation Day);14<sup>th</sup> August(Janmastami);15<sup>th</sup> August(Independence Day);19<sup>th</sup> September(Mahalaya); 26<sup>th</sup> September to 5<sup>th</sup> October(Durga Puja including Birthday of Gandhiji & Id-Uz-Zoha);18<sup>th</sup> October(Kalipuja); 21<sup>st</sup> October (Bhatriditya);4<sup>th</sup> November(Birthday of Guru Nanak); 2<sup>nd</sup> December(Fateh-Dwaz-Duham);25<sup>th</sup> December(Christmas Day) Change the 'Standard (Project Calendar)' calendar to a customized one (e.g. Roll\_CompanyName) in order to reflect these company-wide working and non-working times as well.

- 1) Draw the Gantt chart view.
- 2) Draw the Network Diagram and indicate the Critical path and indicate it in Red.
- 3) Find also the Minimum time of completion for the Project.

## Project Tools-



## Network Diagram-



**The minimum time of completion the project: -**

Study of Existing System + Detailed Design + Testing + Implementation and Maintenance + Re-engineering =  
 $4 + 4 + 2 + 1 + 5 = 16$  days

# Assignment-16

## [Level-0]

### Level 0 Data Flow Diagram for airline reservation System.

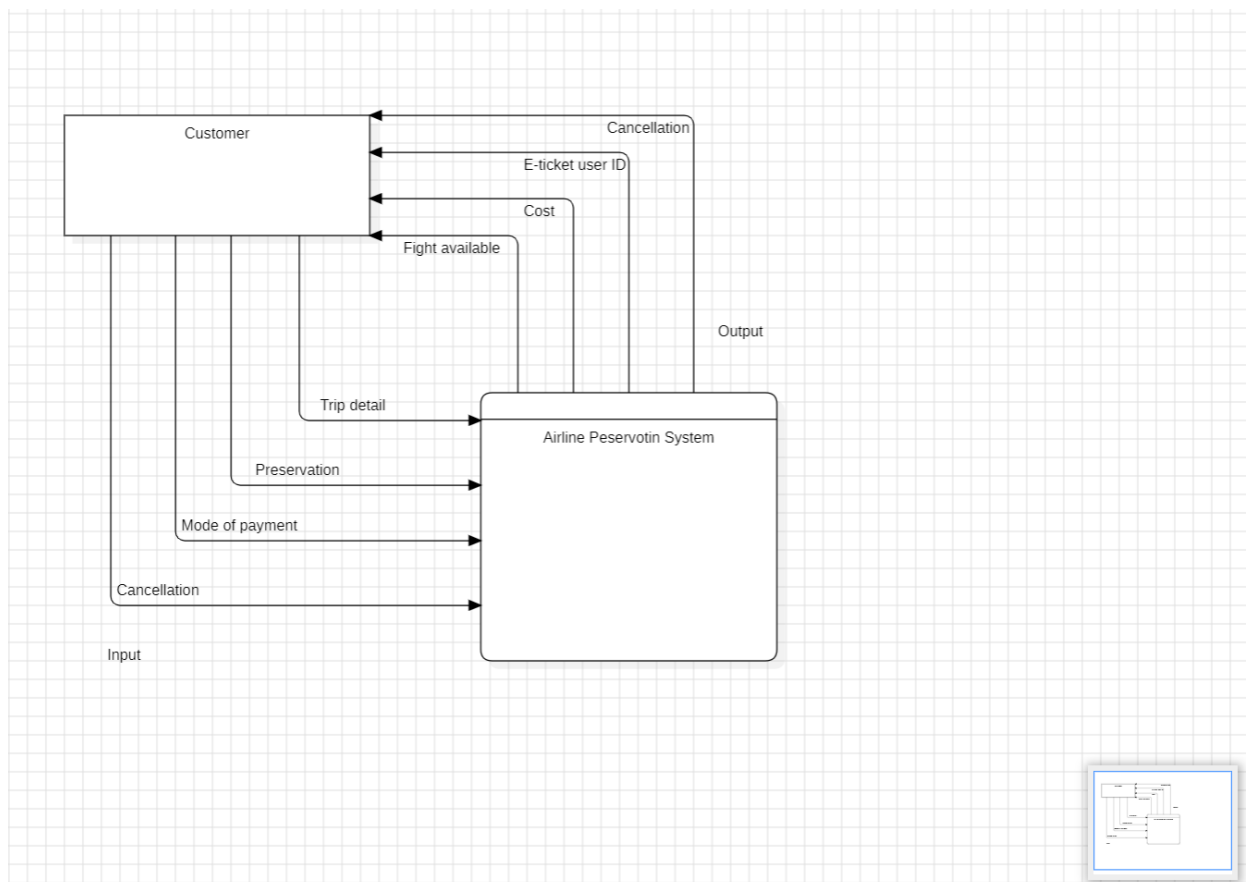
#### Description-

A Level 0 Data Flow Diagram (DFD), also known as a Context Diagram, provides a high-level overview of a system. It represents the entire system as a single process and shows how it interacts with external entities (people, organizations, or other systems) via data flows.

Key Elements of a Level 0 DFD:

1. Single Process – The whole system is represented as one process (a circle or rounded rectangle).
2. External Entities – Sources or destinations of data outside the system (drawn as rectangles).
3. Data Flows – Arrows showing data movement between the external entities and the system.

