

# **B.M.S. COLLEGE OF ENGINEERING BENGALURU**

Autonomous Institute, Affiliated to VTU



Lab Record

## **Object Oriented Analysis and Design**

*Submitted in partial fulfillment for the 5<sup>th</sup> Semester Laboratory*

Bachelor of Technology  
in  
Computer Science and Engineering

*Submitted by:*

**Jayashree Tarai**

**(1BM24CS407)**

Department of Computer Science and Engineering

B.M.S. College of Engineering  
Bull Temple Road, Basavanagudi, Bangalore 560 019  
Aug-Dec 2025

**B.M.S. COLLEGE OF ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE AND**

**ENGINEERING**



***CERTIFICATE***

This is to certify that the Object-Oriented Analysis and Design(23CS6PCSEO) laboratory has been carried out by Jayashree Tarai (1BM24CS407) during the 5<sup>th</sup> Semester Aug-Dec-2025.

Signature of the Faculty Incharge:

NAME OF THE FACULTY: RoopaShree CS

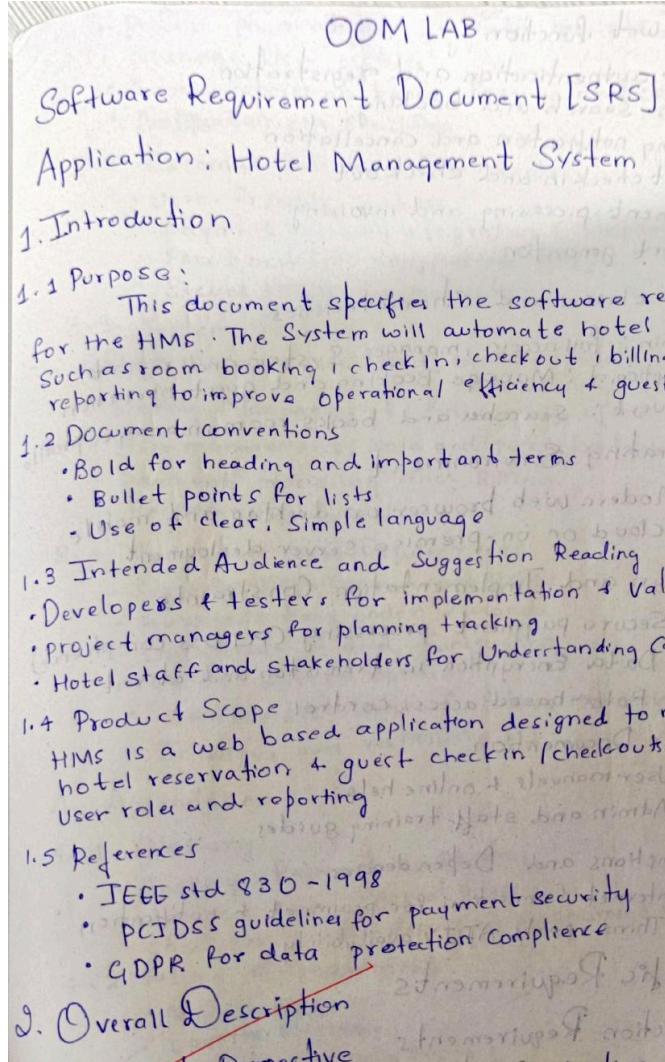
Department of Computer Science and Engineering  
B.M.S. College of Engineering, Bangalore

## Table of Contents

Sl no	content	Pg no
1.	Hotel Management System	4-8
2.	Credit Card Processing System	9-13
3.	Library Management System	14-18
4.	Stock Maintenance System	19-23
5.	Passport Automation System	24-28

# 1. Hotel Management System

## .1 SRS DOCUMENT



2.2 Product Function	BAJ MOO
• User authentication and registration	
• Room search and booking	
• Booking notification and cancellation	
• Guest check-in and check-out	
• Payment processing and invoicing	
• Report generation	
2.3 User classes and characteristics	
• Admin: full access, manages system and reports	
• Receptionist: Manage booking and guest operations	
• Guest: Searches and books rooms, manages profile	
2.4 Operating Environment	
• Modern web browser on desktop and mobile	
• Cloud or on-premise server deployment	
2.5 Design and Implementation Constraints	
• Secure payment processing (PCI DSS compliance)	
• Data encryption in transit and at rest	
• Role-based access control	
2.6 User Documentation	
• User manuals & online help	
• Admin and staff training guides	
2.7 Assumptions and Dependencies	
• Internet connection for payment & notification	
• Third-party API availability	
3. Specific Requirements	
3.1 Function Requirements	
• Secure user registration & login	
• Role-based access control	
• Room availability search by date, type & price	
• Book, modify & cancel reservation	
• Check-in/check-out processing	

- Process payment and generate receipts
- Manage User profiles
- Admin reports on booking Revenue and occupancy
- Notification via emails/SMS

### 3.2 External Interface Requirements

- User-friendly web UI
- Payment gateway integration (Stripe, PayPal)
- Email and SMS notification APIs
- Secure HTTPS Communication

### 3.3 System Features

- Room management: add/update/delete rooms and availability
- Booking Management: full booking lifecycle handling
- User management: role and access controls
- Payment processing and Billing
- Reporting Dashboard

### 3.4 Non-Functional Requirements

- Supports 100+ concurrent users
- Response time under 2 seconds
- Data Security and Encryption
- 99.9% system uptime
- Scalable architecture for future expansion
- Intuitive and responsive UI

## 4. Appendices

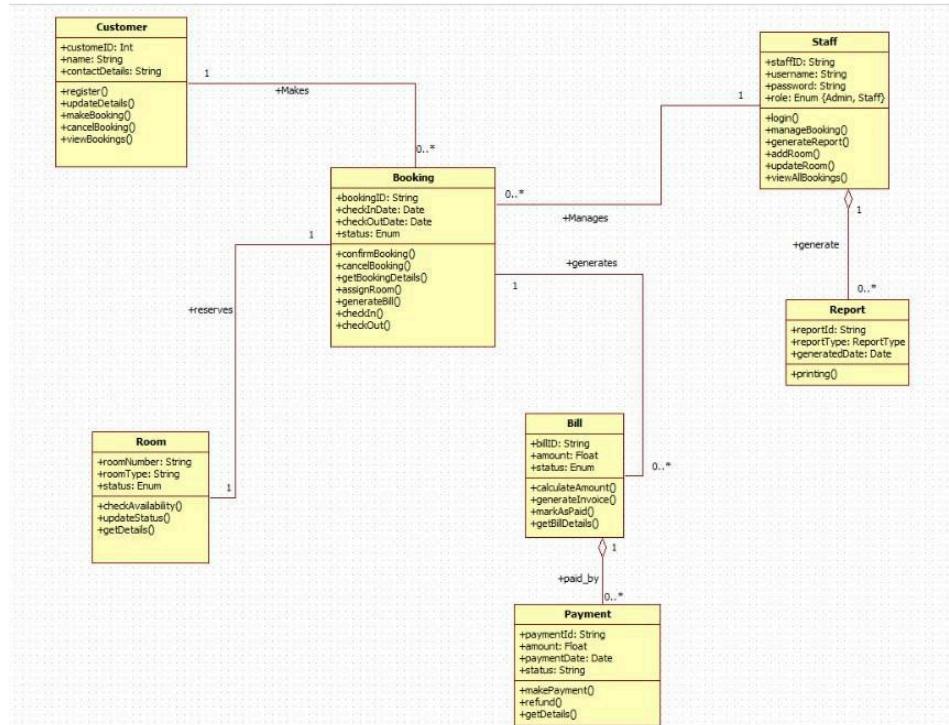
### 4.1 Glossary

- Booking: Reservation of rooms
- checkin: Guest Arrival process
- checkout: Guest departure & payment finalization

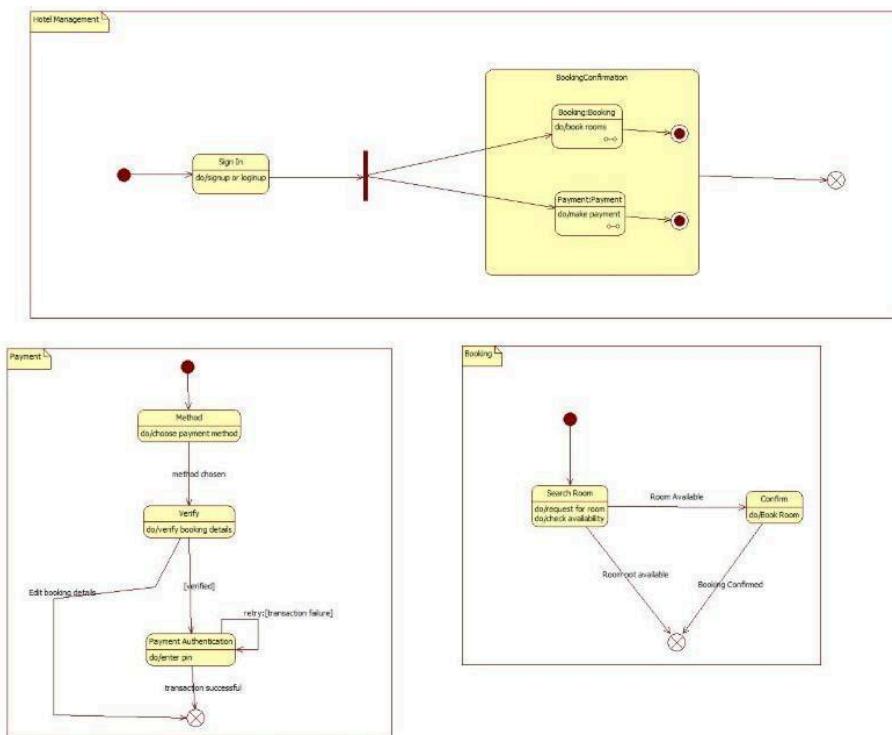
### 4.2 Future Enhancements

- Mobile Apps
- Loyalty Programs
  - Multi-language Support
  - Door lock System Integration
- Multi-language Support

