**FLIGHT MANAGEMENT SYSTEM**

**PREPARED FOR**

JEE Cloud Training Program

Capgemini Technology Services

**PREPARED BY**

Rtvik Gupta

# Abstract:

The Flight Management System is a Java-based booking solution for flight tickets. It consolidates data provided by different airline carriers and hence provides the user details and rates in real-time. Travellers may want to make changes in their bookings. The application allows them to book, cancel, view and update their bookings with ease. Other than this, it eases the management of bookings too. All the bookings, flights, schedules and routes can be viewed, added and modified on a single application by the administrator.

# **Scopes:**

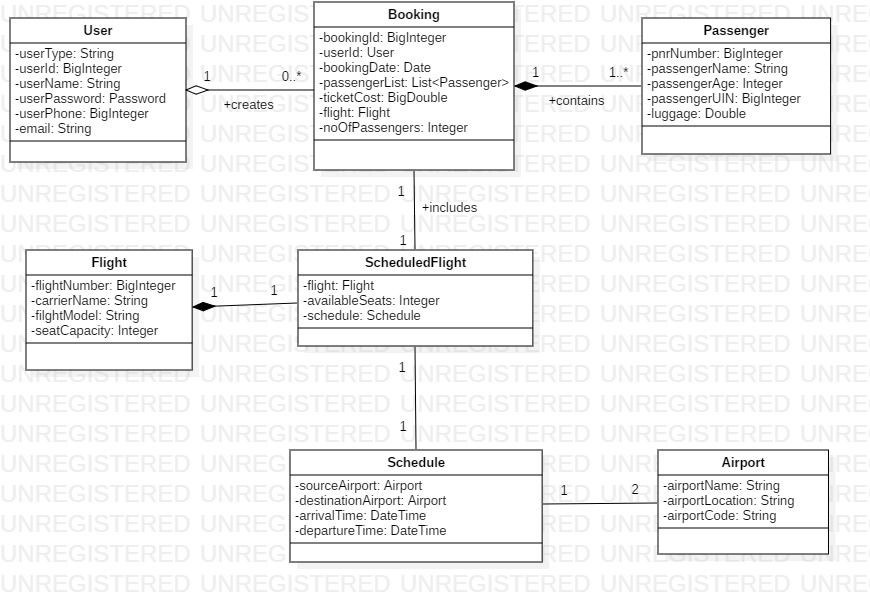
Inscope:

Following is the functionality provided by the system:

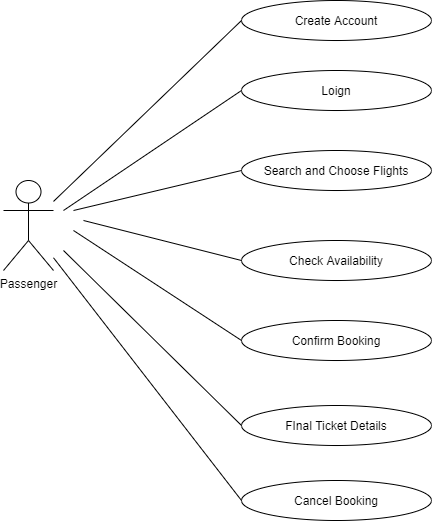
There are two categories of people who would access the system: customer and administrator. Each of these would have some exclusive privileges.

1. The customer can:
   1. Create his user account.
   2. Login into the application.
   3. Check for available flights.
   4. Make a booking.
   5. View the bookings made.
   6. Cancel or modify a booking.
2. The administrator can:
   1. Login into the application.
   2. Add flight, schedule and route details.
   3. View the flight, schedule and route details.
   4. Cancel or modify the flight, schedule and route details.

