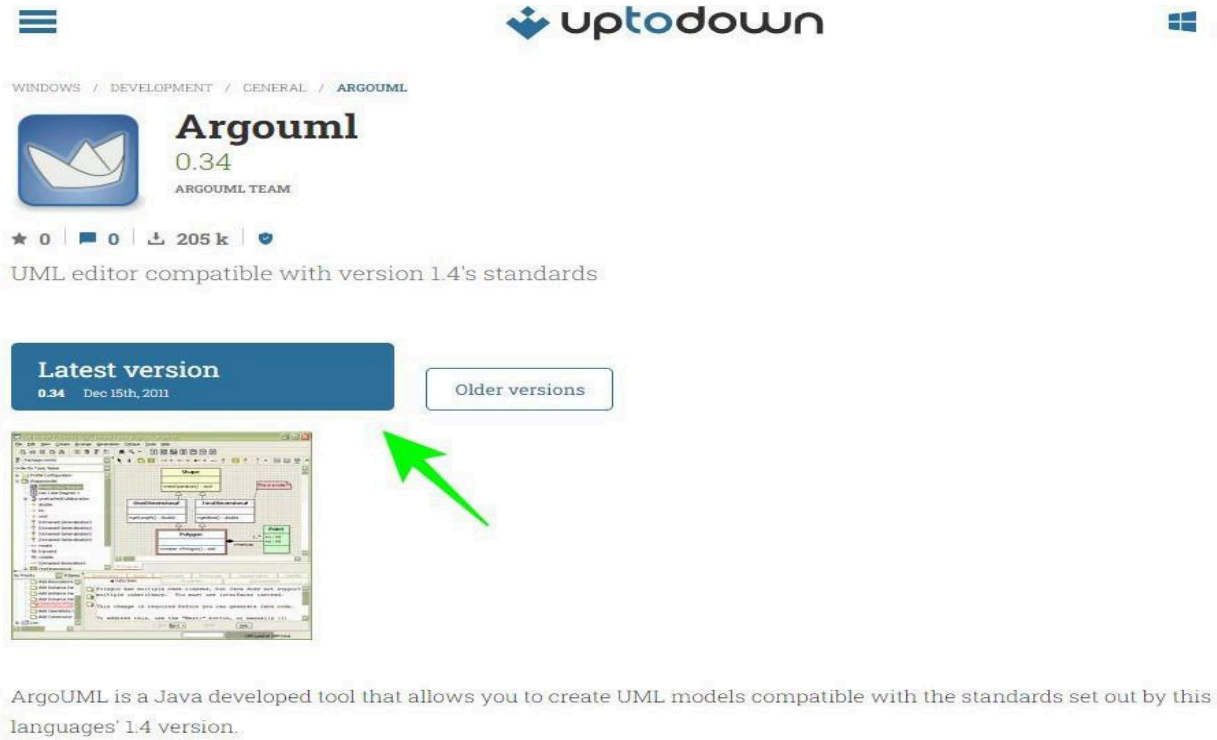
EX. NO:1 **DATE:31/01/2025**

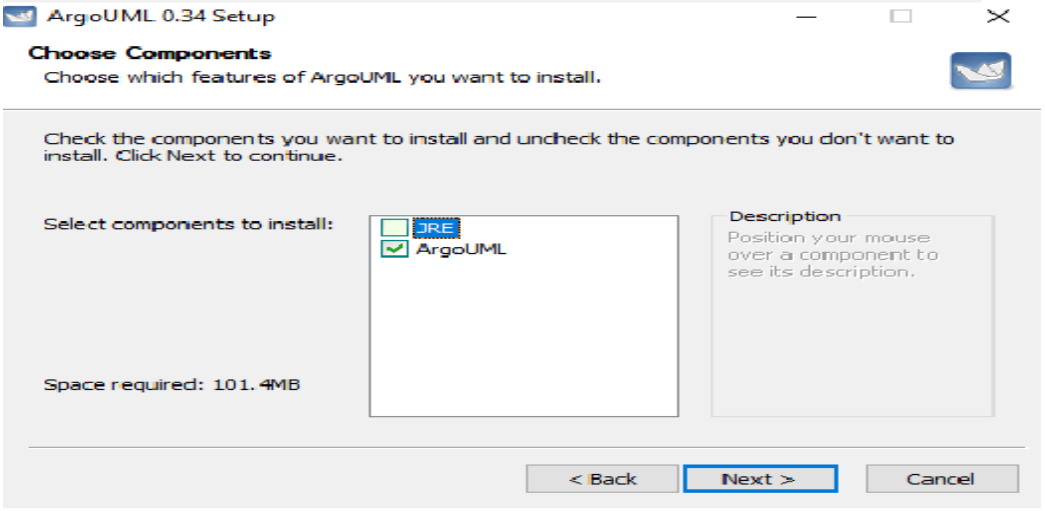
STUDY OF ARGO UML

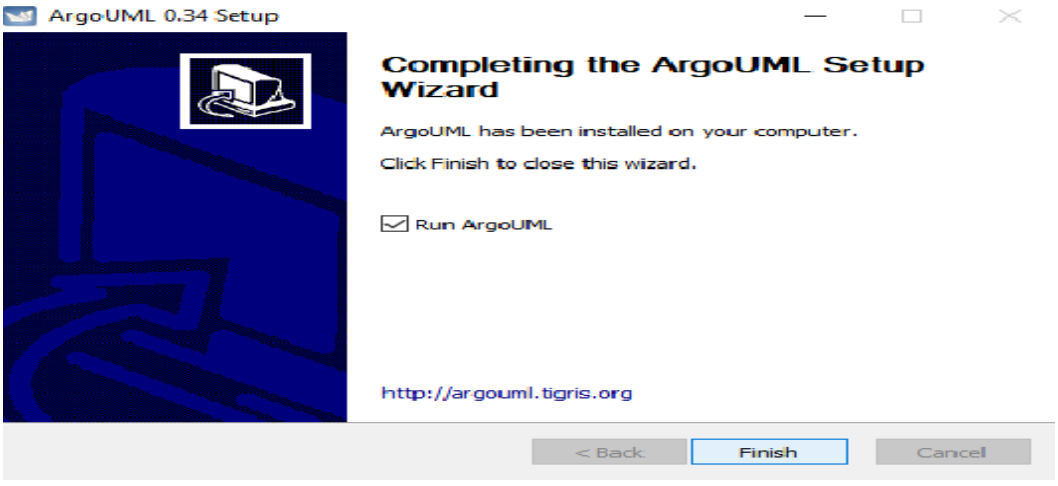
AIM:



ArgoUML is a Java developed tool that allows you to create UML models compatible with the standards set out by this languages' 1.4 version.


ArgoUML 0.34 Setup
Welcome to the ArgoUML Setup Wizard
This wizard will guide you through the installation of ArgoUML.
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.
Click Next to continue.
Next >
Cancel


ArgoUML 0.34 Setup
Choose Components
Choose which features of ArgoUML you want to install.
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.
Select components to install:
JRE
ArgoUML
Description
Position your mouse over a component to see its description.
Space required: 101.4MB
< Back
Next >
Cancel


ArgoUML 0.34 Setup
Completing the ArgoUML Setup Wizard
ArgoUML has been installed on your computer.
Click Finish to close this wizard.
Run ArgoUML
http://argouml.tigris.org
< Back
Finish
Cancel

RESULT:

EX. NO:2 **DATE:07/02/25**

STUDENT RESULT MANAGEMENT SYSTEM

AIM:

OUTPUT:

Student.java

```
public class Student {
public int studentID;
public String name;
public String email; public
void viewResult() {
  }
}
```

Course.java

```
public class Course {
public int courseID;
public int strings;
public String courseName;
}
```

Faculty.java

```
public class Faculty { public
int adminID; public String
username; public String
password; public void
addEditMarks() {
  }
}
```

Result.java

```
public class Result {
public int resultID;
public String grade;
public float marks;
public void generateReportCard() {
   }
}
```

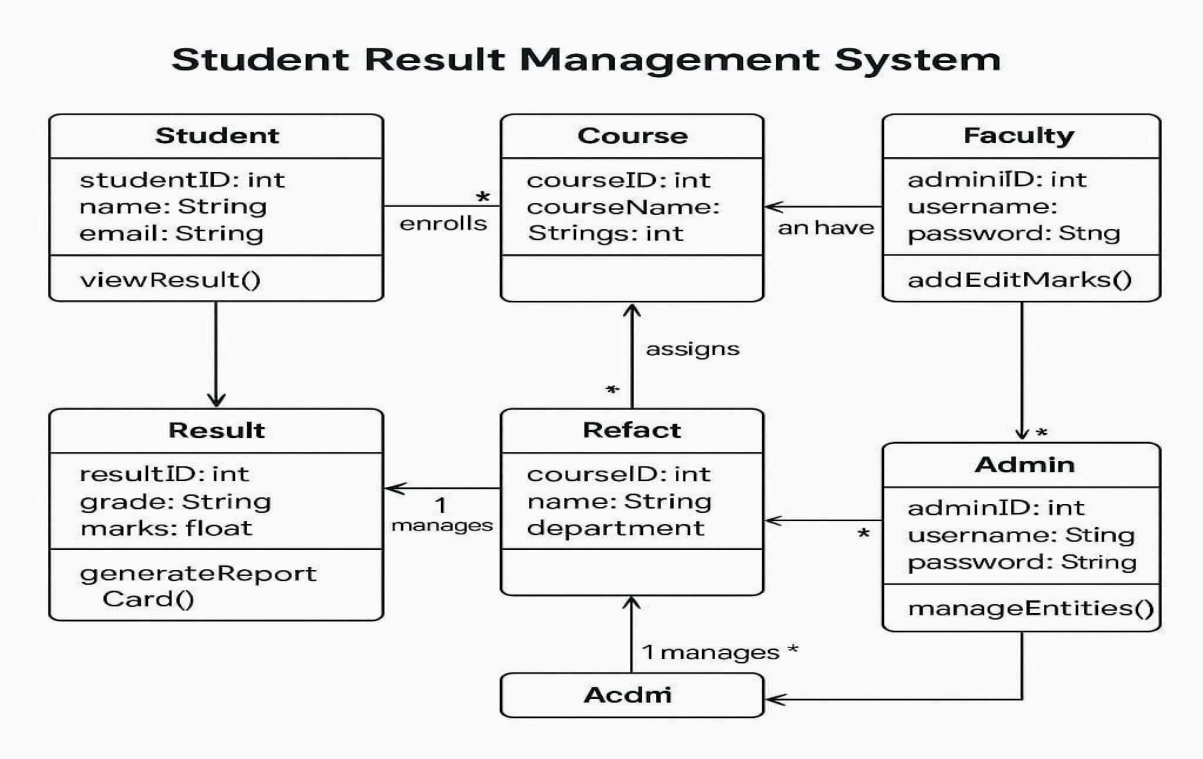
Admin.java

```
public class Admin {
public int adminID;
public String username;
public String password;
public void manageEntities() {
   }
}
```

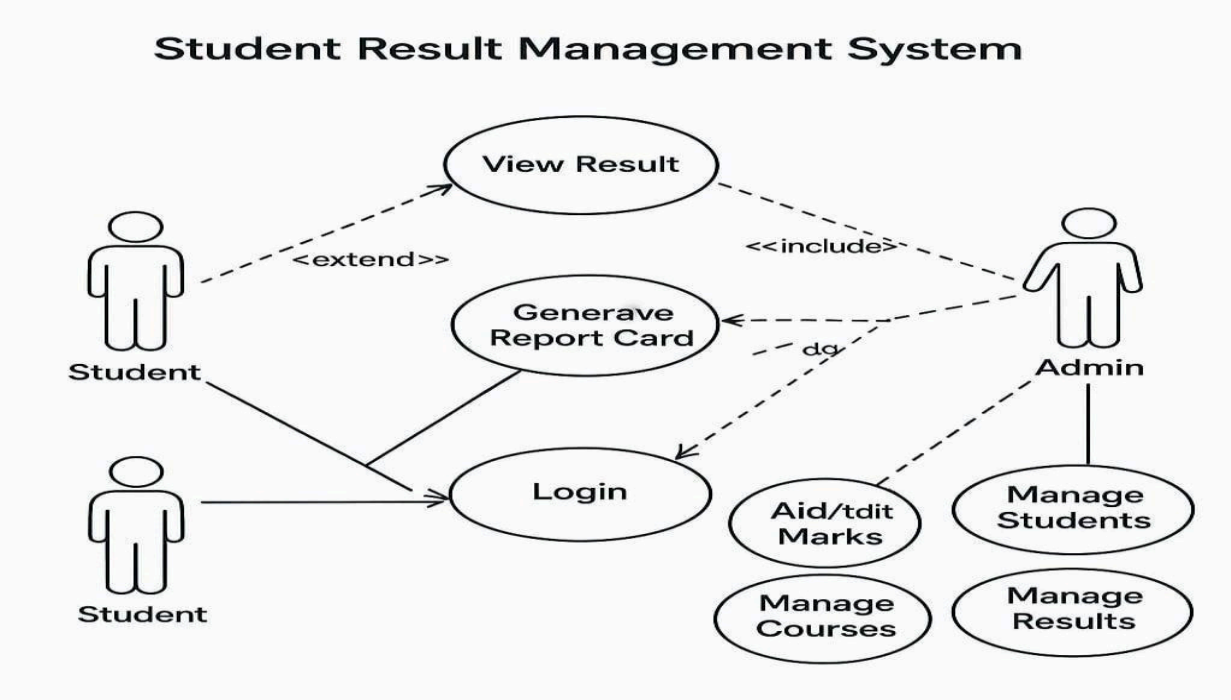
Refact.java

```
public class Refact {
public int courseID;
public String name;
public String department;
}
```

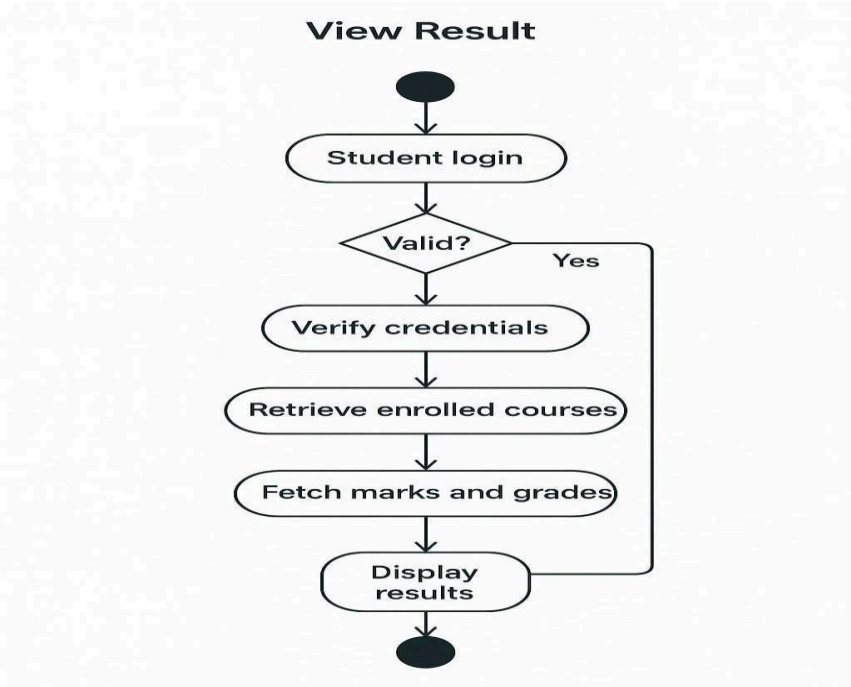
Class Diagram:



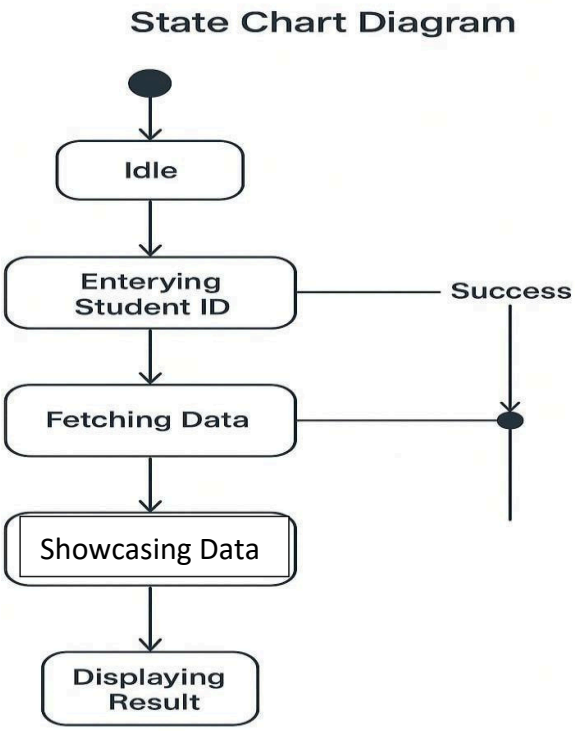
Use-case Diagram:



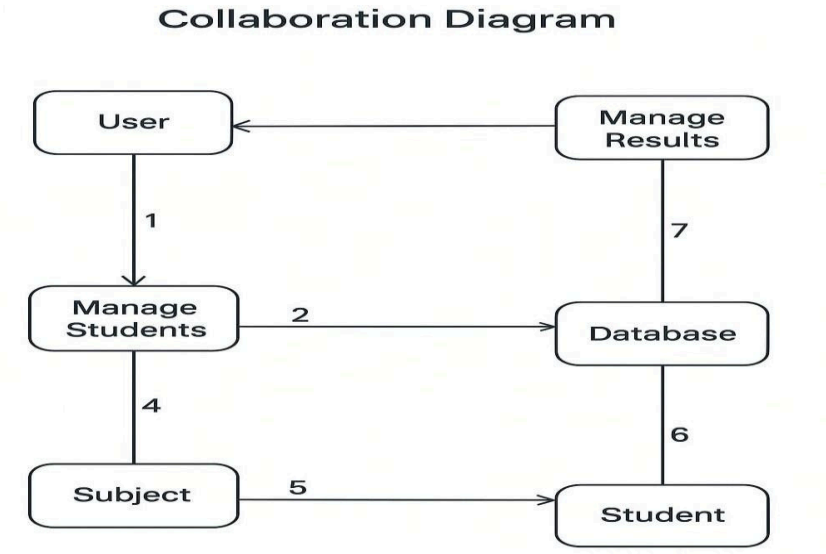
Activity Diagram:



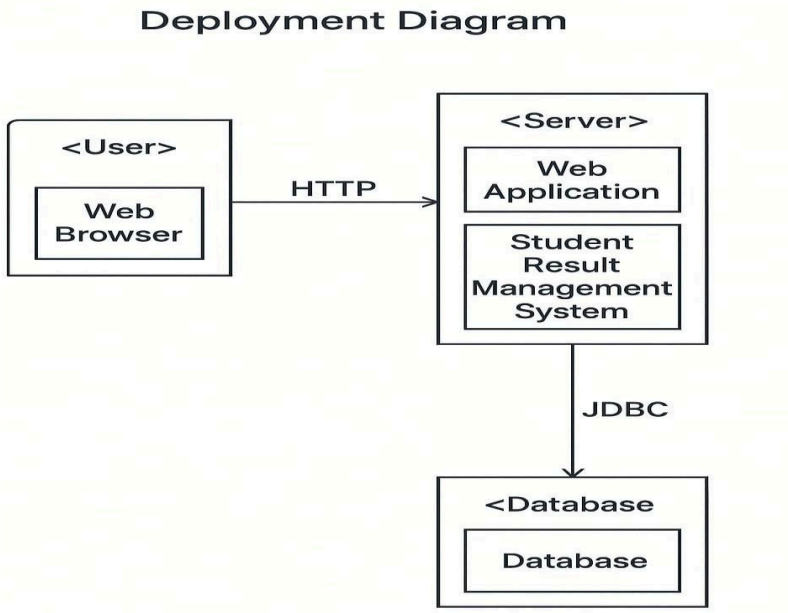
State chart diagram:



Collaboration Diagram:



Deployment Diagram:



RESULT:

EX. NO:3 **DATE:14/02/2025**

INVENTORY CONTROL SYSTEM

AIM:

OUTPUT:

Item.java

```
public class Item {
public int itemID;
public String name;
public int quantity;
public double price;
}
```

Supplier.Java

```
public class Supplier {
public int supplierID;
public String name;
public String contactInfo;
}
```

InventoryManagement.java

```
public class InventoryManager {
public int managerID;
public String name;
public String loginCredentials;
}
```

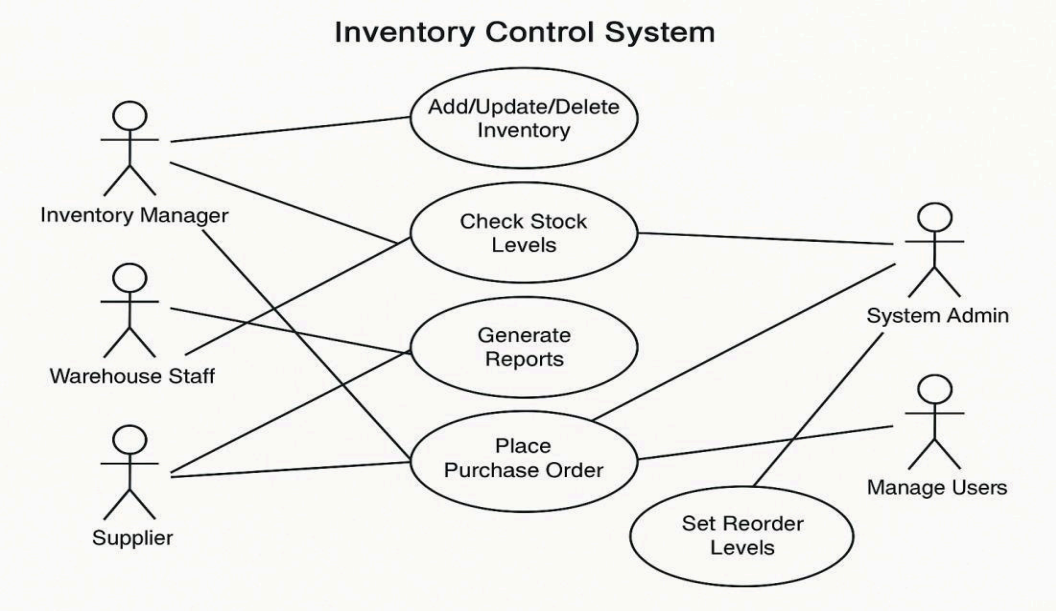
Order.java

```
import java.util.Date;

public class Order {

public int orderID;

public Date date;

public String status;

}
```

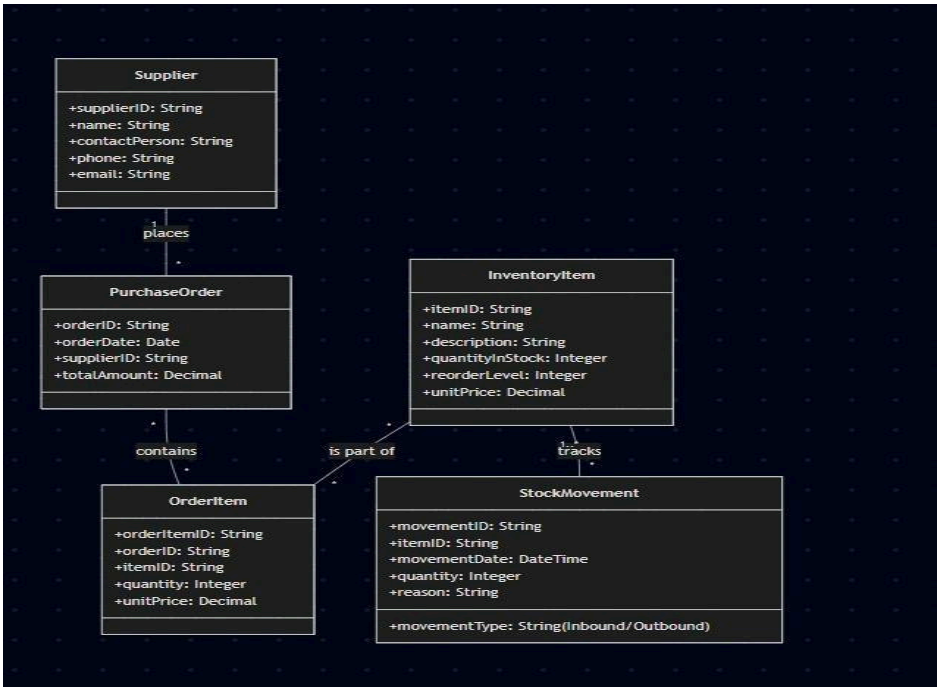
Purchase.java

```
import java.util.Date; public class Purchase { public int purchaseID; public Date date; public double totalAmount; }
```

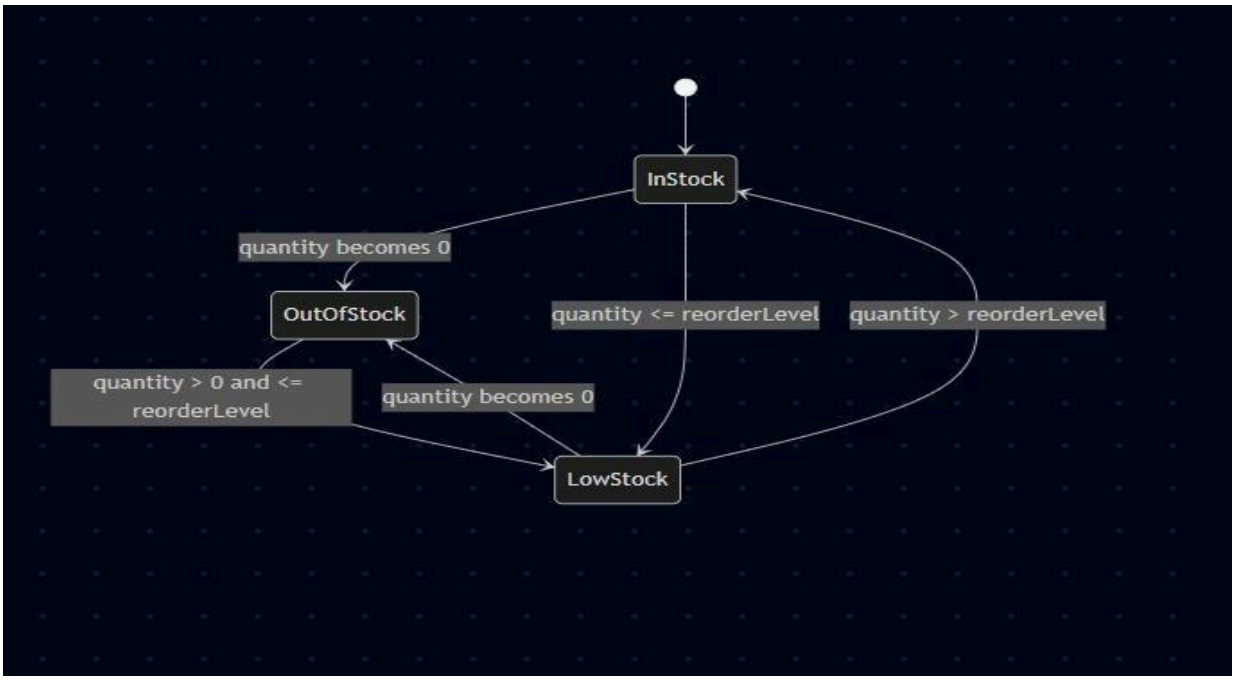
Use Case Diagram:



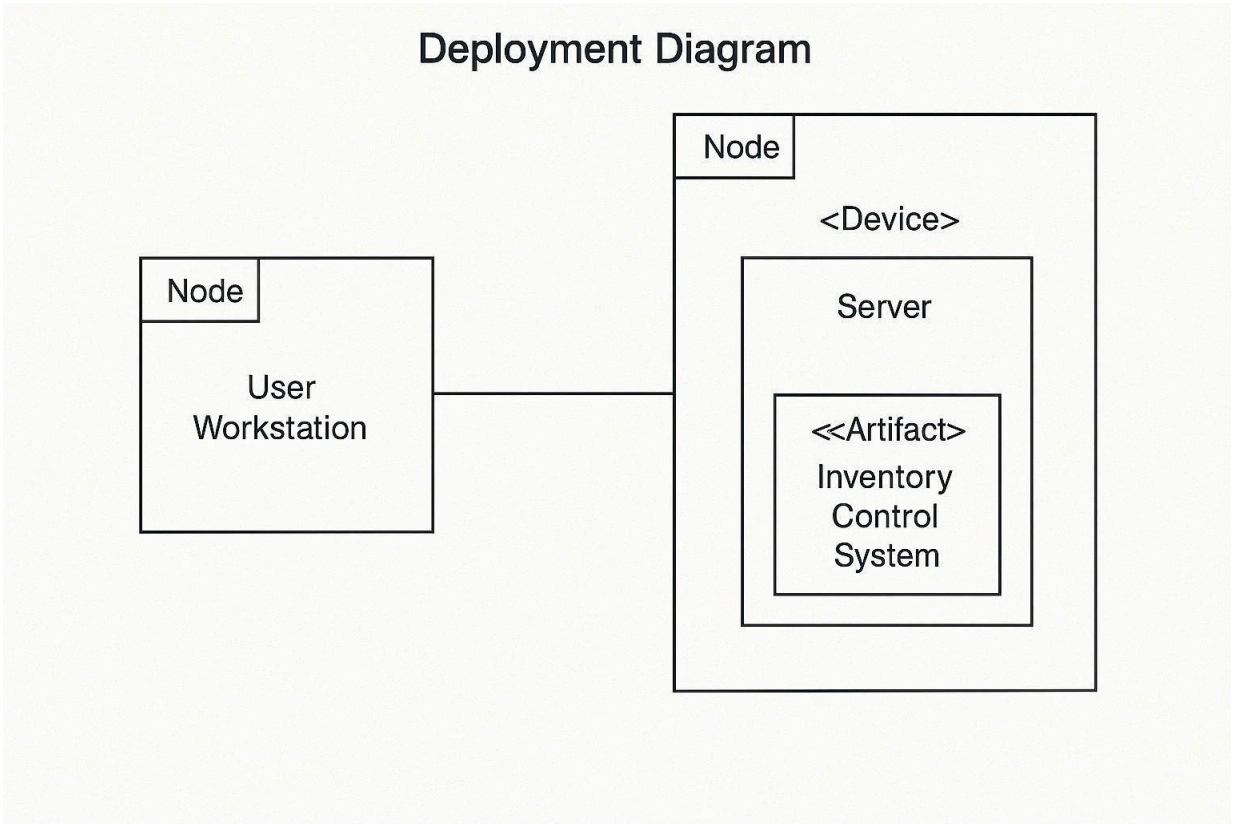
Class Diagram:



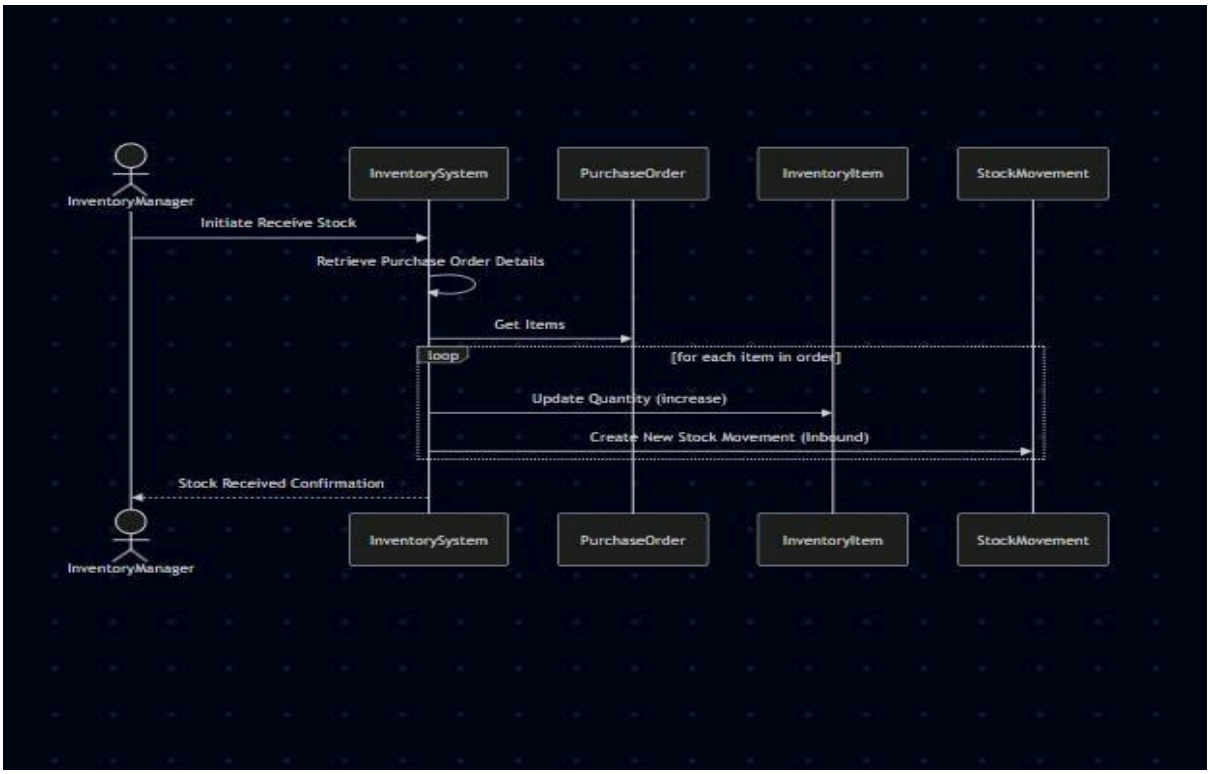
State Chart Diagram:



Deployment Diagram:



Sequence Diagram:



RESULT:

EX. NO:4 **DATE:21/02/2025**

RAILWAY RESERVATION SYSTEM

AIM:

OUTPUT:

Train.java

```
public class Train {
public int trainNo;
public String name;
}
```

Clerk.java

```
public class Clerk {
public int ID;
public void verifyDetails() {
   }
public void cancellation() {
   }
}
```

RailwaySystem.java

```
public class RailwaySystem {
public int ID;
public void responses() {
   }
}
```

Payment.java

```
public class Payment {
   public int amount;
}
```

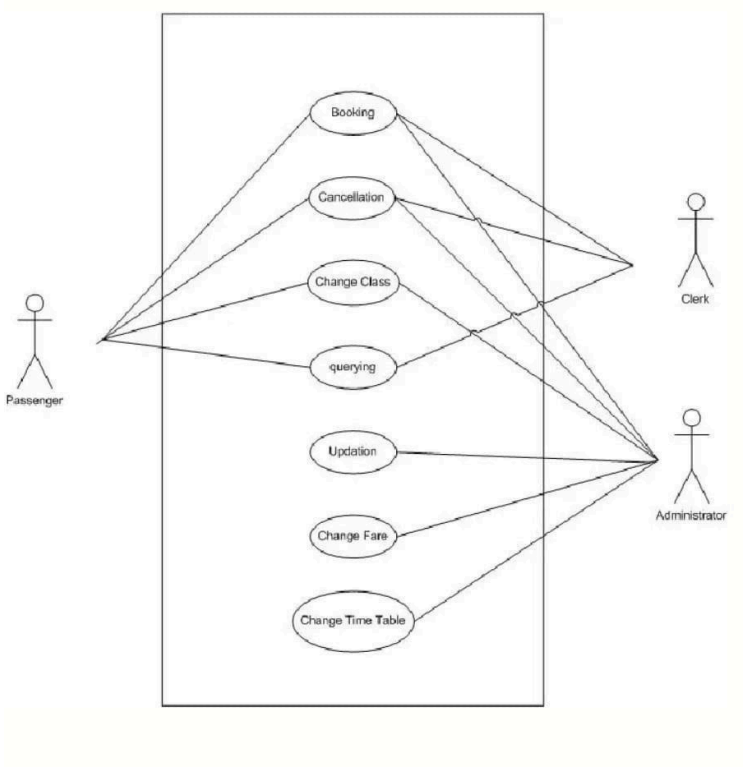
Passenger.java

```
public class Passenger {
public String name;
public String address;
public String gender;
public int date;
public void searchTrain() {
   }
public void bookTrain() {
   }
public void cancelTickets() {
   }
public void payFare() {
   }
}
```

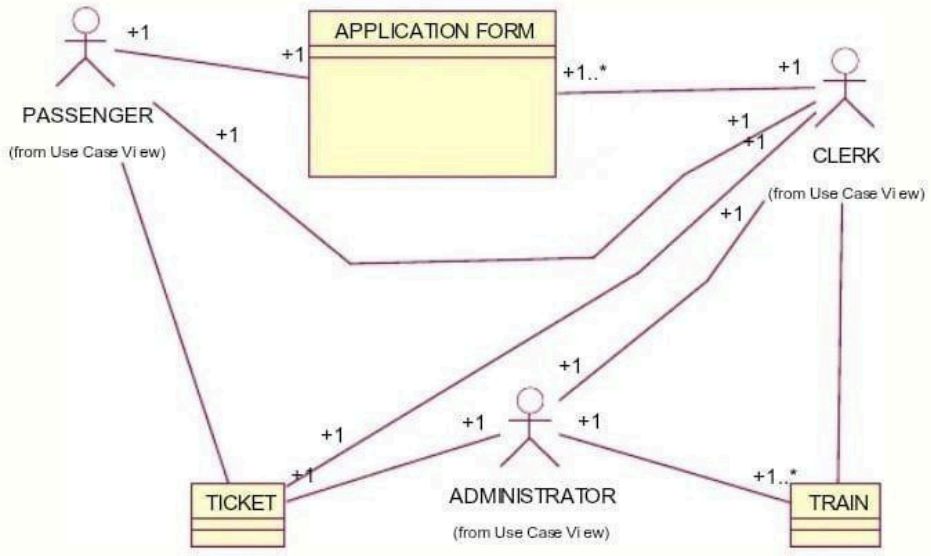
Ticket.java

```
public class Ticket {
public int no;
public String status;
public int noOfPassengers;
public String place;
public void fareAmount() {
   }
public void cancelTickets() {
   }  }
```

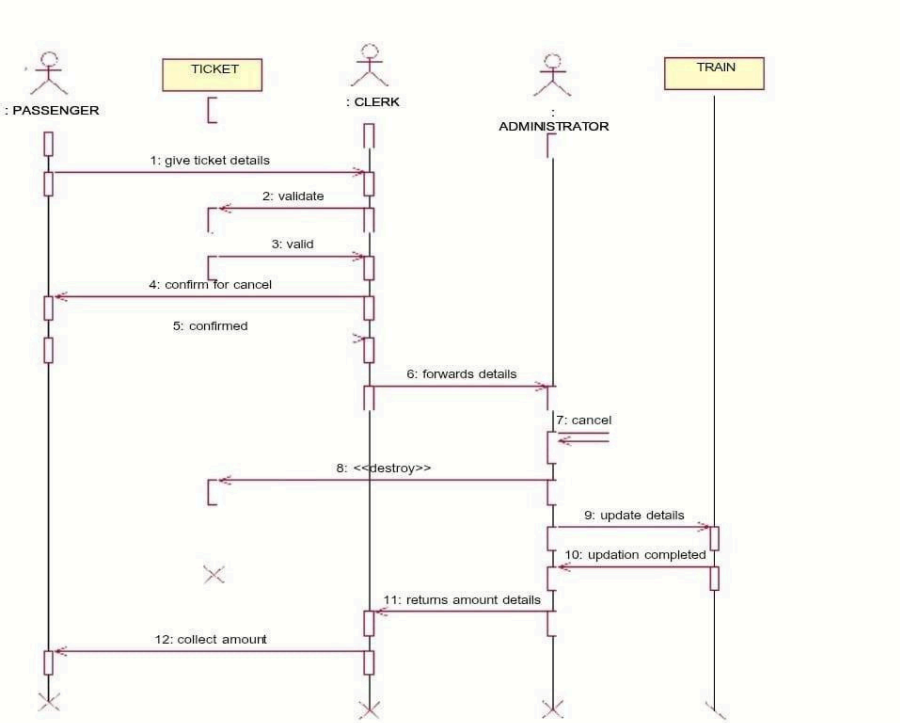
Use Case Diagram:



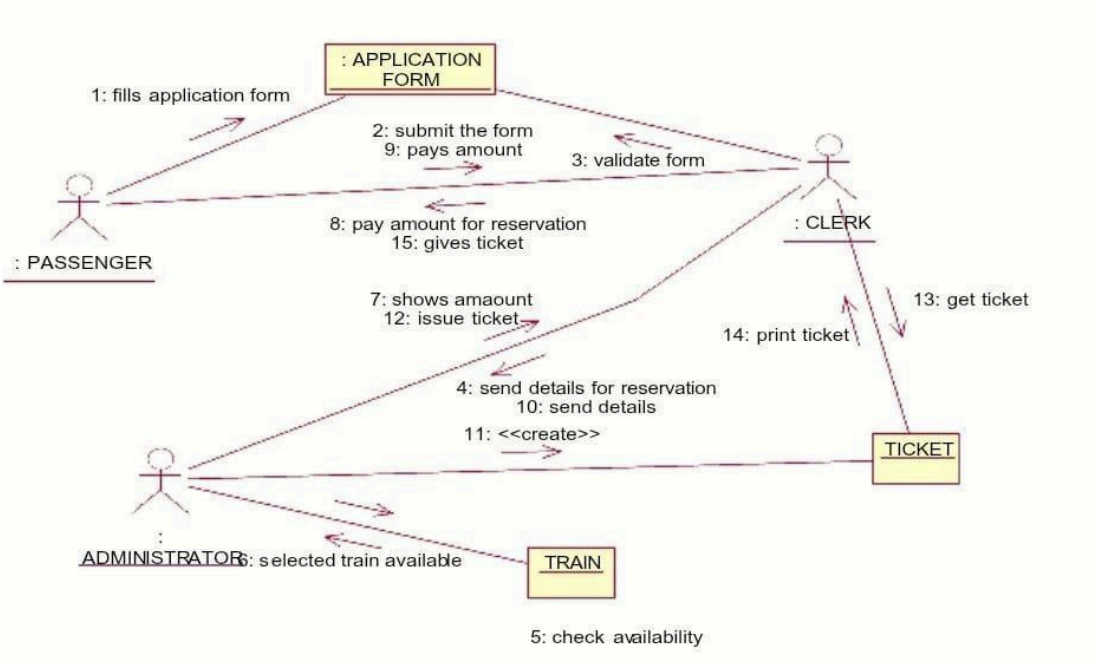
Class Diagram:



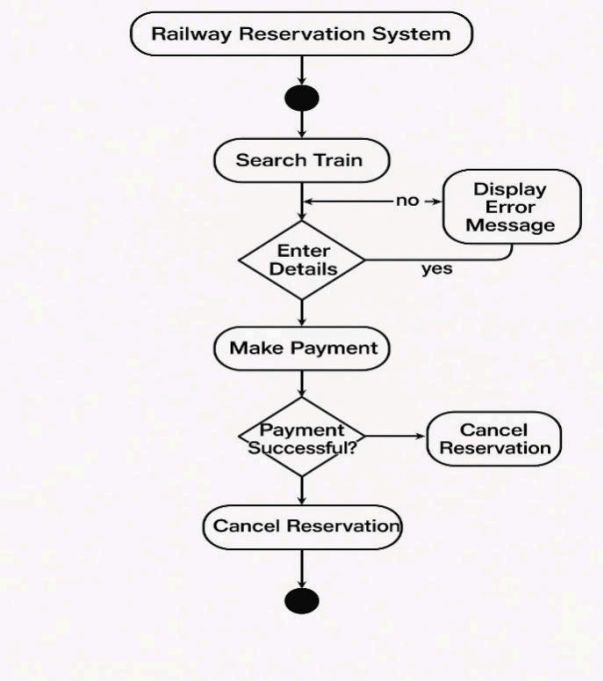
Sequence Diagram:



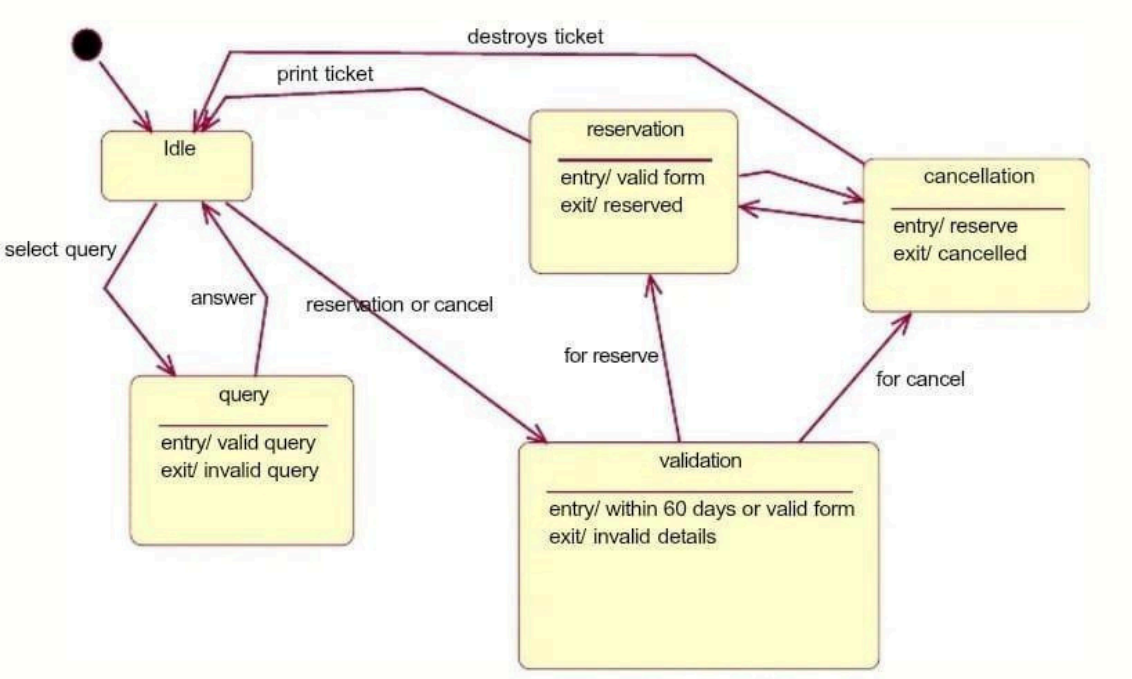
Collaboration Diagram:



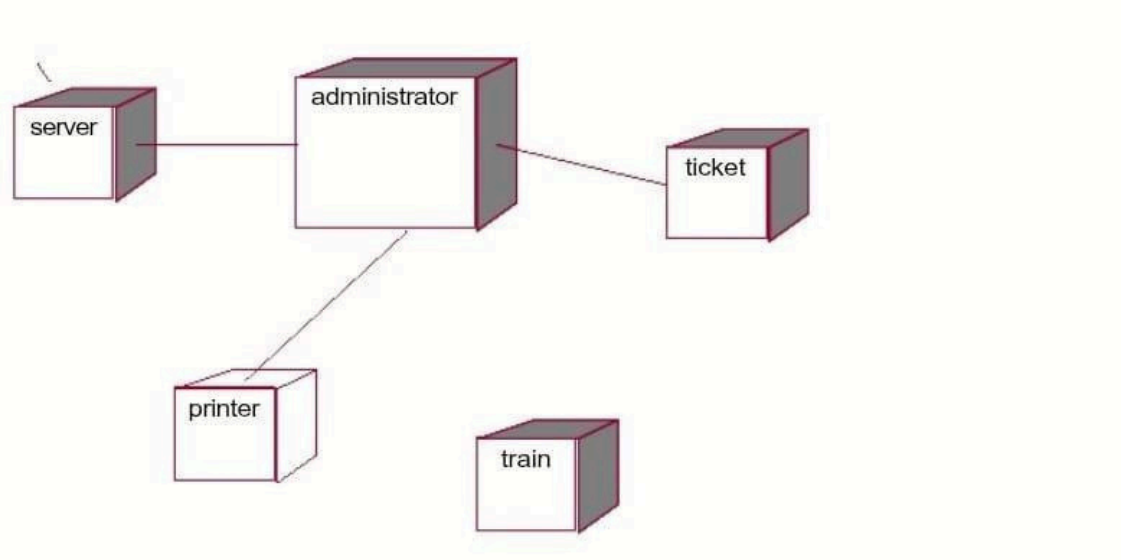
Activity Diagram:



State Chart Diagram:



Deployment Diagram:



RESULT:

EX. NO:5 **DATE:28/02/2025**

HOTEL MANAGEMENT SYSTEM

AIM:

OUTPUT:

Hotel.java

```
import java.util.List;
public class Hotel {
public String name;
public String location;
public void addRoom() {
   }
public void registerCustomer() {
   }
public void makeBooking() {
   }
}
```

Room.java

```
public class Room {
public int roomNumber;
public String type;
public boolean isAvailable;
public double price;
public void checkAvailability() {
   }
public void updateStatus() {
   }
}
```

Customer.java

```
public class Customer {
public int customerId;
public String name;
public String contact;
public void bookRoom() {
   }
public void cancelBooking() {
   }
}
```

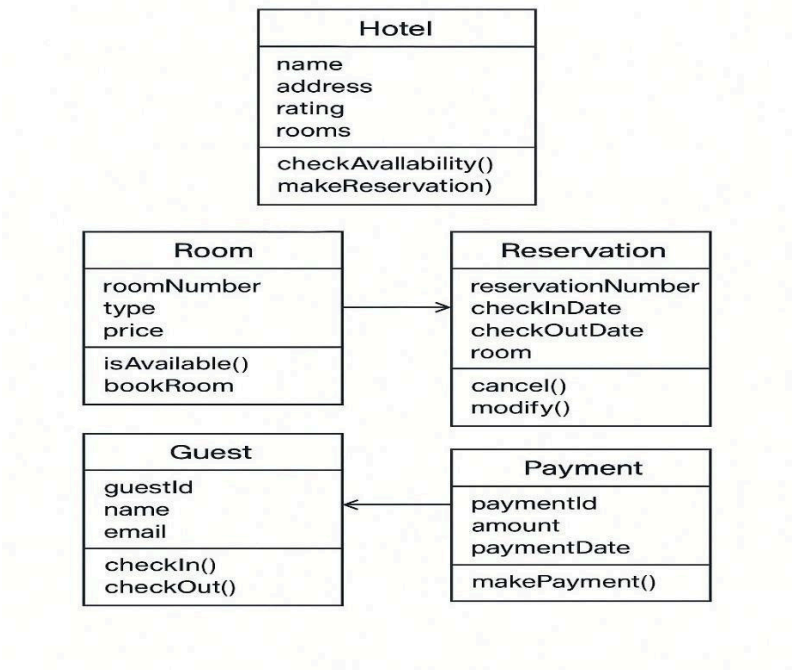
Booking.java

```
import java.util.Date; public class Booking { public int bookingId; public int customerId; public int roomNumber; public Date checkInDate; public Date checkOutDate; public void confirmBooking() {
   }
public void generateInvoice() {
   }
}
```

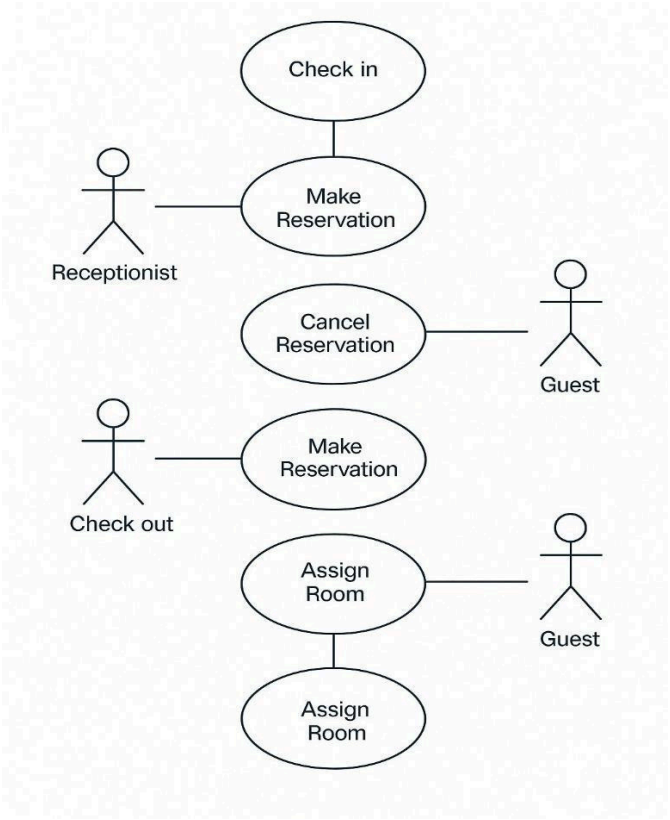
Staff.java

```
public class Staff {

public int staffId;

public String name;

public String role;

public void manageRoomService() {

   }

public void assistCustomer() {

   }

}
```

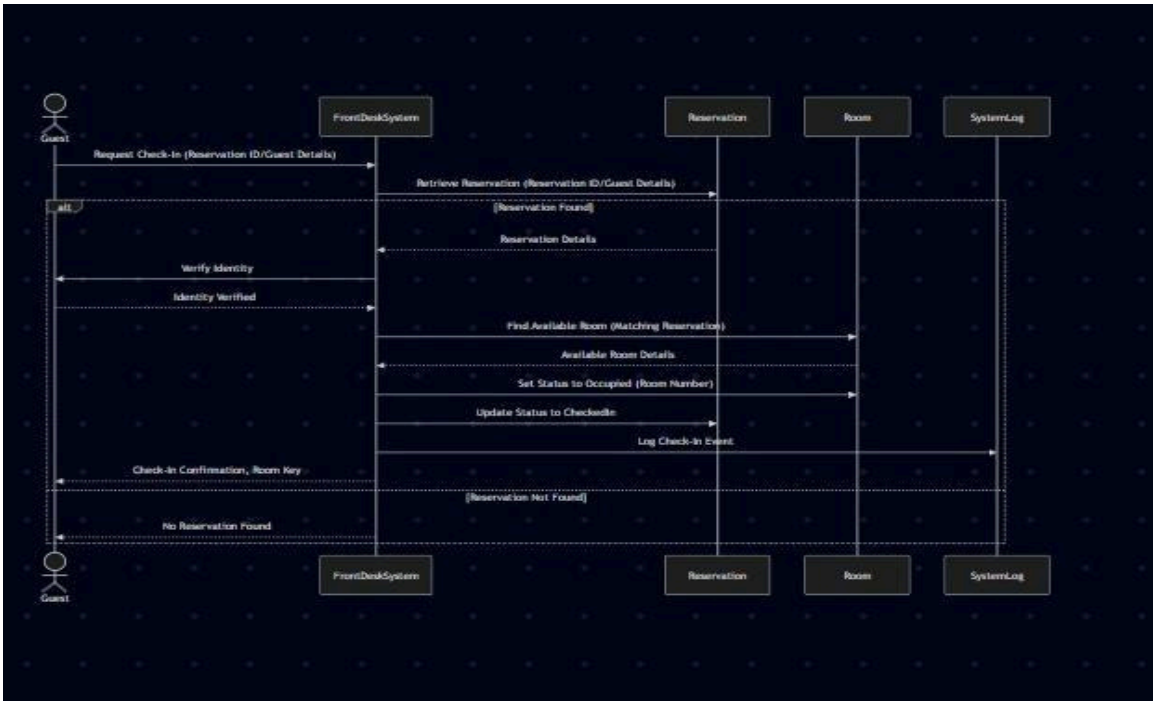
Class Diagram:



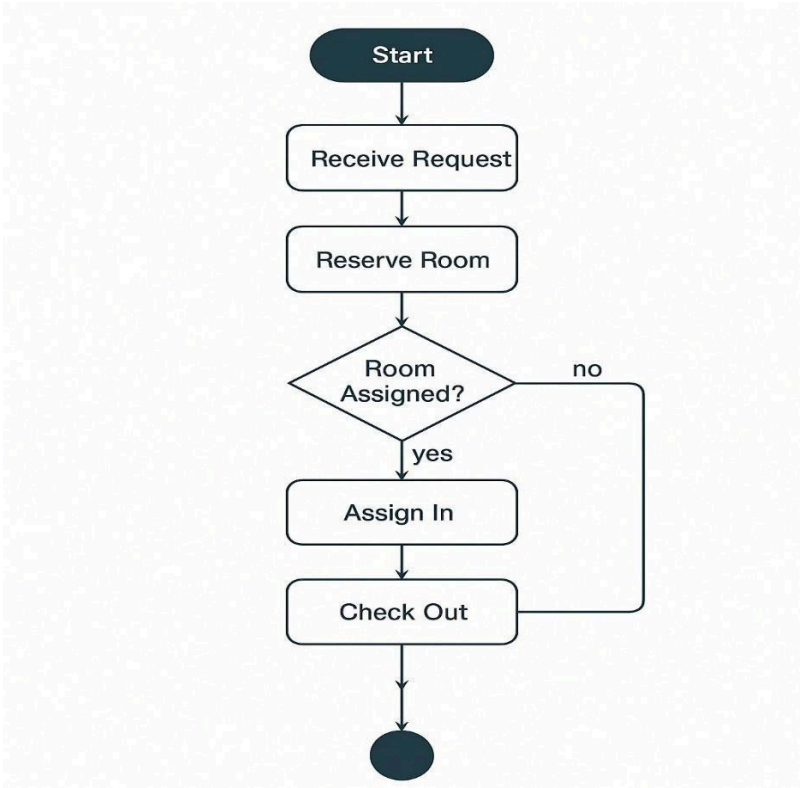
Use Case Diagram:



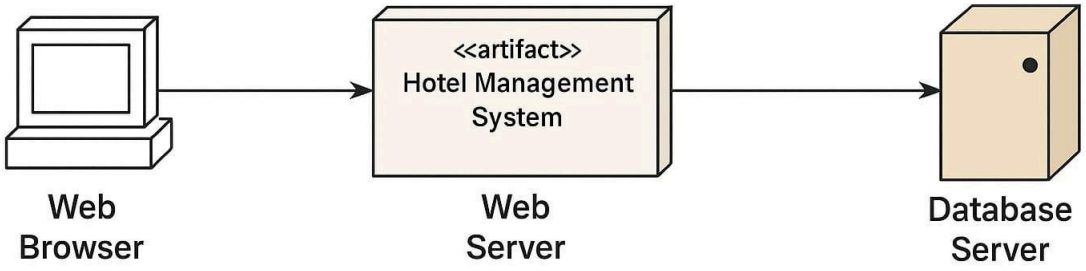
Sequence Diagram:



Activity Diagram:



Deployment Diagram:



Hotel Management System

RESULT:

EX. NO:6 **DATE:07/03/2025**

AUTOMATING THE BANKING SYSTEM

AIM:

OUTPUT:

User.java

```
import java.util.Date;
public class User {
public String username;
public String password;
public Date dob;
public int ac_no;
public void read() {
}
public void write() {
}
}
```

ATMmc.java

```
public class ATM_mc {
public int limit;
public int withdraw;
public int tbalance;
public void login_verify() {

   }
public void check_mc() {
   }
public void confirmation() {
   }
}
```

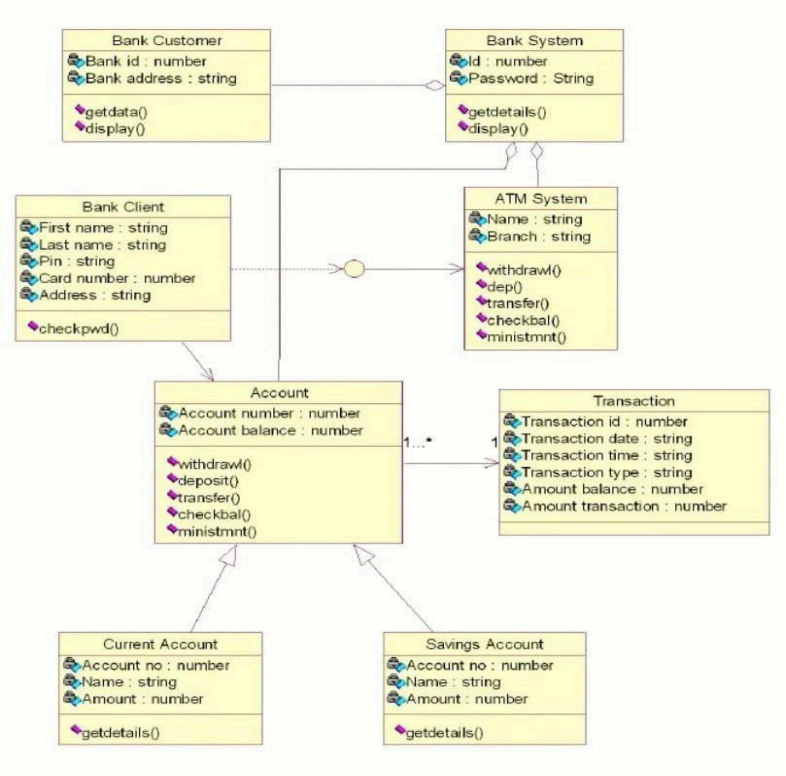
Account.java

```
public class Account {
public int ac_no;
public int balance;
public void withdraw() {
   }
public void deposit() {
   }
public void display_avail() {
   }
}
```

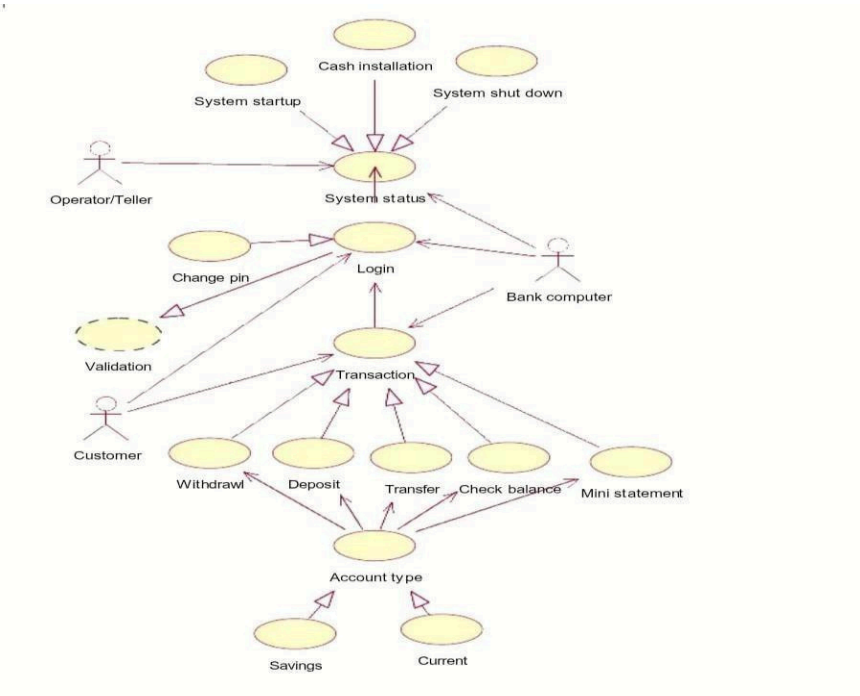
Transaction.java

```
import java.util.Date;
public class Transaction {
public int ac_no;
public Date date;
public String type;
public int amt;
public int balance;
public void ministmt() {
   }
public void display() {
   }
public void update() {
 }
```

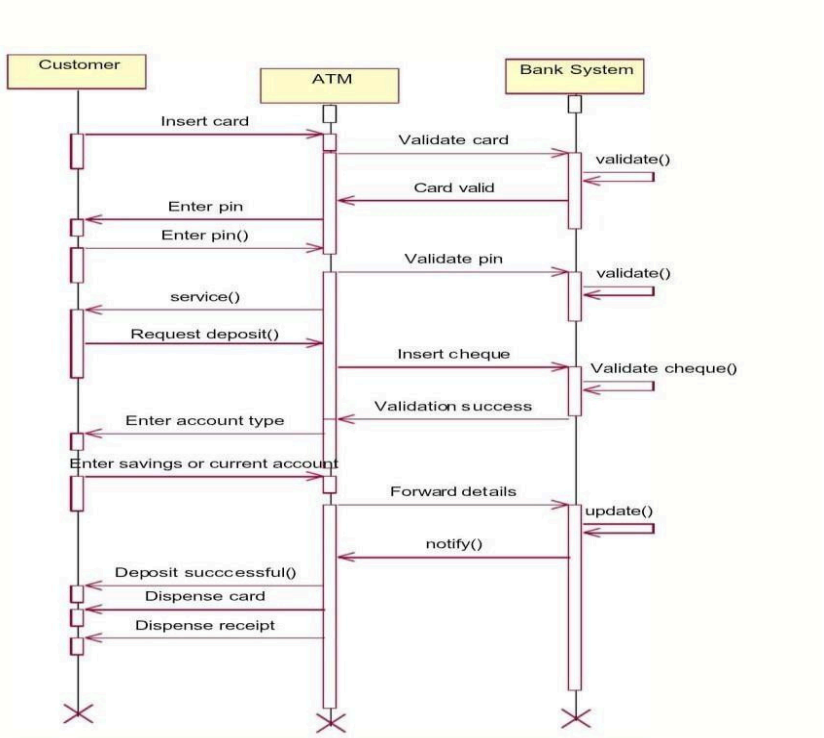
Class Diagram:



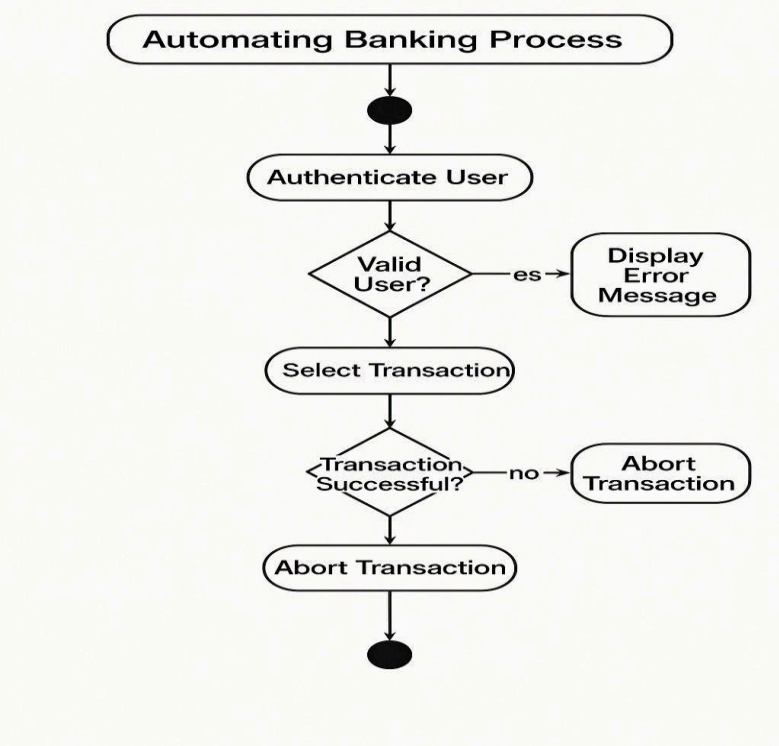
Use case Diagram:



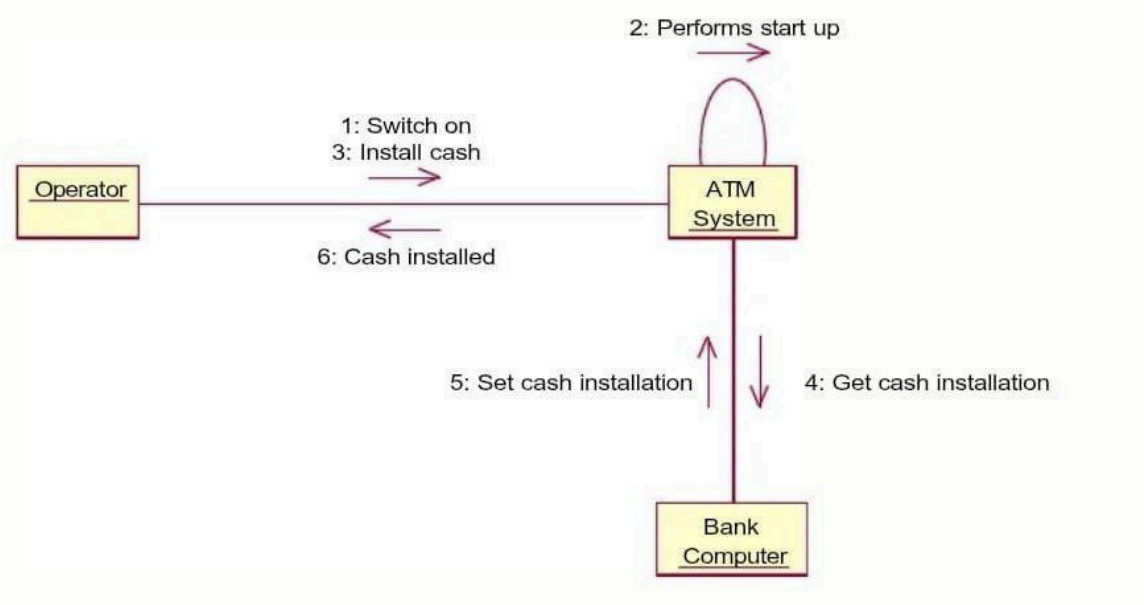
Sequence Diagram:



Activity Diagram:



Collaboration Diagram:



RESULT:

EX. NO:7 **DATE:21/03/2025**

LIBRARY MANAGEMENT SYSTEM

AIM:

OUTPUT:

Book.java

```
public class Book {
public int bookId;
public String author;
public String name;
public float price;
public String rackNo;
public String status;
public String edition;
public String dateOfPurchase;
public void displayBookDetails() {
   }
public void updateStatus() {
   }
}
```

Journals.java

```
public class Journals extends Book {
}
```

Magazines.java

```
public class Magazines extends Book {
}
```

Librarian.java

```
public class Librarian {
public String name;
public String password;
public void searchBook() {
   }
public void verifyMember() {
   }
public void issueBook() {
   }
public void calculateFine() {
   }
public void createBill() {
   }
public void returnBook() {
   }
}
```

Students.java

```
public class Student extends MemberRecord {
}
```

Faculty.java

```
public class Faculty extends MemberRecord {
}
```

MemberRecord.java

```
public class MemberRecord {
public int memberId;
public String type;
public String dateOfMembership;
public int noOfBooksIssued;
public int maxBookLimit;
public String name;
public String address;
public String phoneNo;
public void retrieveMember() {
   }
public void increaseBookIssued() {
   }
public void decreaseBookIssued() {
   }
public void payBill() {
   }
}
```

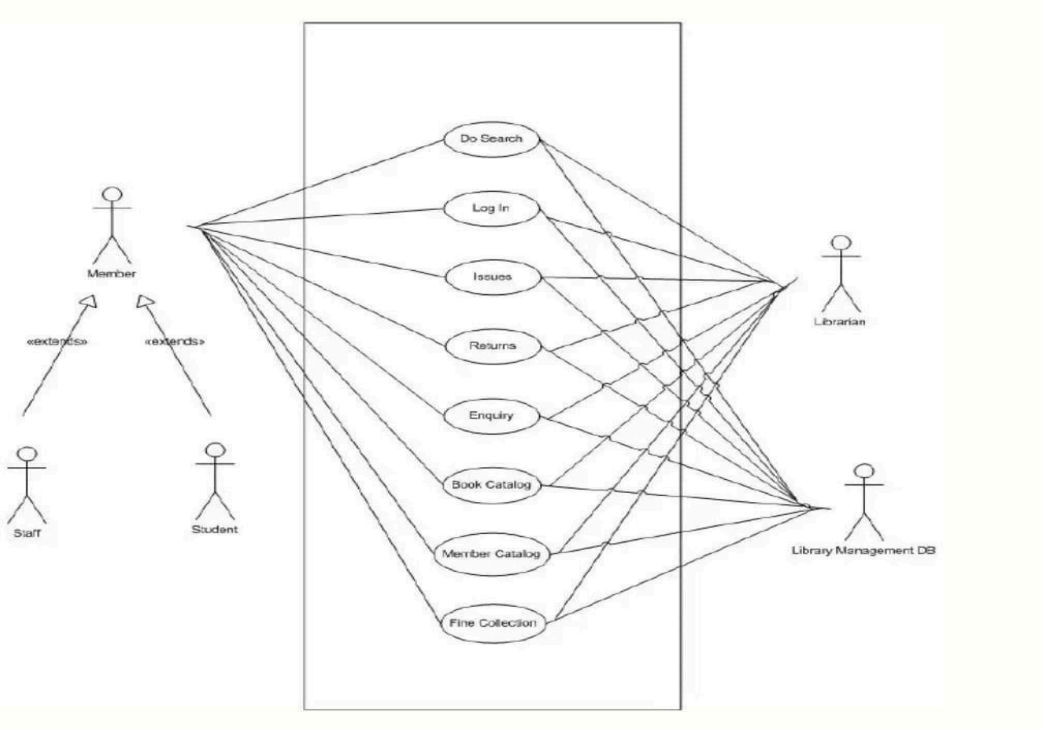
Transaction.java

```
public class Transaction {
public int transId;
public int bookId;
public int memberId;
public String dateOfIssue;
public String dueDate;
public void createTransaction() {
   }
public void deleteTransaction() {
   }
public void retrieveTransaction() {
   }
}
```

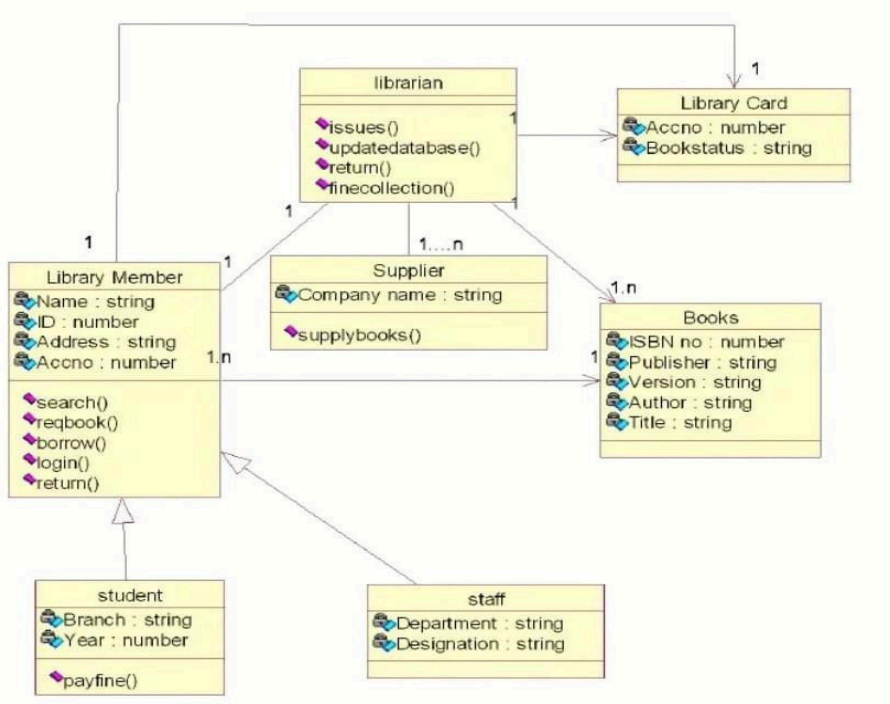
Bill.java

```
public class Bill {
public int billNo;
public String date;
public int memberId;
public float amount;
public void billCreate() {
   }
public void billUpdate() {
   }
}
```

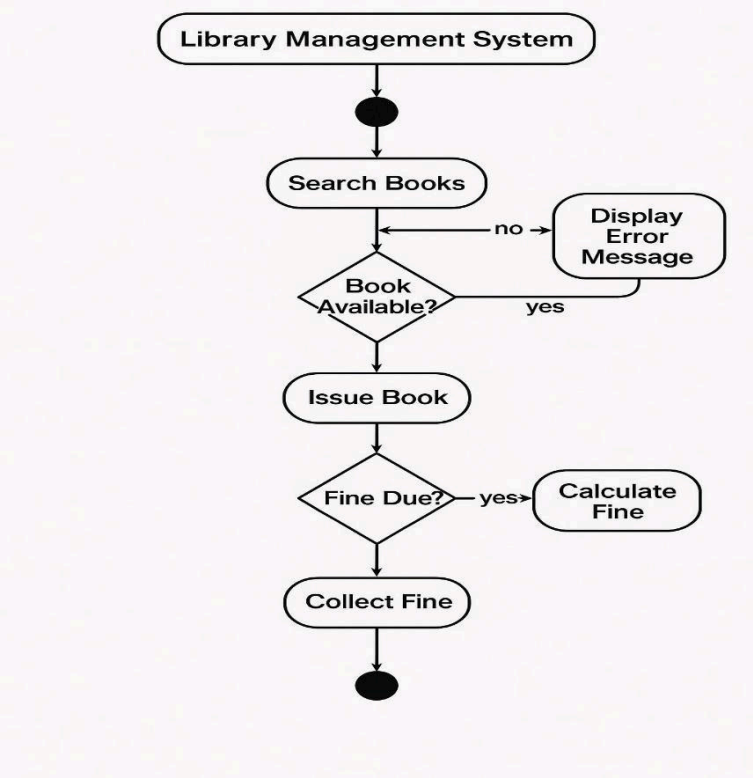
Use case Diagram:



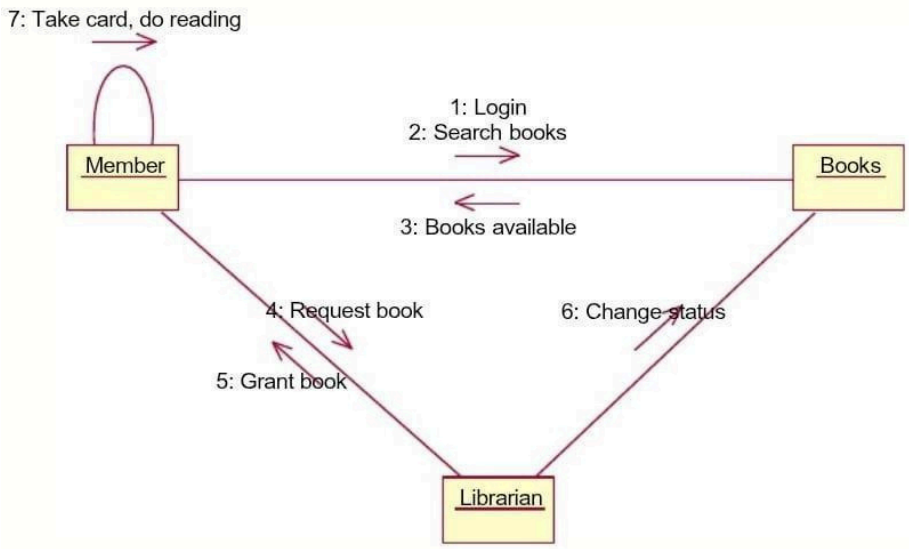
Class Diagram:



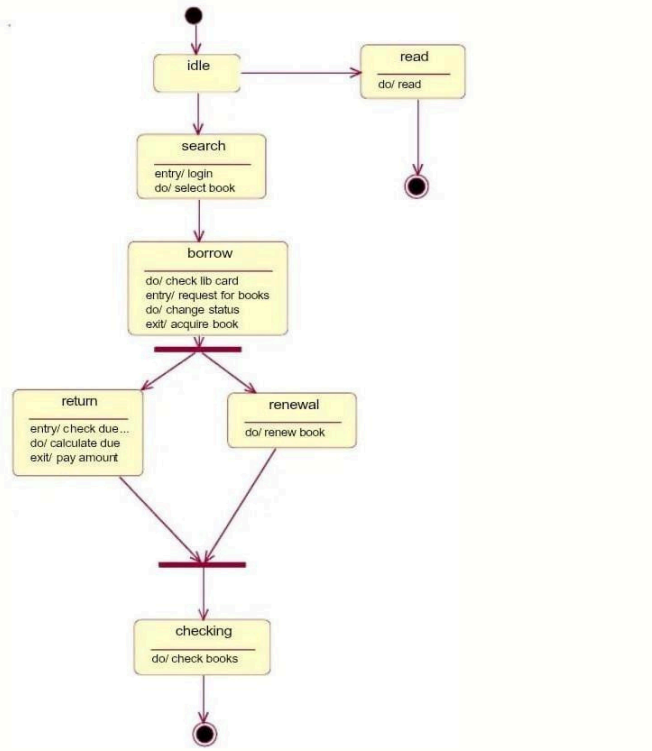
Activity Diagram:



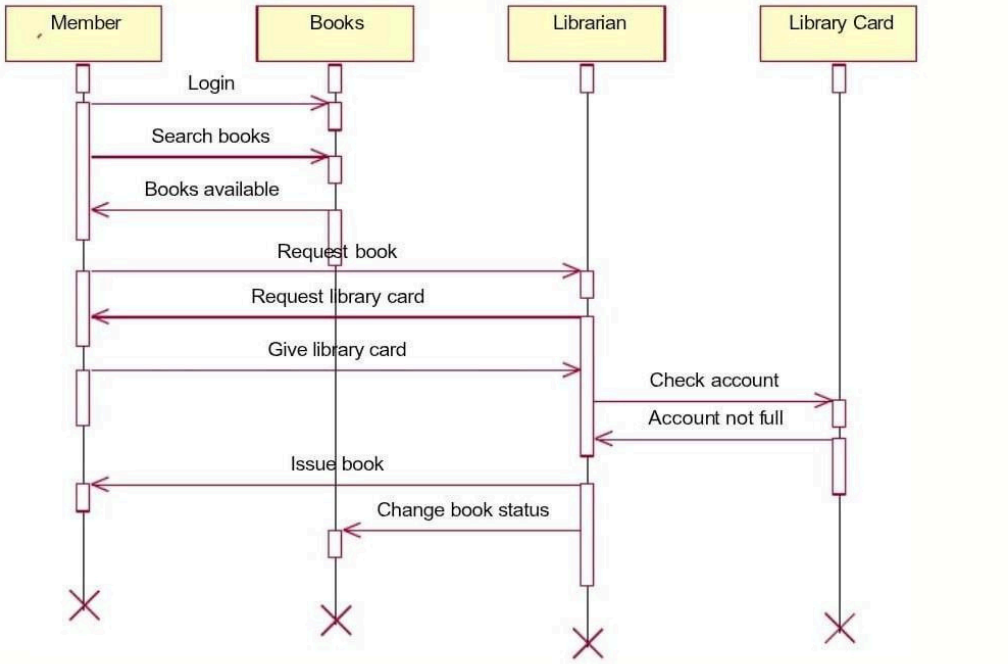
Collaboration Diagram:



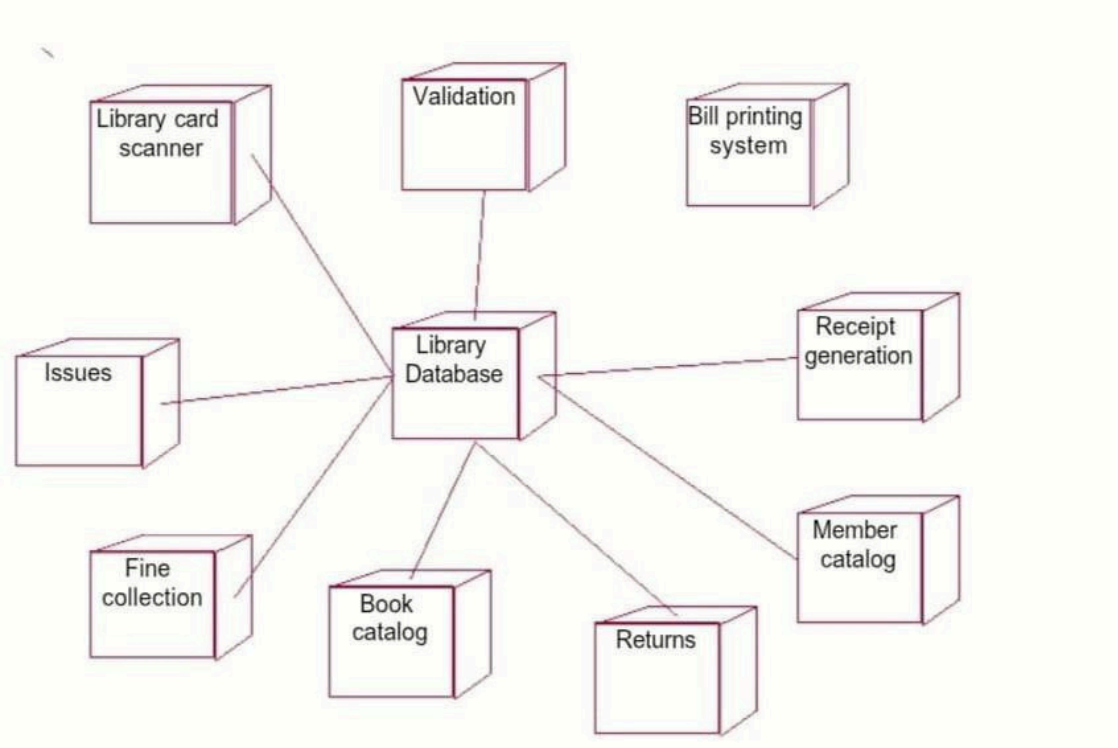
State Chart Diagram:



Sequence Diagram:



Deployment Diagram:



RESULT:

EX. NO:8 **DATE:28/03/2025**

PASSPORT AUTOMATION SYSTEM

AIM:

OUTPUT:

Applicant.java

```
public class Applicant {
private String name;
private String fatherName;
private String dateOfBirth;
private String permanentAddress;
private String temporaryAddress;
private String emailID;
private String phoneNumber;
private String panNo;
private String applicationNo;
private String userName;
private String password;
public void login() {
}
public void submitDetails() {
}
public void checkStatus() {
}
}
```

Database.java

```
public class Database {
private String name;
public void store() {
  }
}
```

PassportAdministrator.java

```
public class PassportAdministrator {
private String username; private
String password; public void login() {

   }
public void verify() {
   }
 public void update() {
   }
}
```

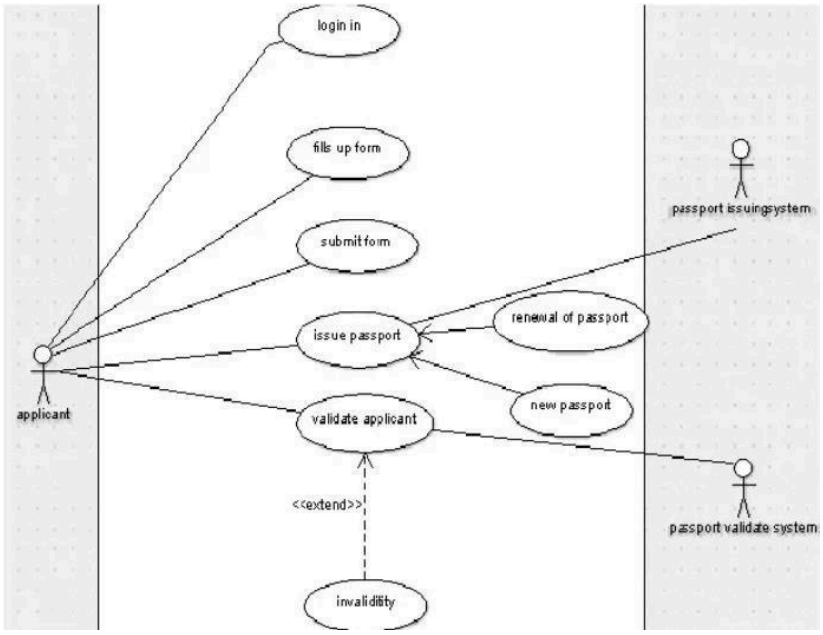
RegionalAdministrator.java

```
public class RegionalAdministrator {
private String username;
private String password;
public void login() {
   }
public void verify() {
   }
public void update() {
   }
}
```

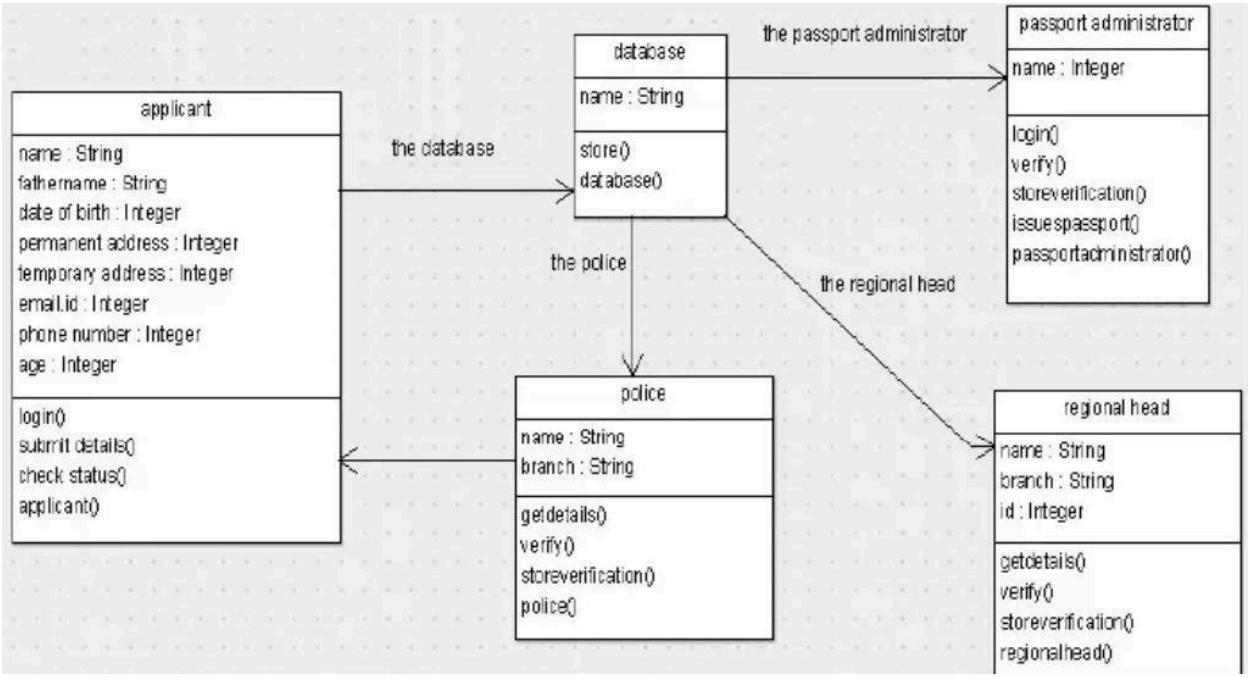
Police.java

```
public class Police {
private String username;
private String password;
public void login() {
   }
public void verify() {
   }
public void update() {
   }
}
```

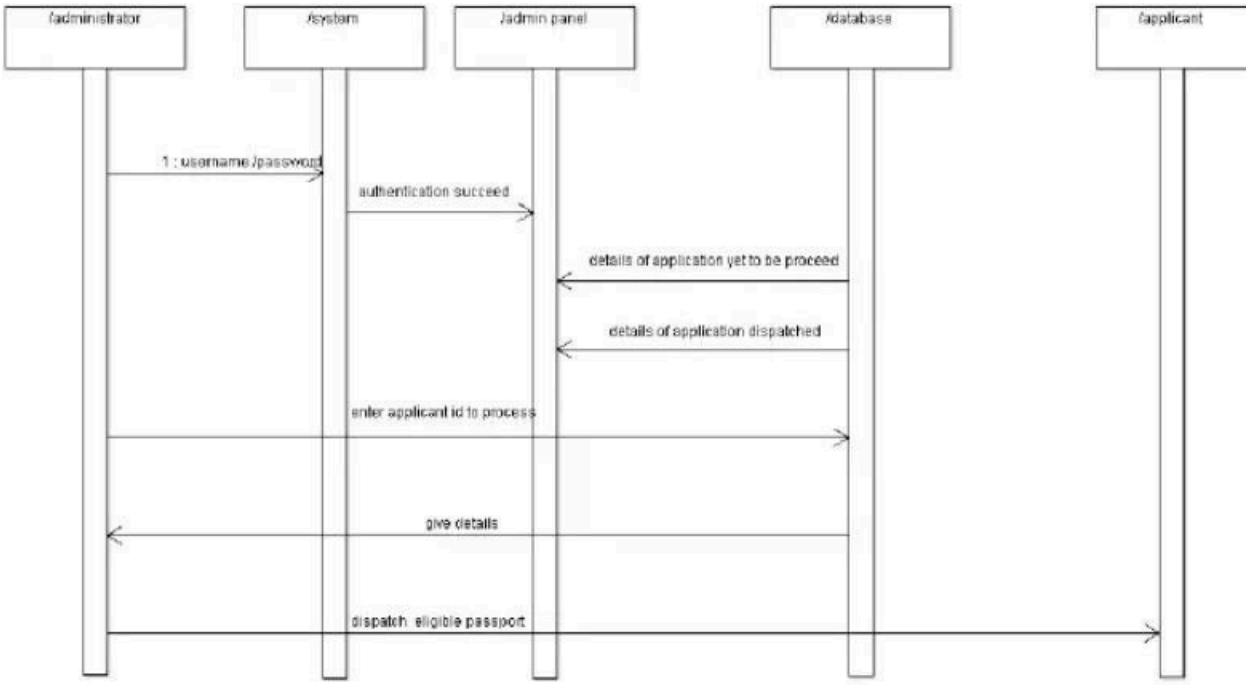
Use-case Diagram:



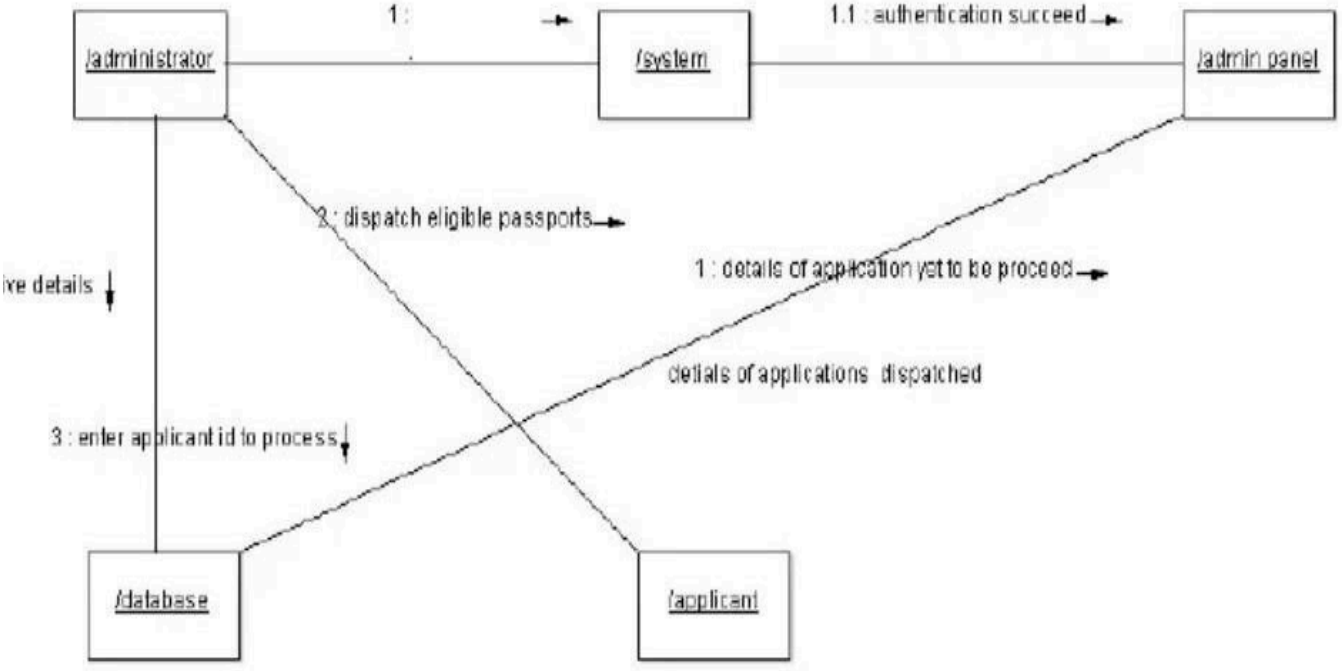
Class Diagram:



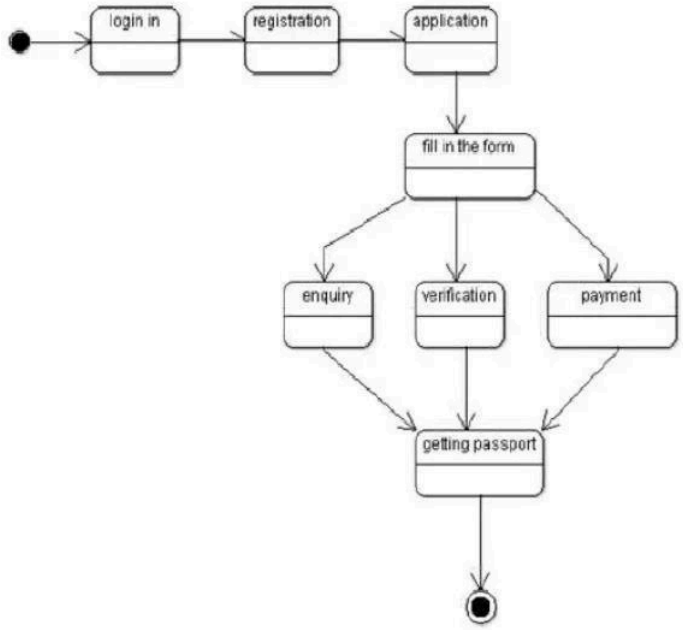
Sequence Diagram:



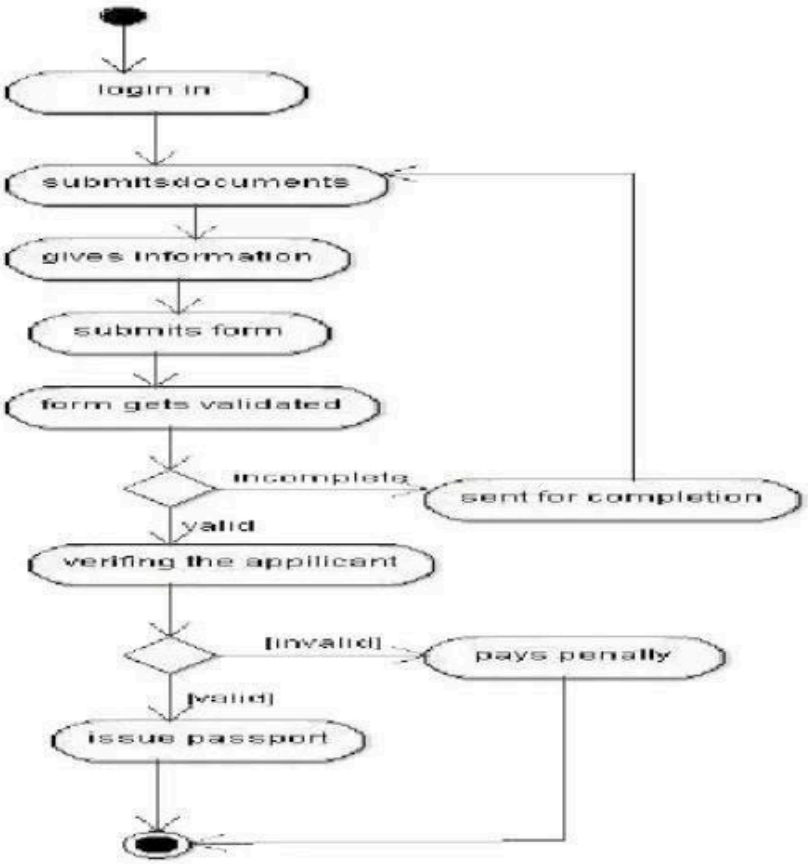
Collaboration Diagram:



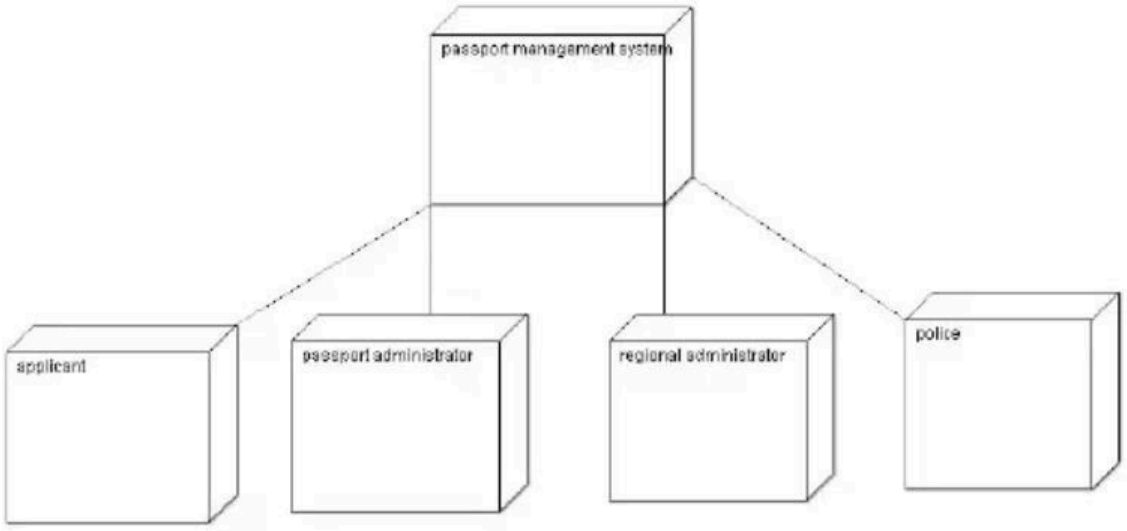
State Chart Diagram:



Activity Diagram:



Deployment Diagram:



RESULT:

EX. NO:9 **DATE:04/04/2025**

E-TICKETING

AIM:

OUTPUT:

Airline.java

```
import java.util.List;
public class Airline {
public String name;
public String code;
public void addFlight() {
   }
public void manageBooking() {
   }
}
```

Flight.java

```
import java.util.Date
public class Flight {
public String flightNumber;
public String source;
public String destination;
public Date departureTime;
public Date arrivalTime;
public void checkAvailability() {
   }
public void updateSchedule() {
   }
}
```

Passenger.java

```
public class Passenger {
   public int passengerId;
   public String name;
   public String passportNumber;
   public String contact;
public void bookTicket() {
   }
public void cancelTicket() {
   }
}
```

Ticket.java

```
public class Ticket {
   public int ticketId;
   public String flightNumber;
   public int passengerId;
   public Date bookingDate;
   public double fare;
    public void generateTicket() {
   }
public void viewTicket() {
   }
}
```

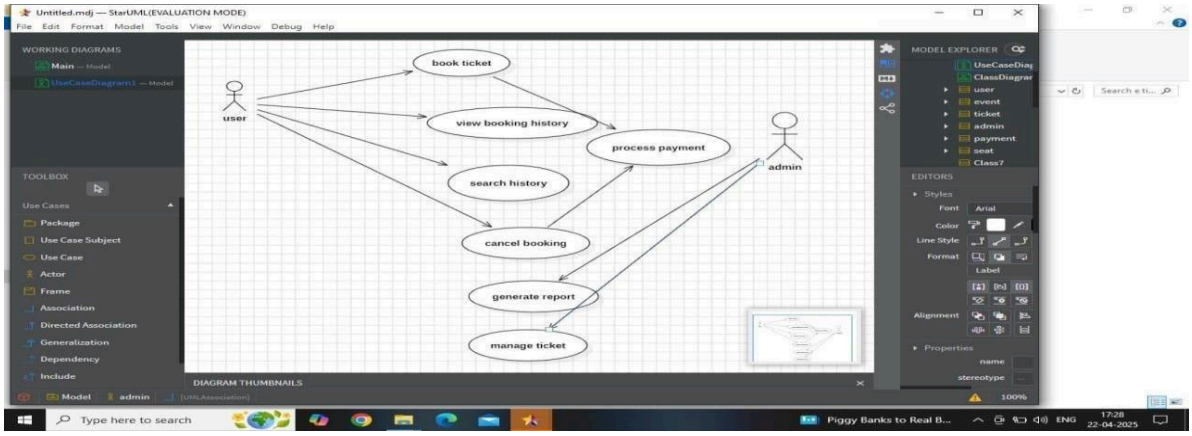
Payment.java

```
public class Payment {

   public int paymentId;

   public int ticketId;

   public double amount;

   public String paymentMode;

public void makePayment() {

   }

public void refund() {

   }

}
```

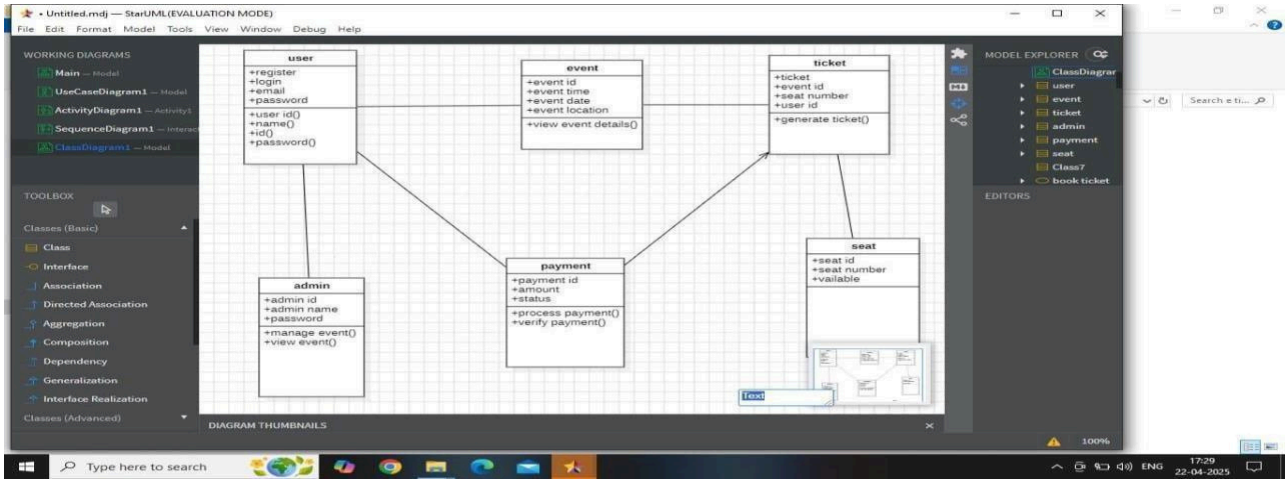
Admin.java

```
public class Admin {

   public     int     adminId;

   public  String  username;

   public String password;

 public void manageFlights() {

   }

public void managePassengers() {

   }
```

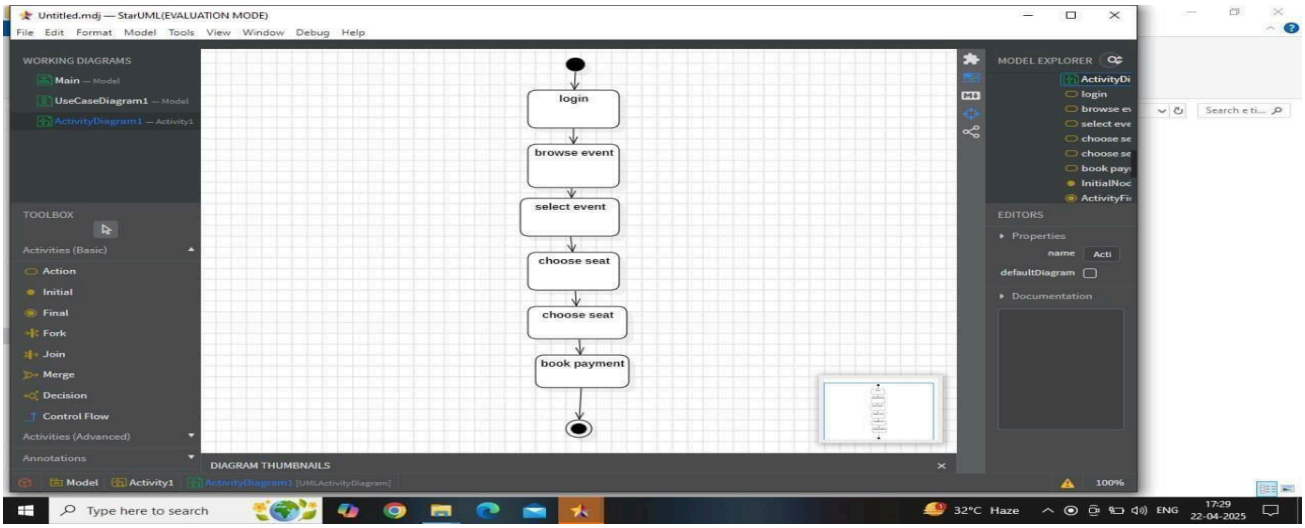
Use-case Diagram:



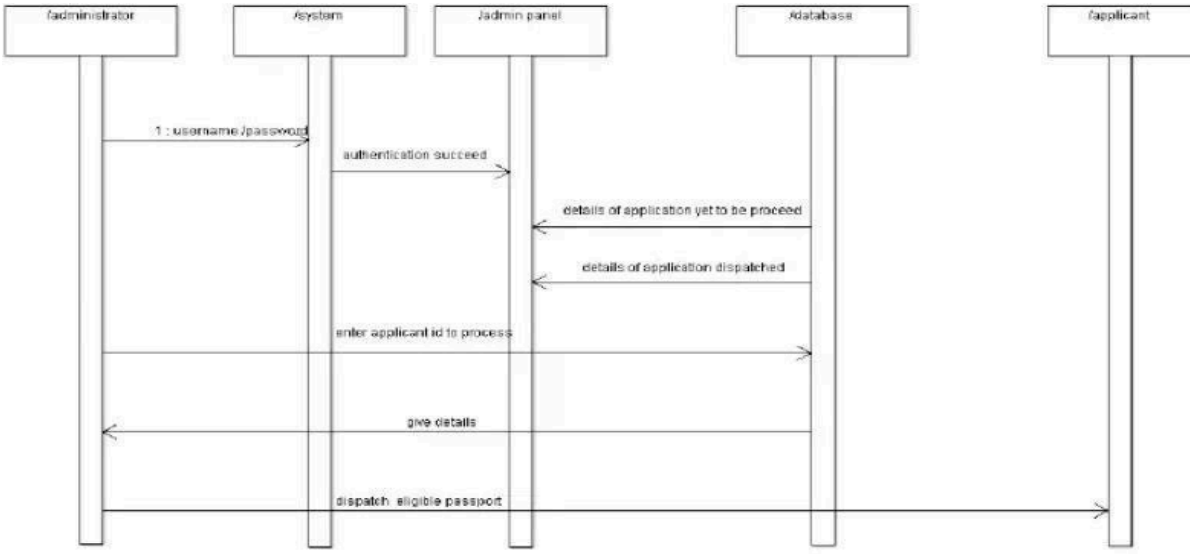
Class Diagram:



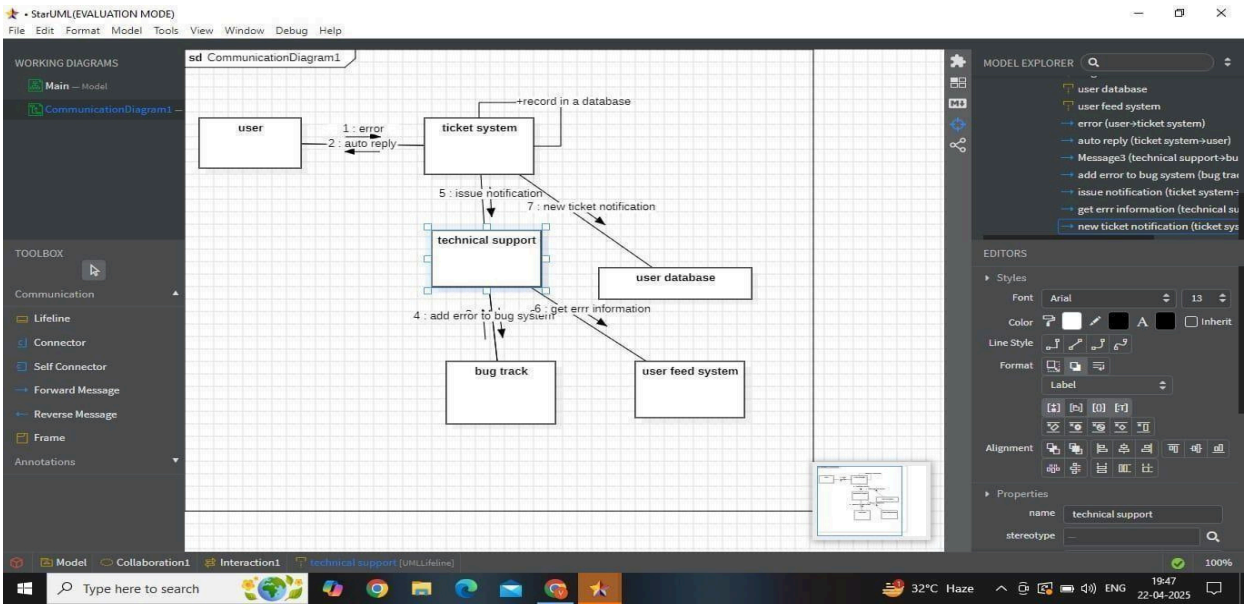
Activity Diagram:



Sequence Diagram:



Collaboration Diagram:



RESULT: