Software Engineering Task-9

Kavya Aggarwal

HU22CSEN0100288

Design and analyze UML diagrams for:

1. Library management system

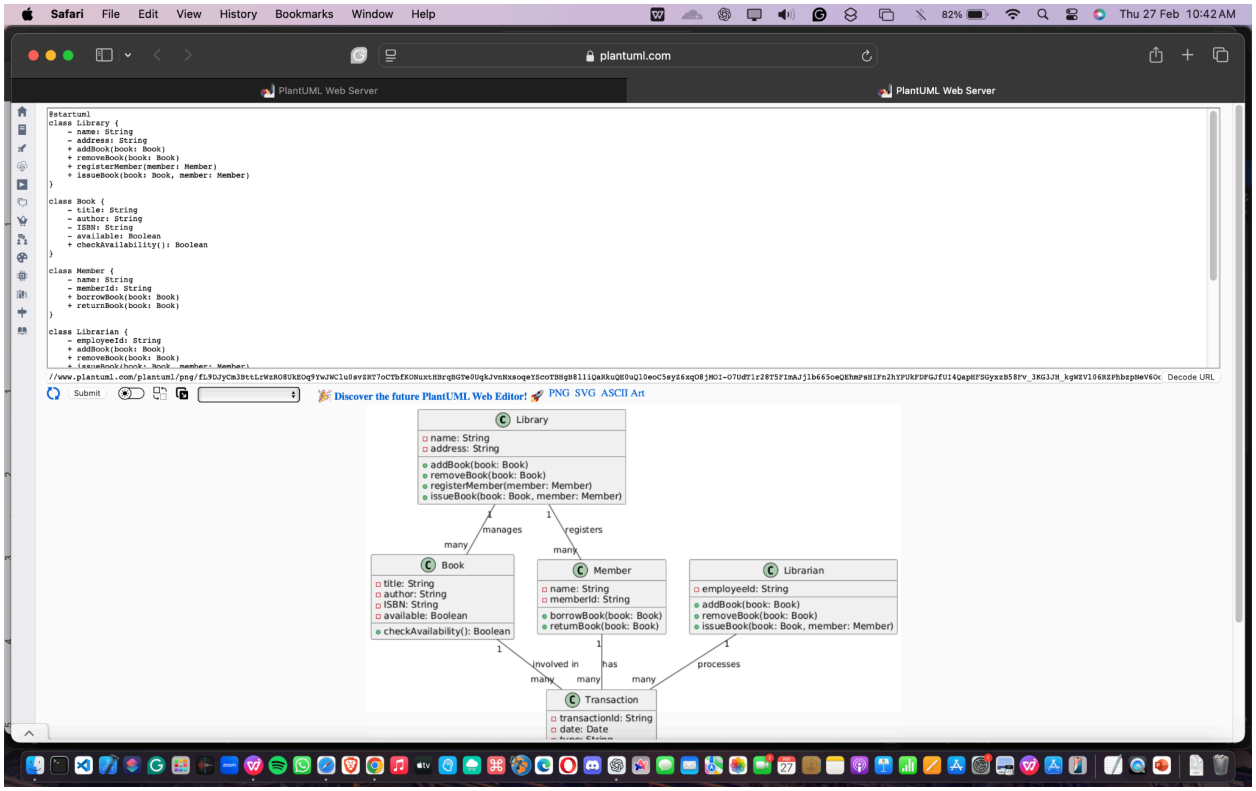
2.ATM management system

3.Hospitality Management System

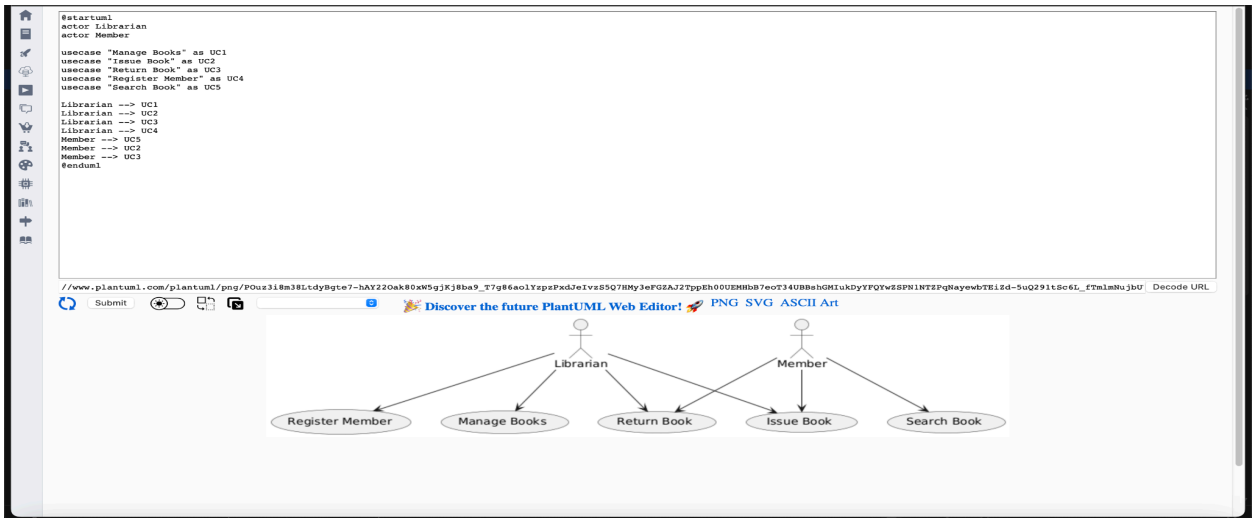
4.Credit Card processing system

1. **Library Management System Diagrams**

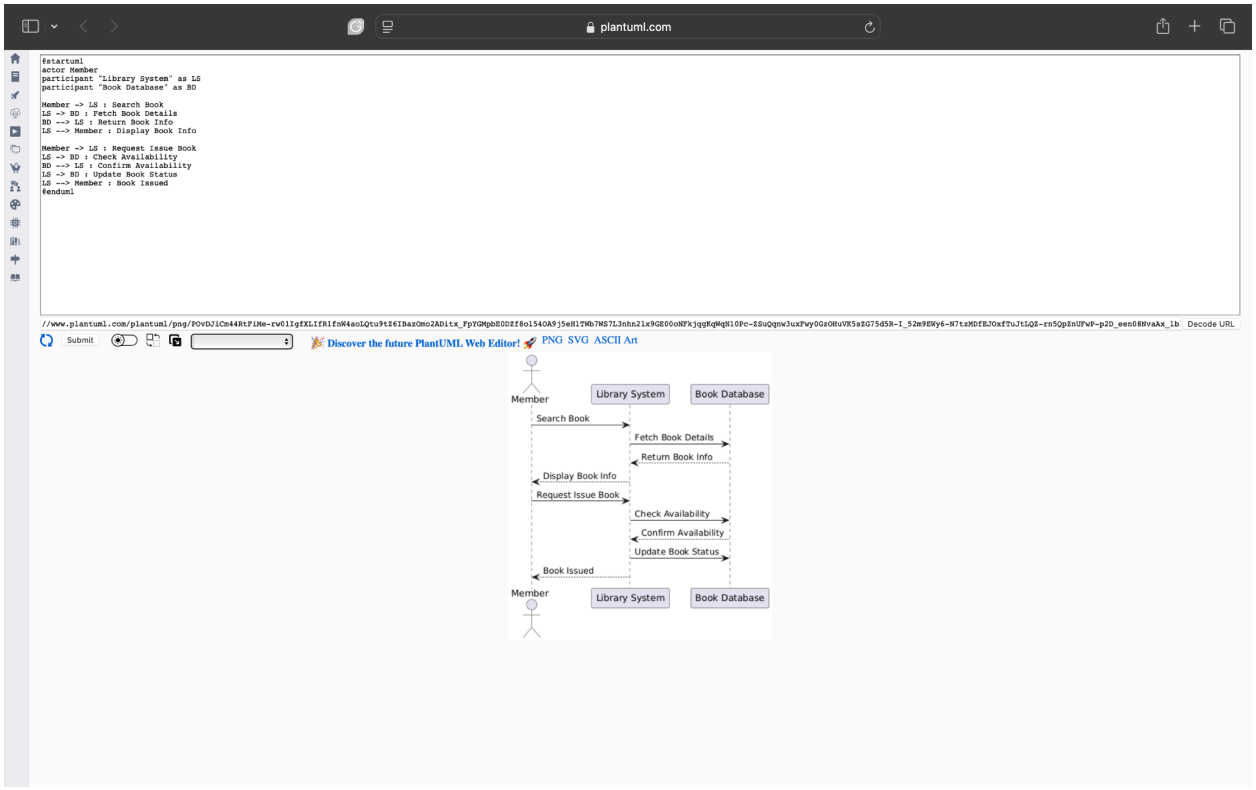
a. UML Class Diagram



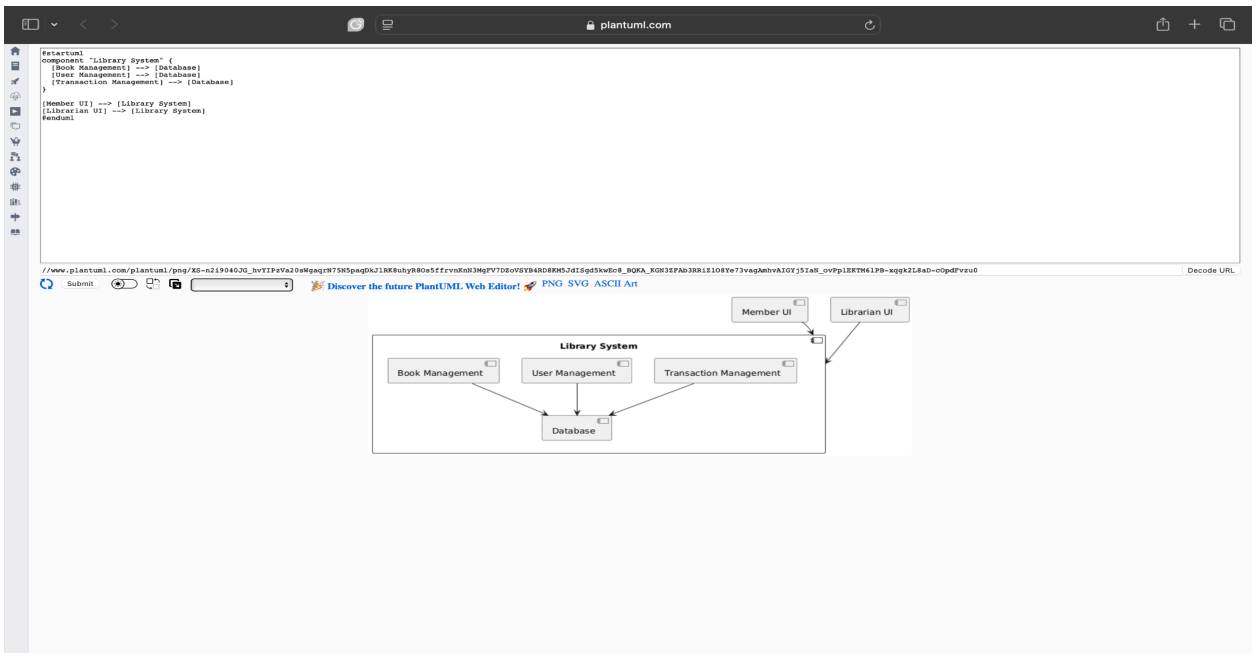
b. Use-Case Diagram



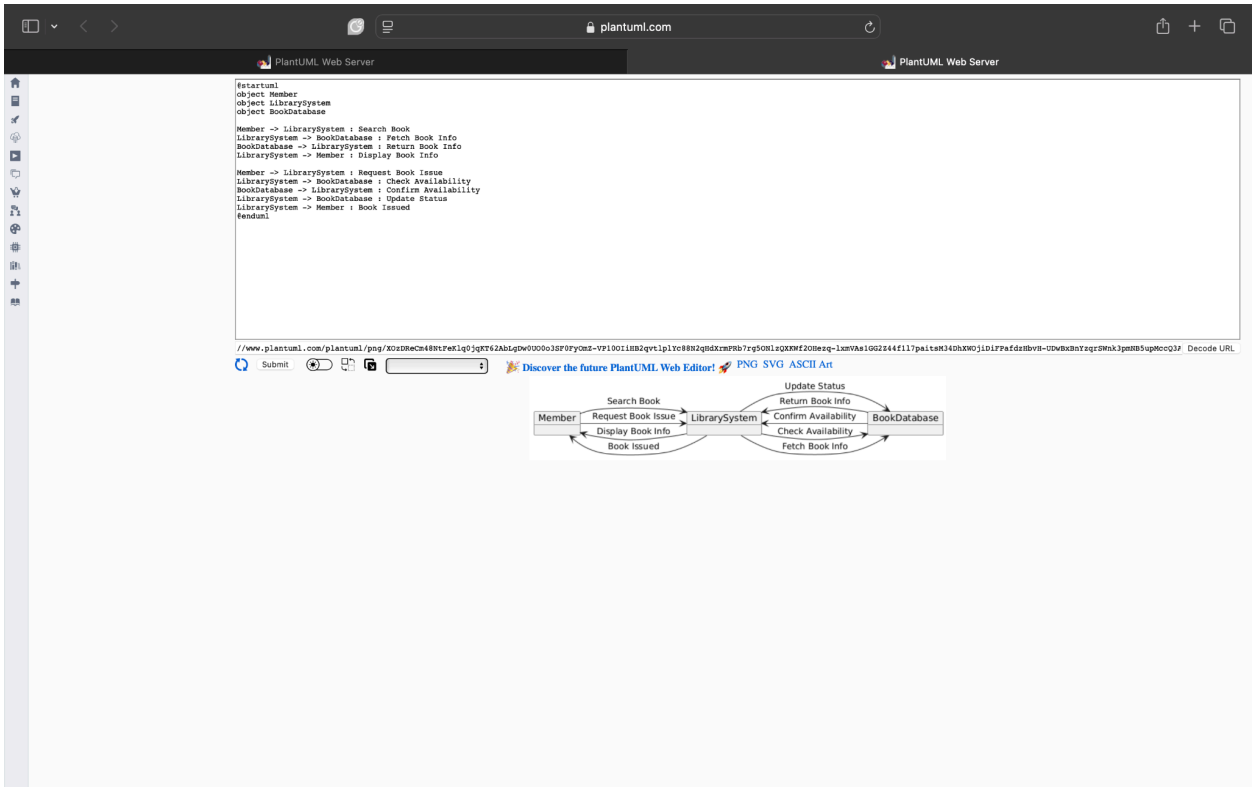
c. Sequence Diagram



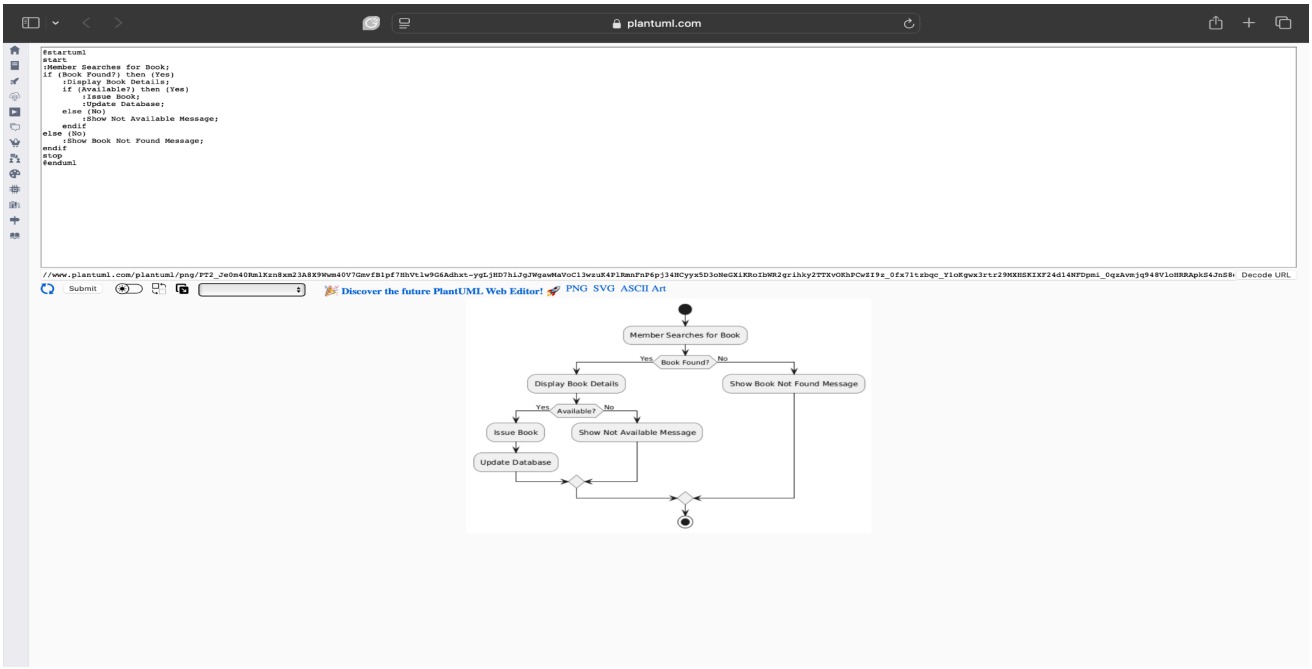
d. Component Diagram



e. Collaboration Diagram

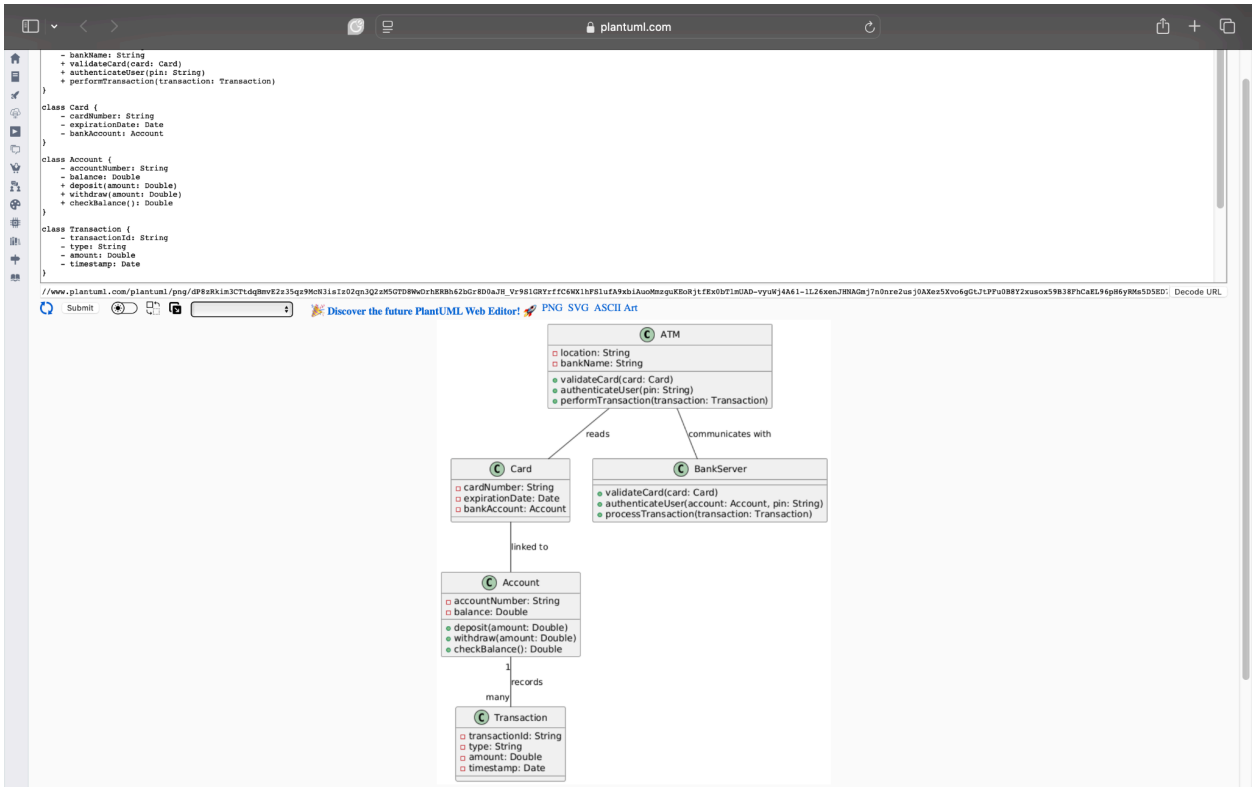


f. Activity Diagram

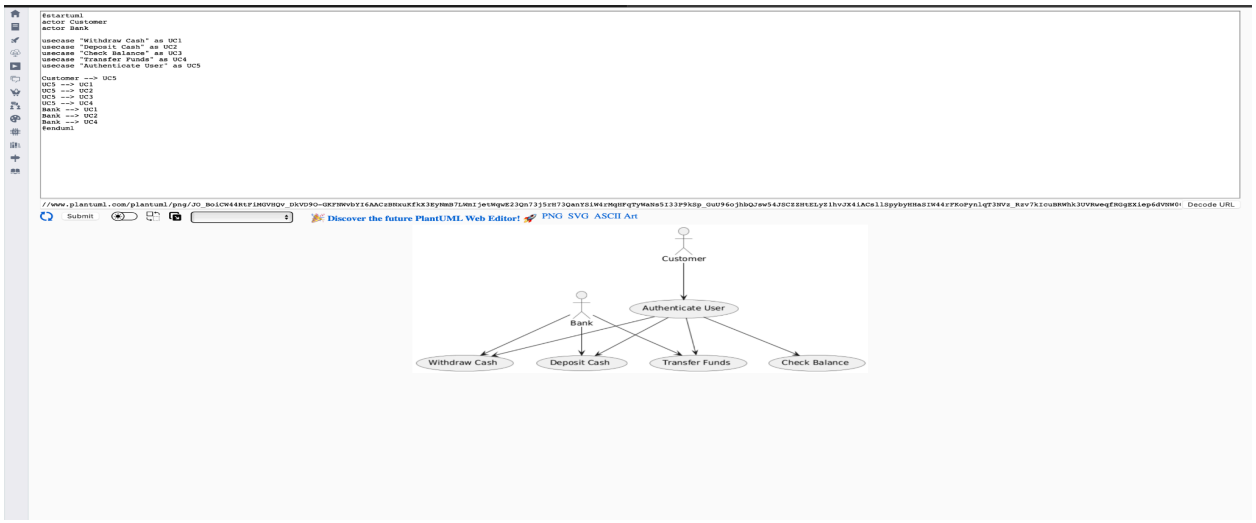


2. ATM Management System

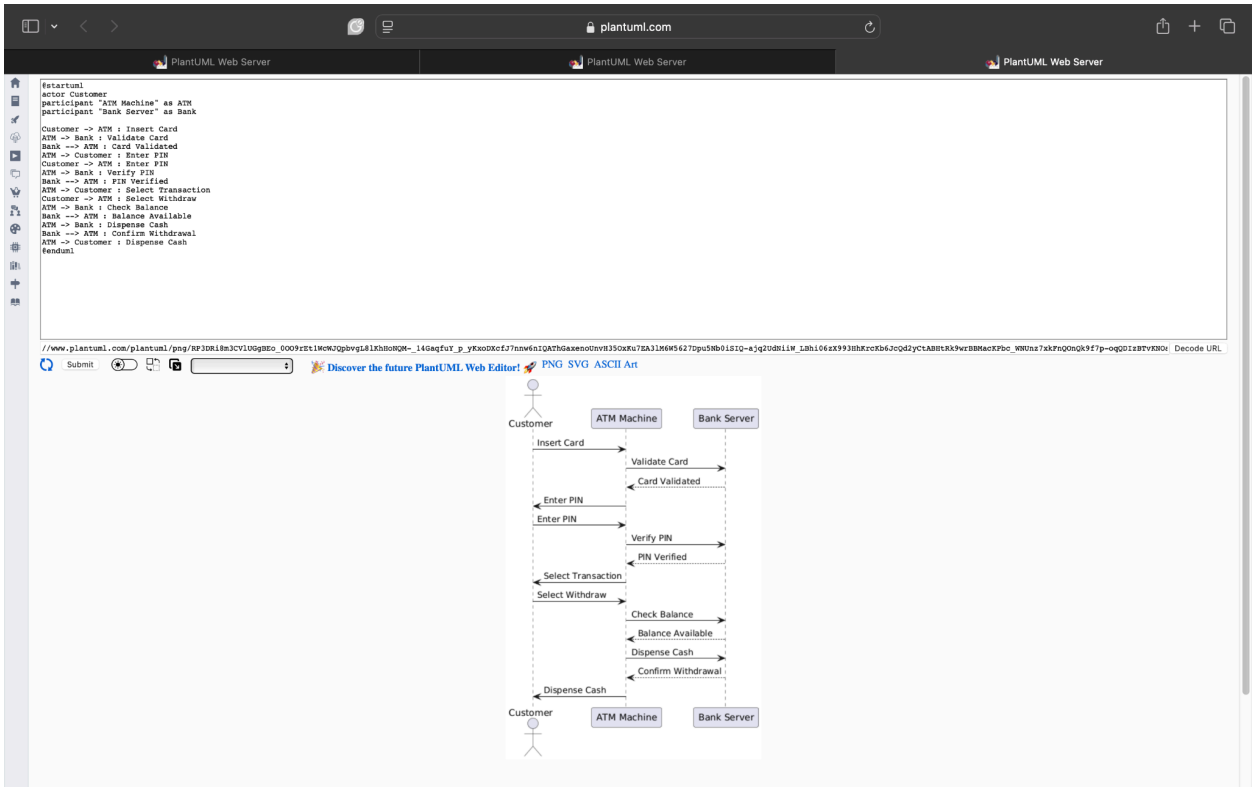
a. Class Diagram



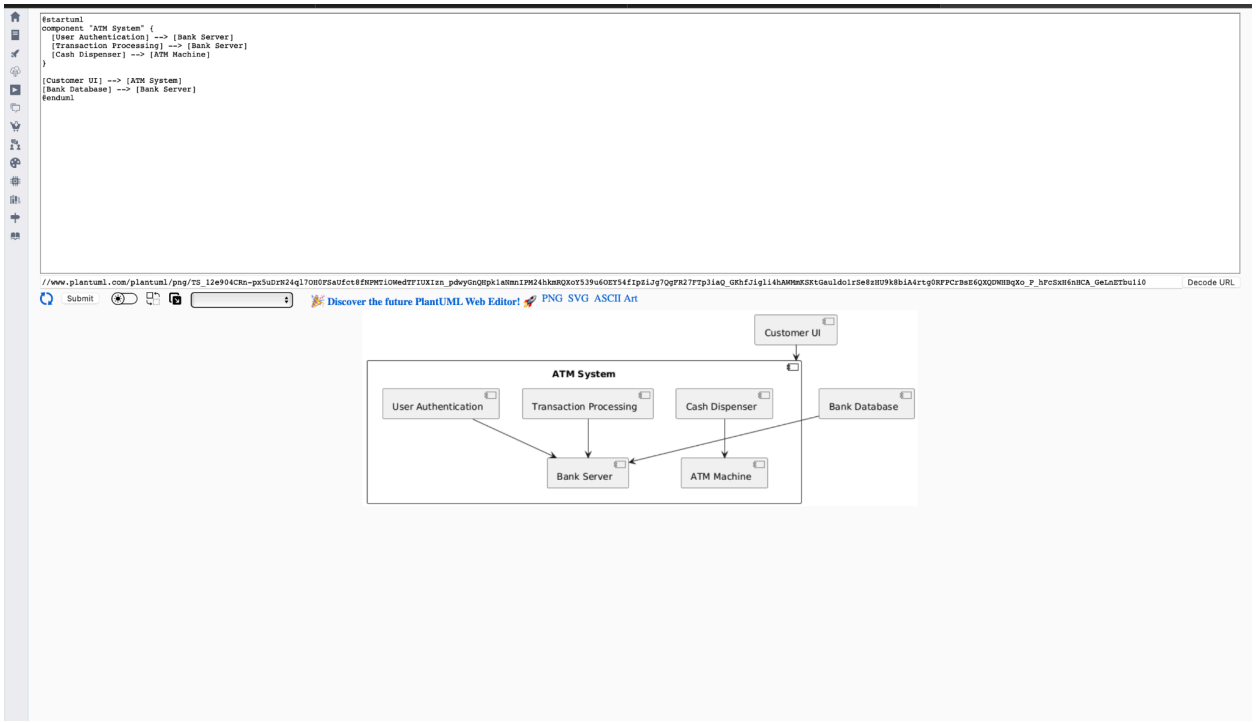
b. Use-Case Diagram



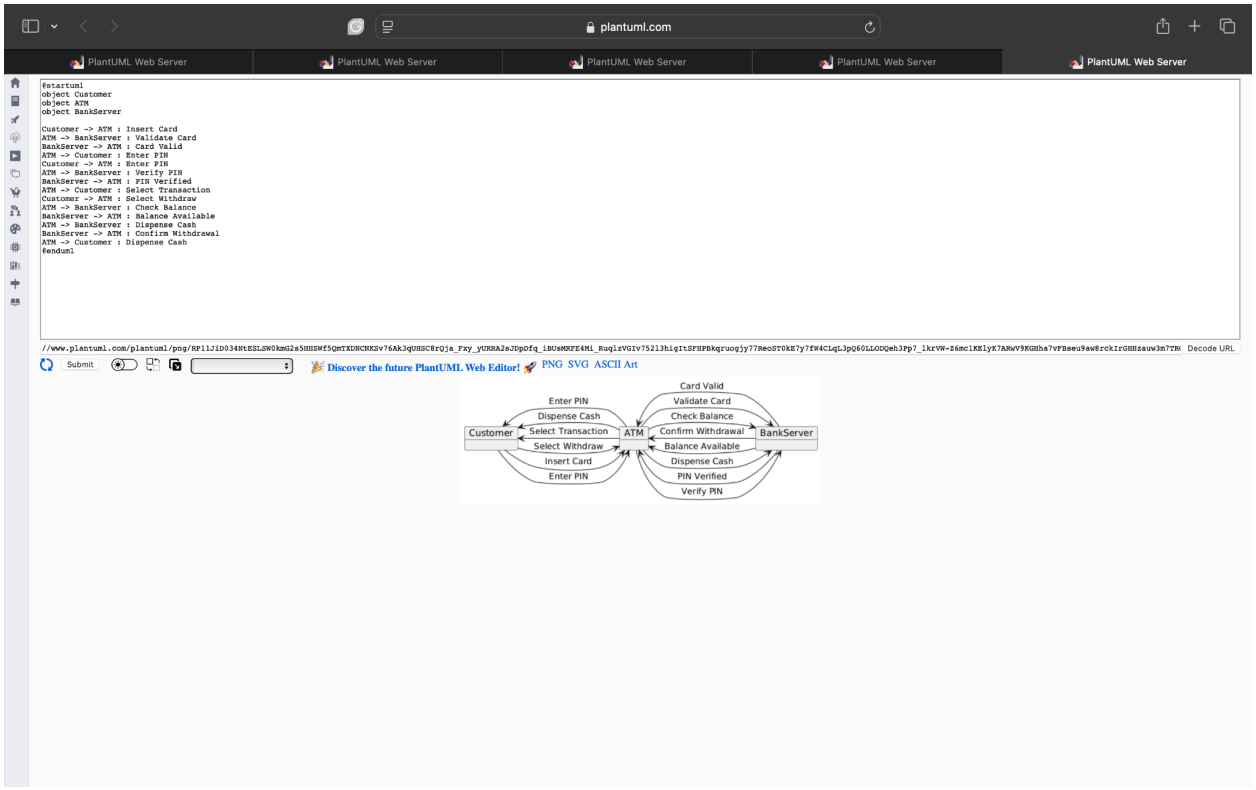
c. Sequence Diagram



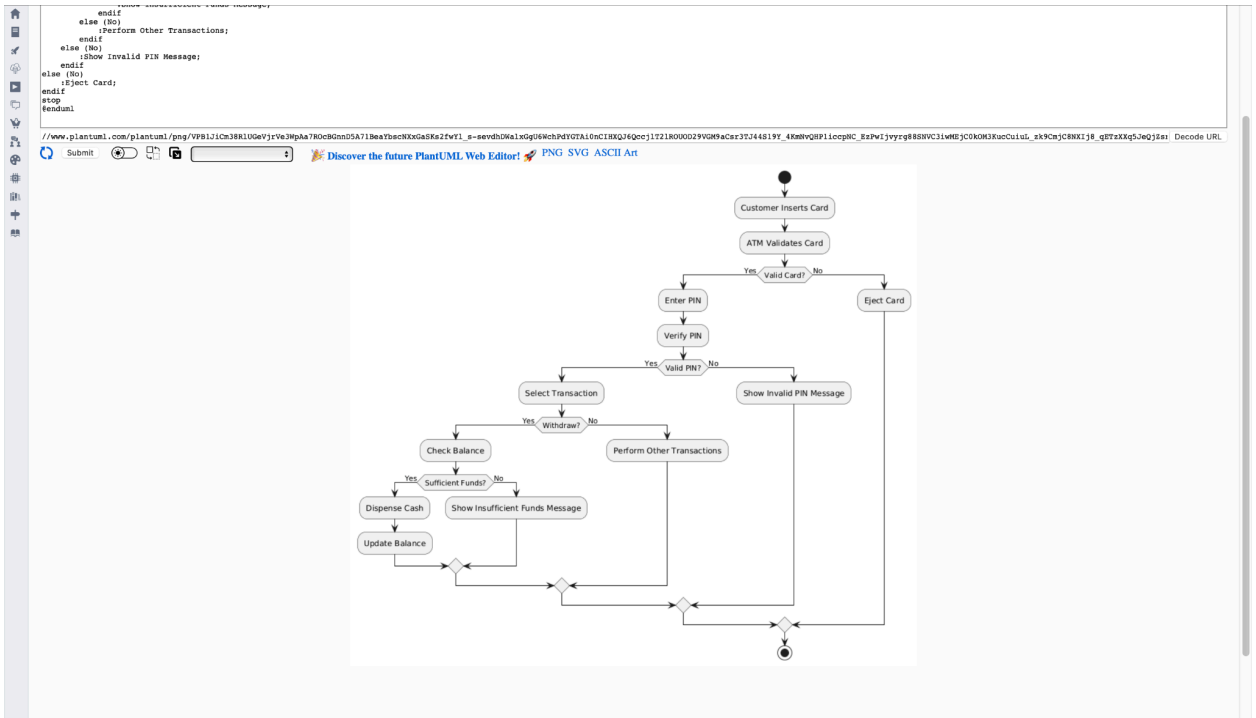
d. Component Diagram



e. Collaboration Diagram

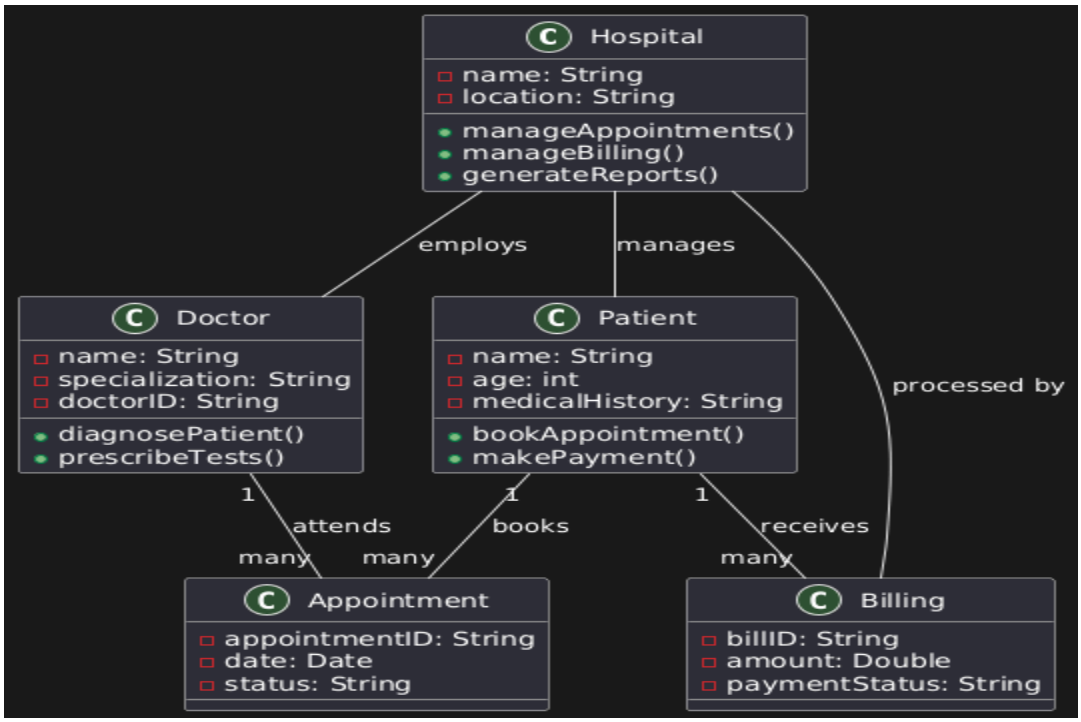


f. Activity Diagram

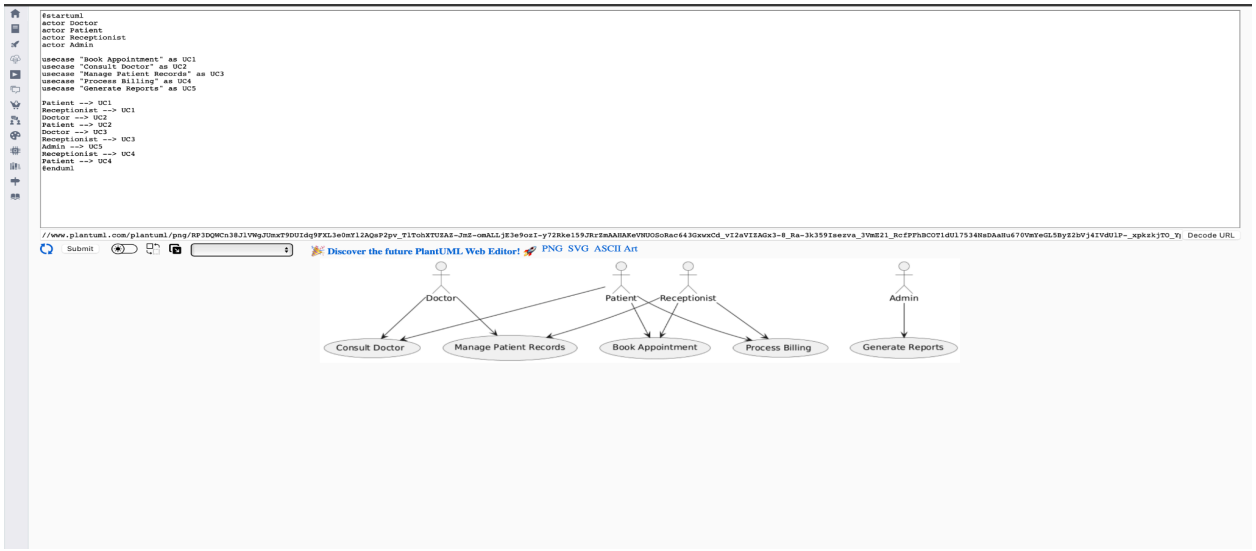


3. Hospital Management System

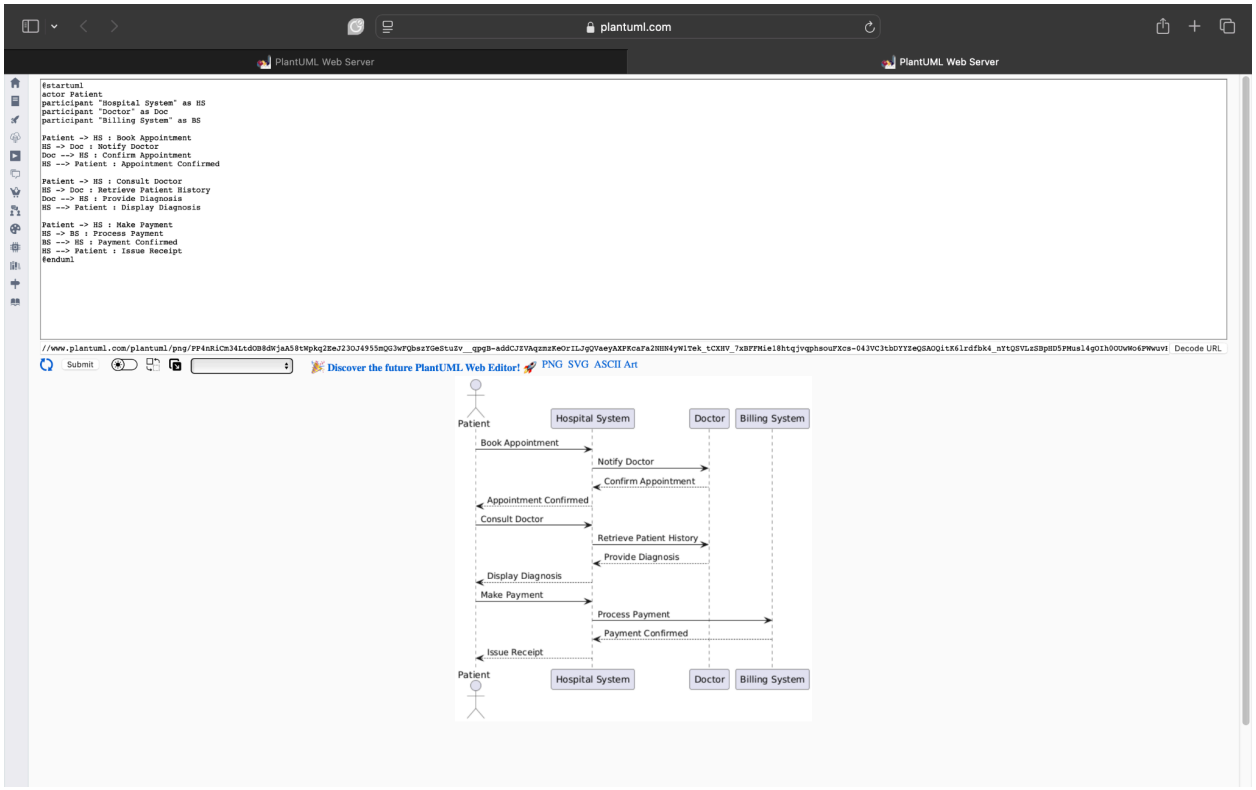
a. Class Diagram



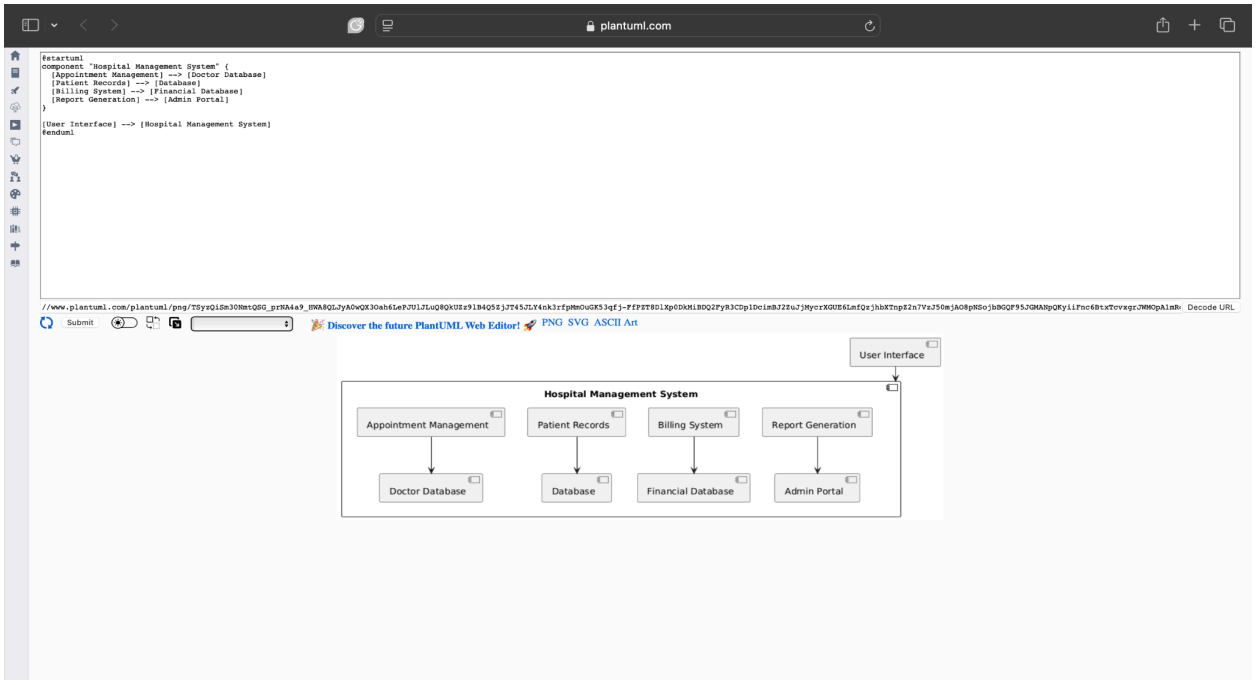
b. Use-Case Diagram



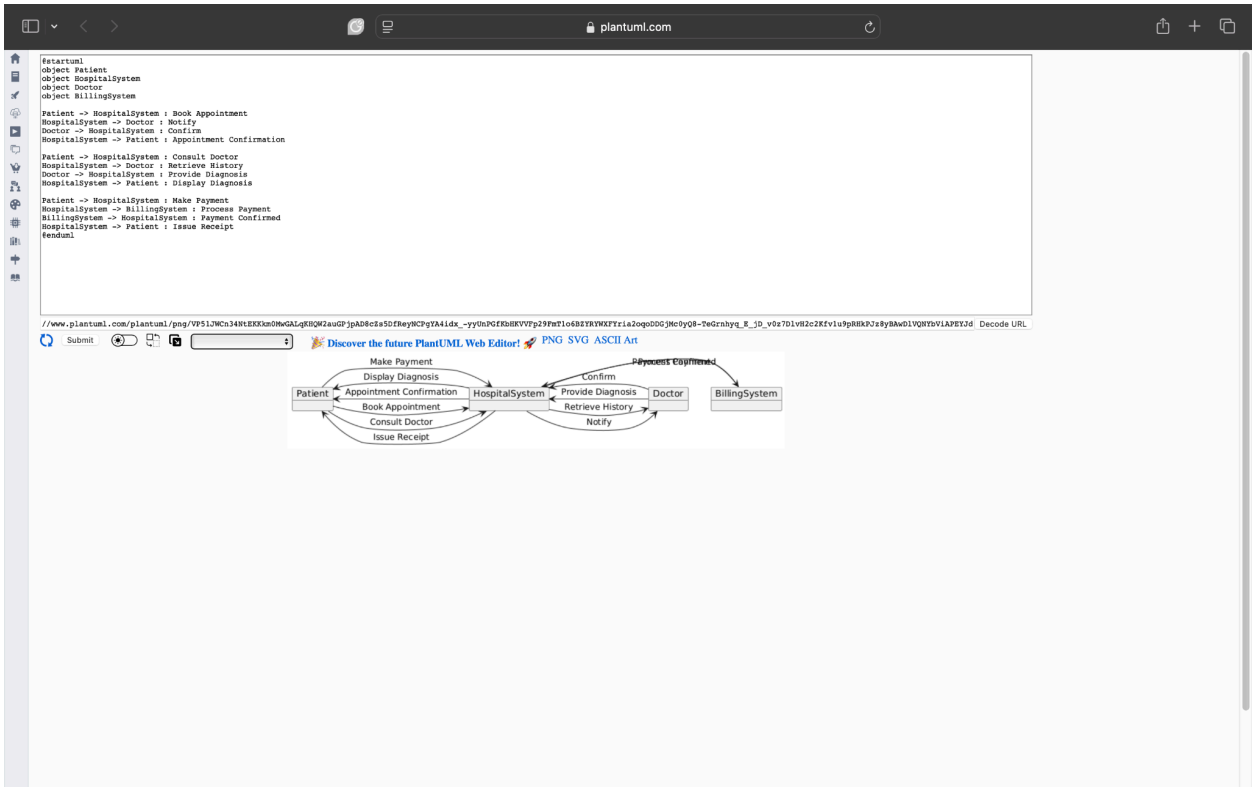
c. Sequence Diagram



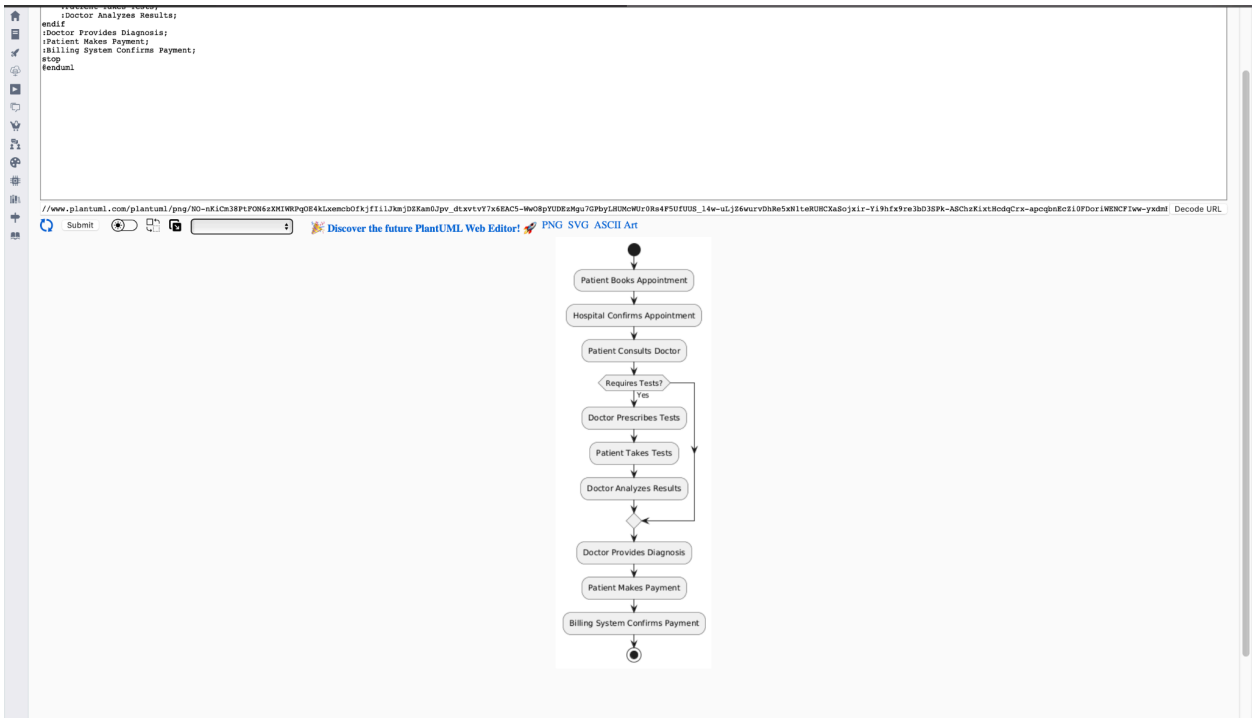
d. Component Diagram



e. Collaboration Diagram

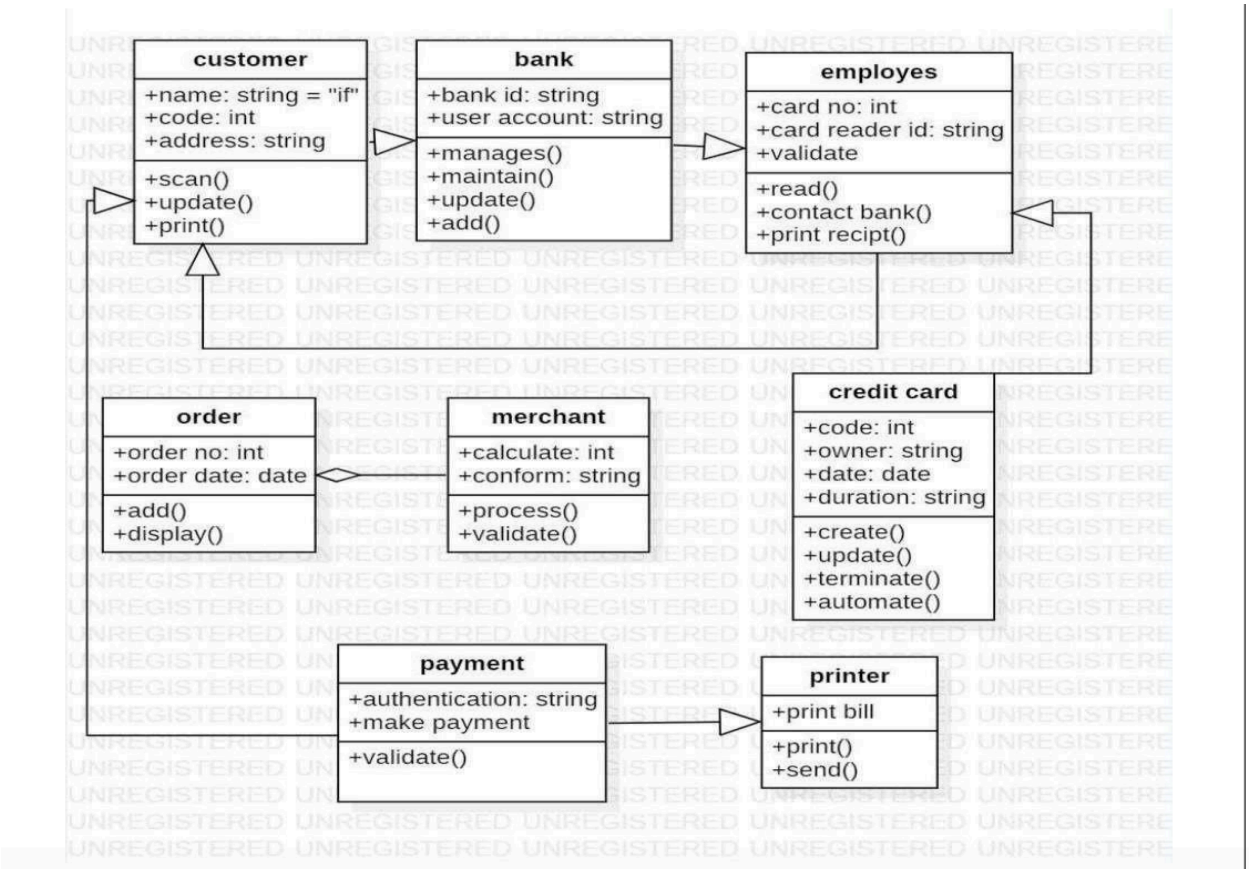


f. Activity Diagram

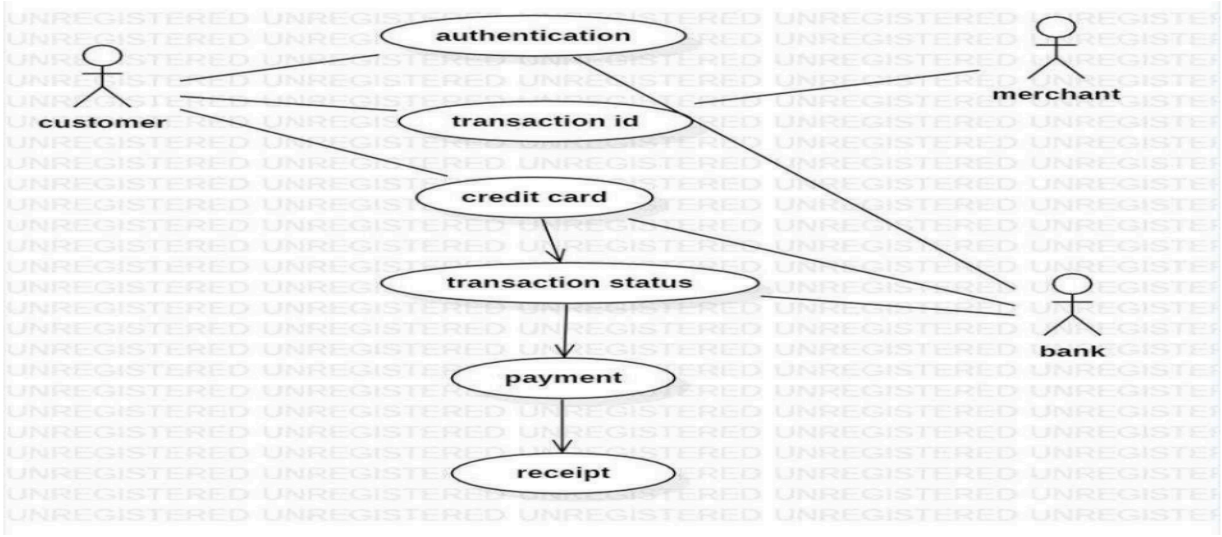


4. Credit Card Processing System

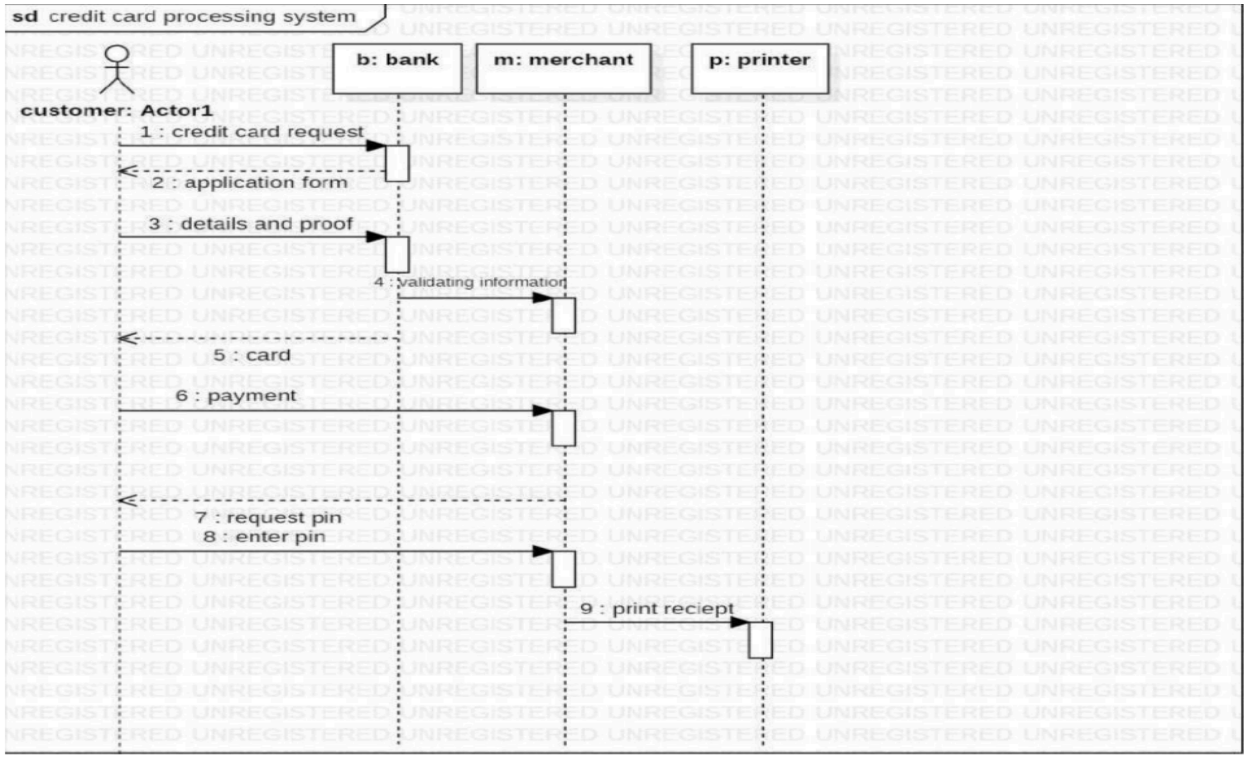
a. Class Diagram



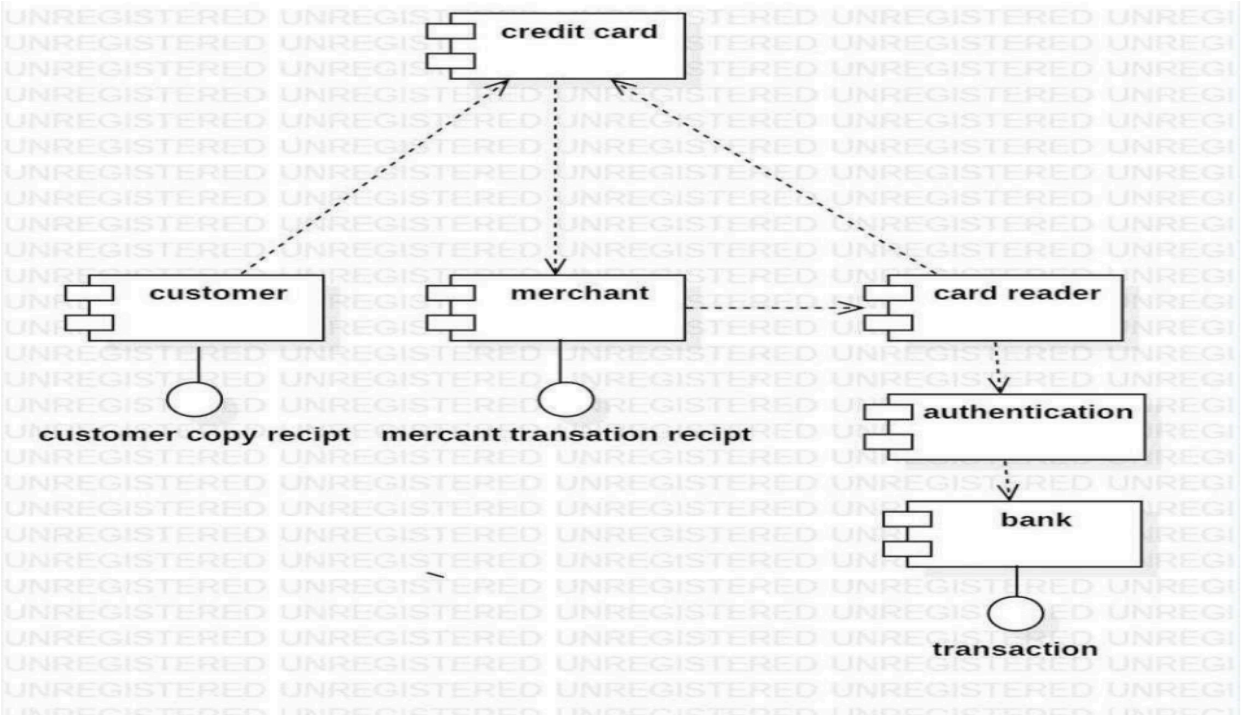
b. Use-Case Diagram



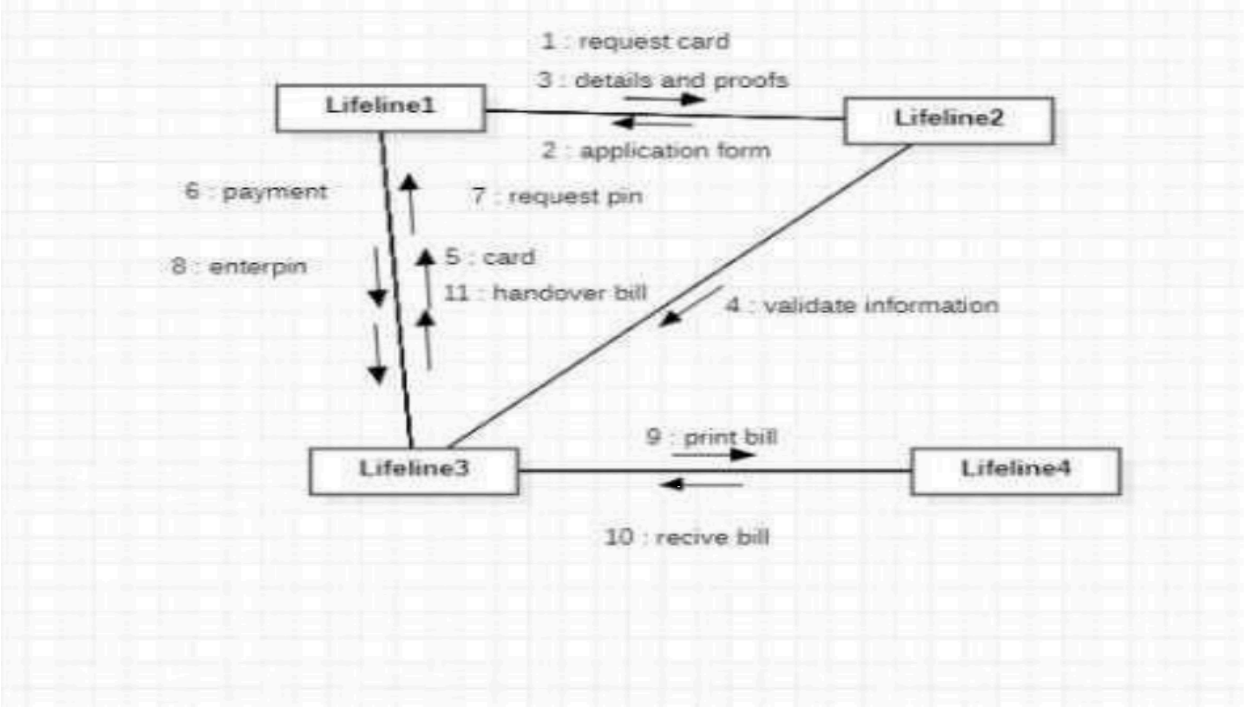
c. Sequence Diagram



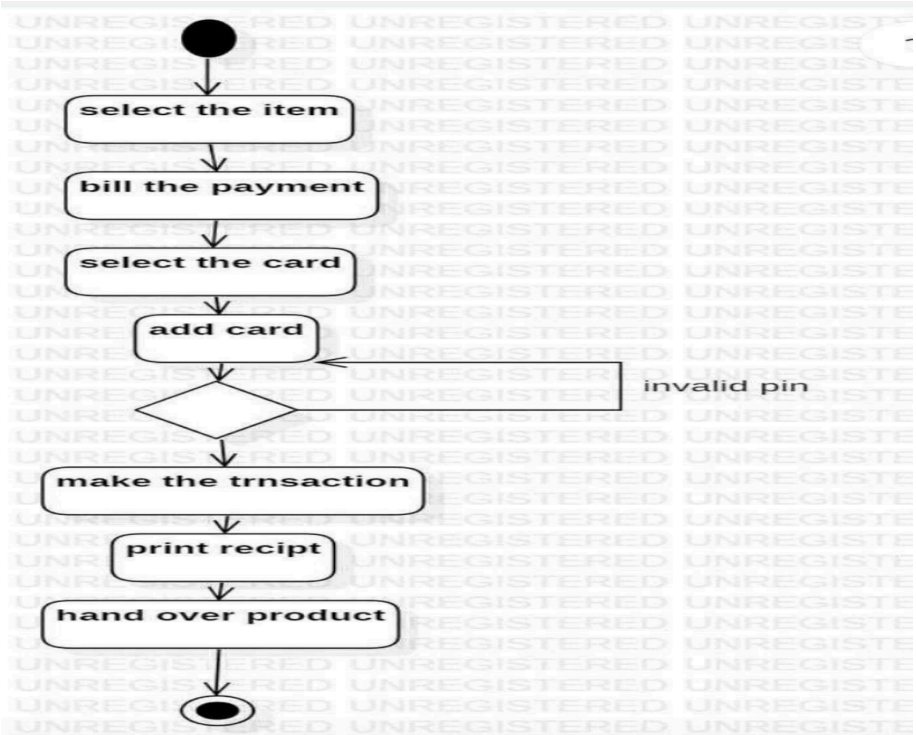
d. Component Diagram



e. Collaboration Diagram



f. Activity Diagram



1. Overview of UML Diagrams

Unified Modeling Language (UML) diagrams are used to visually represent the design and functionality of a system. Below is a brief analysis of each UML diagram type:

1. Use-Case Diagram

- **Purpose: Shows interactions between actors (users) and the system.**
- **Analysis:**
- **Defines major functionalities of the system.**
- **Identifies different roles interacting with the system.**
- **Helps understand the system from a user's perspective.**

2. Sequence Diagram

- **Purpose: Represents the flow of interactions between system components over time.**
- **Analysis:**
- **Demonstrates how objects communicate in sequential order.**
- **Useful in identifying message flow and system behavior.**
- **Essential for understanding event-driven scenarios like user requests and system responses.**

3. Component Diagram

- **Purpose: Depicts the system's components and their dependencies.**
- **Analysis:**
- **Helps in defining the structure of software modules.**
- **Shows how different components interact to form a complete system.**
- **Usefulness:**
- **Useful in high-level architectural design and microservices-based applications.**

4. Collaboration Diagram

- **Purpose: Shows interactions between objects focusing on message exchange.**
- **Analysis:**
- **Similar to a sequence diagram but focuses on object relationships.**
- **Highlights object interactions rather than chronological order.**
- **Useful in defining object roles and responsibilities in a process.**

5. Activity Diagram

- **Purpose: Represents the workflow of a system's activities.**
- **Analysis:**
- **Illustrates how different actions and decisions drive a process.**
- **Useful in business process modeling and logic flow representation.**
- **Helps in identifying parallel and sequential processes.**

6. Class Diagram

- **Purpose: Represents the static structure of the system, including its classes, attributes, and relationships.**
- **Analysis:**
- **Defines the blueprint for system implementation.**
- **Helps in object-oriented design and development.**
- **Shows relationships among various entities.**

2. Brief Descriptions of Management Systems

1. Library Management System

A Library Management System (LMS) automates the process of managing books, members, and transactions in a library. It includes functionalities like book borrowing, returning, catalog management, and member registration.

- **Users: Librarians, Members, Administrators.**
- **Core Features:**
- **Book management (add, remove, update books).**
- **Member registration and tracking.**
- **Issuing and returning books.**
- **Transaction history tracking.**
- **UML Diagrams Used: Use-Case, Sequence, Component, Collaboration, Activity, and Class Diagrams.**

2. ATM Management System

An ATM Management System manages automated teller machine (ATM) transactions such as cash withdrawals, deposits, balance inquiries, and fund transfers. It ensures secure authentication and interaction with the banking system.

- **Users: Customers, Bank Server.**
- **Core Features:**
- **Secure user authentication using PIN.**
- **Transaction processing (withdrawals, deposits, balance inquiries).**
- **Interaction with the bank database for real-time processing.**
- **Error handling and transaction failure rollback.**
- **UML Diagrams Used: Use-Case, Sequence, Component, Collaboration, Activity, and Class Diagrams.**

3. Hospital Management System

A Hospital Management System (HMS) streamlines patient appointments, medical records, billing, and doctor-patient interactions. It enhances healthcare efficiency by integrating various hospital departments.

- **Users: Patients, Doctors, Receptionists, Administrators.**
- **Core Features:**
- **Appointment scheduling and management.**
- **Storing and retrieving patient medical records.**
- **Billing system for patient charges.**
- **Doctor-patient interaction for diagnosis and treatment.**
- **Report generation for hospital analytics.**
- **UML Diagrams Used: Use-Case, Sequence, Component, Collaboration, Activity, and Class Diagrams.**

4. Credit Card Processing System

A Credit Card Processing System (CCPS) enables secure transactions using credit cards for purchases, payments, and fund transfers. It connects merchants, banks, and customers for real-time transaction processing.

- **Users: Customers, Merchants, Banks, Payment Gateways.**
- **Core Features:**
- **Secure credit card validation and authentication.**
- **Payment authorization and transaction settlement.**
- **Fraud detection and prevention mechanisms.**
- **Integration with payment gateways and financial institutions.**
- **Transaction logs and dispute resolution.**
- **UML Diagrams Used: Use-Case, Sequence, Component, Collaboration, Activity, and Class Diagrams.**