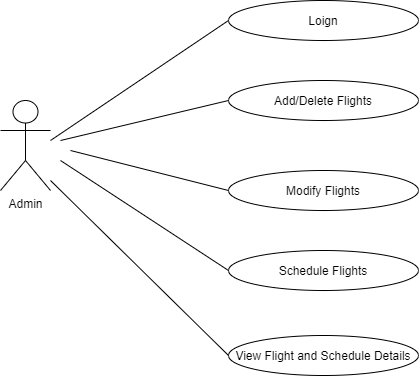
# **Class Diagram:**



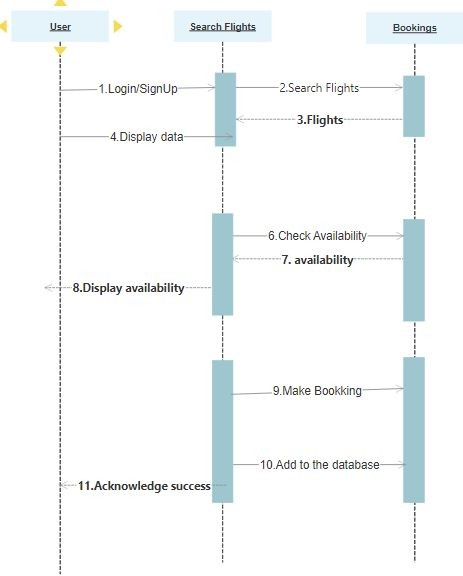
# **Use Case Diagram for Passenger:**



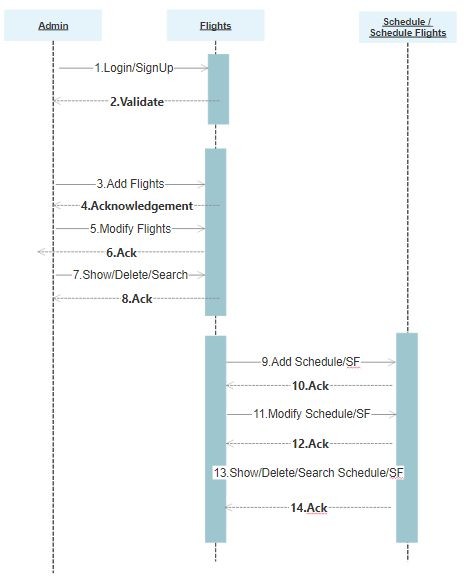
# **Use Case Diagram for Admin:**



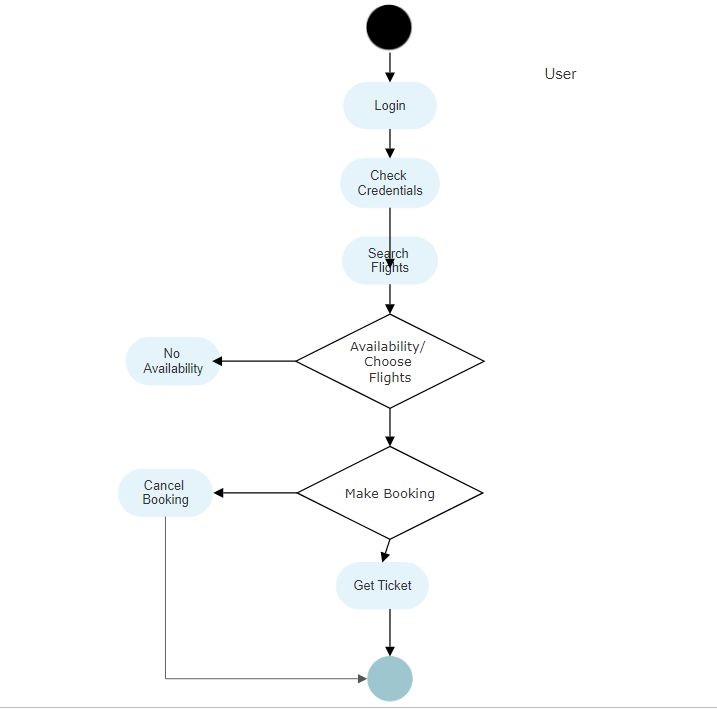
# **Sequence Diagram for Passenger:**



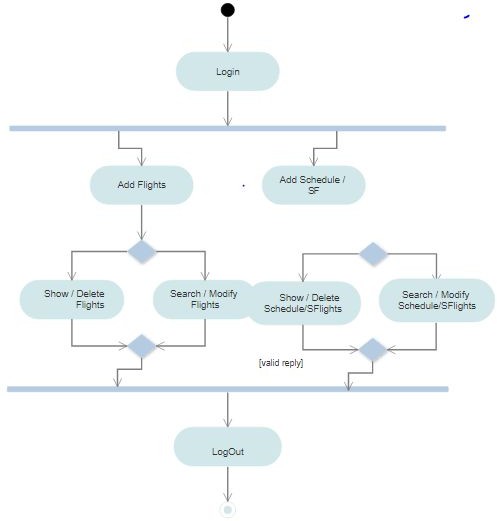
# **Sequence Diagram for Admin:**



# **Activity Diagram for Passenger:**



# **Activity Diagram for Admin:**



# **Class and Method Description:**

## **DTO Layer:**

1. **User:** This class stores the user type (admin or the customer) and all user information.

Attributes:

userType: String. userId: BigInteger userName: String

userPassword: String userPhone:BigInteger userEmail: String

1. **Passenger**: This class stores all the details of the travelling passenger.

Attributes:

pnrNumber:BigInteger passengerName:String passengerAge: Integer passengerUIN:BigInteger Luggage: Double

## **Service Layer:**

1. **UserServiceImpl:**

Methods:

addUser(User):User :-

Adds a new user

viewUser(BigInteger):User :-

Shows the details of a user identifiable by the user id.

viewUser():List<User :- Shows the details of all users.

updateUser(User):User :- Updates the details of a user.

deleteUser(BigInteger):void Removes a user as per the user id.

validateUser(User): void :- Validates the attributes of a user.