

# HOTEL MANAGEMENT SYSTEM

## SOFTWARE REQUIREMENT SPECIFICATION

LAB - I  
Software Requirement Specification (SRS)  
20/9/24

**1. Introduction**

1.1 Purpose of this document  
The document outlines the requirements for the development of Hotel Management System. The purpose is to define the system's functionality, behavior, and performance to streamline hotel operations like room booking, billing and reporting.

1.2 Scope : It aims to automate the tasks related to hotel operations. It will enable hotel staff to manage bookings, room availability, customer details and financial transactions efficiently.

1.3 Overview : The HOMS is an application designed for hotel operators and customers. It includes modules for room reservations, billing and customer information management. The system ensures security and flexibility while handling different types of rooms, amenities and pricing.

**2. Description**

The hotel management system allows hotel staff to manage room reservations, track check-in and check-out processes and manage customer data. Features include room availability tracking, customer record management, billing and report generation.

**3. Functional Requirements:**

- Room Booking: Allow customers to book rooms based on availability, room type and price.
- Check in and check out: track customer check-ins and check-outs updating room status in real-time.
- Room Availability: Show real-time room status and pricing.

#### 4. Interface Requirements:

- The system will have a web based interface available to customers and hotel staff.
- Integration with the database to store and retrieve customer, room and booking information.

#### 5. Performance Requirements:

- The system should support concurrent user access without performance degradation.
- The system should handle up to 500 concurrent transactions during peak times.
- Memory usage should remain below 70% during normal operation.

#### Design constraints:

- The system will be developed using Java and a MySQL database.
- The system should support both desktop and mobile devices.

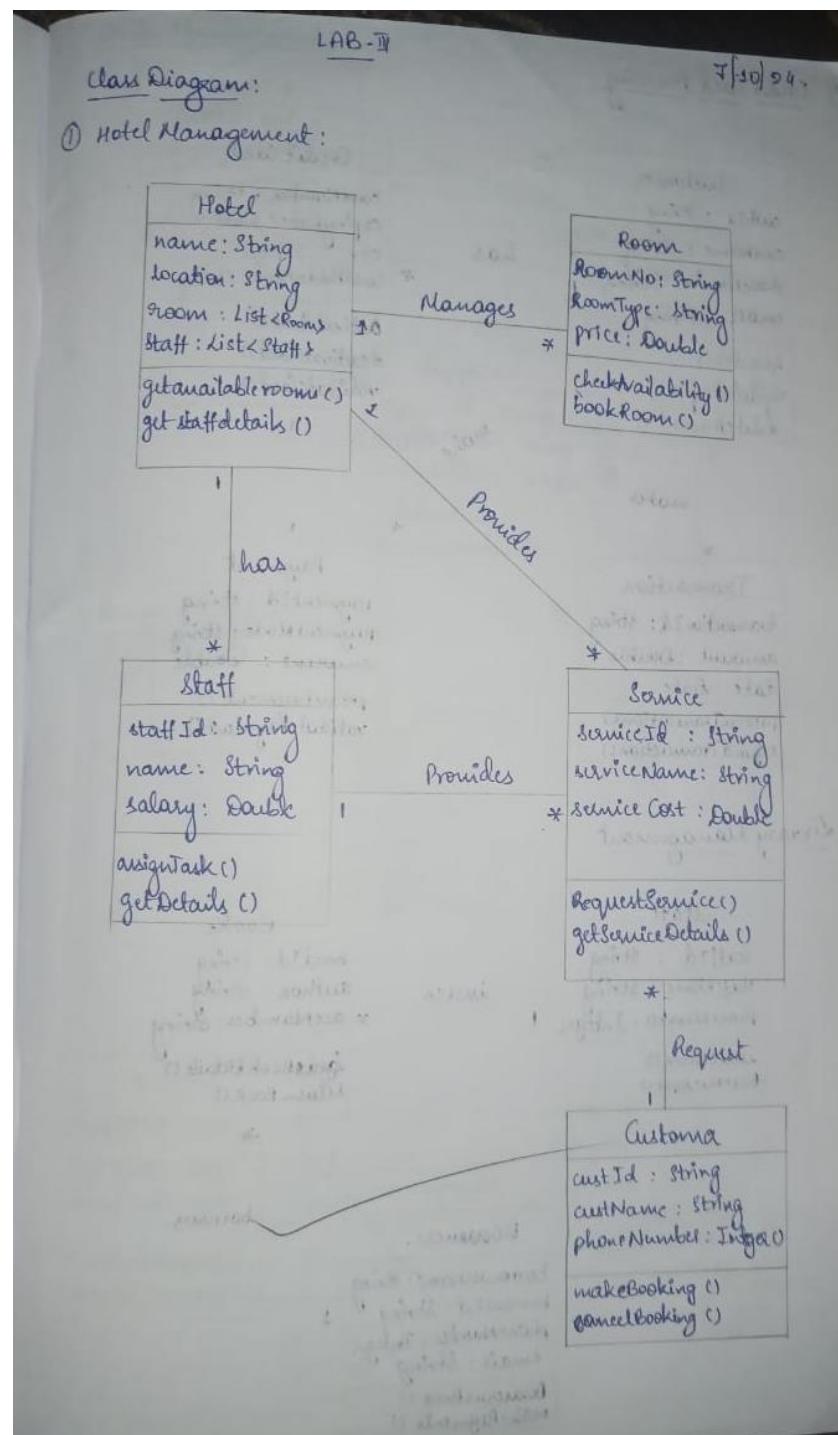
#### Non-functional Requirements:

- System downtime should not exceed 1% annually.
- The system should be able to handle an increase in customer and room capacity.
- Ensure consistent data when multiple users access or modify records.

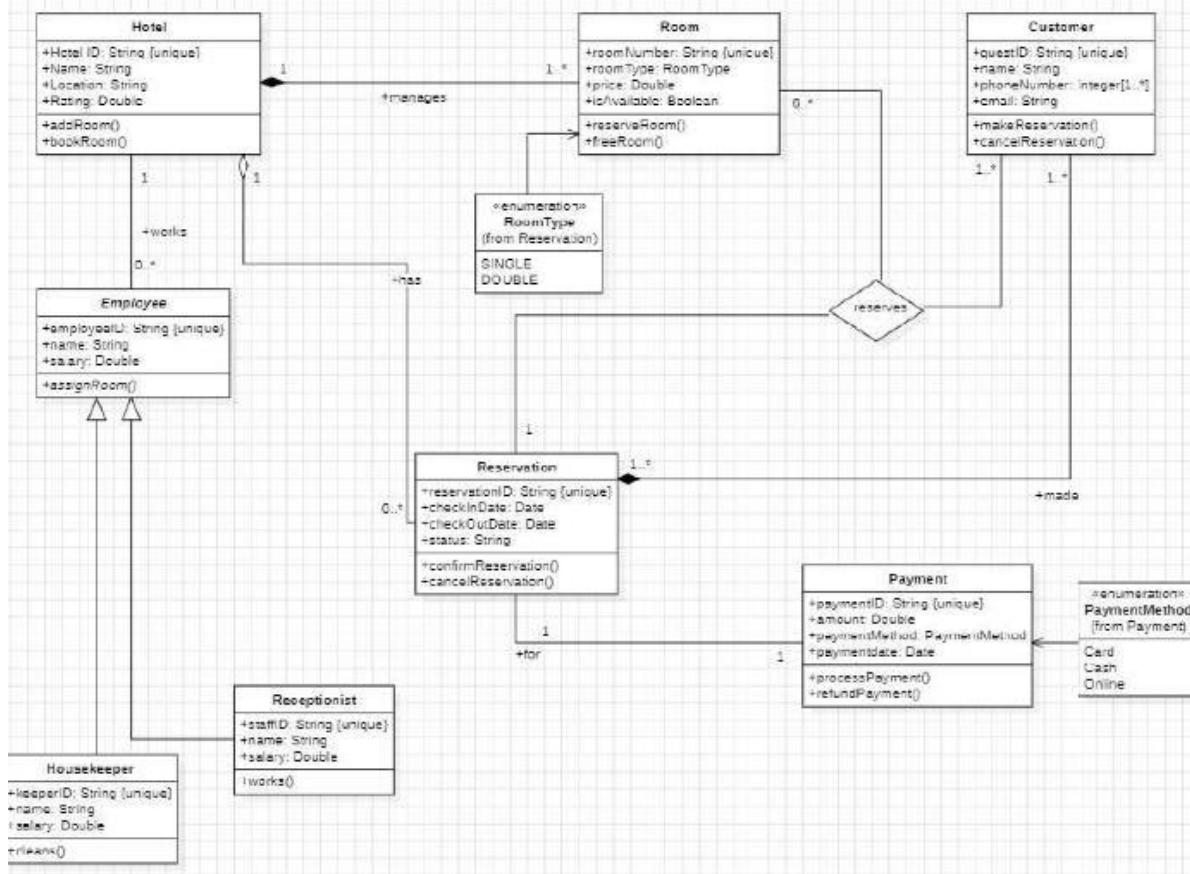
#### Budget and schedule:

- The entire project can take around 6 months for completion starting with requirement specification, design, develop, deploy and testing.
- Budget: Around \$150000 could be estimated
  - 1) Requirement specification: \$ 15,000
  - 2) Design phase: \$ 25,000
  - 3) Development: \$ 80,000
  - 4) Deploy and testing: \$ 27,500
  - 5) Maintenance: \$ 8,500

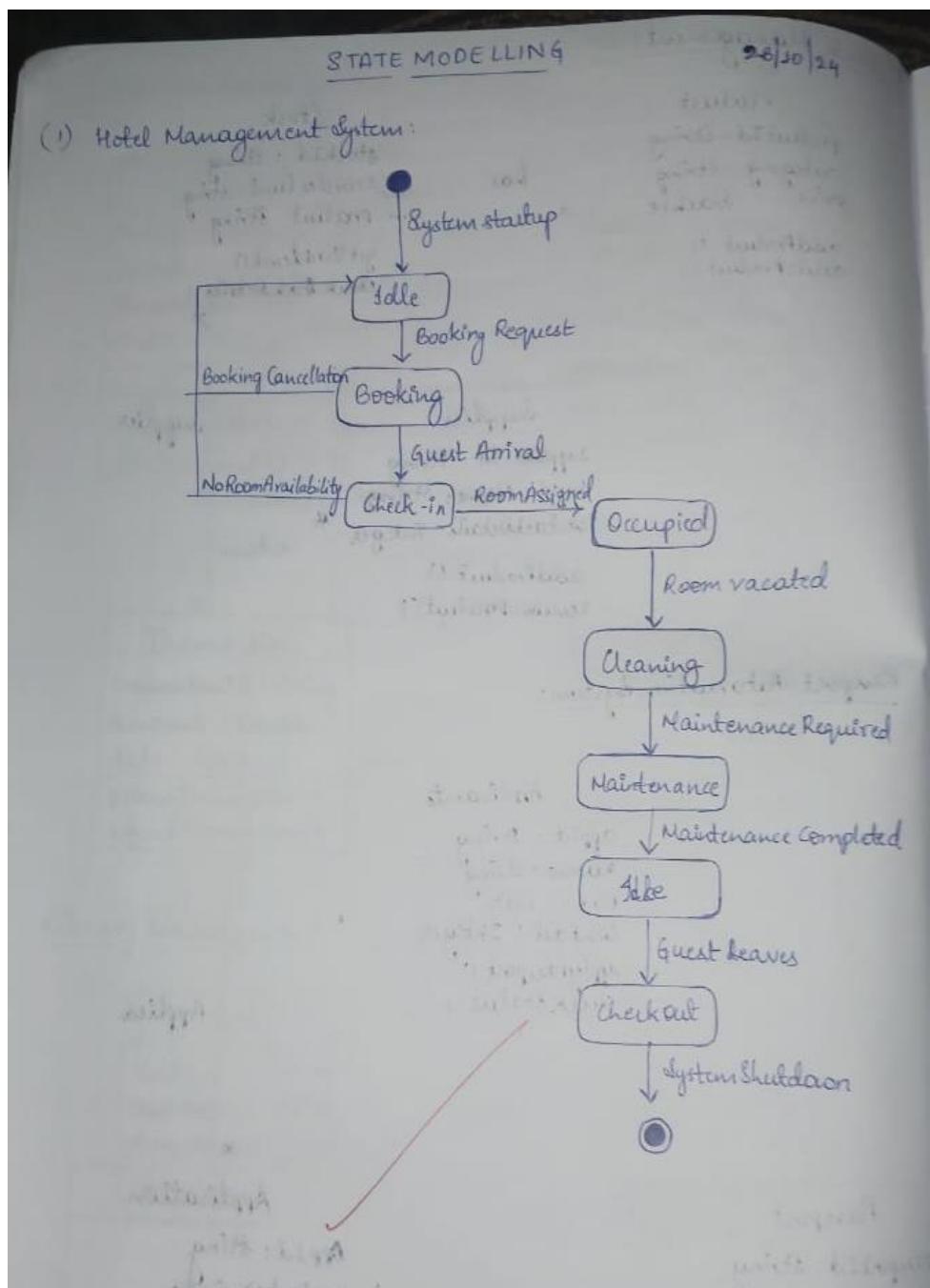
## CLASS DIAGRAM

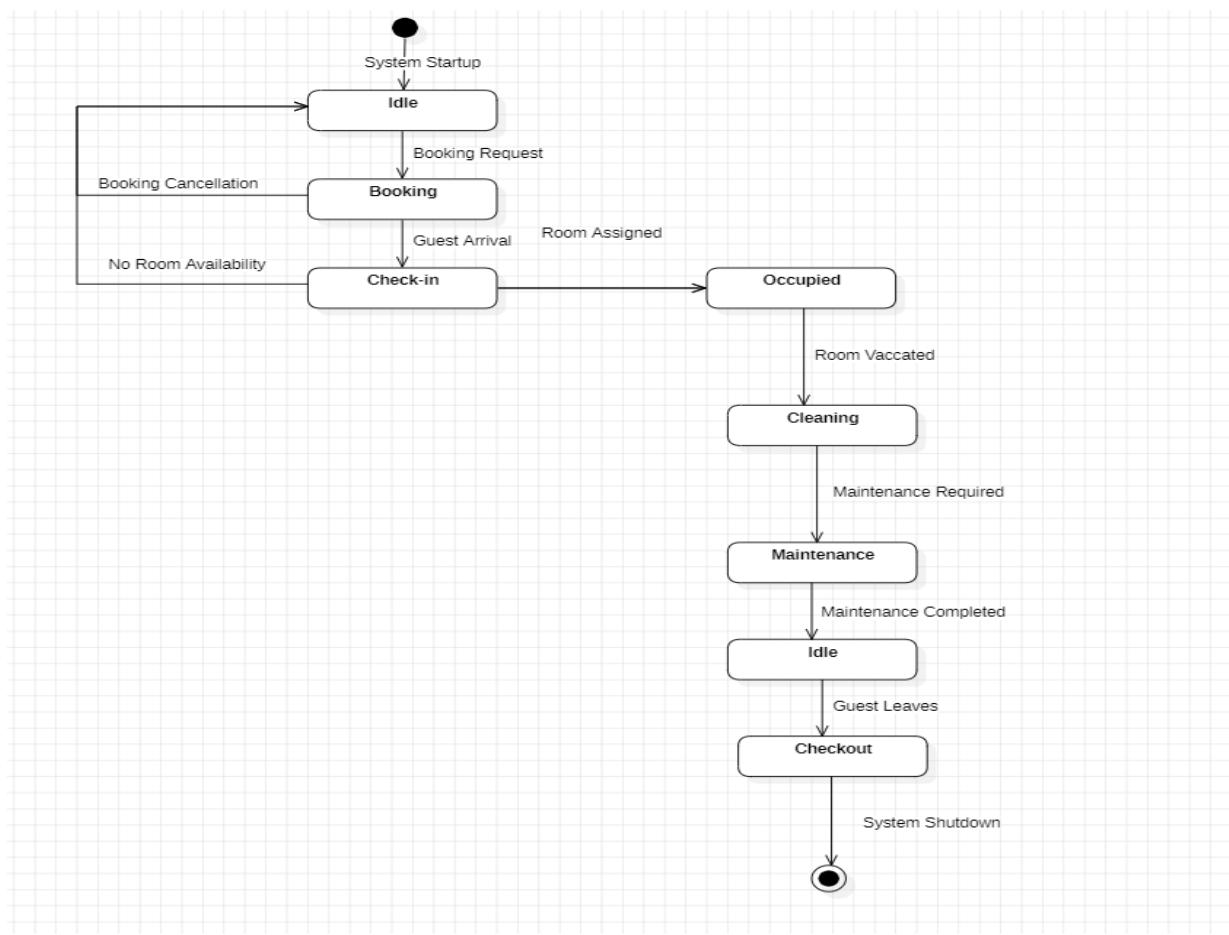


HOTEL MANAGEMENT SYSTEM - CLASS DIAGRAM

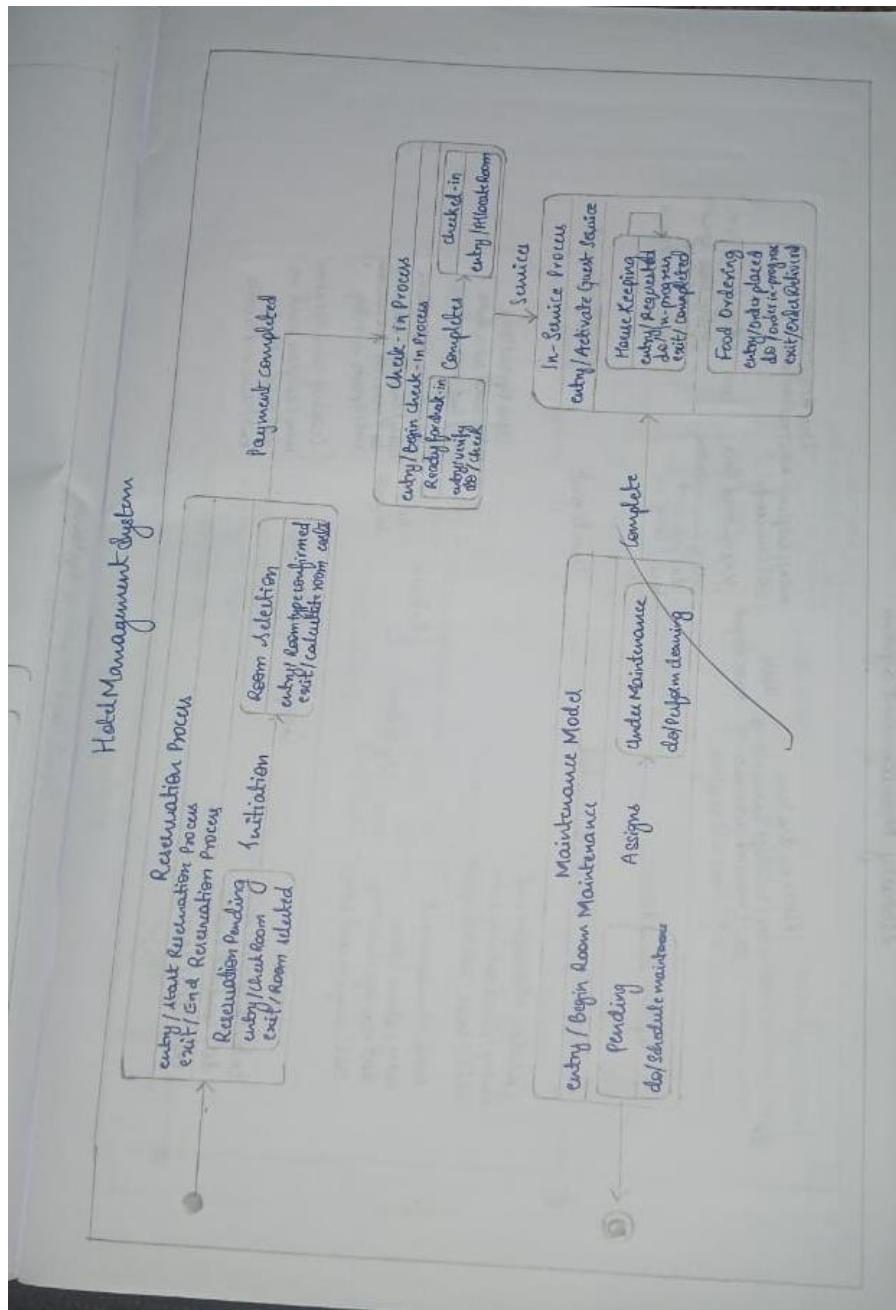


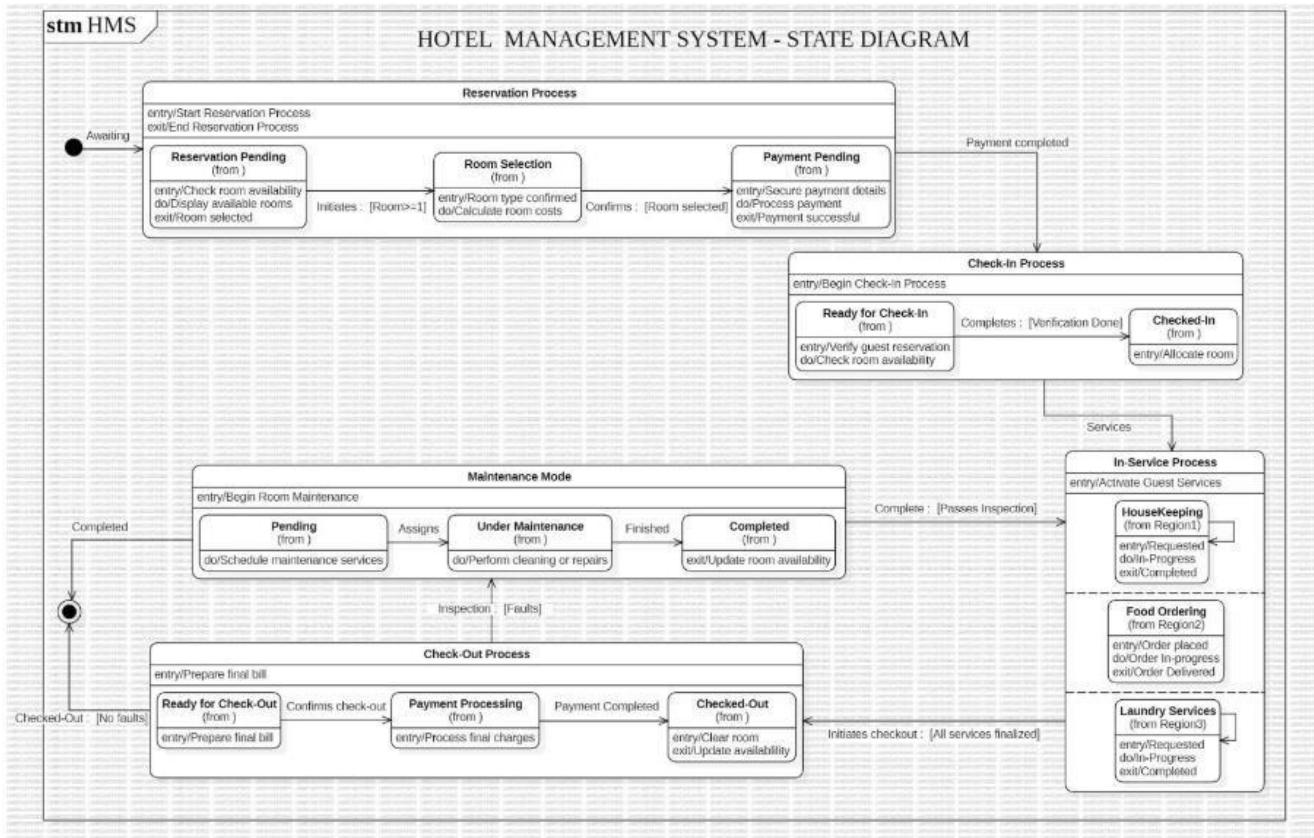
## STATE MODEL



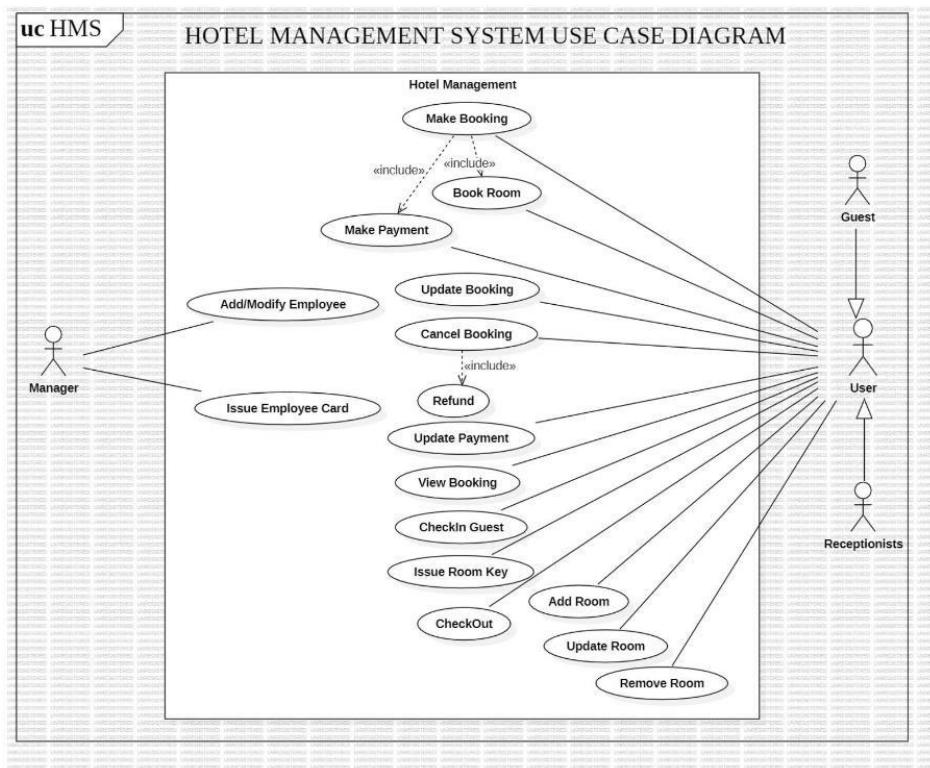
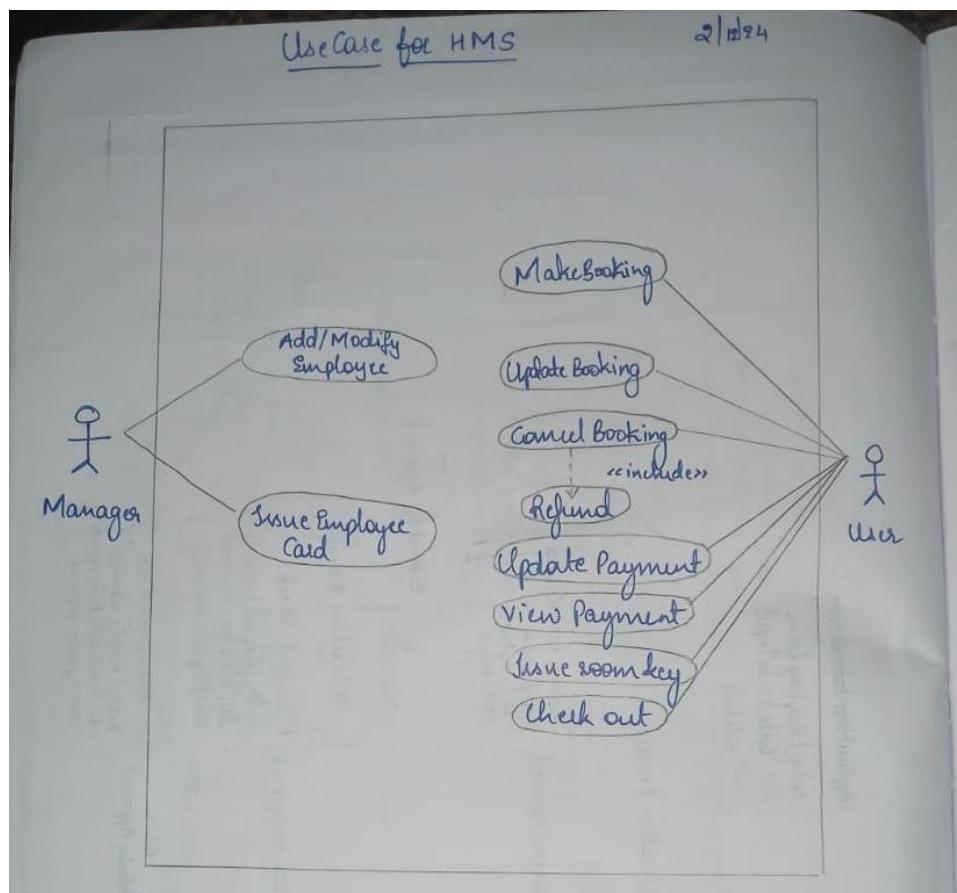


## ADVANCED STATE MODEL

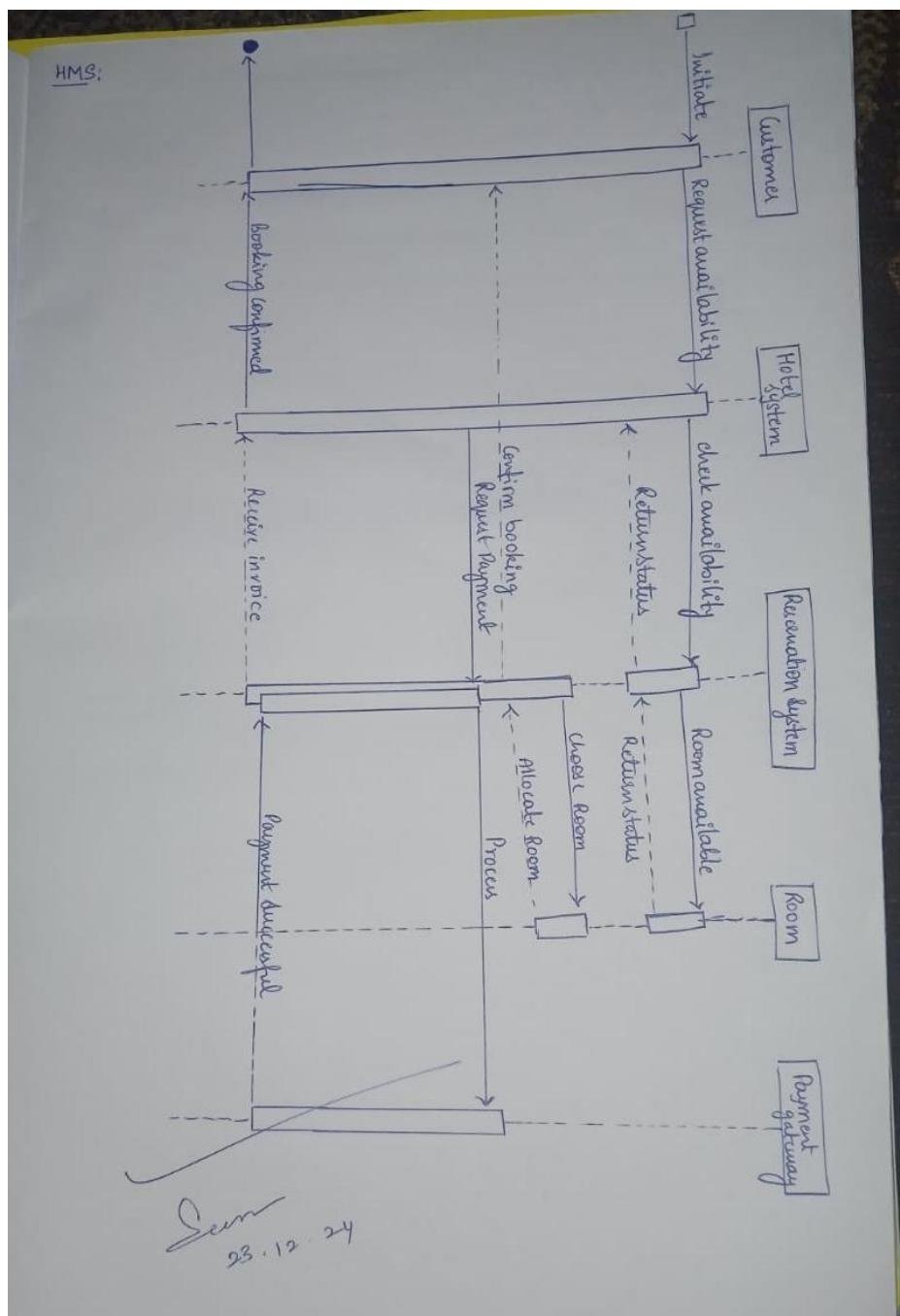


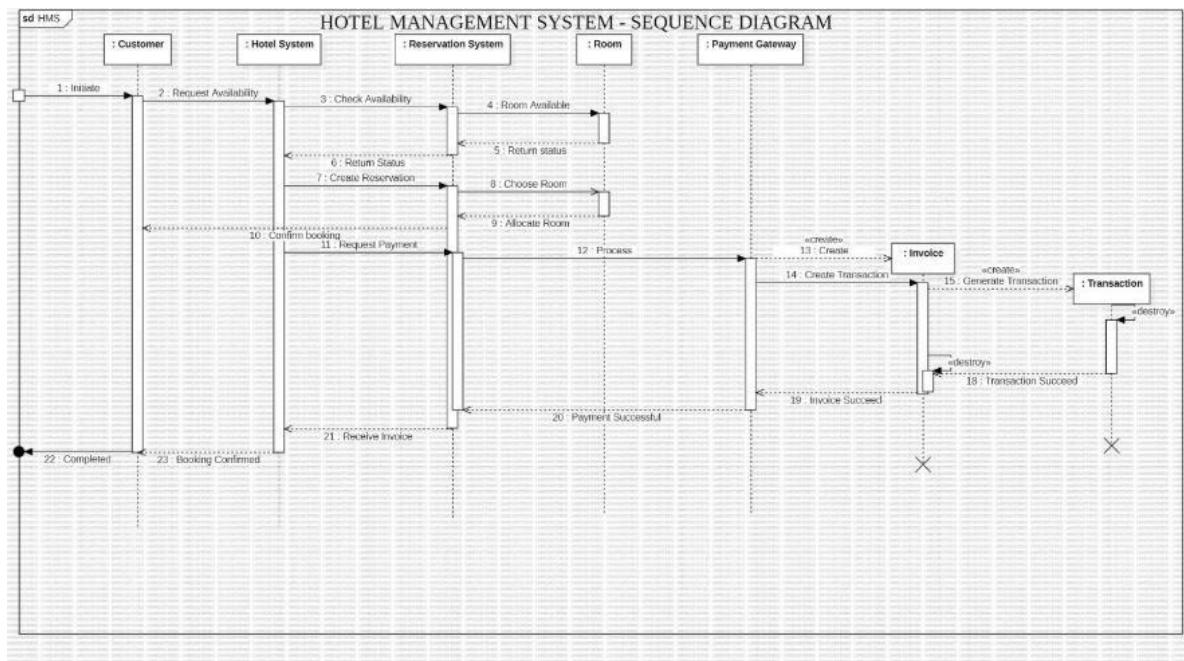


## USE CASE

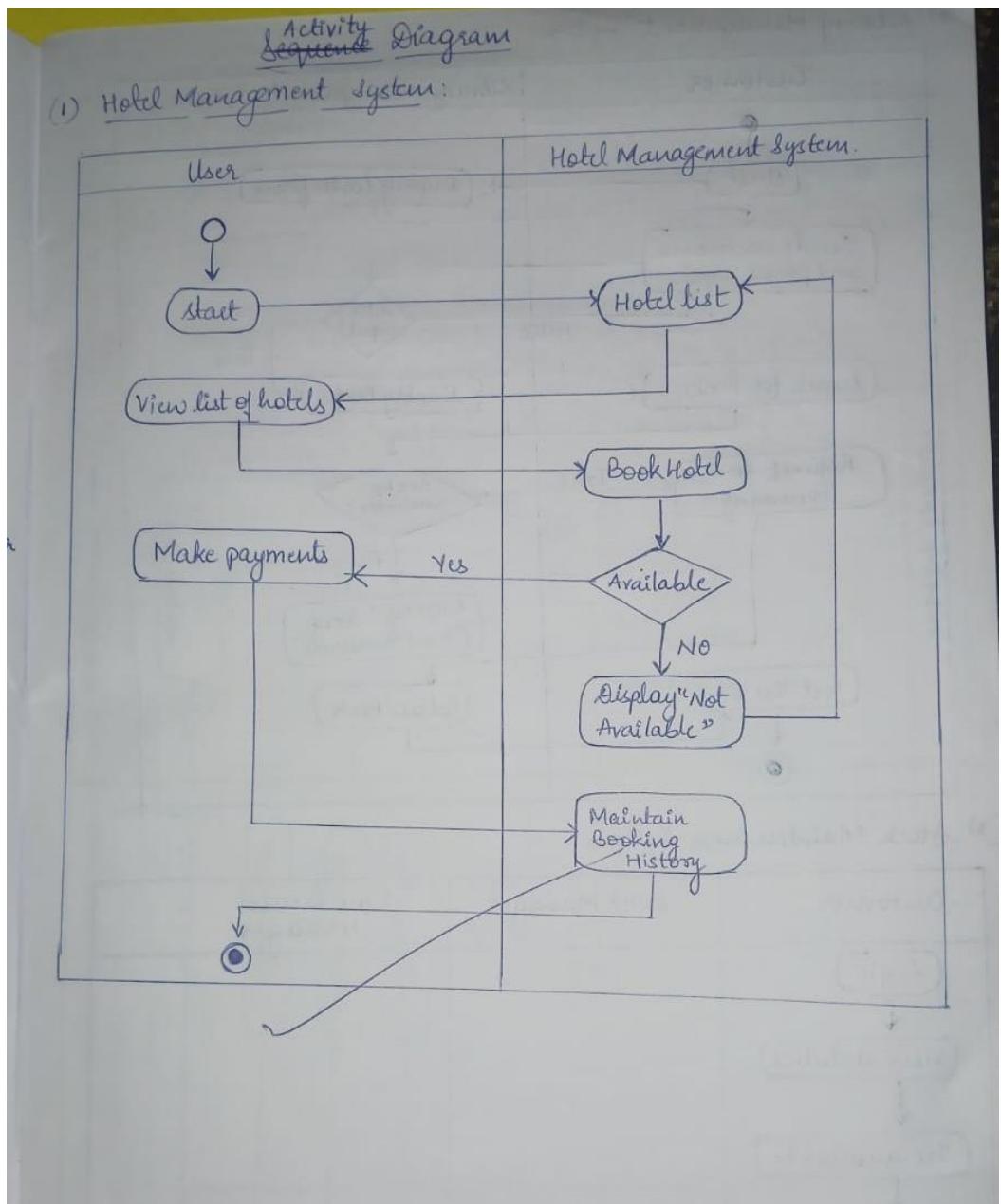


## SEQUENCE DIAGRAM



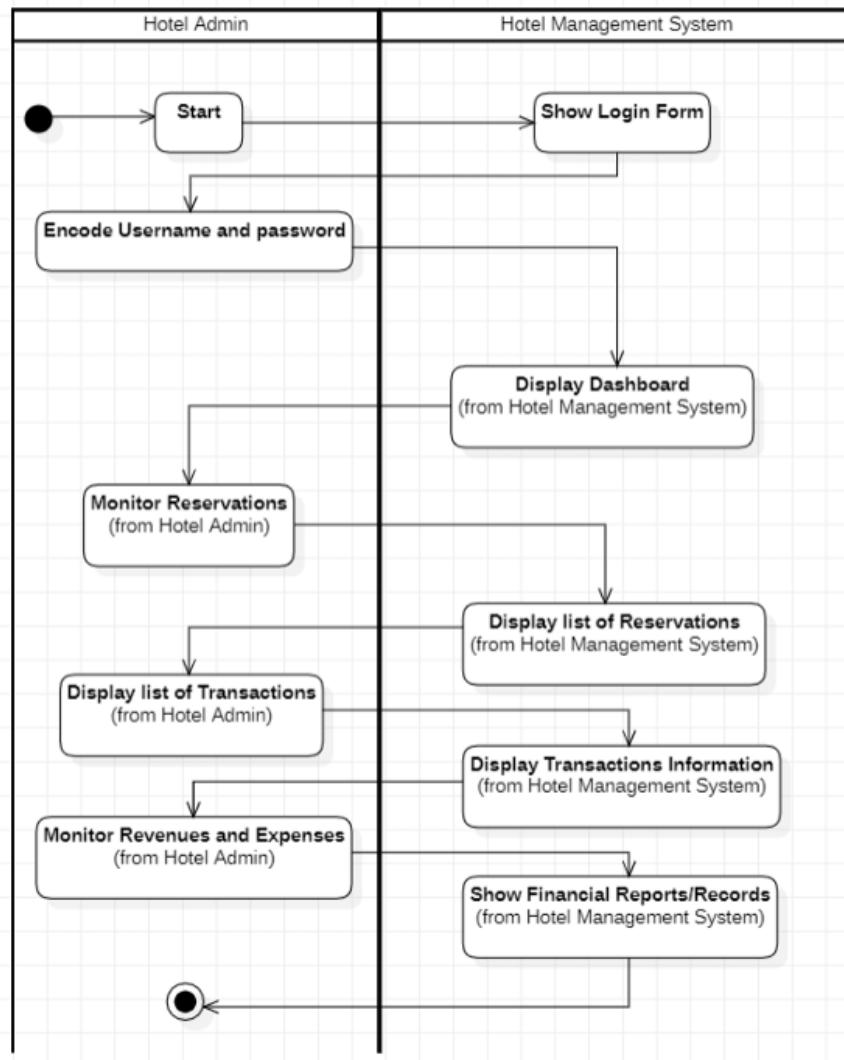


## ACTIVITY DIAGRAM



act HMS

## HOTEL MANAGEMENT SYSTEM - ACTIVITY DIAGRAM



# CREDIT CARD PROCESSING

## SOFTWARE REQUIREMENT SPECIFICATIONS

Credit Card Processing      23/9/24

**1. Introduction:**

1.1 Purpose of this document  
It describes the requirements for a credit card processing system for secure and efficient card transactions, ensuring accuracy and security.

1.2 Scope of this document  
It can handle credit card payments, authorizations, and settlements. It provides fraud detection and ensures security.

1.3 The system processes payments, verifies card details, and generates transaction records. It ensures the secure transfer of information between banks, cardholders, and users.

**2. General Description:**

The system allows merchant to accept credit card payment online and in-store, ensuring security and reliability. Users can include financial institution, customers, etc.

**3. Functional Requirements:**

- Authenticate transactions between card user and card issuer.
- Maintain record history.
- Process multiple transactions for efficiency.

**4. Interface Requirement:**

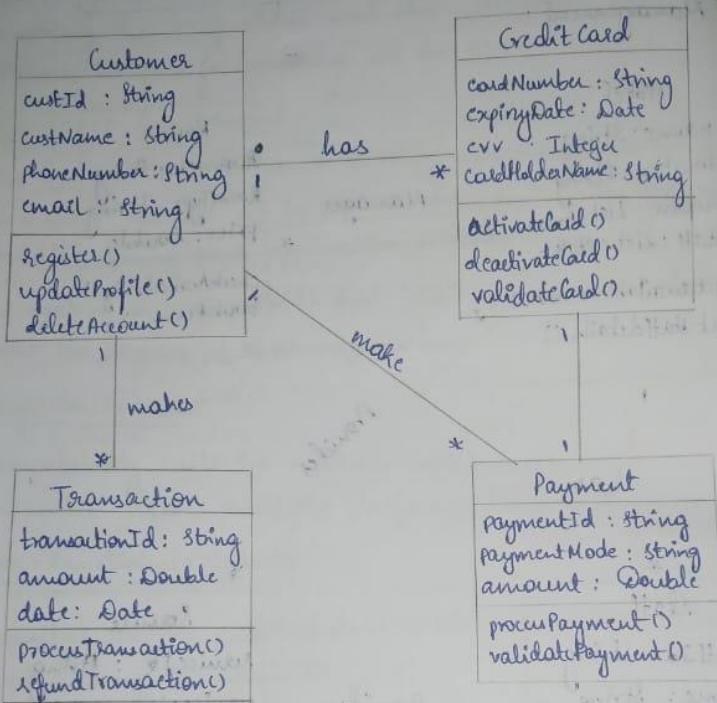
- Good UI to process payments and view transactions.
- Interface for customers to make payment and view receipts.

**5. Performance Requirement:**

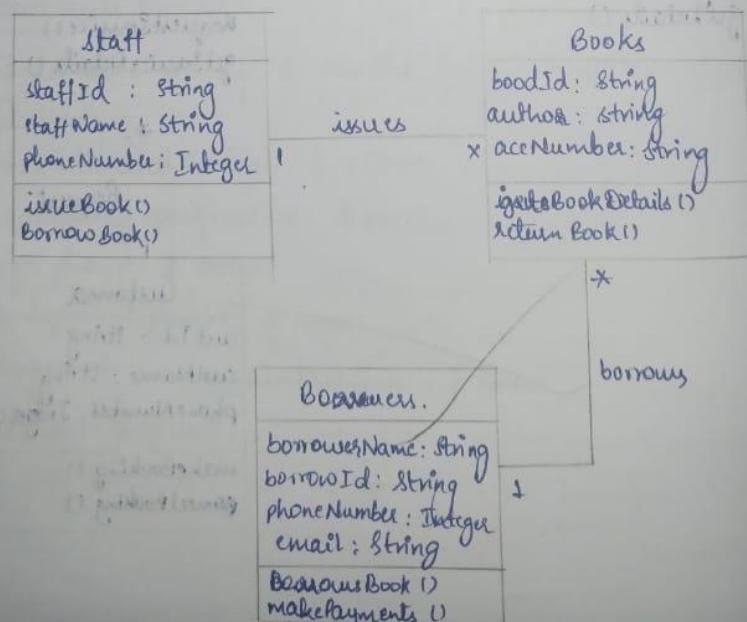
- Handle multiple transaction about 1000 transactions/second.
- Authenticate within 3 seconds.

## CLASS DIAGRAM:

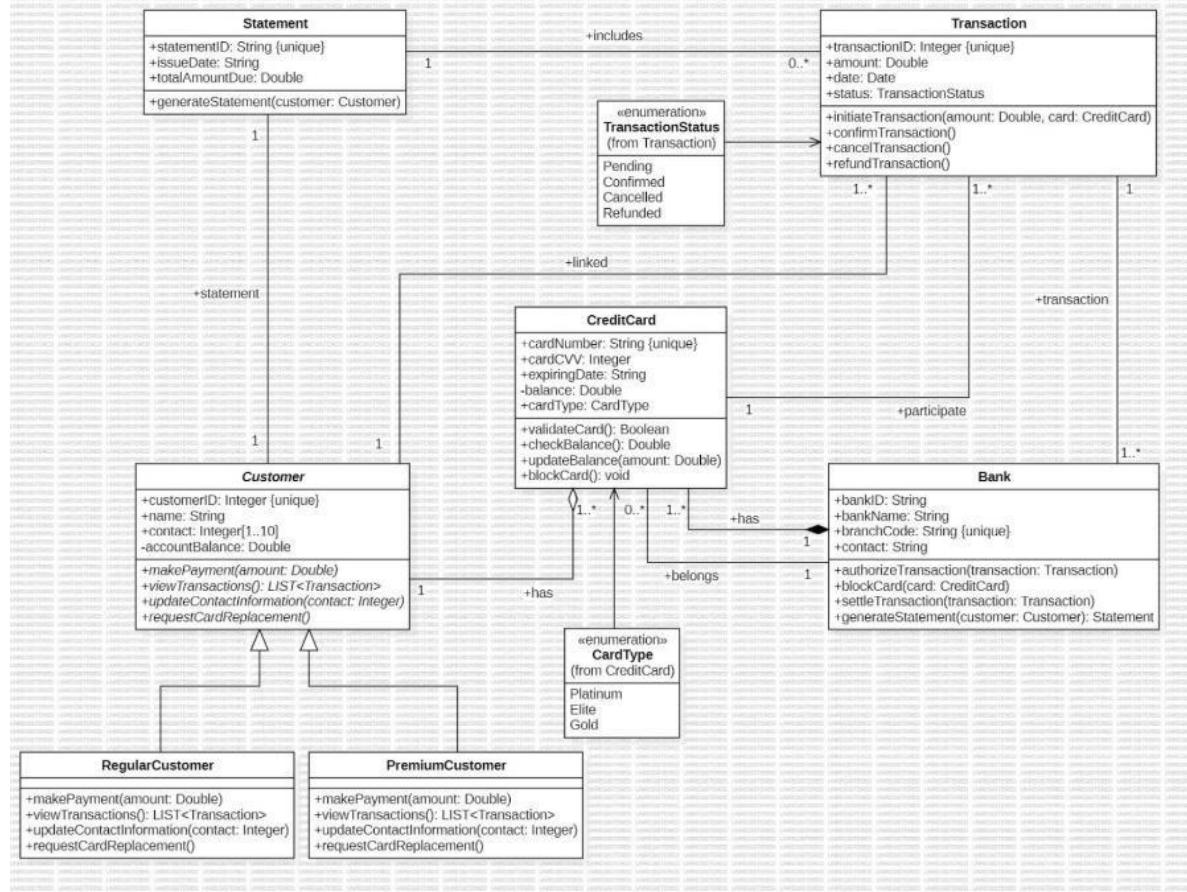
### 2) Credit Card Processing



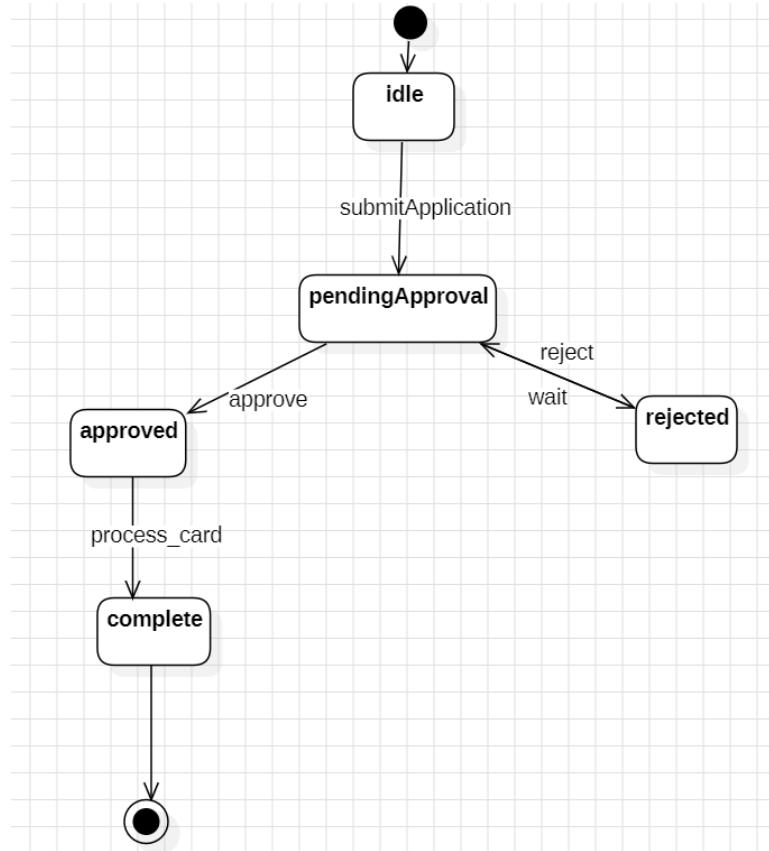
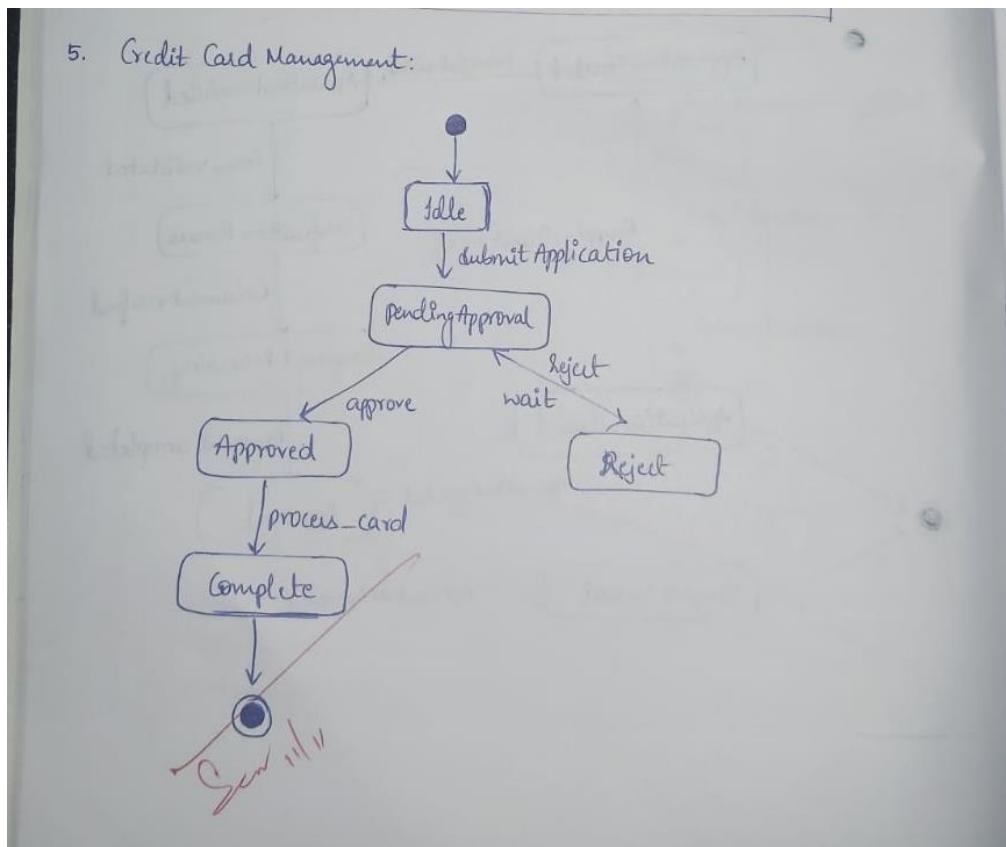
### 3) Library Management



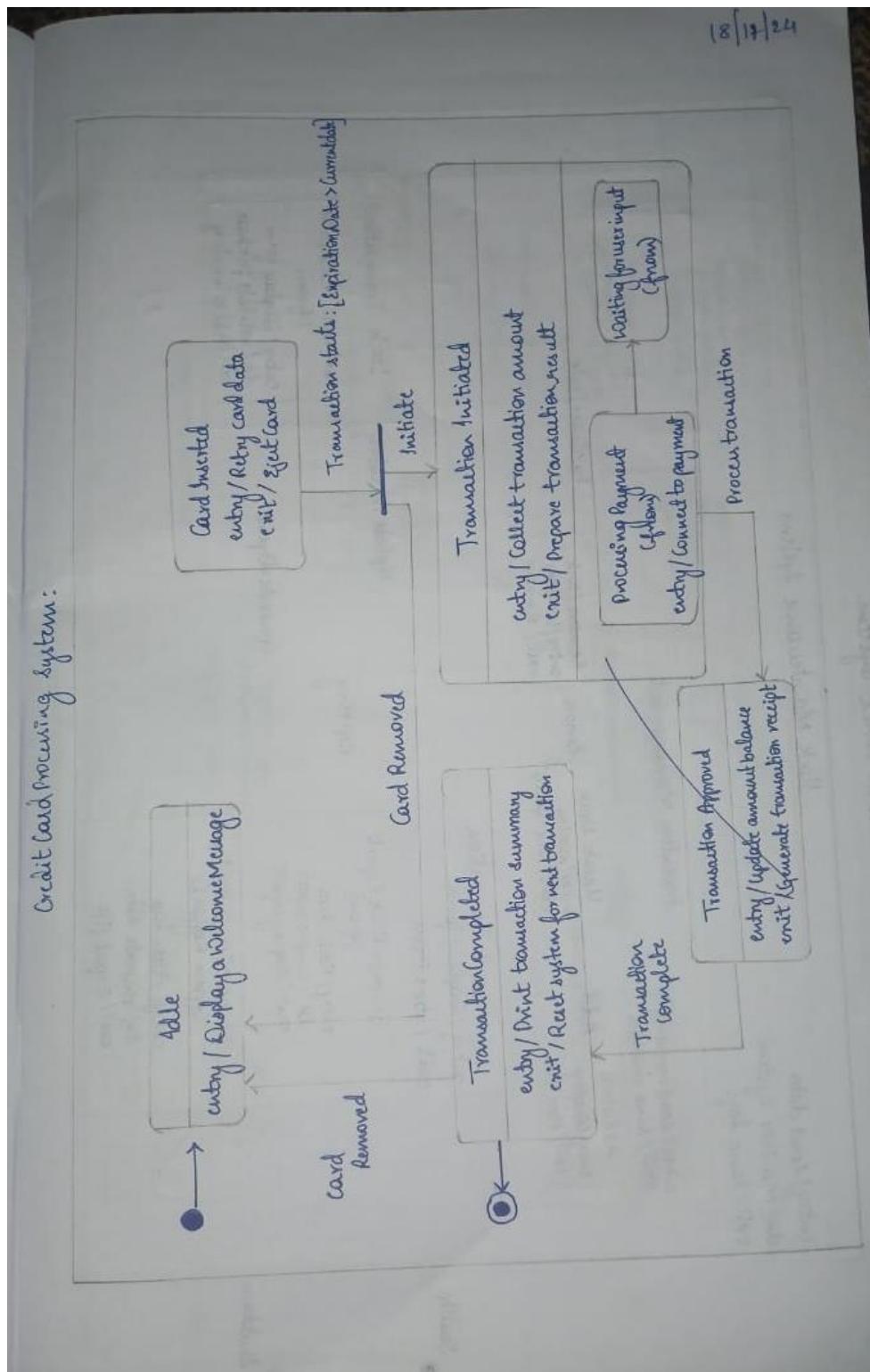
## CREDIT CARD PROCESSING SYSTEM - CLASS DIAGRAM

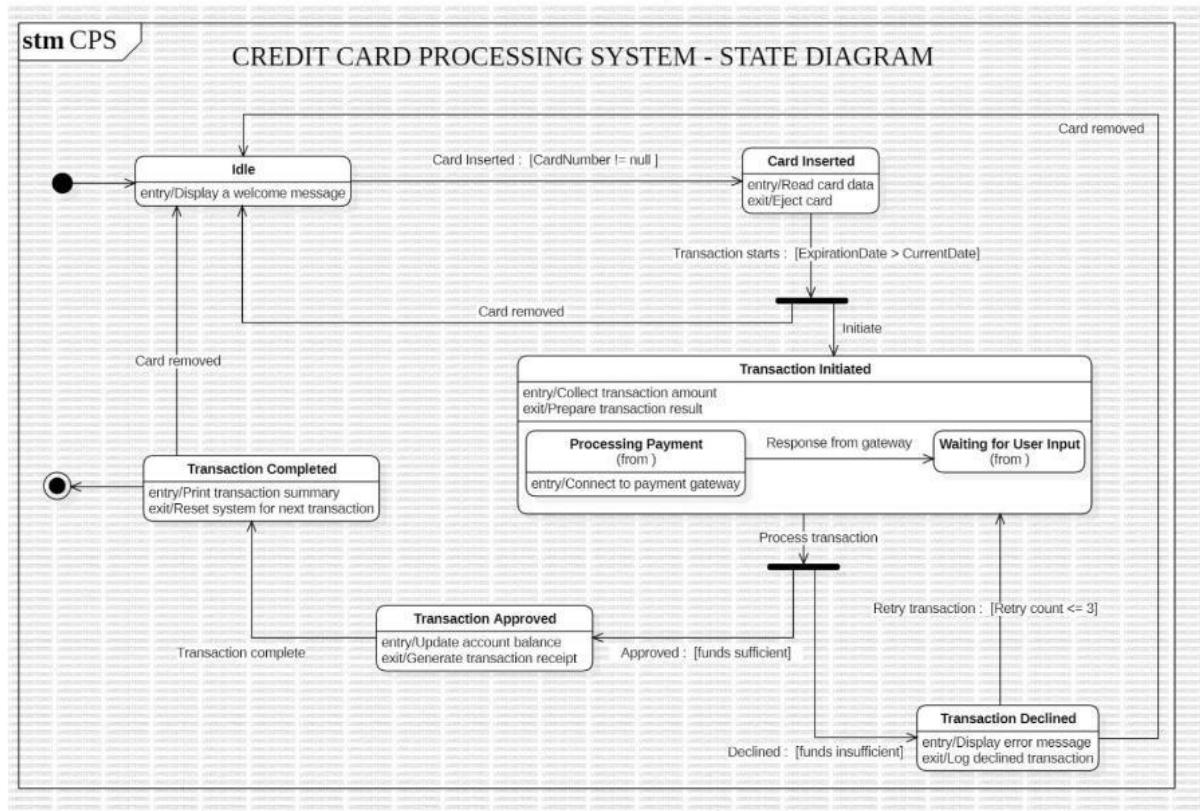


## STATE MODEL:

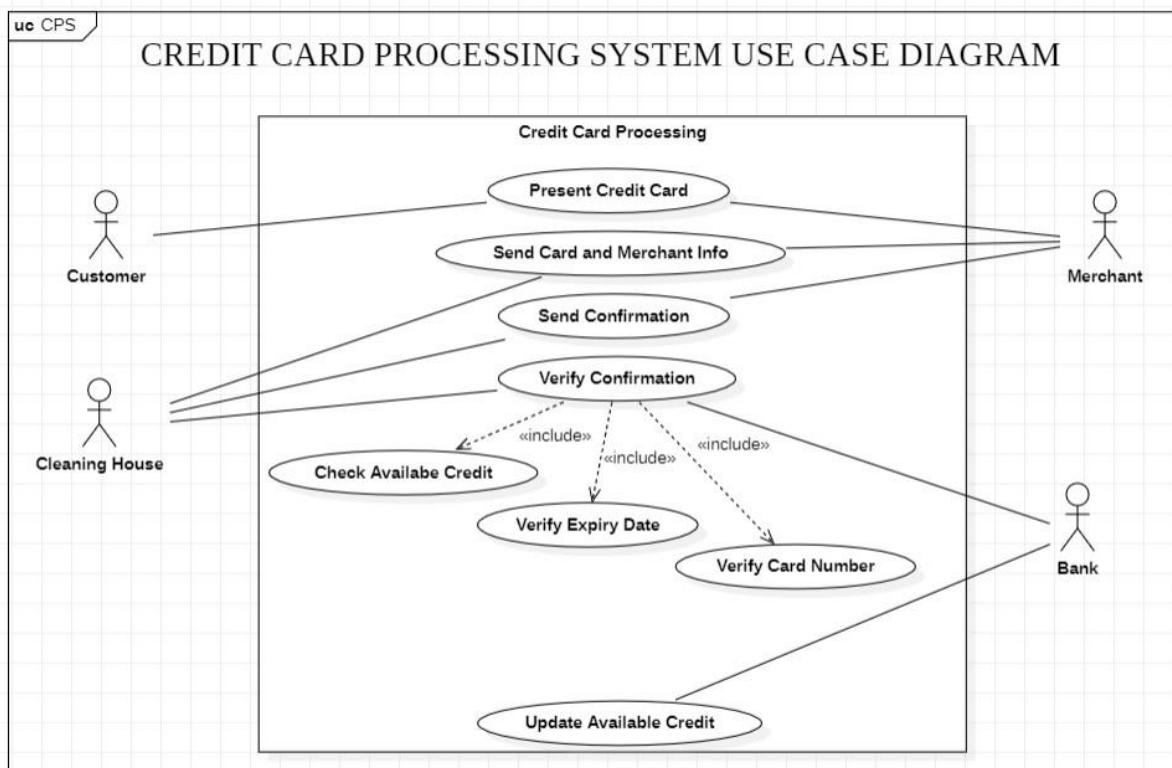
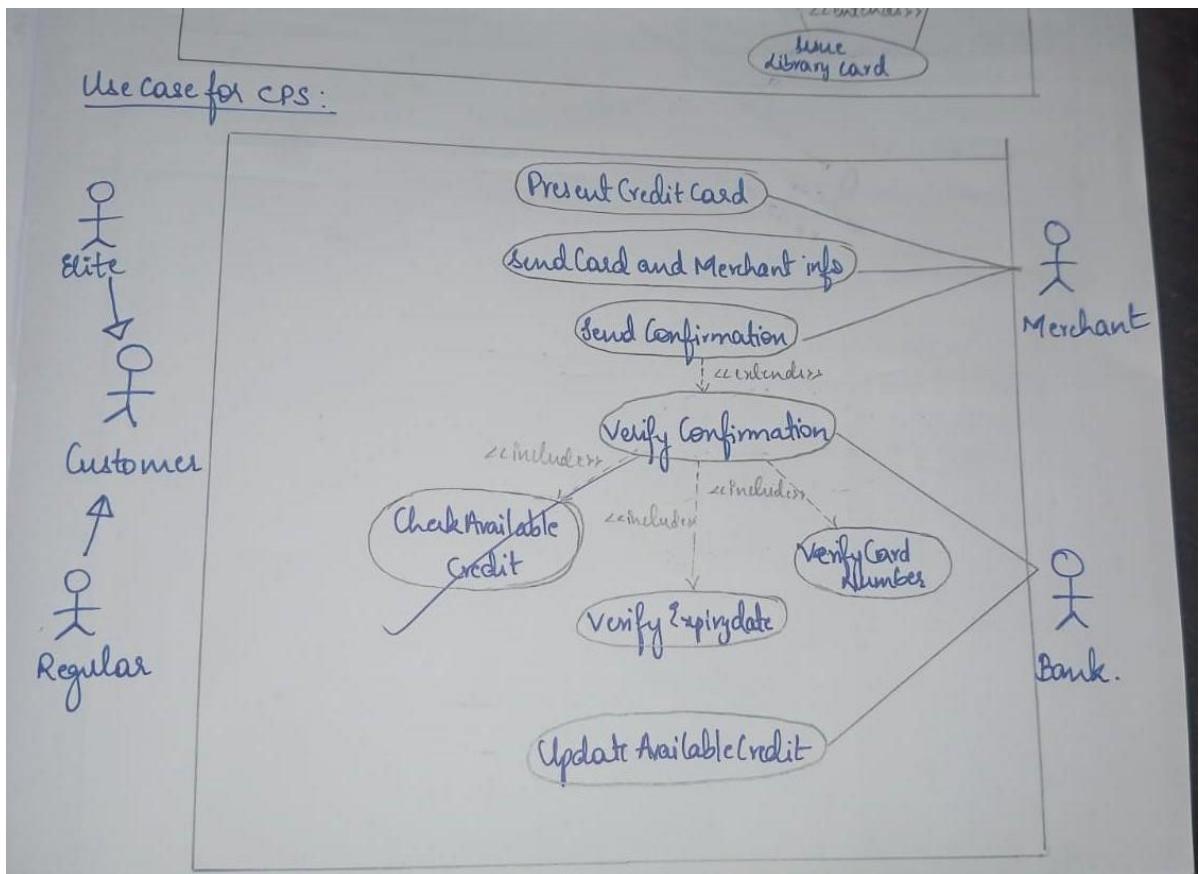


## ADVANCED STATE MODEL:

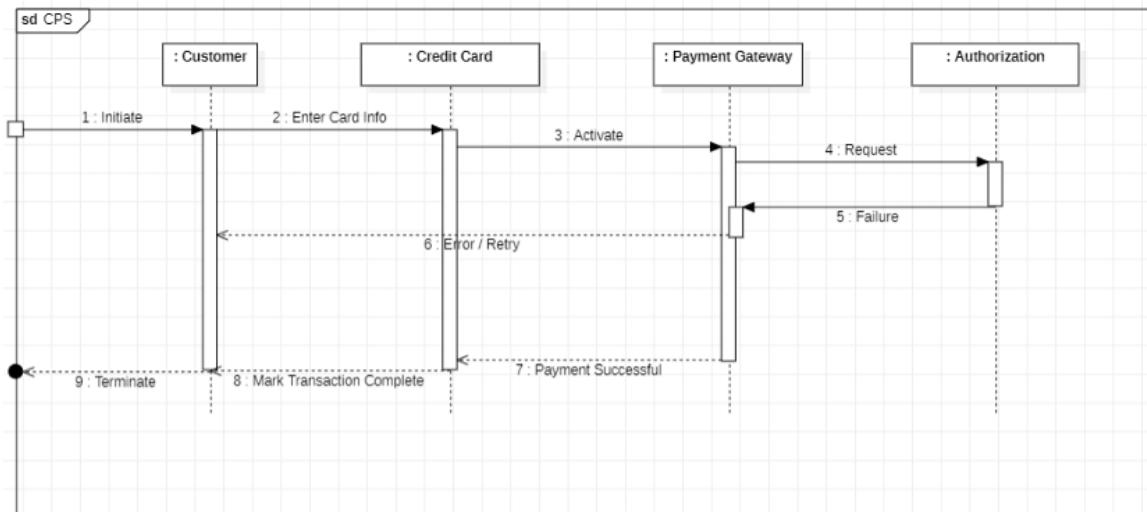
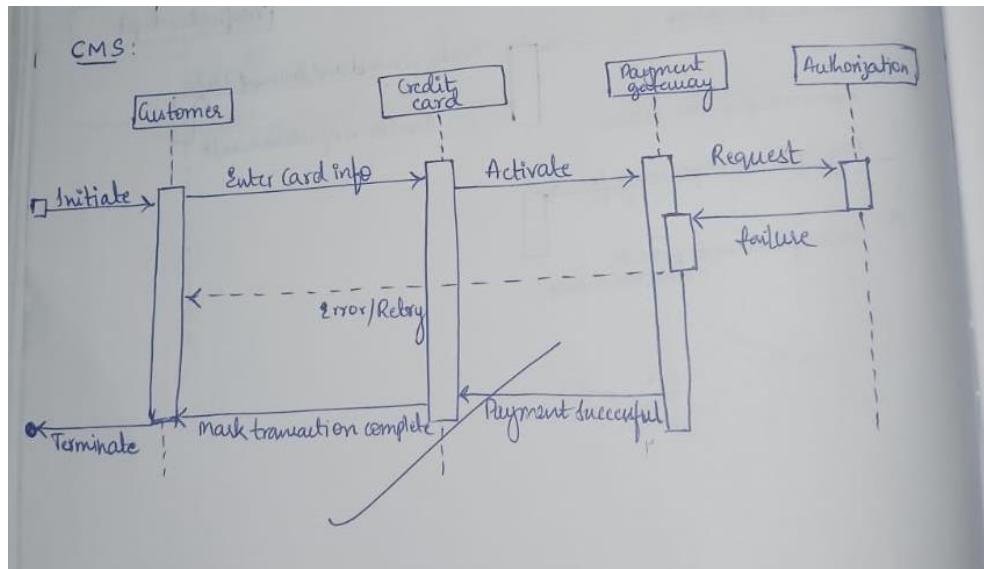




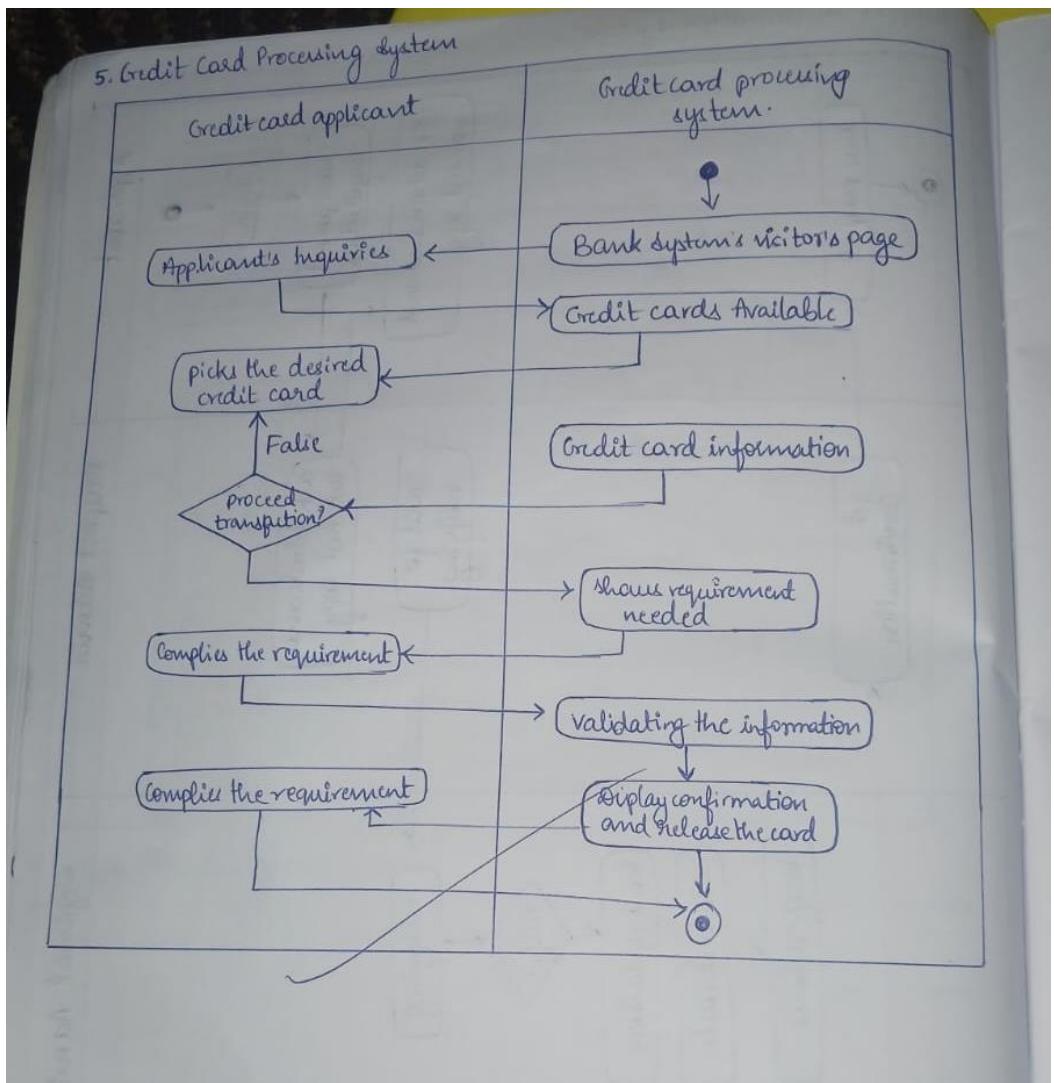
## USE CASE:



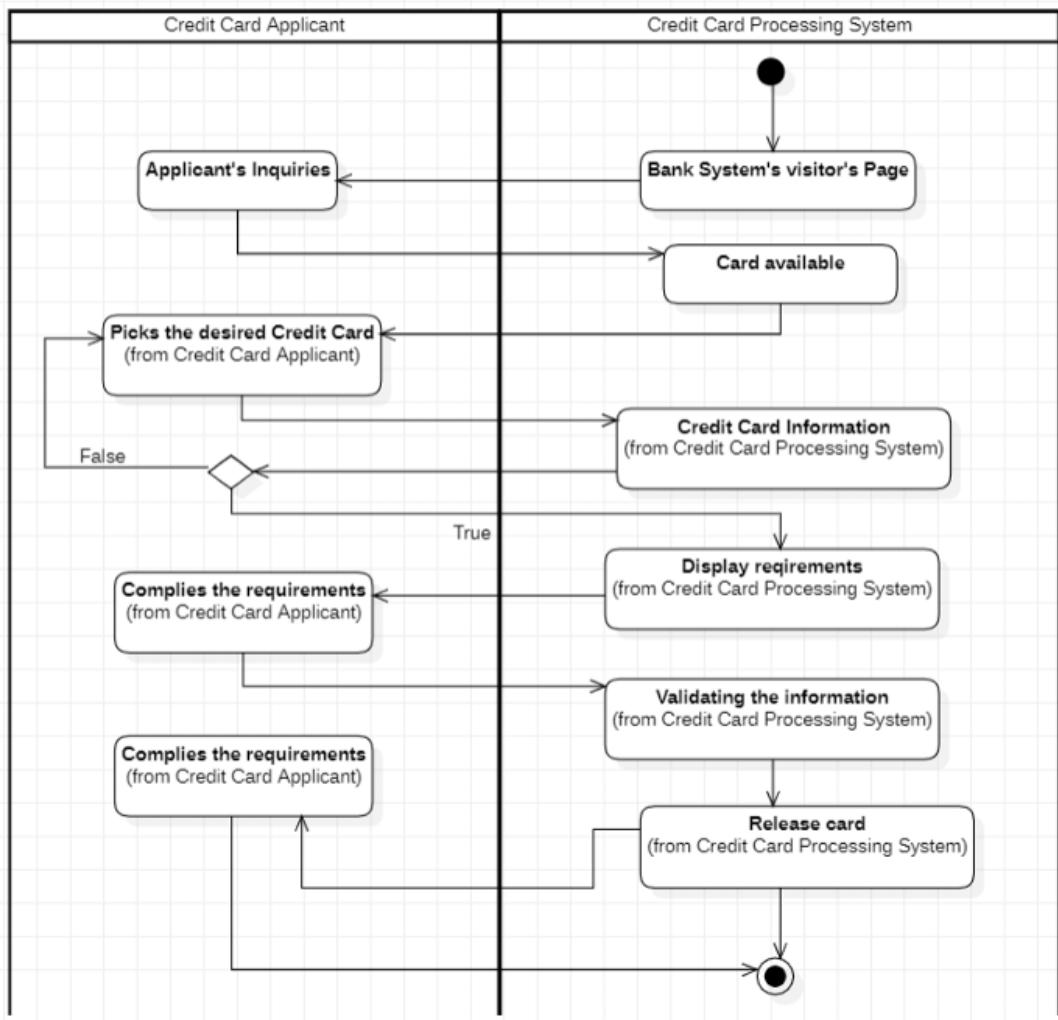
## SEQUENCE DIAGRAM:



## ACTIVITY DIAGRAM:



## CREDIT CARD PROCESSING SYSTEM - ACTIVITY DIAGRAM



# PASSPORT MANAGEMENT SYSTEM

## SOFTWARE REQUIREMENTS SPECIFICATION

Passport Automation System

- 1. Introduction**
  - 1.1 Purpose of this document:**

This document outlines the requirements for a passport automation system, designed to simplify the passport application, verification and issuance processes.
  - 1.2 Scope of this document:**

The Passport Automation System allows citizens to apply for passports online, submit necessary documents, track application status, and schedule appointments. It integrates with government databases for verification of applicant details and provides authorities with tools to process applications, conduct background checks and issue passports.
  - 1.3 Overview:**

The system ensures a seamless flow of passport applications, verification of personal details, and the processing of payments. This document defines the system's core functions, performance criteria and interfaces.
- 2. General Description:**

The PAS allows users to apply for new passports, renewals and track the application status. It also enables government officials to verify applicant information, schedule interviews and approve or reject applications.
- 3. Functional Requirements:**
  - Online Application Submission:** Users can submit passport application online by filling out required forms and uploading necessary documents.
  - Document verification:** The system verifies uploaded documents like certificates, ID proof, etc. by cross-referencing government databases.

#### 4. Interface Requirements:

- User interface: a well developed web-based application for handling the verifications and other processes.
- Database interface for handling all the data.

#### 5. Performance Requirements:

- The system must process and verify documents within 3-5 sec.
- It should handle up to 10000 concurrent user requests, especially during peak application periods.
- The system should provide real-time updates to applicants about the progress of their applications.

#### 6. Design constraints:

- Compatibility with the modern web browsers.
- Accessible across multiple platforms (web and mobile).

#### 7. Non-Functional Attributes:

- Security: All the personal data must be encrypted in storage and during transmission.
- Reliability: It should be reliable with minimal downtime for maintenance.

#### 8. Schedule and Budget:

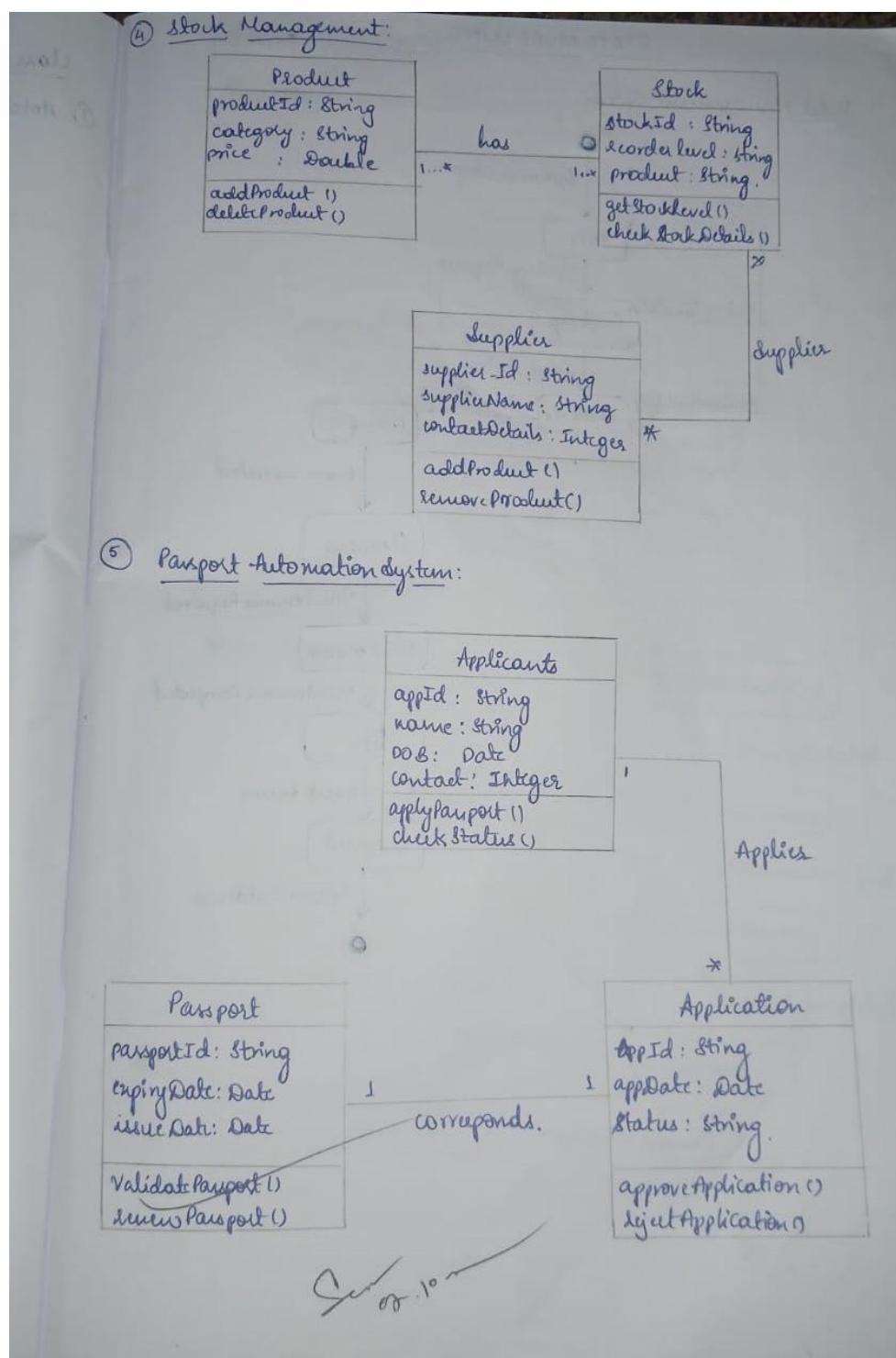
It would take around 6 months for the completion of project.

#### Budget:

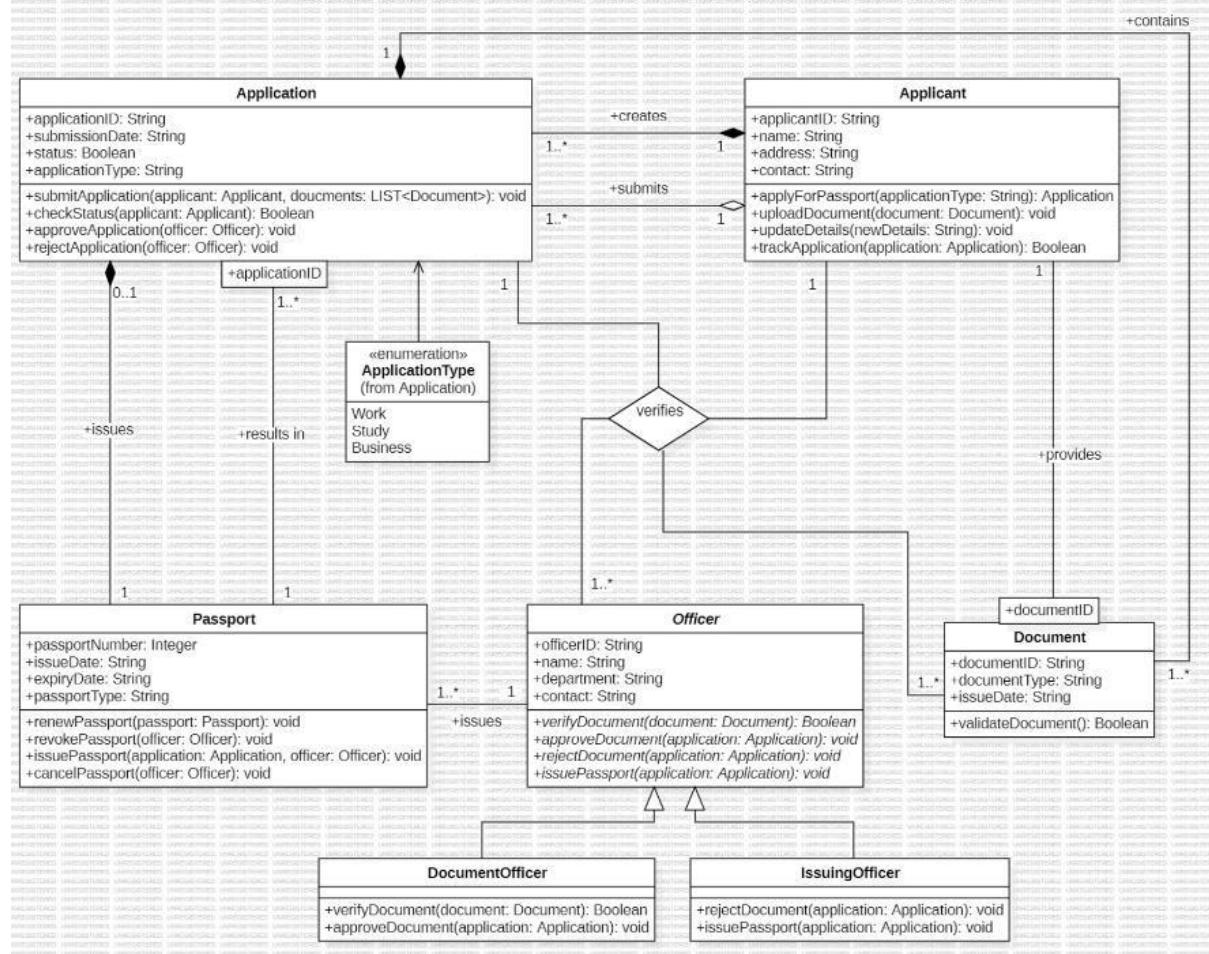
The total estimated Budget: \$ 55000

- Requirement specification: \$ 8000
- Design phase: \$ 20000
- Development phase: \$ 12000
- Deployment and maintenance: \$ 5000
- Testing: \$ 10000

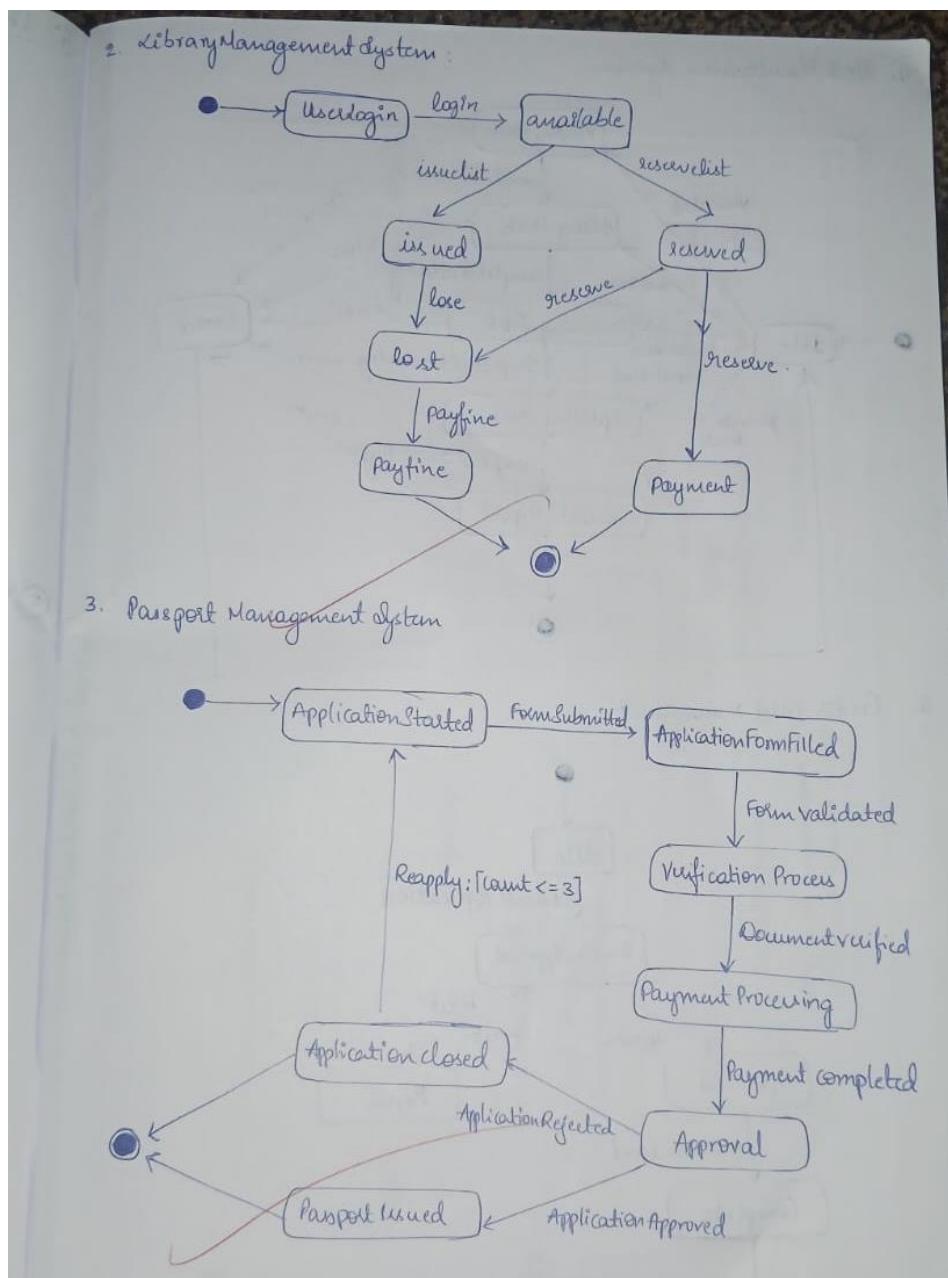
## CLASS DIAGRAM:

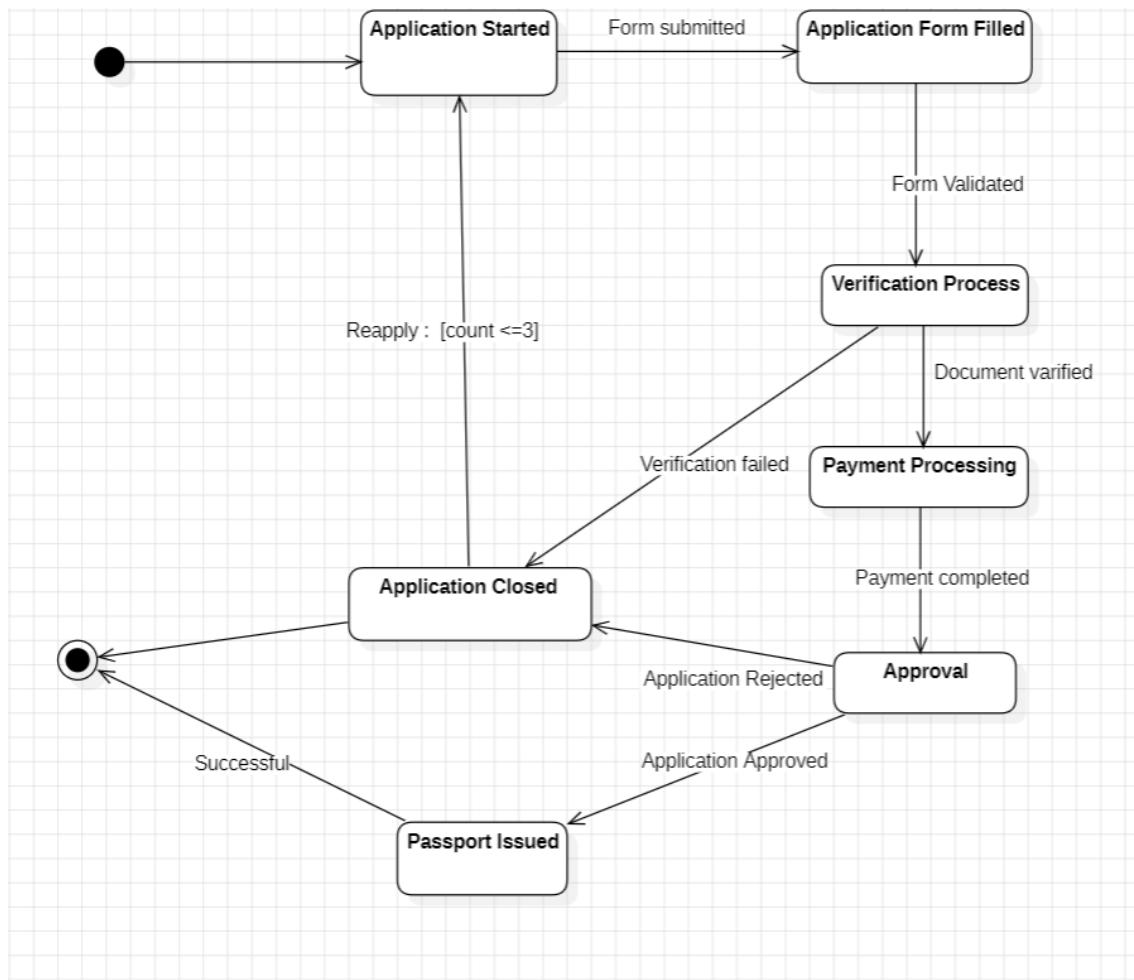


## PASSPORT AUTOMATION SYSTEM - CLASS DIAGRAM

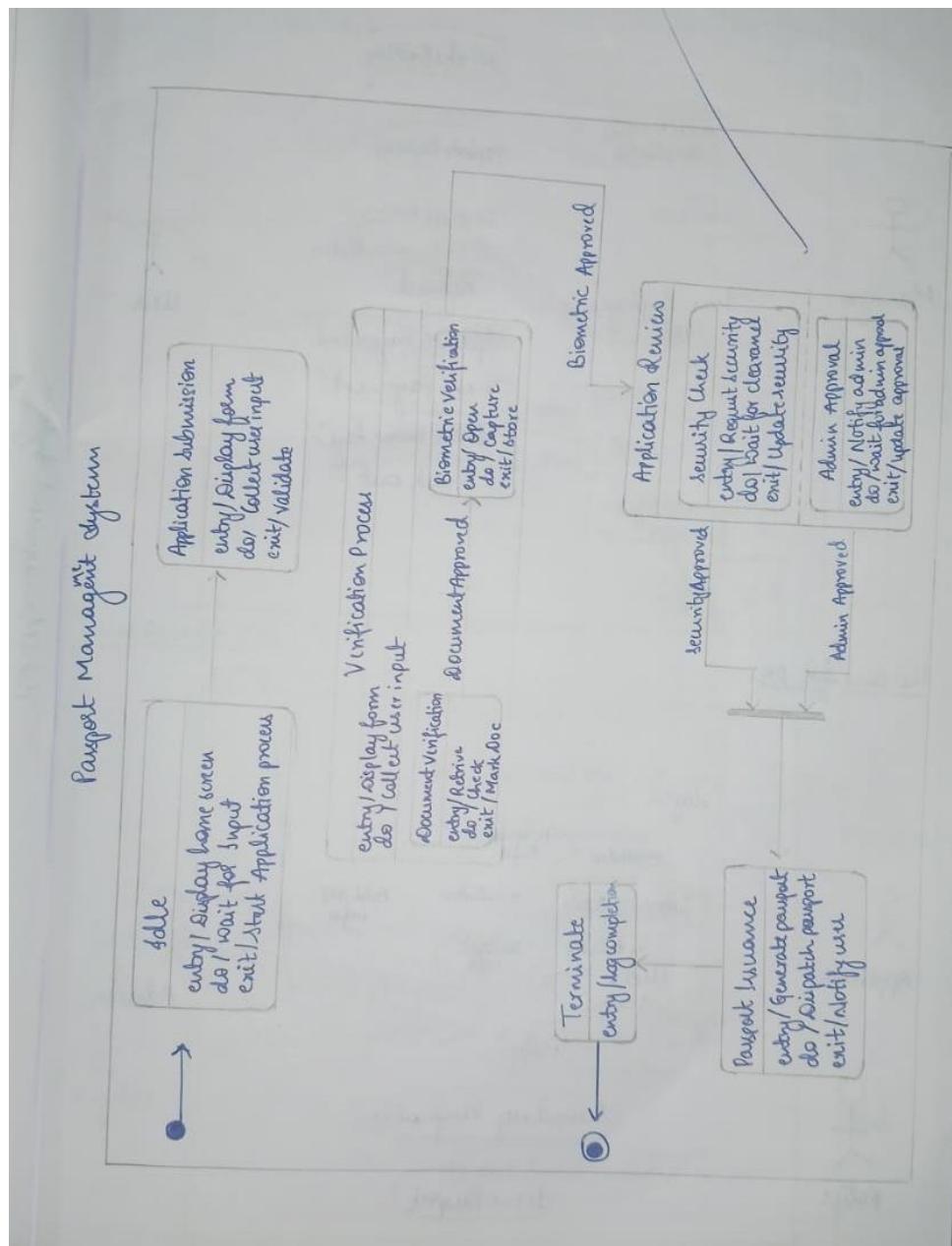


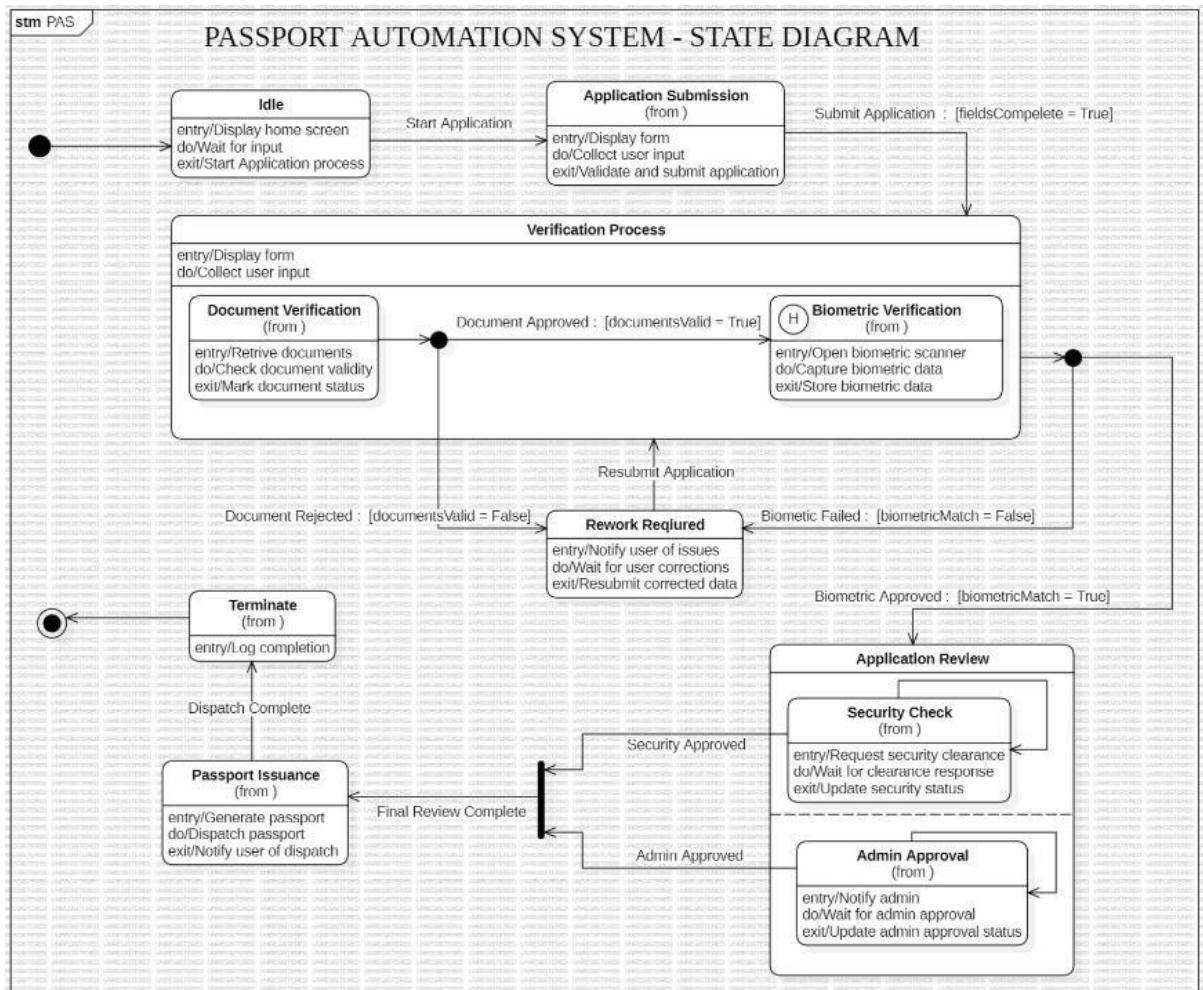
## STATE DIAGRAM:



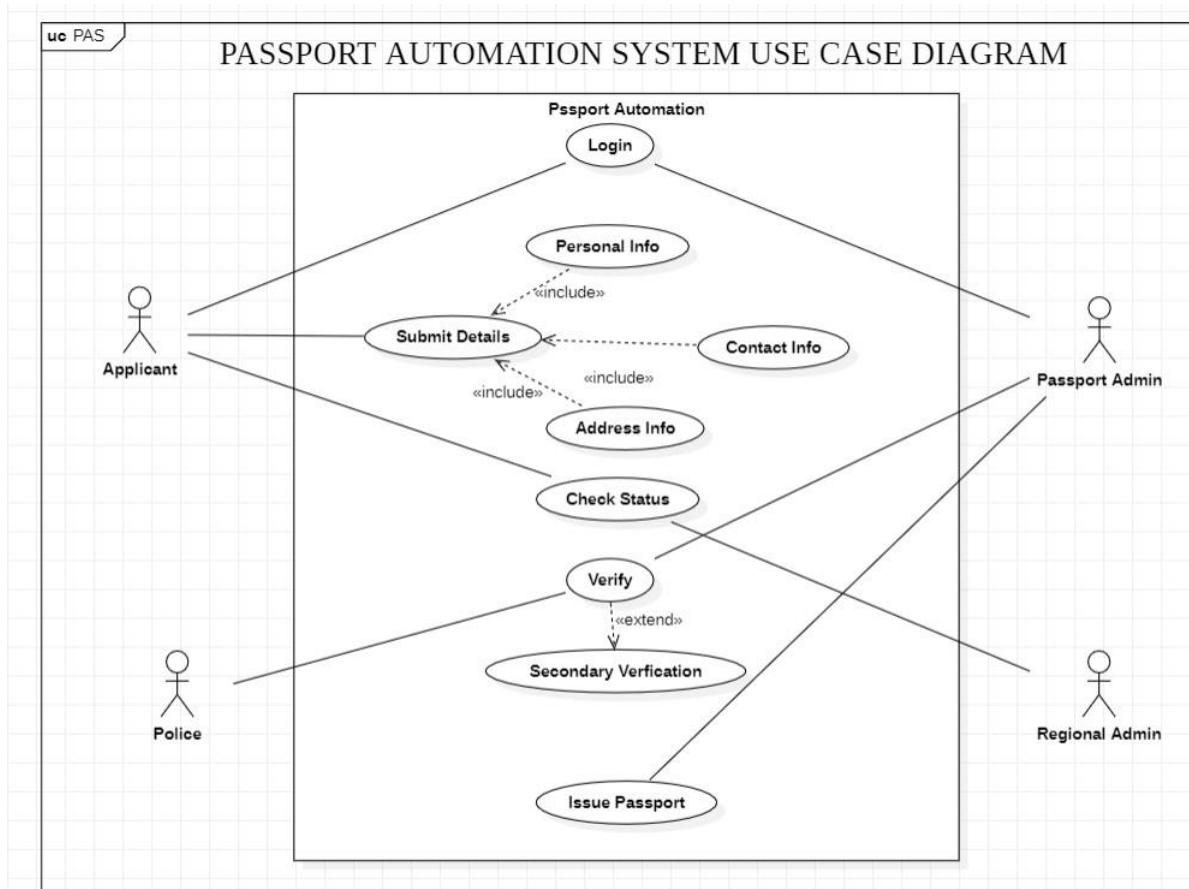
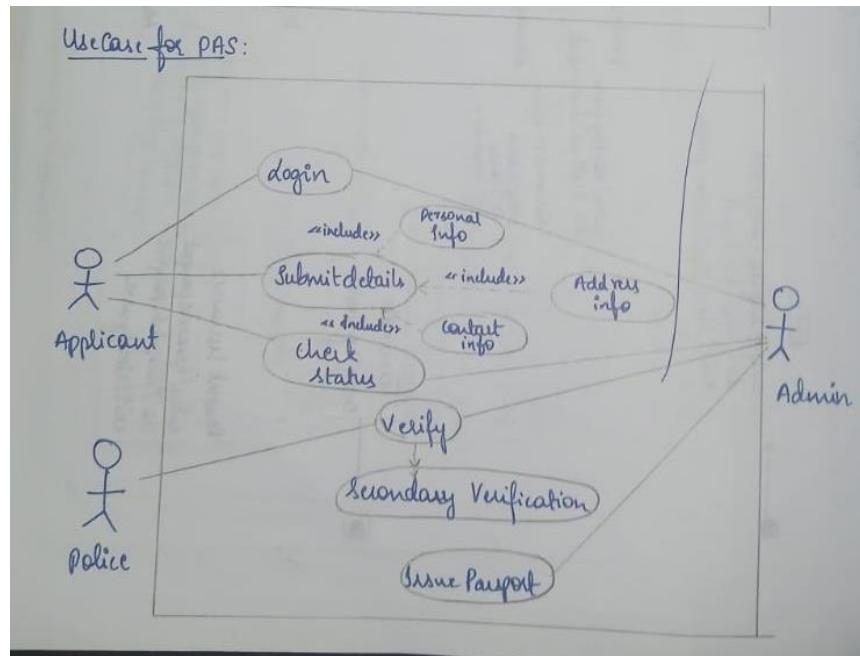


## ADVANCED STATE MODEL

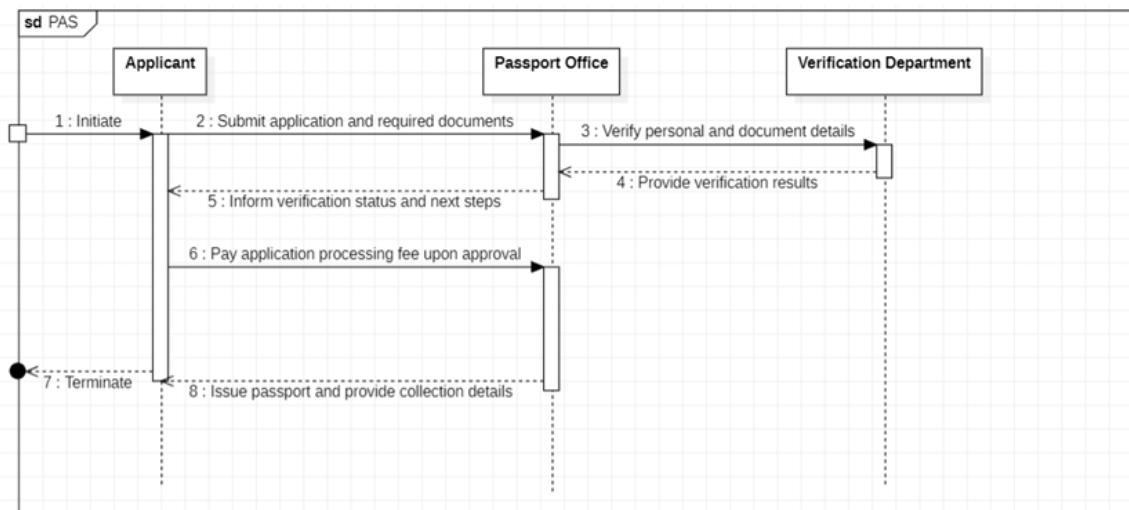
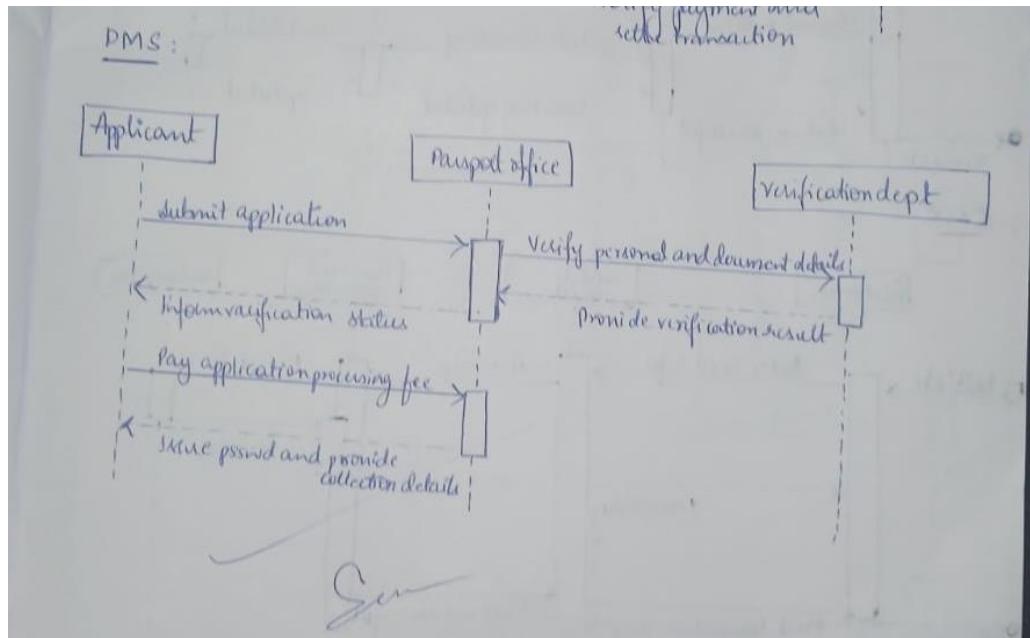




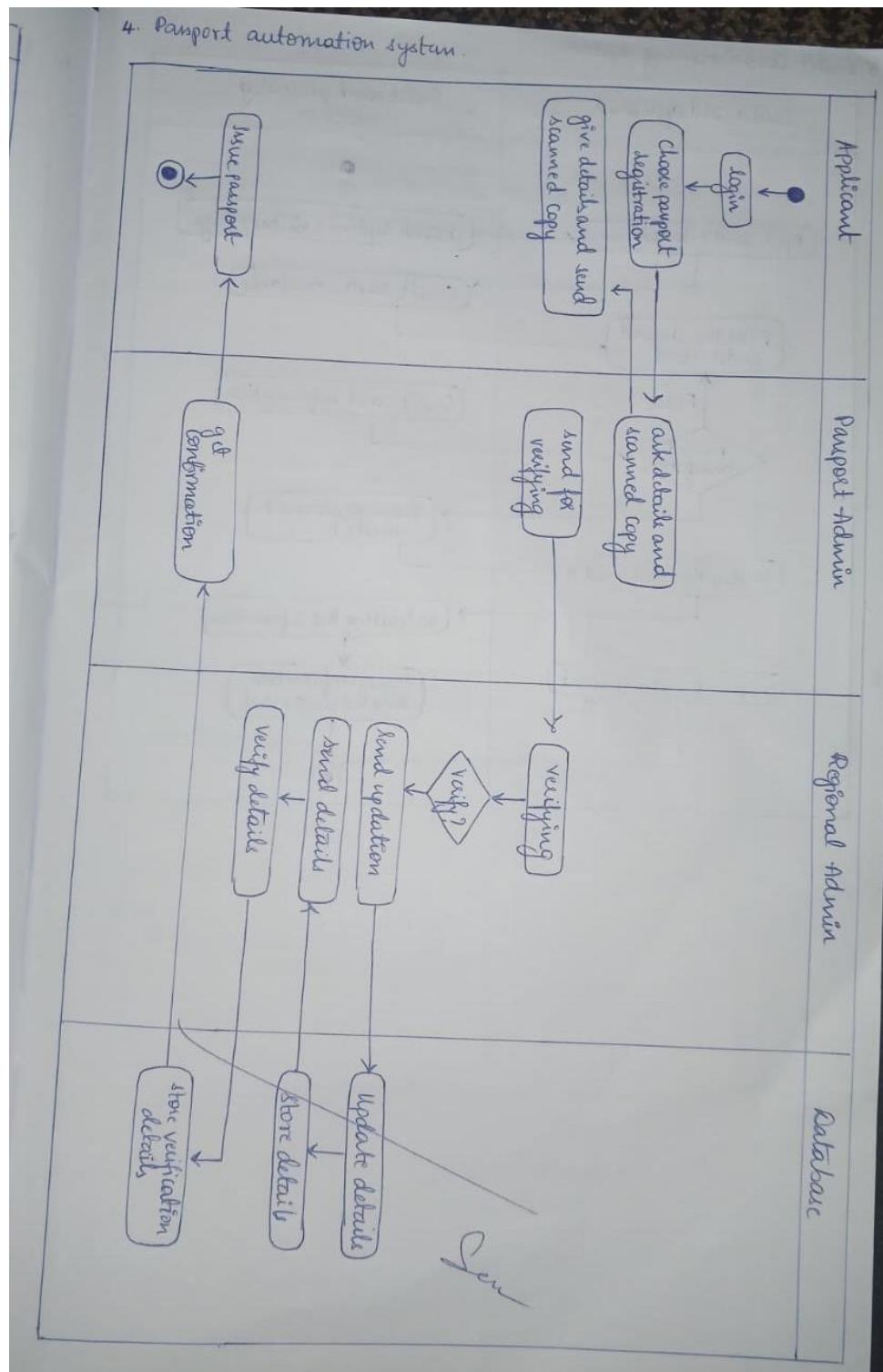
## USE CASE:



## SEQUENCE DIAGRAM:

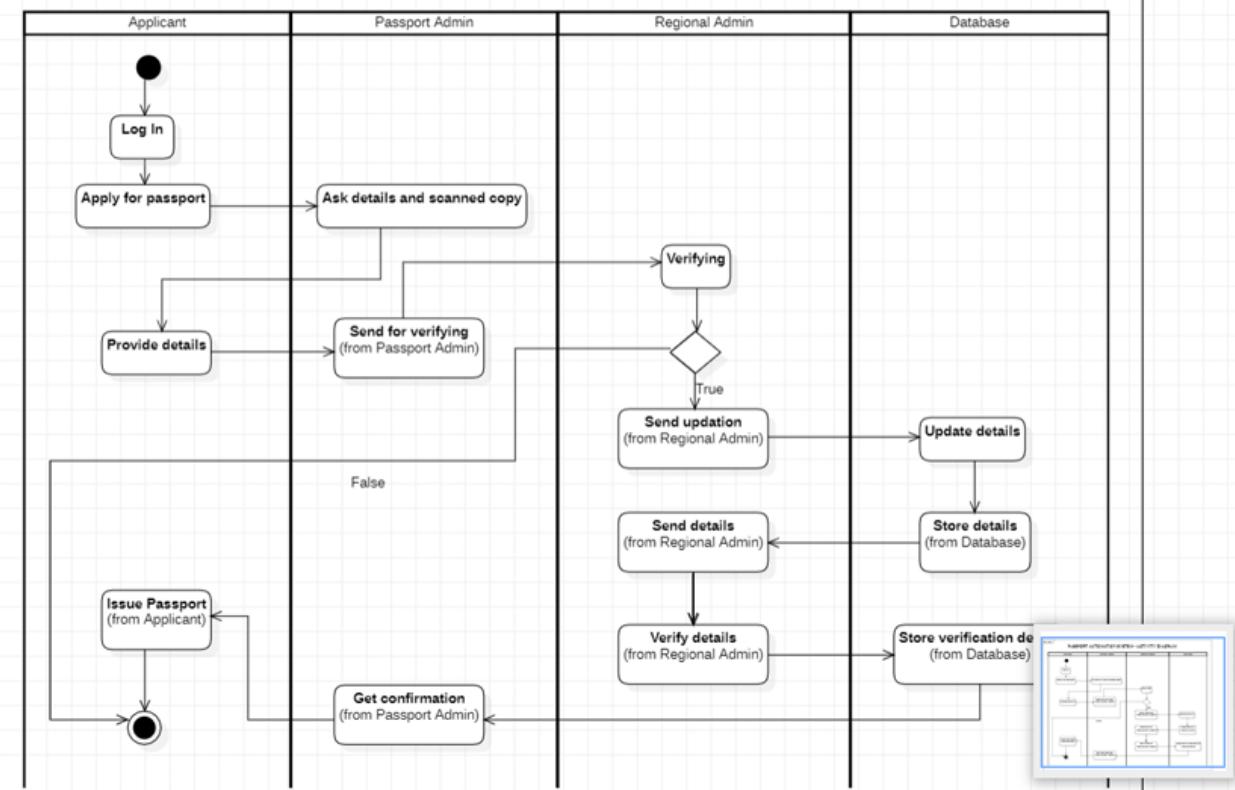


## ACTIVITY DIAGRAM:



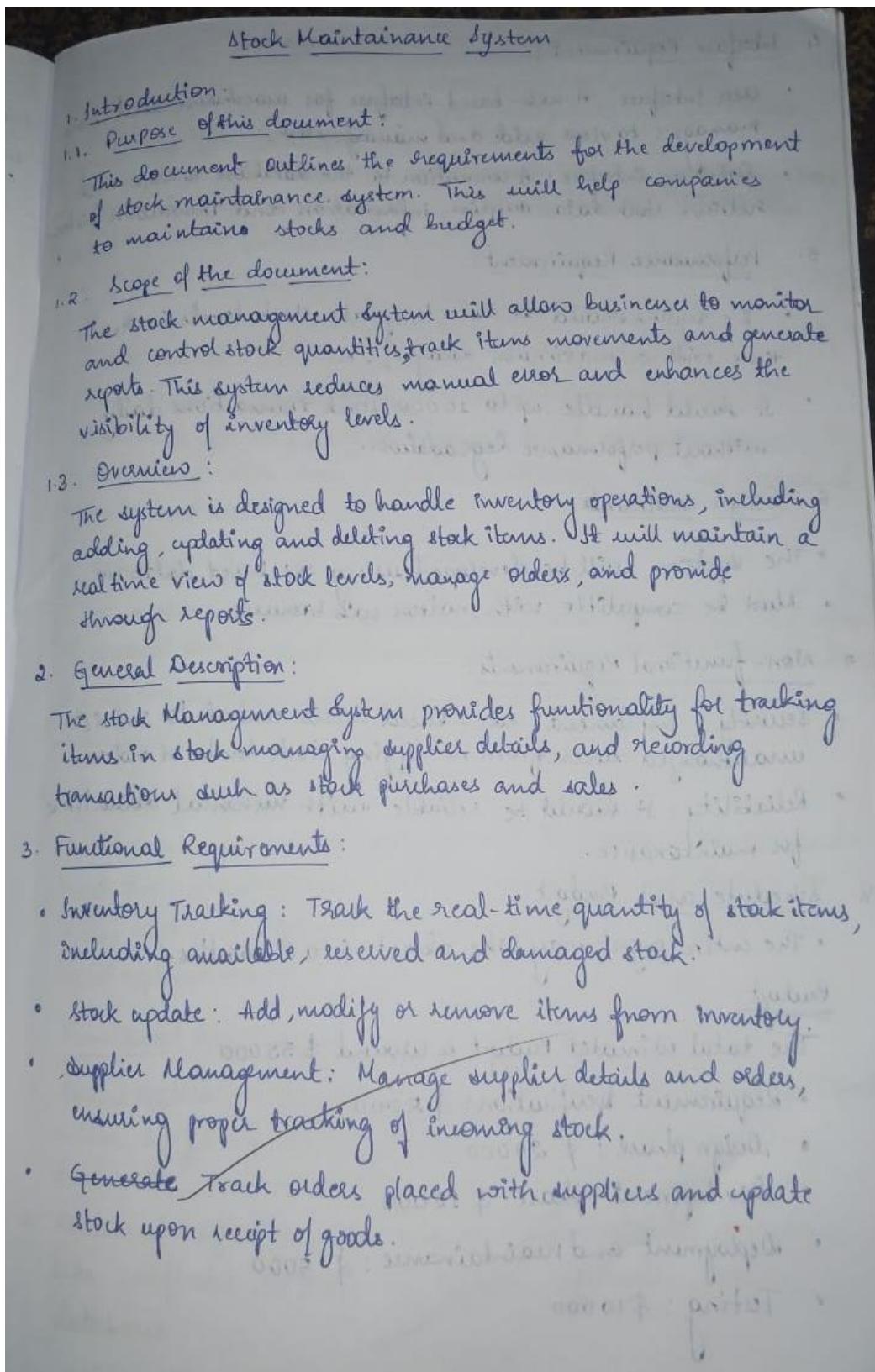
act PAS

## PASSPORT AUTOMATION SYSTEM - ACTIVITY DIAGRAM



# STOCK MANAGEMENT SYSTEM

## SOFTWARE REQUIREMENT SPECIFICATION



#### 4. Interface Requirement:

- User Interface: A web-based interface for warehouse and store managers to view, add and manage stock.
- Database Interface: A connection to the database to store and retrieve stock data, duplicate information and transaction history.

#### 5. Performance Requirement:

- The system should be able to update stock levels in real time with a maximum delay of 1 sec.
- It should handle up to 10000 stock transactions daily without performance degradation.

#### 6. Design Constraints:

- The system will be developed using SQL based database.
- Must be compatible with modern web browser.

#### 7. Non-functional Requirements:

- Security: Implement role-based access control to restrict unauthorized users from modifying stock levels or orders.
- Reliability: It should be reliable with minimal downtime for maintenance.

#### 8. Schedule and Budget:

- The entire project may take about 6 to 8 months.

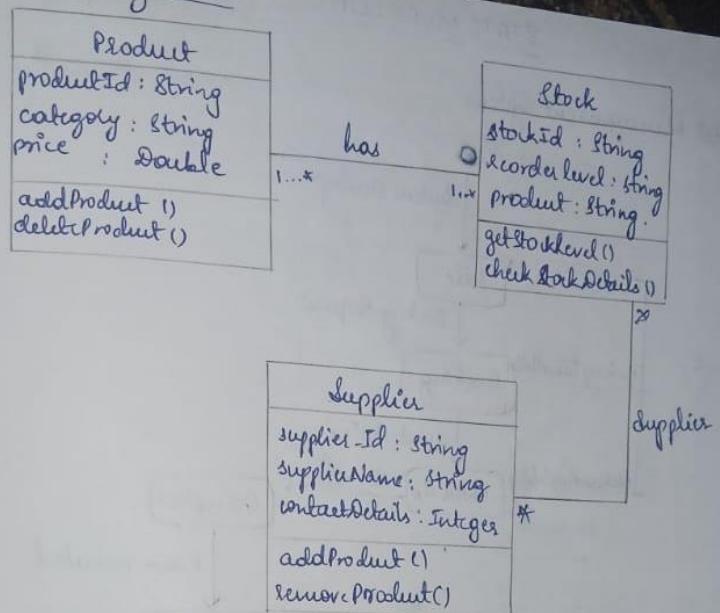
#### Budget

The total estimated budget is around \$ 55000

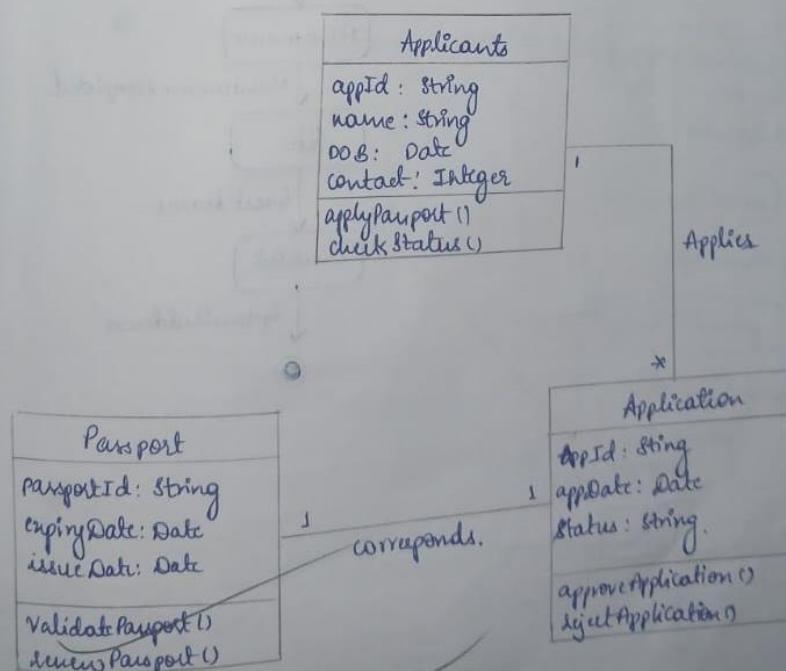
- Requirement Specification: \$ 8000
- Design phase: \$ 20000
- Development phase: \$ 12000
- Deployment and maintenance: \$ 5000
- Testing: \$ 10000

## CLASS DIAGRAM

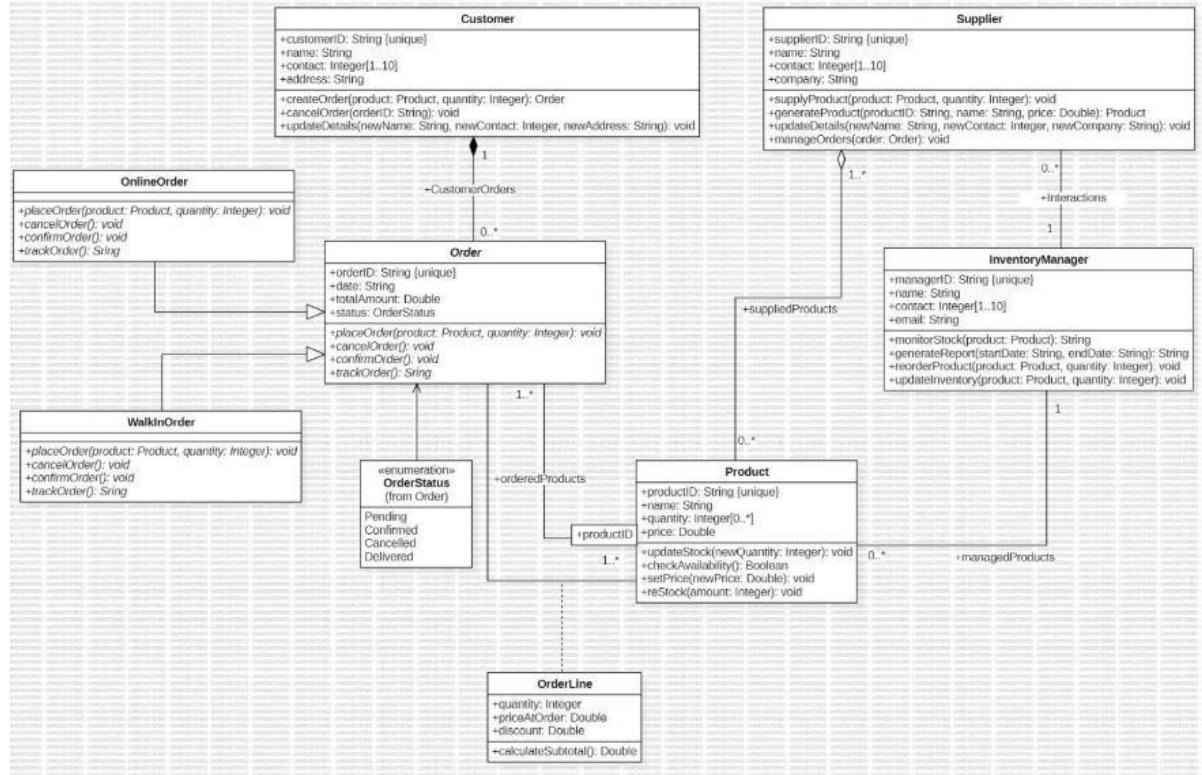
④ Stock Management:



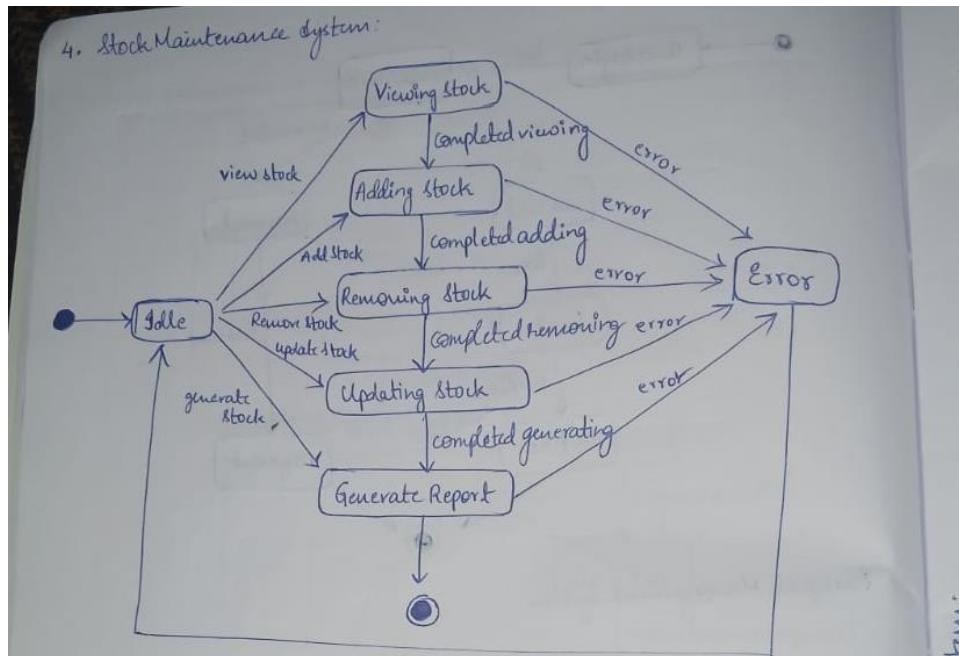
⑤ Passport Automation System:



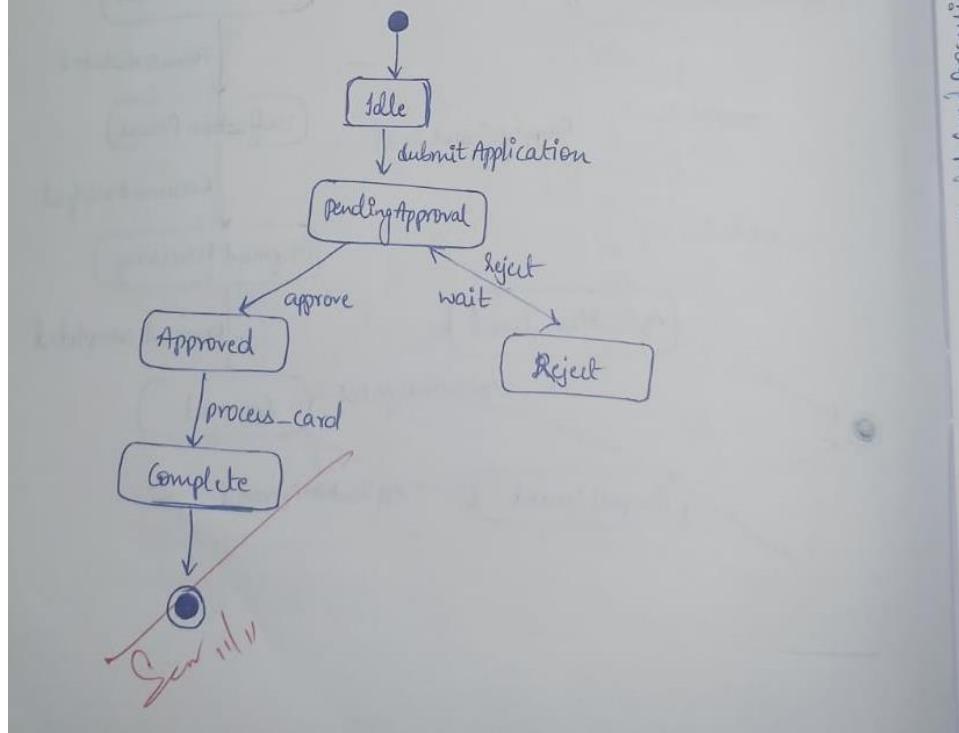
## STOCK MAINTENANCE SYSTEM - CLASS DIAGRAM

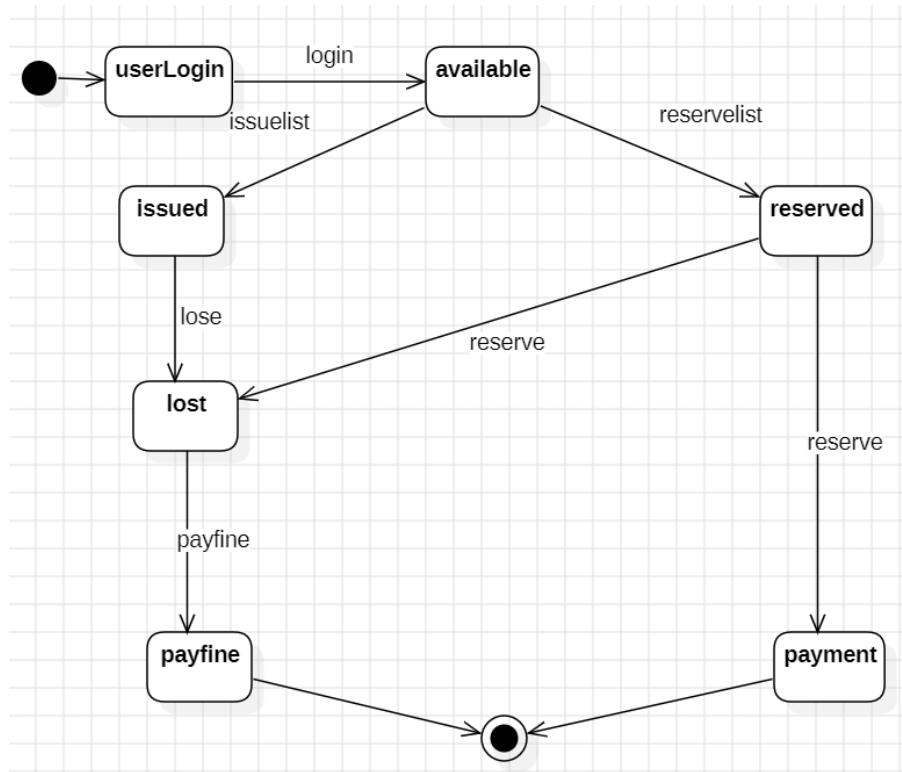


## STATE MODEL

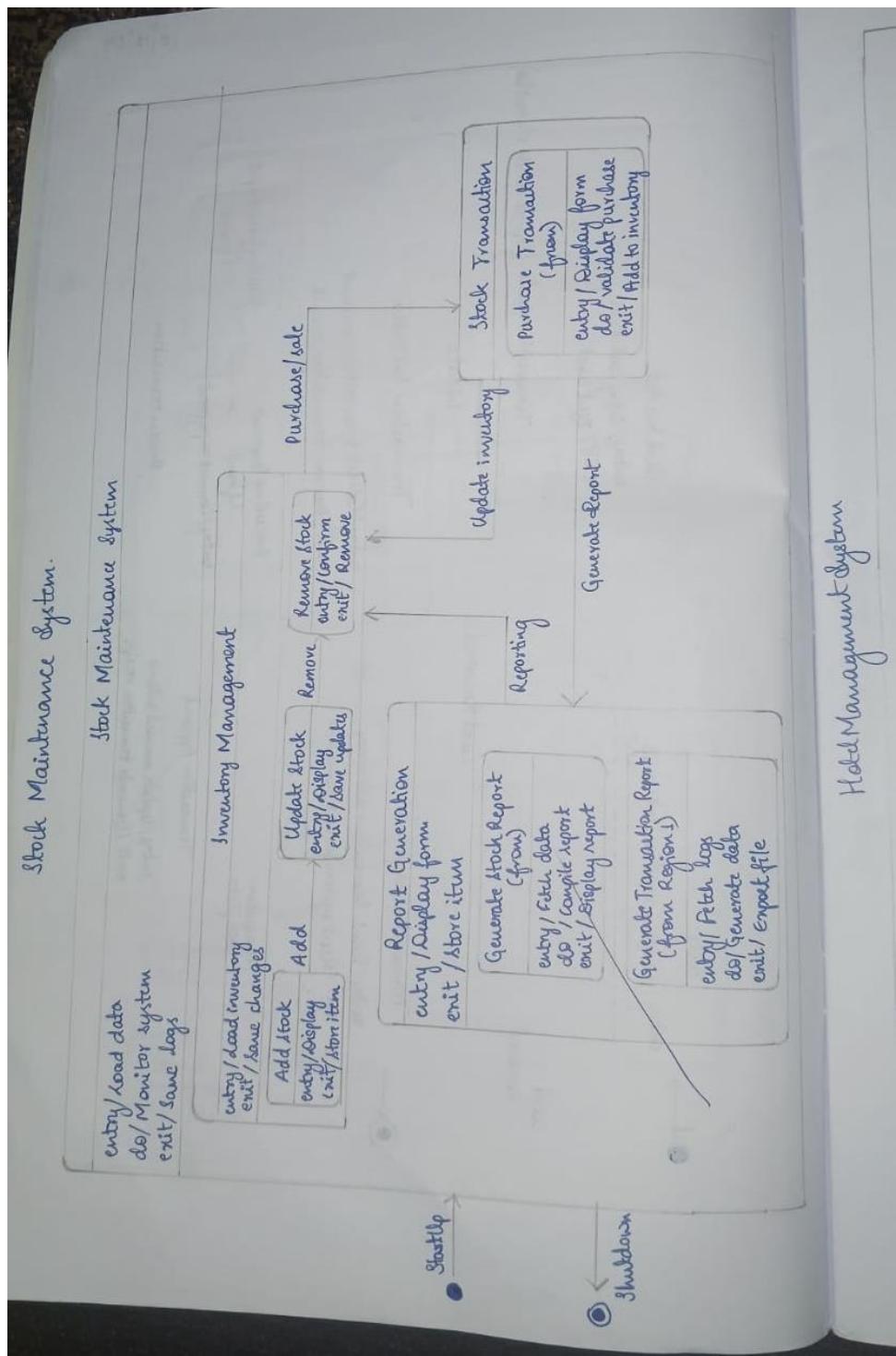


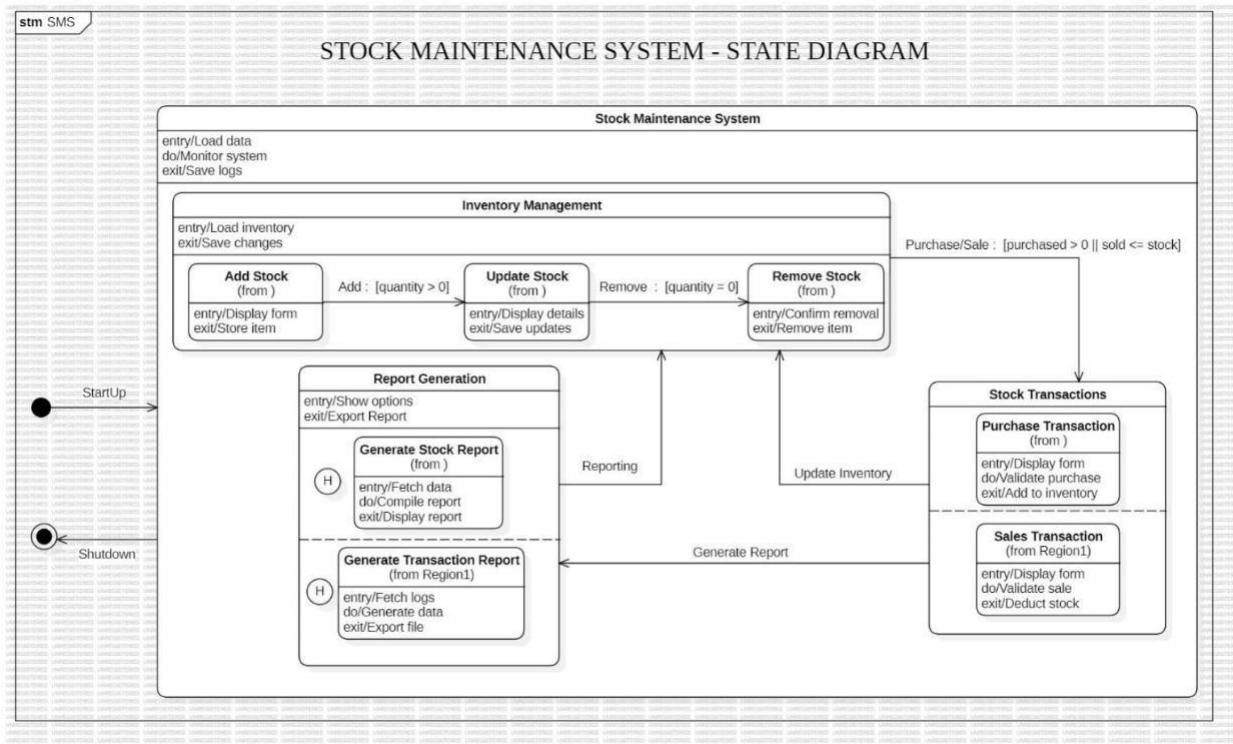
5. Credit Card Management:



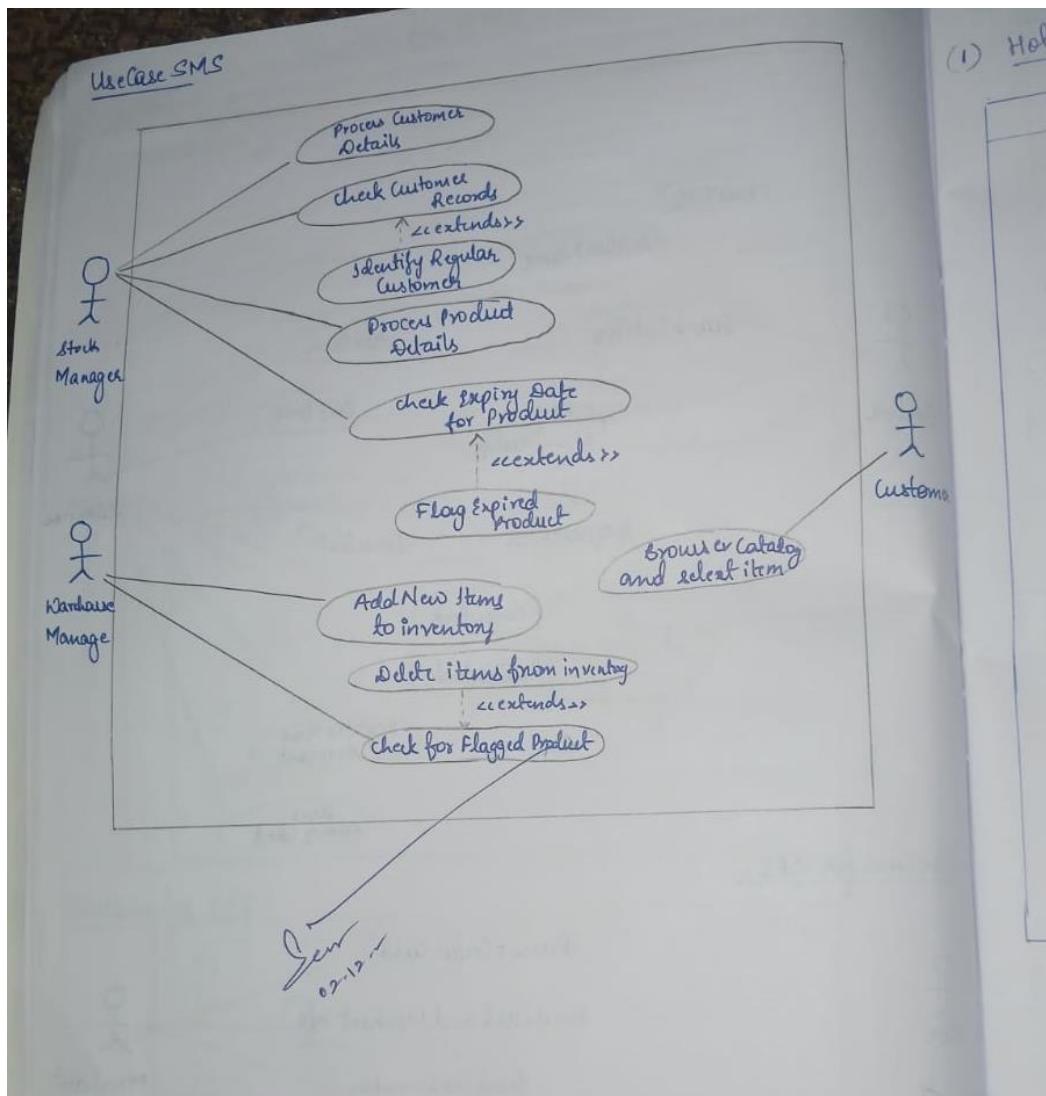


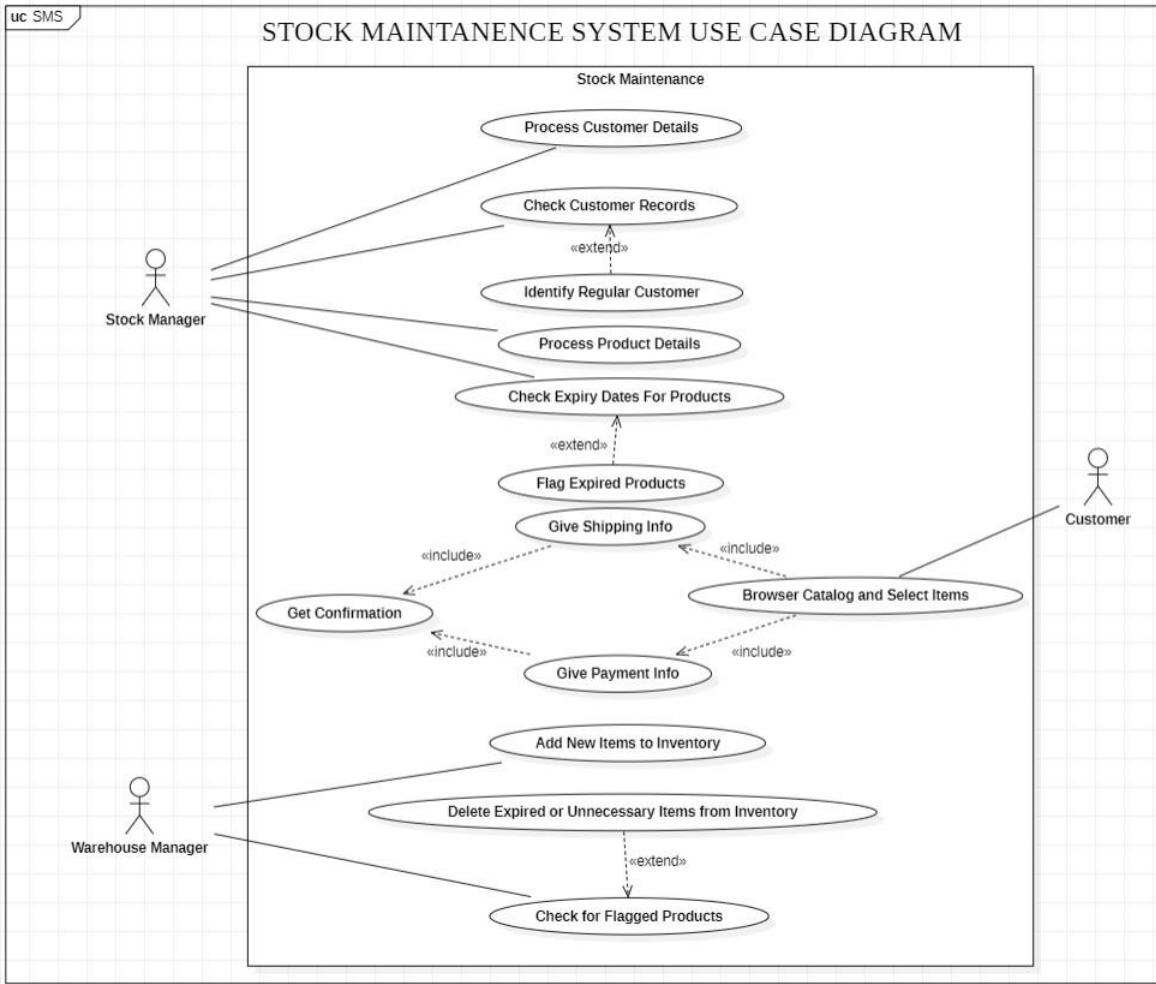
## ADVANCE STATE MODEL



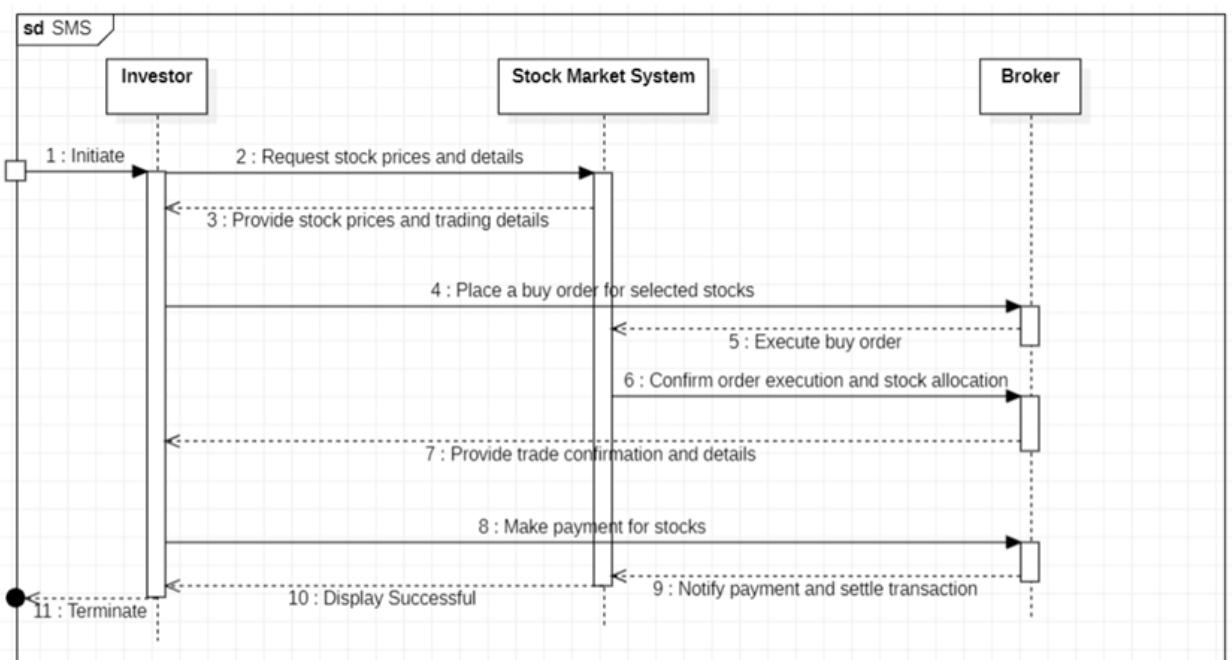
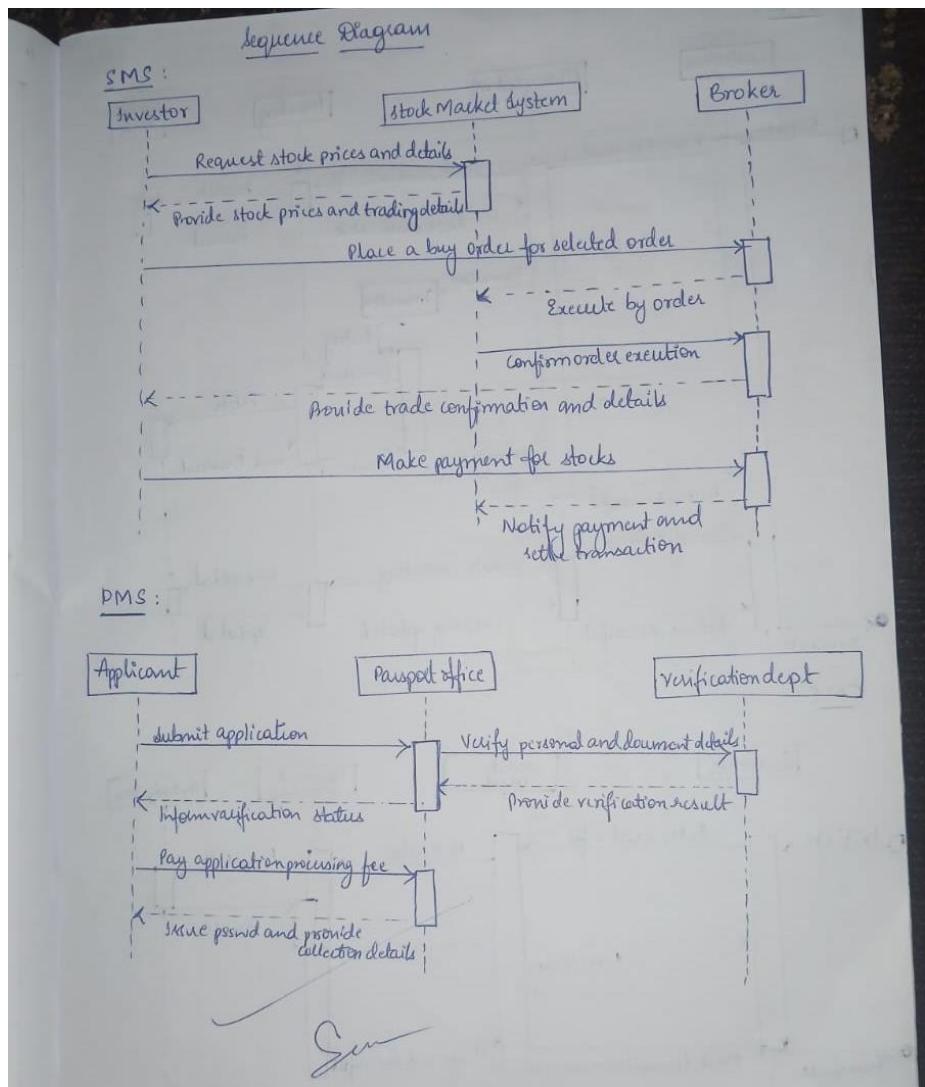


## USE CASE:

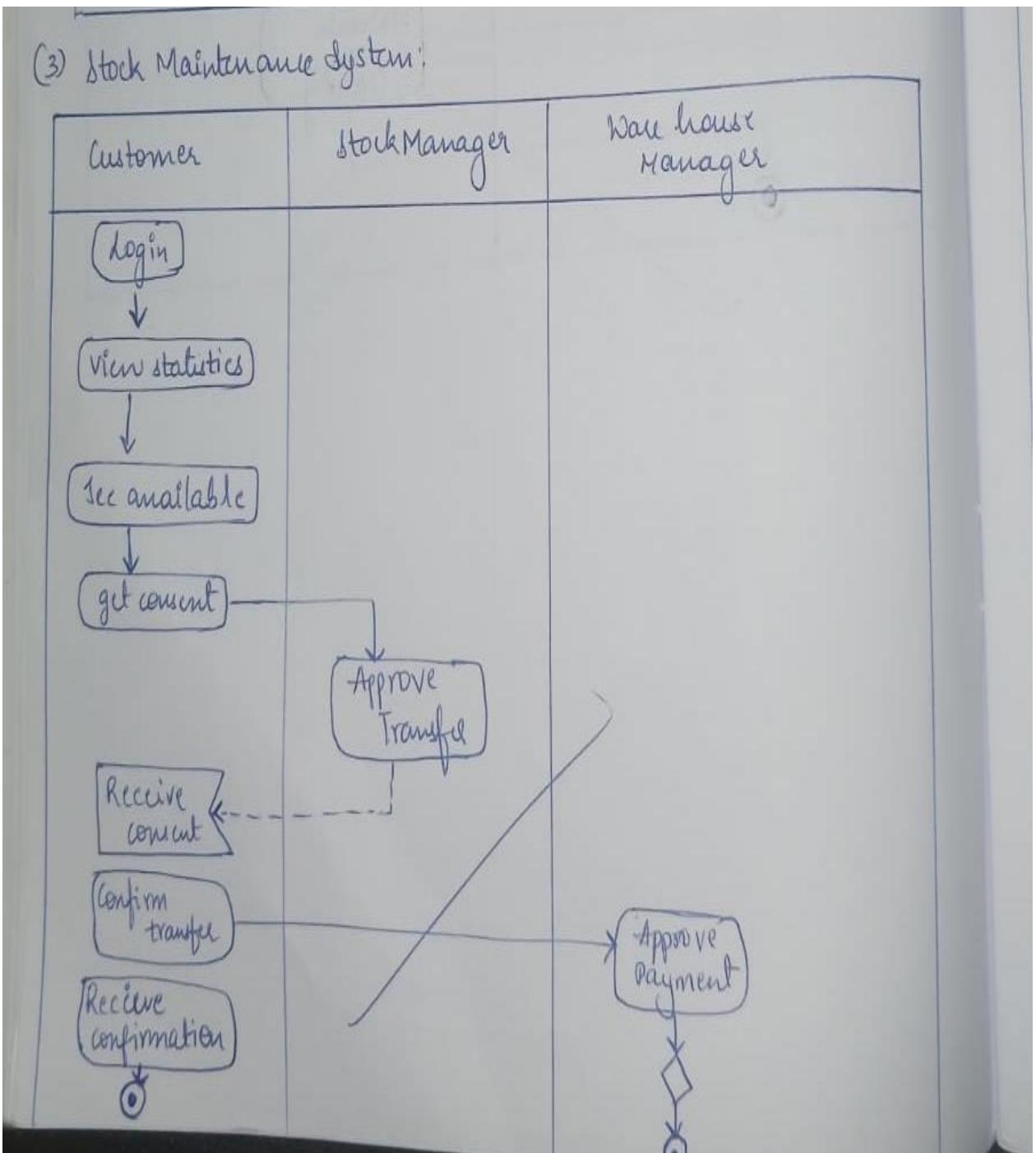




## SEQUENCE DIAGRAM:

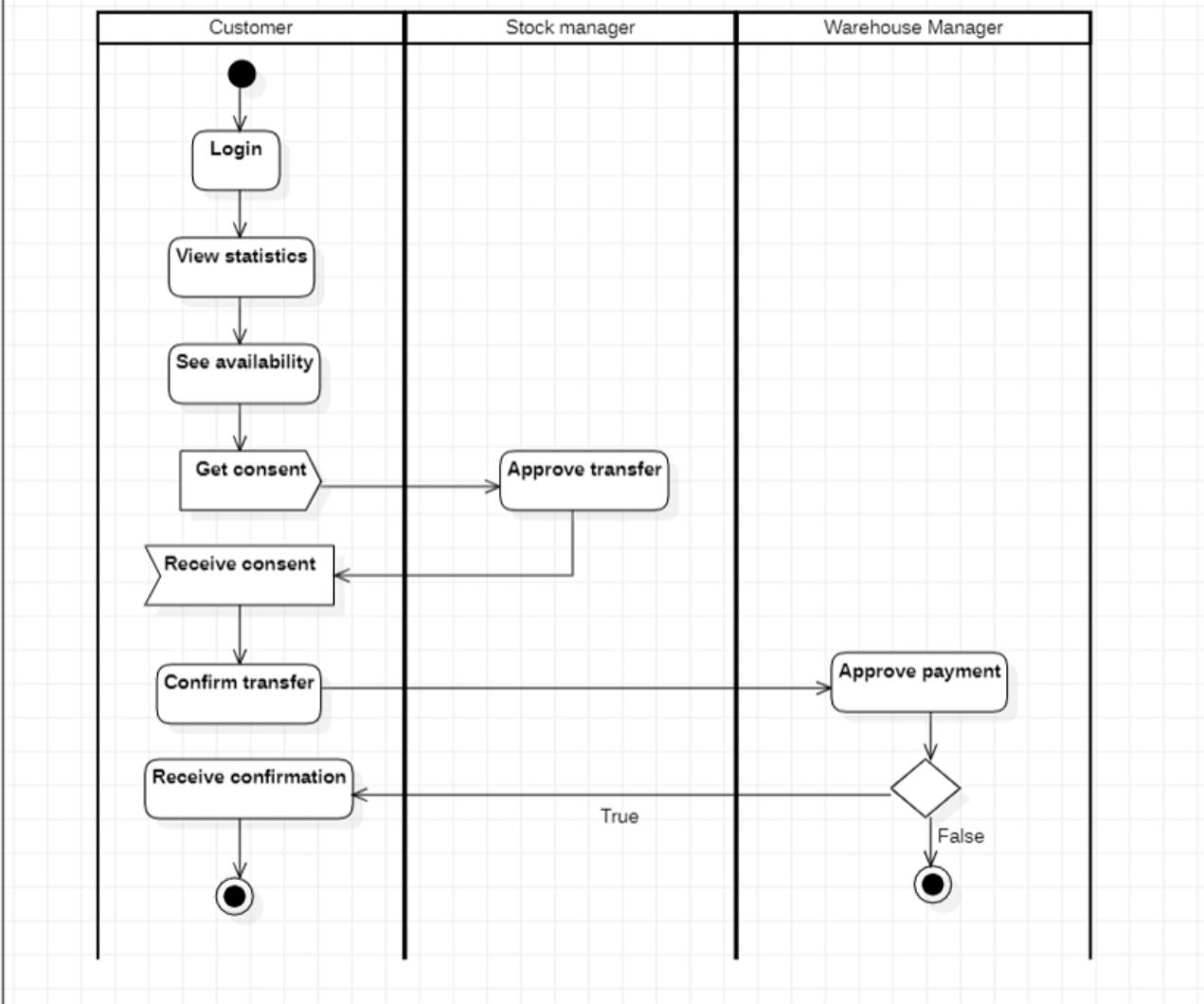


## ACTIVITY DIAGRAM



act SMS

## STOCK MAINTENANCE SYSTEM - ACTIVITY DIAGRAM



# LIBRARY MANAGEMENT SYSTEM

## SOFTWARE REQUIREMENT SPECIFICATION

LAB - 1  
Software Requirement Specification (SRS)  
23/9/24

1. Introduction

1.1 Purpose of this document:  
This document outlines the requirements for the development of hotel Management system. The purpose is to define the system's functionality, behaviour, and performance to streamline hotel operations like room booking, billing and reporting.

1.2 Scope: It aims to automate the tasks related to hotel operations. It will serve hotel staff to manage bookings, room availability, customer details and financial transactions efficiently.

1.3 Overview: The HMS is an application designed for hotel operators and customers. It includes modules for room reservations, billing and customer information management. The system ensures security and flexibility while handling different types of rooms, amenities and pricing.

2. Description:  
The hotel management system allows hotel staff to manage room reservations, track check-in and check-out processes and manage customer data. Features include room availability tracking, automatic record management, billing and report generation.

3. Functional Requirements:

- Room Booking: Allow customers to book rooms based on availability, room type and price.
- Check-in and checkout: track customer check-ins and check-outs updating room status in real-time.
- Room Availability: show real-time room status and pricing.

#### 4. Interface Requirements:

- The system will have a web-based interface available to customers and hotel staff.
- Integration with the database to store and retrieve customer, room and billing information.

#### 5. Performance Requirements:

- The system should support concurrent user access without performance degradation.
- The system should handle up to 500 concurrent transactions during peak times.
- Memory usage should remain below 70% during normal operation.

#### 6. Design Constraints:

- The system will be developed using Java and a MySQL database.
- The system should support both desktop and mobile devices.

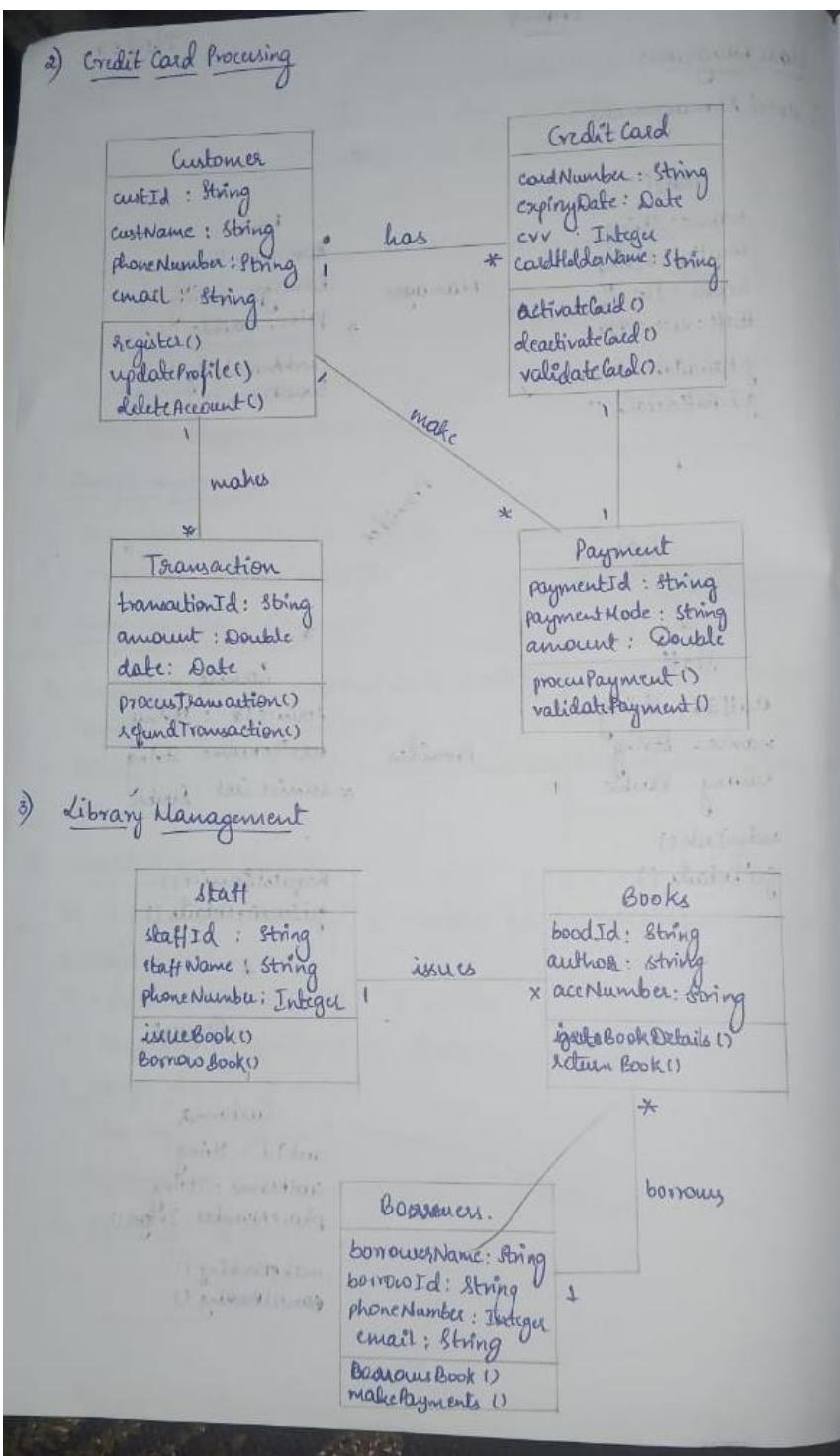
#### 7. Non-functional requirements:

- System downtime should not exceed 1.1. annually.
- The system should be able to handle an increase in customer and room capacity.
- Ensure consistent data when multiple users access or modify records.

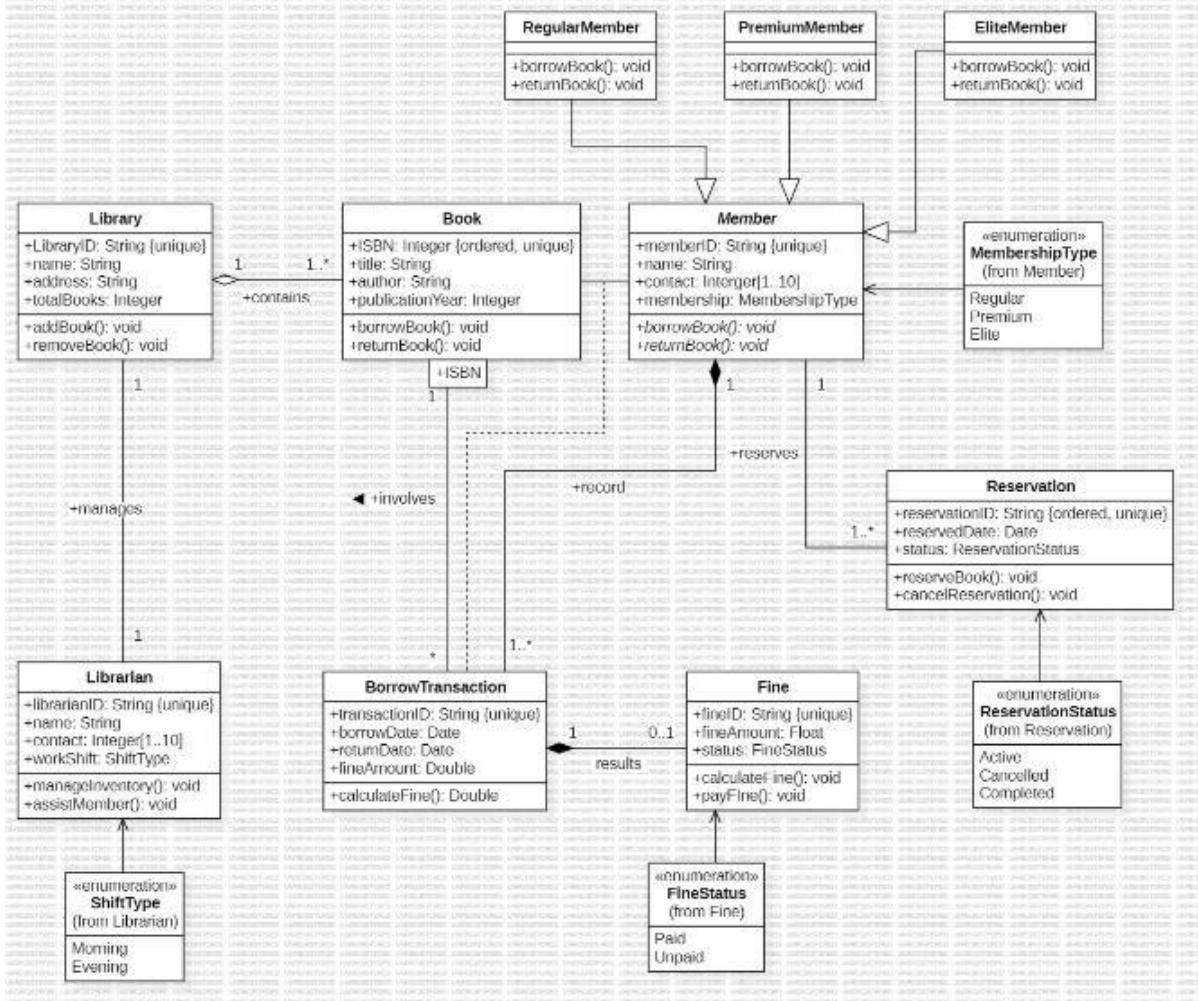
#### 8. Budget and schedule:

- The entire project can take around 6 months for completion starting with requirement specification, design, develop, deploy and testing.
- Budget: Around \$150000 could be estimated
  - 1) Requirement specification : \$ 15000
  - 2) Design phase : \$ 25000
  - 3) Development : \$ 80000
  - 4) Deploy and testing : \$ 27500
  - 5) Maintenance : \$ 8500.

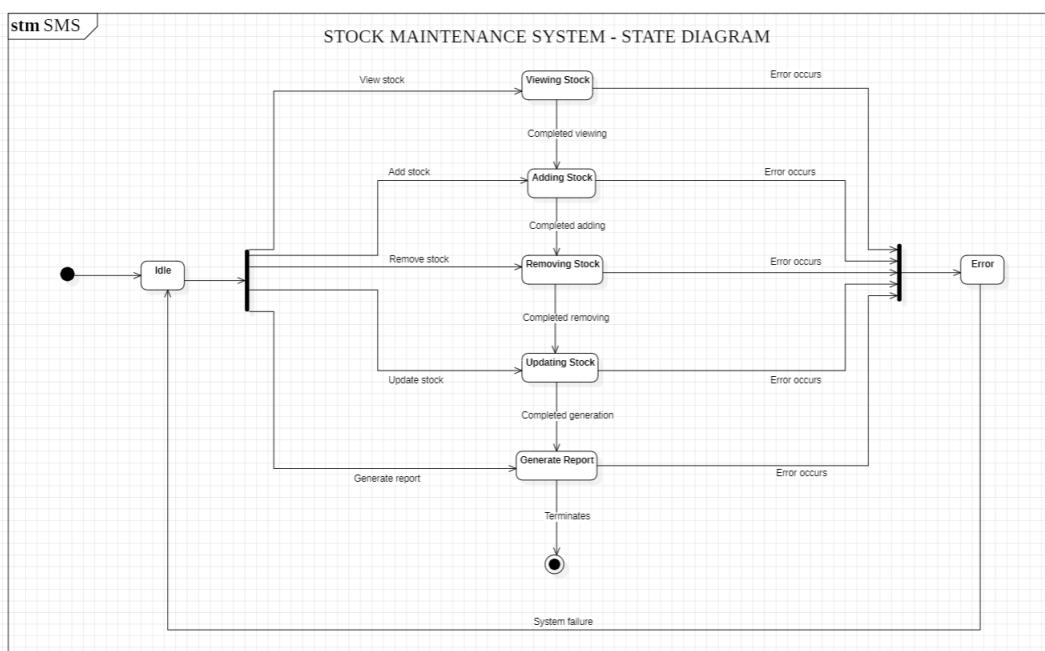
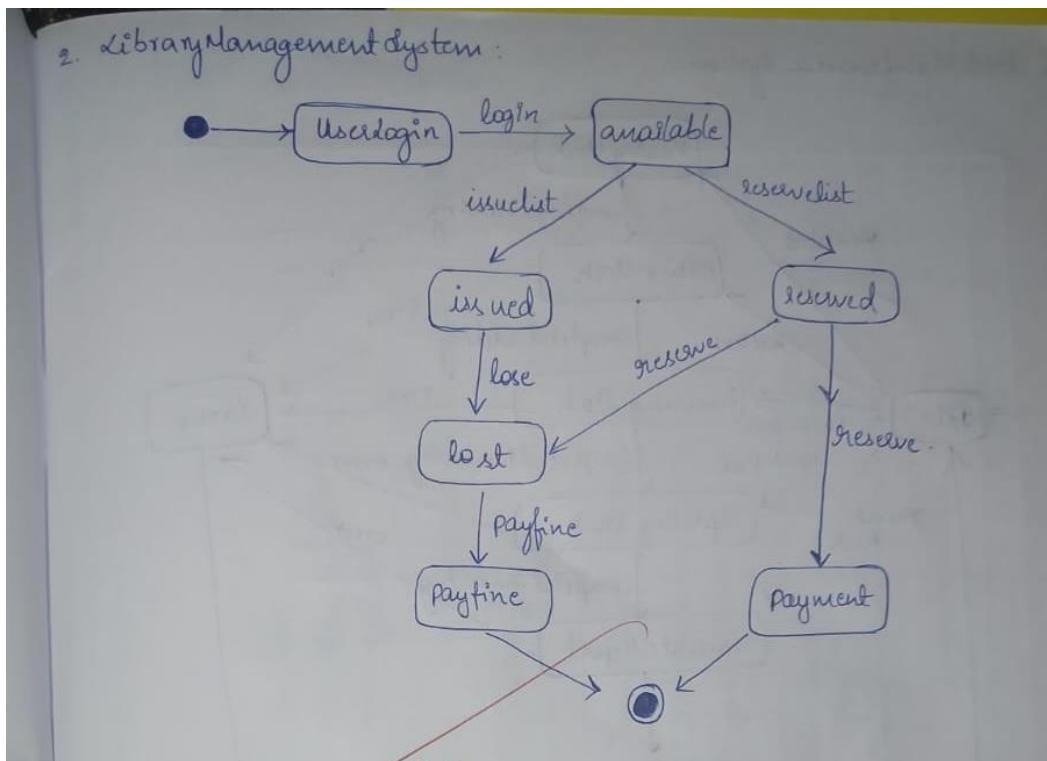
## CLASS DIAGRAM



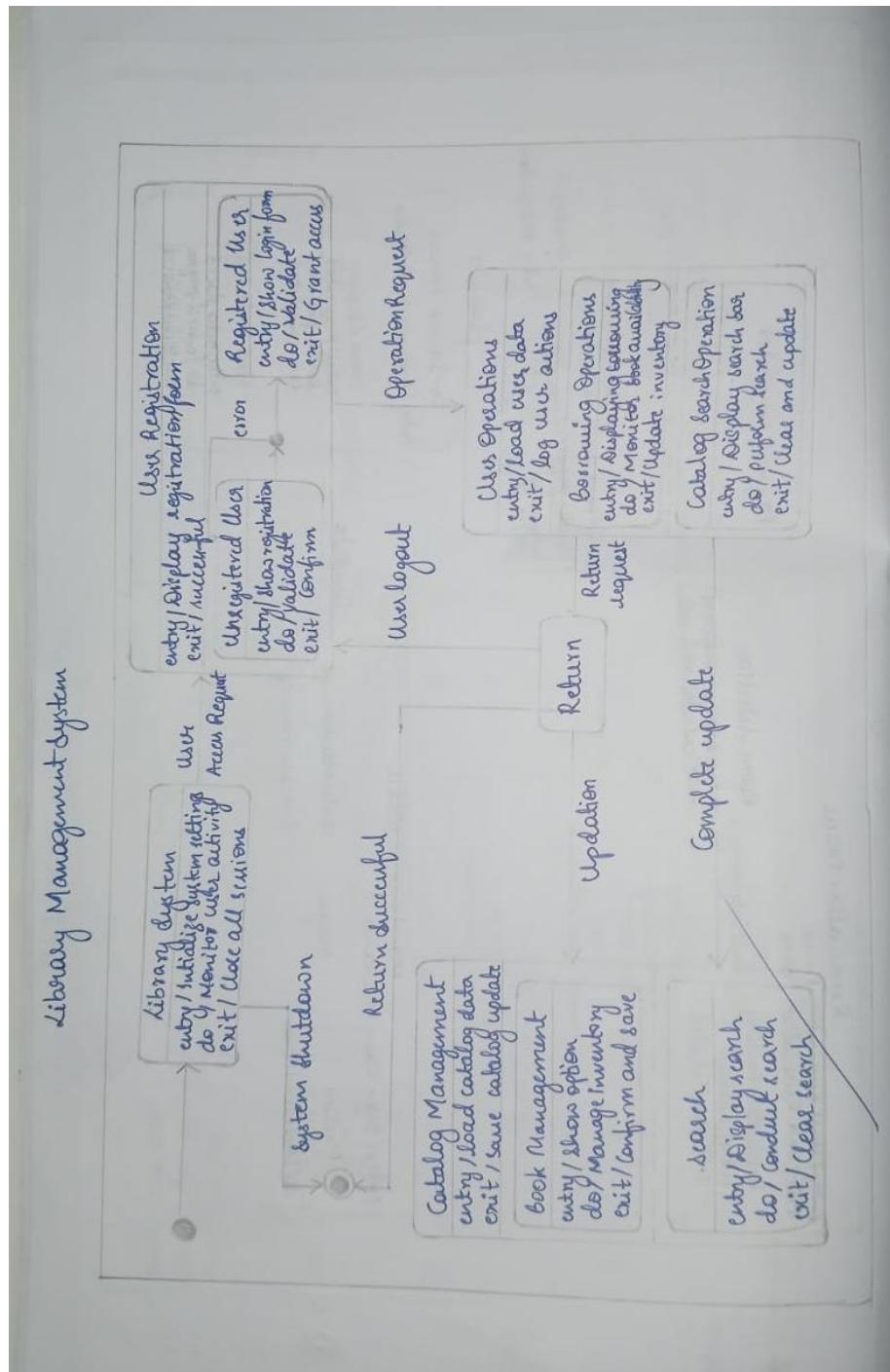
## LIBRARY MANAGEMENT SYSTEM - CLASS DIAGRAM

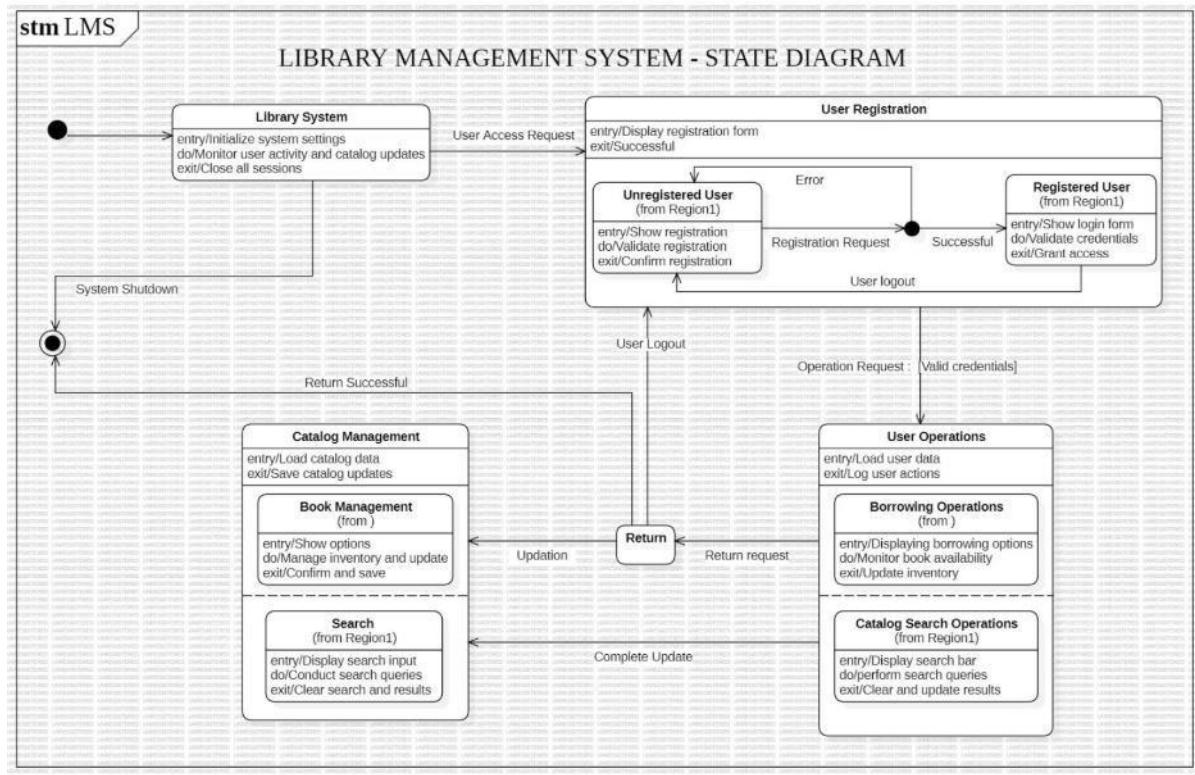


## STATE MODEL

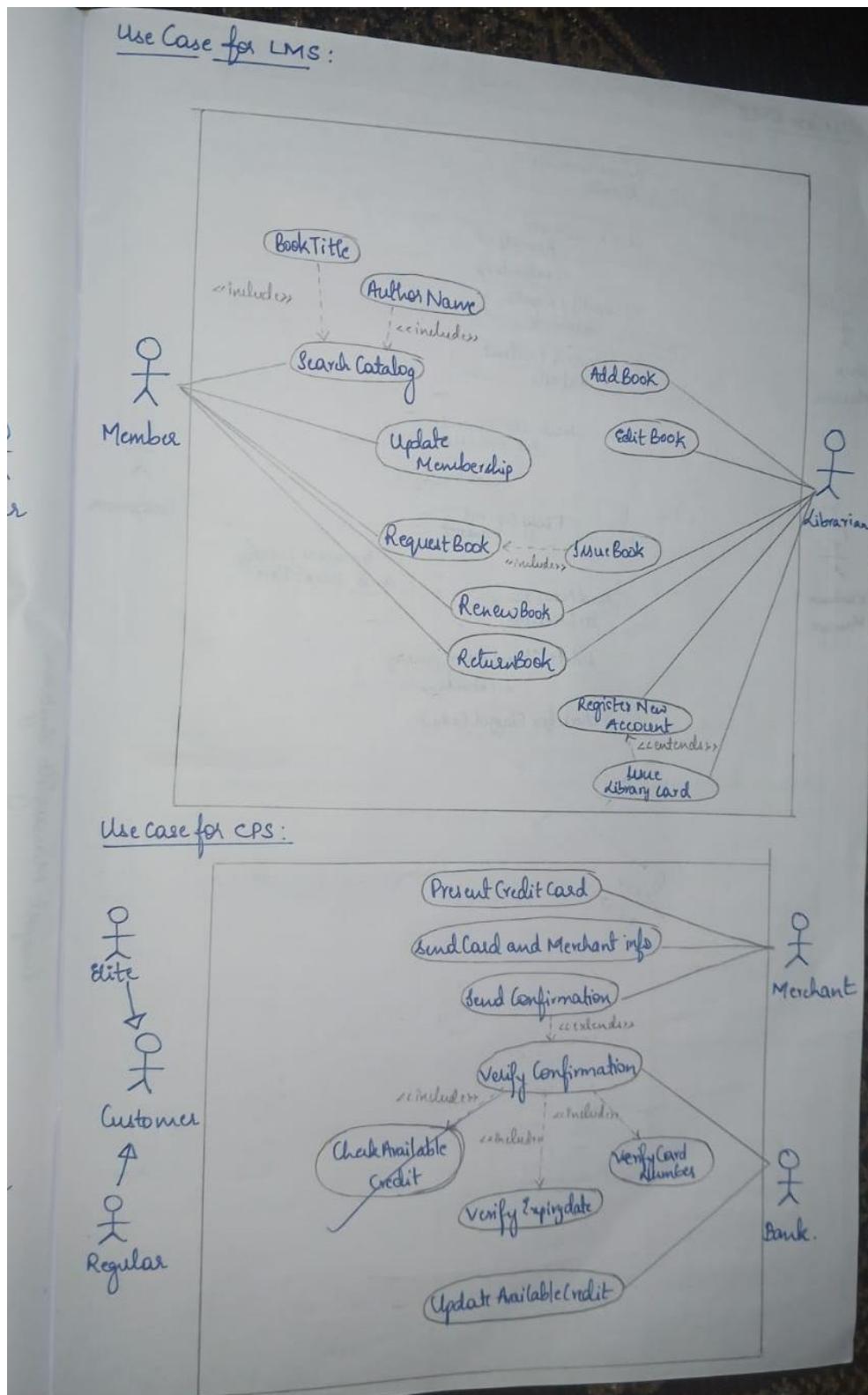


## ADVANCED STATE MODEL

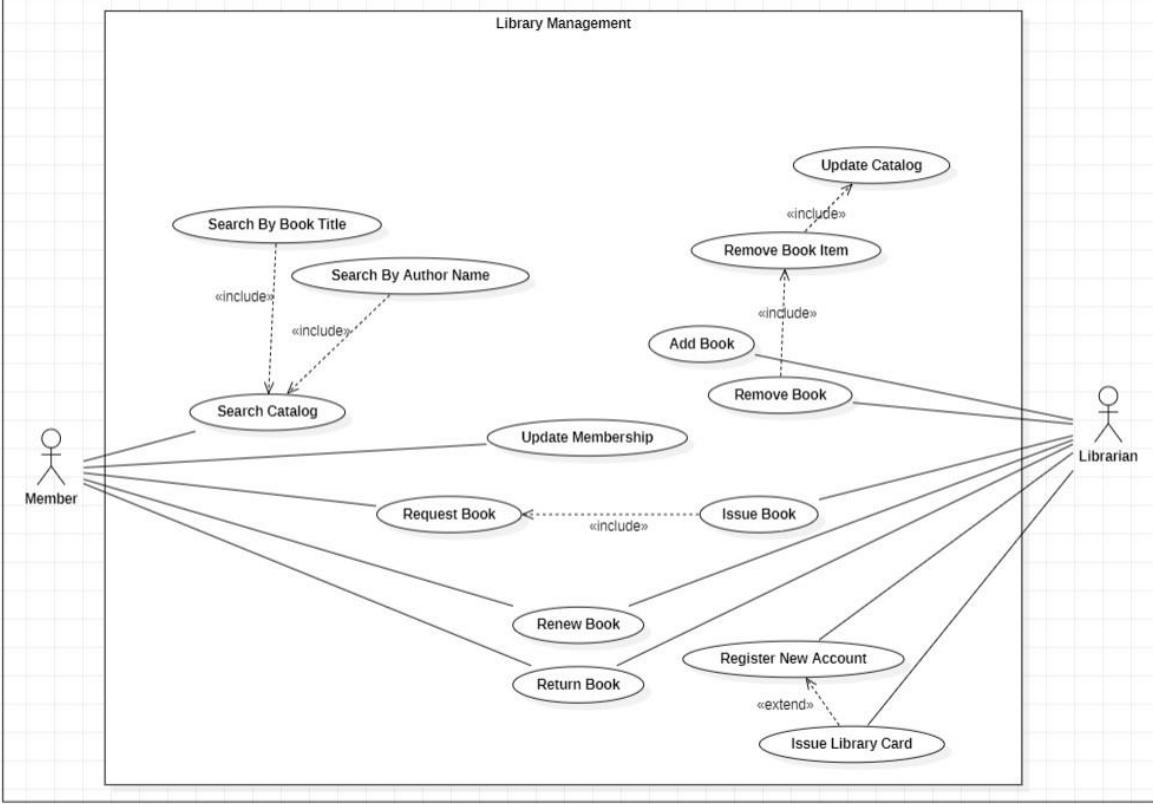




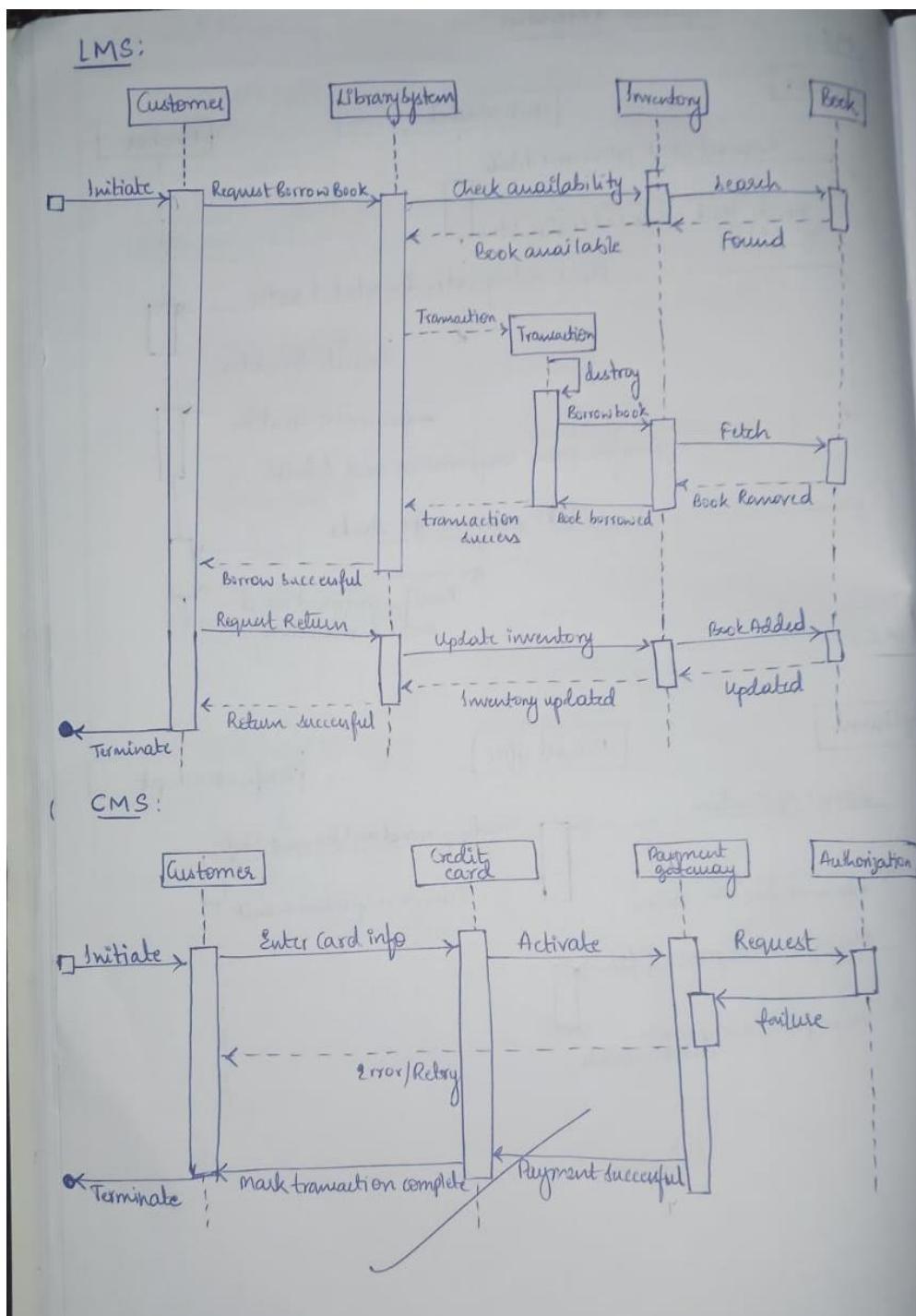
## USE CASE

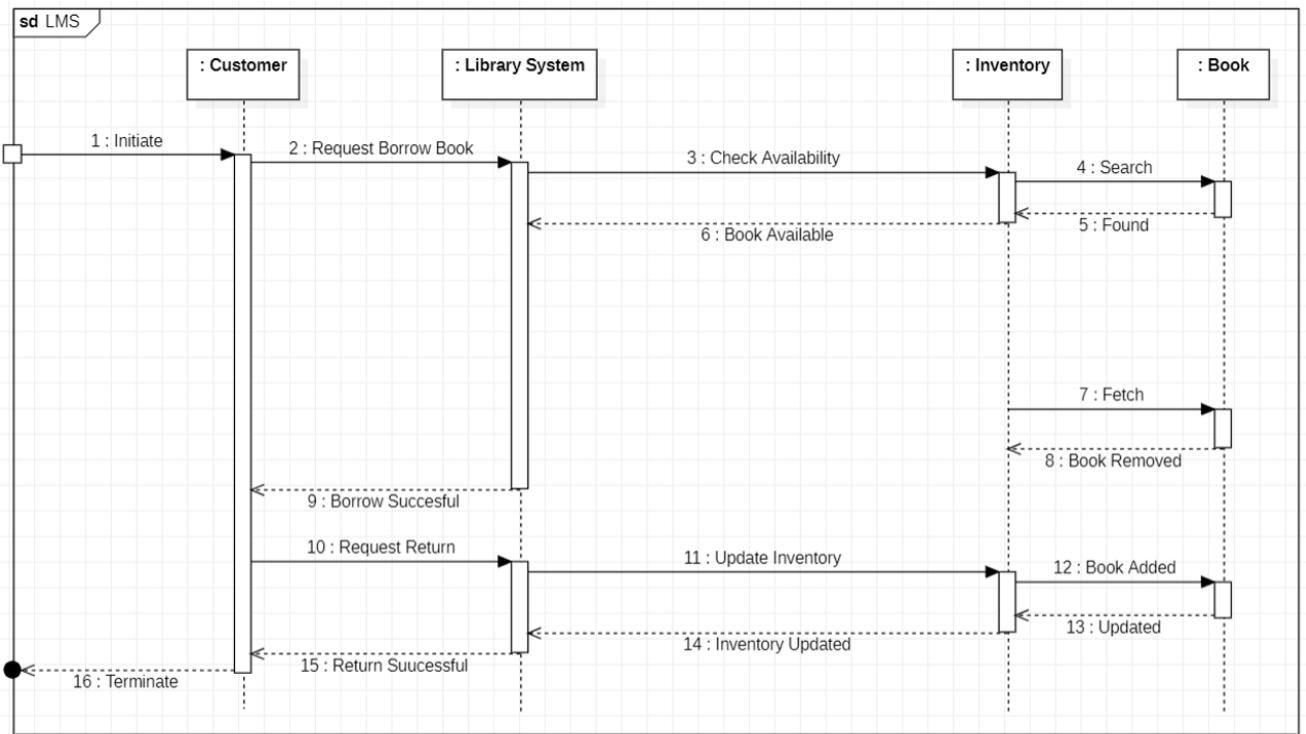


## LIBRARY MANAGEMENT SYSTEM USE CASE DIAGRAM

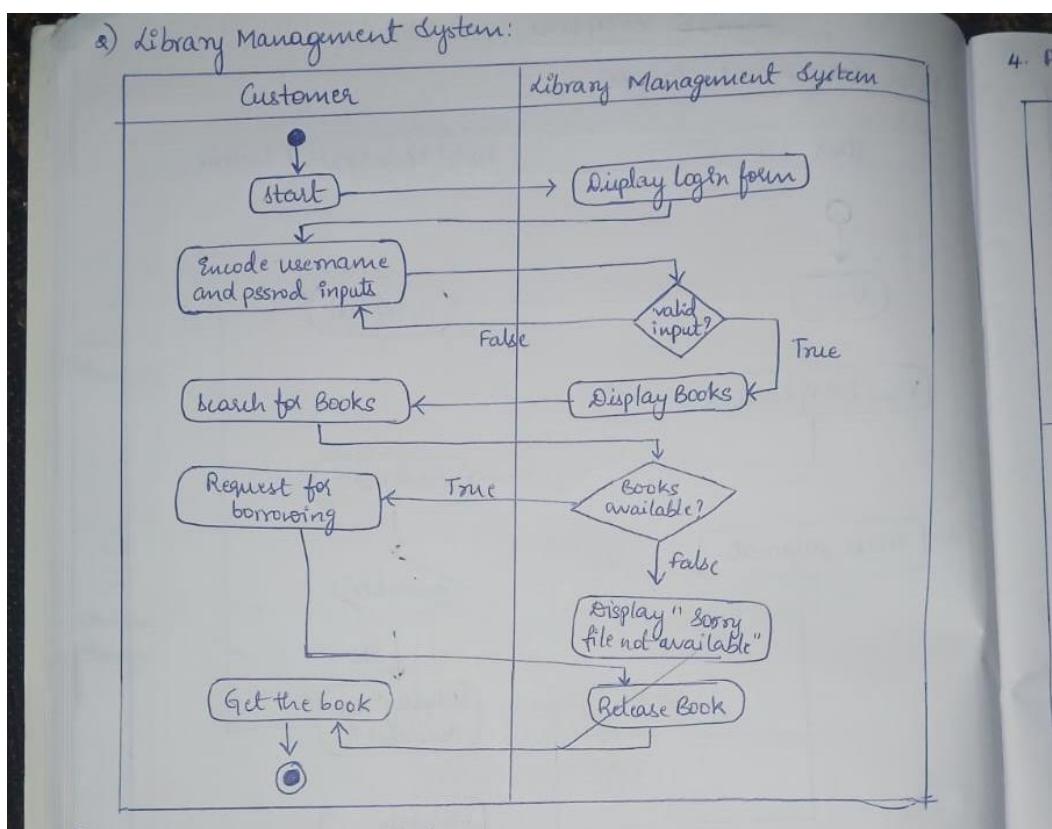


## SEQUENCE DIAGRAM





## ACTIVITY DIAGRAM



## LIBRARY MANAGEMENT SYSTEM - ACTIVITY DIAGRAM