## .2 CLASS DIAGRAM

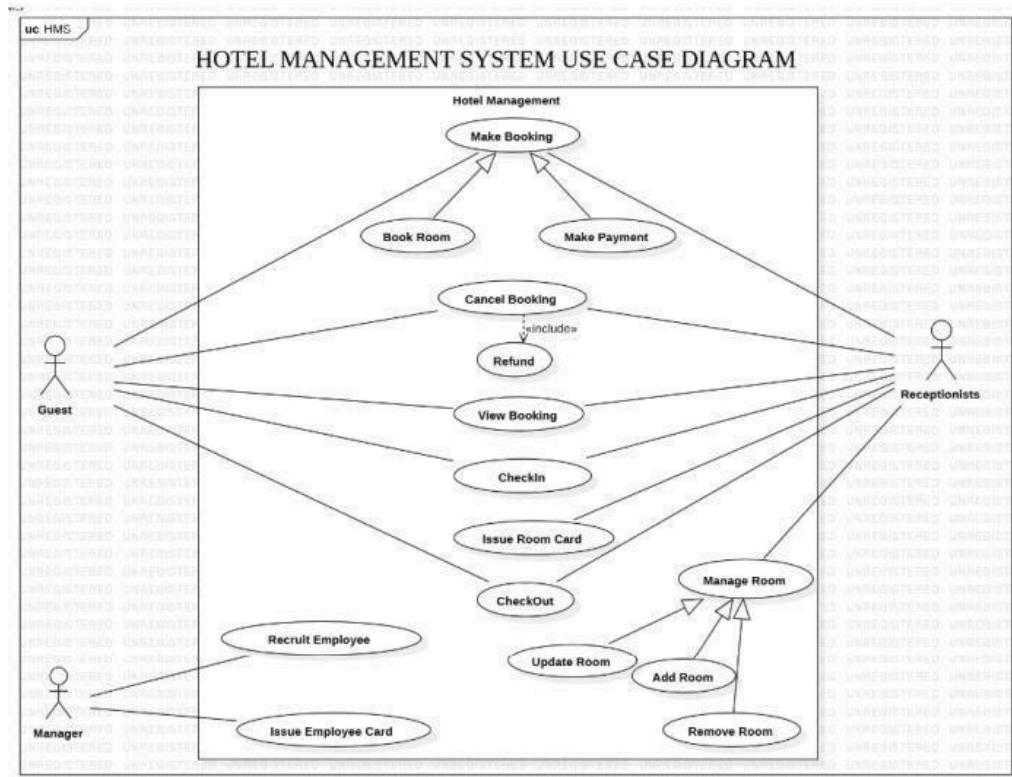


## .3 STATE DIAGRAM

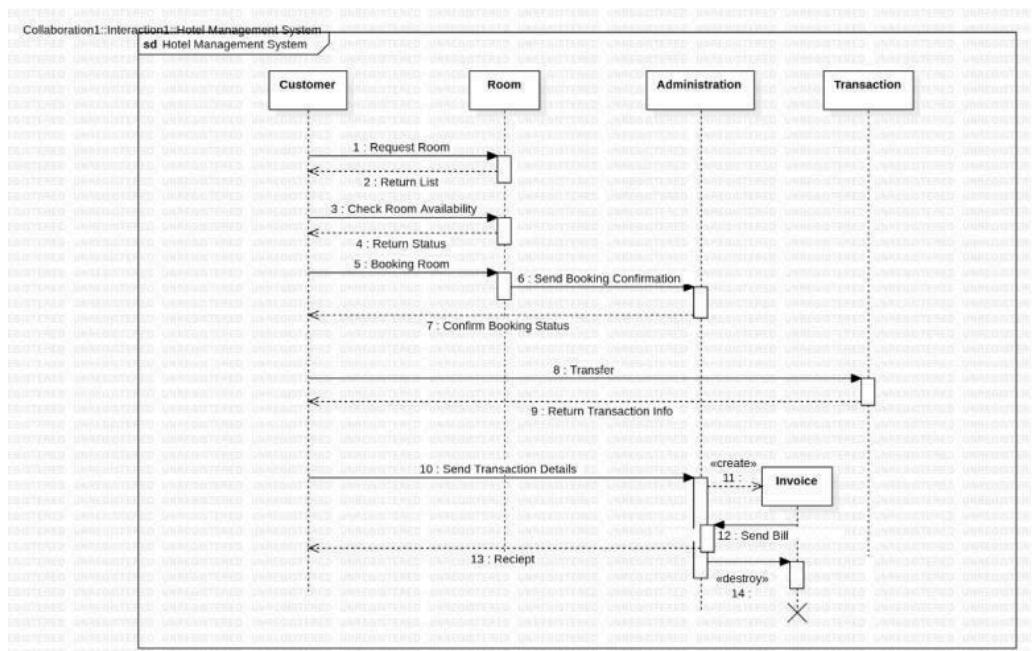


## .4 INTERACTION MODELS

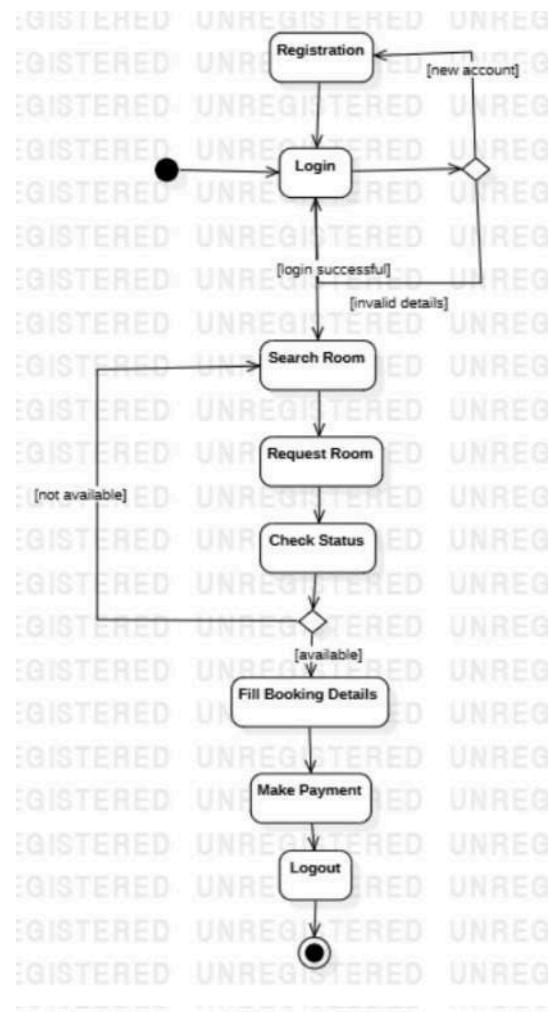
### a. Use case model



### b. Sequence model



### c. Activity model



## 2.Credit Card Processing

### 2.1 SRS DOCUMENT

**Application: Credit Card Processing System**

**1. Introduction**  
Credit card processing is interaction of the customer to the credit card processor allows getting a new card and also to get the details of the card and makes the copy of purchase made.

**1.1 Purpose**  
The credit card makes shopping easier and the customer or purchaser need not to carry cash along with them helps to keep track of all estimations.

**1.2 Document Convention**  
This document uses standard IEEE formatting. Requirements are stated using shall for mandatory features. Use case references are italicized. Number Sections Correspond to specific categories of System requirements.

**1.3 Intended Audience and Reading Suggestion**  
This SRS is intended for developers, project managers, testers and stakeholders. Readers are encouraged to start with the introduction for context, followed by overall system description and specific requirements based on their roles.

**1.4 Project Scope**  
The CCPS is a Secure, cloud-based system that processes credit card payments. It supports authorization, settlement, refund handling and fraud detection. The system aligns with organizational goals to modernize and secure transaction handling.

**1.5 References**  
Include PCI DSS 4.0, ISO 8583, messaging standards, credit card network API guide.

**2. Overall Description**

**2.1 Product Perspective**  
CCPS is a standalone system replacing older payment system. It interface with card networks, merchant system, and internal banking API's via secure channel. The system supports integration with external services through standardized API's.

**2.2 Product Function**  
The system authorizes transaction, captures payment, process settlement, handles chargeback and generate reports. It includes user management, fraud detection and web-based admin portal.

**2.3 User classes and characteristics**  
User groups include customers, merchant, admin staff and developers. Each group has different access levels and technical expertise, with administrators requiring more advanced features for monitoring & control.

**2.4 Operating Environment**  
The system runs on a cloud platform with Linux servers, Docker containers & PostgreSQL database. Web access is through modern browsers and API use.

**2.5 Design and Implementation**  
Constraints include compliance with PCI DSS integration using ISO 8583 standards.

**2.6 User Documentation**  
User manuals, admin guides, API documentation will be provided in both HTML & PDF formats.

**2.7 Assumption & Dependencies**  
The system depends on external card networks, stable banking API, and ongoing access to cloud infrastructure.

### 3. Specific Requirements

#### 3.1 Functional Requirements

- Authorize transaction with 3sec
- capture payment after merchant confirm
- process merchant - initiated refunds
- keep transaction logs for 2 years

#### 3.2 External Interface Requirements

- Use standard verification for payment network messaging
- provide restful API's for merchant

#### 3.3 System Features

- Role based access control
  - Real time monitoring dashboard
  - Automated backup & recovery
- #### 3.4 Non-functional Requirements
- 99.9% System availability
  - Encrypt all Sensitive data
  - Comply with PCI
  - Handle 10,000 transaction / sec
  - UI responses with 2 seconds

### 4 Appendices

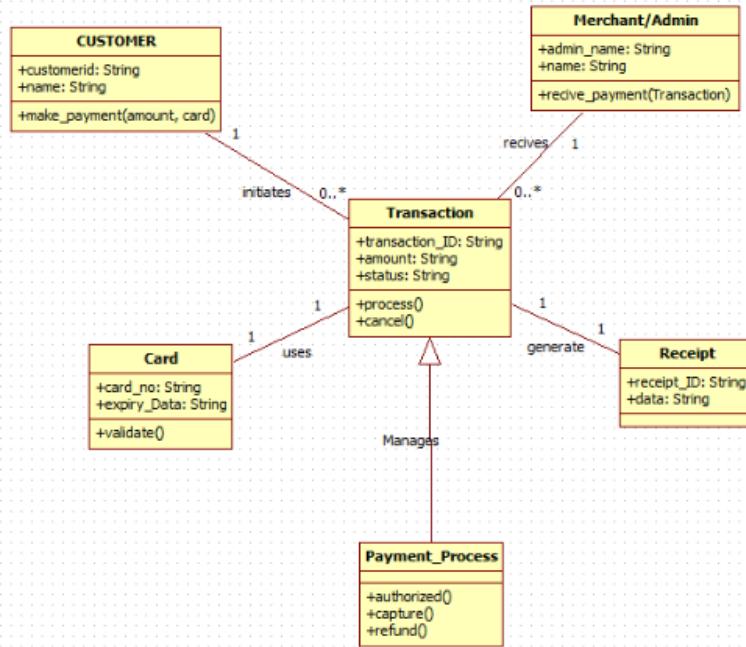
#### 4.1 Glossary

- \* Authorization validation
- \* capture : finalizes payment
- \* PCI DES : security standard
- ISO8583 : Transaction message format

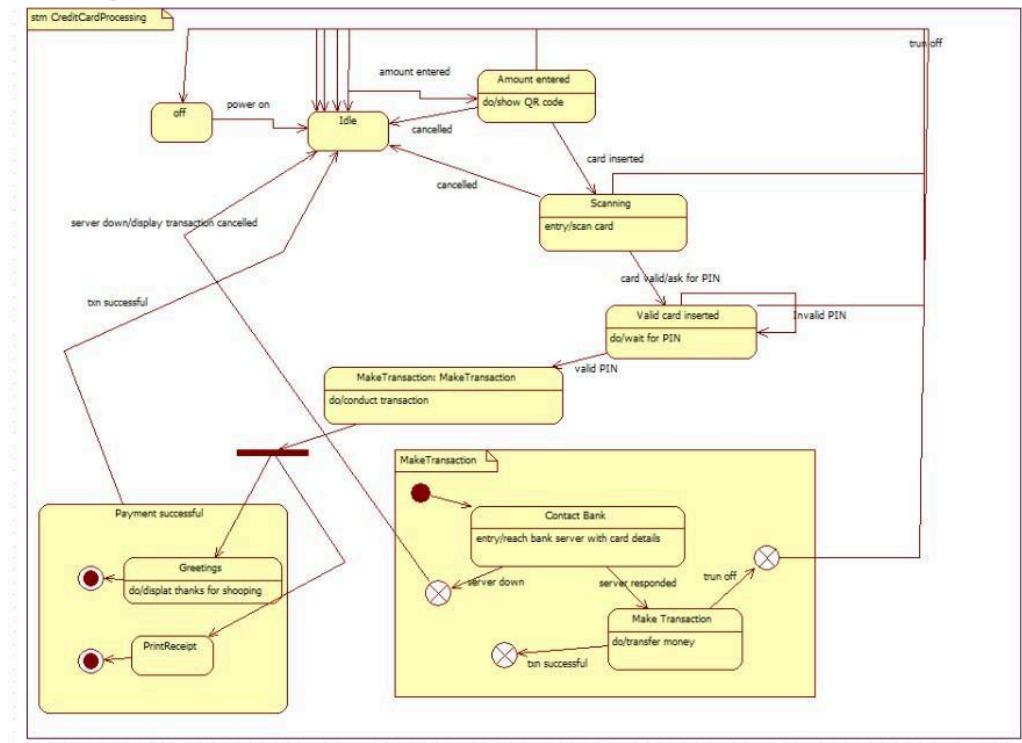
#### 4.2 Future Enhancement

- ML Based fraud detection
- mobile / digital Wallet
- Multicurrency support

## 2.3 Class Diagram

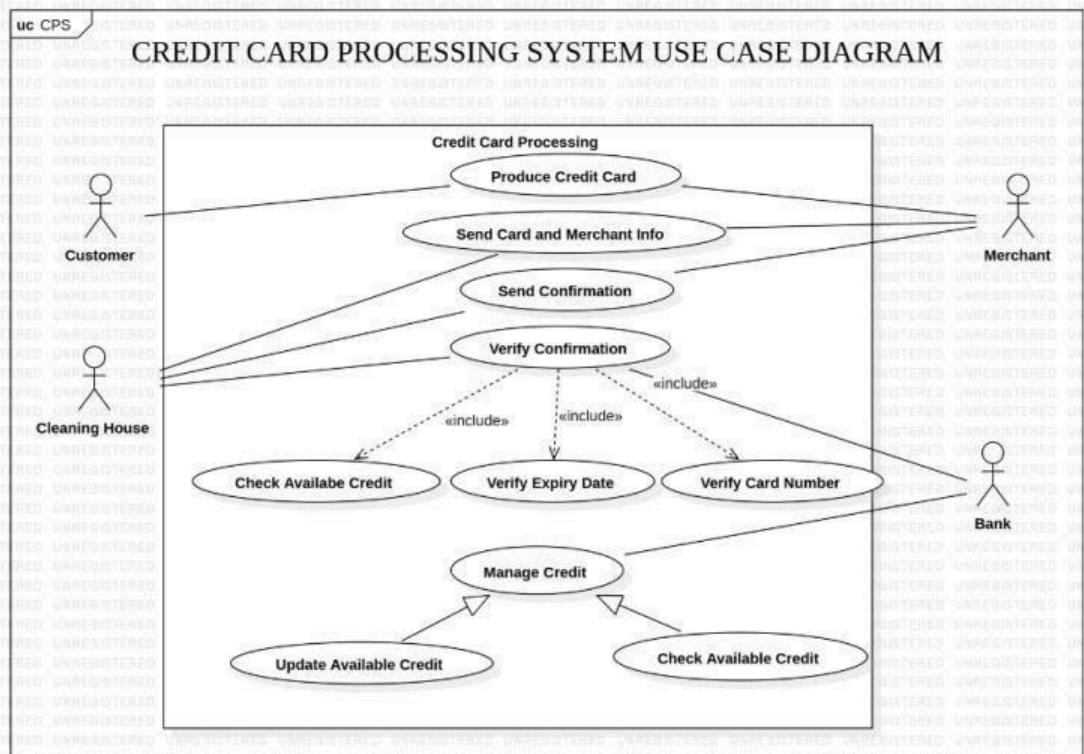


## 2.4 State Diagram

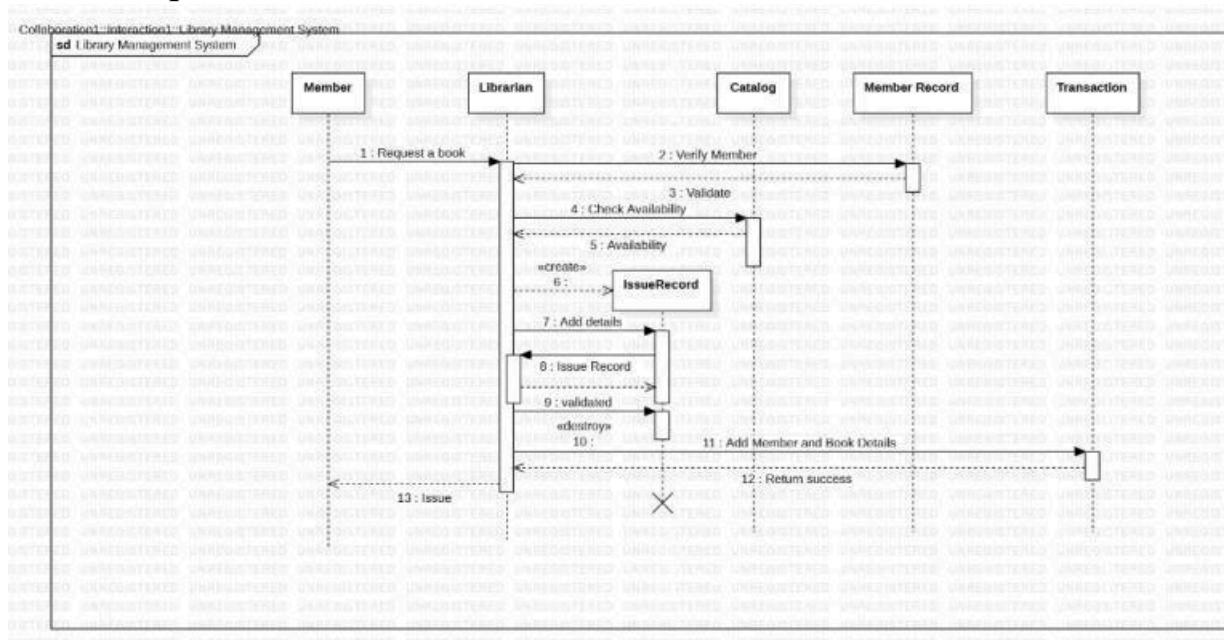


## 2.5 Interaction Models

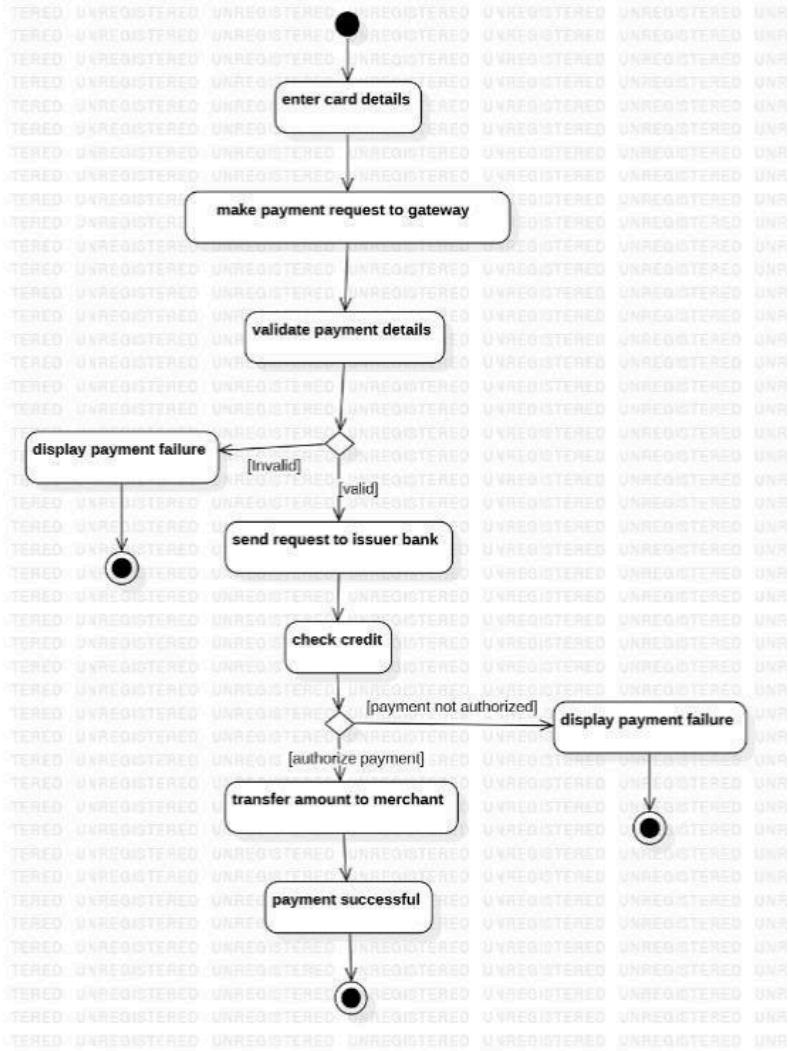
### a. Use Case Model



### d. Sequence model

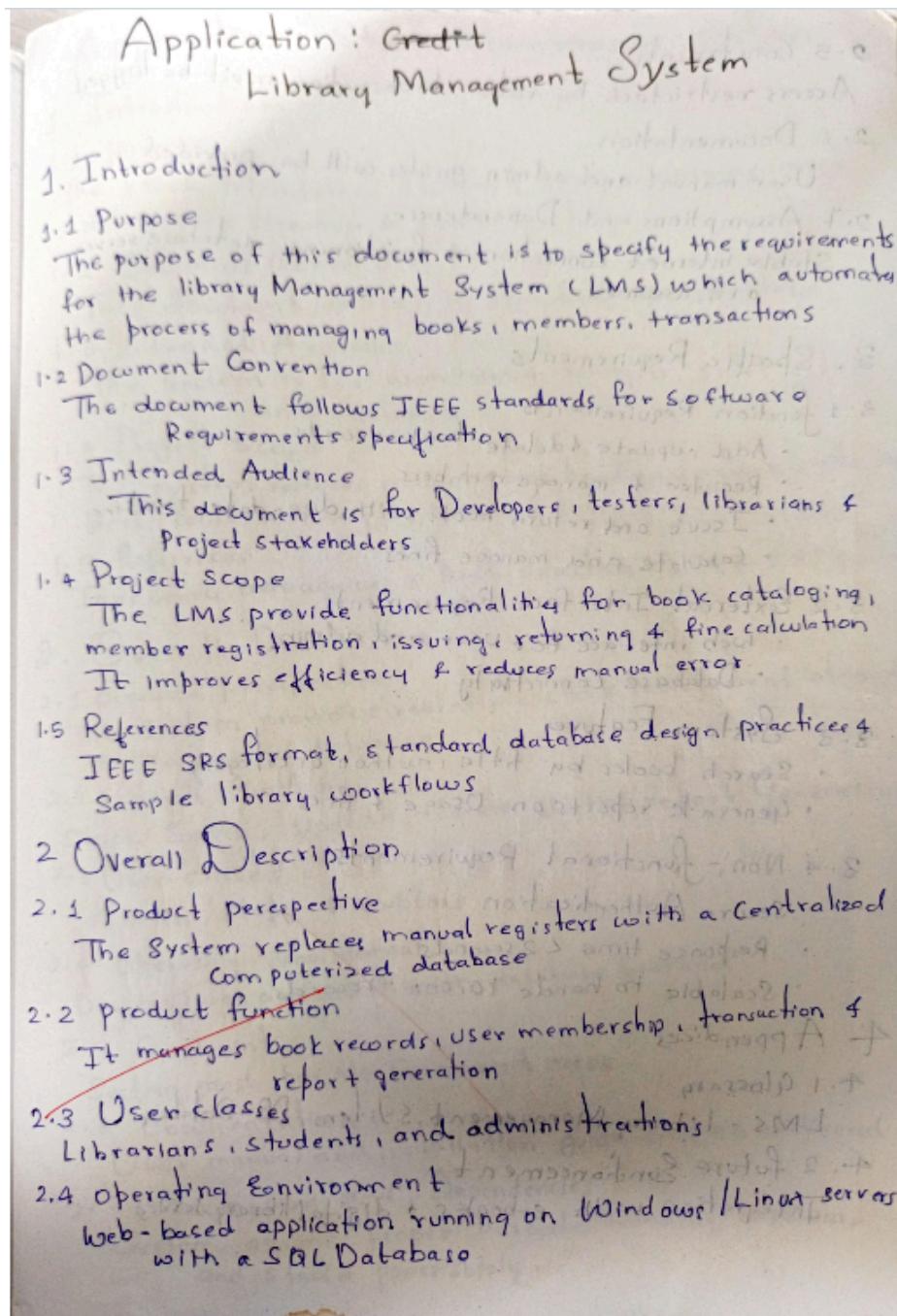


### c. Activity Model



### 3. Library management system

#### 3.1 SRS Document



## 2.5 Constraints

Access restricted by roles: transaction must be logged

## 2.6 Documentation

User manual and admin guides will be provided

## 2.7 Assumptions and Dependencies

Stable internet connection & functioning database server  
are assumed

## 3. Specific Requirements

### 3.1 Function Requirements

- Add, update & delete book records
- Register & manage members
- Issue and return books with due-date tracking
- Calculate and manage fines

### 3.2 External Interface Requirements

- Web interface for user and admin
- Database connectivity

### 3.3 System Features

- Search books by title, author or category
- Generate reports on Usage & fines

### 3.4 Non-functional Requirements

- Secure Authentication
- Response time < 2 seconds
- Scalable to handle 10,000+ records

## 4 Appendices

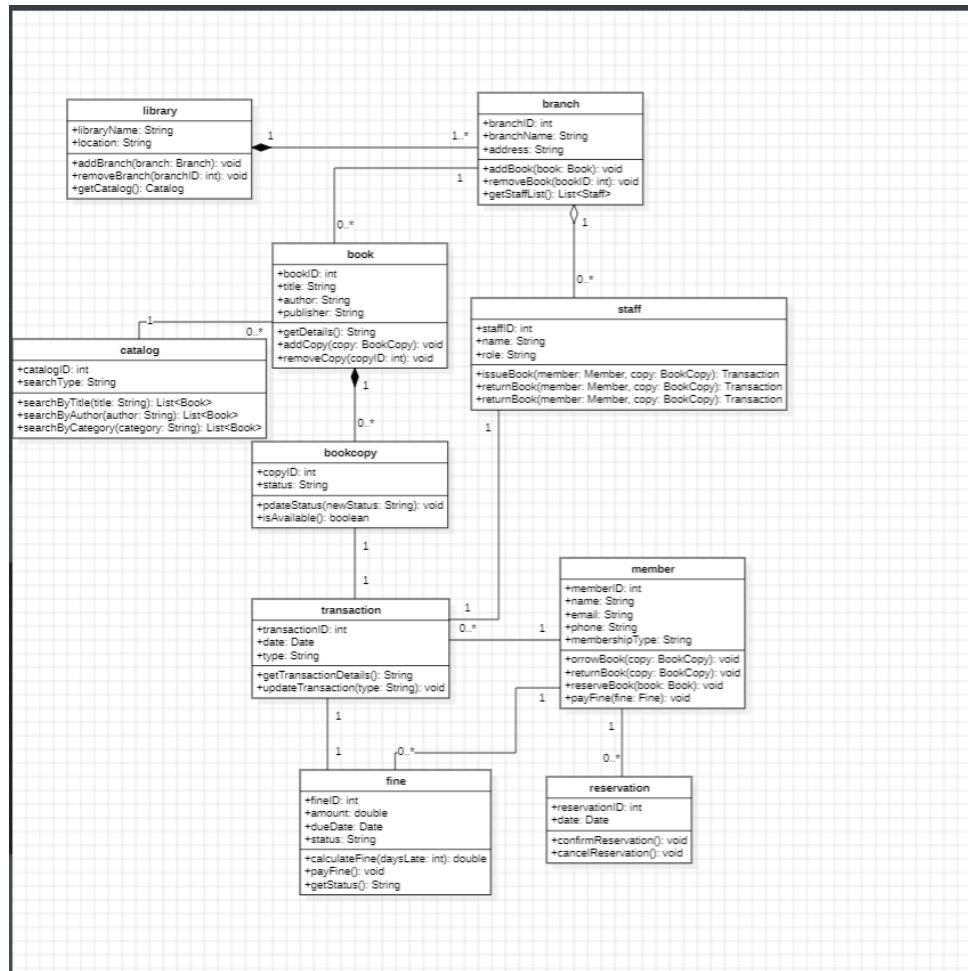
### 4.1 Glossary

LMS - Library Management System, DB - Database

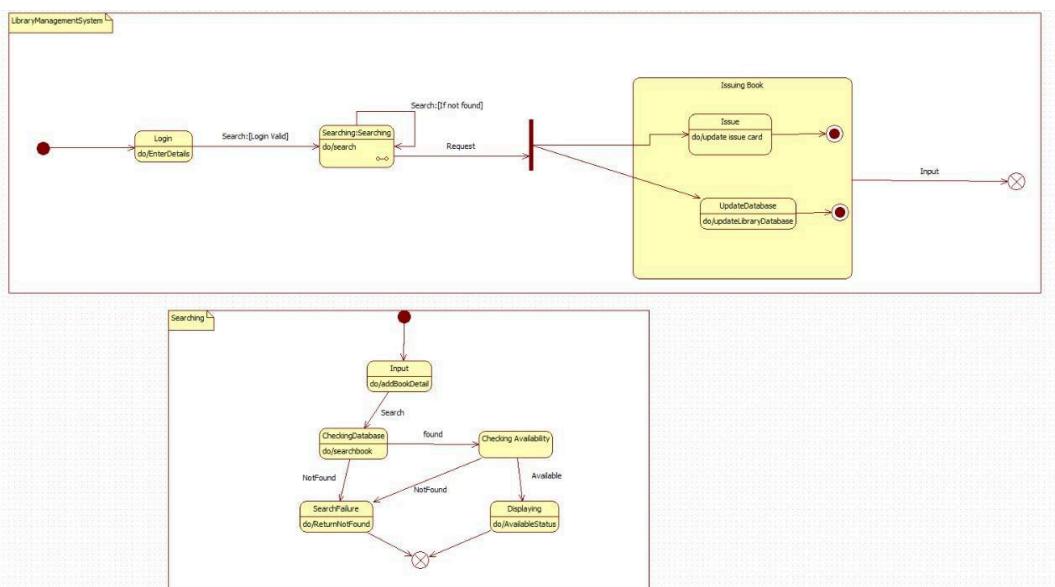
### 4.2 Future Enhancement

Integration with e-books & digital library services

### 3.3 Class Model

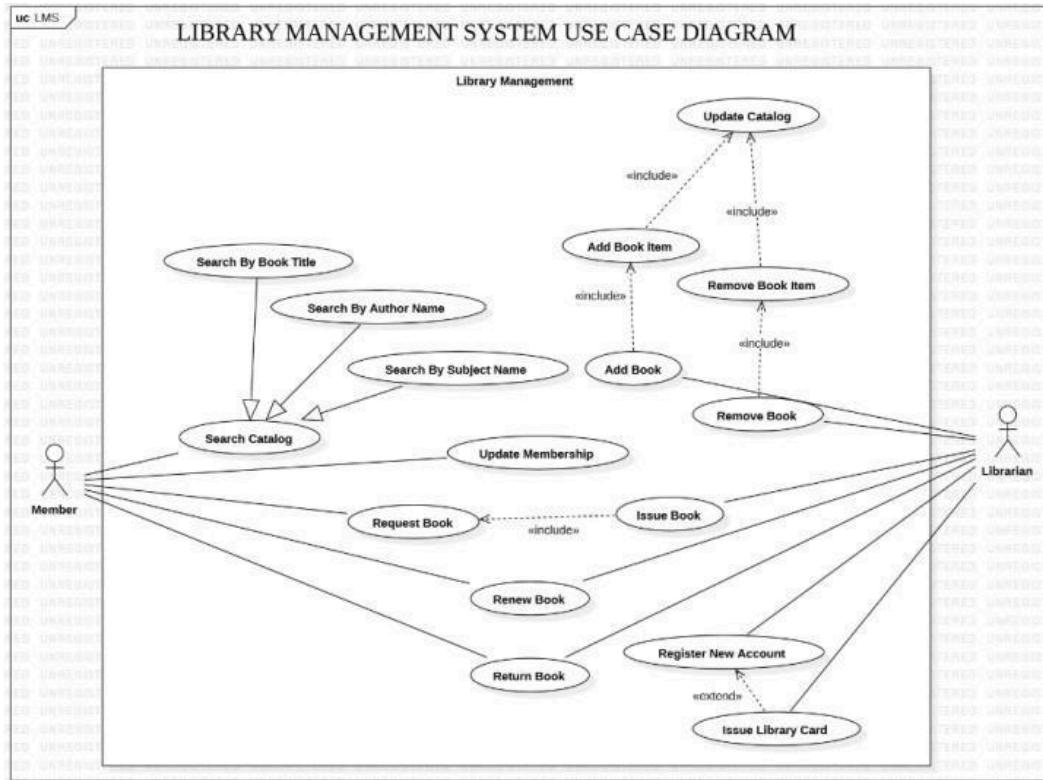


### 3.4 State model

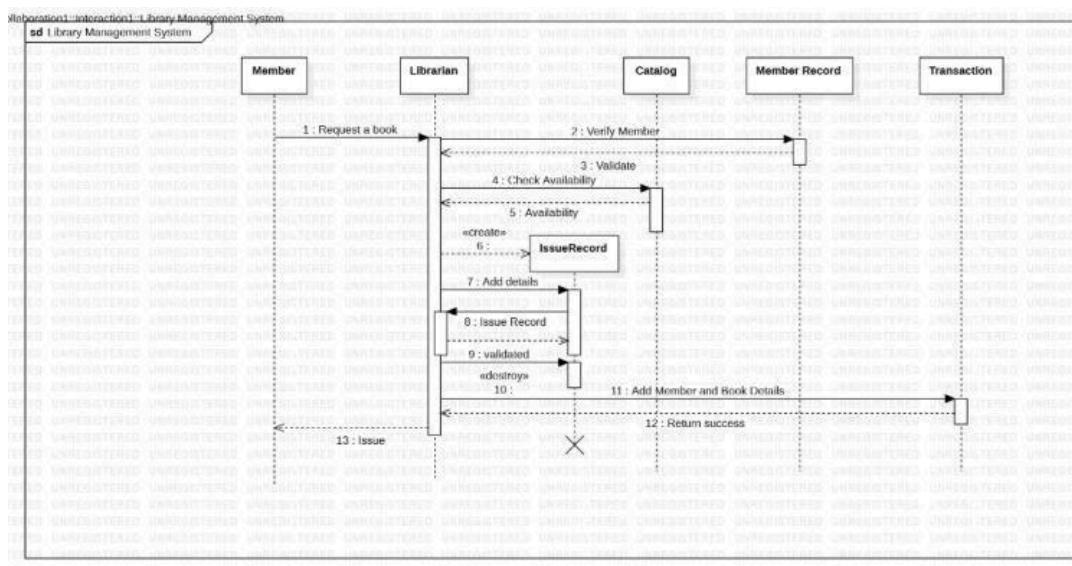


## 3.5 Interaction Models

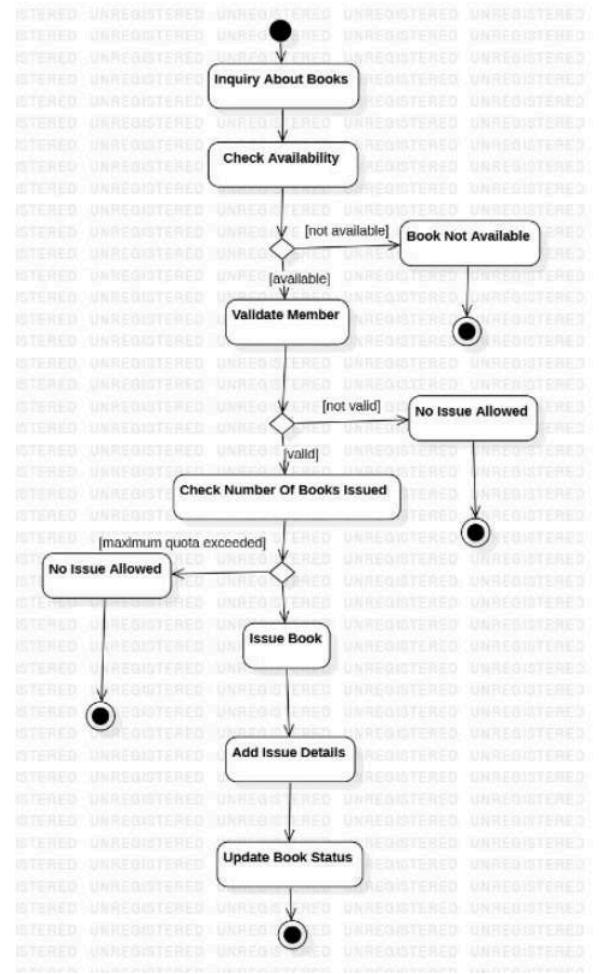
### a. Use Case Model



### b. Sequence Model



### c. Activity Model



## 4. Stock Maintenance System

### 4.1 SRS Document

**Application - Stock Maintenance System**

#### 1. Introduction

##### 1.1 Purpose

The Stock Maintenance System (SMS) maintains inventory details and streamlines stock tracking.

##### 1.2 Document Convention

This document adheres to standard SRS structure.

##### 1.3 Intended Audience

The system is for warehouse managers, sales staff, and developers.

##### 1.4 Project Scope

The system reduces manual stock handling, prevents shortages, and ensures timely updates of inventory.

##### 1.5 References

Inventory management best practices, IEEE SRS guidelines.

#### 2. Overall Description

##### 2.1 Product Perspective

The system provides real-time stock management integrated with a central database.

##### 2.2 Product Function

Stock Entry, Update, Reorder Alerts & Report Generation.

##### 2.3 User Classes

Admin, staff & auditors.

##### 2.4 Operating Environment

Desktop/web application with database backend.

##### 2.5 Constraints

System must handle concurrent access.

##### 2.6 Documentation

User manual and installation guide will be delivered.

##### 2.7 Assumption and Dependencies

System assumes proper barcode scanning hardware and stable power supply.

### 3. Specific Requirements

#### 3.1 Functional Requirements

- Add and update stock items
- Track stock levels with alerts
- Generate purchase orders when below threshold
- produce sales & inventory reports

#### 3.2 External Interface Requirements

- Barcode Scanner Support
- Database and reporting modules

#### 3.3 System Features

- Real-time stock tracking
- Low stock notifications

#### 3.4 Non-functional Requirements

- 99% availability
- Secure data storage
- Accuracy tolerance <1%

### 4. Appendices

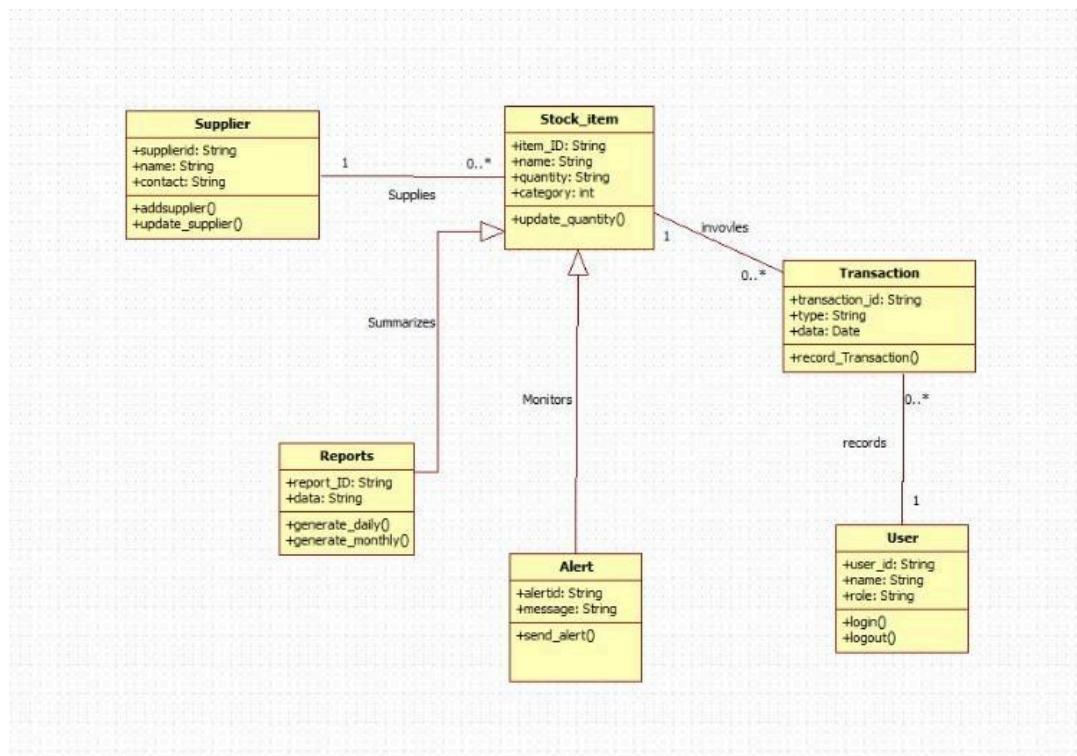
#### 4.1 Glossary

SMS - Stock Maintenance System, SKU - Stock Unit

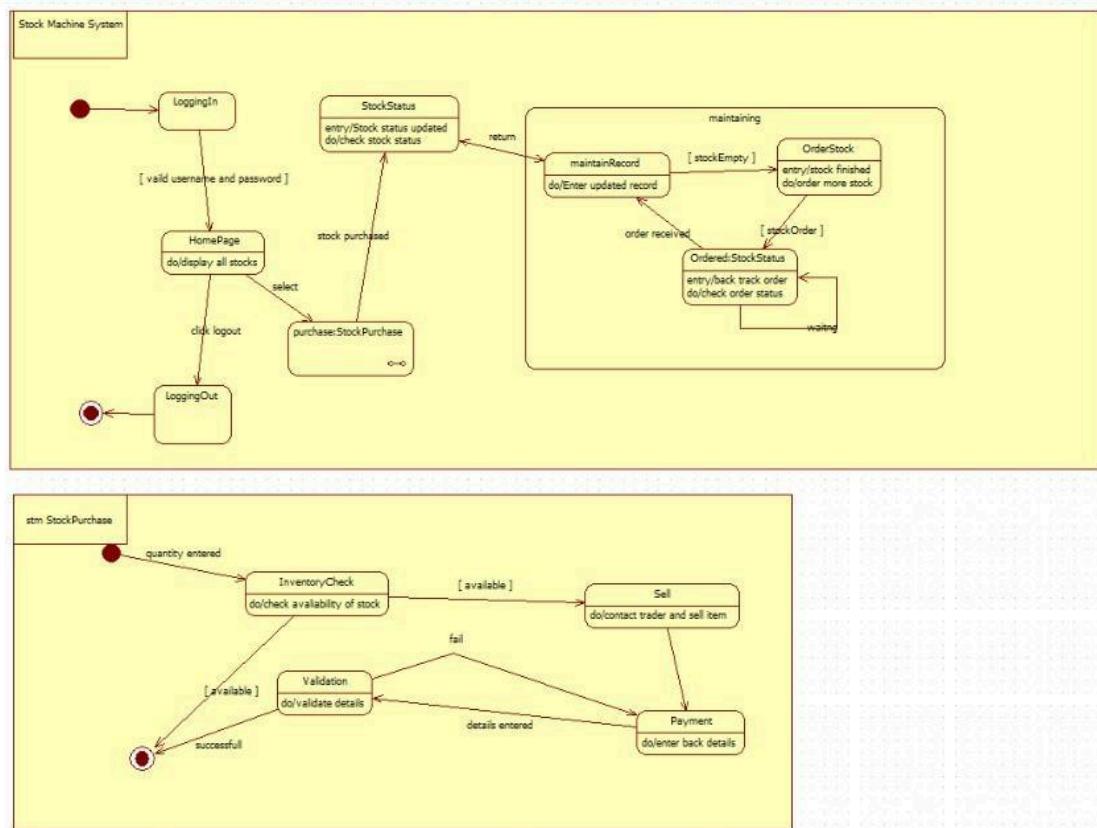
#### 4.2 Future Enhancement

Mobile app integration + AI-based demand forecasting

## 4.3 Class model

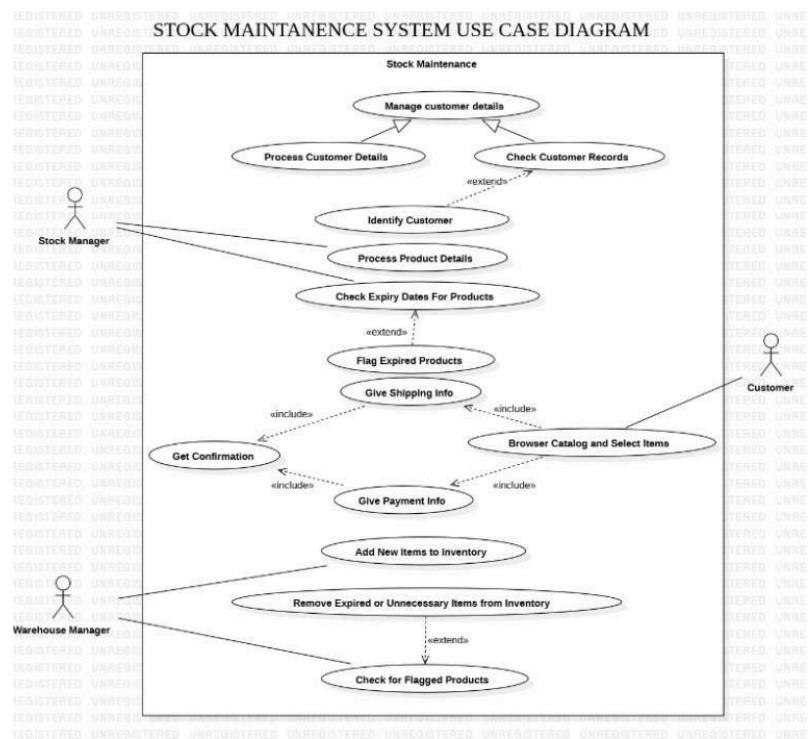


## 4.4 State Model

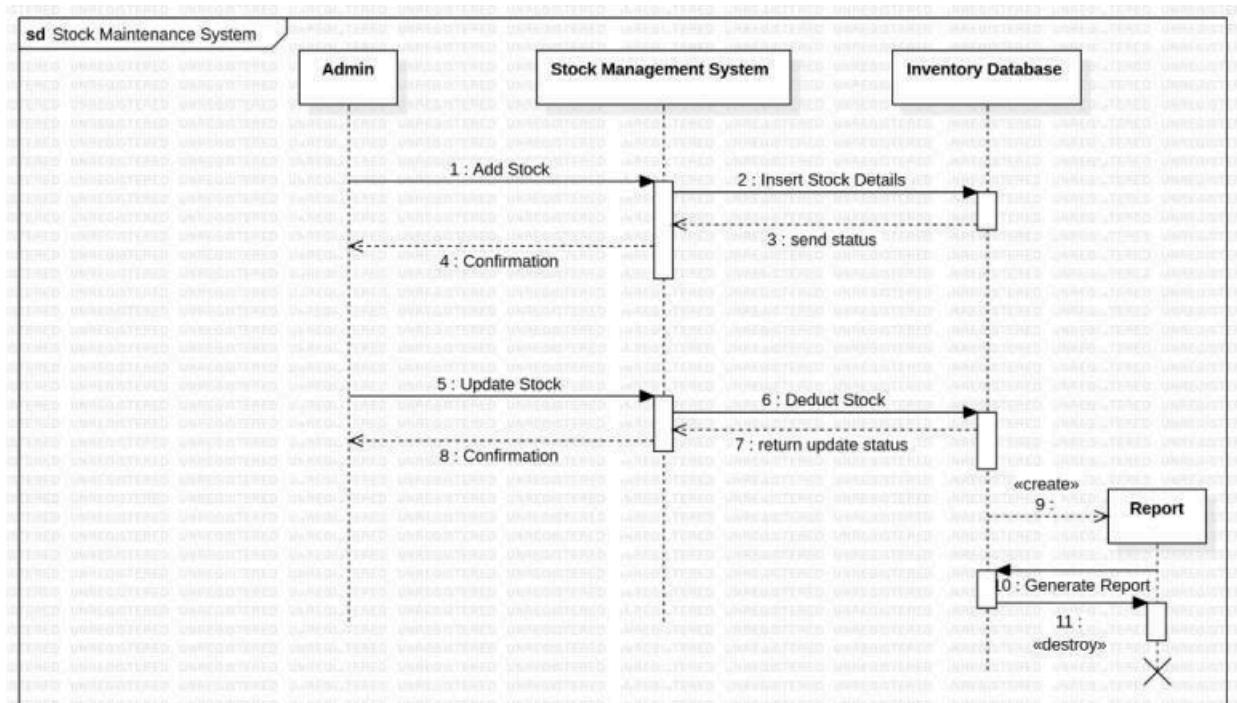


## 4.5 Interaction Models

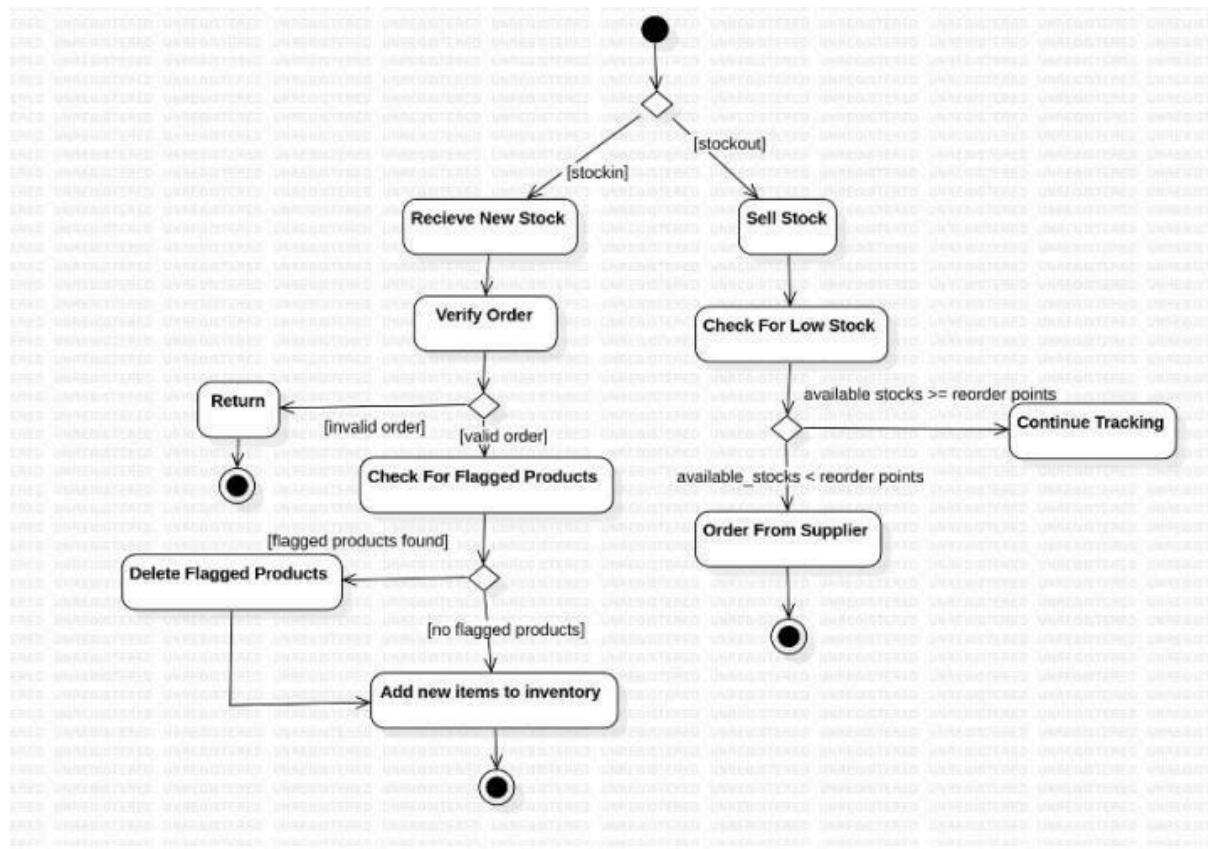
### a. Use Case Model



### d. Sequence Model



## e. Activity Diagram



## 5 .Passport Automation System

### 5.1 SRS Document

Application : Passport Automation System

1. Introduction

1.1 Purpose  
This document specifies requirements for an automated Passport Automation System (PAS) that handles passport & application & verification online.

1.2 Document Convection  
IEEE style has been followed for clarity and consistency.

1.3 Intended Audience  
Government officials, applicants & development teams.

1.4 Project Scope  
The system replaces manual passport processing ensuring faster application handling & tracking.

1.5 References  
Government Passport issuance guidelines, IEEE SRS Standards.

2. Overall Description

2.1 Product Perspective  
The system provides an online platform connected to the government database.

2.2 Product function  
Application Submission, Verification, payment & appointment scheduling.

2.3 User classes  
Applicants, Verification officers & administrators.

2.4 Operating Environment  
Web-based system with secure authentication, connected to national databases.

2.5 Constraints  
Must comply with government regulations and data security policy.

2.6 Documentation  
User guidelines and admin manuals will be supplied.

2.7 Assumptions and Dependencies  
Assume connectivity with police & national ID Database

### 3. Specific Requirements

#### 3.1 Functional Requirements

- \* Submit online application
- \* Upload required documents
- \* Schedule appointments
- \* Verify applicant details with police & government records
- \* process online payments

#### 3.2 External Interface Payment

- web portal for users
- Integration with payment gateways & government databases

#### 3.3 System Features

- Real-time application tracking
  - Automated SMS/ email notifications
- 3.4 Non-functional Requirements
- High security for personal data
  - 99.99% Uptime
  - Fast response within 3 sec

### 4. Appendices

#### 4.1 Glossary

- PAS - passport Automation System

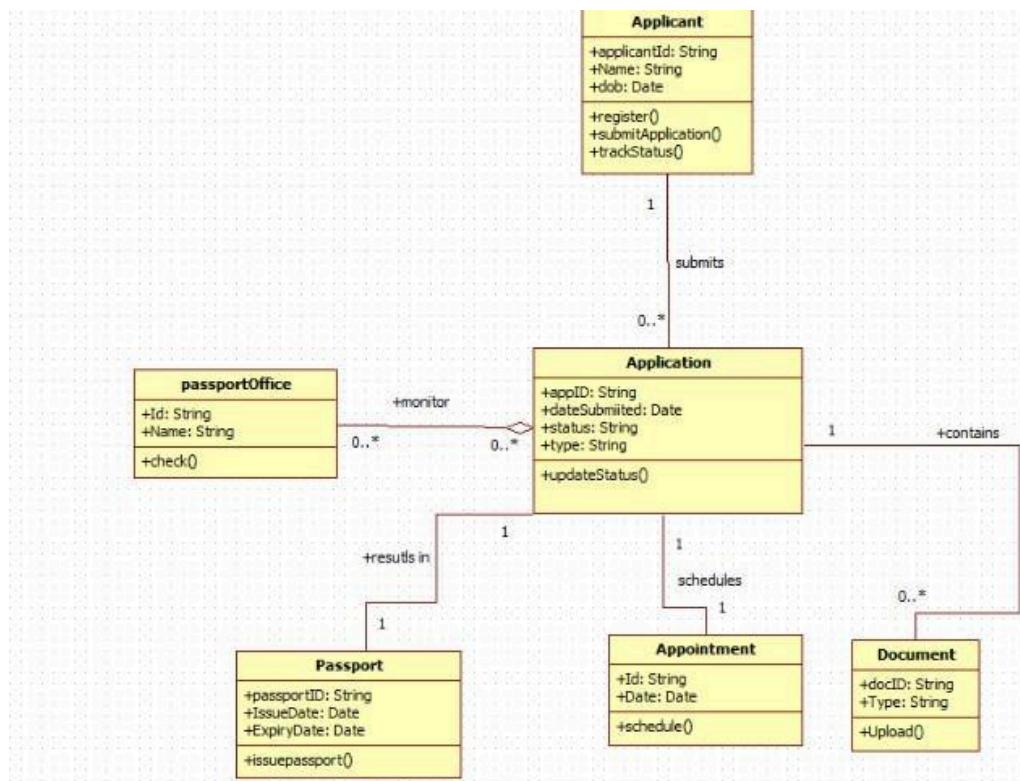
ID - Identification

#### 4.2 Future Enhancement

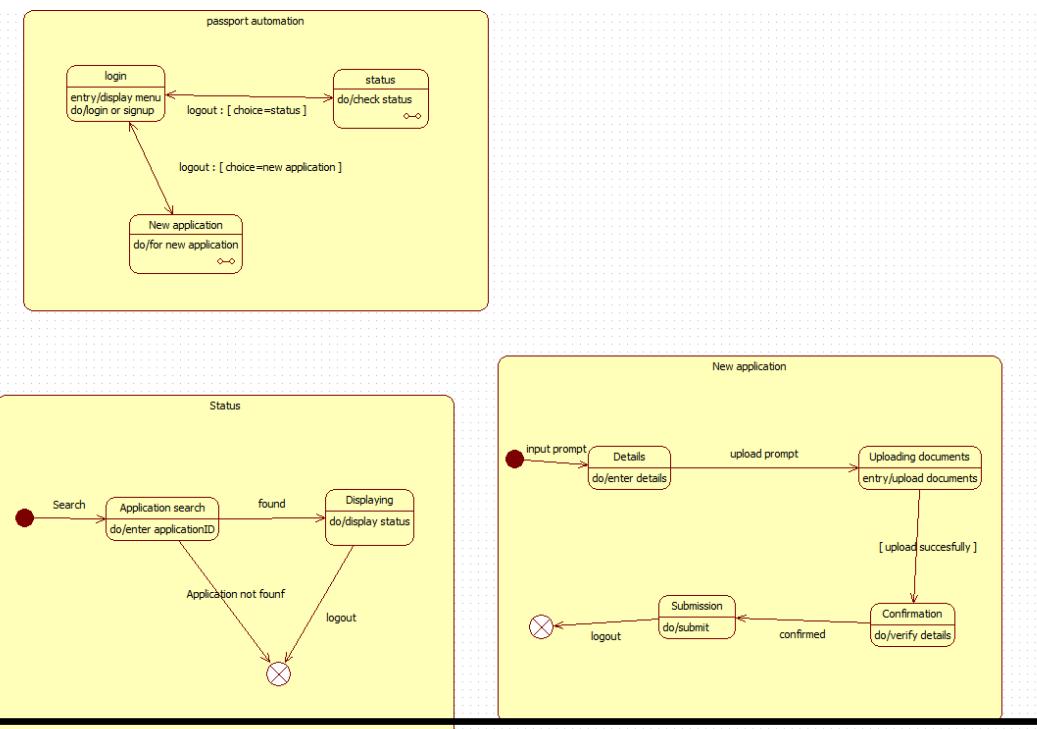
• Integration with Biometric Systems

& mobile application

## 5.2 Class Model

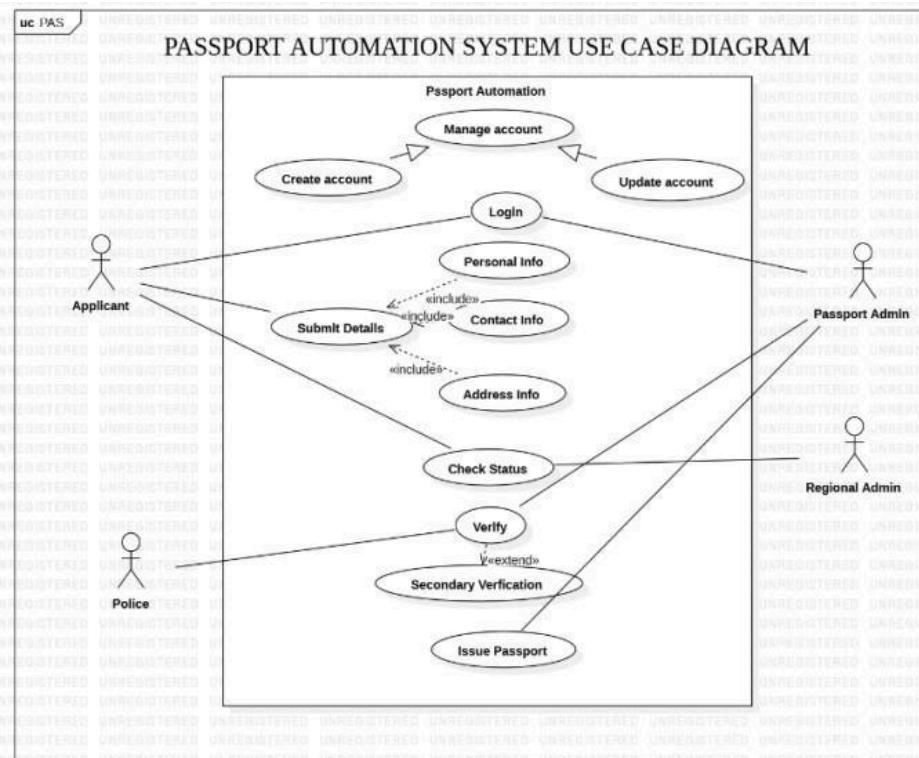


## 5.3 State Model

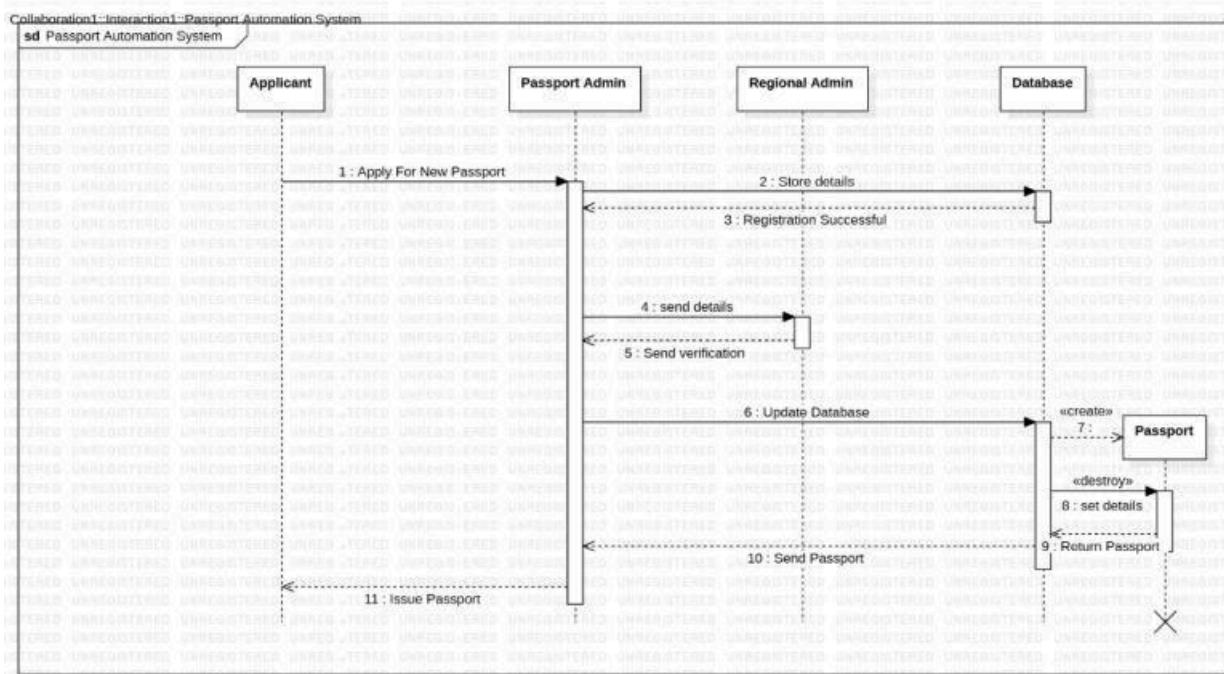


## 5.4 Interaction Models

### a. Use Case Model



### a. Sequence Model



## b. Activity model

