ENSF 480 - Principles of Software Design

Term Project (Design Phase)

Group 11 (B01):

Harris Hasnain - 30139872

Hooriya Amjad - 30141172

Al Farhana Siddique - 30157968

Part 1:

1) Systems Description:

System Description: Web-Based Flight Reservation and Management System

Overview: The project aims to develop a web-based system for a single airline company, catering to a diverse range of users, including registered users, airline agents, system administrators, etc. This system provides a user-friendly interface for browsing available flights, booking flights, managing flight data, and ensuring a seamless and efficient travel experience. The system offers a variety of features and functionalities tailored to meet the specific needs of each user category.

Key Functionalities:

For All Users:

- All users have the ability to:
    - Browsing Available Flights
    - Selecting Desired Flights
    - Browsing Seat Maps Graphically
    - Selecting Desired Seats
    - Selecting Ticket Cancellation Insurance
    - Making Payments
    - Receiving Tickets and Payment via Email
    - Flight Cancellation

For Company Employees (Airline Agents and Flight Attendants):

- Employees can access and view the list of passengers for a specific flight including information such as
    - Seat numbers
    - Contact details

For System Admins:

- System administrators have the ability to manage flight-related information in the database, including:
    - Browsing lists of flights.
    - Browsing lists of crew members for each flight.
    - Browsing lists of aircrafts.
    - Adding or removing crew members.
    - Adding or removing aircrafts.
    - Adding, removing, or modifying flight information.
    - Having access to lists of registered users for record-keeping and communication purposes.

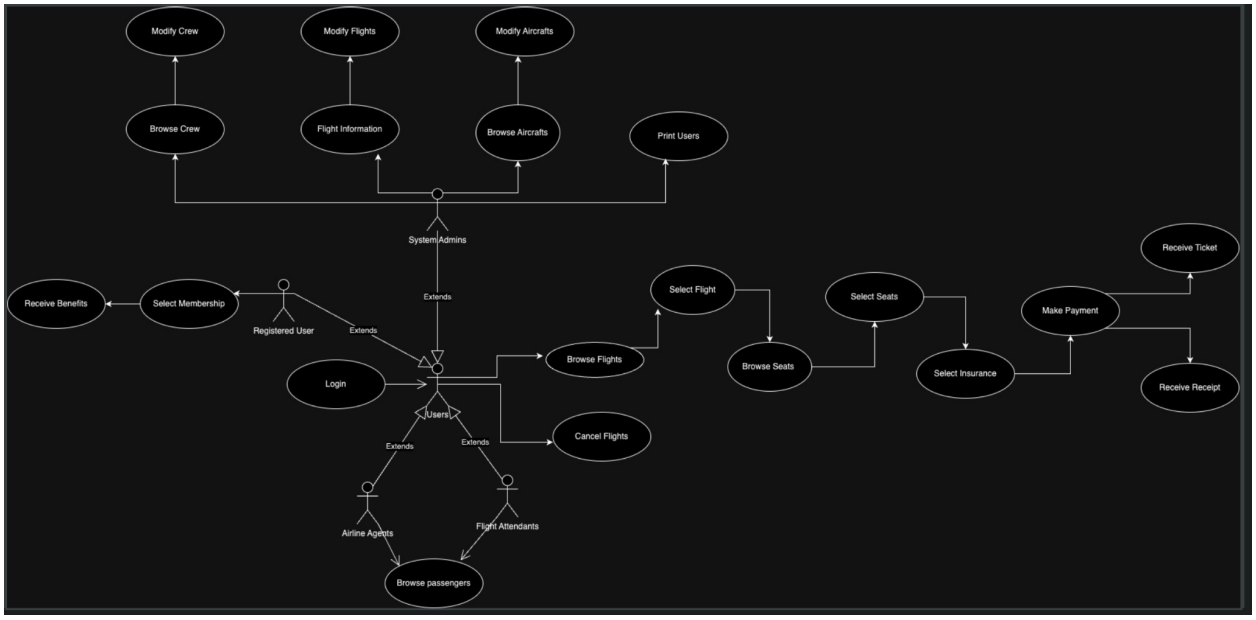
Additional Information:

1. **Seat Options and Pricing:**

- The system allows users to choose from different seat types, including Ordinary, Comfort, and Business-Class. Pricing varies accordingly, with Comfort seats costing approximately 40% more than Ordinary seats, and Business-Class seats being priced at more than double the cost of Ordinary seats.

2. **Registered Users:**
   - Registered users can enjoy additional benefits, including:
     - Receiving monthly promotion news.
     - Accessing airport lounges at discounted rates.
     - Receiving a complimentary companion ticket once a year.
     - Get a company credit card.
   - Registered user information, such as name, address, and more, is securely stored in the company's database.

2) Use Case Diagram:



3) System Scenarios:

- **Actors extend User:**
  - Airline Agent
  - Flight Attendant
  - System Admin
  - Registered Users

- **Use Cases for all Users:**
  - Browse Available Flights
    - The user accesses the system and browses a list of available flights.
    - **System Response:** The system displays a list of available flights, displaying flight details like departure times, arrival times, origin, destination, etc.
    - Users can select desired flights from the available flights based on their preferences and requirements.
  - Select Flight
    - After browsing the available flights, the user or airline agent selects a specific flight of interest.
    - **System Response:** The system displays flight information, including the seat map, class options, and other relevant details.
    - Users can review this information to make an informed decision about the flight selection.
  - Browse Seat Map
    - When a flight is selected, the user views the seat map for that flight.
    - **System Response:** The system presents a graphical representation of the seating arrangement for the chosen flight, indicating available and occupied seats.
    - Users can visually inspect the seat map to decide on their preferred seat location.
  - Select Seat
    - While viewing the seat map, the user selects a seat for the flight.
    - **System Response:** The system records the selected seat, ensuring it's reserved for the user.
    - Users can choose between ordinary, comfort, or business-class seats as per their preferences.
  - Select Ticket Cancellation Insurance
    - During the booking process, users are presented with the option to add ticket cancellation insurance.
    - **System Response:** The system presents a dropdown menu for the user's choice.
    - Users can select or decline the insurance and provide the required details if they choose to purchase it.
  - Make Payment

- After confirming the flight and any additional services (like insurance), they proceed to make the payment.
- **System Response:** The system presents a payment form where the user enters credit card details.
- Users complete the payment transaction to finalize the booking.

◆ Receive Ticket and Payment Receipt via Email
- Following a successful payment, they expect to receive their flight ticket and payment information.
- **System Response:** The system sends an email to the address logged into the system, containing the necessary details.
- Users can check their email for the ticket information and verify the booking..

◆ Cancel Flight
- Users can access their account to view their booked flights and initiate a cancellation process.
- **System Response:** The system presents a list of booked flights and allows the user to select the flight they wish to cancel.
- User cancels and the system sends a cancellation confirmation email to the user.

➔ **Use Cases for Employees:**

◆ Browse List of Passengers in a Flight
- Employees can view the list of passengers on a selected flight when browsing flights
- **System Response:** The system presents a list of available flights which the employee chooses from, and then displays a list of passengers on the flight and displays their name, seat numbers, etc.
- Employees explore the passengers based on their preferences and requirements.

➔ **Use Cases for System Admins:**

◆ Browse List of Flights
- Admins access the system and browse available flights.
- **System Response:** The system retrieves a list of available flights, displaying flight details like departure times, arrival times, and ticket prices.
- Admins can explore the available flights based on their preferences and requirements.

◆ Browse List of Crews
- Admins can browse a list of crew members available by the company for each available flight.
- **System Response:** The system retrieves a list of crew members, displaying crew details like crew names, position, etc.
- Admins can explore the available crews based on their preferences and requirements.

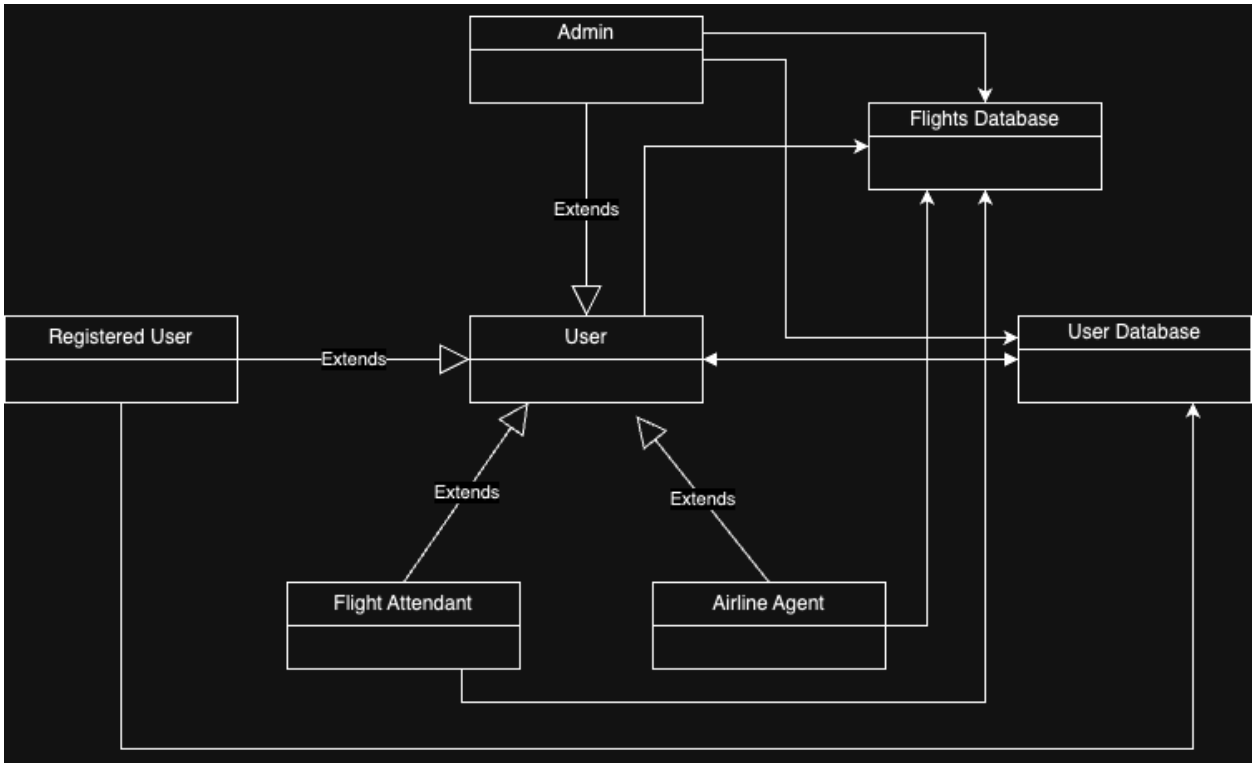
◆ Browse List of Aircrafts
- Admins can browse a list of aircrafts available by the company.
- **System Response:** The system retrieves a list of available aircrafts based on the entered information, displaying aircrafts details like the aircraft model.

        - Admins can explore the available aircrafts based on their preferences and requirements.
    - Add/Remove Crew Member
        - Admins can browse a list of crew members available by the company and add or remove a crew member.
        - **System Response:** The system presents a list of crew members the company is offering for the specific flight and displays their names, positions, etc.
        - Admins can remove or add a crew in the list of crew members.
    - Add/Remove Aircraft
        - Admins can browse a list of aircrafts available by the company and add or remove an aircraft.
        - **System Response:** The system presents a list of aircrafts the company offers and displays their aircraft modal.
        - Admins can remove or add an aircraft in the list of aircrafts.
    - Add/Remove Flight Destinations
        - **System Admins:** Admins can browse the system to manage the flight destinations that the airline services.
        - **System Response:** The system displays a list of current flight destinations and options to add new destinations or remove existing ones.
        - **Admin Actions:** Input new destinations, confirm removals, and update the list of flight destinations offered by the airline.
    - Add/Remove/Modify Flight Information
        - **System Admins:** Admins can browse flights and add, remove, or modify the details of the selected flight.
        - **System Response:** The system displays existing flight information and allows for modifications, deletions, or additions of flight information.
        - **Admin Actions:** Alter flight times, dates, seat availability, pricing, or any related flight details and save these changes to update the flight information.
    - Print List of Registered Users
        - **System Admins:** Admins can request a list of all registered users for reporting or audit purposes.
        - **System Response:** Compiles and presents a comprehensive list of registered users with relevant details such as names, registration dates, membership information, etc.
        - **Admin Actions:** Admins can save this list as a text file.

➔ **Special Use Cases for Registered Users:**
    - Register for Membership
        - **Registered User:** User decides to take advantage of membership benefits and begins the registration process.
        - **System Response:** Provides a membership registration form requesting additional user details, membership level selection, and payment information if applicable, finally a membership information email will be sent to the user.
        - **Registered User Actions:** Fill out the form with the necessary details, select a membership tier, and submit their application for processing.
    - Apply for Company's Credit Card
        - **Registered User:** User Opts to apply for the company's branded credit card to avail of additional travel benefits and rewards.

- **System Response:** The system presents the user with an application form for the credit card, detailing the benefits, terms, and conditions, finally an application information email will be sent to the user.
- **Registered User Actions:** Complete the application form with financial details required for the credit card and submit the application.

4) Conceptual Model:



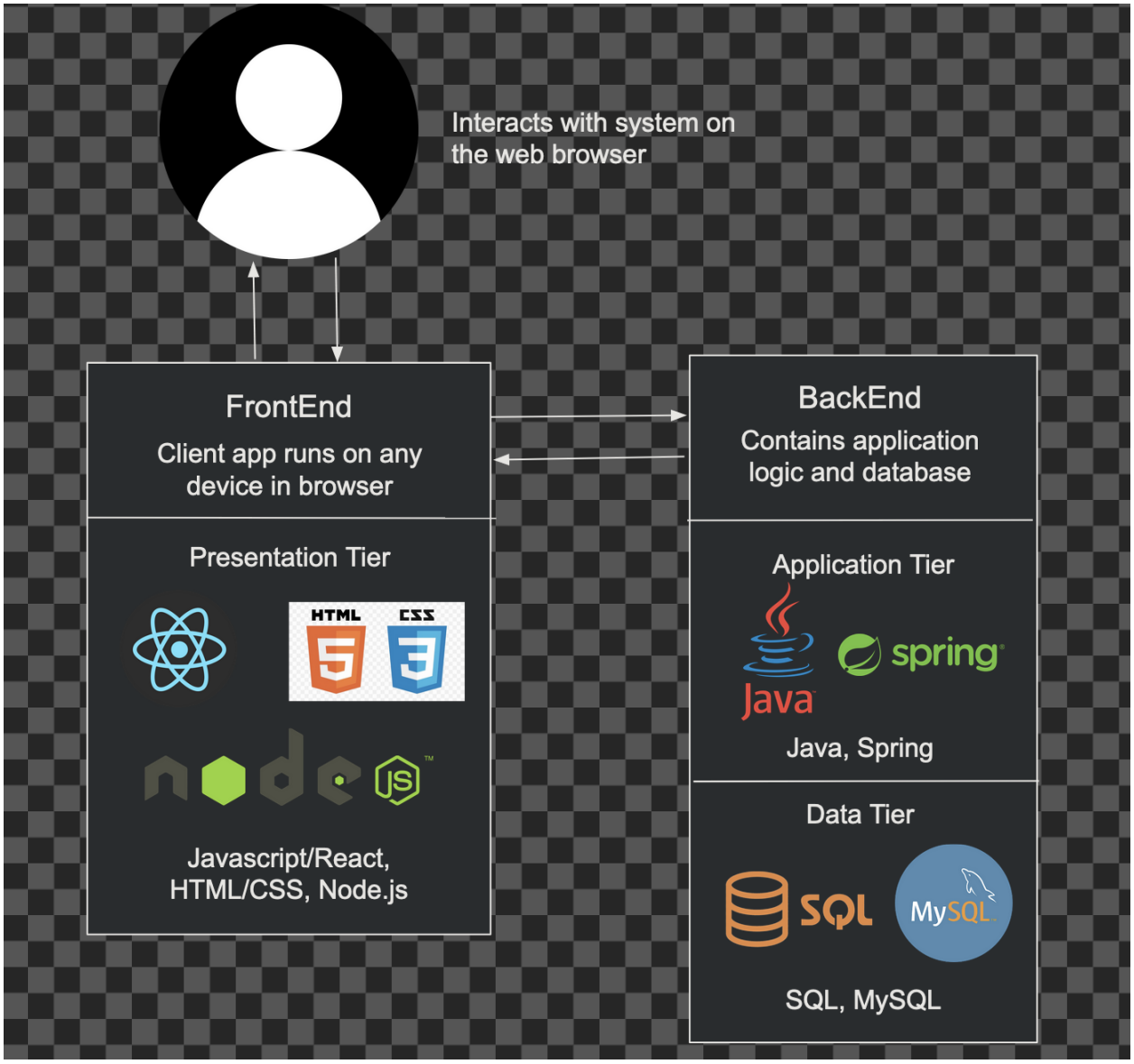
Part 2:

1) System Architecture:

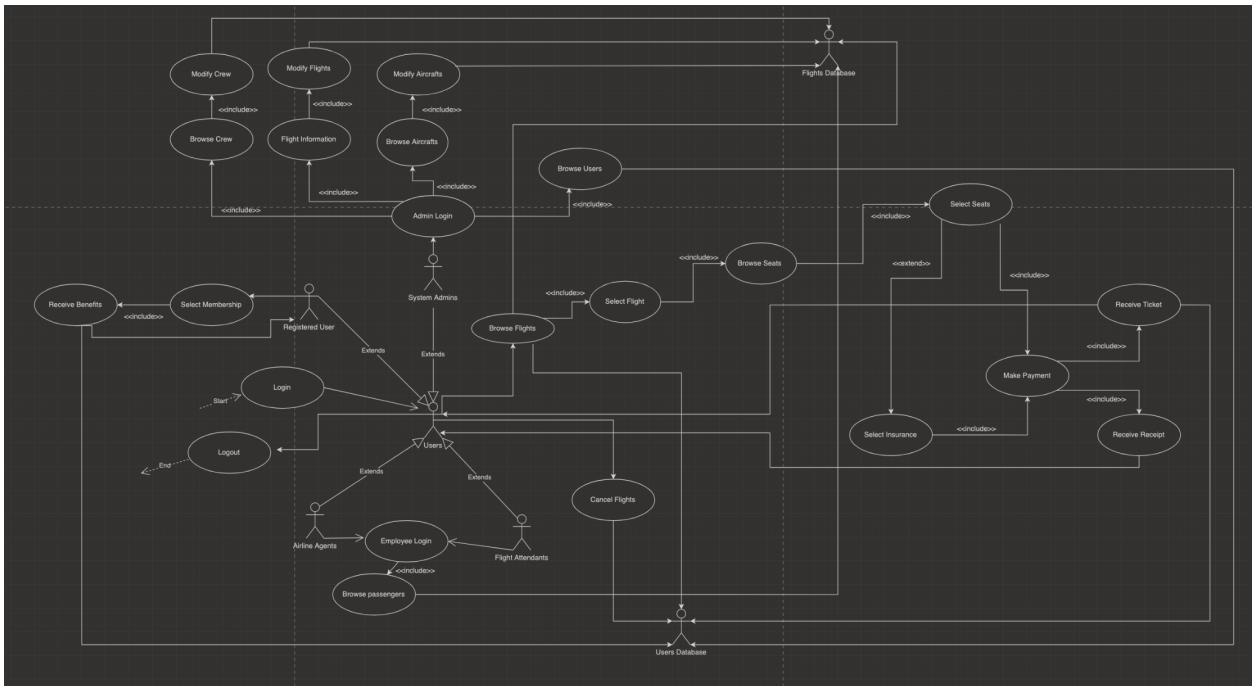
System Architecture Overview:

The Flight Reservation Web Application will follow a client-server architecture, with a web-based front end developed using JavaScript/React, HTML/CSS and powered by Node.js, backend developed using Java and the Spring framework, and a SQL database for data storage. The system will cater to different types of users, including regular users or registered users, flight attendants and airline agents (employees), and system admins.

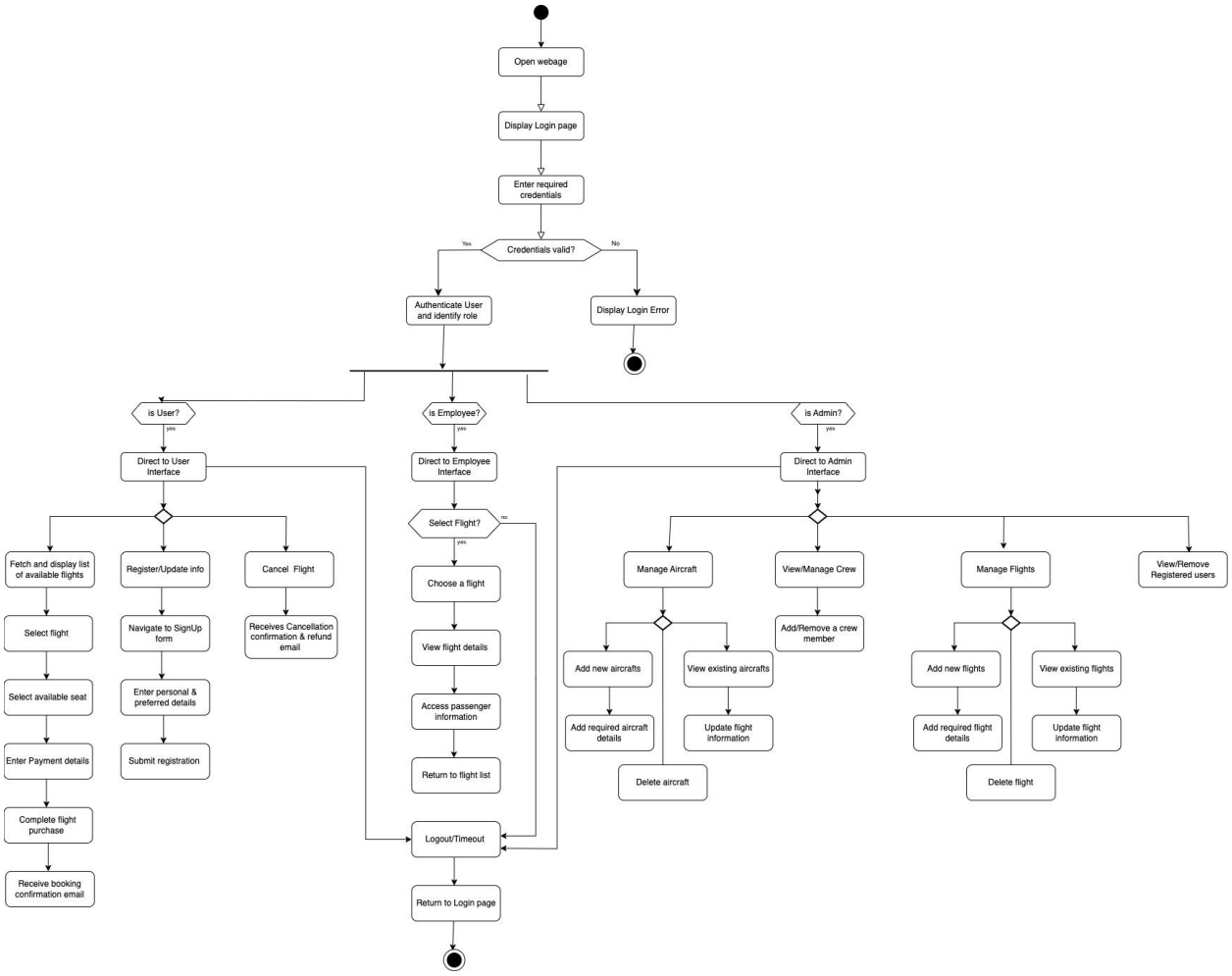
1. **Client-Side (Front End):**
    - Developed using JavaScript/React, HTML/CSS, and Node.js
    - Provides a user-friendly interface for various actions like browsing flights, selecting seats, making payments, etc.
    - Different views for different user roles (Users, Employees, and Admins).
2. **Server-Side (Back End):**
    - Developed using Java and the Spring framework to handle server-side logic.
    - Manages communication with the database and processes user requests.
    - Implements business logic such as flight management, aircraft management, seat booking logic, etc.
3. **Database:**
    - Implemented database using SQL and MySQL.
    - Stores information related to flights, seats, crews, and other relevant data.
    - Designed to support efficient retrieval and storage of information for quick application responsiveness.


Interacts with system on the web browser
FrontEnd
Client app runs on any device in browser
Presentation Tier
HTML
CSS
Javascript/React, HTML/CSS, Node.js
BackEnd
Contains application logic and database
Application Tier
Java
spring
Java, Spring
Data Tier
SQL
MySQL
SQL, MySQL

2) Updated Use Case Diagram:

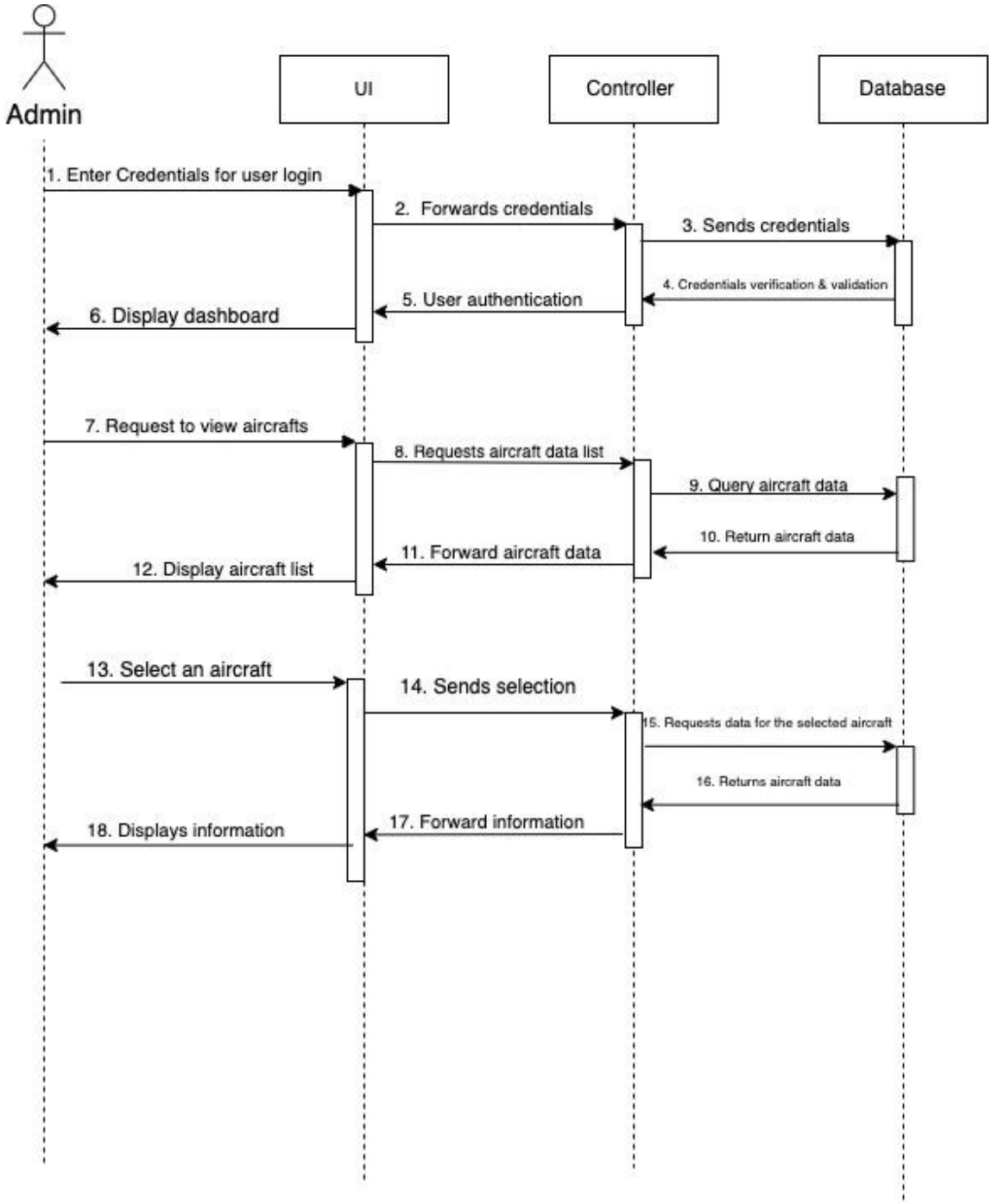


3) Activity Diagram:

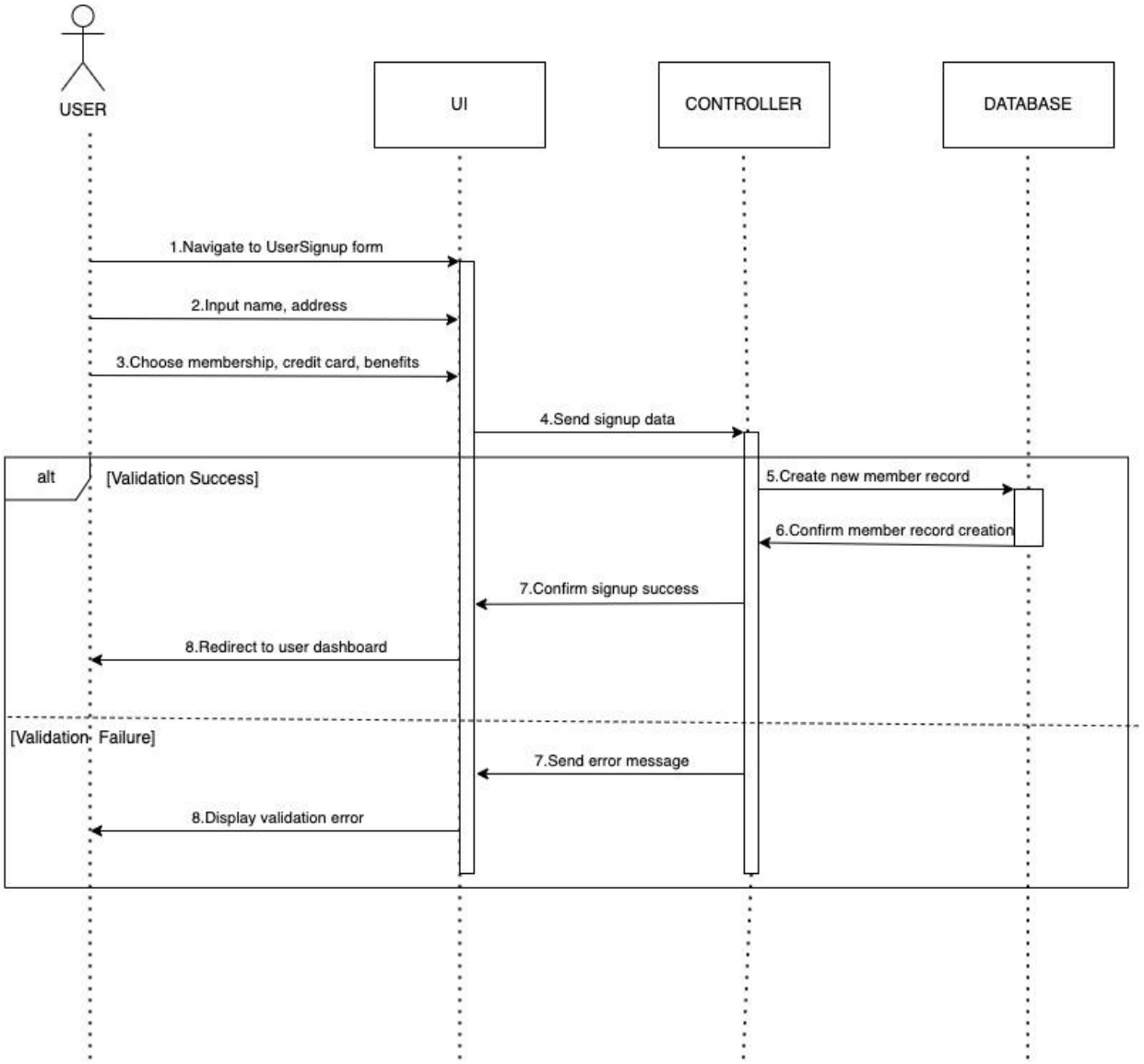


4) Sequence Diagrams:

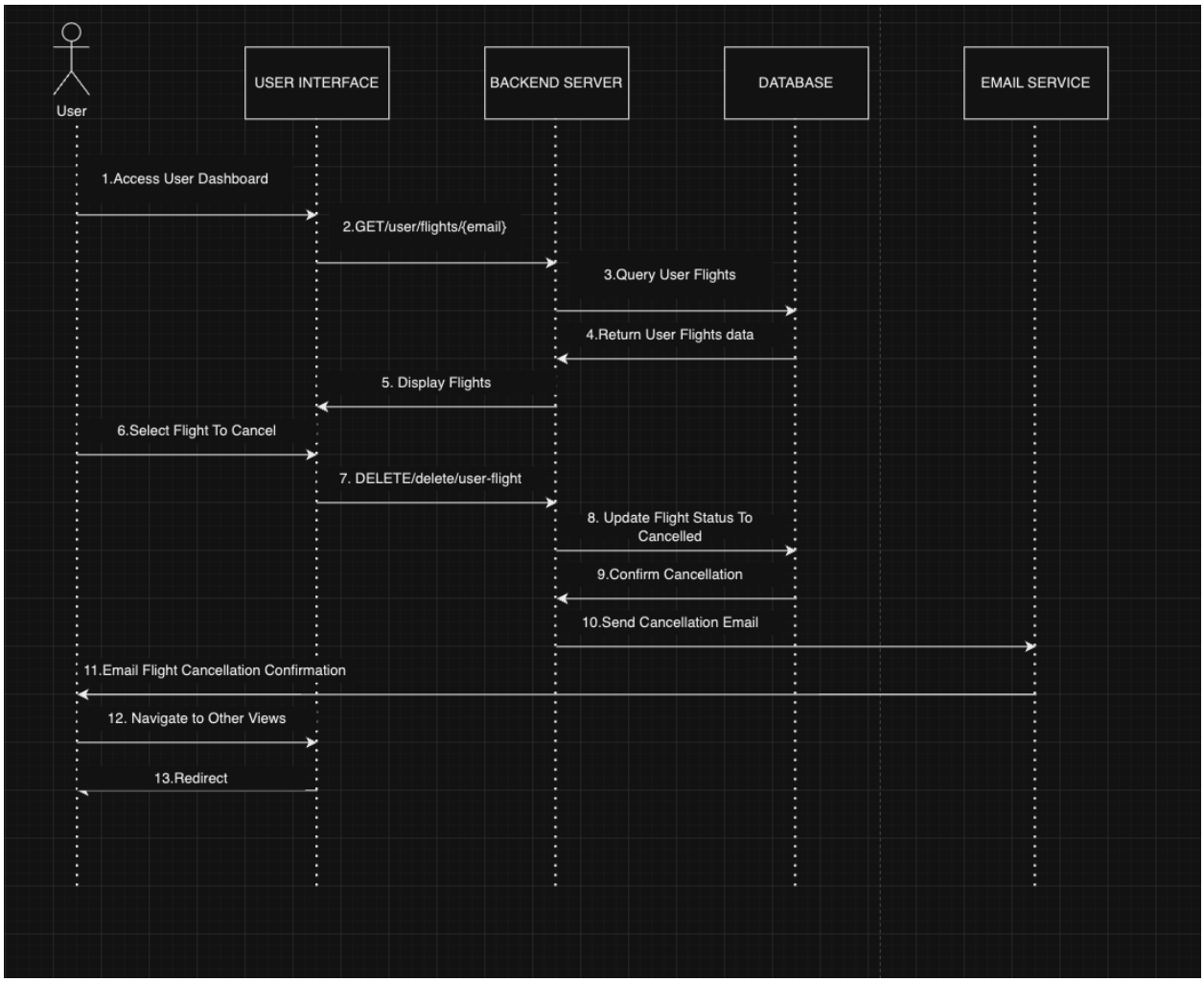
SD1: Browsing aircrafts (Admin)



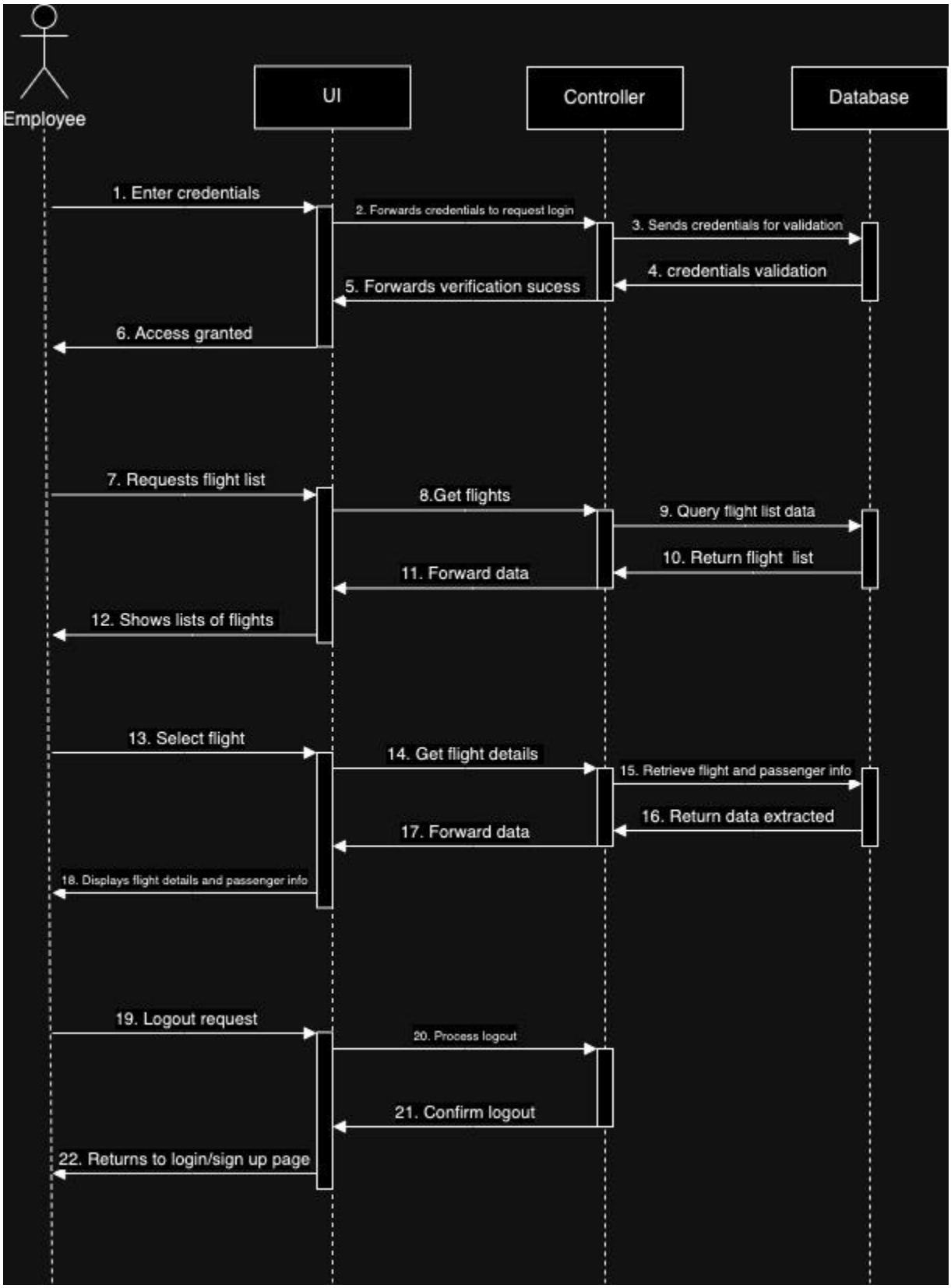
SD2: User SignUp



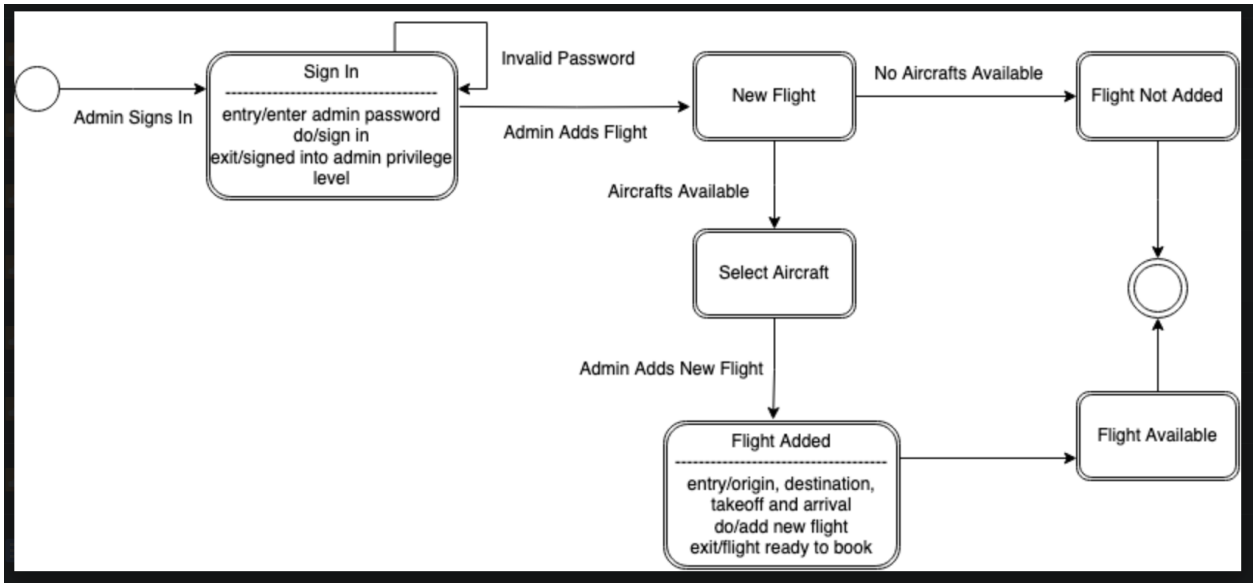
SD3: Cancellation of flights (User)

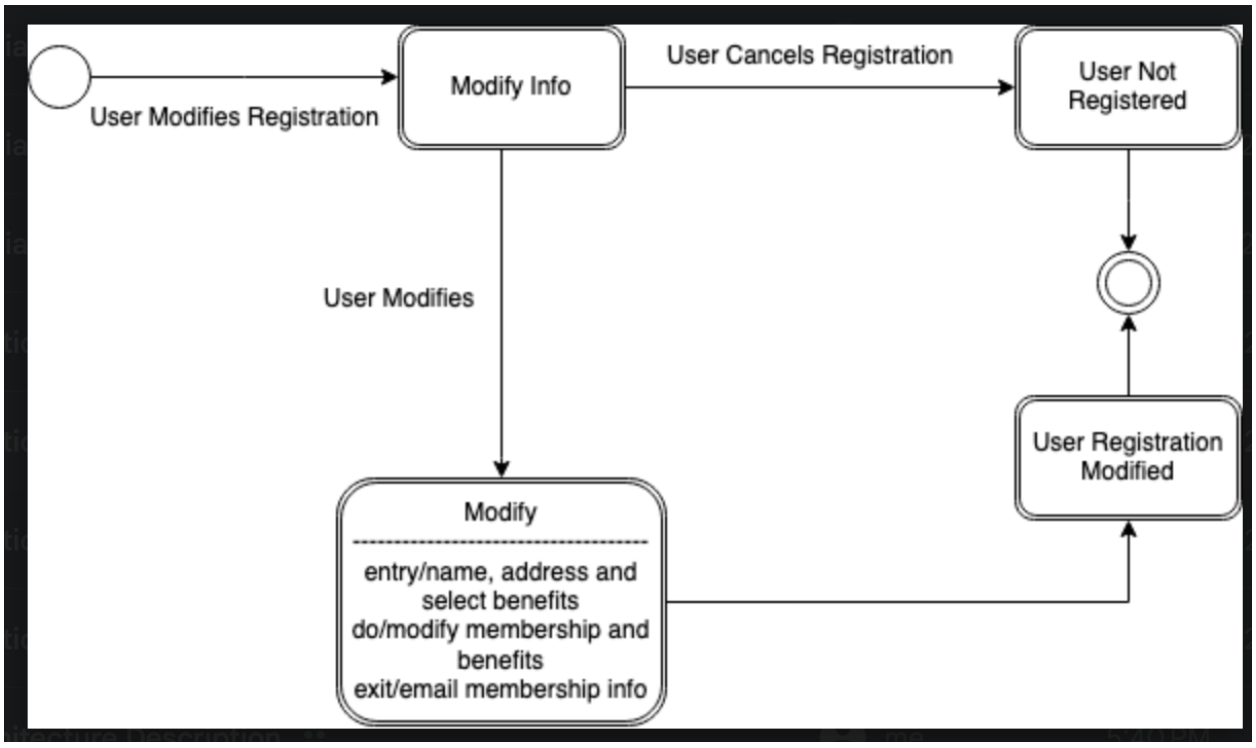


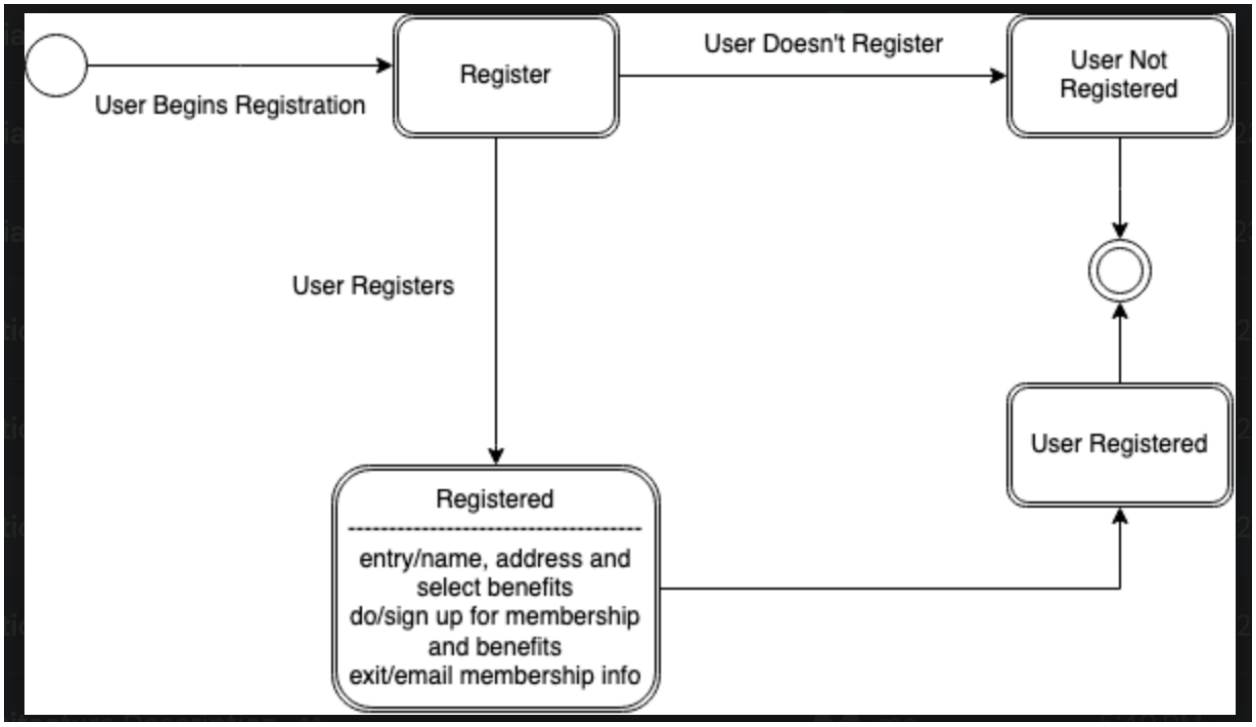
SD4:

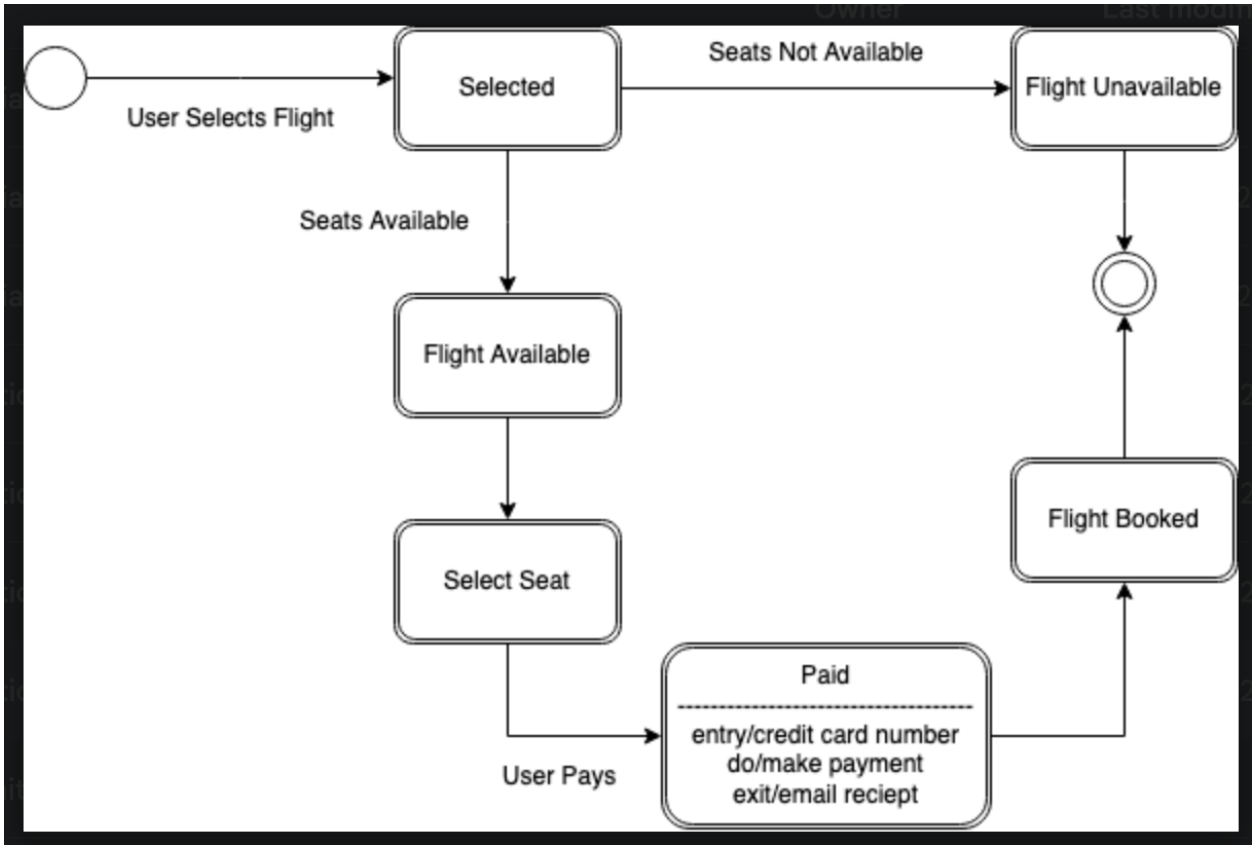

Employee
UI
Controller
Database
1. Enter credentials
2. Forwards credentials to request login
3. Sends credentials for validation
4. credentials validation
5. Forwards verification sucess
6. Access granted
7. Requests flight list
8.Get flights
9. Query flight list data
10. Return flight list
11. Forward data
12. Shows lists of flights
13. Select flight
14. Get flight details
15. Retrieve flight and passenger info
16. Return data extracted
17. Forward data
18. Displays flight details and passenger info
19. Logout request
20. Process logout
21. Confirm logout
22. Returns to login/sign up page

5) State Transition Diagrams:

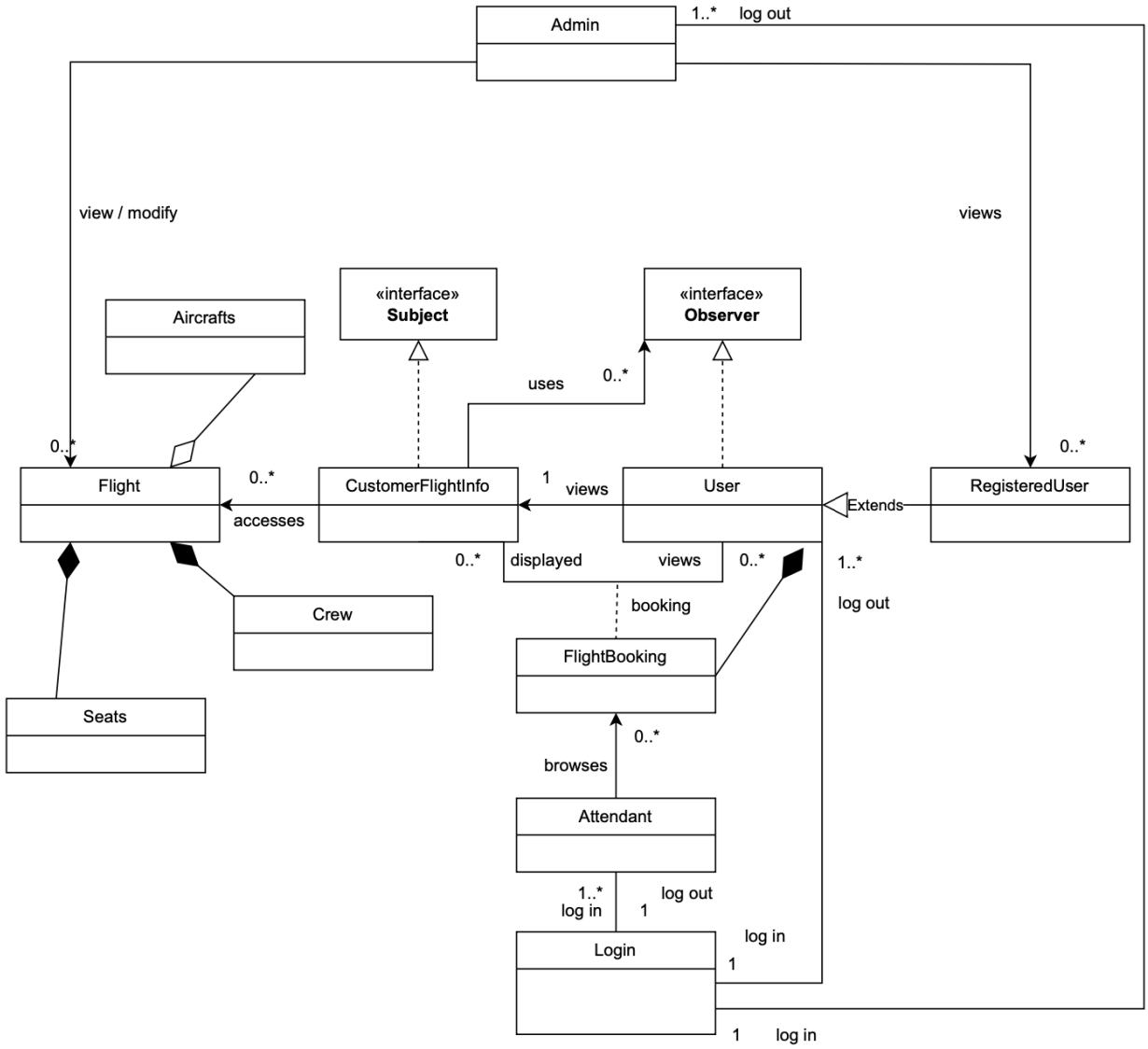





User Begins Registration
Register
User Doesn't Register
User Not Registered
User Registers
User Registered
Registered
entry/name, address and select benefits
do/sign up for membership and benefits
exit/email membership info

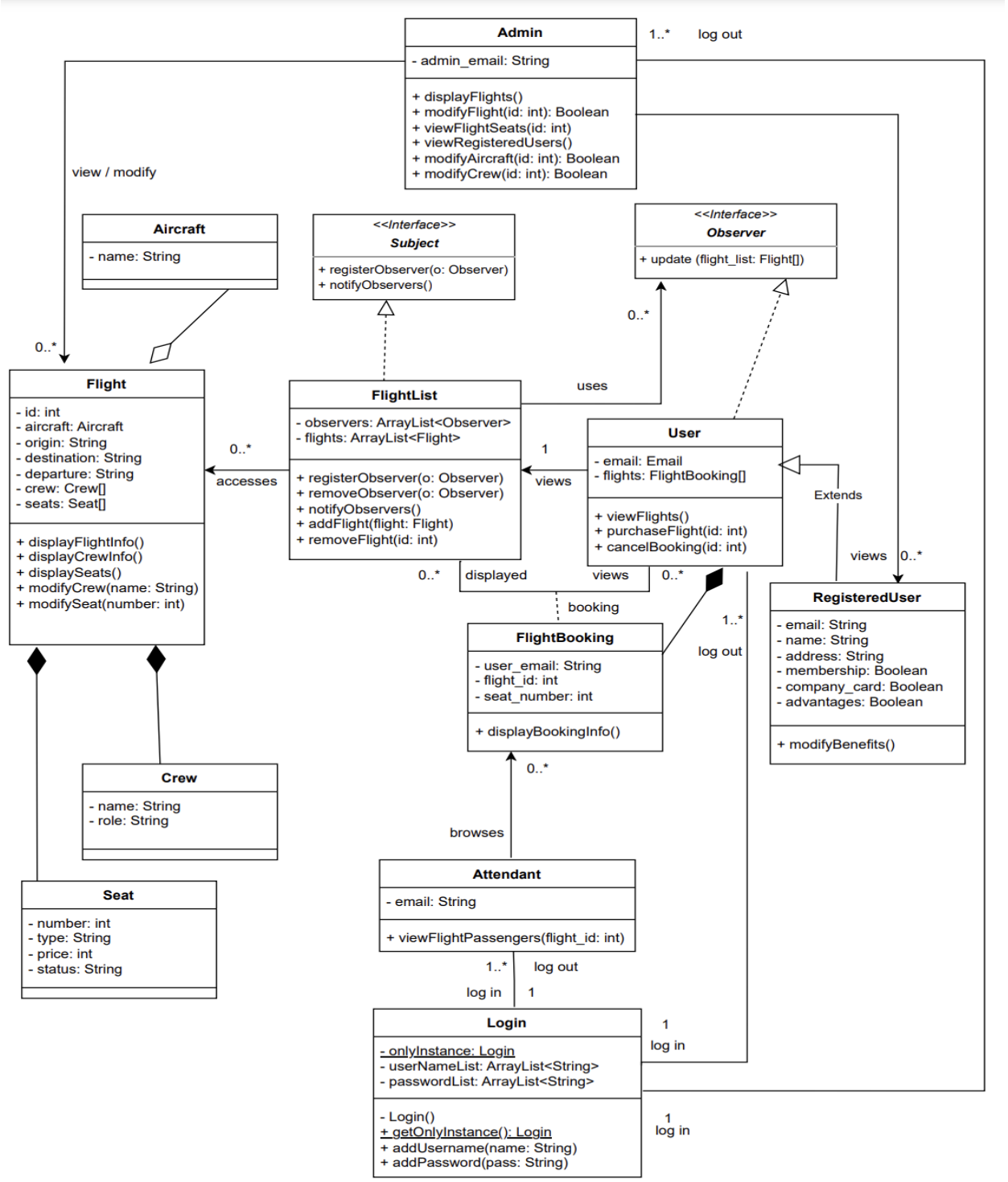

User Selects Flight
Selected
Seats Not Available
Flight Unavailable
Seats Available
Flight Available
Select Seat
Flight Booked
User Pays
Paid
entry/credit card number
do/make payment
exit/email reciept

6) Domain Class Diagram (no attributes / functionalities):

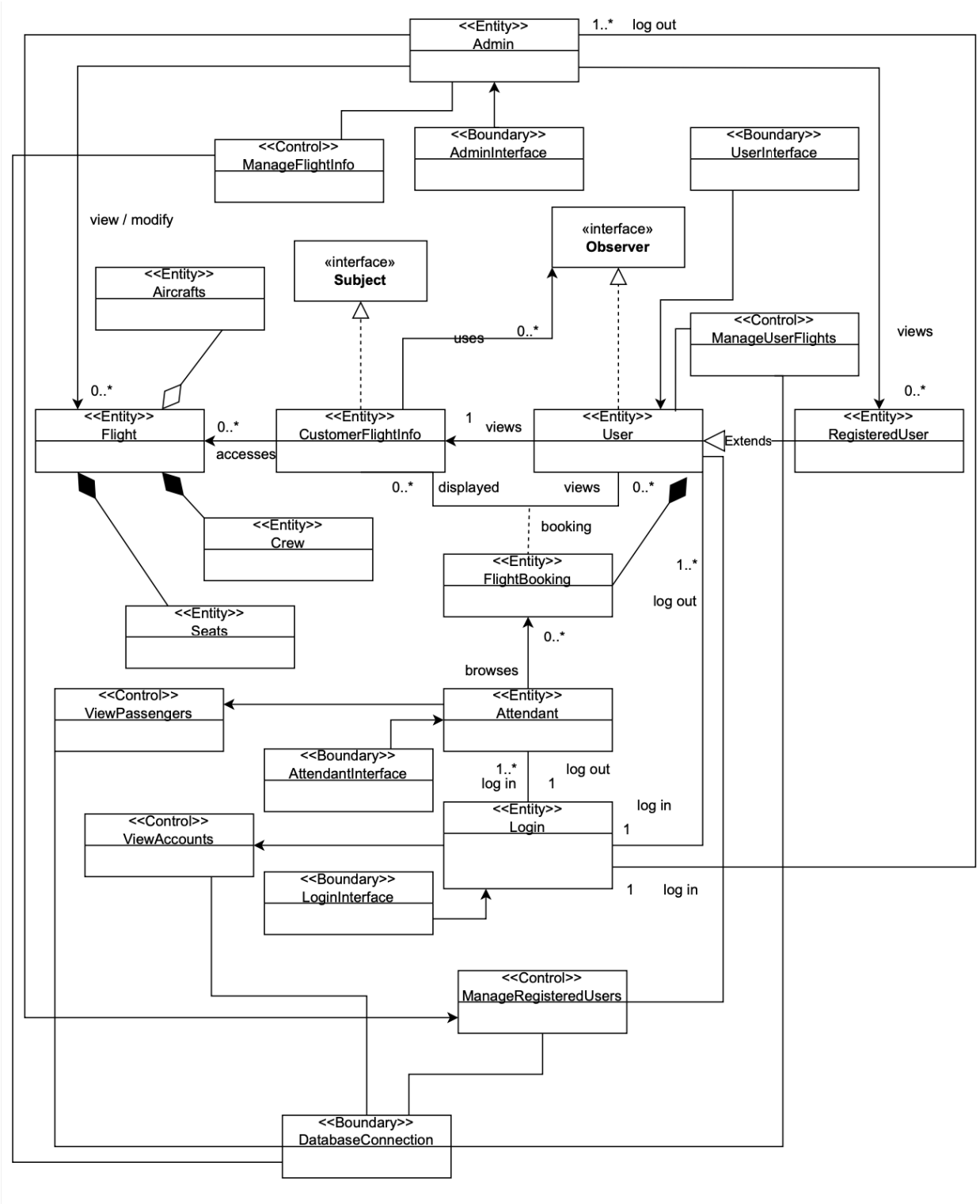


7) Domain Class Diagram (with attributes / functionalities):

- Design patterns used:
  - Singleton pattern (login information)
  - Observer pattern (user as observer, flight list as subject)

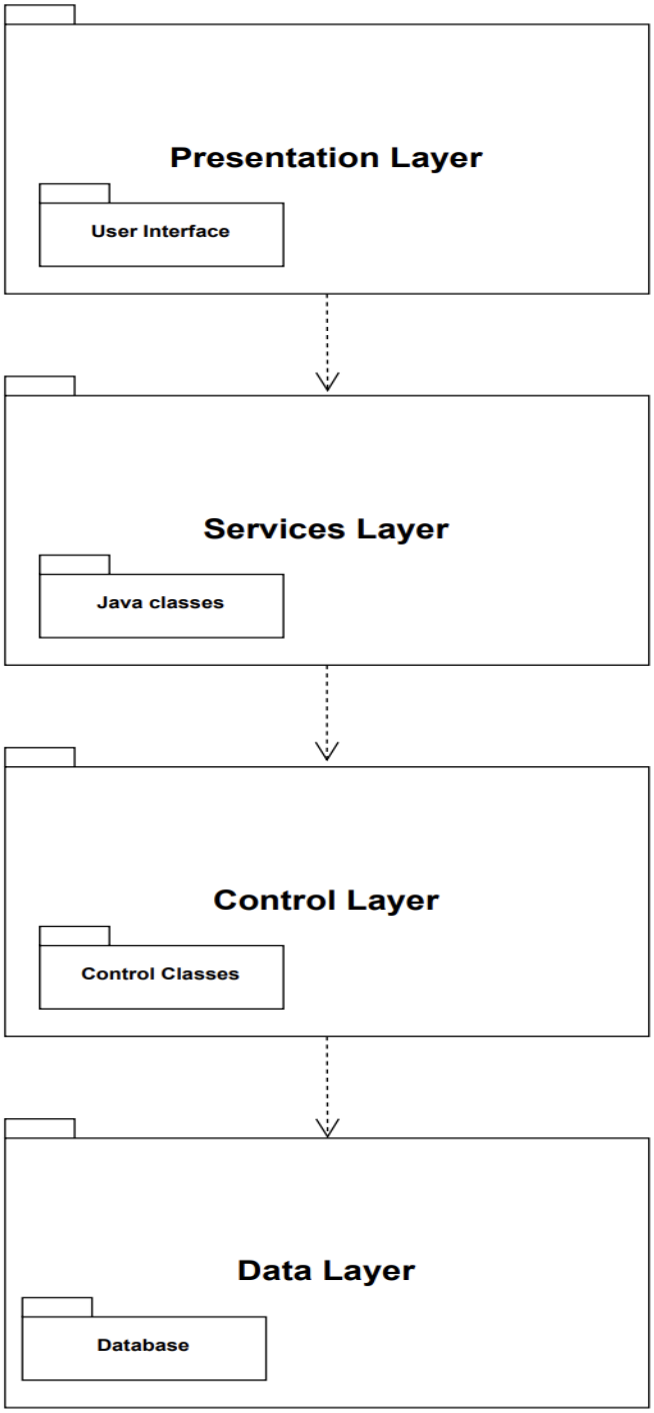


Part 3 (boundary, control and entity classes):



Part 4:

1) Package Diagram:



2) Deployment Diagram: