B.M.S. COLLEGE OF ENGINEERING BENGALURU

Autonomous Institute, Affiliated to VTU

Lab Record

Software Engineering and Object-Oriented Modeling

Submitted in partial fulfillment for the 6th Semester Laboratory

Bachelor of Engineering
in
Computer Science and Engineering

Submitted by:

Prannav D

1BM21CS046

Department of Computer Science and Engineering
B.M.S. College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
Mar-June 2024

B.M.S. COLLEGE OF ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Object-Oriented Analysis and Design(22CS6PCSEO) laboratory has been carried out by Prannav D(1BM21CS046) during the 6th Semester Mar-June-2024.

Signature of the Faculty Incharge:

Surabhi A (Assistant Professor) Department of

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

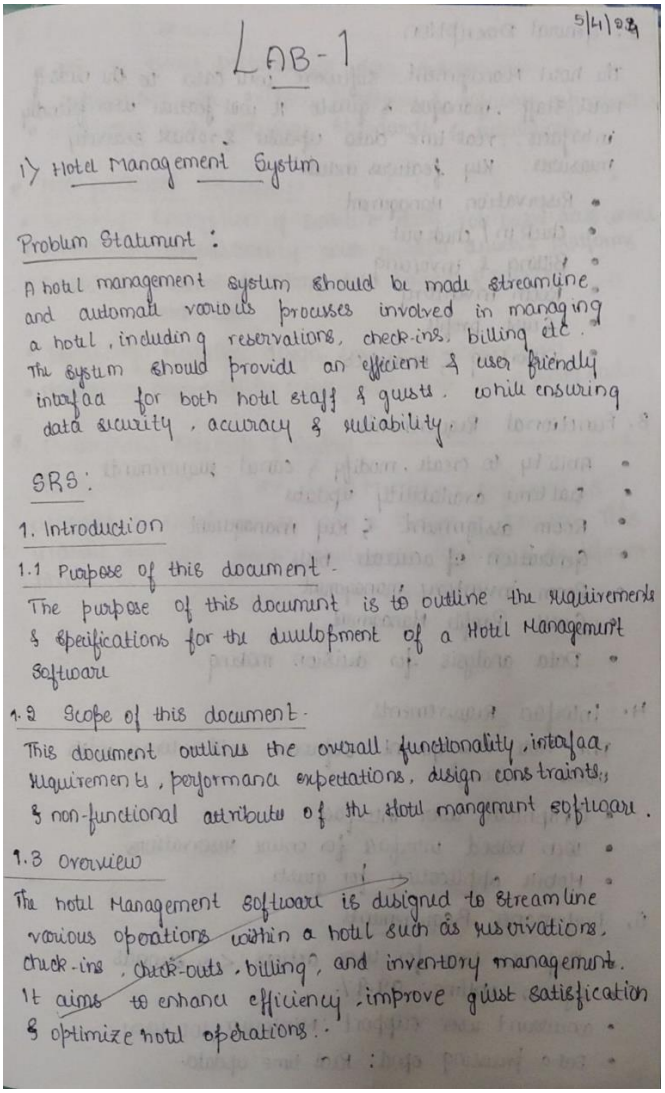
Contents

1. Hotel Management System

1.1 Problem Statement

A hotel management system should be designed to streamline and automate various processes involved in managing a hotel, including reservations, check-ins, billing, etc. The system should provide an efficient and user-friendly interface for both hotel staff and guests while ensuring data security, accuracy, and reliability.

1.2SRS-Software Requirements Specification



LAB-1

5/4/24

1) Hotel Management System

Problem Statement :

A hotel management system should be made streamline and automate various processes involved in managing a hotel, including reservations, check-ins, billing etc. The system should provide an efficient & user friendly interface for both hotel staff & guests. while ensuring data security, accuracy & reliability.

SRS :

1. Introduction

1.1 Purpose of this document :

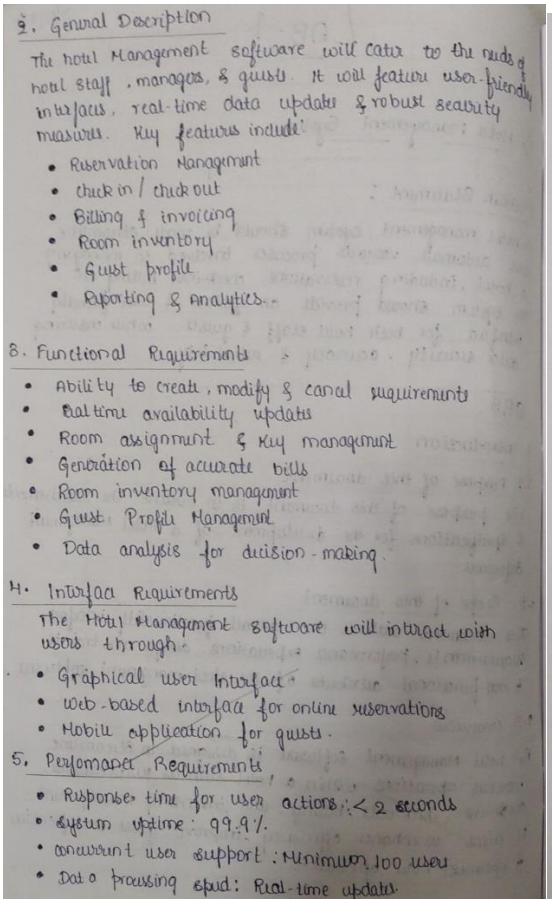
The purpose of this document is to outline the requirements & specifications for the development of a Hotel Management Software

1.2 Scope of this document :

This document outlines the overall functionality, interface, requirements, performance expectations, design constraints, & non-functional attributes of the hotel mangement software.

1.3 Overview

The hotel Management software is designed to streamline various operations within a hotel such as reservations, check-ins, check-outs, billing, and inventory management. It aims to enhance efficiency, improve guest satisfication & optimize hotel operations.



4. General Description

The hotel Management software will cater to the needs of hotel staff, managers, & guests. It will feature user-friendly interfaces, real-time data updates & robust security measures. Key features include:

- Reservation Management
- check in / check out
- Billing & invoicing
- Room inventory
- Guest profile
- Reporting & Analytics.

8. Functional Requirements

- Ability to create, modify & cancel requirements
- Real time availability updates
- Room assignment & Key management
- Generation of accurate bills
- Room inventory management
- Guest Profile Management
- Data analysis for decision-making.

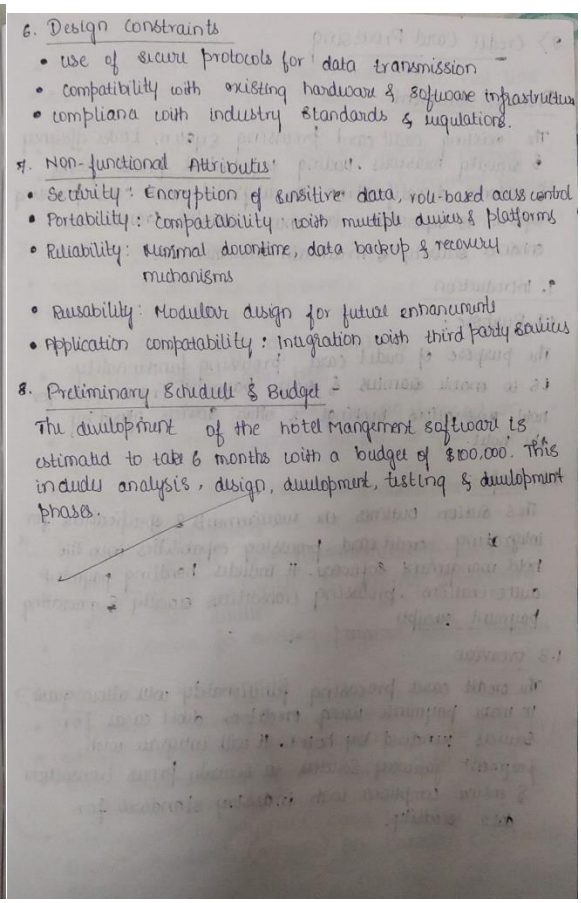
4. Interface Requirements

The Hotel Management software will interact with users through:

- Graphical user Interface
- Web-based interface for online reservations
- Mobile application for guests.

5. Performance Requirements

- Response time for user actions: < 2 seconds
- System uptime: 99.9%
- concurrent user support: Minimum 100 users
- Data processing speed: Real-time updates.



6. Design Constraints

- use of secure protocols for data transmission
- Compatibility with existing hardware & software infrastructure
- compliance with industry standards & regulations.

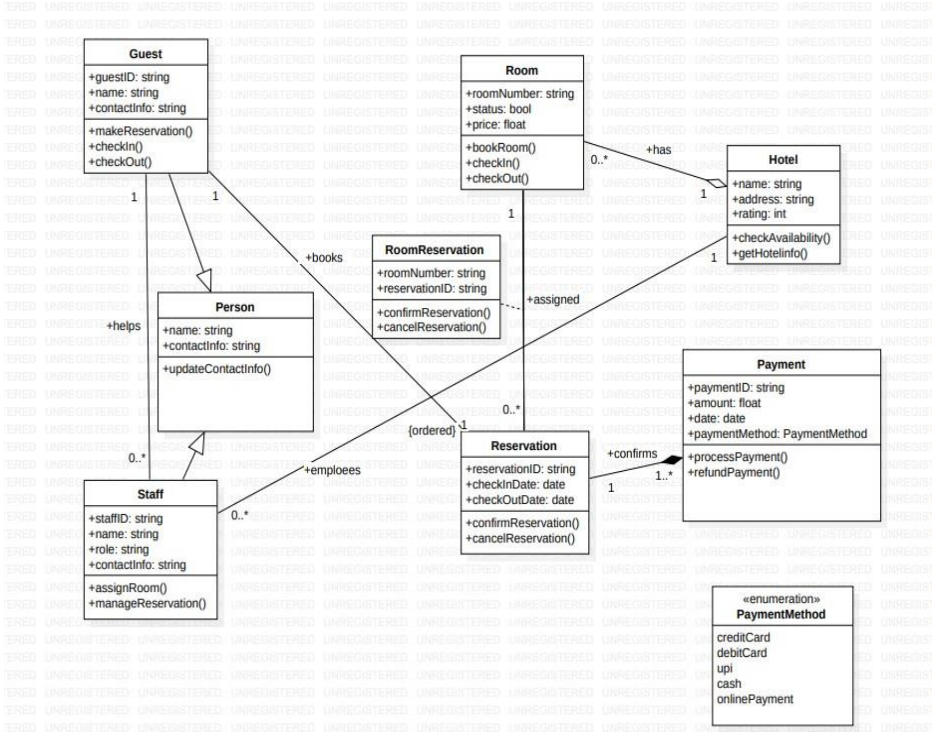
7. Non-functional Attributes:

- Security: Encryption of sensitive data, role-based access control
- Portability: Compatability with multiple devices & platforms
- Reliability: Minimal downtime, data backup & recovery mechanisms
- Reusability: Modular design for future enhancements
- Application compatability: Integration with third party services

