Exp 6:

E-TICKETING

AIM:

To draw the diagrams[use case, activity, sequence, collaboration, class, state chart,

component, deployment, package] for the E-ticketing system.

SOFTWARE REQUIREMENTS SPECIFICATION

SL.NO SOFTWARE REQUIREMENTS SPECIFICATION

1.0 Hardware Requirements

1.1 Software Requirements

1.2 Problem Analysis and Project Plan

1.3 Project description

1.4 Reference

1.0 HARDWARE REQUIREMENTS:

 Intel Pentium Processor I3/I5

1.1 SOFTWARE REQUIREMENTS:

Rational rose / Argo UML

1.2 PROBLEM ANALYSIS AND PROJECT PLANNING

In the E-Ticketing system the main process is a applicant have to login the database then

the database verifies that particular username and password then the user must fill the details about

their personal details then selecting the flight and the database books the ticket then send it to the

applicant then searching the flight or else cancelling the process

1.3 PROJECT DESCRIPTION:

This software is designed for supporting the computerized e-ticketing. This is widely used
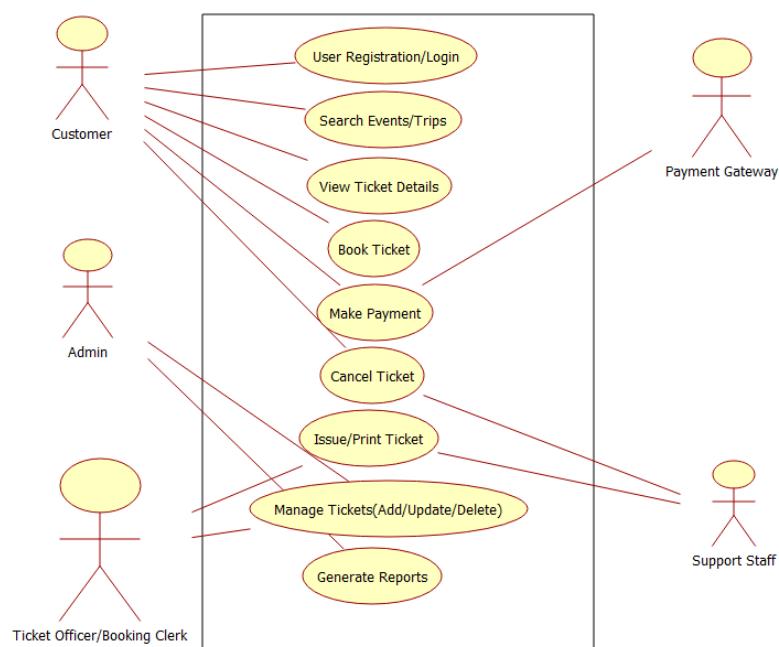
by the passenger for reserving the tickets for their travel. This E-ticketing is organized by the

