

23/09/2024

LAB-1: SOFTWARE REQUIREMENT SPECIFICATION

I. HOTEL MANAGEMENT SYSTEM.

1. INTRODUCTION:

1.1: Purpose of this document:

This document aims to highlight the requirements required to build a software for hotel Management system. The attributes required to create a software that facilitates a hotel's reservation management and administrative tasks.

1.2: Scope of this document:

This document encompasses the requirement specification for a hotel management system software. It aims to give a brief overview of the functionalities, requirements and its constraints. It outlines the estimated development time and cost.

1.3 Overview:

The Hotel Management System provides a software for storing hotel details facilitate transactions and manage room reservation. It also provides outlook to many functionalities such as house-keeping, front desk assistance, guest check-in/ check-out and more.

2. General description:

The system will enable hotel staff to efficiently manage reservations, automate check-in and check-out processes, track guest preferences, generate accurate billing and access comprehensive reporting tools.

3. Functional Requirements:

- i) Reservation management
- ii) Guest Management

- iii) Billing and Payment
- iv) Reporting and Analytics
- v) Customer Support
- vi) Check availability of rooms

4. Interface Requirements

4.1: User Interface

The user interface shall be web-based, and easy to navigate, ensuring a seamless user experience across different devices.

4.2: External System Interface

The system may need to interface with external systems such as:

- Payment gateways
- Property management system
- Channel management system

5. Performance Requirements:

- i) Response time
- ii) Availability
- iii) Capacity

Design

6. Design Constraints:

- i) Technology stack
- ii) Hardware limitations
- iii) Security compliance

7. Non-Functional Attributes:

- i) Security
- ii) Reliability
- iii) Usability
- iv) Maintainability

8. Preliminary Schedule and Budget

8.1: Schedule

The initial estimate of project duration is 3 months.

Detailed timeline will be finalized upon further analysis.

8.2: Budget

The estimated project budget is 1,00,000. A more detailed financial plan will be developed during the project planning phase.

req. analysis: 40,000

Design & implementation: 20,000

Maintenance & evolution: 40,000.

II. CREDIT CARD PROCESSING SYSTEM

1. INTRODUCTION:

1.1: Purpose of this document:

This document defines the requirements for a software system designed to process credit card transactions. It outlines the system's functionality, performance expectations, and design constraints, serving as a comprehensive guide for the development team.

1.2: Scope of this document

This document encompasses the functional and non-functional requirements for the Credit Card Processing System, including user interface specifications, transaction processing details, for interacting with external systems like payment gateways and banks.

1.3: Overview

The Credit Card Processing System will be a robust and secure software solution enabling business to accept credit card payments from customers. It will manage transaction authorizations while adhering to industry security standards.

2. General description:

The CCPS will be used by business to accept credit card payments online or in-store. The system will handle transaction authorizations, refunds, and chargebacks, ensuring security and compliance with PCI-DSS standards.

3. Functional Requirements:

i) User Authentication —

ii) Credit card authorization and validation

iii) Transaction Processing and Settlement

- iv) Refund processing]
- v) Fraud detection mechanism]
- vi) Reporting and Auditing]

4. Interface Requirements:

The system will integrate with Point of sale devices and e-commerce websites. The interface will communicate with banks and card networks for transaction approvals.

5. Performance Requirements:

The system must process thousands of transactions per second with minimal latency. Response time should be within milliseconds for transaction approval or rejection.

6. Design Constraints:

- Must comply with payment card industry data security standards. Integration with various card networks and banking systems is required.

7. Non-Functional Attributes:

- i) Security - Must use tokenization and encryption for all transactions
- ii) Portability - Should work across different platforms
- iii) Scalability - Must handle increased transaction volume during peak times.

8. Preliminary Schedule and Budget:

Development is estimated to take 9 months with 2,00,000 INR, considering integration with banks.

Requirement analysis & Design: 40,000

Design & Development / Implementation: 20,000

Maintenance & evolution: 40,000

4:2:4 rule

8th
30/9

LAB-2:

III. LIBRARY MANAGEMENT SYSTEM

1. INTRODUCTION:

1.1: Purpose of this document

This document specifies the requirements necessary for a software system design of a Library Management System which includes the initial information required to build the project. It defines the parameters to be considered to make the software more reliable, maintainable and efficient.

1.2: Scope of the document:

This document aims to identify the necessary functionalities, interface properties, project time and cost estimations. With this information the goals of the project are understood and analysed. It helps to get a clear description and requirement to be focused while implementing the project.

1.3: Overview

The Library management system will be a easy to access, store and manage. It will maintain user details confidential. It will provide a secure payment gateway for subscribing to journals, magazines etc. It enables user to access books online, book details and maintains records of books rented and returned.

2. General description:

The library management system will allow publisher to publish their books, research papers, documents, video lectures, journals, etc. The software will record user activity and maintain history of the recent reads or media accessed for easy navigation to ~~the~~ if the user wants to visit previous documents. It makes

managing records easier and checking availability of books become easier.

3. Functional Requirements:

- i) Publisher Authentication
- ii) Customer service
- iii) Book catalog with availability
- iv) Staff management
- v) Filters of Book categories
- vi) Categorizing Books, journals, audios, video lectures and magazines
- vii) Feedback and complaint system for customers,

4. Interface requirements.

The user interface should be easy to navigate (user friendly). Website with less response time which facilitates website availability to large number of people. It should display the book name, publisher name, publish date, book cover, author name, etc. It should be responsive, interactive and should be available for multiple devices. It should include filters and for easy navigation.

5. Performance Requirement.

- i) facilitate large customers at once (capacity)
- ii) Response time
- iii) Reliability (two crns)
- iv) Availability

6. Design constraints:

i) payment gateway to facilitate transactions, referrals etc.
It should include all payment methods.

7. Non functional Attributes:

- i) interactive user interface
- ii) compatibility
- iii) privacy and confidentiality

iv) High Performance and reliability

v) Scalability

vi) Security.

5. Preliminary Schedule and Budget.

The estimated time for development is 5 months with
1,50,000 INR required for implementation. If the project
requirement analysis : 40,000

design & development : 20,000

specification, evaluation, maintenance and evolution : 40,000

IV STOCK MAINTENANCE SYSTEM

1. INTRODUCTION:

1.1 Purpose of the document:

This document outlines the requirements including objectives, functionalities and performance expectations for the stakeholders.

1.2. Scope of the document:

This document covers essential features such as inventory tracking, stock management and reporting aimed at improving operational efficiency and minimizing losses. Development cost and timeline estimates are included as well.

1.3 Overview:

The stock maintenance system automates stock management processes providing an intuitive interface for tracking inventory, managing orders and generating reports.

2. General descriptions: Interface requirement

Stock maintenance system mainly focuses on the following requirements.

1. User Interface - Web based user friendly dashboard
2. Database - Connect to relational database
3. API - API for integration with accounting and ecommerce platforms.

3. General descriptions:

It has real time monitoring of stock levels which can be added, updated and deleted. It can process purchase orders and task deliveries. It also has capabilities to generate reports on stock & orders. It can notify users of low stock and expiration date.

4. Functional requirements:

- User can / should be able to manage stock items.
- providing real time stock tracking.
- purchase order and management
- Generate alerts for low stock and expirations.
- Report generating in inventory turn over.

5. Performance Requirements:

- 5.1. Response Time - queries within less response time.
- 5.2. Concurrent users - support a large scale of users simultaneously.
- 5.3. Data storage - must simultaneously manage large data sets.

6. Design constraints:

- Develop using specified technologies (Java, python)
- Ensure compatibility with existing inventory hardware.
- Comply with data protection regulations.

7. Non-Functional attributes:

- Security: implementing user authentication.
- Reliability: Ensure 99.9% uptime.
- Scalability: Accommodate future growth.
- Usability: Intuitive interface, for minimal training.

8. Preliminary Schedule and Budget:

- Project Duration - estimated at 4 months
- 1.5 months - requirement analysis and design
- 1 month - implementation and validation
- 1.5 months - evaluation and management

Budget approximation: £90,000

Requirement: £35,000

Implementation and verification - £10,000

II PASSPORT AUTOMATION SYSTEM

1. INTRODUCTION.

1.1 Purpose of the document:

This document outlines the requirements for the PAs, detailing objectives, functionalities and performance expectations for all stakeholders.

1.2 Scope:

It covers features like application submission, document verification, appointment scheduling, and reporting, aimed at enhancing operational efficiency and user experience.

1.3 Overview:

The PAs Streamlined passport application and issuance processes, providing an intuitive interface for applicants to submit applications and track their status.

2. General description:

This application enables users to submit applications for passport and facilitates the user verification and authentication by submission of documents. It allows scheduling of appointments for interviews, provides smooth payment transactions and tracking status of the application process.

3. Functional Requirements:

- Allow online application creation and submission.
- Enable document verification against official databases.
- Allow appointment scheduling for interviews.
- Provide real-time application status tracking.
- Process payments securely and provide confirmation.

4. Interface Requirements:

- User Interface: web-based and user friendly across devices

Database Integration: Secure relational database for application storage

API: Provide APIs for integration with identity verification services.