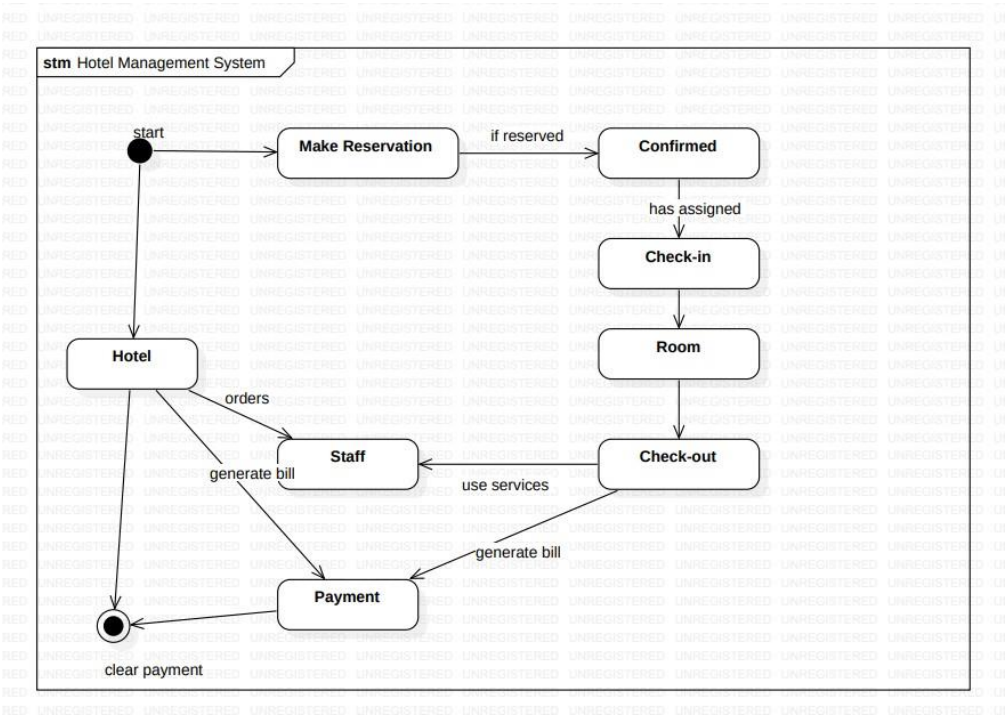
8. Preliminary Schedule & Budget -

The development of the hotel Mangement software is estimated to take 6 months with a budget of $100,000. This includes analysis, design, development, testing & development phase.

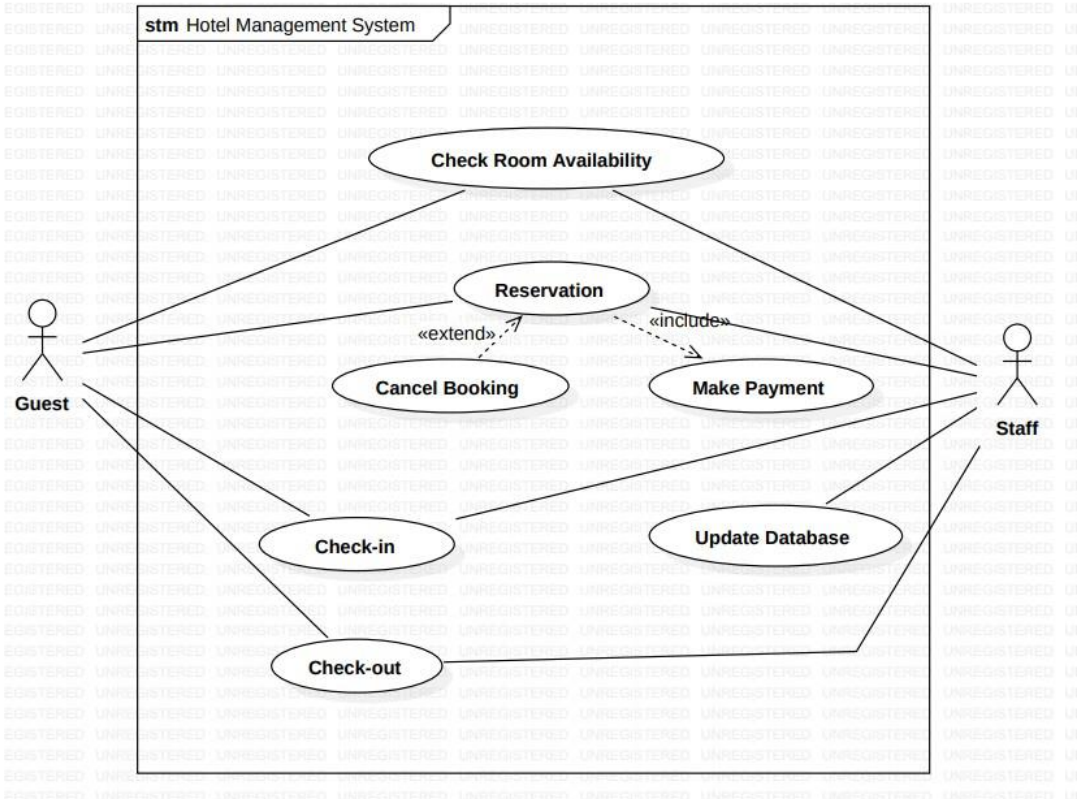
1.3 Class Diagram



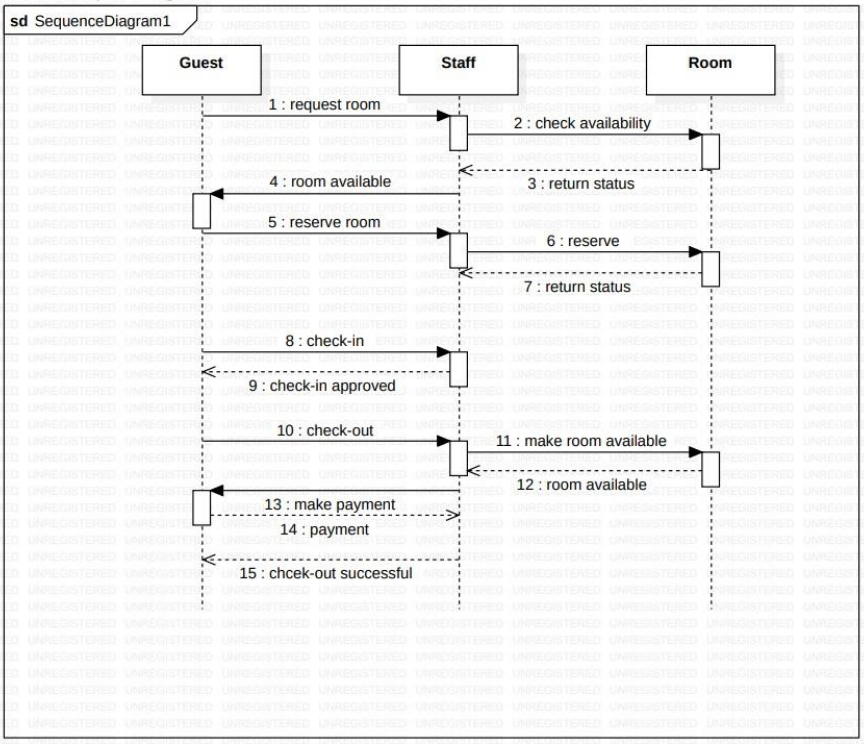
1.4State Diagram



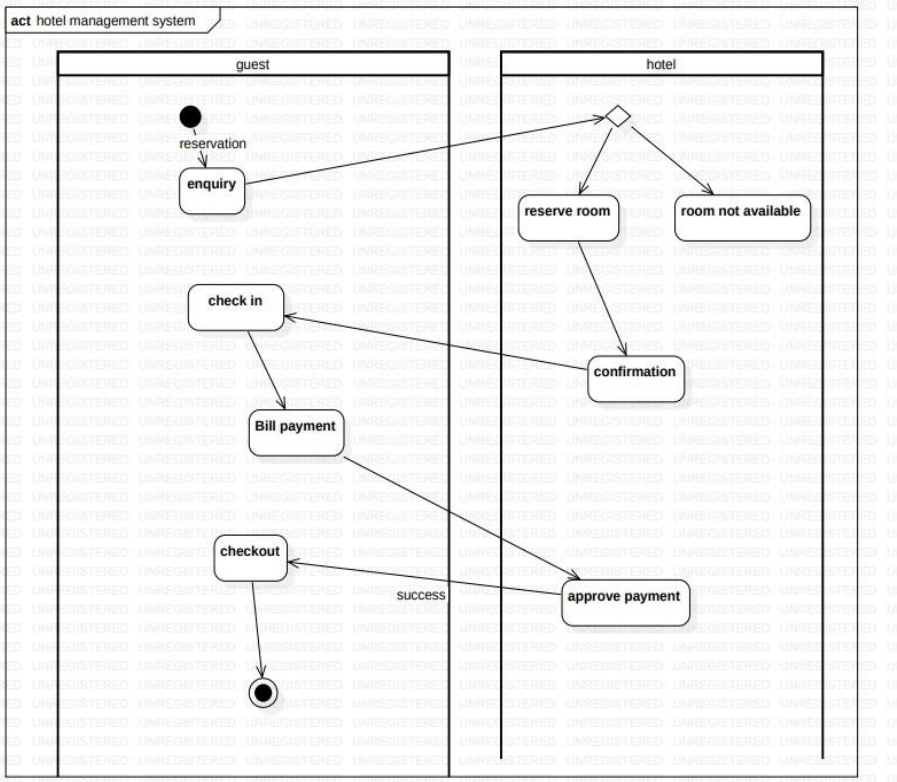
1.5Use Case Diagram



1.6 Sequence Diagram



1.7 Activity diagram

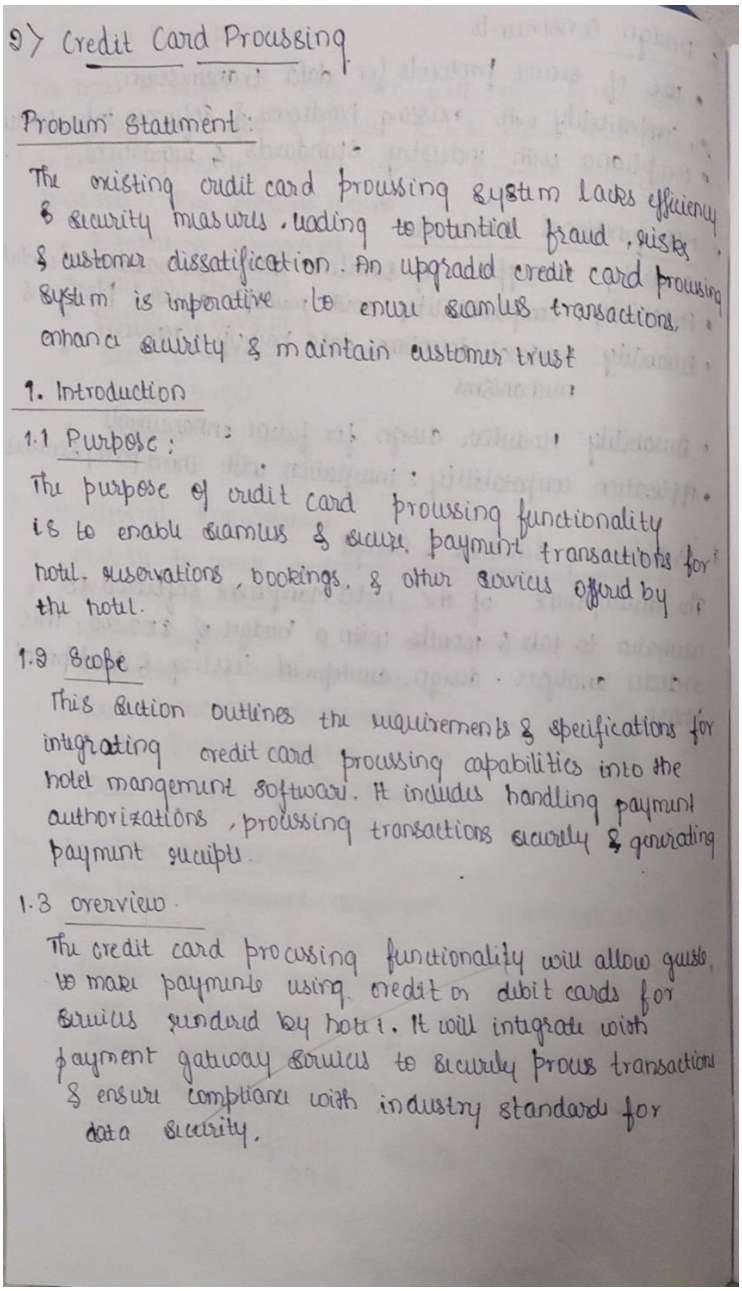


2. Credit Card Processing System

2.1 Problem Statement

The existing card processing system lacks efficiency and security measures, leading to potential fraud risks and customer dissatisfaction. An upgraded credit card processing system is imperative to ensure seamless transactions, enhance security, and maintain customer trust.

2.2 SRS-Software Requirements Specification



2) Credit Card Processing

Problem Statement:

The existing credit card processing system lacks efficiency & security measures, leading to potential fraud risks & customer dissatification. An upgraded credit card processing system is imperative to ensure seamless transactions, enhance security & maintain customer trust