#### Output-



## [Level-1]

### Level 1 Data Flow Diagram for airline reservation System.

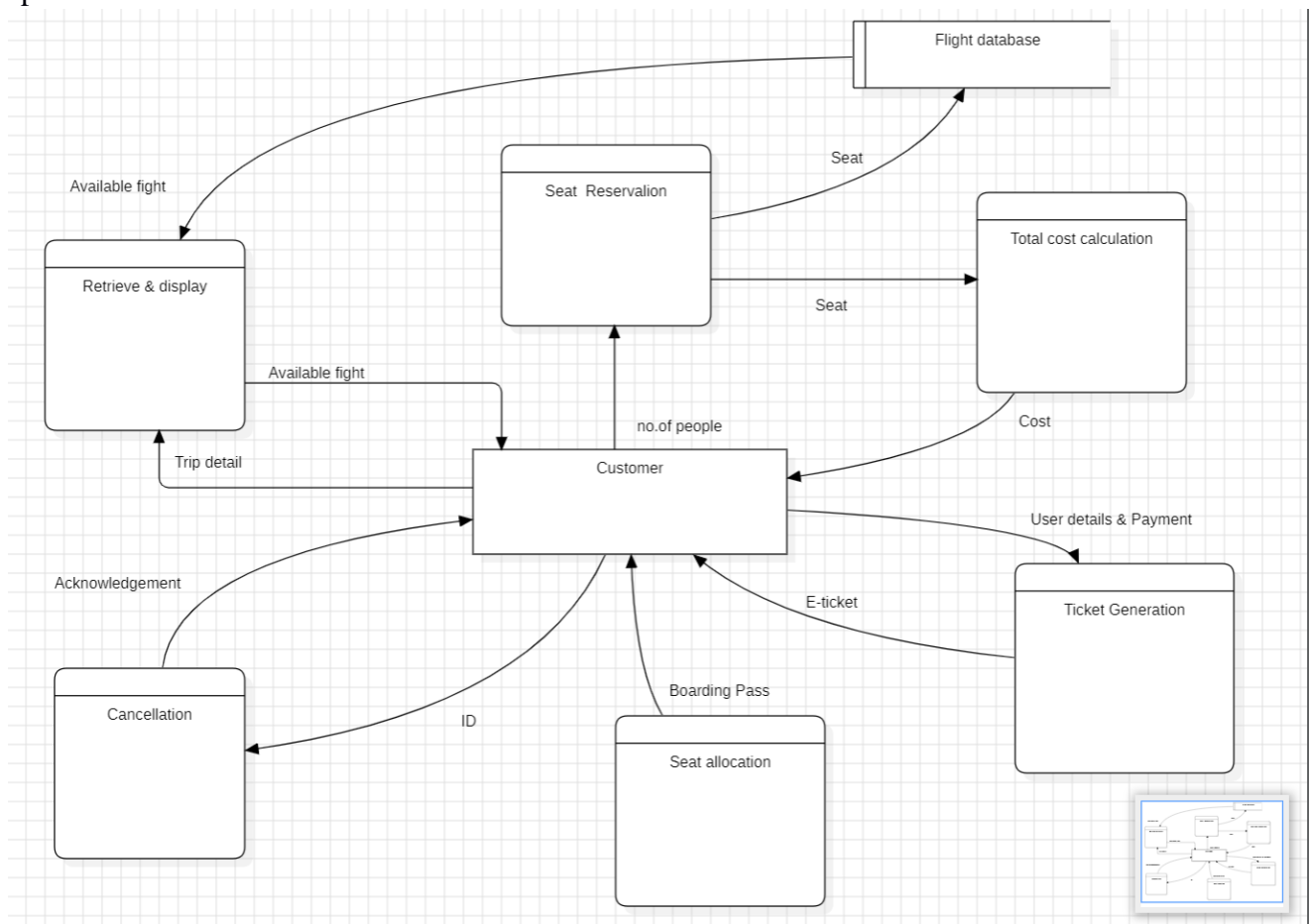
#### Description-

A **Level 1 Data Flow Diagram (DFD)** is a more detailed breakdown of a system's main process shown in the **Level 0 DFD (context diagram)**. It shows the major **sub-processes**, **data stores**, **data flows**, and **external entities** that interact with the system.

#### Key Elements of a Level 1 DFD:

1. **Processes** – Represented by circles or rounded rectangles. They break down the main system into sub-processes.
2. **External Entities** – Represented by squares. These are sources or destinations of data outside the system.
3. **Data Flows** – Represented by arrows. These show the movement of data between entities, processes, and data stores.
4. **Data Stores** – Represented by open-ended rectangles. These show where data is stored in the system.

#### Output-



## [Level-2 ]

### Level 2 Data Flow Diagram for airline reservation System.

#### Description-

A **Level 2 Data Flow Diagram (DFD)** breaks down one of the Level 1 processes into even more detailed subprocesses, data flows, and data stores. It helps in understanding how a specific process operates at a more granular level.

#### Purpose of Level 2 DFD:

- To decompose a **single Level 1 process** into more detailed steps.
- To show **internal logic**, data handling, and flow within that particular process.

#### Output-