5. Performance Requirements:

Response Time:
S.1 Provide low response time

S.2 concurrent users: Support up to 500 users simultaneously

S.3 Data storage: store large data sets

6. Design Constraints:

- Comply with data protection and privacy regulations
- Ensure compatibility with existing government verification systems
- Developed using specified technologies (Java, Python)

7. Non-Functional Attributes:

- Security: Implement robust security measures and data encryption
- Reliability: Implement reliable system
- Scalability: Accommodate large number of users
- Usability: requires minimal training

8. Project Schedule and Budget:

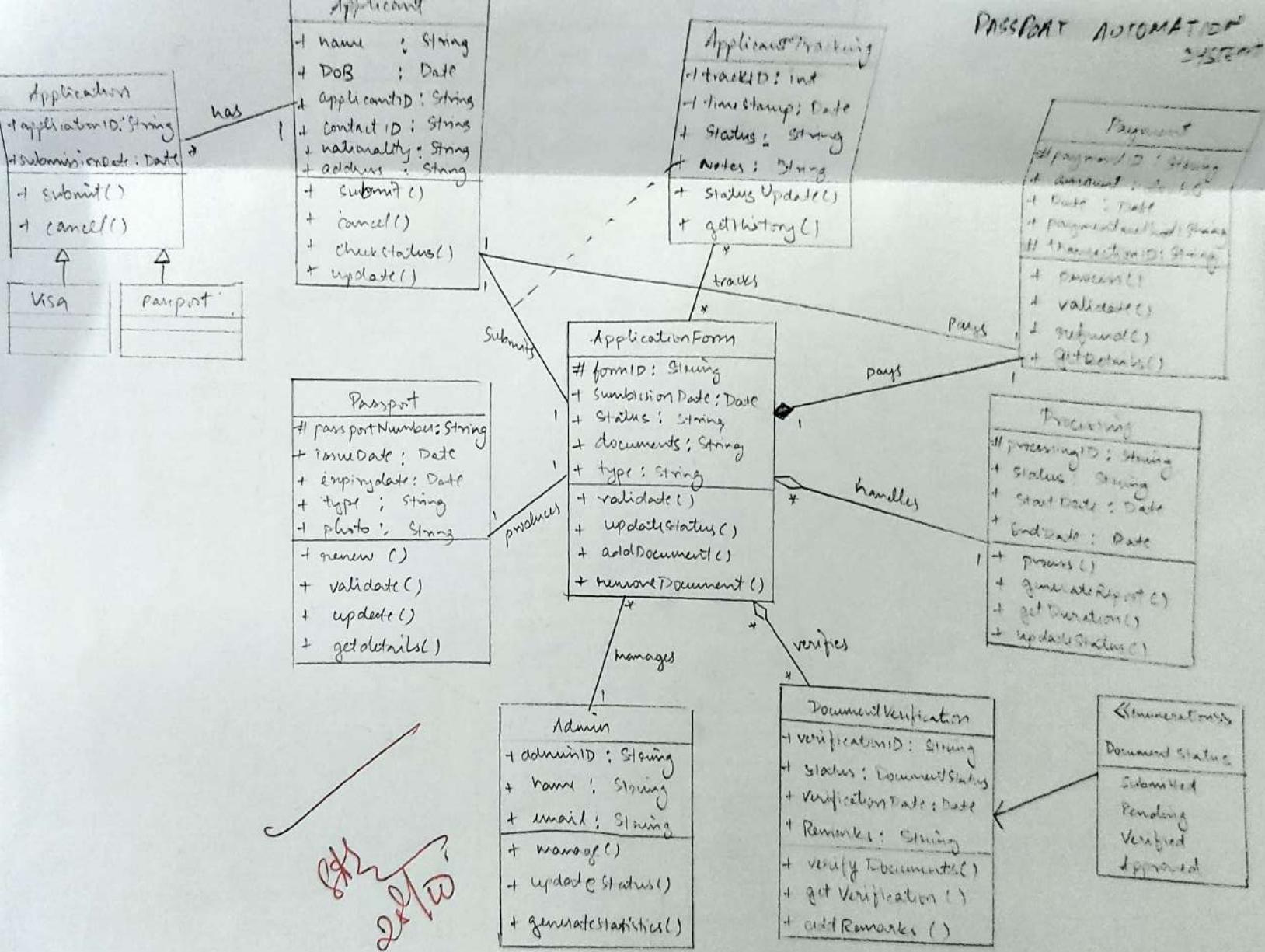
- Project Duration - estimated 6 months.

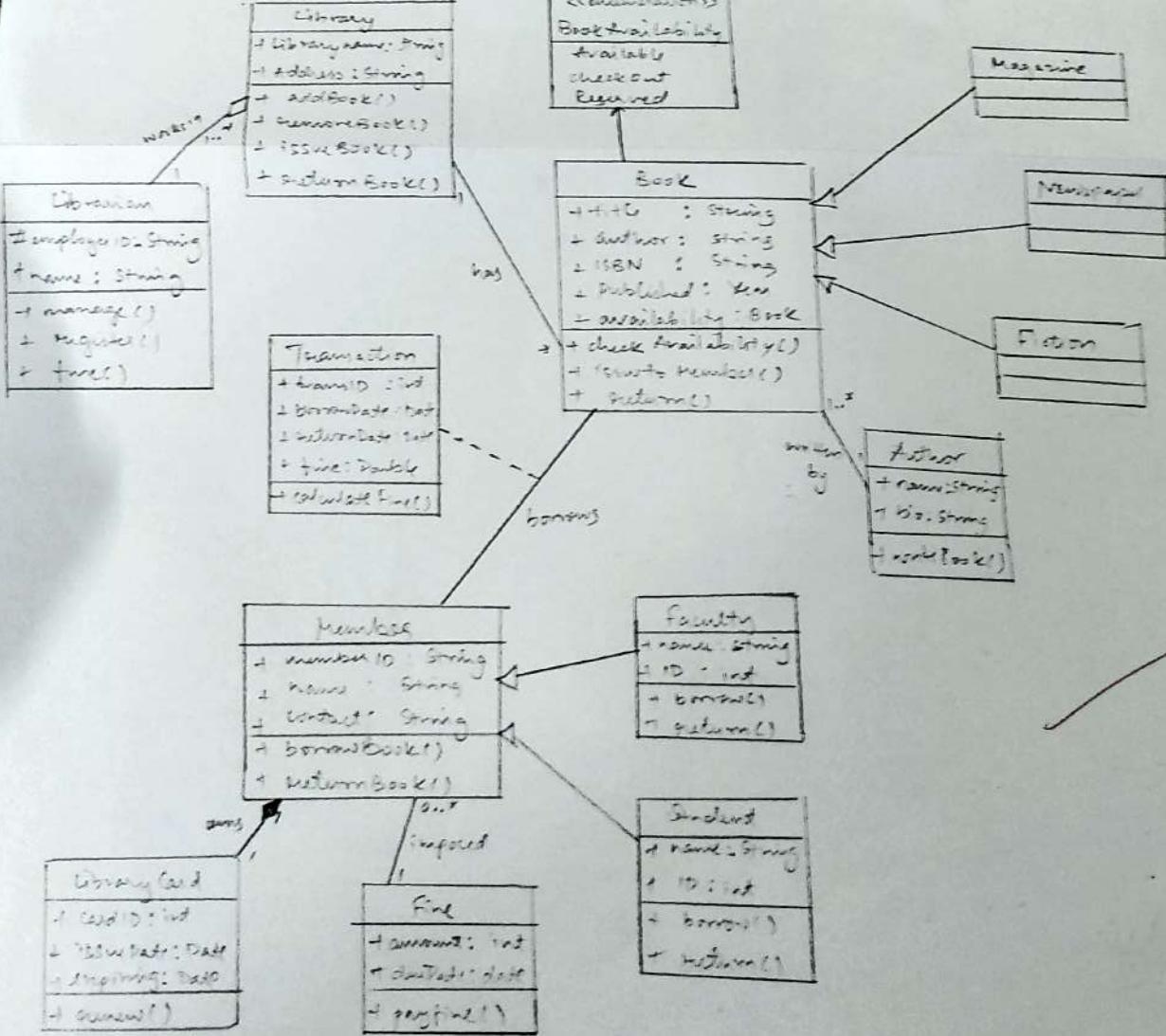
- Budget approximation - ₹ 1,00,000.

~~Other Requirement~~ - ₹ 40,000

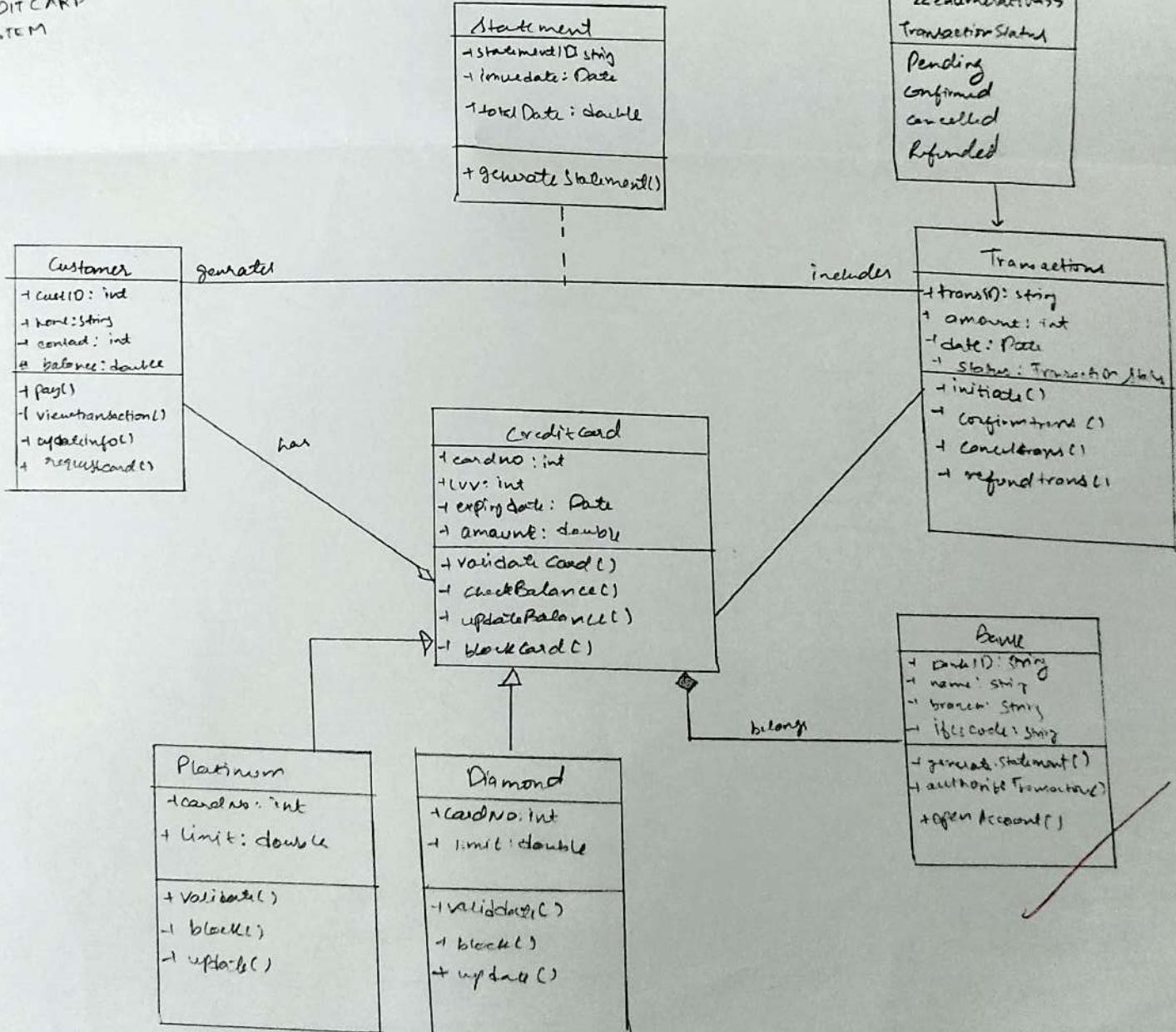
~~Design and Implementation~~ - ₹ 20,000

PASSPORT AUTOMATION SYSTEM

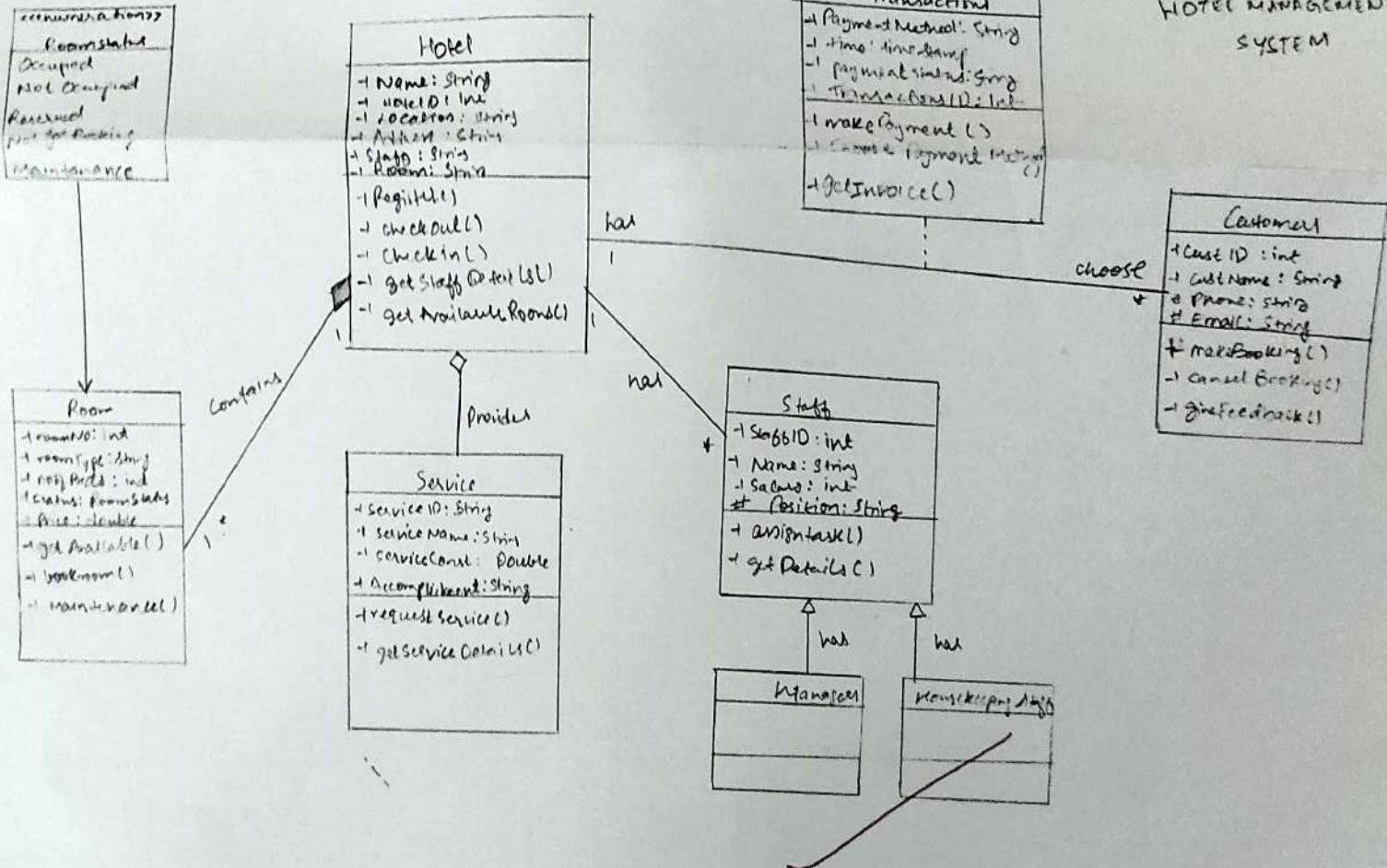




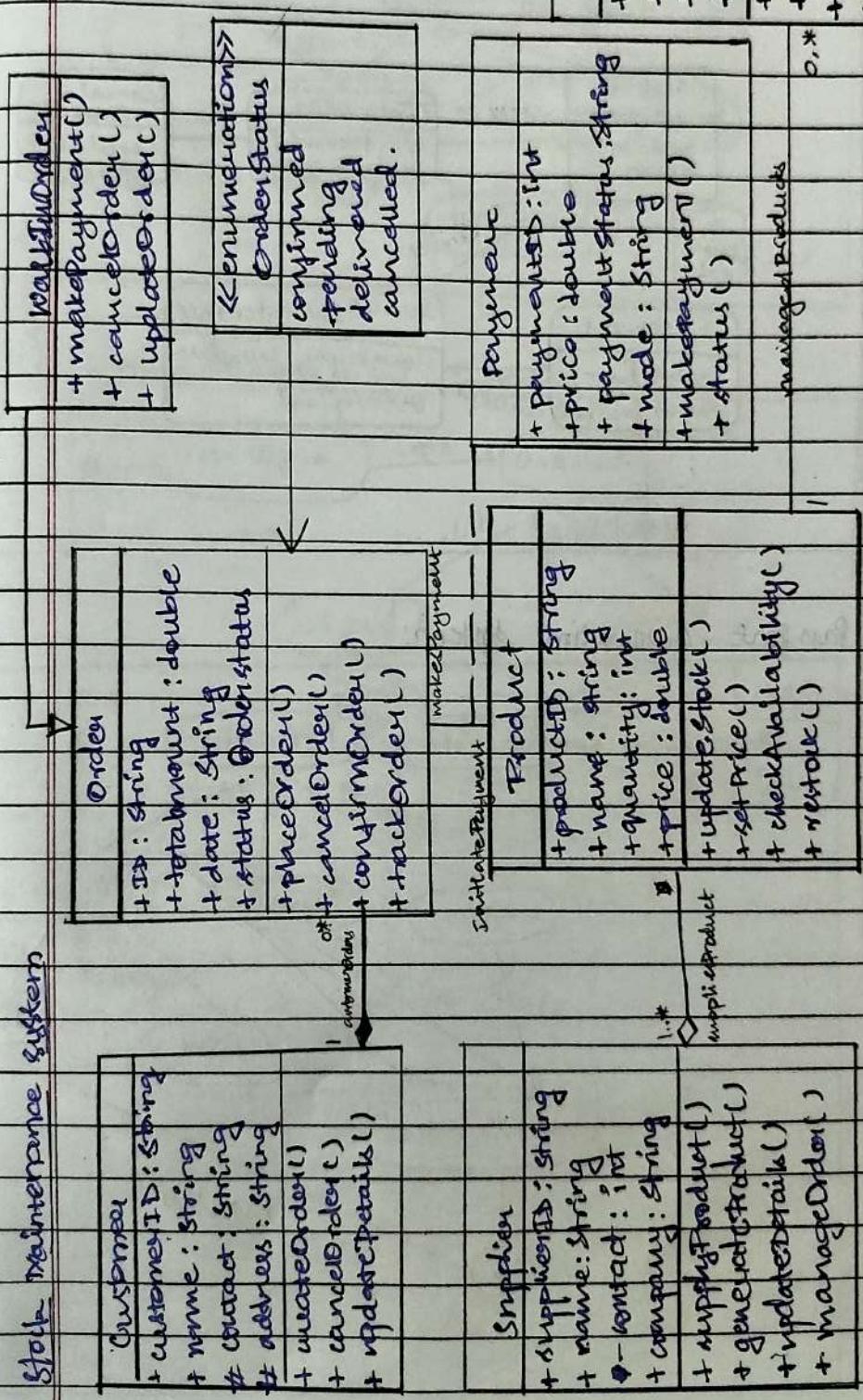
CREDIT CARD SYSTEM



HOTEL MANAGEMENT SYSTEM



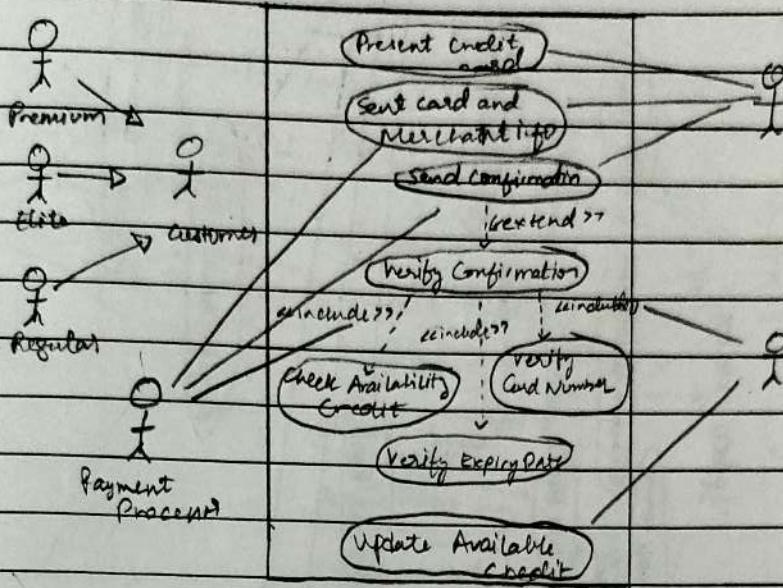
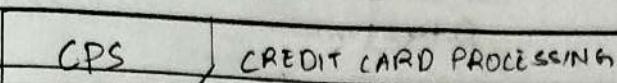
① Store Management System



Page No. _____
Date _____

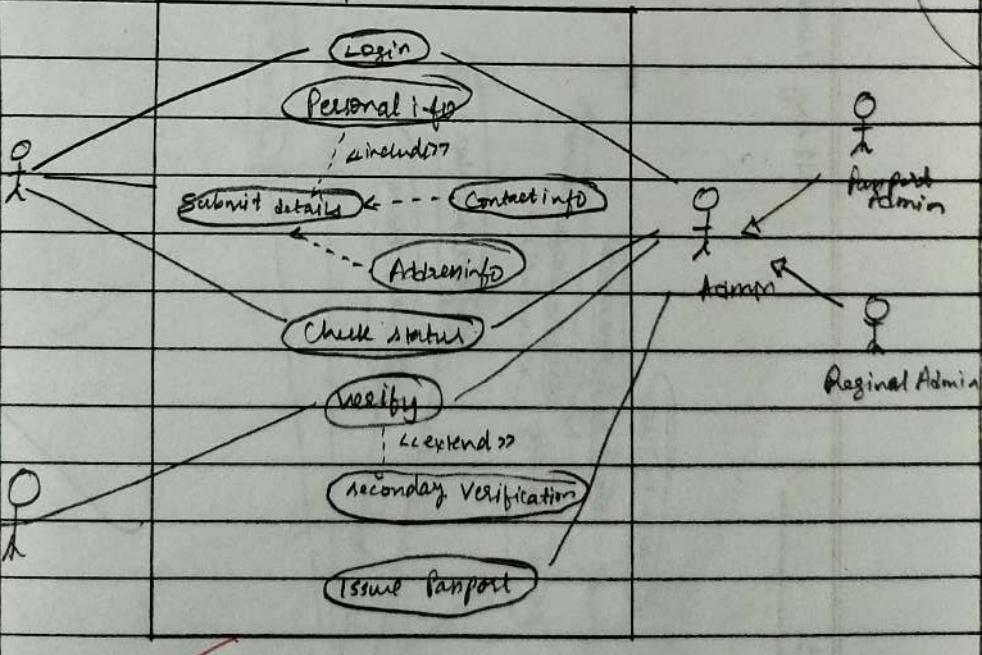
CREDIT CARD PROCESSING

USE CASE



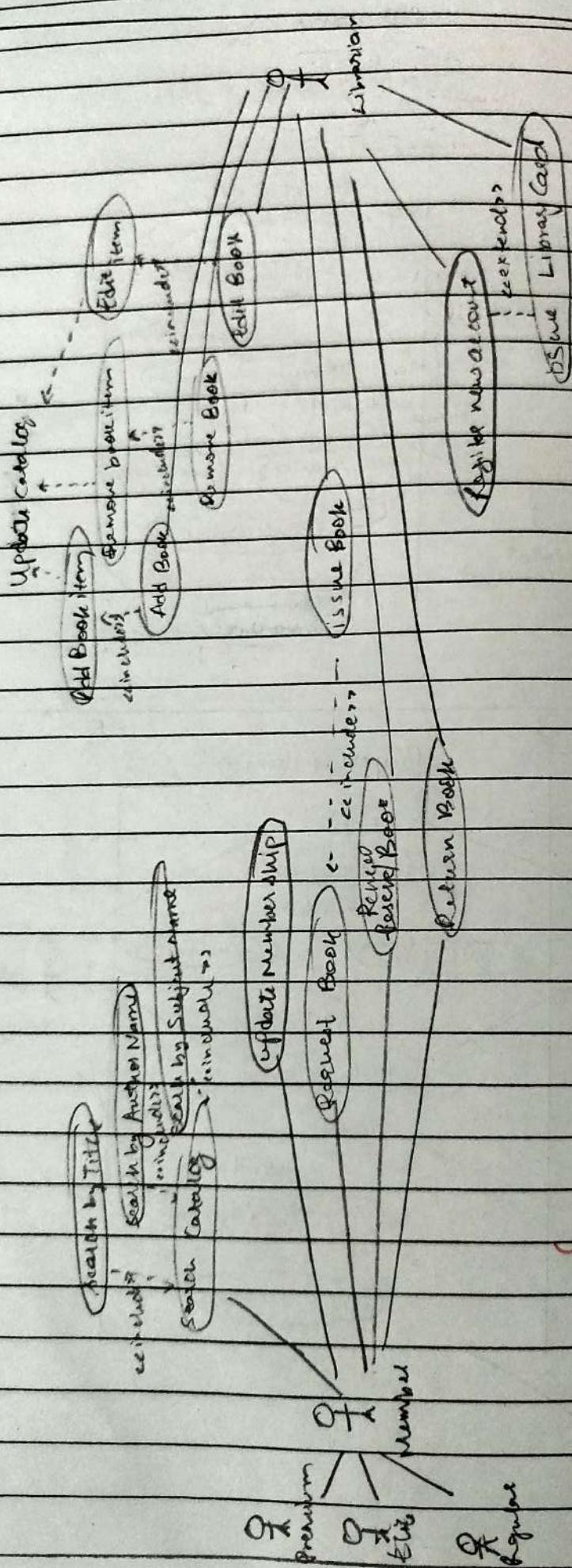
PAS

Passport Automation



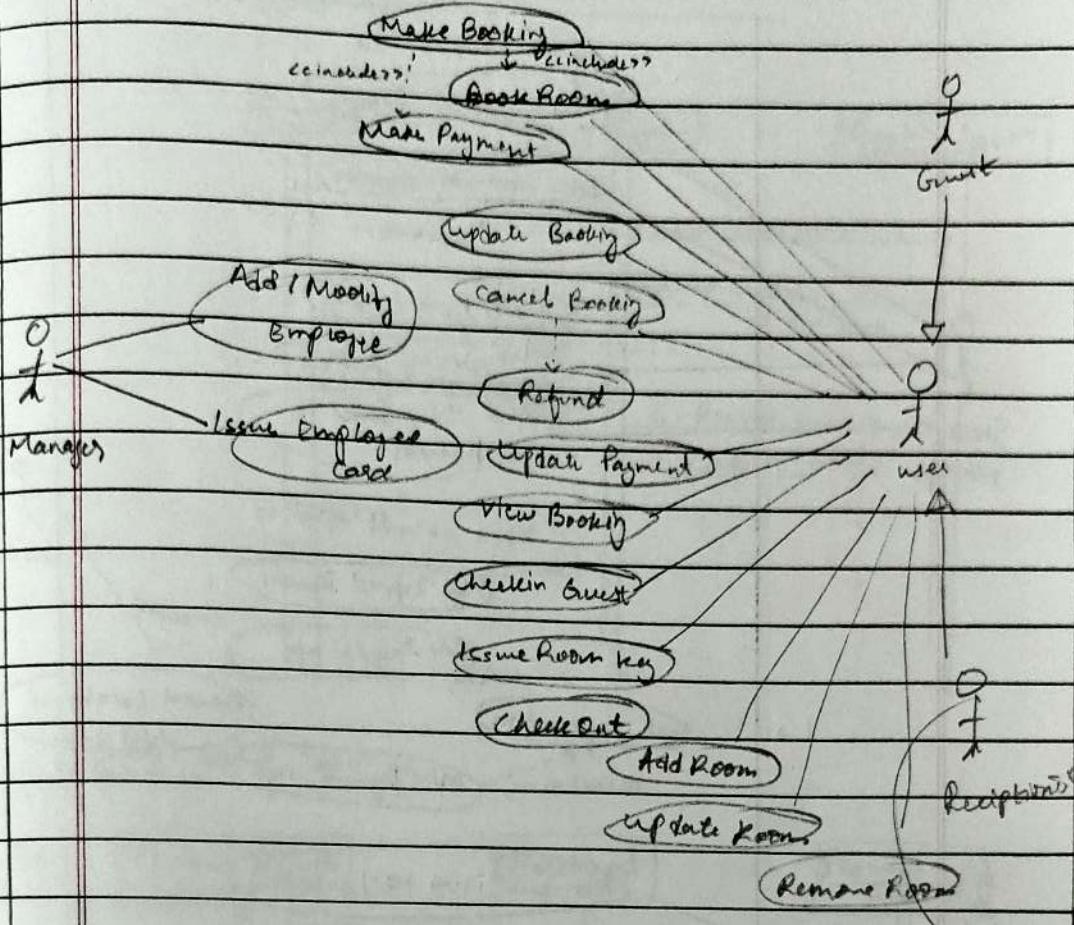
LMS

Lifelong Management

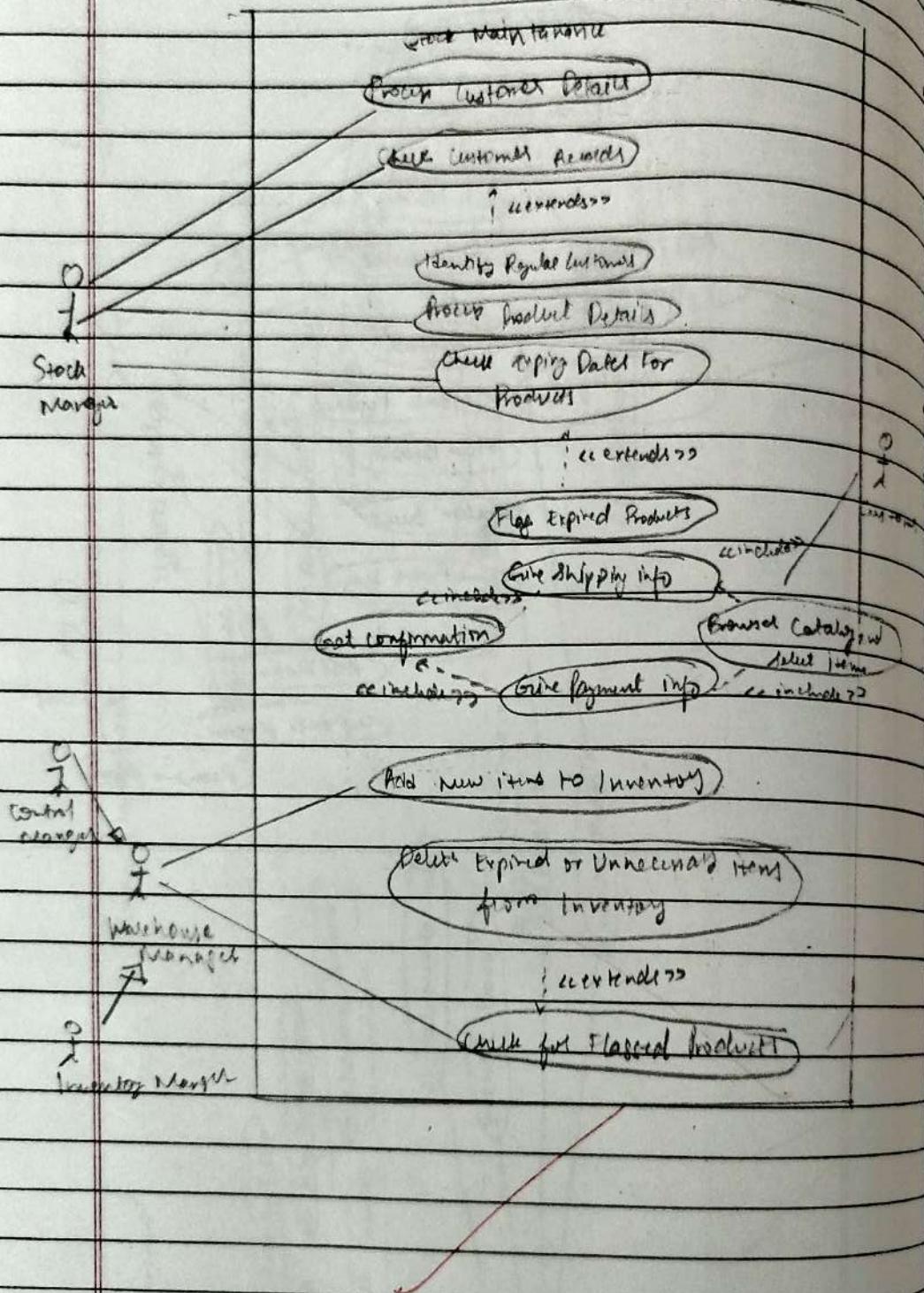


HMS

Hotel Management

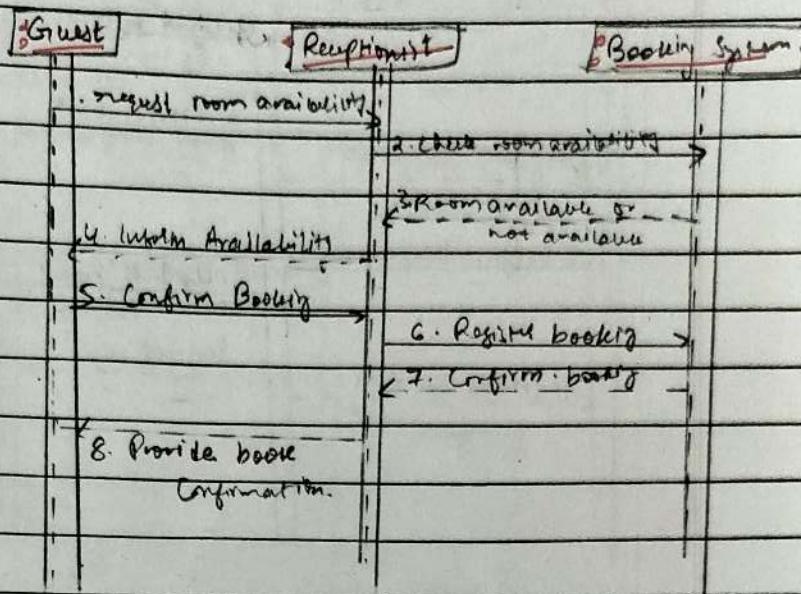


STOCK MAINTENANCE SYSTEM

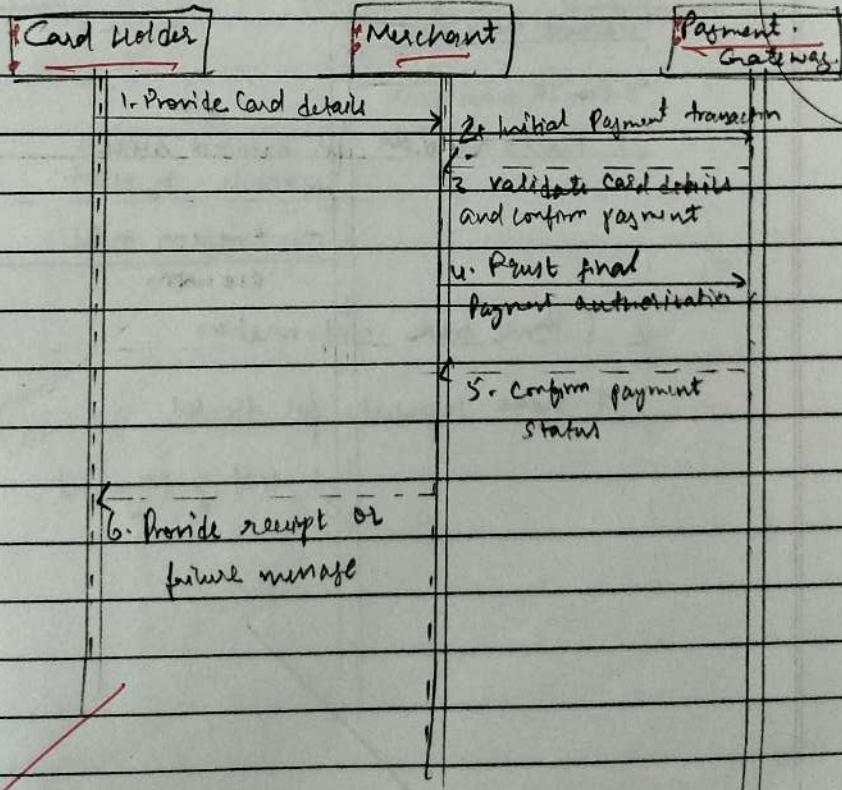


Sequence diagram

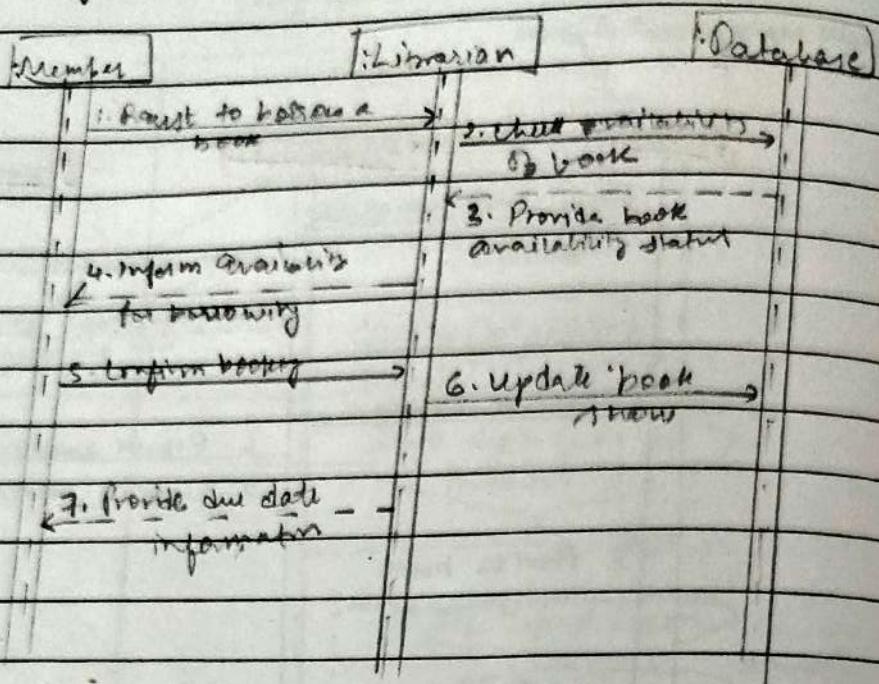
Hotel Management System



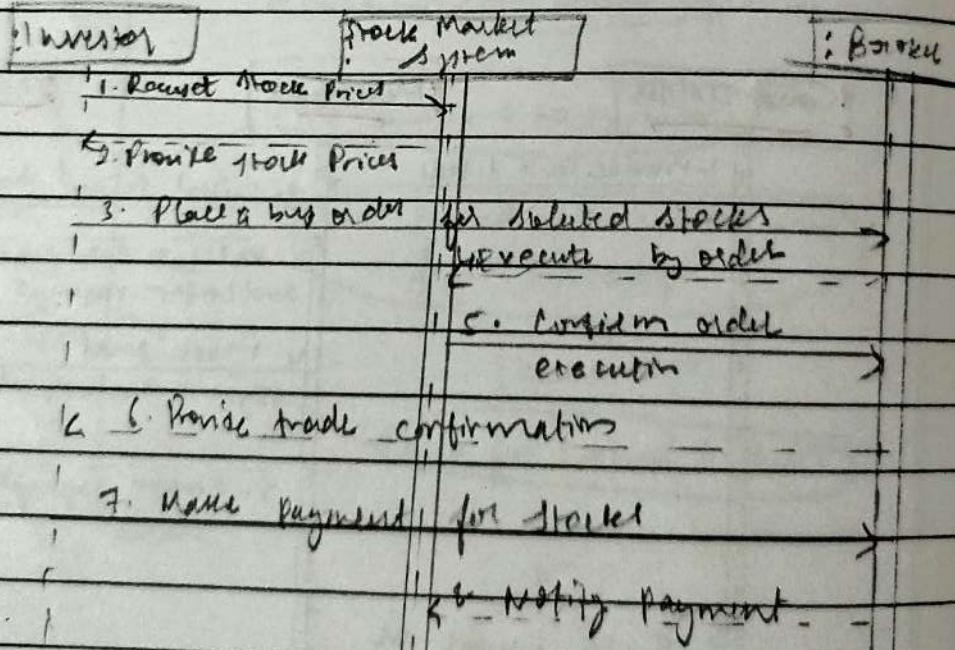
Credit Card Payment



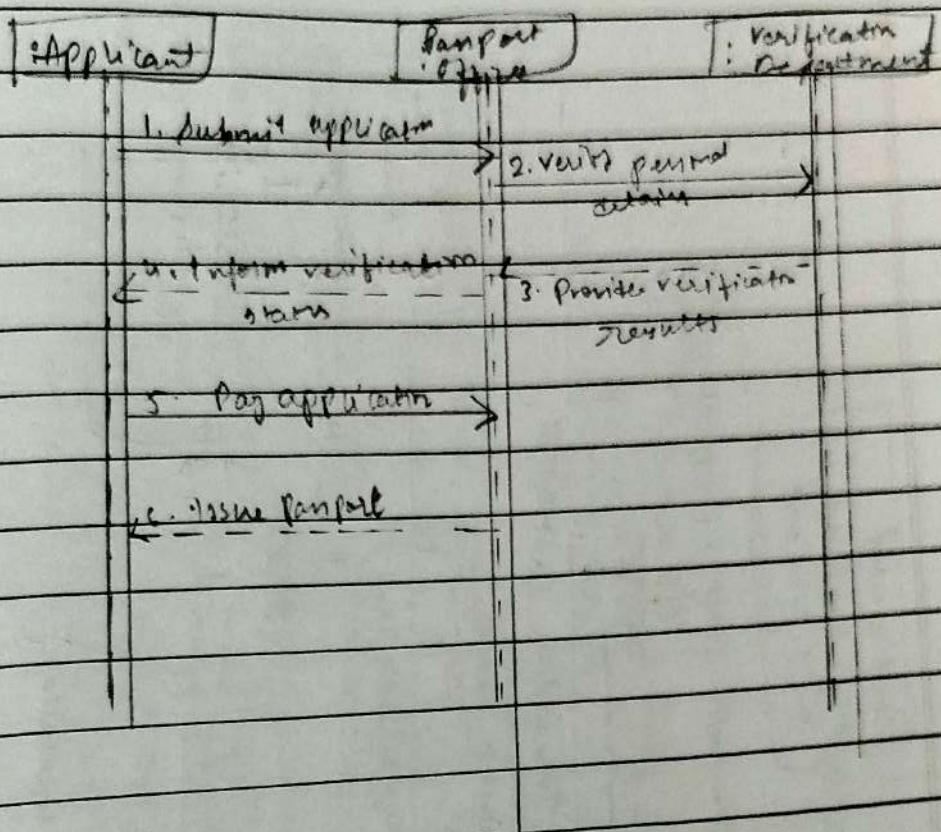
3. Library Management System



4. Stock Maintenance System

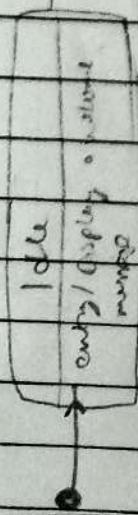


5. Passport Automation System.



active
passive
present

Am Credit Card Payment Alert



Card Inserted

Card Number / Pin / Card Type / Card Holder Name / Expiry Date / CVV / PIN / Card Type / Card Holder Name / Expiry Date / CVV / PIN

Transaction Initiated : [Transaction Code]
Card Number / Expiry Date / CVV / PIN

Transaction Initiated
card issued transaction amount
exit / prepare transaction result
processing payment
(return)
entry connect to payment gateway

Transaction Completed

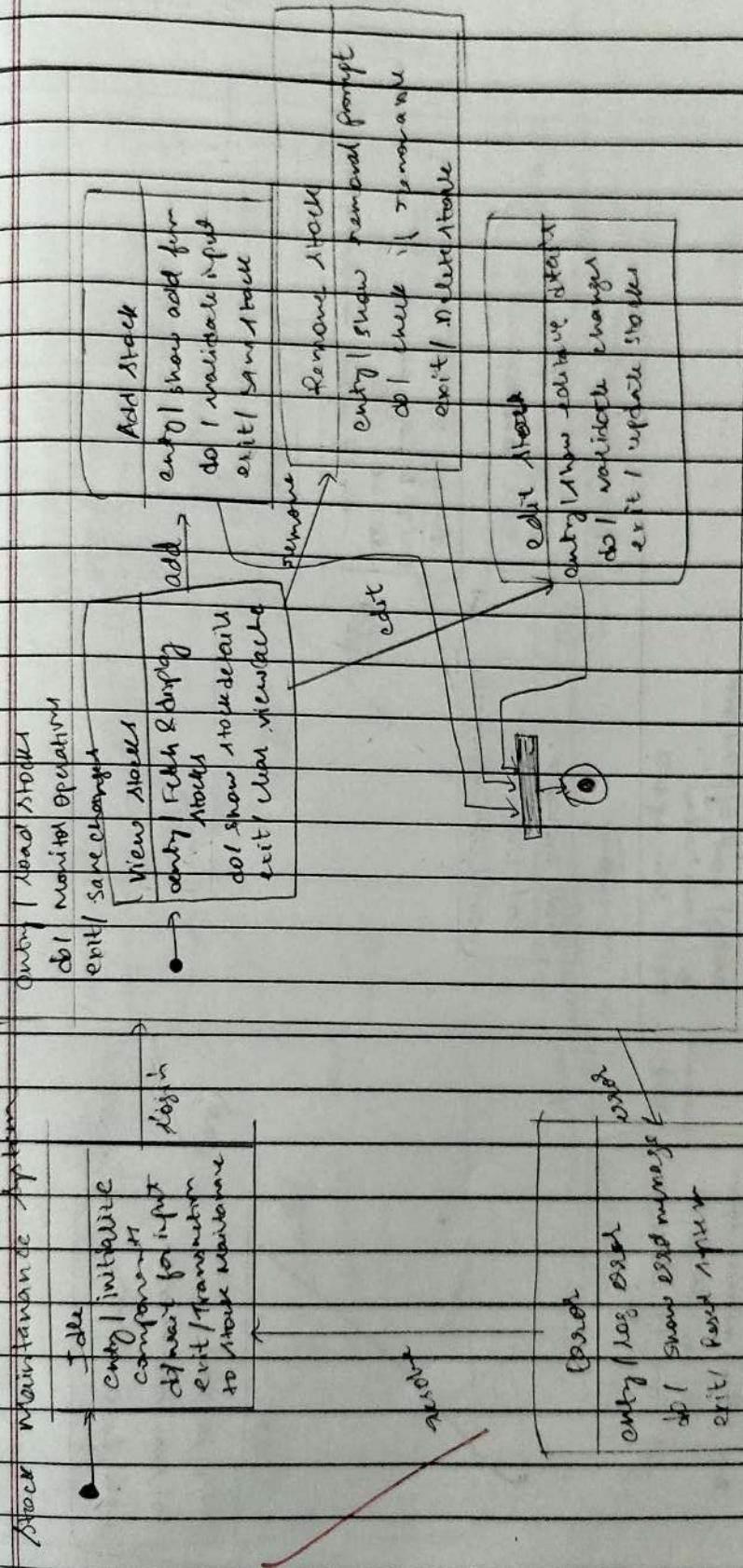
entry / point transaction
exit / last option for
next transaction

Process transaction
Transaction Declined
entry / Display card message
exit / log declined transaction

Waiting for user input
from

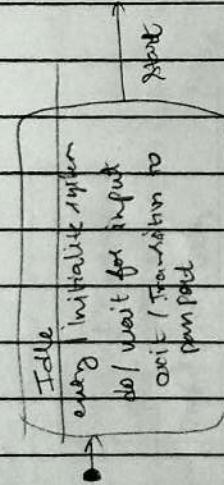
Page No. _____
Date _____

Hoelle *montana*

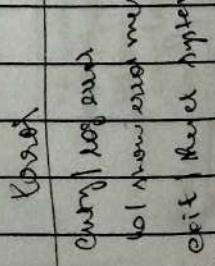
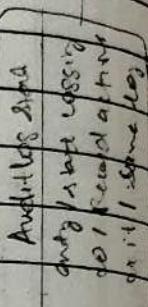
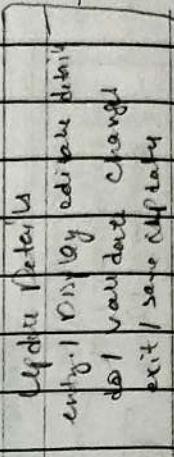
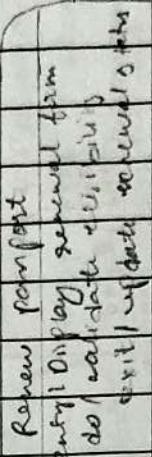
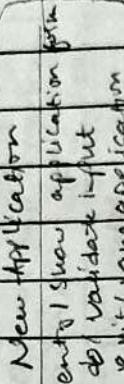
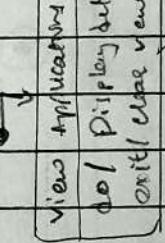


Dam Port Automation System

Porter Screening



entry / load application
do / prompt request
exit / save update

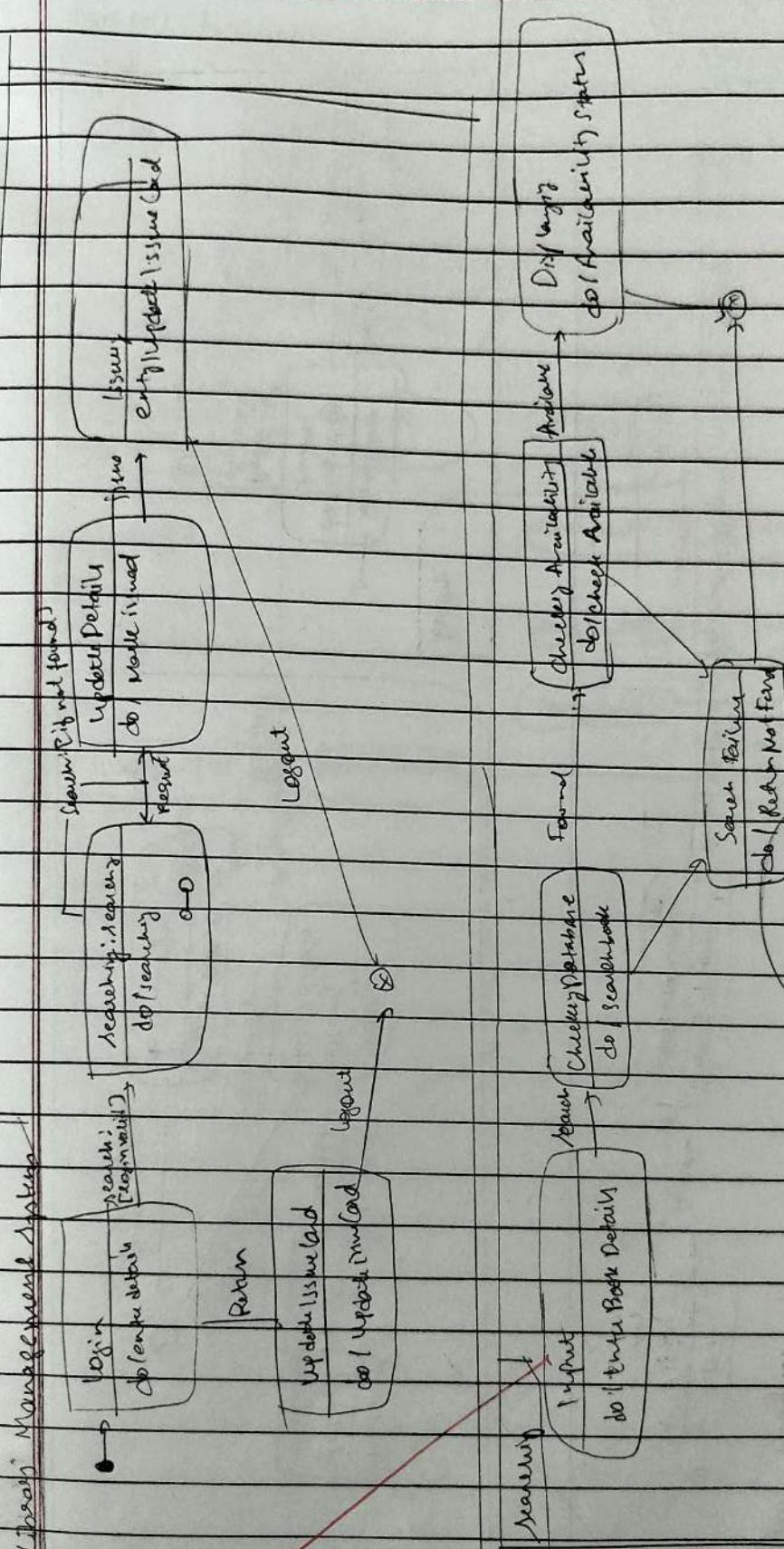


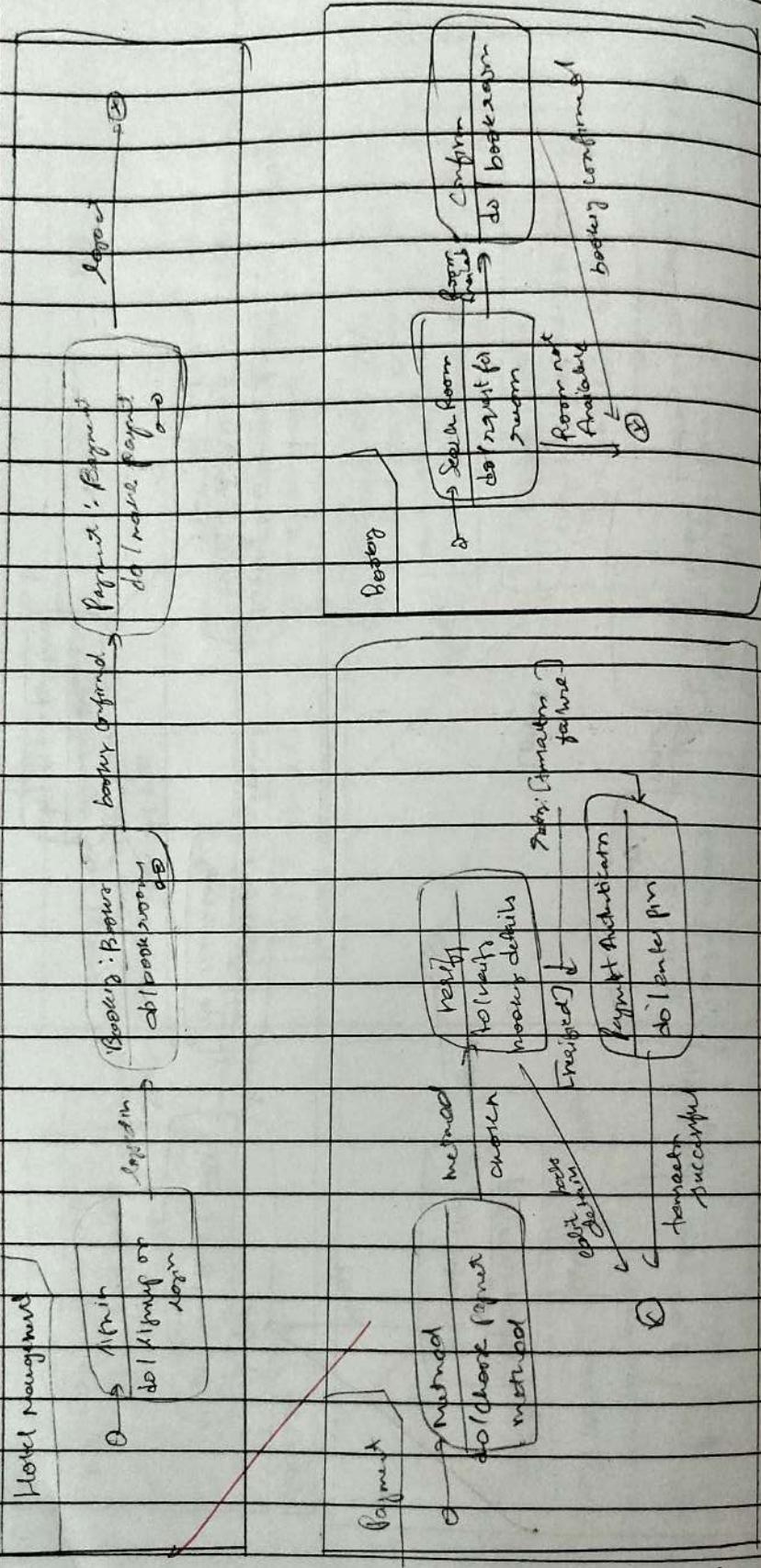
Logout

entry / log out

do / show menu

exit / back system

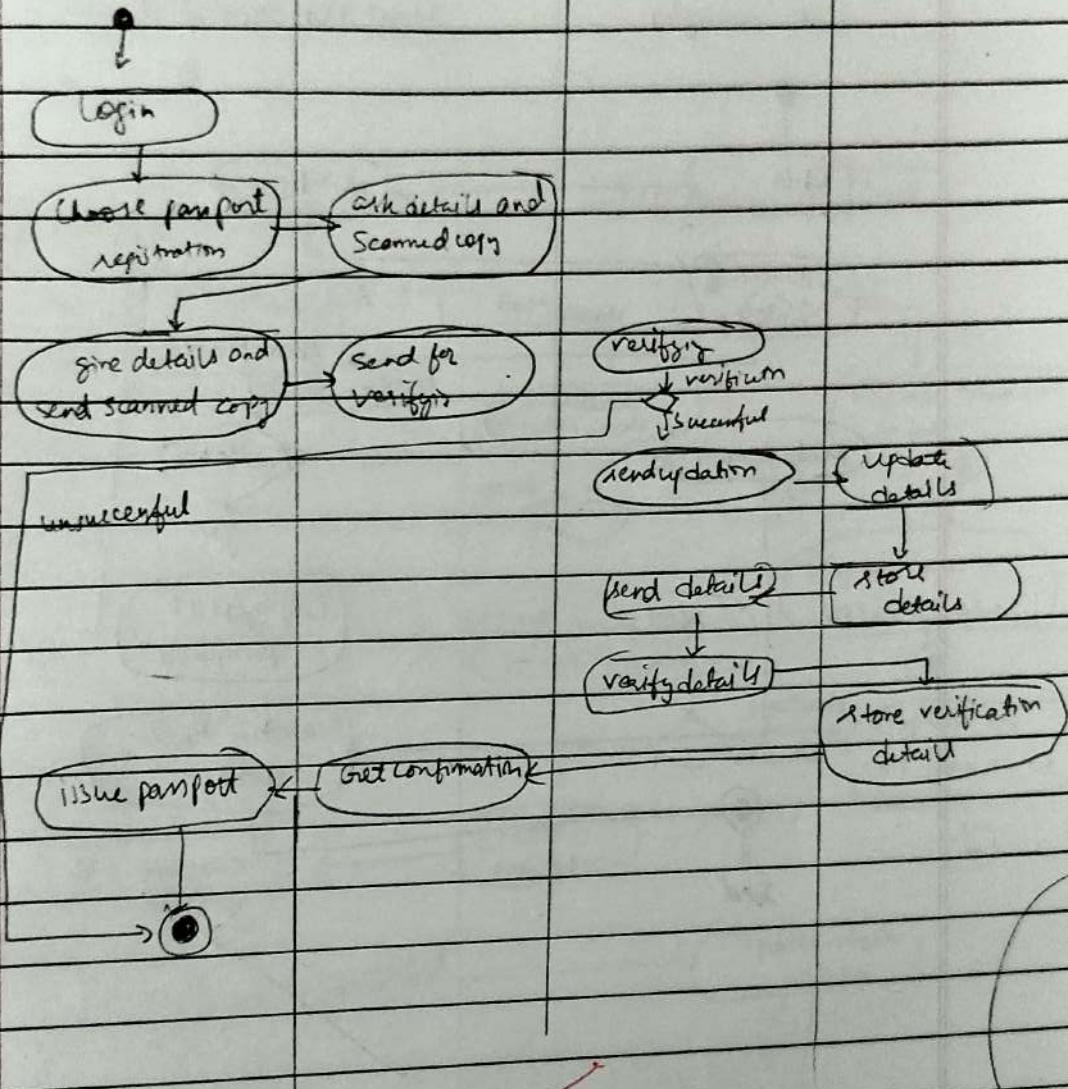




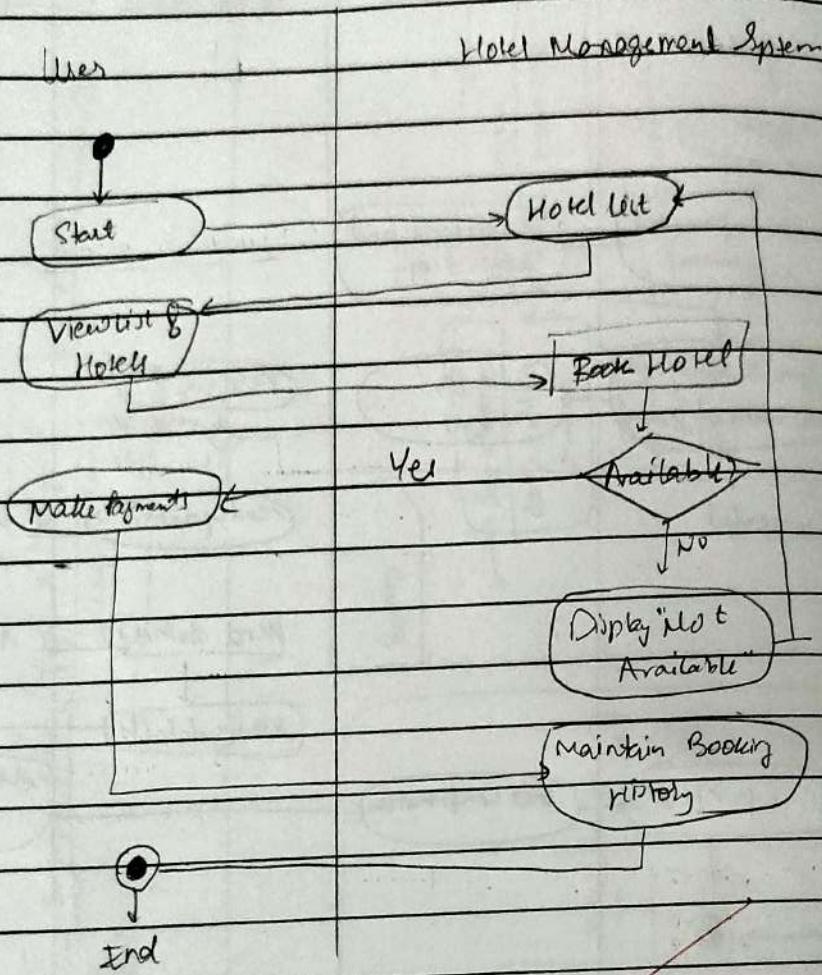
Advanced Activity Diagram

Passport Automation System

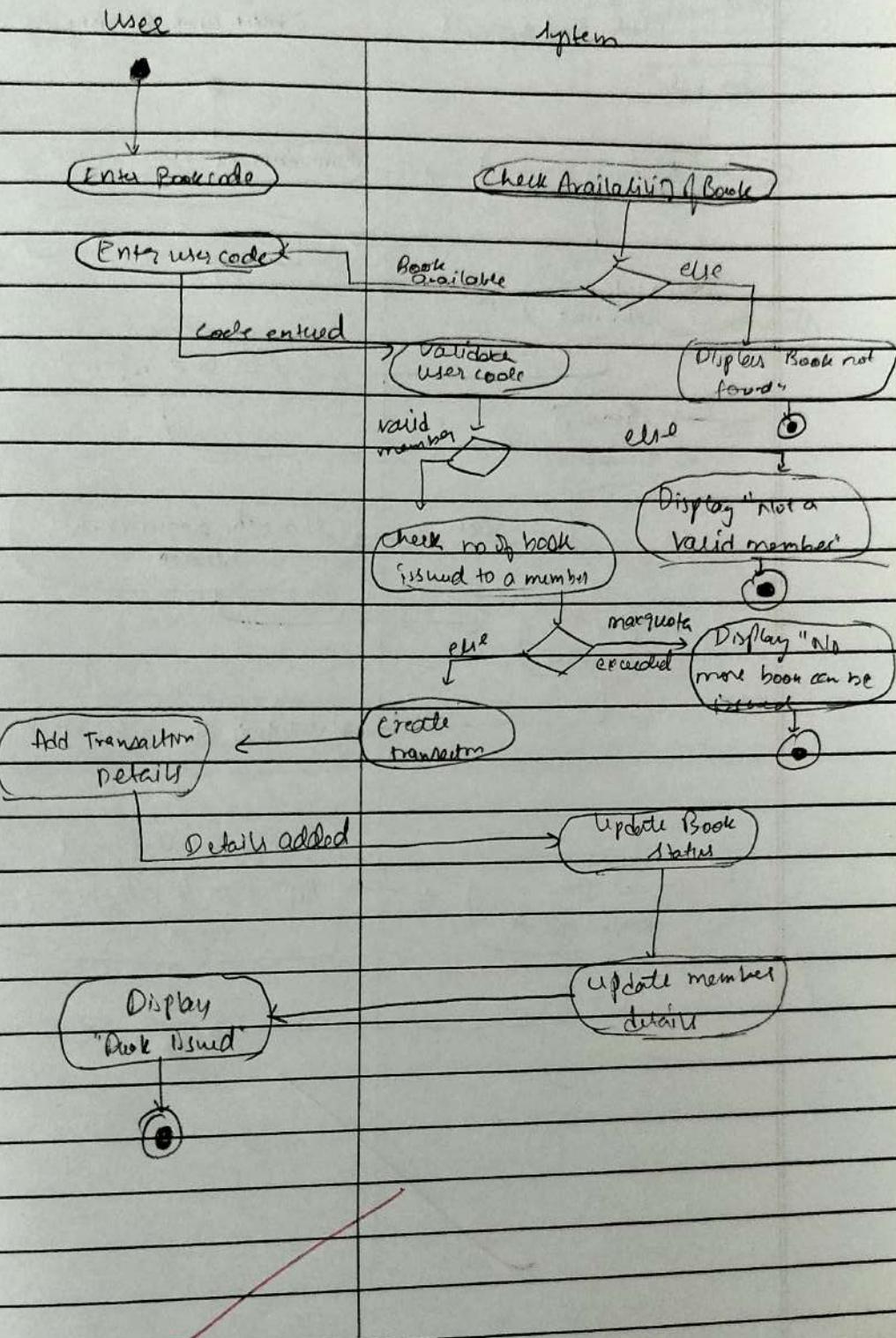
1. Applicant Passport admin Regional admin Database



Hotel Management System



Library Management System



Credit card Processing System