1. Introduction

1.1 Purpose:

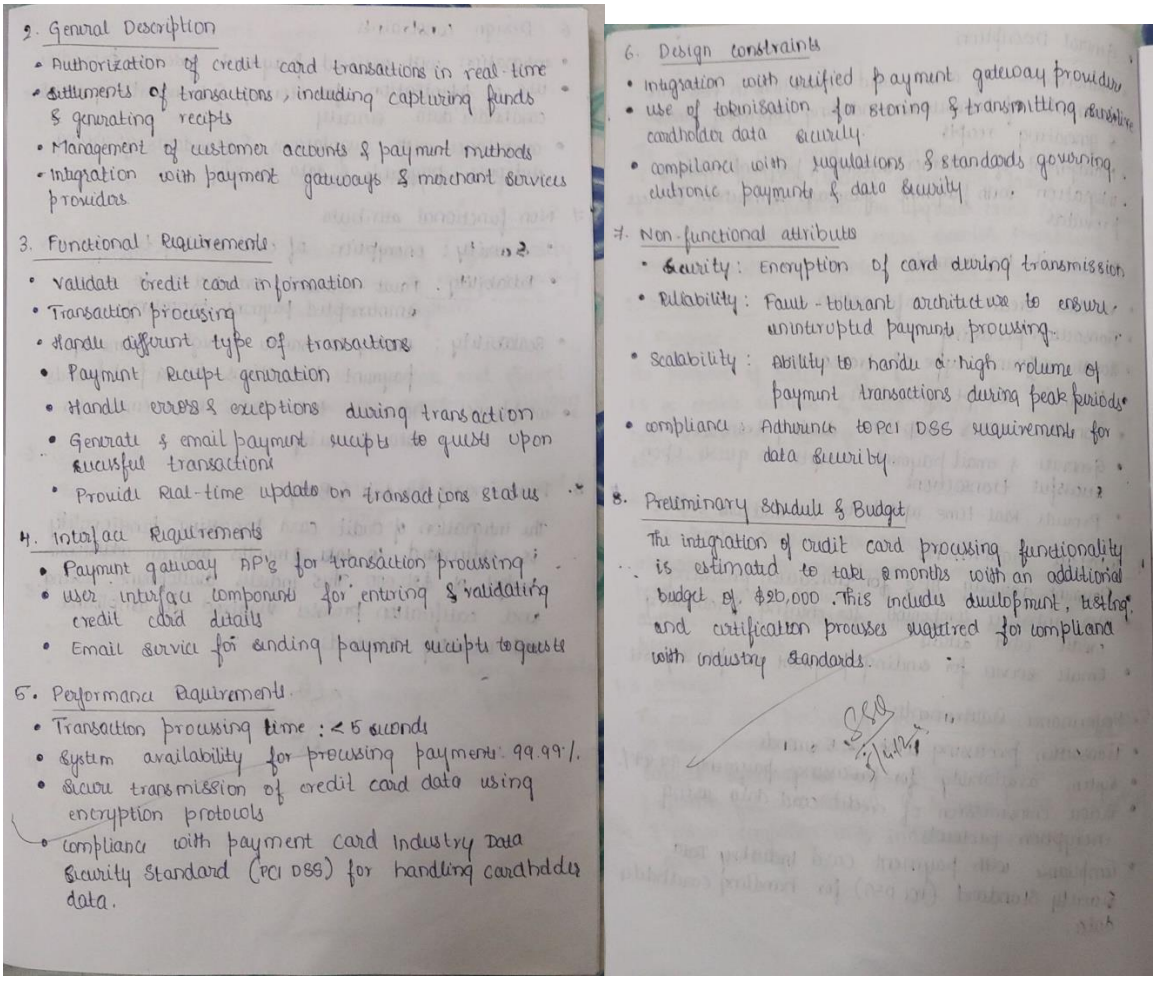
The purpose of credit card processing functionality is to enable seamless & secure payment transactions for hotel reservations, bookings, & other services offered by the hotel.

1.2 Scope

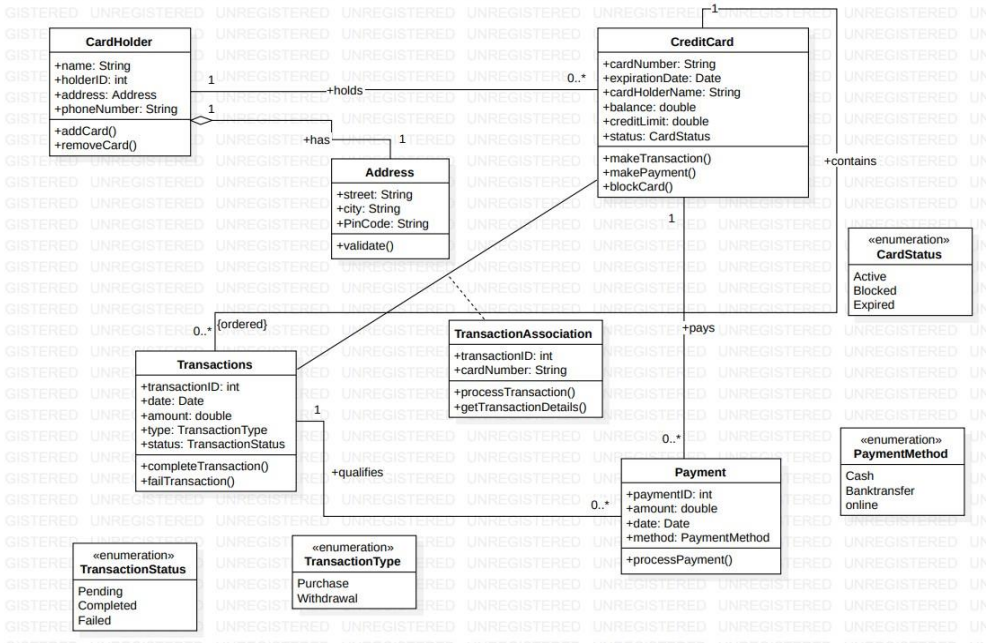
This section outlines the requirements & specifications for integrating credit card processing capabilities into the hotel mangement software. It includes handling payment authorizations, processing transactions securely & generating payment receipts.

1.3 overview

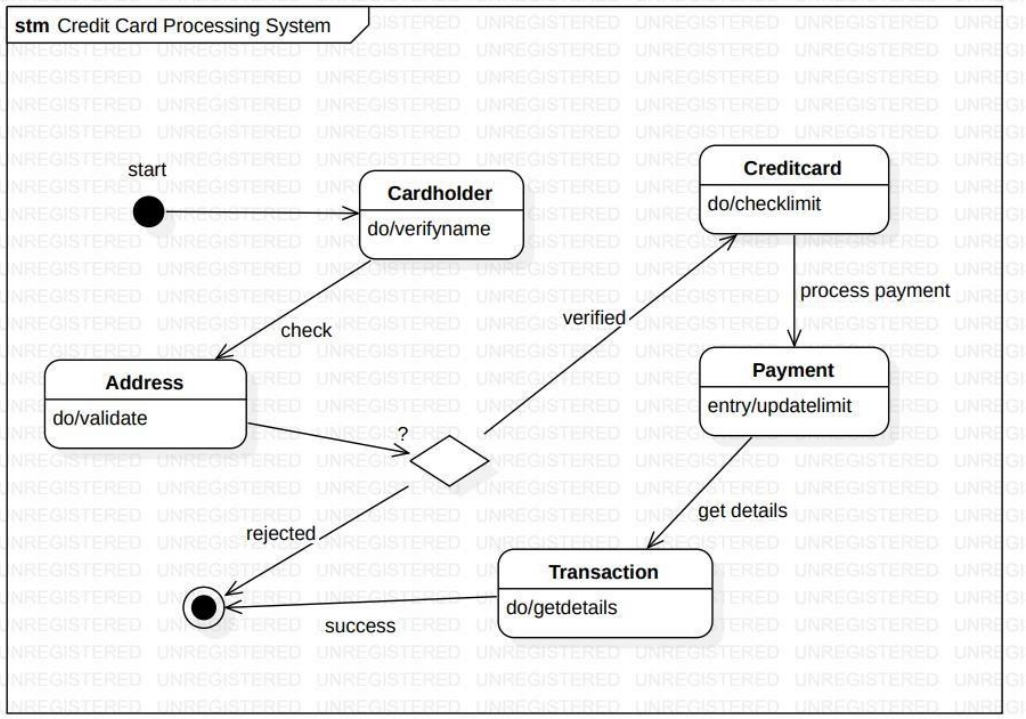
The credit card processing functionality will allow guests to make payments using credit or debit cards for services rendered by hotel. It will integrate with payment gateway services to securely process transactions & ensure compliance with industry standards for data security.



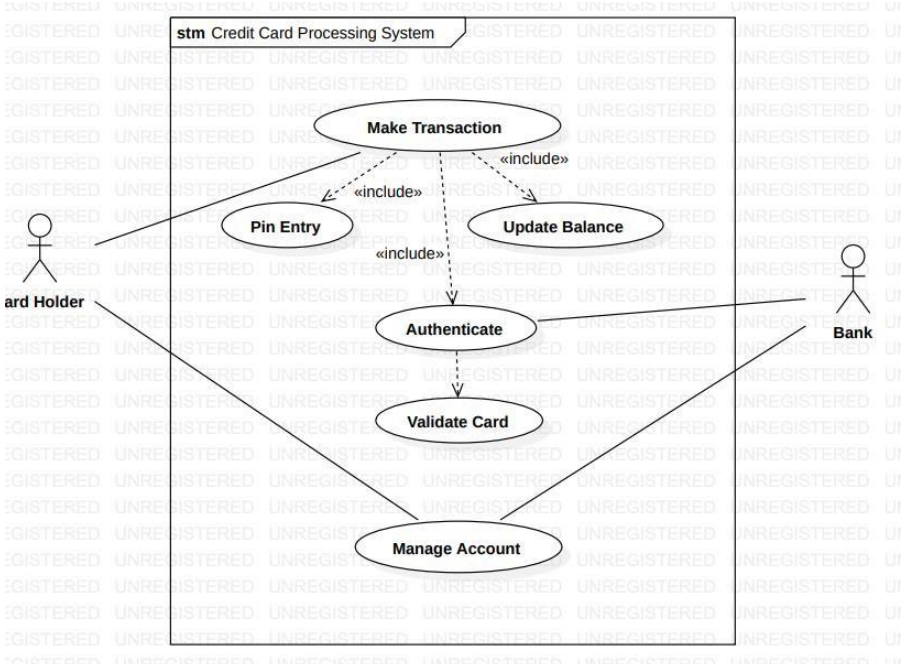
2.3 Class Diagram



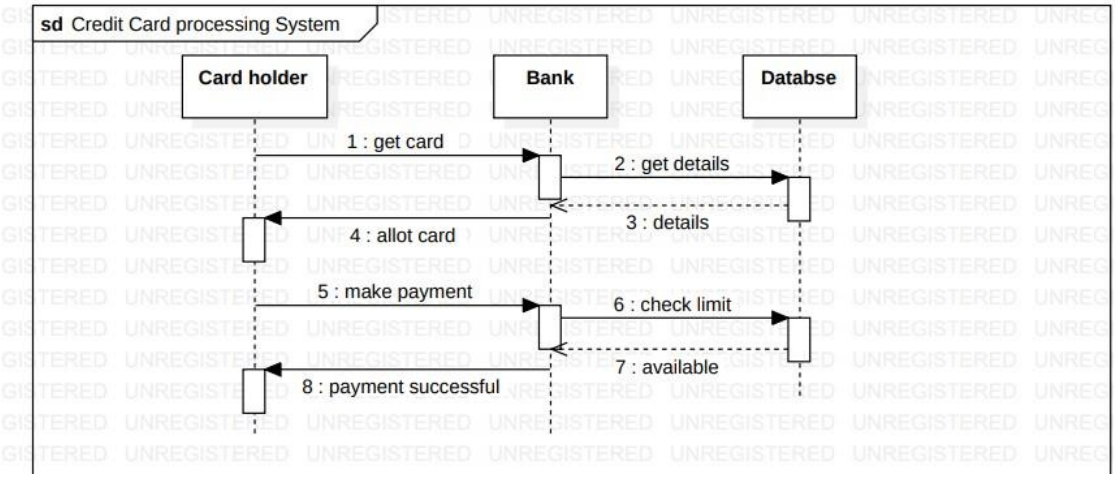
2.4 State Diagram



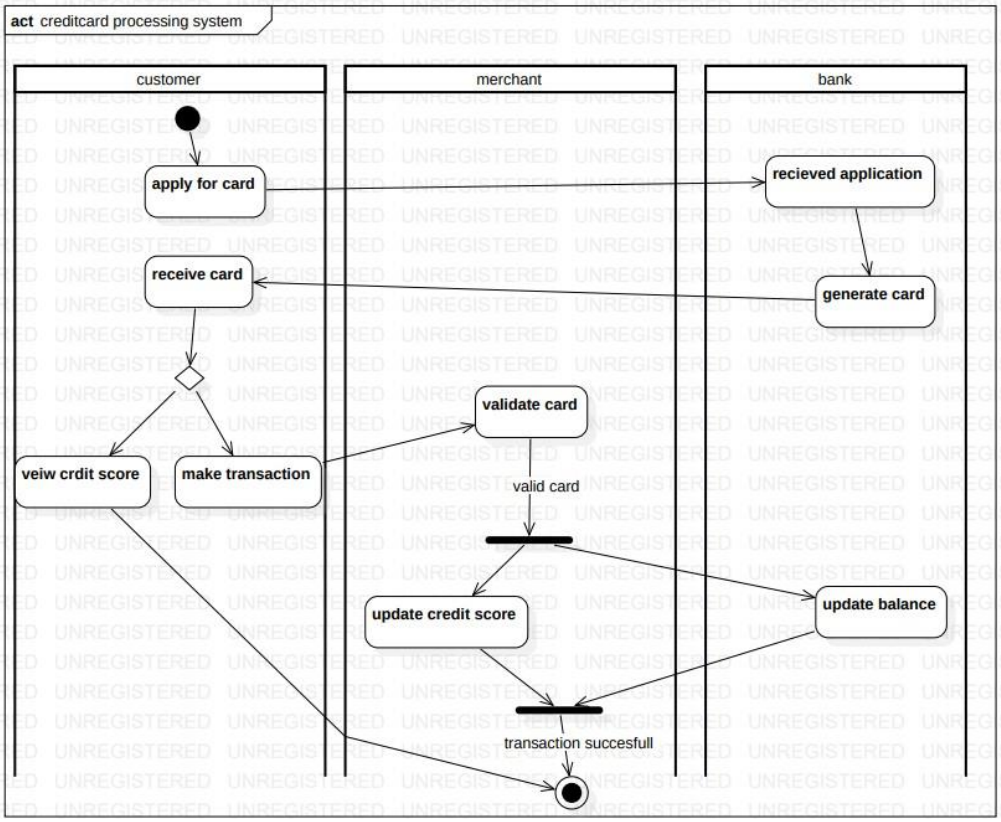
2.5 Use Case Diagram



2.6 Sequence Diagram



2.7 Activity diagram

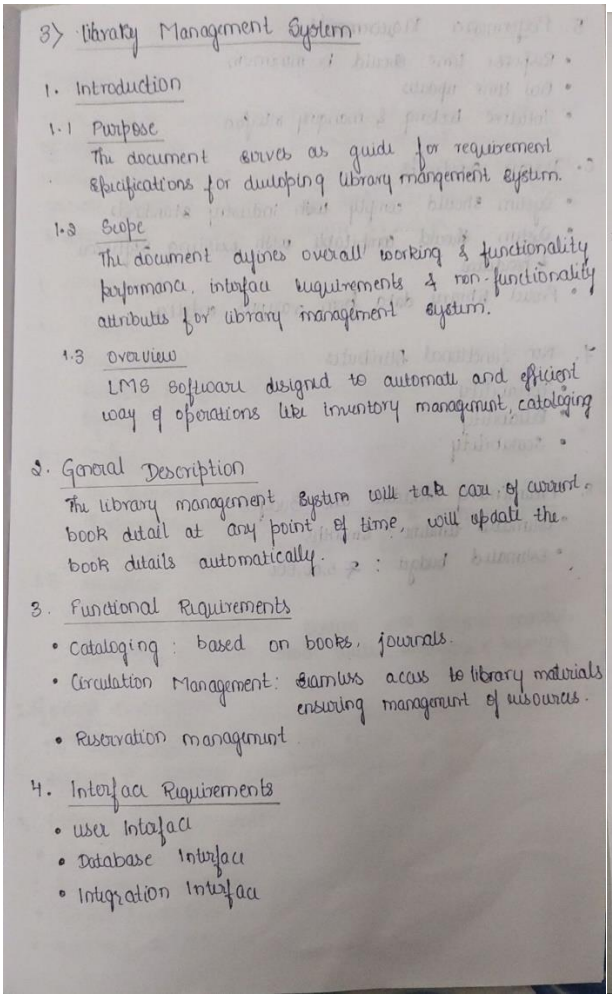


3. Library Management System

3.1 Problem Statement

Design and implement a Library Management System (LMS) for object modeling software. The system should efficiently manage the various operations related to library management, such as adding, updating, and deleting books, managing member records, handling borrowing and returning of books, and providing administrative functionalities for librarians.

3.2 SRS-Software Requirements Specification



3) Library Management System

1. Introduction

1.1 Purpose

The document serves as guide for requirement specifications for dveloping library mangement system.

1.2 Scope

The document defines overall working & functionality performance, interface requirements & non-functionality attributes for library management system.

1.3 Overview

LMS software designed to automate and efficient way of operations like inventory management, cataloging

2. General Description

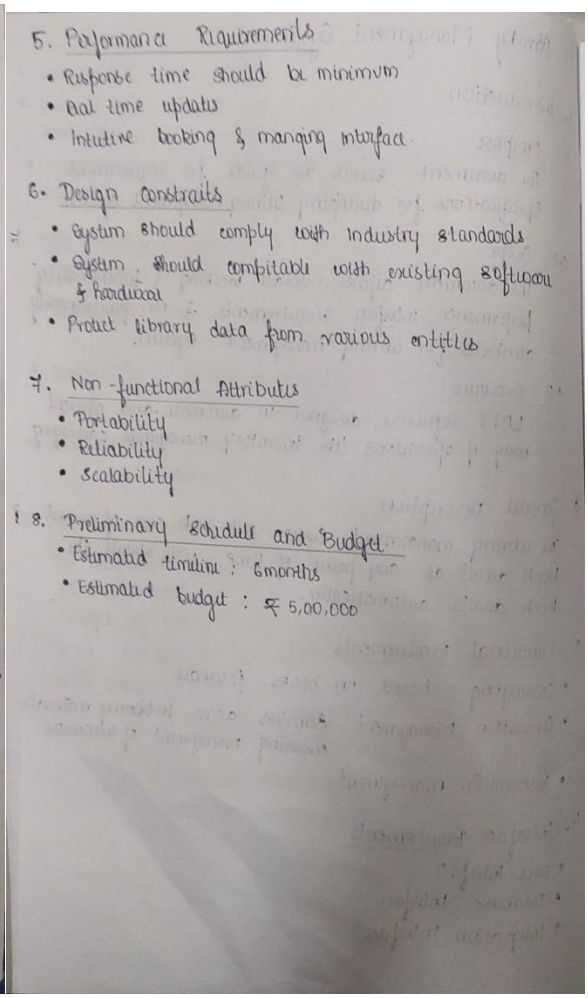
The library management system will take care of current book detail at any point of time, will update the book details automatically.

3. Functional Requirements

- Cataloging : based on books, journals.
- Circulation Management: Seamless access to library materials ensuring management of resources.
- Reservation management

4. Interface Requirements

- user Intafac
- Database Interfac
- Integration Intefac



5. Performance Requirements

- Response time should be minimum
- Real time updates
- Intuitive booking & manging interfac

6. Design constraits

- System should comply with industry standards
- System should compitable with existing software & hardware
- Protect library data from various entities

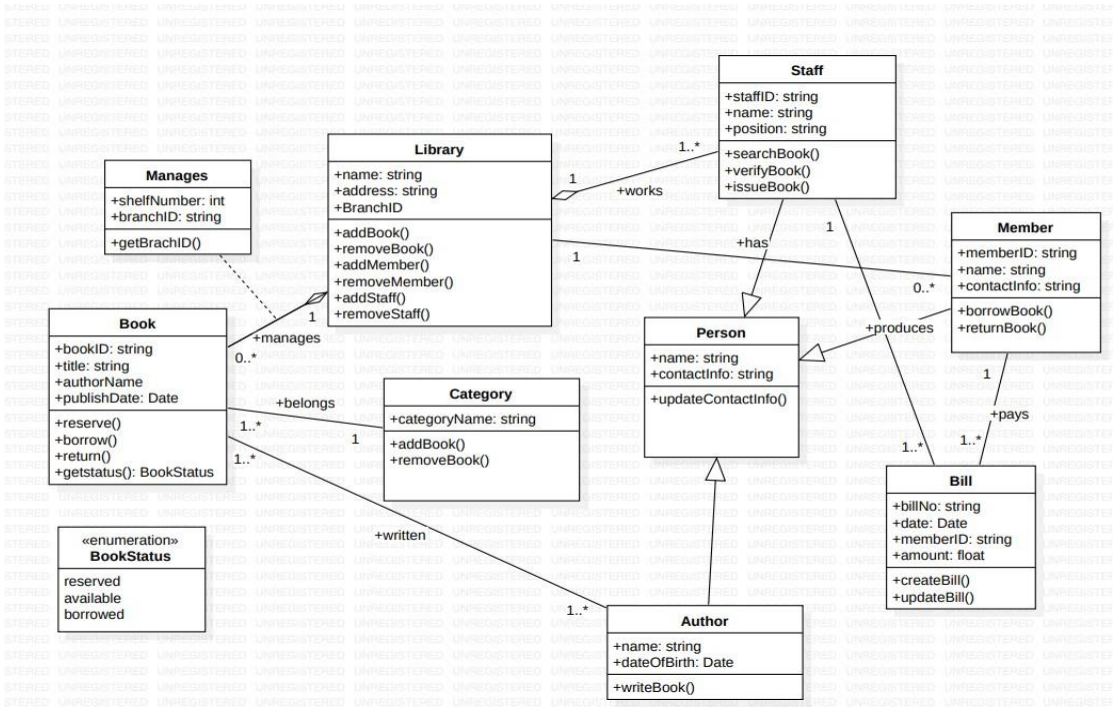
7. Non-functional Attributes

- Portability
- Reliability
- Scalability

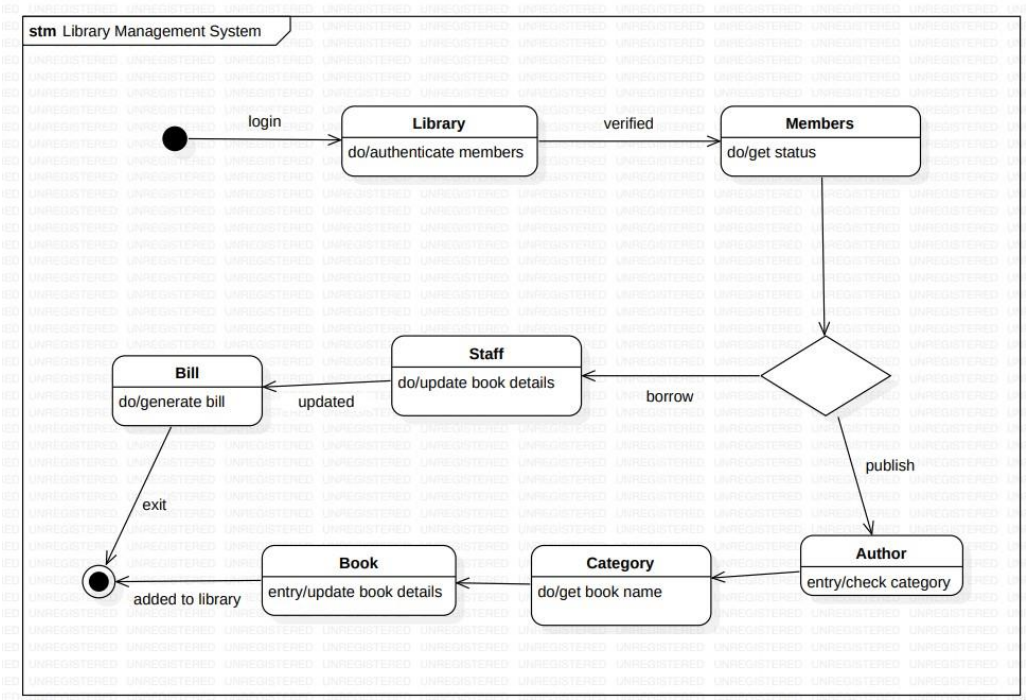
8. Preliminary Schedule and Budget

- Estimated timeline : 6months
- Estimated budget : ₹ 5,00,000

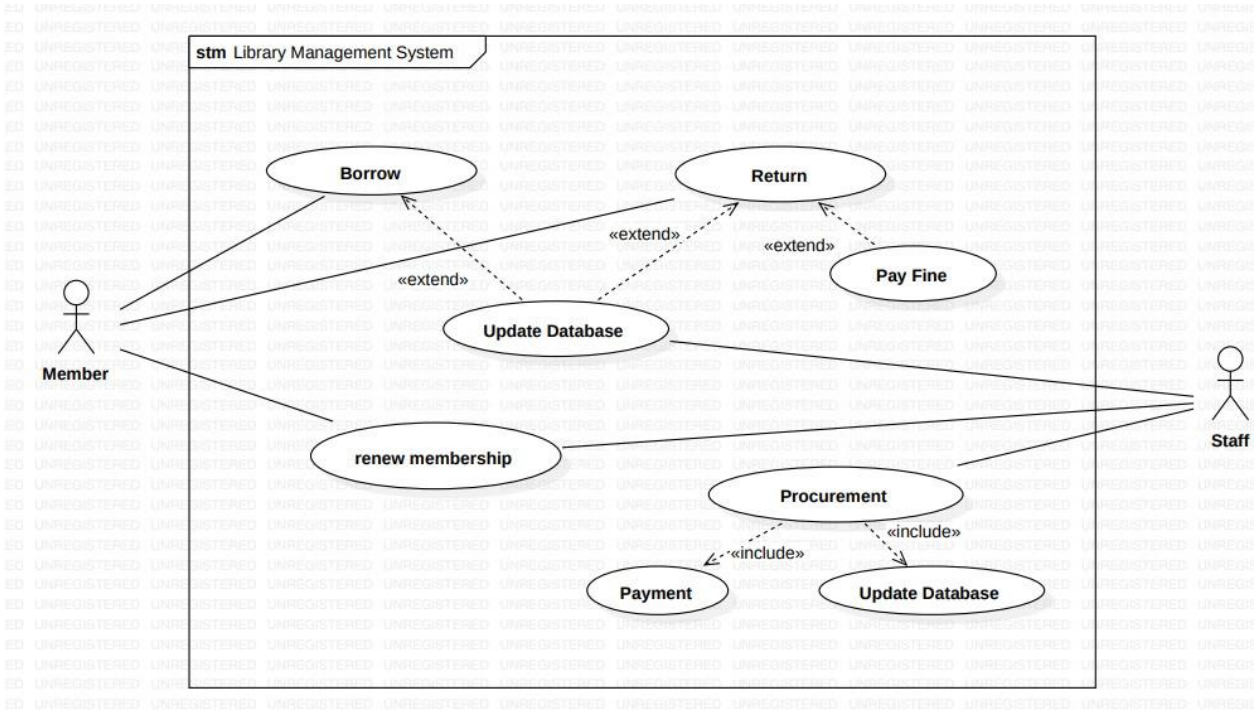
3.3 Class Diagram



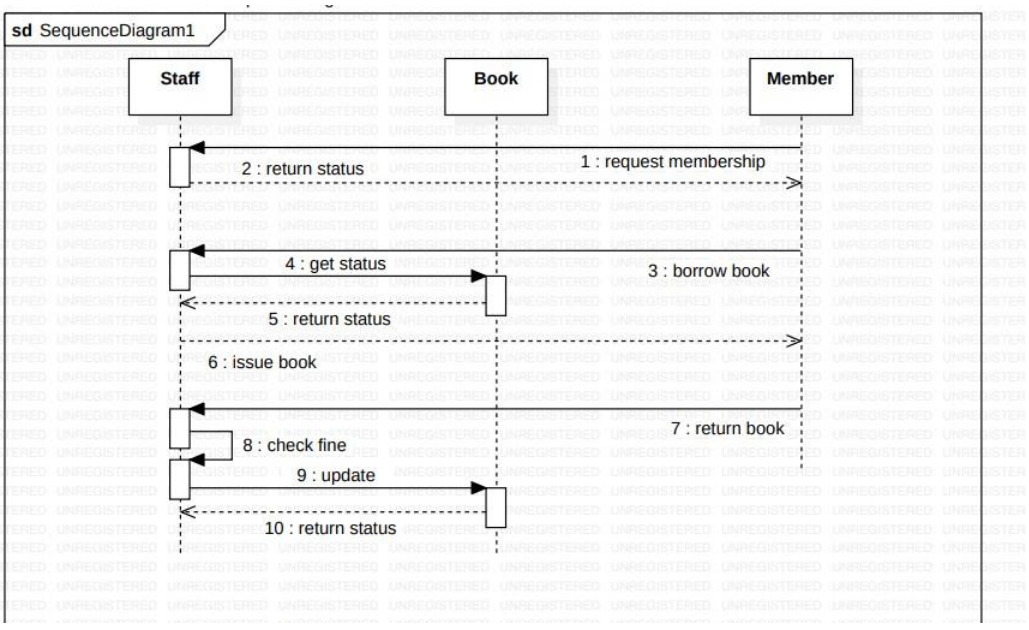
3.4 State Diagram



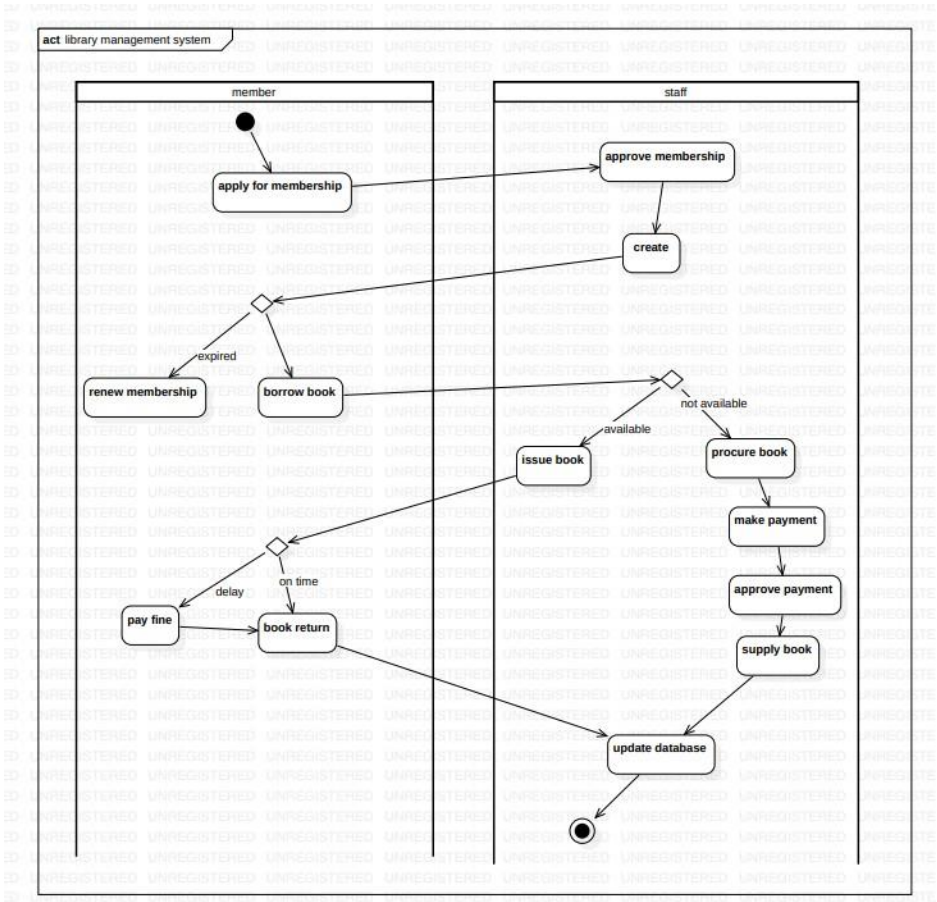
3.5 Use Case Diagram



3.6 Sequence Diagram



3.7 Activity diagram

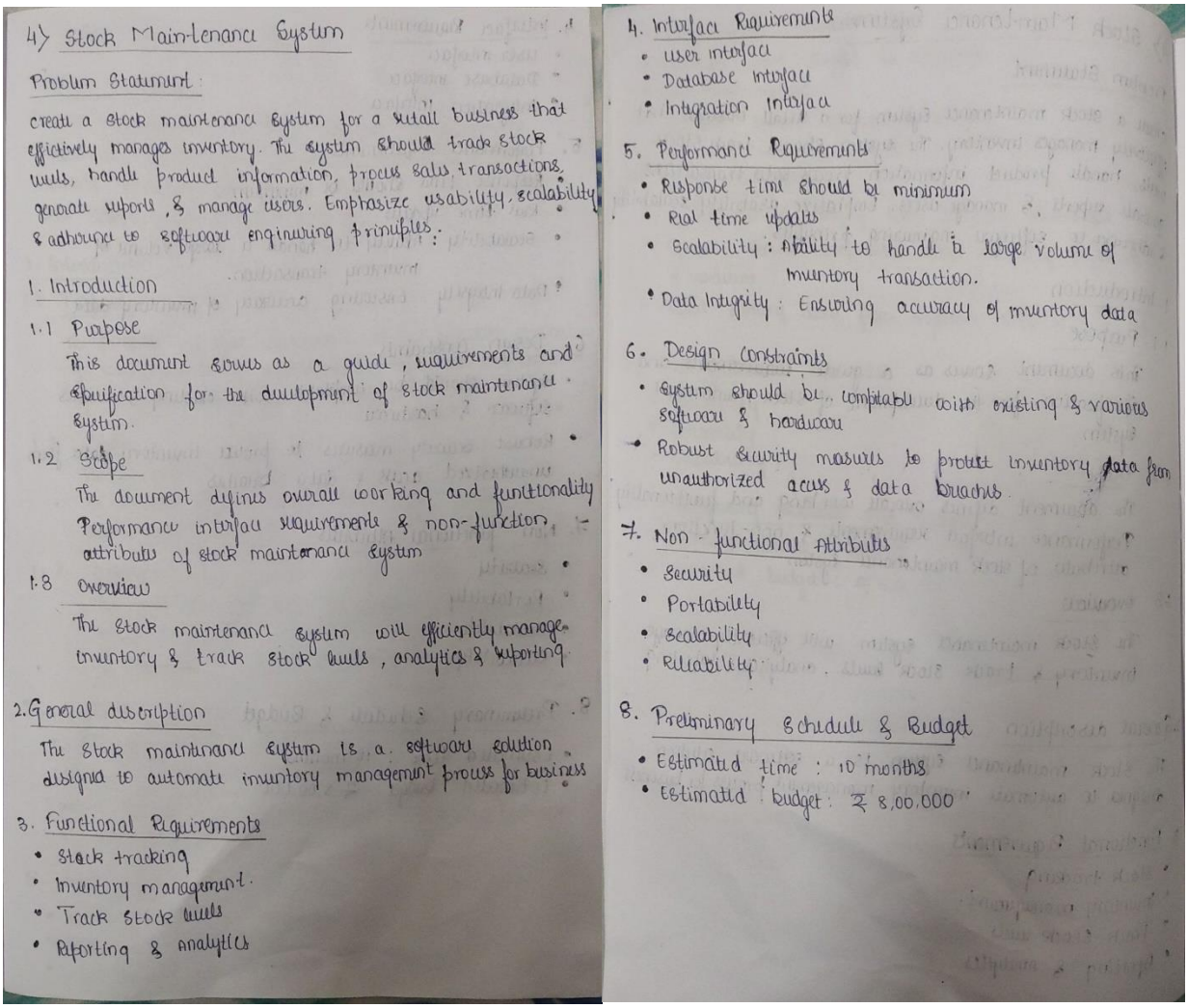


4. Stock Maintenance System

4.1 Problem Statement

Create a stock management system named Bystim for a small business that effectively manages inventory. The system should track stock levels, handle product information, process sales transactions, generate reports, and manage analysis. Emphasize usability, scalability, and adherence to industry-standard engineering principles.

4.2 SRS-Software Requirements Specification



4) Stock Main-tenanca System

Problem Statement:

create a stock maintenanca system for a retail business that effectively manages inventory. The system should track stock levels, handle product information, process sales transactions, generate reports, & manage users. Emphasize usability, scalability & adherence to software engineering principles.

1. Introduction

1.1 Purpose

This document serves as a guide, requirements and specification for the development of stock maintenance system.

1.2 Scope

The document defines overall working and functionality Performance interface requirements & non-function attributes of stock maintenance system

1.3 Overview

The stock maintenance system will efficiently manage inventory & track stock levels, analytics & reporting

2. General description

The stock maintenance system is a software solution designed to automate inventory management process for business

3. Functional Requirements

- Stock tracking
- Inventory management
- Track stock levels
- Reporting & analytics

4. Interface Requirements

- user interface
- Database interface
- Integration interface

5. Performance Requirements

- Response time should be minimum
- Real time updates
- Scalability: Ability to handle a large volume of inventory transaction.
- Data Integrity: Ensuring accuracy of inventory data

6. Design Constraints

- System should be compatable with existing & various software & hardware
- Robust security masures to protect inventory data from unauthorized acess & data breaches.

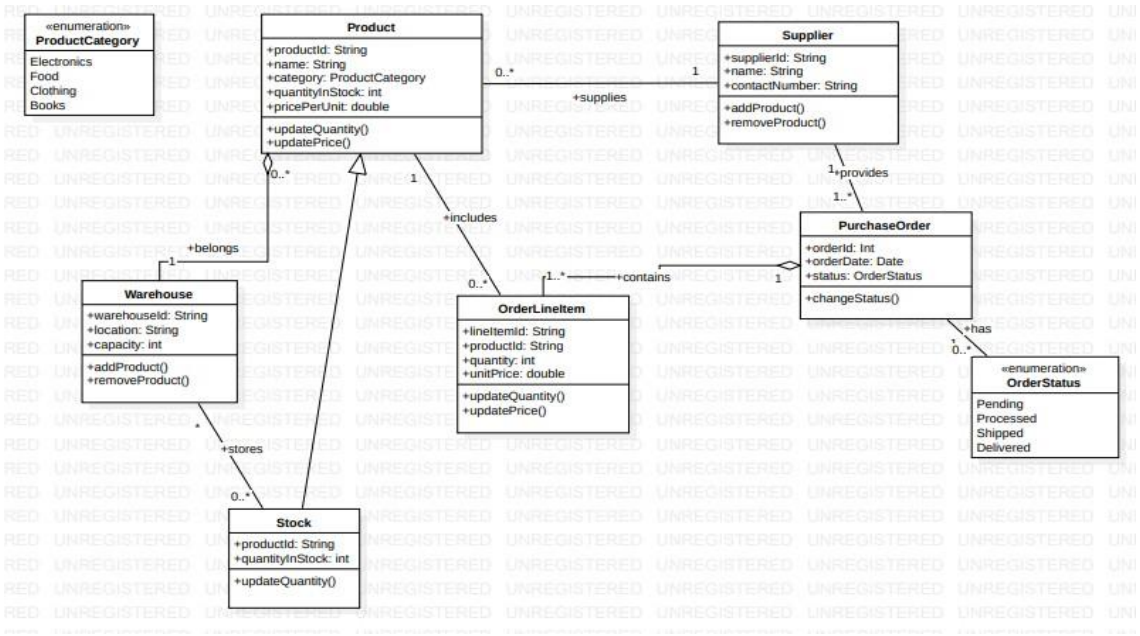
7. Non-functional Attributes

- Security
- Portability
- Scalability
- Reliability

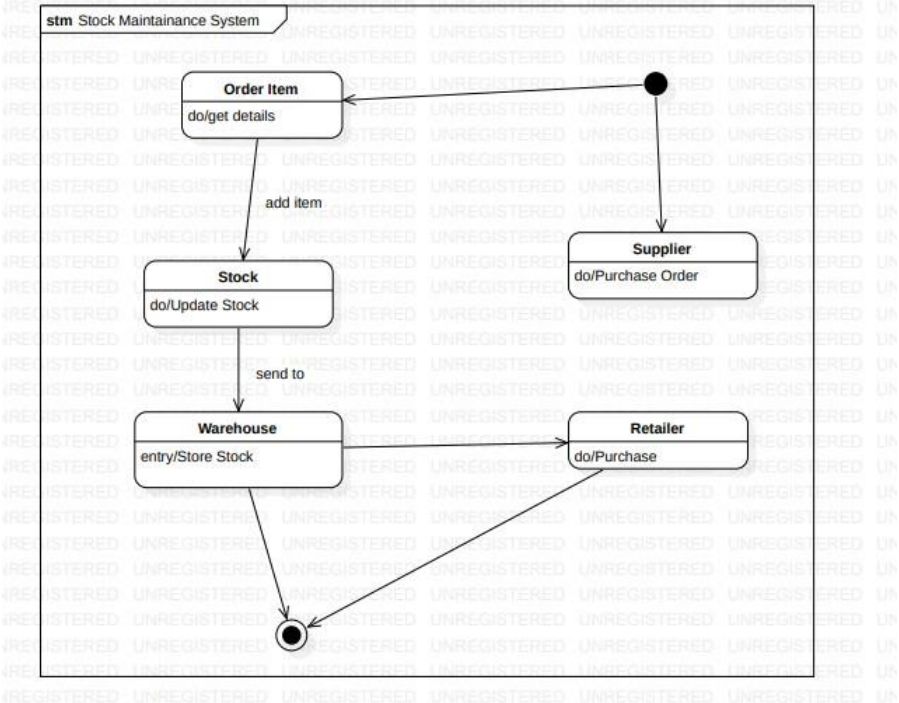
8. Preliminary Schedule & Budget

- Estimated time : 10 months
- Estimated budget: ₹ 8,00,000

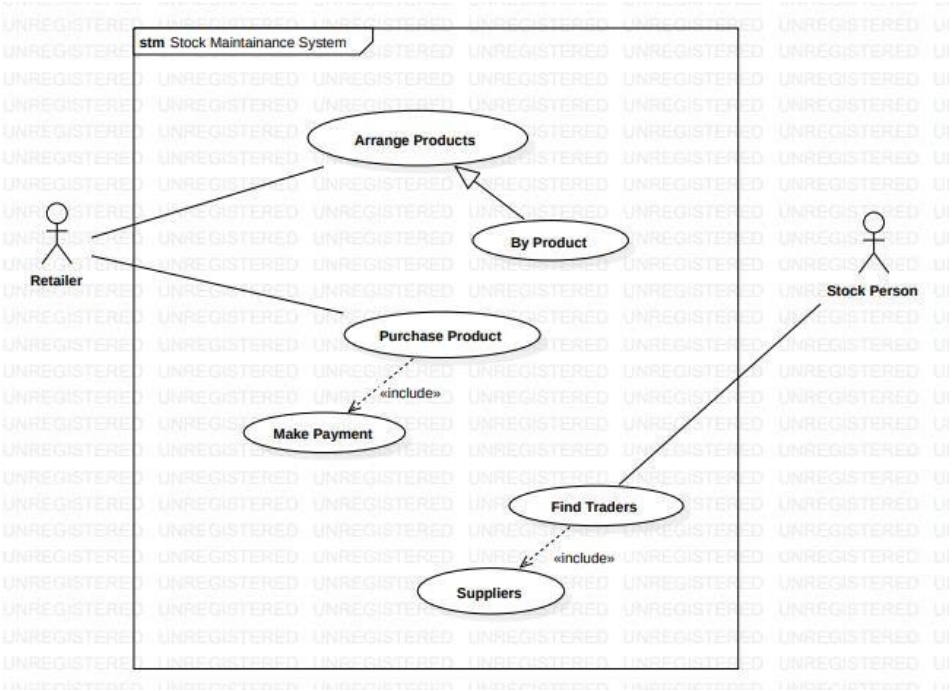
4.3 Class Diagram



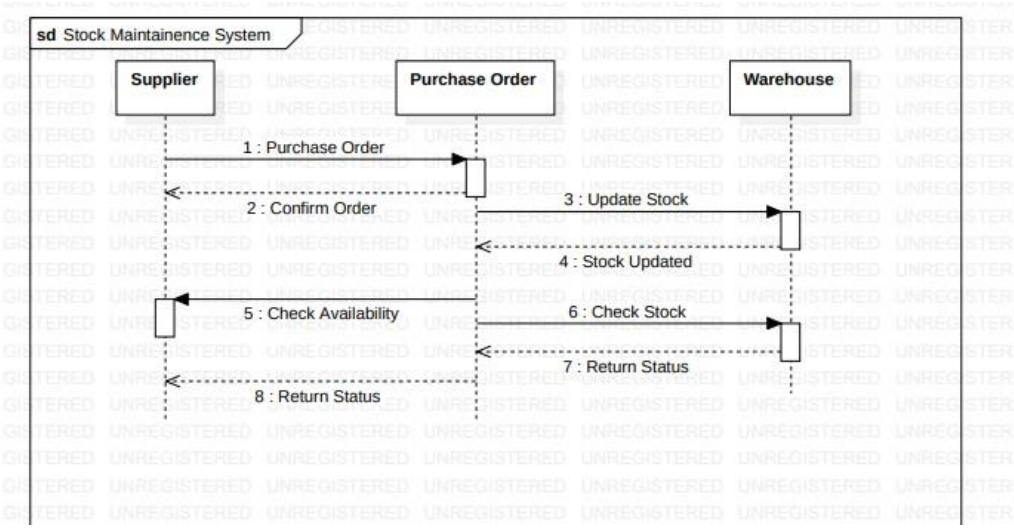
4.4 State Diagram



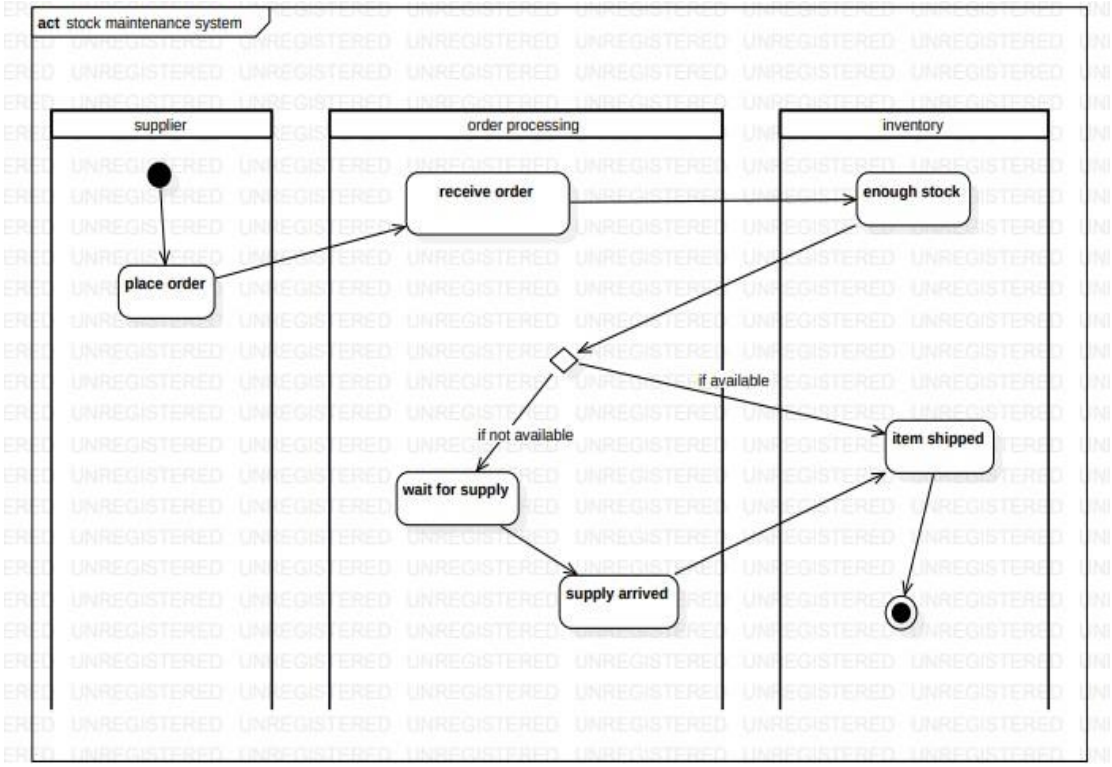
4.5 Use Case Diagram



4.6 Sequence Diagram



4.7 Activity diagram

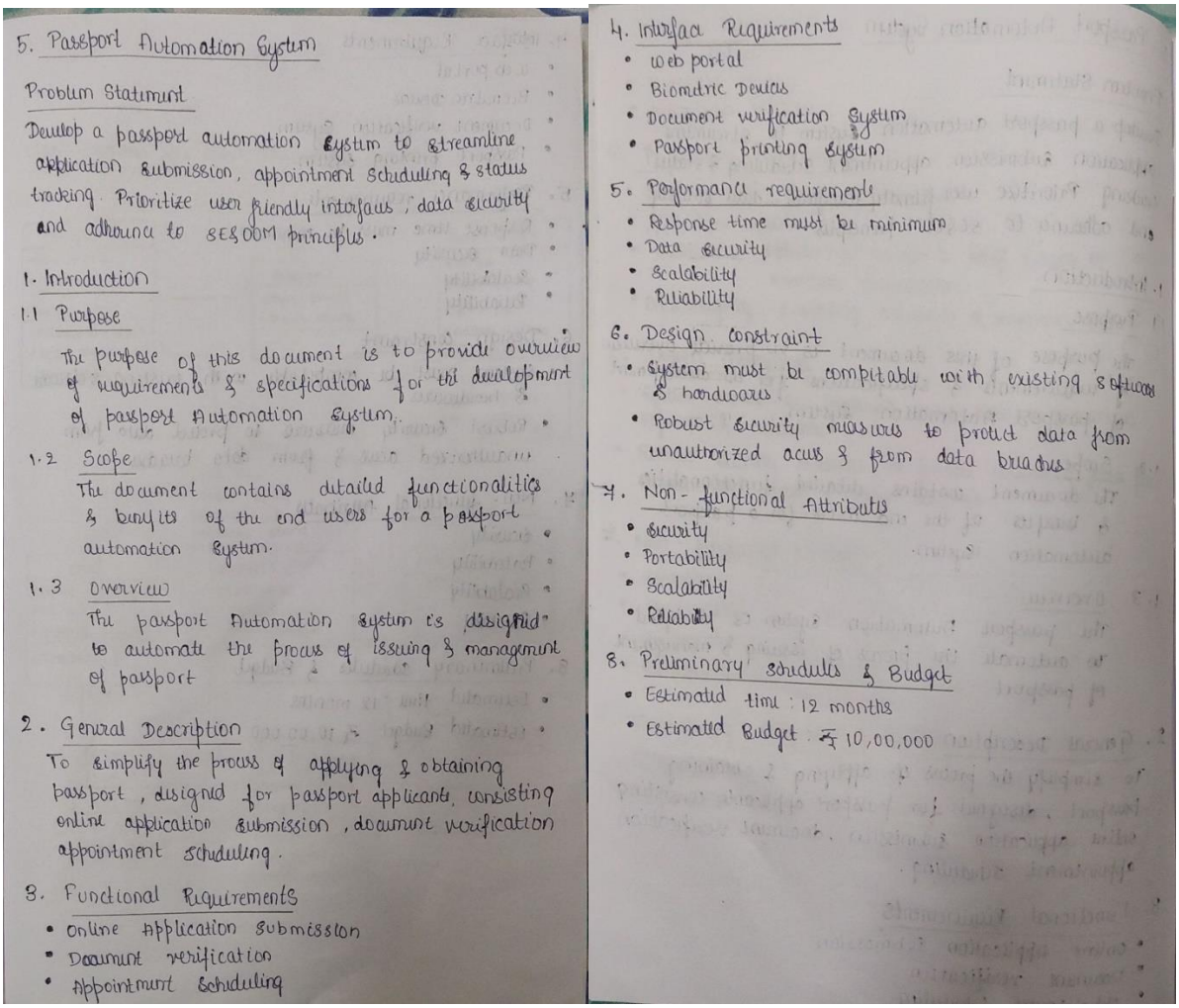


5. Passport Automation System

5.1 Problem Statement

Develop a passport automation system to streamline application submission, appointment scheduling, and status tracing.

5.2 SRS-Software Requirements Specification



5. Passport Automation System

Problem Statement

Develop a passport automation system to streamline application submission, appointment scheduling & status tracking. Prioritize user friendly interfaces, data security and adherence to SESOOM principles.

1. Introduction

1.1 Purpose

The purpose of this document is to provide overview of requirements & specifications for the development of passport Automation System.

1.2 Scope

The document contains detailed functionalities & benefits of the end users for a passport automation System.

1.3 Overview

The passport Automation System is designed to automate the process of issuing & management of passport

2. General Description

To simplify the process of applying & obtaining passport, designed for passport applicants, consisting online application submission, document verification appointment scheduling.

3. Functional Requirements

- Online Application submission
- Document verification
- Appointment Scheduling

4. Interface Requirements

- web portal
- Biometric Devices
- Document verification System
- Passport printing System

5. Performance requirements

- Response time must be minimum
- Data security
- Scalability
- Reliability

6. Design constraint

- System must be compitable with existing software & hardwares
- Robust security measures to protect data from unauthorized acess & from data breaches

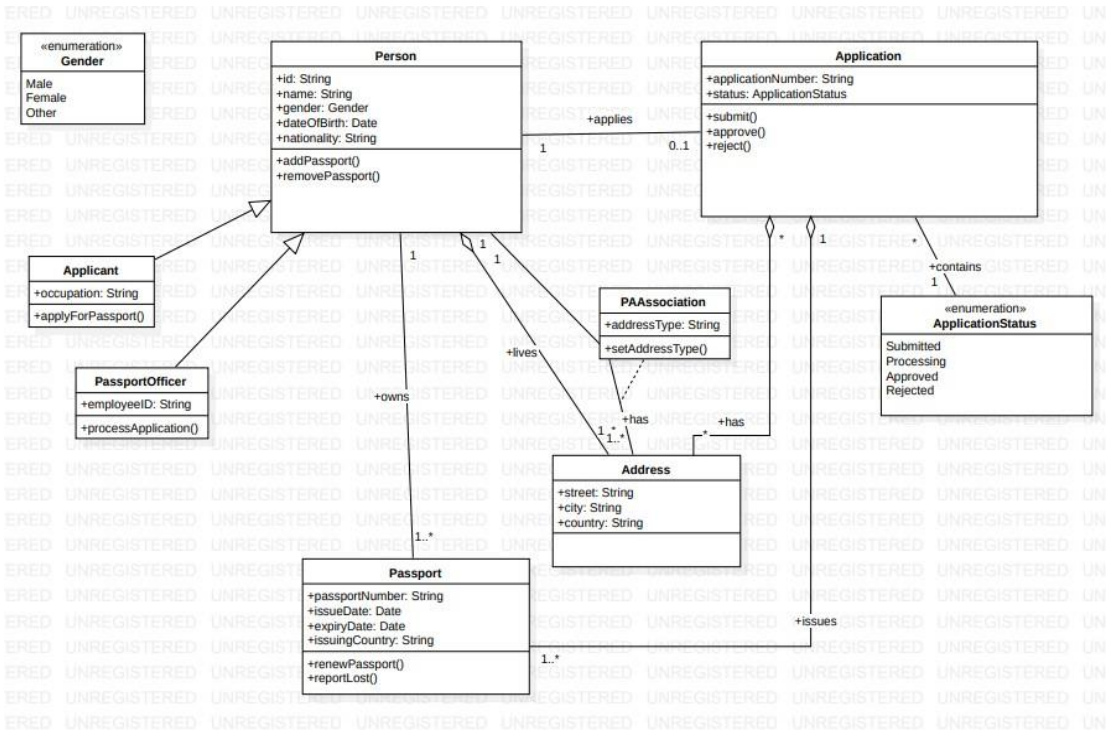
7. Non-functional Attributes

- Security
- Portability
- Scalability
- Reliability

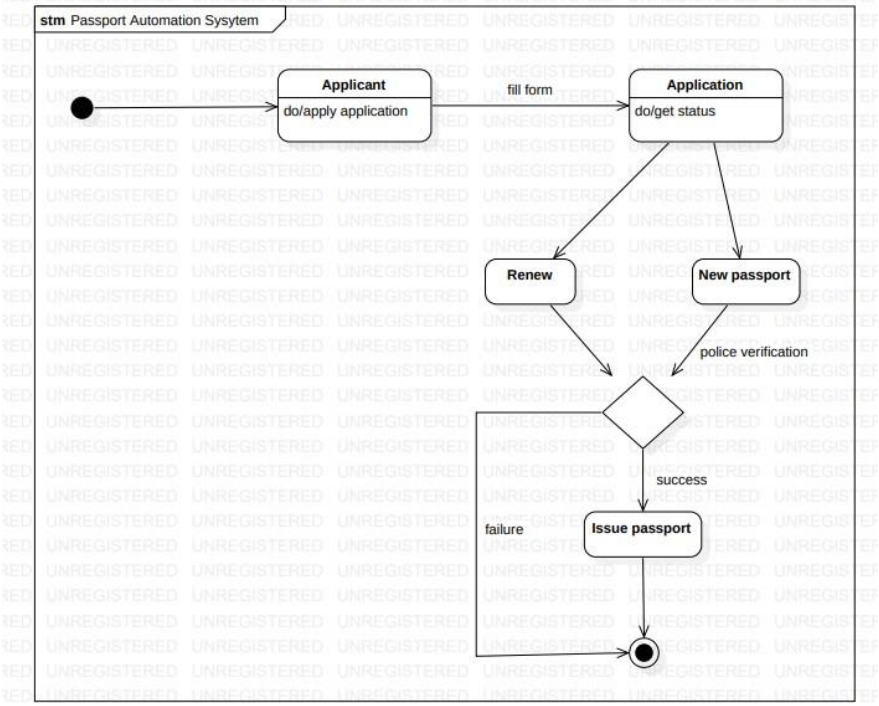
8. Preliminary schedules & Budget

- Estimated time : 12 months
- Estimated Budget : ₹ 10,00,000

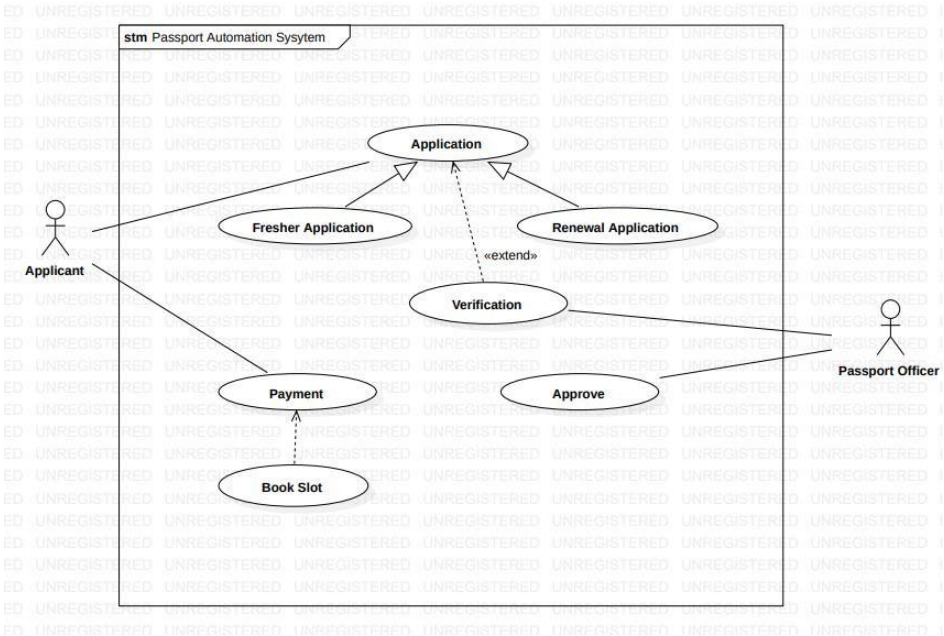
5.3 Class Diagram



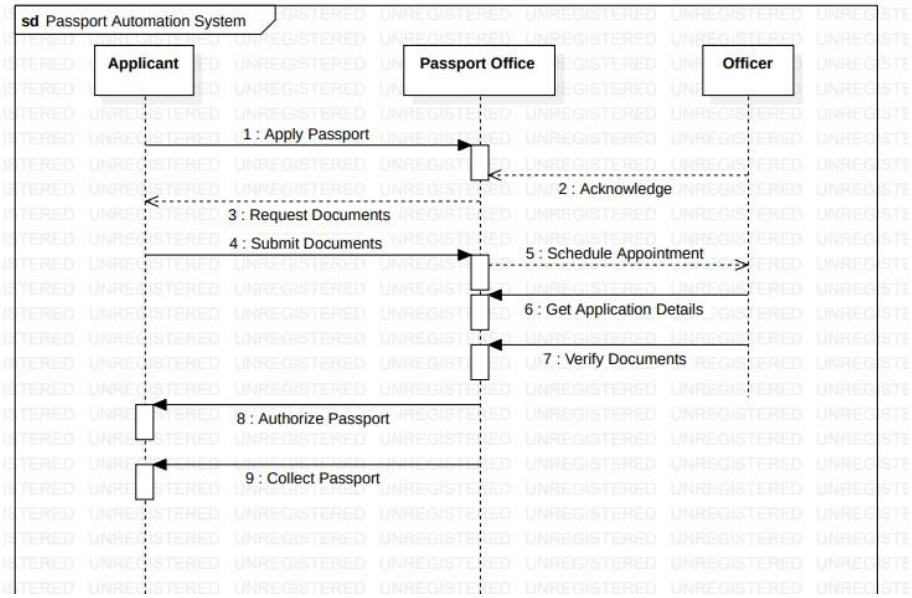
5.4 State Diagram



5.5 Use Case Diagram



5.6 Sequence Diagram



5.7 Activity diagram