

**B.M.S. COLLEGE OF ENGINEERING BENGALURU**  
Autonomous Institute, Affiliated to VTU



Lab Record

**Software Engineering and Object-Oriented Modelling**

*Submitted in partial fulfillment for the 6<sup>th</sup> Semester Laboratory*

Bachelor of Engineering  
in  
Computer Science and Engineering

*Submitted by:*

**Amshu G M**  
**(1BM21CS019)**

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 **Amslu G M (1BM21CS019)** during the 6<sup>th</sup> Semester Mar-June-2024.

Signature of the Faculty Incharge:

Syed Akram,  
Assistant Professor,  
Department of Computer Science and Engineering  
B.M.S. College of Engineering, Bangalore

## **Table of Contents**

<b>SI. No.</b>	<b>Title</b>	<b>Page No.</b>
1	Hotel Management System	1-7
2	Credit Card Processing	8-13
3	Library Management System	14-20
4	Stock Maintenance System	21-27
5	Passport Automation System	28-34

# **1. HOTEL MANAGEMENT SYSTEM**

## **Problem Statement:**

The problem statement for the Hotel Management System project is to develop a software solution that can streamline and automate various operations and processes within a hotel, including reservation management, room allocation, check-in/check-out procedures, billing and payment processing, inventory management, and staff scheduling.

## **SRS (Software Requirements Specification):**

### **1. Introduction:**

#### a. Purpose:

The purpose of the Hotel Management System is to improve the efficiency and effectiveness of hotel operations by automating manual tasks and providing a centralized platform for managing various hotel functions.

#### b. Scope:

The scope of the project includes developing a user-friendly interface for hotel staff, integrating different modules for managing reservations, room allocation, billing, inventory, and staff scheduling, and ensuring seamless communication and data exchange between these modules.

#### c. Overview:

The Hotel Management System aims to provide a comprehensive software solution that enhances the overall management and operation of a hotel, leading to improved guest satisfaction, optimized resource utilization, and increased revenue.

### **2. General Description:**

The Hotel Management System will be a web-based application that can be accessed from different devices and platforms. It will include features such as guest registration, reservation management, room status tracking, billing and invoicing, inventory management, reporting, and administrative controls.

### **3. Functional Requirements:**

- User registration and login functionality
- Guest registration and check-in/check-out process
- Reservation management, including room availability and booking
- Room allocation and assignment
- Billing and payment processing
- Inventory management for tracking and managing hotel supplies
- Staff scheduling and management
- Reporting and analytics for monitoring hotel performance

### **4. Interface Requirements:**

- Intuitive and user-friendly interface for hotel staff
- Responsive design to support different devices and screen sizes
- Integration with external systems (if required) such as payment gateways or property management systems

### **5. Performance Requirements:**

- Fast and responsive system performance to ensure quick operations
- Scalability to handle a large number of concurrent users and data transactions
- Reliable data storage and retrieval to prevent data loss or corruption

### **6. Design Constraints:**

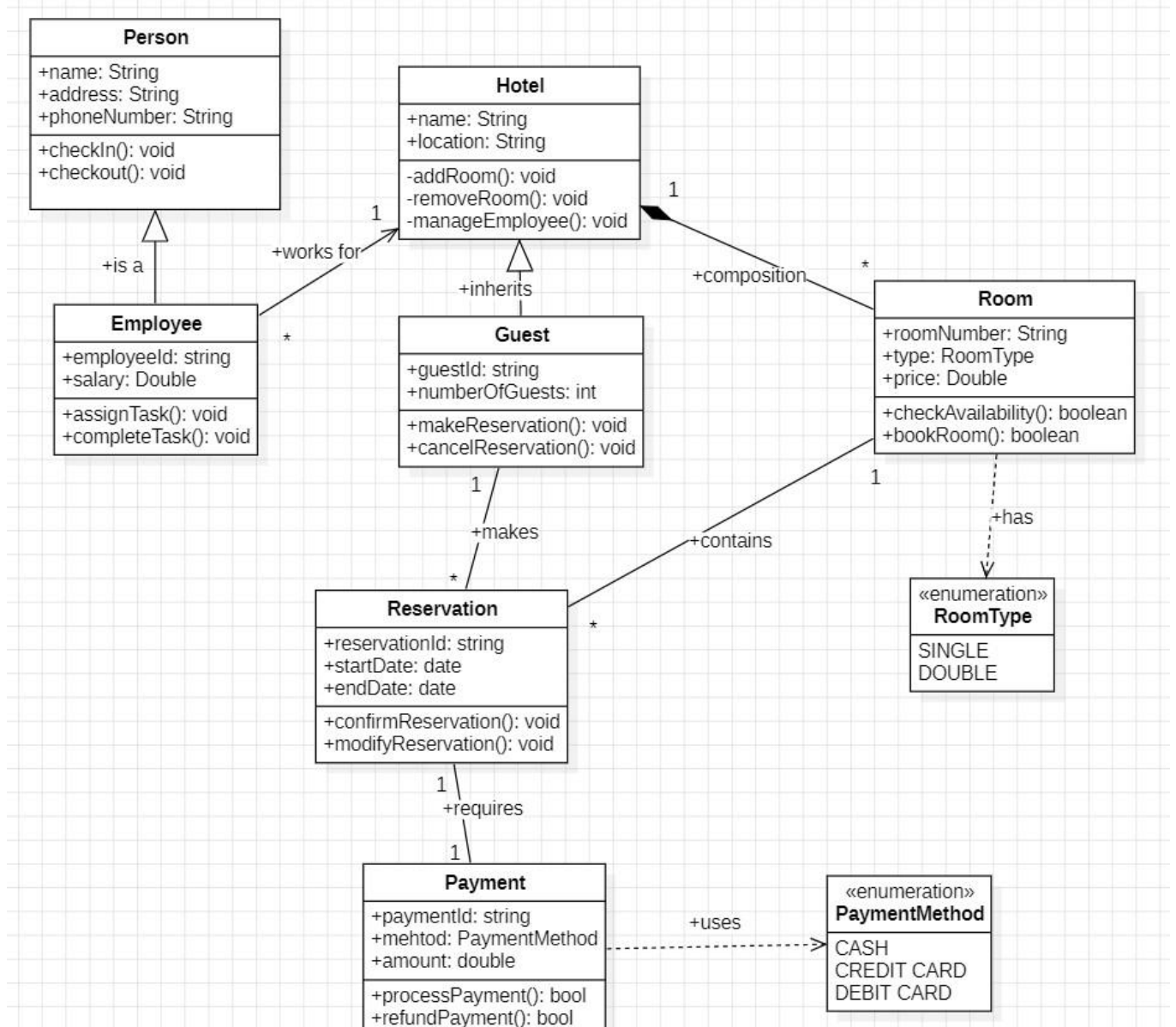
- Compliance with security standards and regulations to protect sensitive guest information
- Integration with existing hotel systems or technologies (if applicable)
- Compatibility with popular web browsers and operating systems

### **7. Non-Functional Attributes:**

- Usability: The system should be intuitive and easy to navigate for hotel staff.
- Reliability: The system should be reliable and available 24/7 to handle hotel operations without interruptions.
- Security: The system should ensure the confidentiality and integrity of guest and hotel data.
- Performance: The system should deliver fast response times and handle concurrent user requests efficiently.
- Maintainability: The system should be easy to maintain and update with minimal downtime.

# CLASS DIAGRAM

Figure 1.1 Class Diagram of Hotel Management System



# STATE DIAGRAM

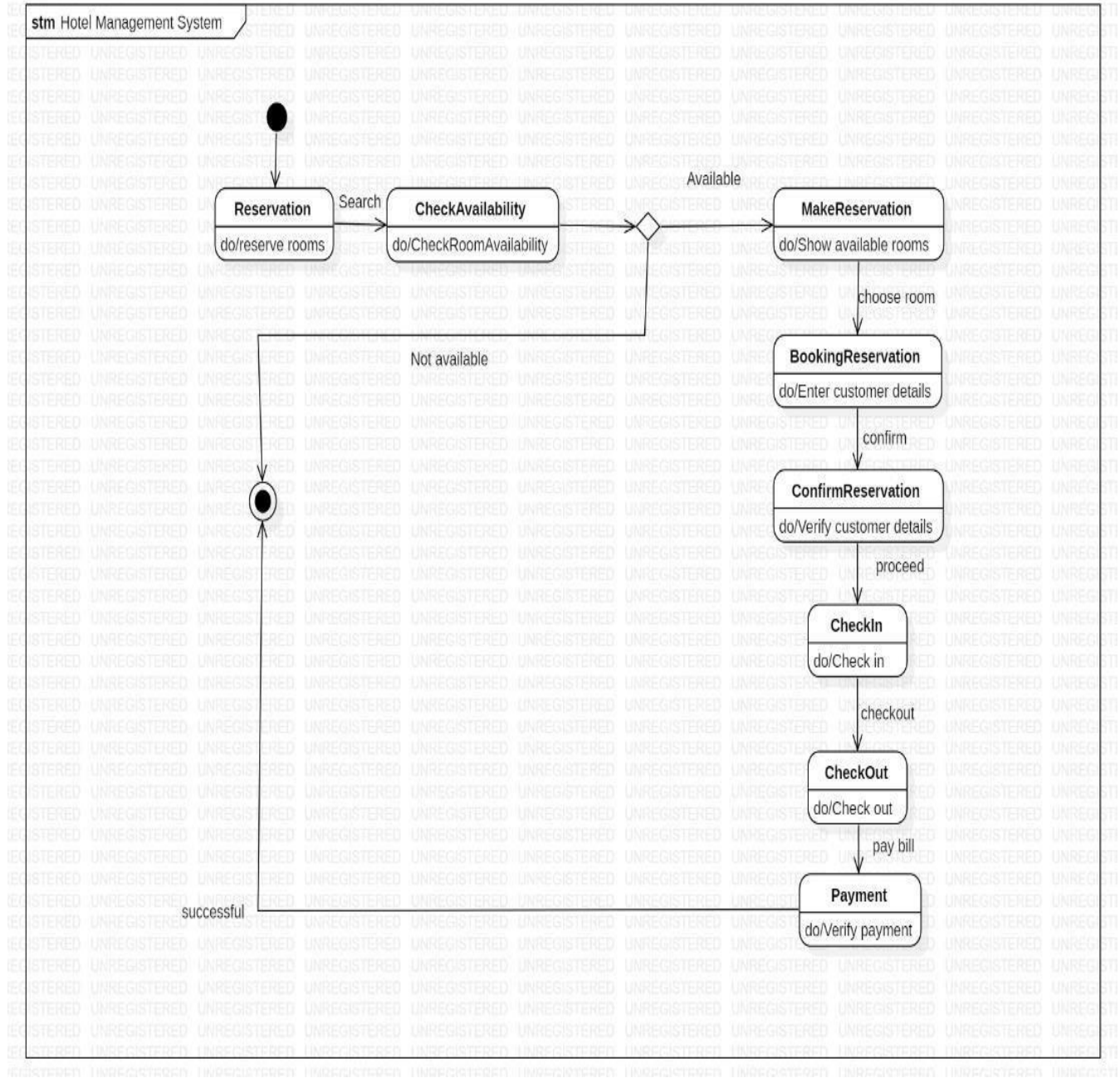


Figure 1.1 State Diagram of Hotel Management System

## USE-CASE DIAGRAM

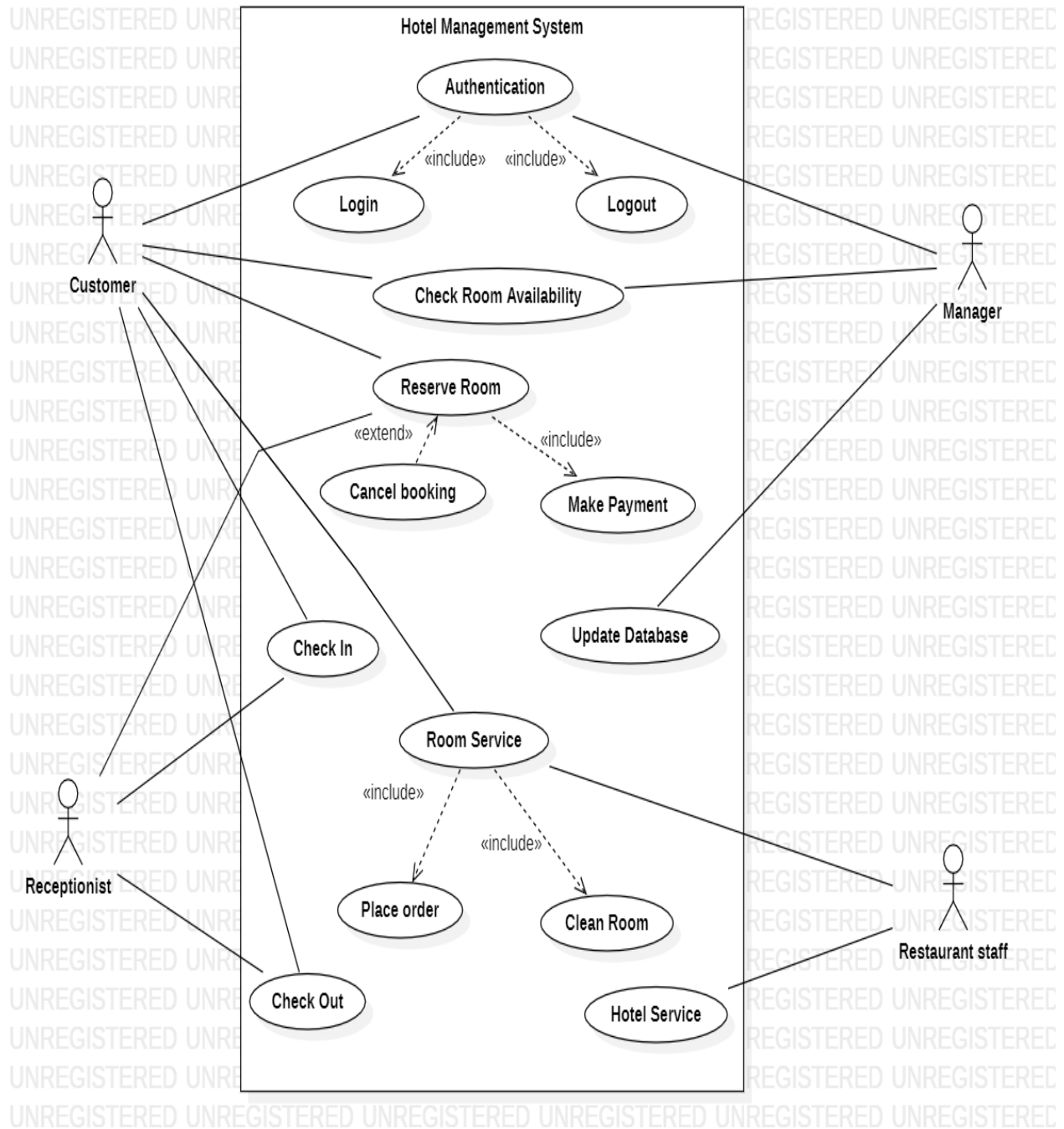


Figure 1.3 Usecase Diagram of Hotel Management System



## SEQUENCE DIAGRAM

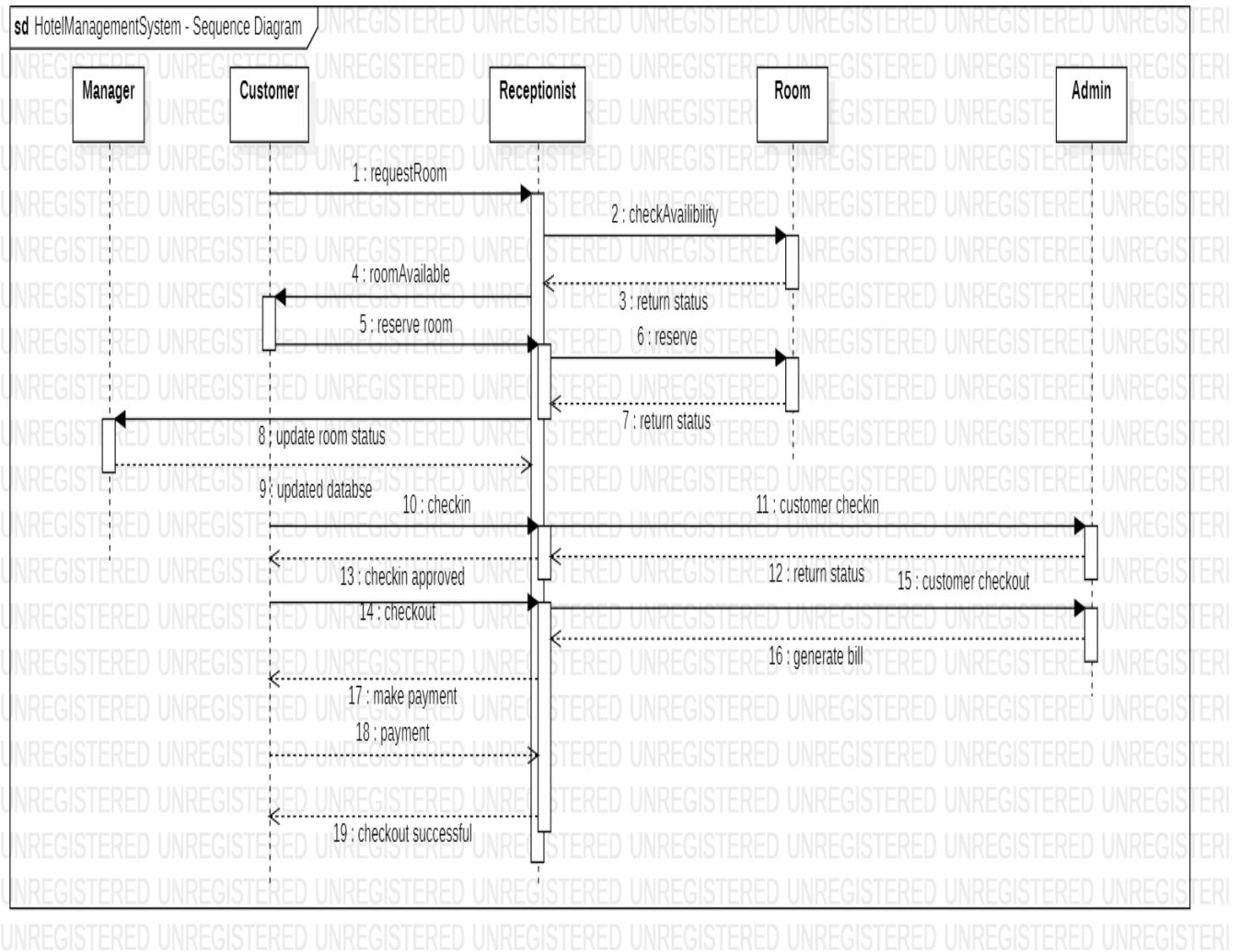


Figure 1.4 Sequence Diagram of Hotel Management System

## ACTIVITY DIAGRAM

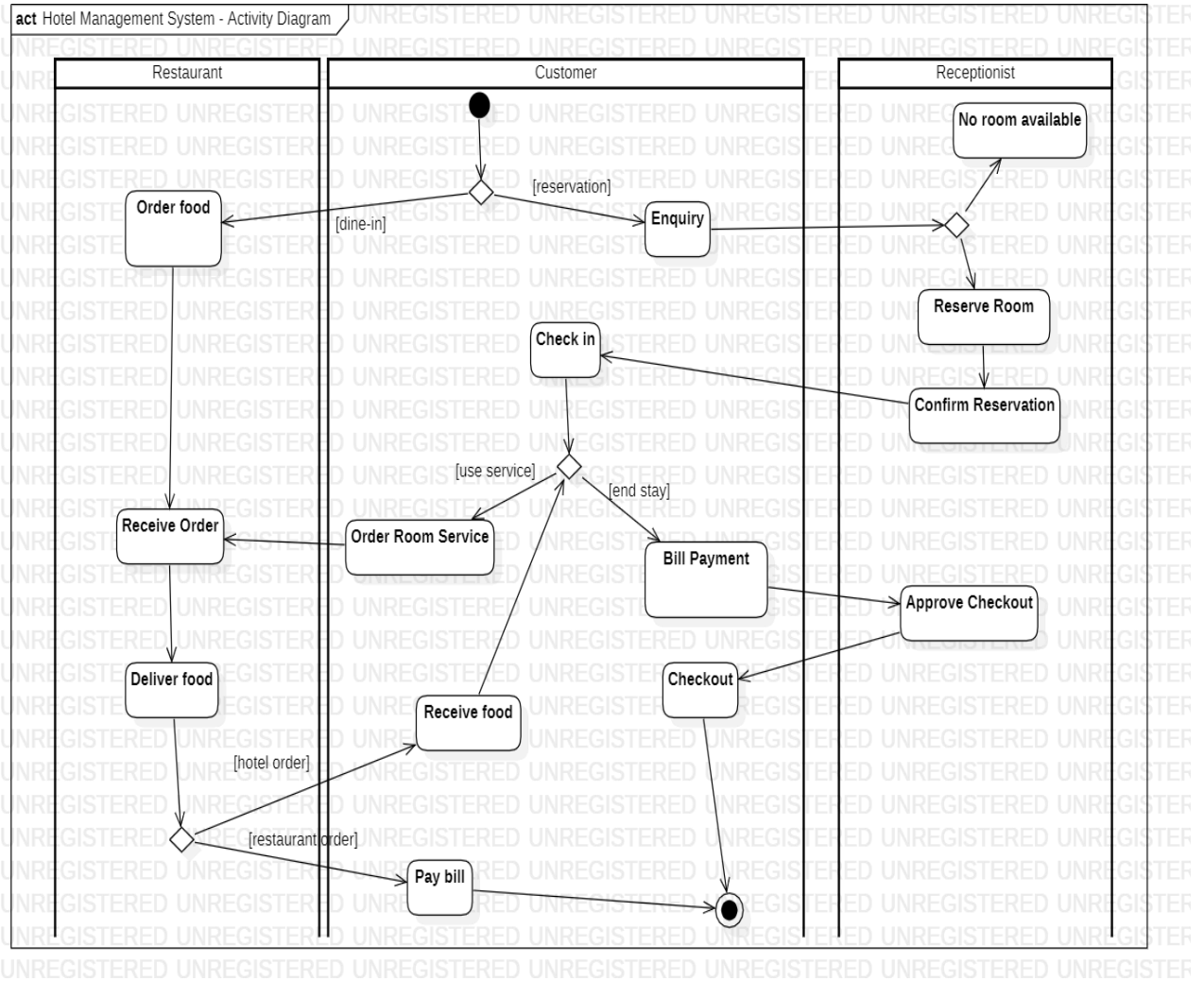


Figure 1.5 Activity Diagram of Hotel Management System

# **CREDIT CARD PROCESSING**

## **A. Problem Statement:**

The problem statement for the Credit Card Processing project is to develop a system that enables secure and efficient processing of credit card transactions for merchants. The system should ensure the confidentiality and integrity of cardholder information while providing a seamless payment experience for customers.

## **B. SRS (Software Requirements Specification):**

### **1. Introduction:**

#### **a. Purpose:**

The purpose of the Credit Card Processing system is to facilitate the secure processing of credit card transactions for merchants. The system should comply with industry standards and regulations to protect cardholder data and provide a reliable payment processing solution.

#### **b. Scope:**

The scope of the project includes developing a robust and scalable platform that supports various payment methods, ensures data encryption and secure transmission, integrates with payment gateways, and provides reporting and reconciliation capabilities for merchants.

#### **c. Overview:**

The Credit Card Processing system aims to simplify and streamline the payment process for merchants by securely handling credit card transactions, reducing the risk of fraud, and providing reliable transaction processing and reporting capabilities.

### **2. General Description:**

The Credit Card Processing system will be a software application that integrates with merchant websites or point-of-sale systems to handle credit card transactions. It will include features such as card data encryption, transaction authorization, payment gateway integration, fraud detection, and reporting capabilities.

### **3. Functional Requirements:**

- Securely capture and encrypt credit card data during transactions
- Verify cardholder information and perform transaction authorization
- Integrate with payment gateways to facilitate payment processing
- Support various payment methods (credit cards, debit cards, etc.)
- Provide real-time transaction status updates and notifications
- Generate reports for transaction history, settlements, and reconciliation

### **4. Interface Requirements:**

- Integration with merchant websites or point-of-sale systems for transaction initiation
- Integration with payment gateways or third-party processors for transaction processing
- API or web service interfaces for data exchange with external systems
- User interfaces for merchant administration, reporting, and configuration

### **5. Performance Requirements:**

- Fast and responsive transaction processing to minimize customer wait times
- High availability and reliability to ensure uninterrupted payment processing
- Scalability to handle a large volume of transactions during peak periods
- Secure data storage and transmission to protect cardholder information

### **6. Design Constraints:**

- Compliance with Payment Card Industry Data Security Standard (PCI DSS) requirements to protect cardholder data
- Integration with various payment processors or acquirers to support different card types and networks

### **7. Non-Functional Attributes:**

- Usability: The system should provide a user-friendly interface for merchants to manage transactions and access reports.
- Security: The system should implement robust security measures to protect sensitive cardholder data and prevent unauthorized access.
- Performance: The system should deliver fast and reliable transaction processing, with minimal downtime or system slowdowns.
- Scalability: The system should be able to handle an increasing number of transactions as the merchant's business grows.
- Reliability: The system should be highly available and provide failover mechanisms to ensure continuous payment processing.

# CLASS DIAGRAM

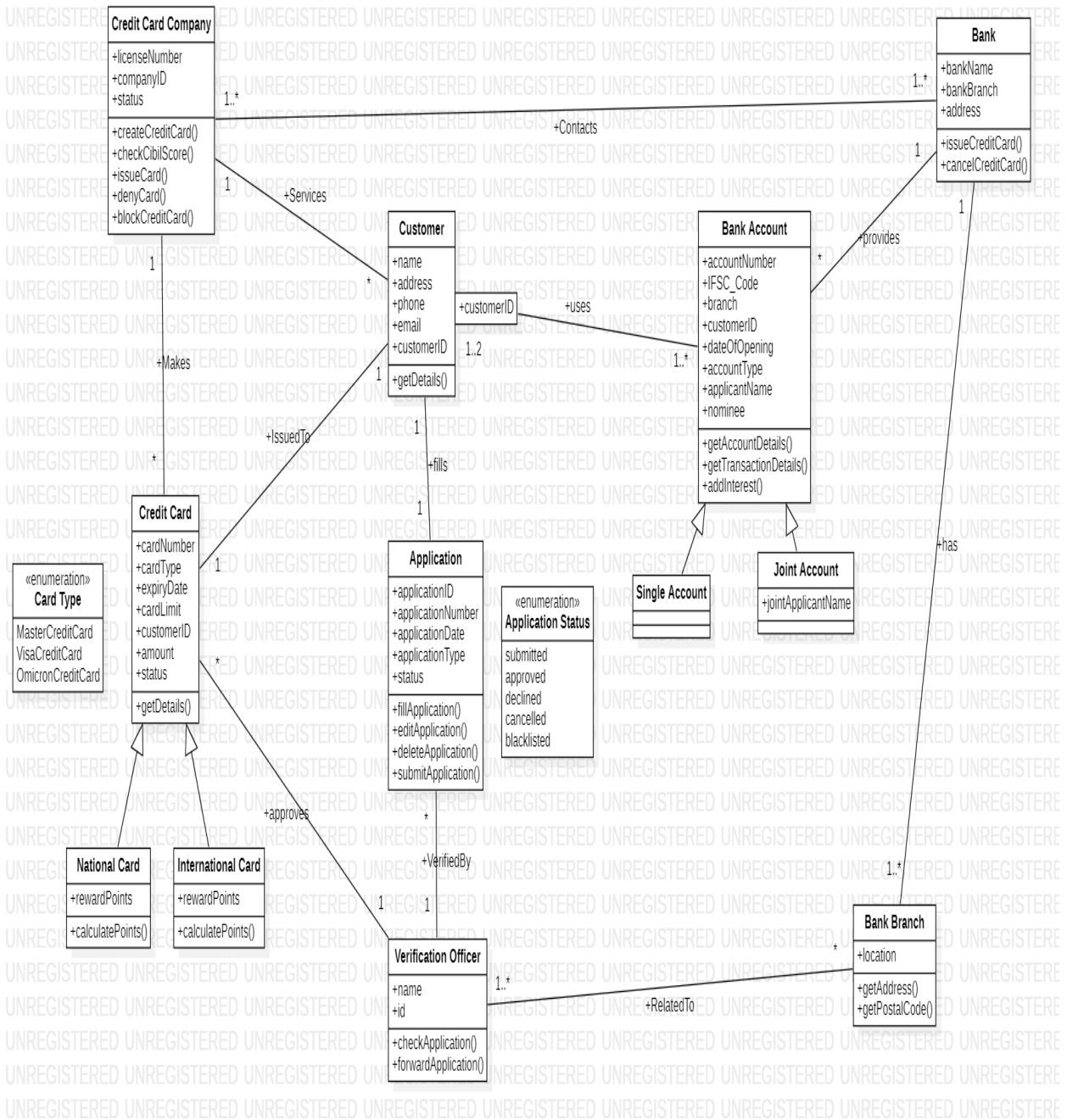


Figure 2.1 Class Diagram of Credit Card Processing System

## STATE DIAGRAM

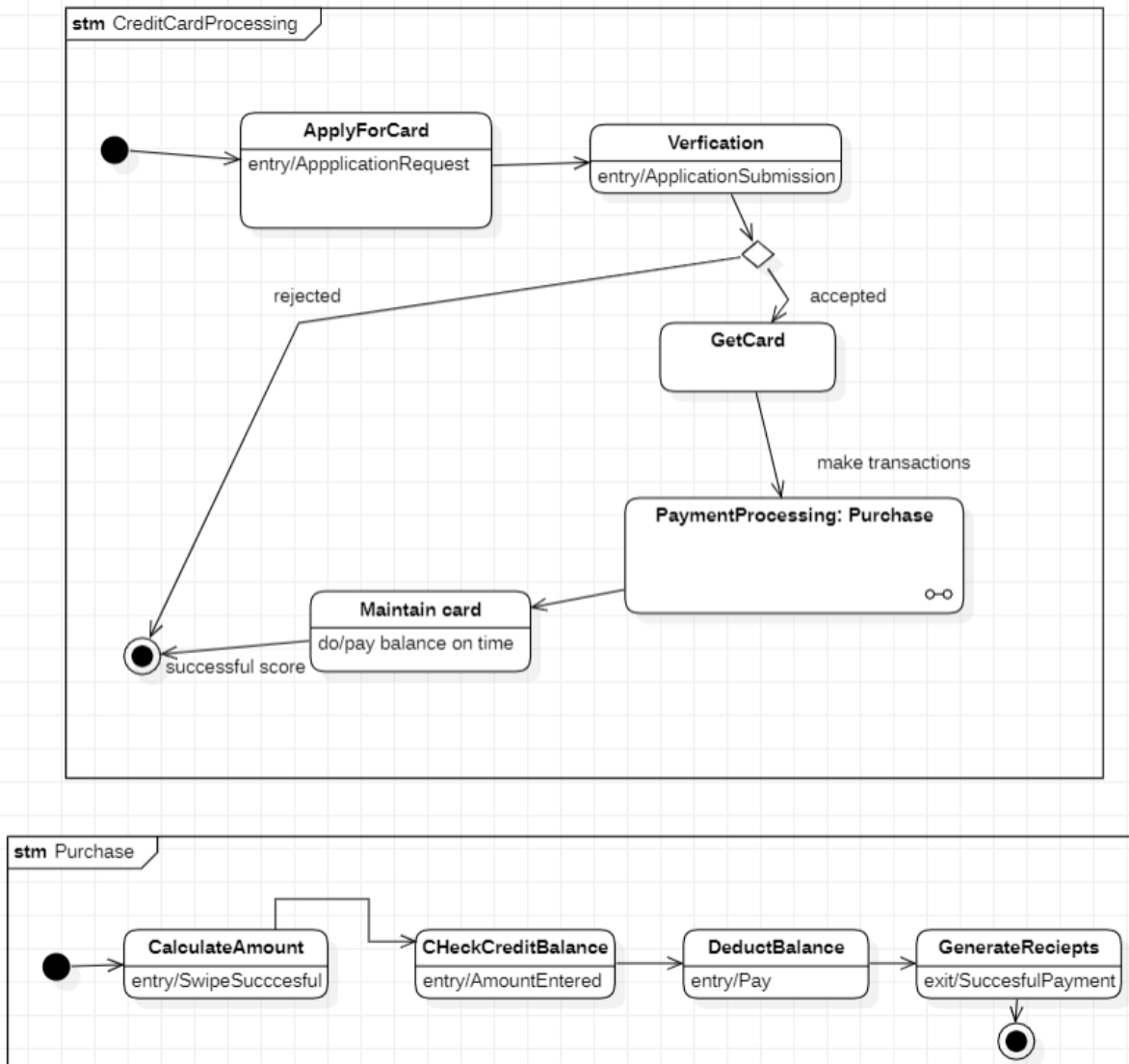


Figure 2.2 State Diagram of Credit Card Processing System

## USE-CASE DIAGRAM

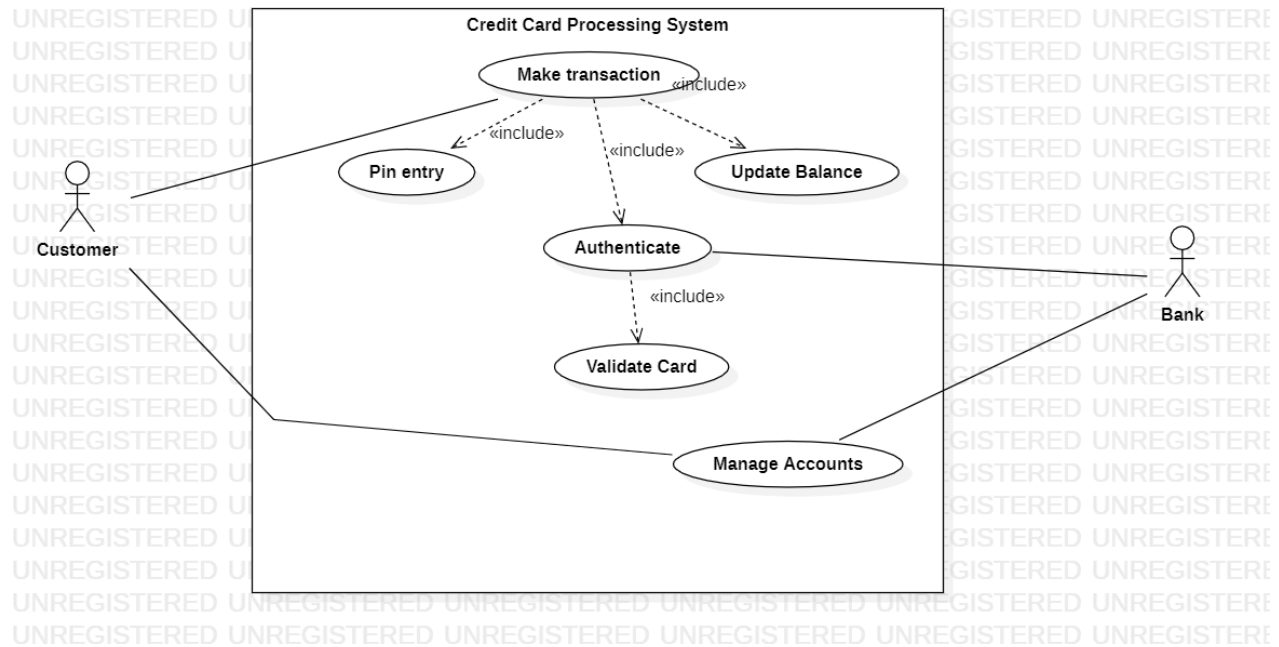


Figure 2.2 Usecase Diagram of Credit Card Processing System

## SEQUENCE DIAGRAM

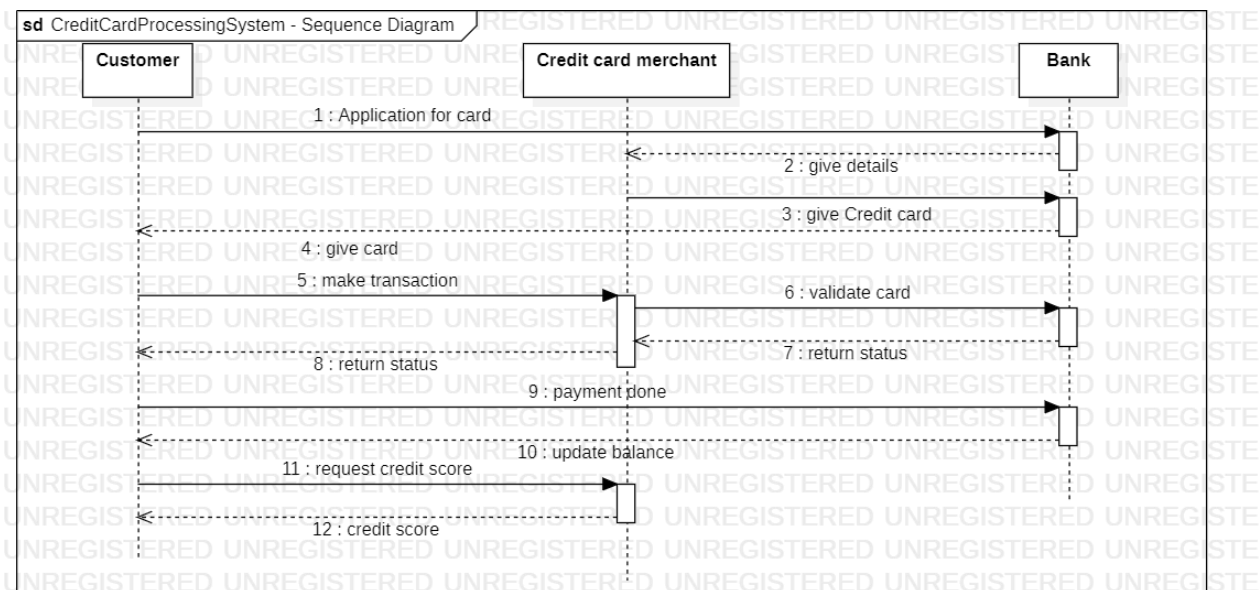


Figure 2.4 Sequence Diagram of Credit Card Processing System

## ACTIVITY DIAGRAM

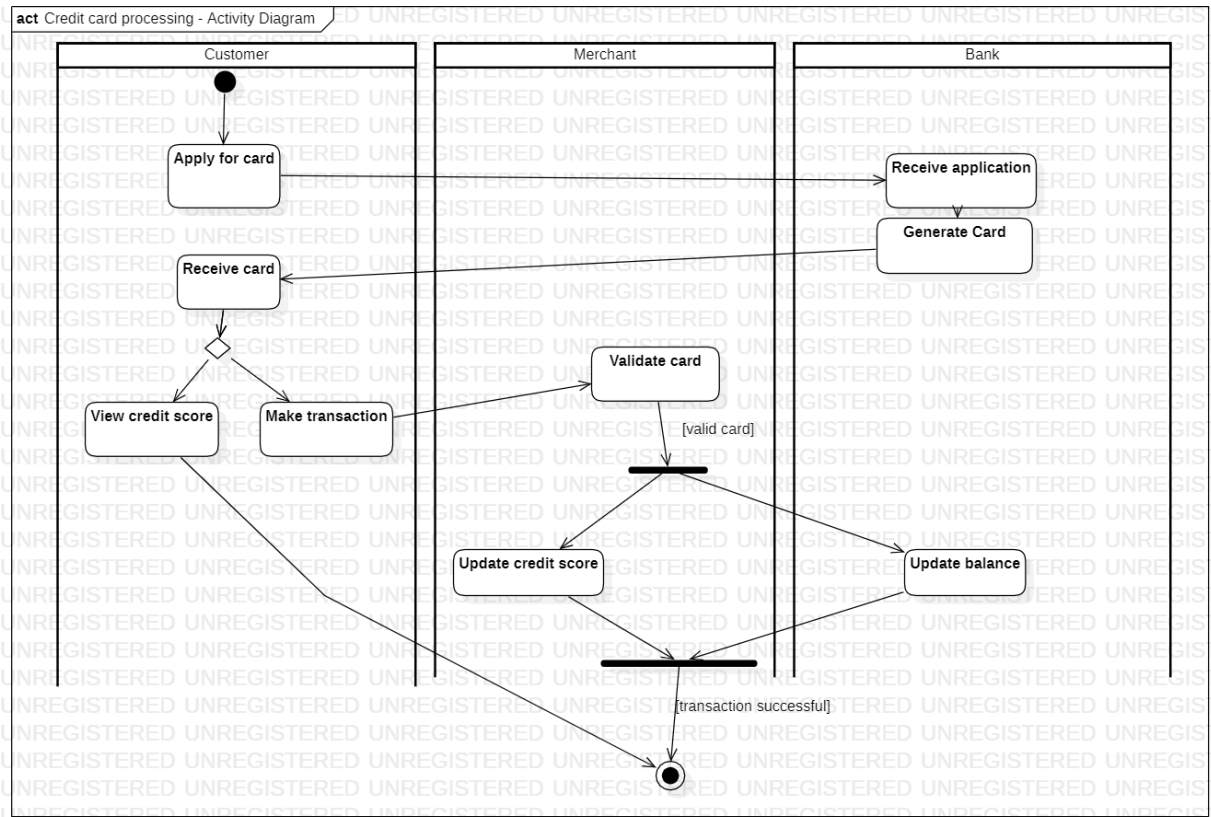


Figure 2.5 Activity Diagram of Credit Card Processing System



# **LIBRARY MANAGEMENT SYSTEM**

## **A. Problem Statement:**

The problem statement for the Library Management System project is to develop a software solution that automates and improves the management of library operations. The system should enable efficient cataloging and tracking of library resources, streamline lending and return processes, provide a user-friendly interface for library staff and patrons, and generate reports for administrative purposes.

## **B. SRS (Software Requirements Specification):**

### **1. Introduction:**

#### **a. Purpose:**

The purpose of the Library Management System is to automate and streamline library operations, including resource management, lending and return processes, and administrative tasks.

#### **b. Scope:**

The scope of the project includes developing a comprehensive software solution that encompasses features such as cataloging and indexing of library resources, patron management, circulation management, reporting and analytics, and administrative controls. The system should support multiple libraries within a network, if applicable.

#### **c. Overview:**

The Library Management System is designed to modernize and optimize library operations, making it easier for staff to manage resources, track borrowing activities, and generate reports.

### **2. General Description:**

The Library Management System will be a web-based application that can be accessed by library staff and patrons. It will include modules for cataloging and indexing library resources, managing patron records, handling lending and return processes, generating reports and statistics, and providing administrative controls.

### **3. Functional Requirements:**

- Cataloging and indexing of library resources, including books, periodicals, and multimedia items
- Patron management, including registration, record keeping, and membership management
- Lending and return processes, including check-in/check-out, due date management, and fine calculation
- Reservation and hold management for popular or limited resources
- Reporting and analytics capabilities for generating circulation reports, inventory reports, and financial reports
- Administrative controls for managing system settings, user roles, and permissions

### **4. Interface Requirements:**

- Intuitive and user-friendly interface for library staff and patrons
- Responsive design to support different devices and screen sizes
- APIs or data exchange capabilities for integrating with other library systems or databases

### **5. Performance Requirements:**

- Fast and responsive system performance to handle simultaneous user interactions and transactions
- Scalability to accommodate a growing number of library resources and patrons
- Reliable data storage and retrieval to prevent data loss or corruption
- Efficient search and retrieval of library resources to minimize response times

### **6. Design Constraints:**

- Integration with existing library systems or technologies (if applicable)
- Compatibility with popular web browsers and operating systems

### **7. Non-Functional Attributes:**

- Usability: The system should provide an intuitive and easy-to-use interface for library staff and patrons.
- Reliability: The system should be reliable and available 24/7 to support library operations without interruptions.
- Security: The system should ensure the confidentiality and integrity of patron data and library resources.
- Maintainability: The system should be easy to maintain and update with minimal downtime.
-

# CLASS DIAGRAM

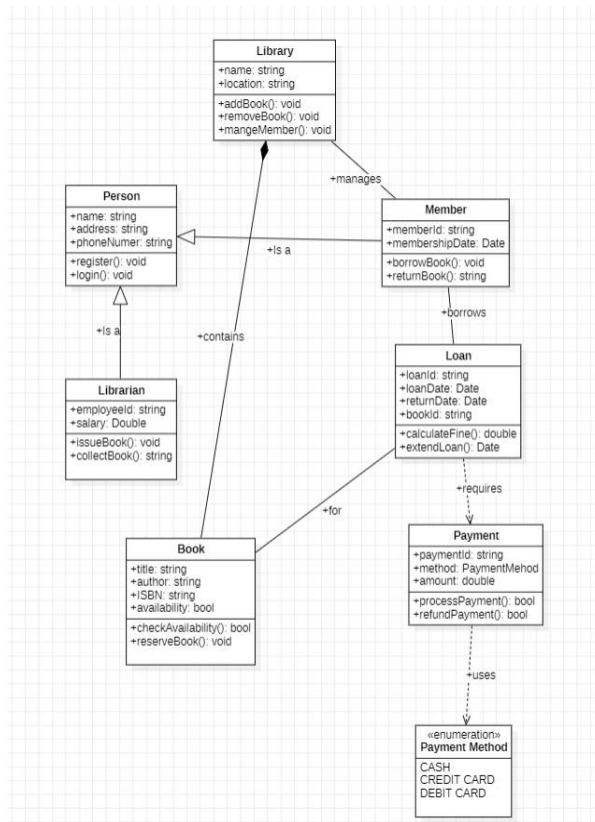


Figure 3.1 Class Diagram of Library Management System

## STATE DIAGRAM

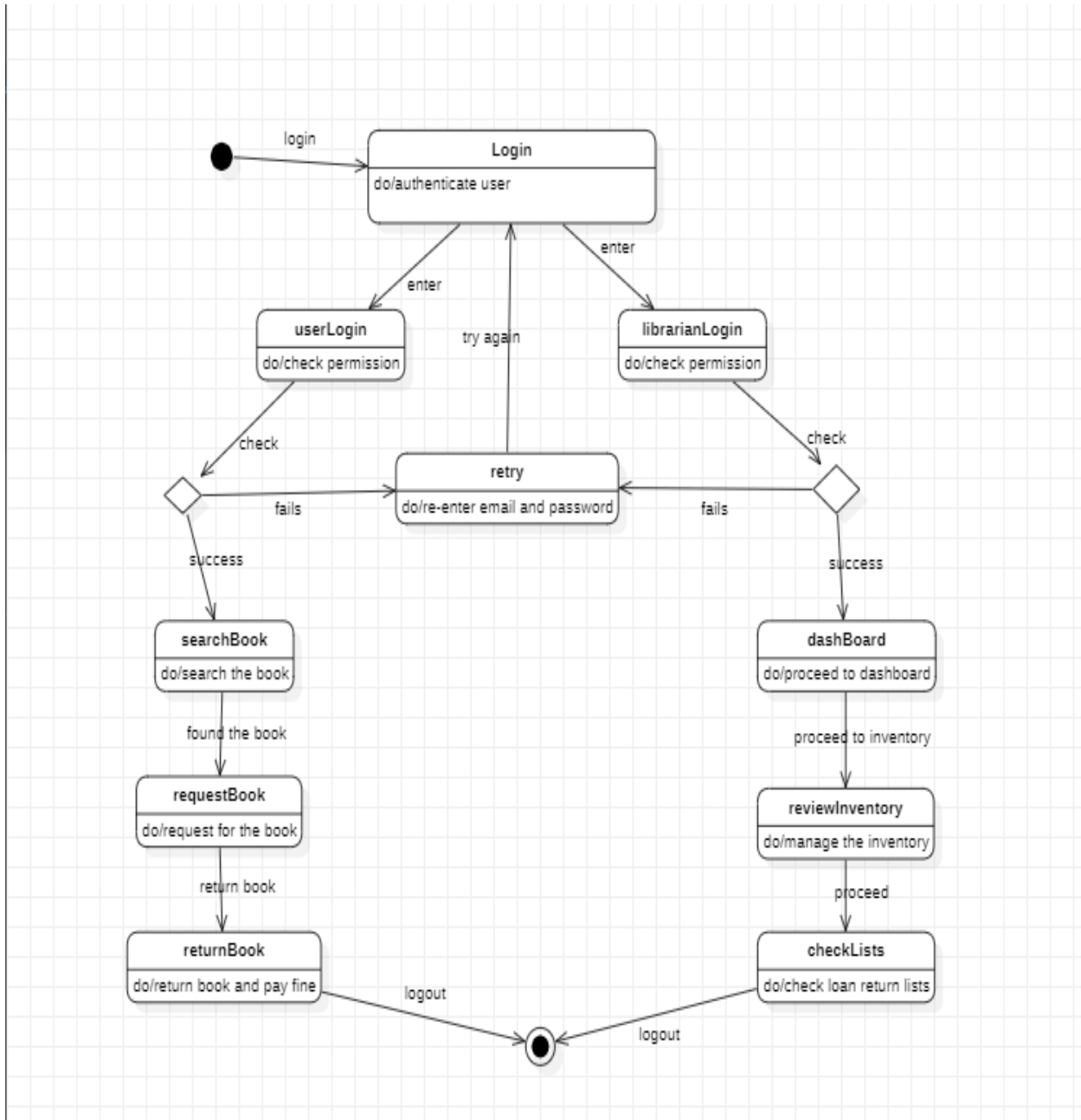


Figure 3.2 State Diagram of Library Management System

## USE-CASE DIAGRAM

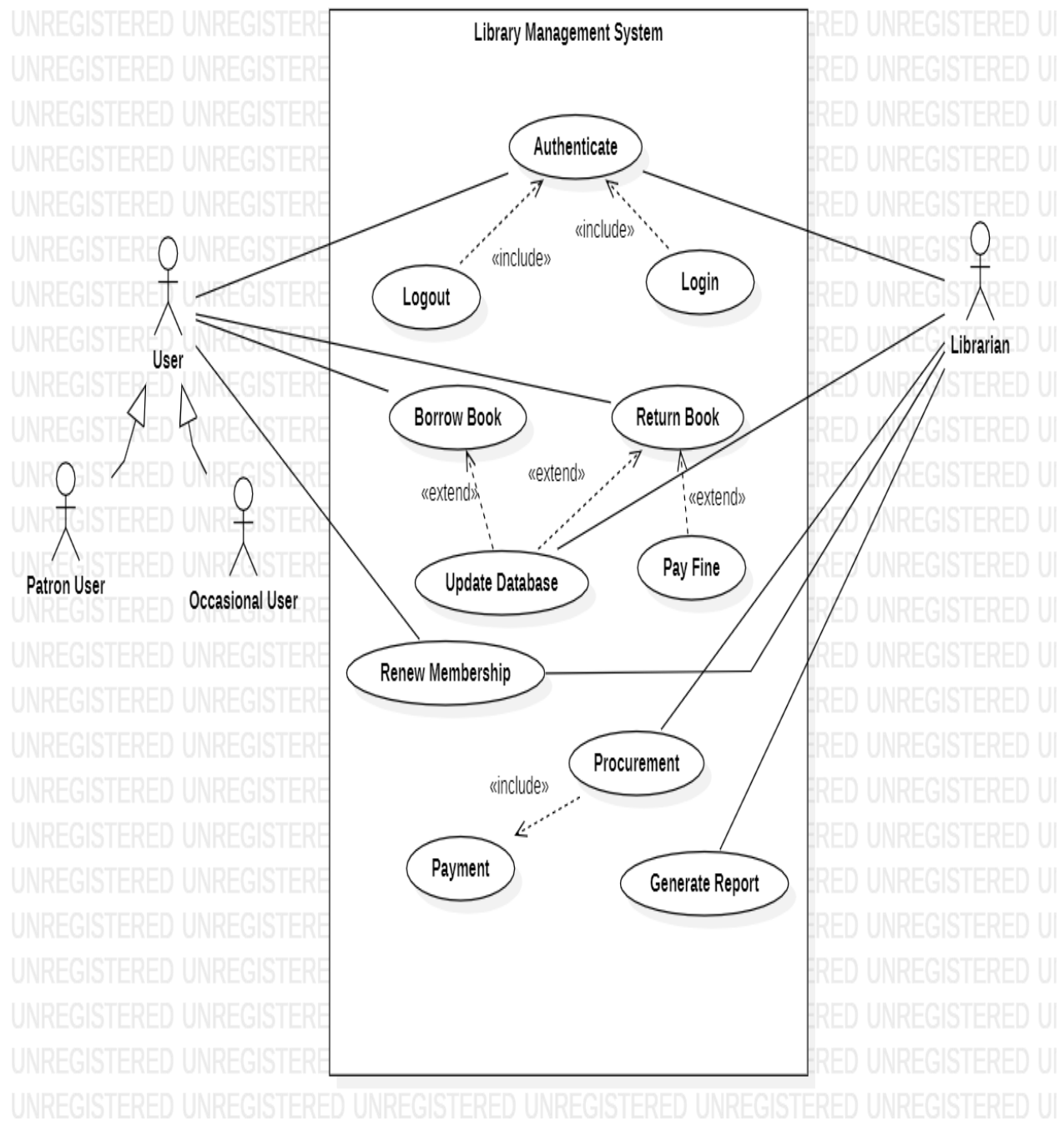
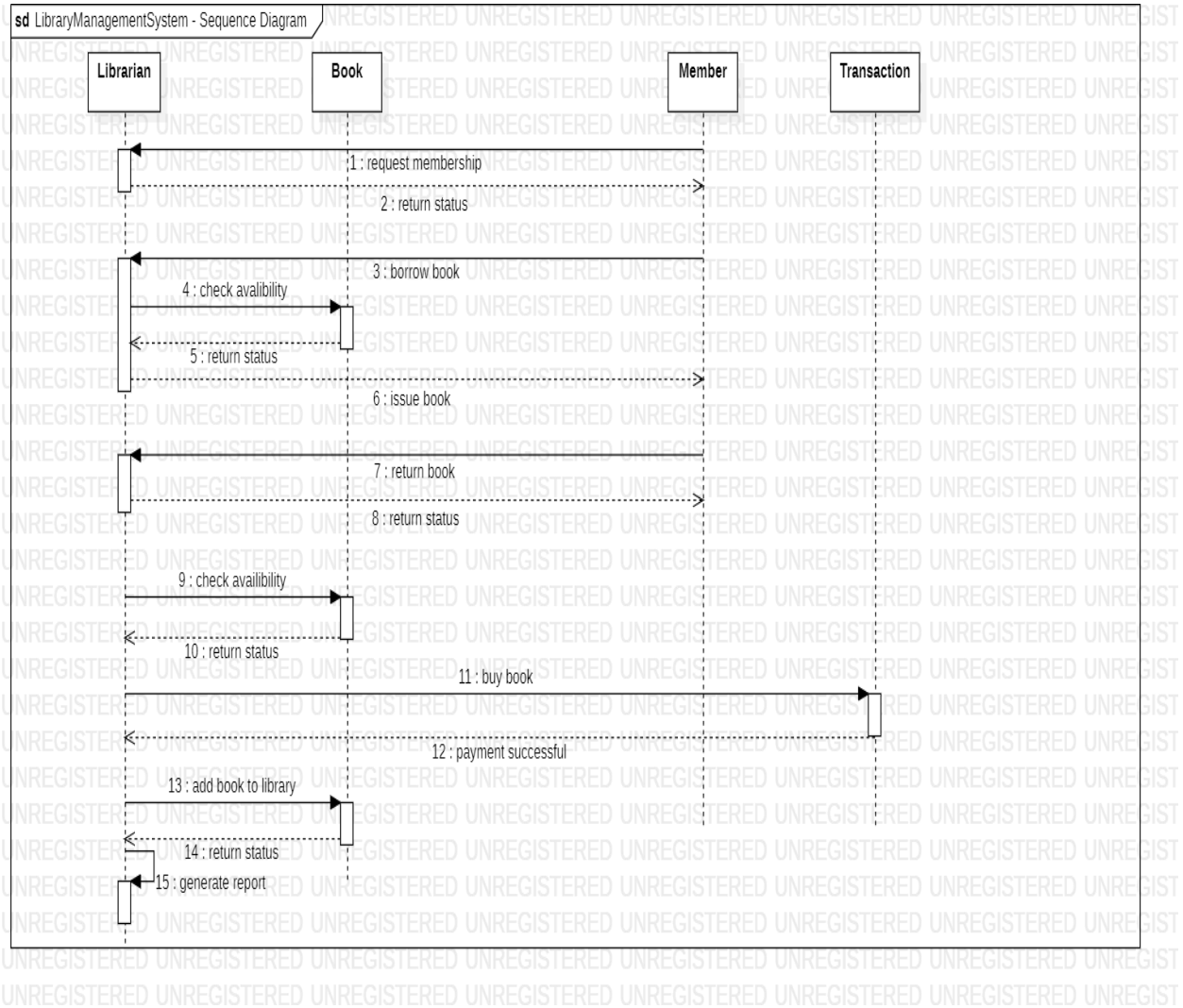


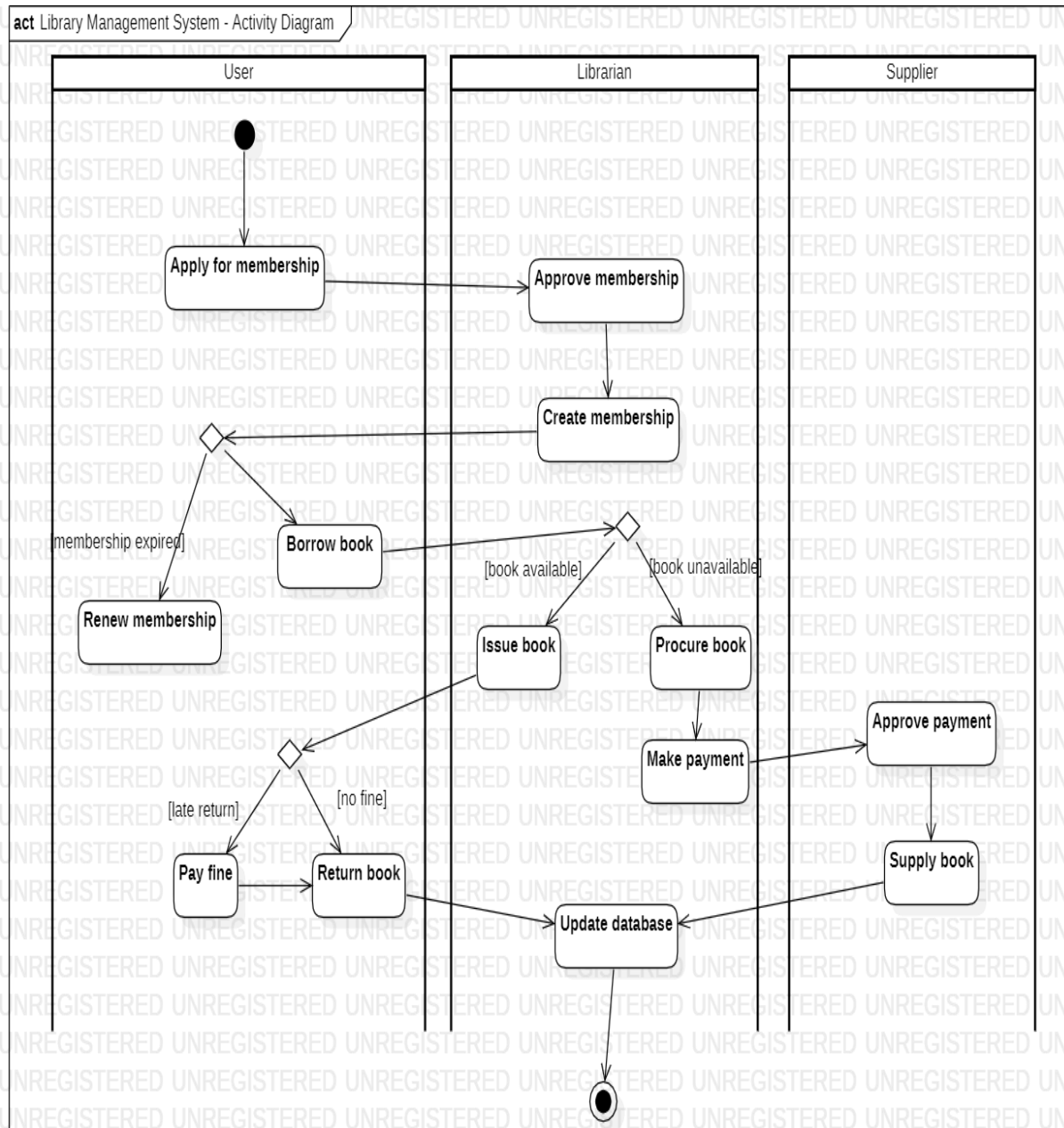
Figure 3.3 Usecase Diagram of Library Management System

## SEQUENCE DIAGRAM



*Figure 3.4 Sequence Diagram of Library Management System*

## ACTIVITY DIAGRAM



*Figure 3.5 Activity Diagram of Library Management System*

# **STOCK MAINTAINANCE SYSTEM**

## **A. Problem Statement:**

The problem statement for the Stock Maintenance System project is to develop a software solution that enables efficient management and tracking of stock inventory for businesses. The system should provide real-time visibility of stock levels, automate inventory replenishment processes, facilitate stock movement tracking, and generate reports for analysis and decision-making.

## **B. SRS (Software Requirements Specification):**

### **1. Introduction:**

#### **a. Purpose:**

The purpose of the Stock Maintenance System is to streamline the management and tracking of stock inventory for businesses. The system aims to provide accurate and up-to-date information on stock levels, automate stock replenishment processes, improve stock movement tracking, and generate reports for inventory analysis and decision-making.

#### **b. Scope:**

The scope of the project includes developing a comprehensive software solution that encompasses features such as stock item management, stock movement tracking, stock replenishment automation, reporting and analytics, and administrative controls. The system should cater to businesses of varying sizes and industries.

#### **c. Overview:**

The Stock Maintenance System is designed to optimize stock inventory management, enabling businesses to efficiently track stock levels, streamline stock replenishment processes, and make informed decisions based on accurate inventory data.

### **2. General Description:**

The Stock Maintenance System will be a web-based or desktop application that can be accessed by authorized personnel within the business. It will include modules for stock item management, stock movement tracking, stock replenishment automation, reporting and analytics, and administrative controls.



### **3. Functional Requirements:**

- Stock item management, including creating, updating, and deleting stock items
- Stock movement tracking, recording stock inflows and outflows, and capturing relevant details such as source, destination, and quantity
- Stock replenishment automation, generating purchase orders or requisitions based on predefined reorder levels or thresholds
- Stock adjustment functionality, allowing for manual adjustments to stock quantities, such as write-offs or corrections
- Stock location management, organizing stock items based on different locations, bins, or warehouses
- Reporting and analytics capabilities, generating reports on stock levels, stock movement history, stock valuation, and stock turnover
- Administrative controls for managing system settings, user roles, and permissions

### **4. Interface Requirements:**

- Intuitive and user-friendly interface for authorized personnel to manage stock inventory
- Responsive design to support different devices and screen sizes
- Integration with barcode scanners or other automated identification technologies for efficient stock tracking
- APIs or data exchange capabilities for integrating with other business systems such as accounting or purchasing systems

### **5. Performance Requirements:**

- Fast and responsive system performance to handle simultaneous user interactions and stock transactions
- Scalability to accommodate a growing number of stock items and business operations
- Reliable data storage and retrieval to prevent data loss or corruption
- Efficient stock search and retrieval functionality to minimize response times

### **6. Design Constraints:**

- Compliance with security standards and regulations to protect sensitive stock and business information
- Integration with existing business systems or technologies (if applicable)
- Compatibility with popular web browsers and operating systems

## **7. Non-Functional Attributes:**

- Usability: The system should provide an intuitive and user-friendly interface for authorized personnel to manage stock inventory efficiently.
- Reliability: The system should be reliable and available 24/7 to support business operations without interruptions.
- Security: The system should ensure the confidentiality and integrity of stock and business data.
- Performance: The system should deliver fast response times and handle concurrent user requests efficiently.
- Maintainability: The system should be easy to maintain and update with minimal downtime.

## CLASS DIAGRAM

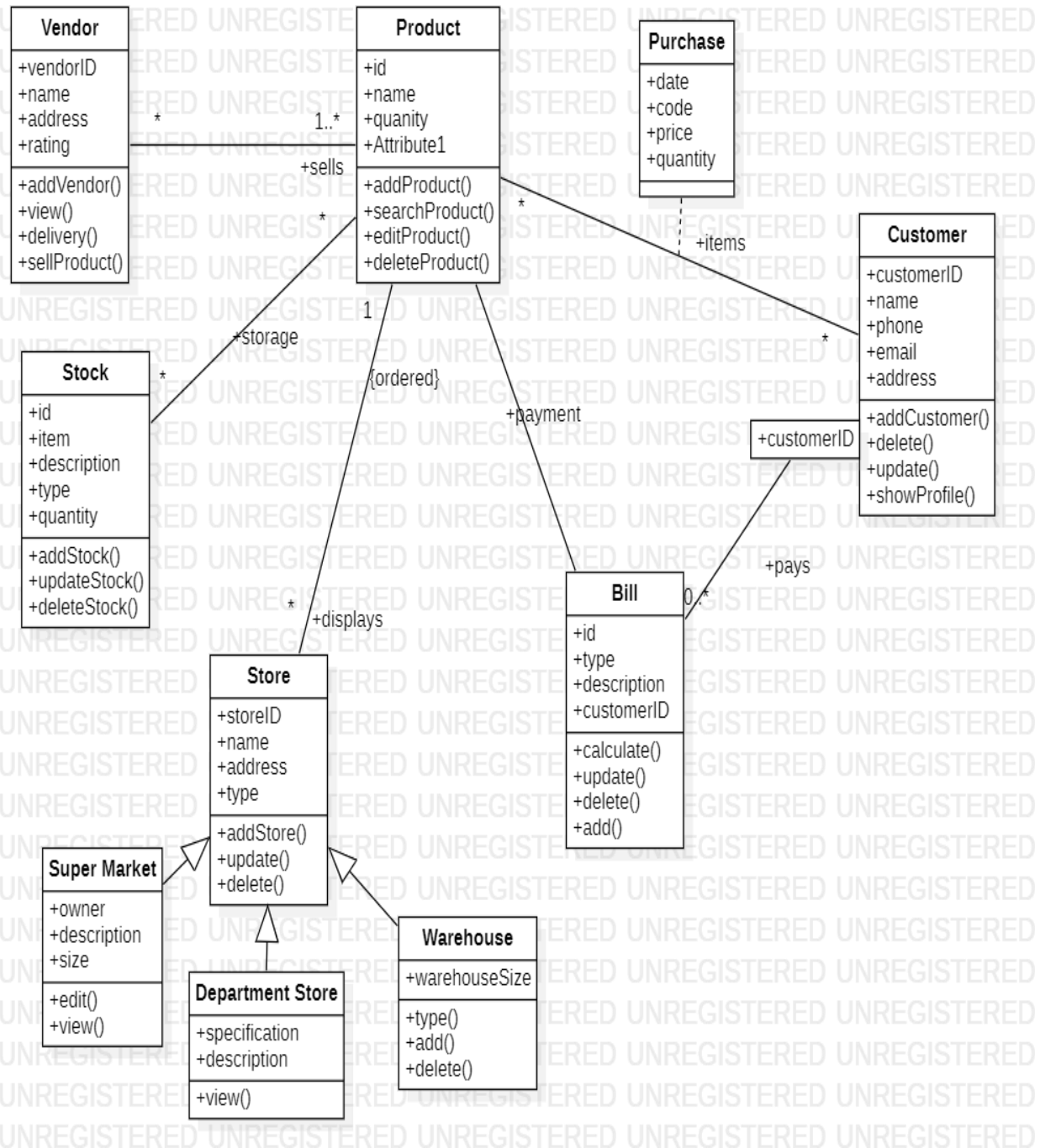


Figure 4.1 Class Diagram of Stock Maintenance System

## STATE DIAGRAM

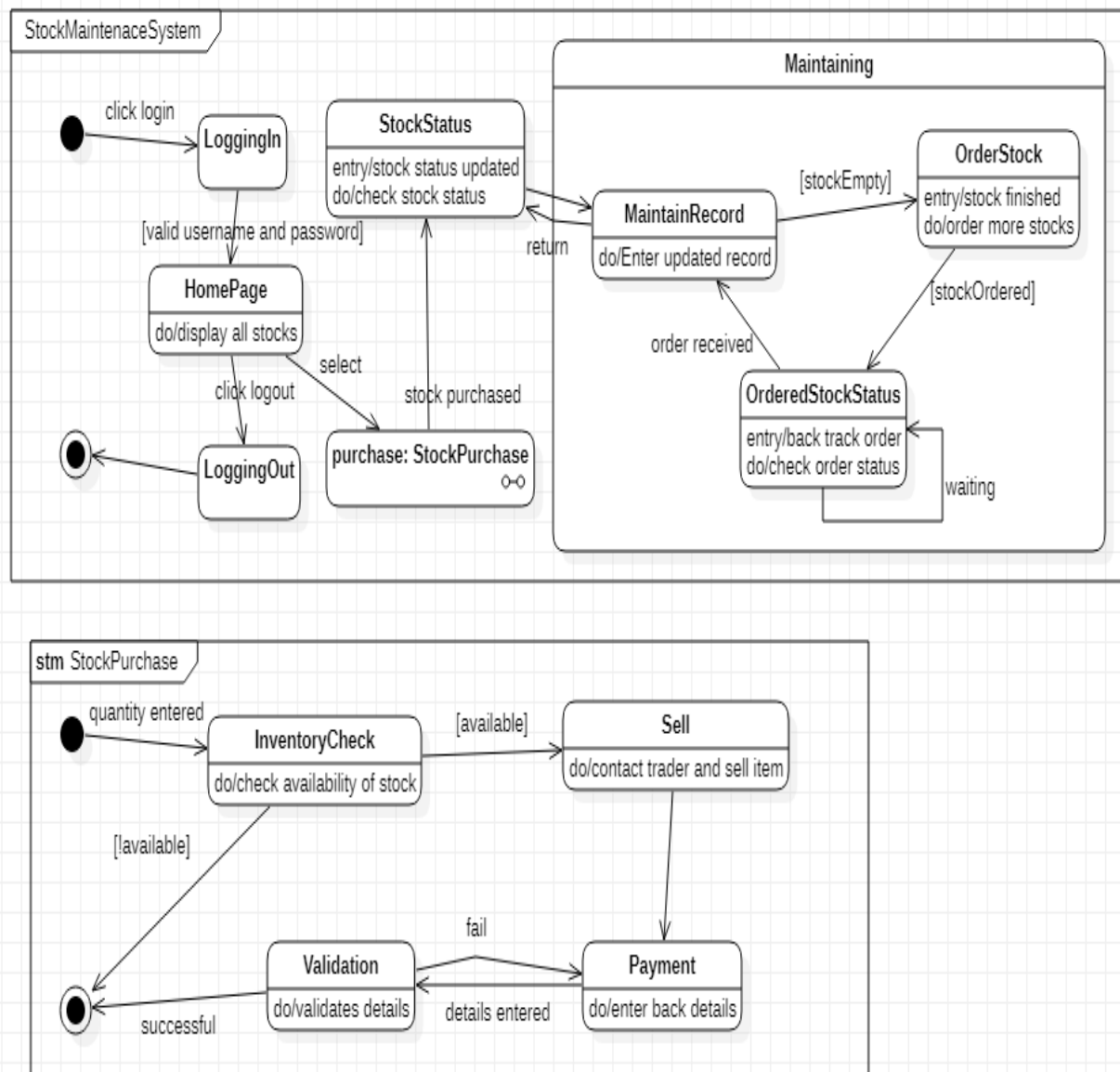


Figure 4.2 State Diagram of Stock Maintenance System

## USE-CASE DIAGRAM

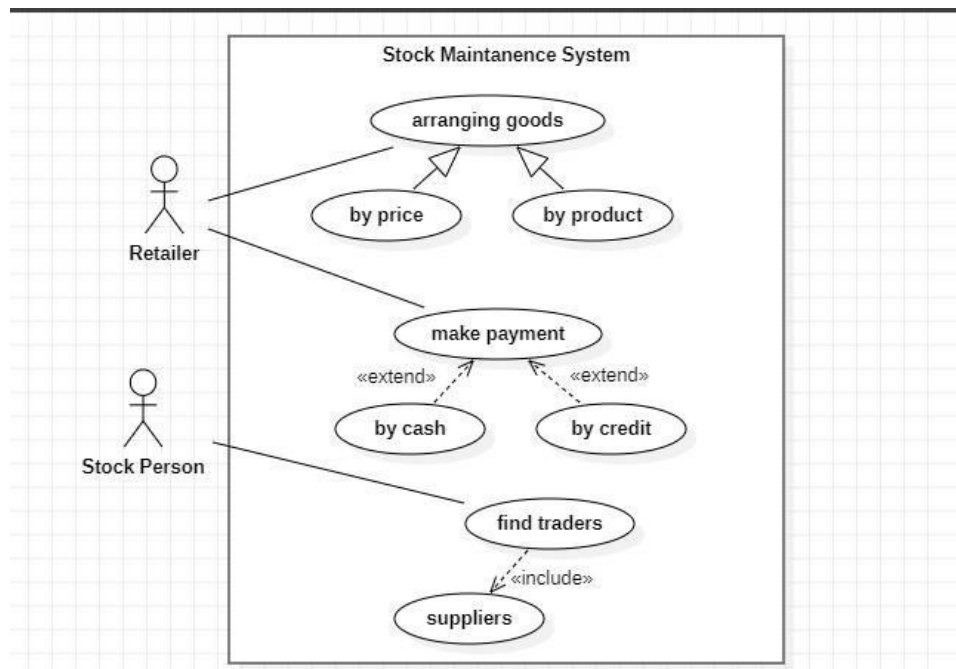


Figure 4.3 Use case Diagram of Stock Maintenance System

## SEQUENCE DIAGRAM

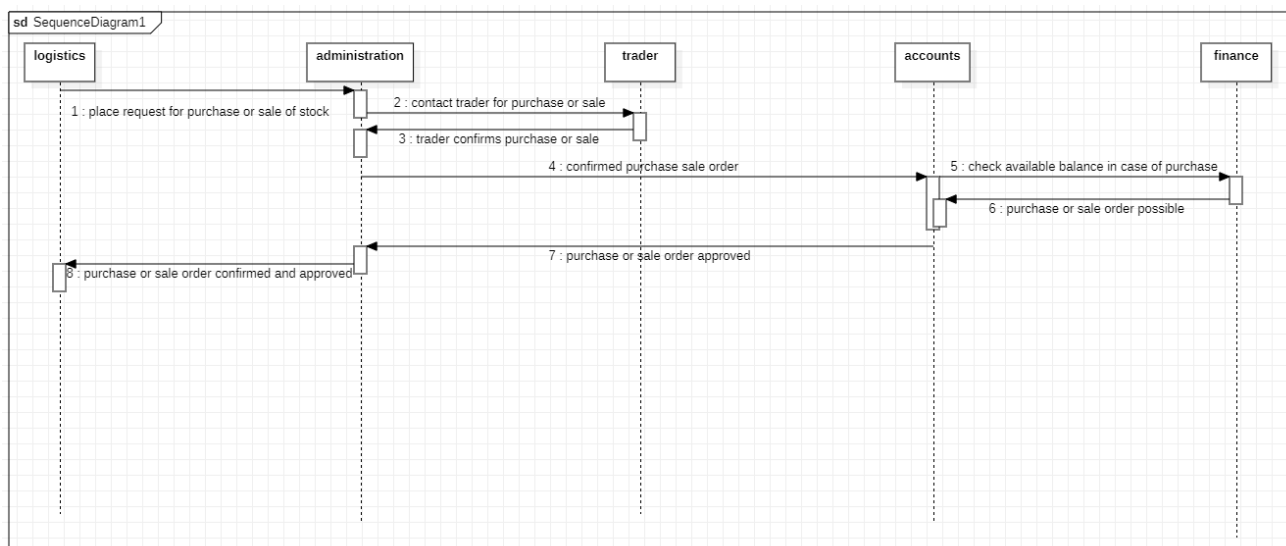


Figure 4.4 Sequence Diagram of Stock Maintenance System

## ACTIVITY DIAGRAM

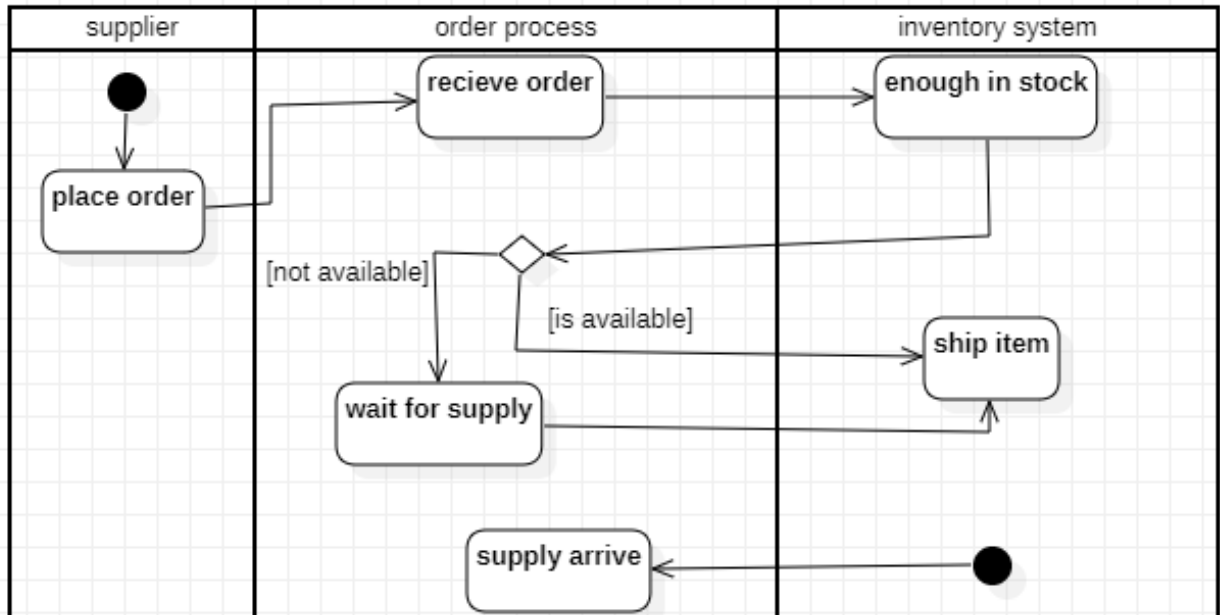


Figure 4.5 Activity Diagram of Stock Maintenance System

# **PASSPORT AUTOMATION SYSTEM**

## **A. Problem Statement:**

The problem statement for the Passport Automation System project is to develop a software solution that automates and streamlines the process of passport issuance and management. The system should facilitate efficient application processing, document verification, appointment scheduling, and tracking of passport applications, while ensuring data security and accuracy.

## **B. SRS (Software Requirements Specification):**

### **1. Introduction:**

#### **a. Purpose:**

The purpose of the Passport Automation System is to automate and streamline the process of passport issuance and management. The system aims to provide a user-friendly platform for applicants, improve the efficiency of application processing and document verification, enable online appointment scheduling, and ensure secure and accurate management of passport applications.

#### **b. Scope:**

The scope of the project includes developing a comprehensive software solution that encompasses features such as applicant registration, application processing, document verification, appointment scheduling, fee collection, passport printing, and administrative controls. The system should cater to both individual applicants and passport office staff.

#### **c. Overview:**

The Passport Automation System is designed to enhance the passport issuance process by automating manual tasks, reducing processing time, and improving data accuracy. It aims to provide a seamless and efficient experience for passport applicants and streamline operations for passport office staff.

### **2. General Description:**

The Passport Automation System will be a web-based application that can be accessed by both passport office staff and applicants. It will include modules for applicant registration, application processing, document verification, appointment scheduling, fee collection, passport printing, and administrative controls.

### **3. Functional Requirements:**

- Applicant registration and profile management
- Online application submission and tracking
- Document verification and validation
- Appointment scheduling for passport application submission and collection
- Fee calculation and online payment processing
- Passport printing and dispatch management
- Reporting and analytics capabilities for monitoring application status and performance
- Administrative controls for managing system settings, user roles, and permissions

### **4. Interface Requirements:**

- User-friendly interfaces for applicants to register, submit applications, track progress, and schedule appointments
- Intuitive interfaces for passport office staff to process applications, verify documents, schedule appointments, and manage passport printing and dispatch
- Integration with payment gateways for secure online fee collection
- APIs or data exchange capabilities for integration with external systems such as document verification databases or postal services

### **5. Performance Requirements:**

- Fast and responsive system performance to handle simultaneous user interactions and application processing
- Scalability to accommodate a large volume of passport applications during peak periods
- Reliable data storage and retrieval to prevent data loss or corruption
- Secure data transmission and storage to protect applicant information

### **6. Design Constraints:**

- Compliance with security standards and regulations to protect sensitive applicant data
- Integration with existing government systems or databases (if applicable)
- Compatibility with popular web browsers and operating systems

### **7. Non-Functional Attributes:**

- Usability: The system should provide an intuitive and user-friendly interface for both applicants and passport office staff.
- Reliability: The system should be reliable and available 24/7 to support passport issuance operations without interruptions.



- Security: The system should ensure the confidentiality and integrity of applicant data and document verification processes.
- Performance: The system should deliver fast response times and handle concurrent user requests efficiently.
- Maintainability: The system should be easy to maintain and update with minimal downtime.

## CLASS DIAGRAM

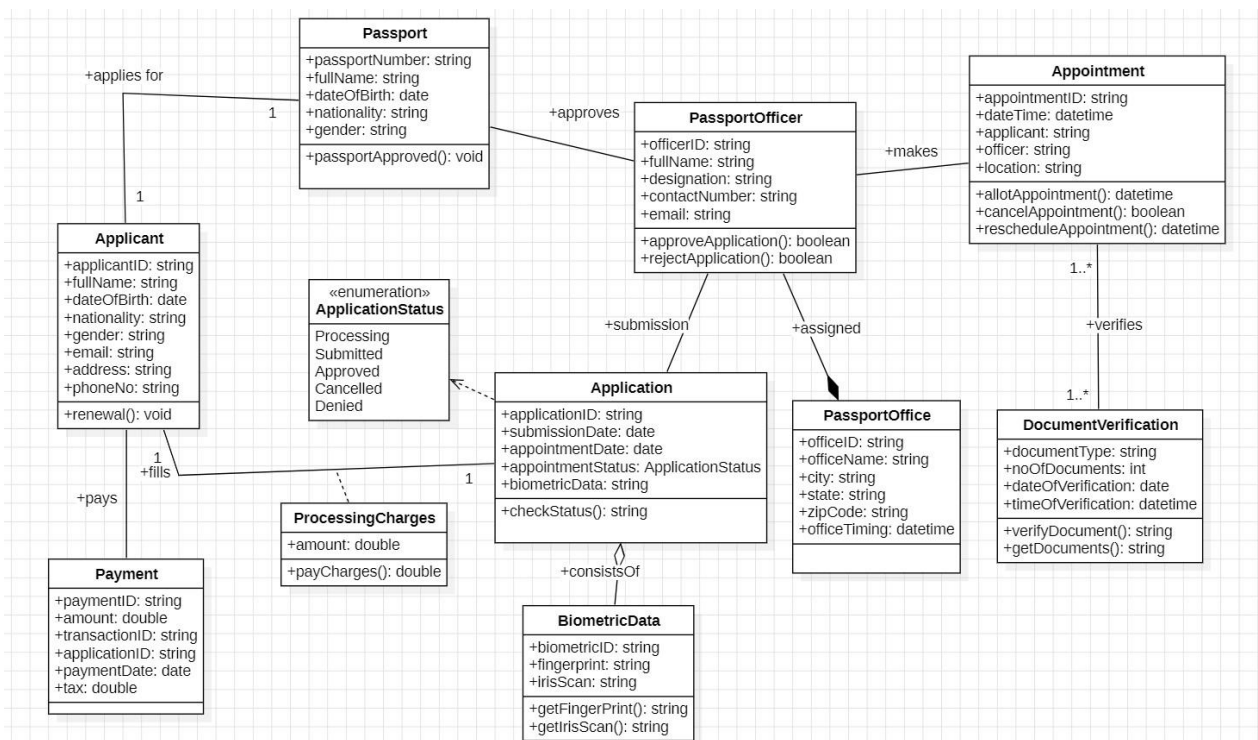


Figure 5.1 Class Diagram of Passport Automation System

## STATE DIAGRAM

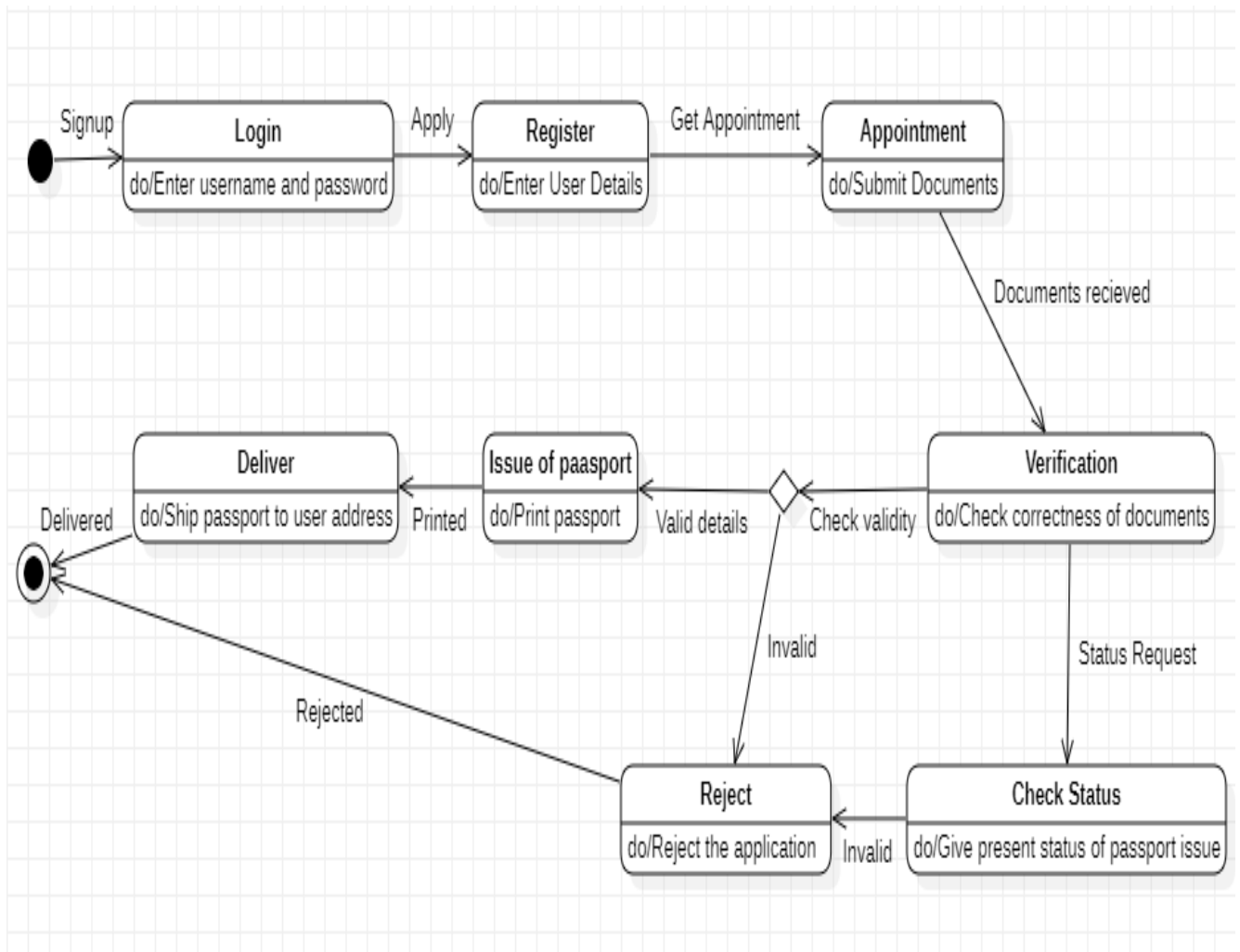


Figure 5.2 State Diagram of Passport Automation System

## USE-CASE DIAGRAM

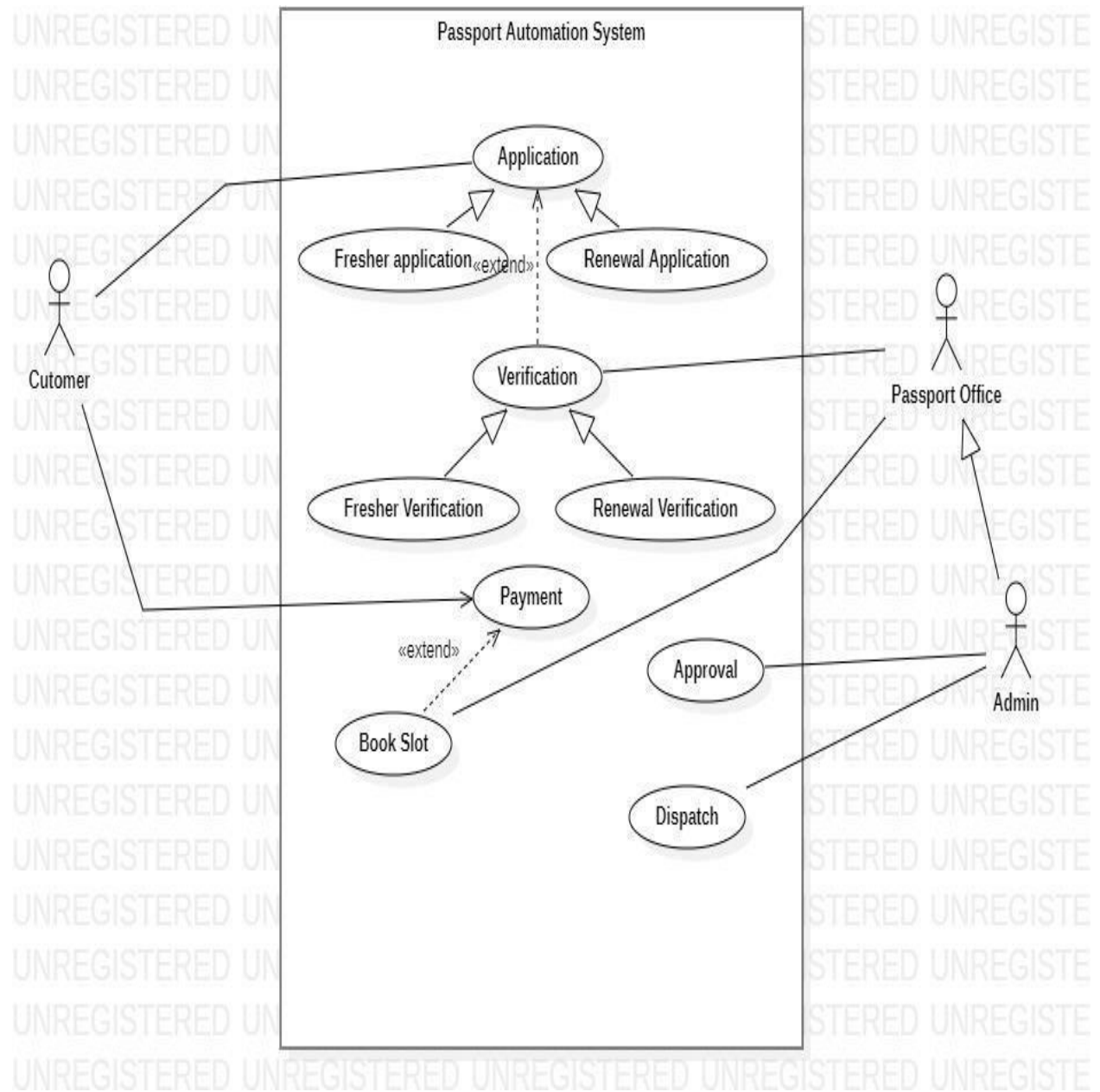


Figure 5.3 UseCase Diagram of Passport Automation System

## SEQUENCE DIAGRAM

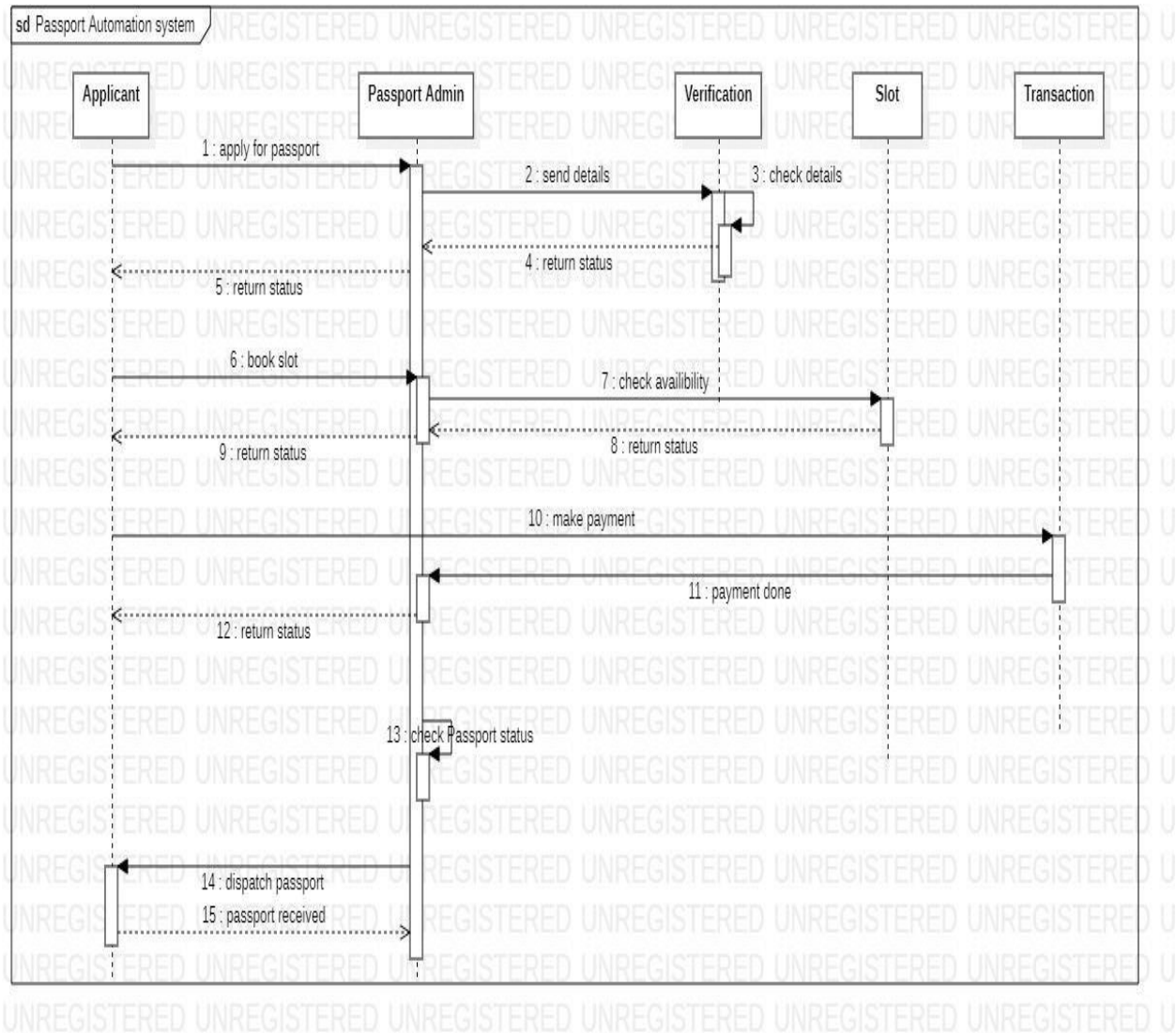


Figure 5.4 Sequence Diagram of Passport Automation System

## ACTIVITY DIAGRAM

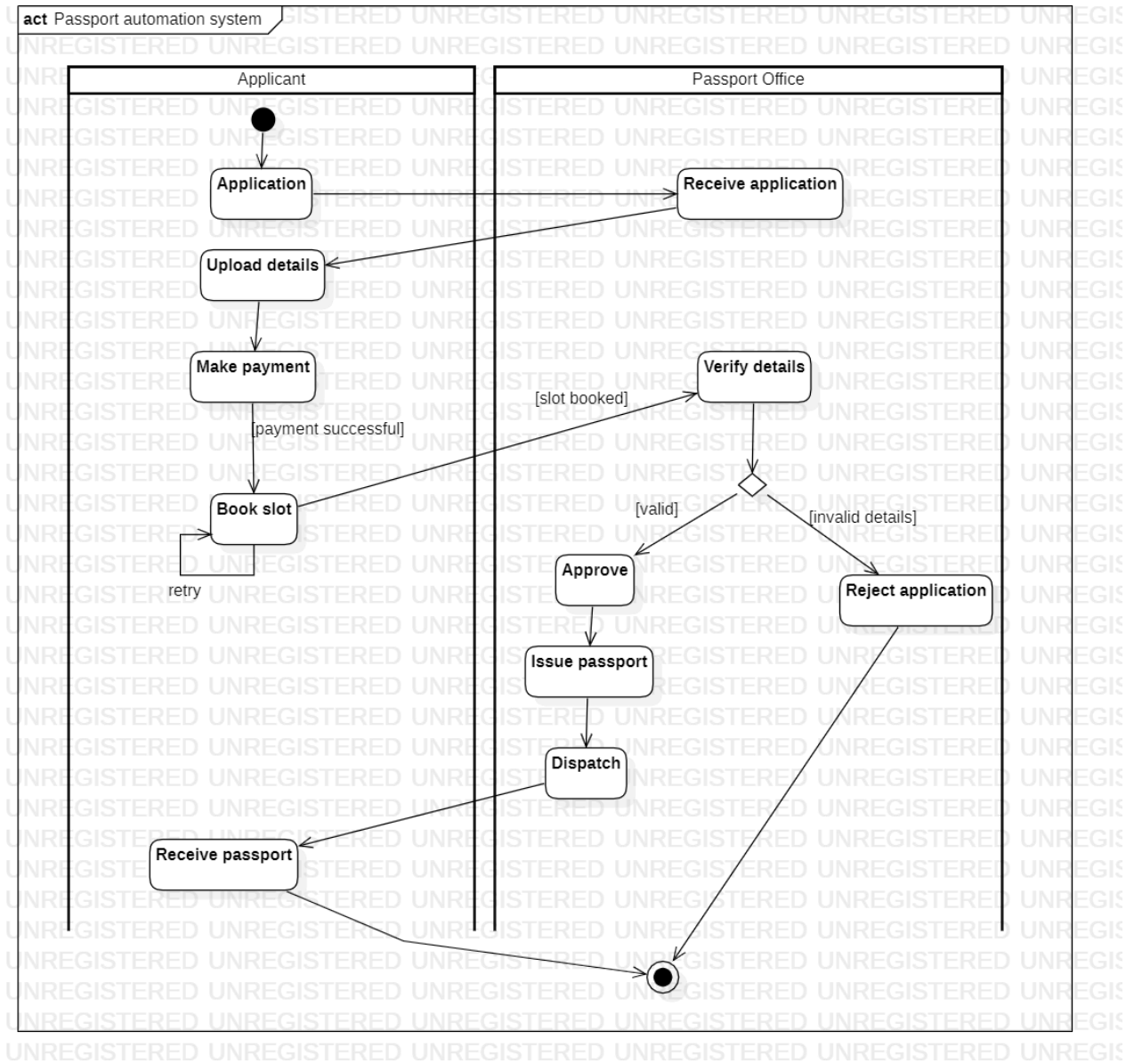


Figure 5.5 Activity Diagram of Passport Automation