central system. The information is provided from the railway reservation system.
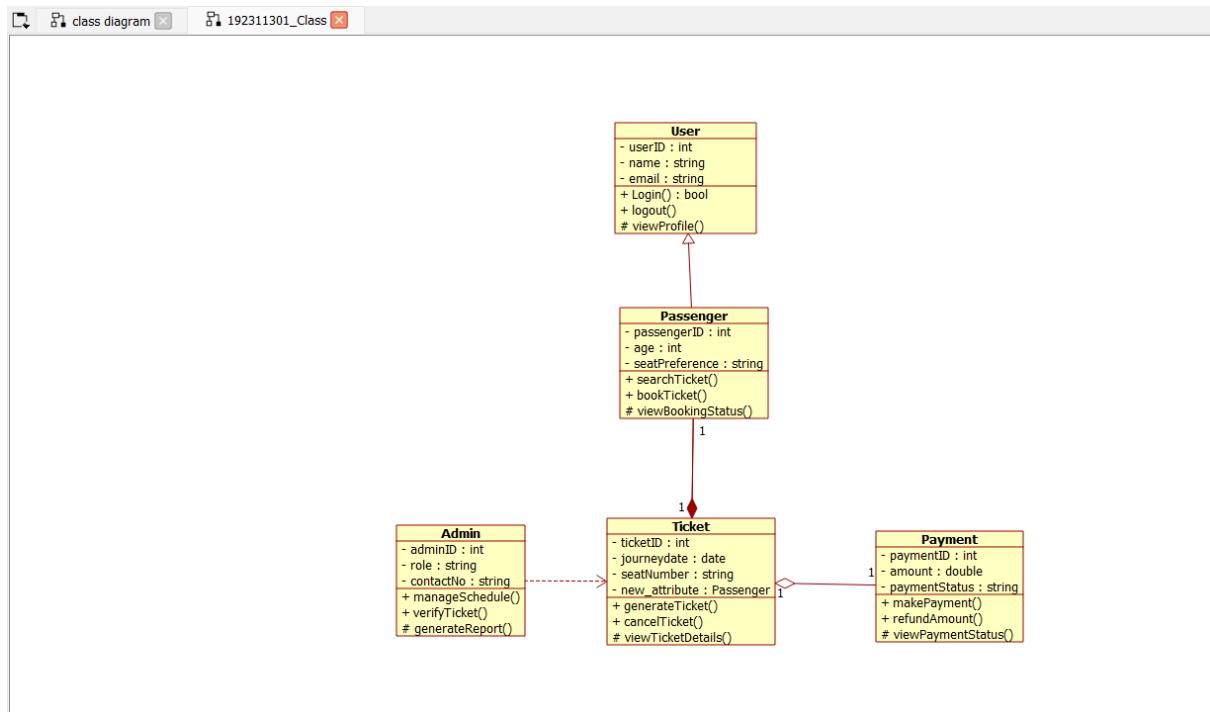
1.4 REFERENCES:

IEEE Software Requirement Specification format.
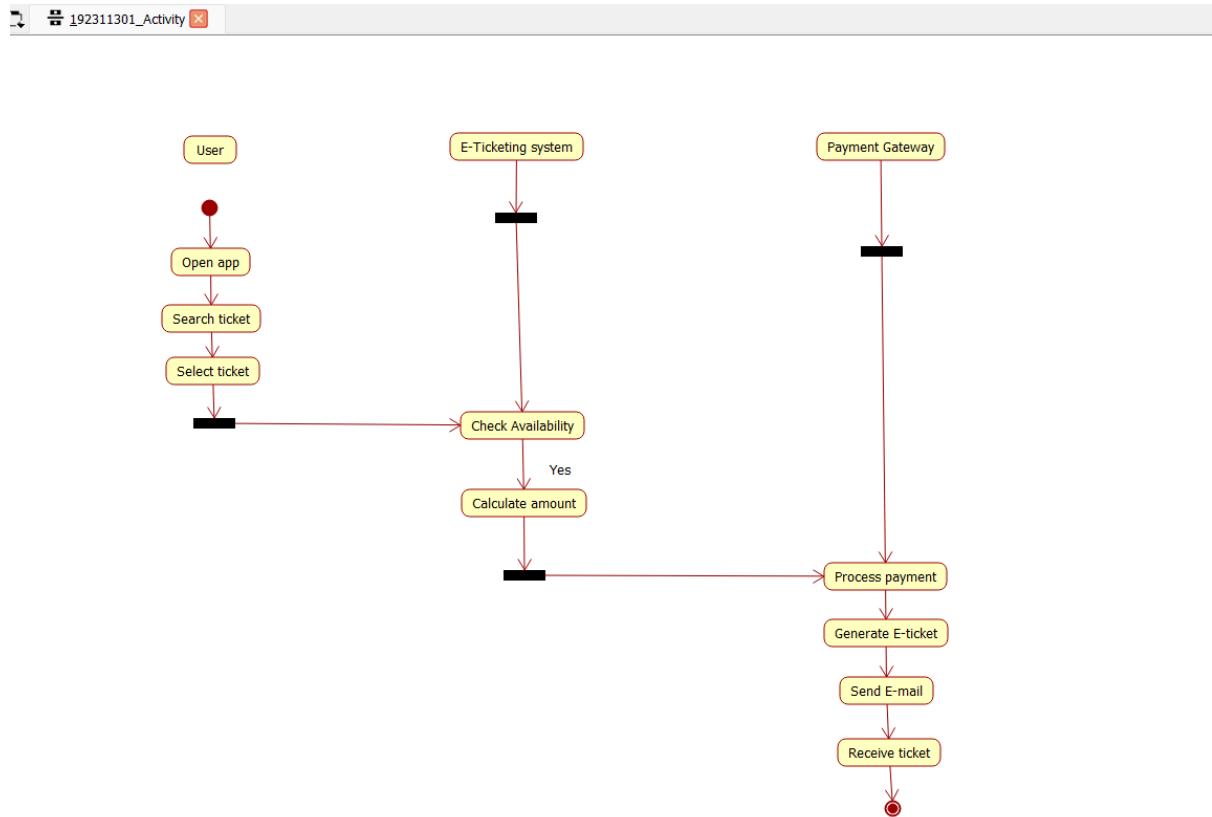
Usecase Diagram:

Class Diagram:

User
- userID : int
- name : string
- email : string
+ Login() : bool
+ logout()
# viewProfile()

Passenger
- passengerID : int
- age : int
- seatPreference : string
+ searchTicket()
+ bookTicket()
# viewBookingStatus()

1

1

Admin
- adminID : int
- role : string
- contactNo : string
+ manageSchedule()
+ verifyTicket()
# generateReport()

Ticket
- ticketID : int
- journeydate : date
- seatNumber : string
- new_attribute : Passenger
+ generateTicket()
+ cancelTicket()
# viewTicketDetails()

1

Payment
- paymentID : int
- amount : double
- paymentStatus : string
+ makePayment()
+ refundAmount()
# viewPaymentStatus()

Activity Diagram :

Sequence Diagram:
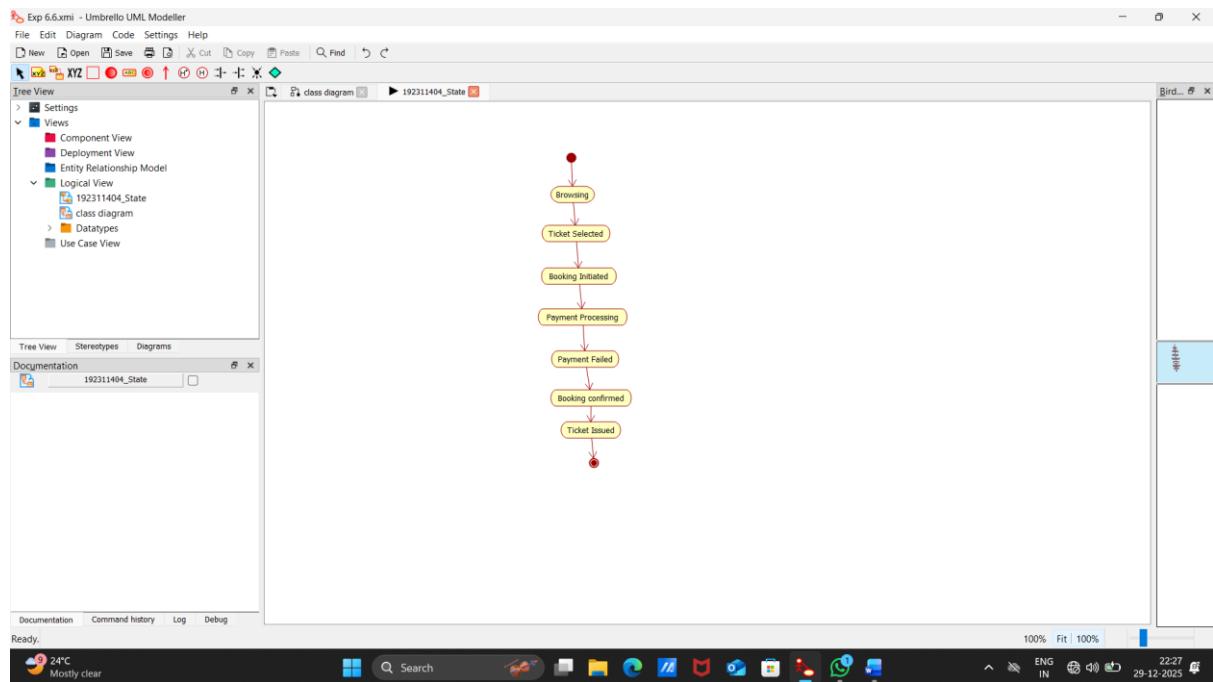
Collaboration diagram:
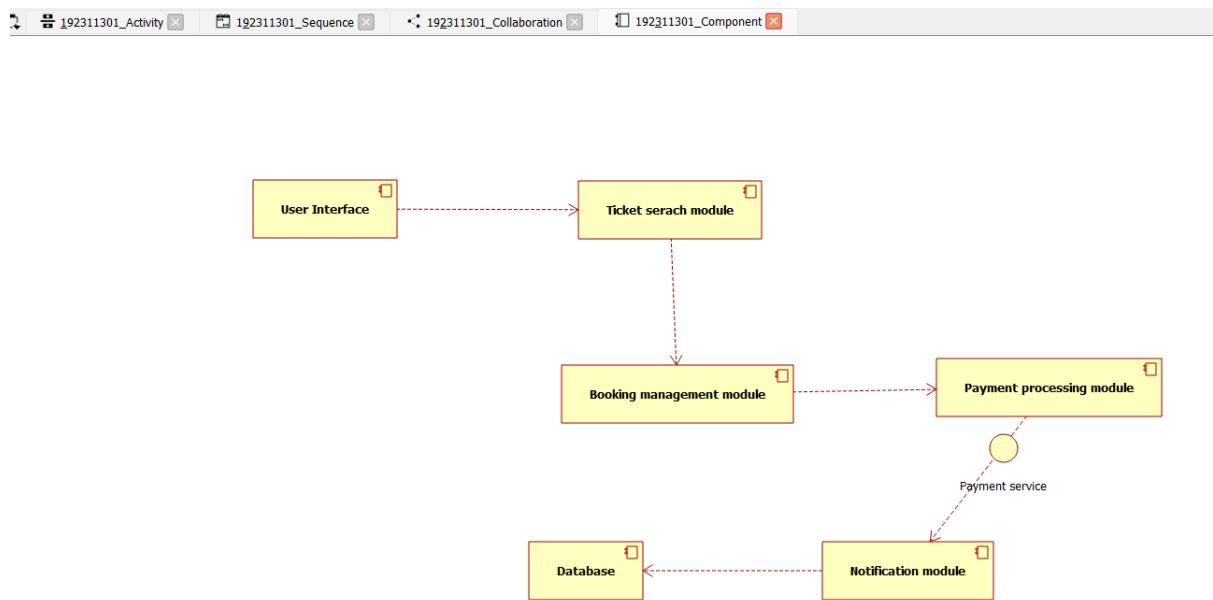
1: Search trip
u : User → s : Search Service
3: Schedule list
b : Booking controller
4: Select trip
9: Availability confirmed
11: Payment Success
6: Check seat availability
b : Booking form
5: Submit booking
s2 : Schedule
8: Seat available
10: Process payment
p : Payment Gateway
12: Generate E ticket
2: Get Schedule details
15: Ticket Confirmed
7: Fetch seat status
n : Notification Service
t : Ticket Generator
13: Store ticket
d : Database
14: Send Ticket
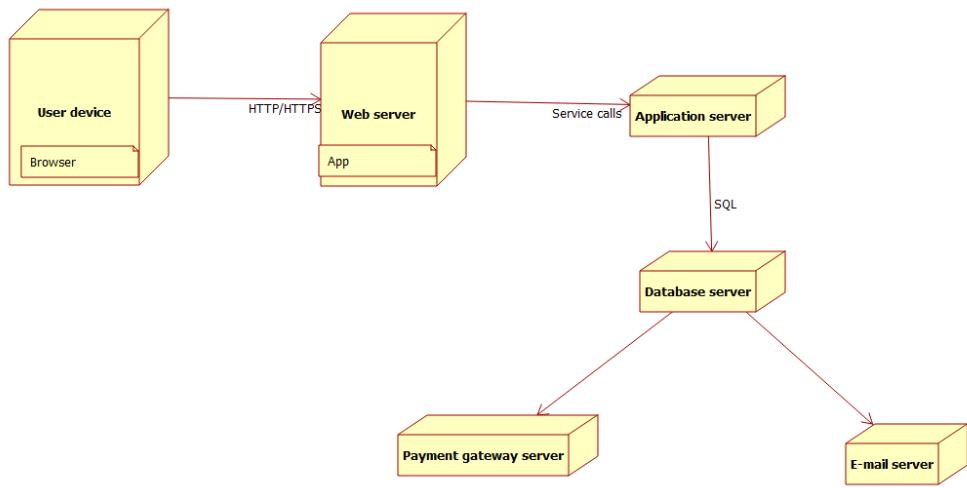
State chat diagram:
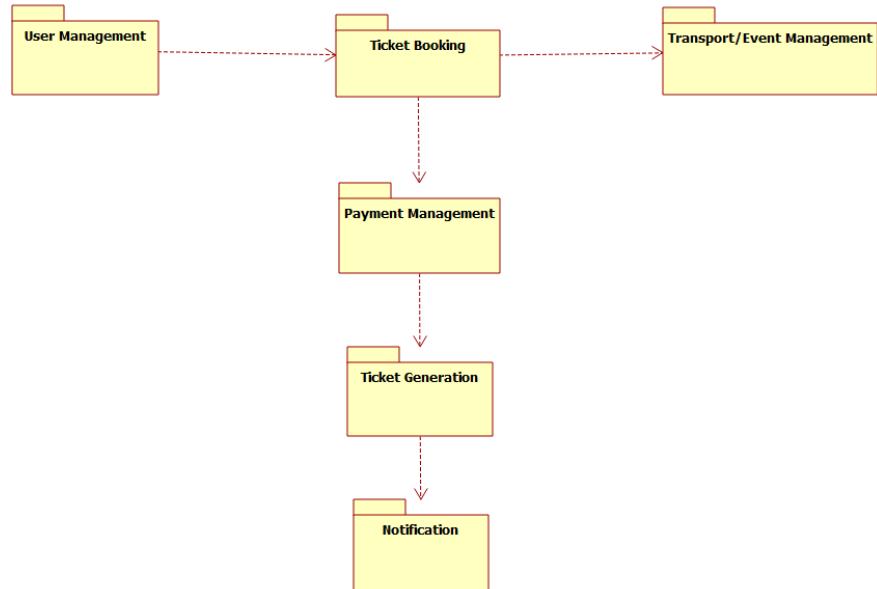
Component Diagram:

Deployement Diagram :

Package diagram:

```
User Management  ----→  Ticket Booking  ----→  Transport/Event Management
                              |
                              ↓
                      Payment Management
                              |
                              ↓
                      Ticket Generation
                              |
                              ↓
                         Notification
```

PROGRAM CODING:

PASSENGER:

Public class passenger

{

 Public integer passenger passenger name;

 Public integer passenger passenger age;

 Public integer train no;

 Public void passenger()

 {

 }

```
Public void new operation()

{

 }

}
```

CENTRAL MANAGEMENT SYSTEM:

```
Public class central management

{

 Public integer train name;

 Public integer passenger name;

 Public void reservation()

 {

 }

 Public void cancellation()

 {

 }

 Public void status()

 {

 }

 Public void login()

 {
```

```
 }
Private void management()

 {

 }

 }
```

RAILWAY RESERVATION SYSTEM:

```
Public class railway reservation system

{Public integer trainno;

 Public integer train name;

 Public integer passenger name;

 Public void status()

 {

 }

 Public void reservation()

 {

 }

 Public void cancellation()

 {

 }

 Public void railway reservation system()

 {
```

}

}

RESULT:

Thus the diagrams[use case, activity, sequence, collaboration, class, statechart, component,

deployment, package] for the E-ticketing system has been designed, executed and output is

verified.