



UE21CS352B - Object Oriented Analysis & Design using Java

Mini Project Report

Travel Booking System

Submitted by:

Adithya TG	PES1UG21CS035
Akshar S	PES1UG21CS060
Adithya Mahesh	PES1UG21CS032
Sai Amara Prasad	PES1UG21CS001

6th Semester A Section

Prof. Course Instructor Name
Designation

January - May 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

PROBLEM STATEMENT

The project aims to develop a comprehensive flight booking application to help ease travel management system. Users can browse for available flights, make bookings and view their booking history. The project supports two types of users: regular users and administrators. Regular users can browse flights, make bookings, and cancel their bookings. Administrators have additional privileges such as managing users, flights and bookings.

Key Features of this project are:

User Management: Users can register, log in and update their profiles. Administrators can manage user accounts.

Flight Management: Flights are stored with details like source, destination, departure date, return date, price and available seats.

Booking Management: Users can book flights, view their booking history, and cancel bookings. Administrators can view all bookings.

Discounts: Discounts are applied to booking prices based on predefined rules.

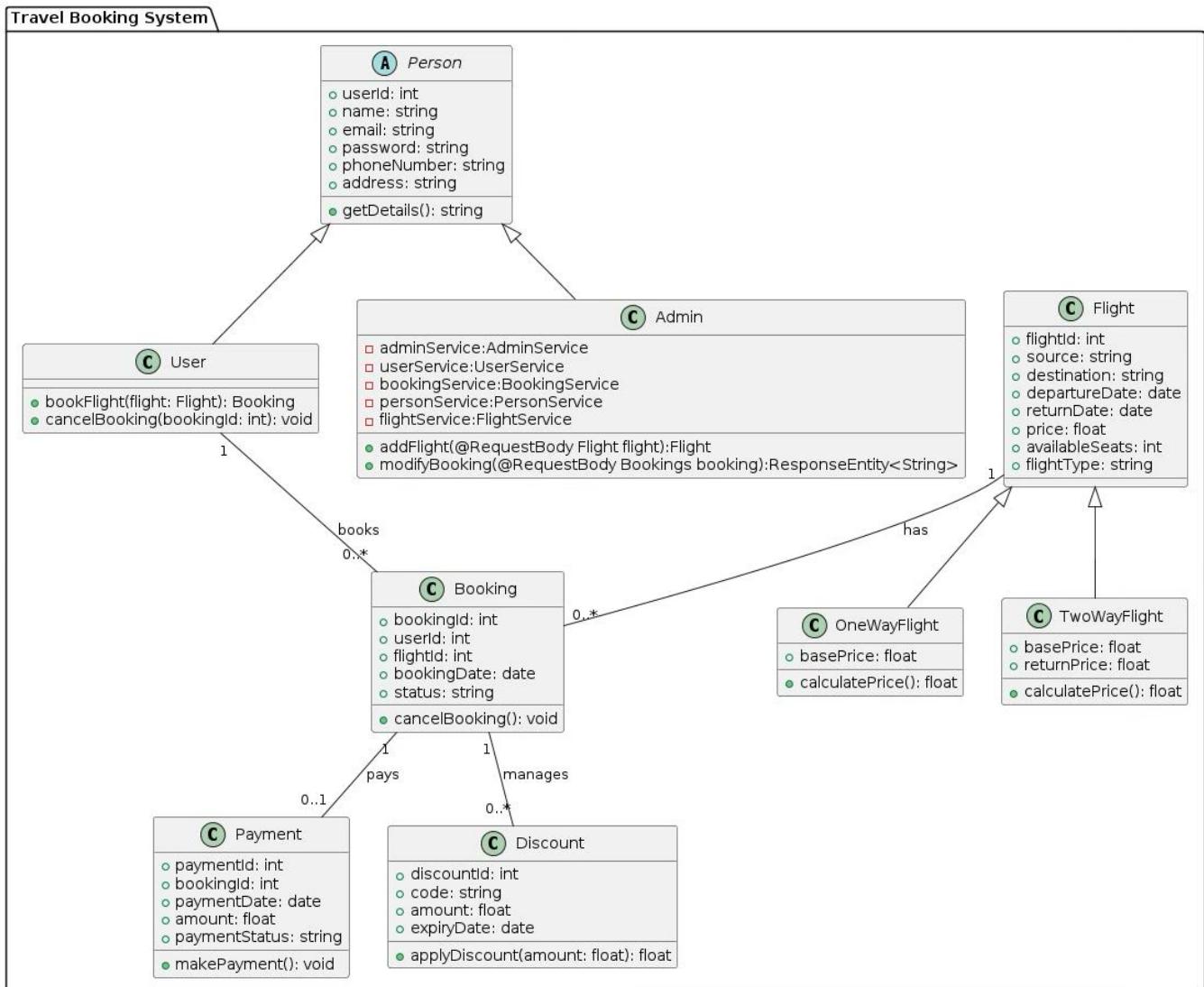
Database Integration: Data is stored using JPA entities and managed through repositories.

Spring Boot Application: The system is built using Spring Boot for easy deployment and management.

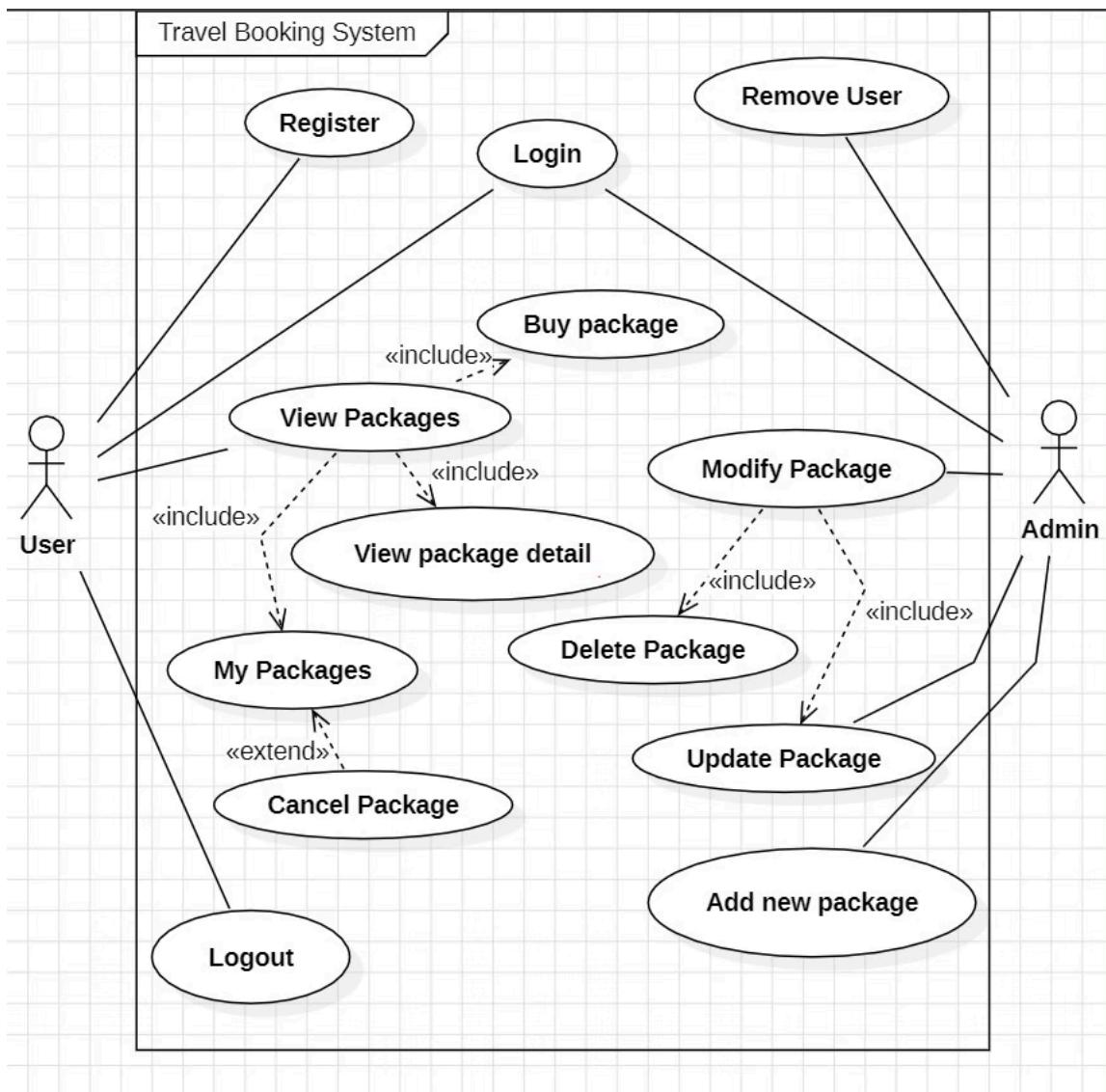
Overall, the project aims to provide a user-friendly platform for booking flights, managing bookings and administering the application.

MODELS

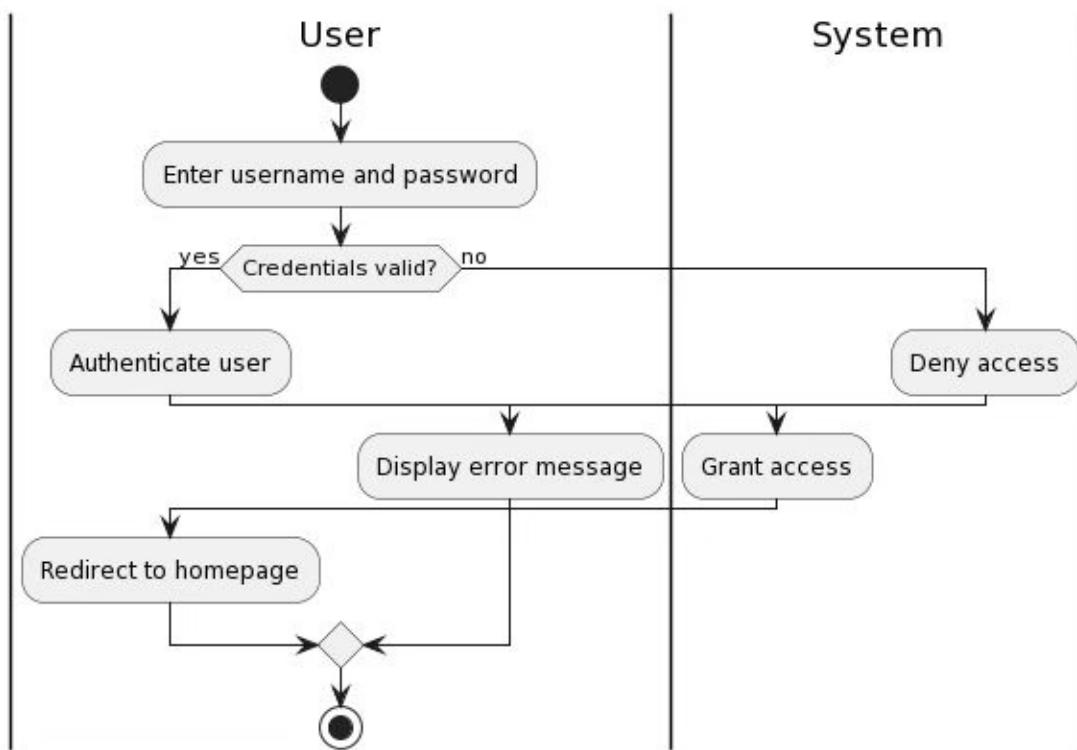
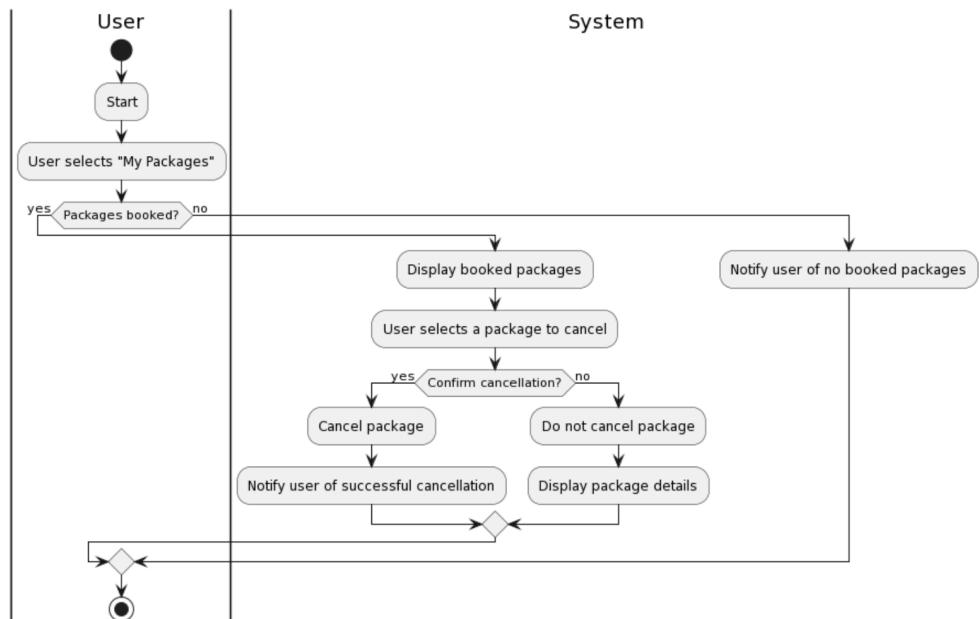
CLASS DIAGRAM:

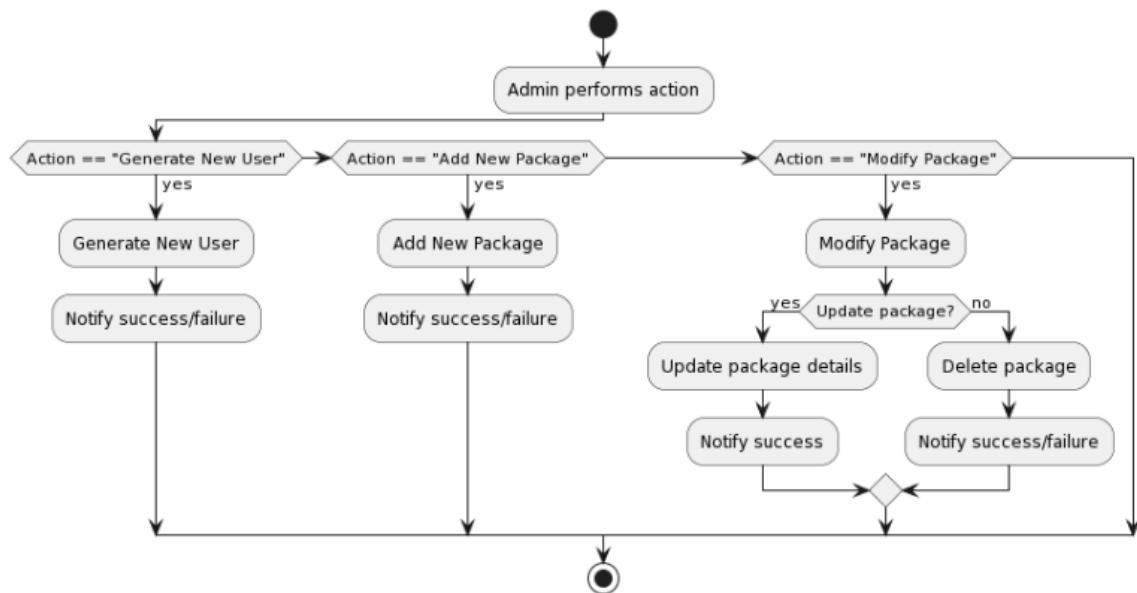
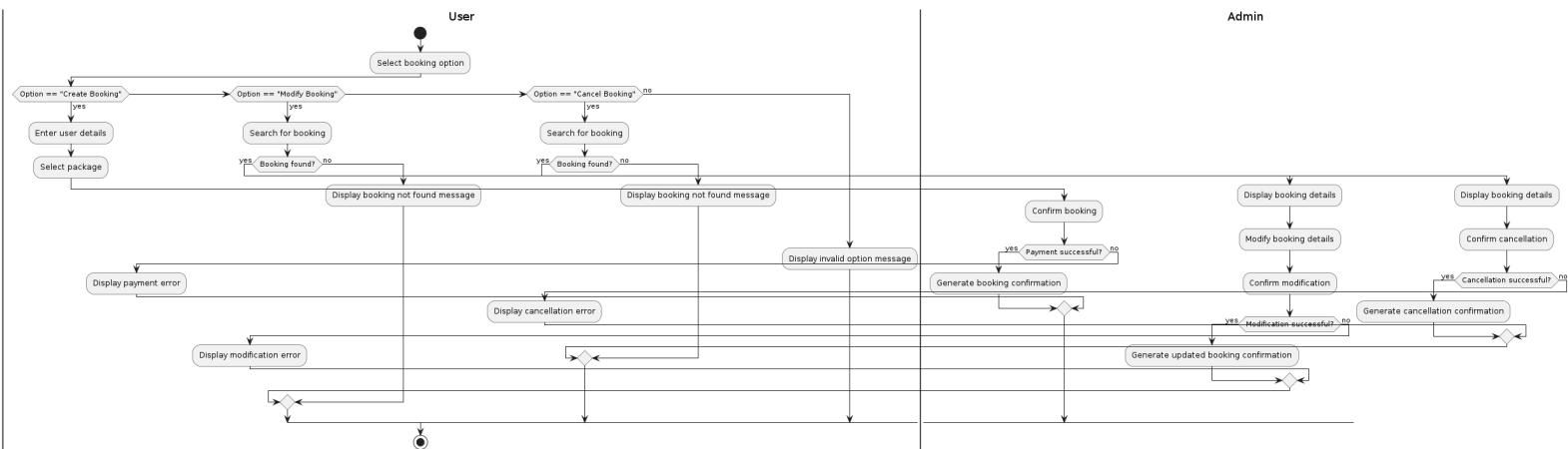


USE-CASE DIAGRAM:

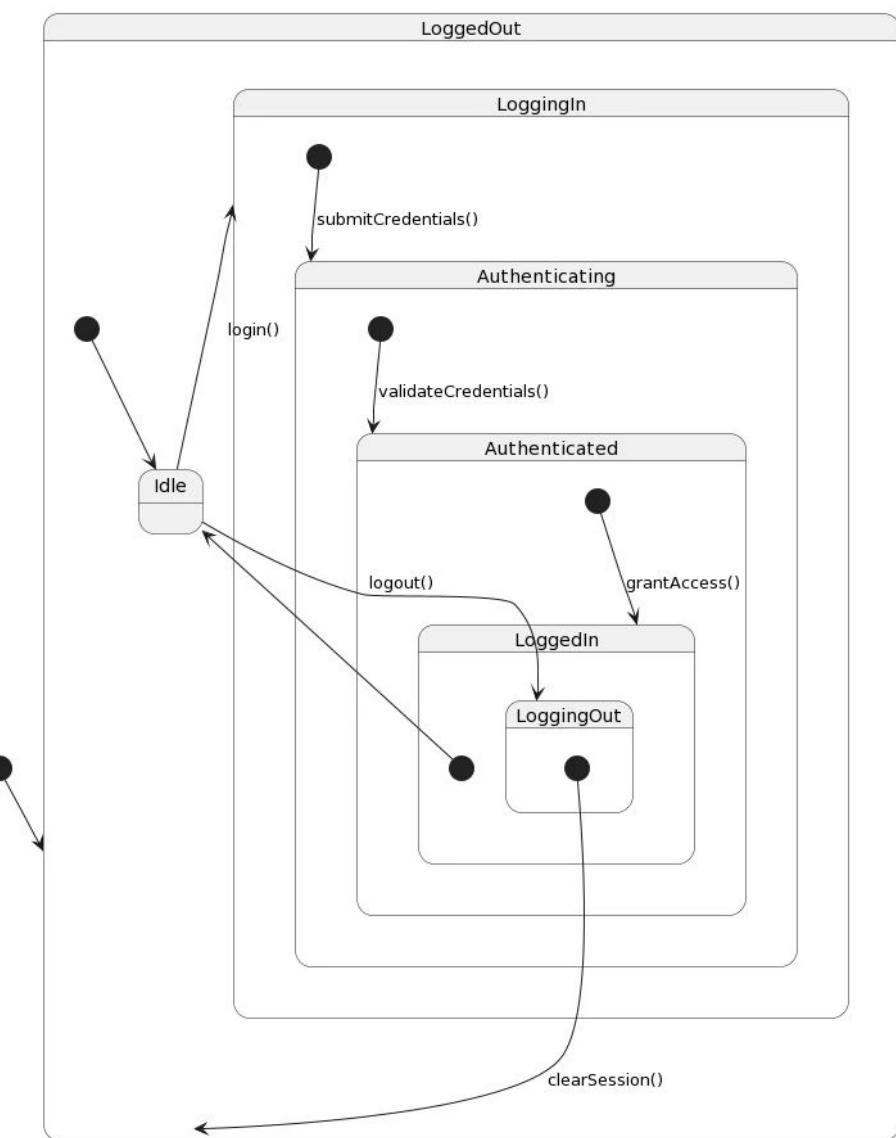


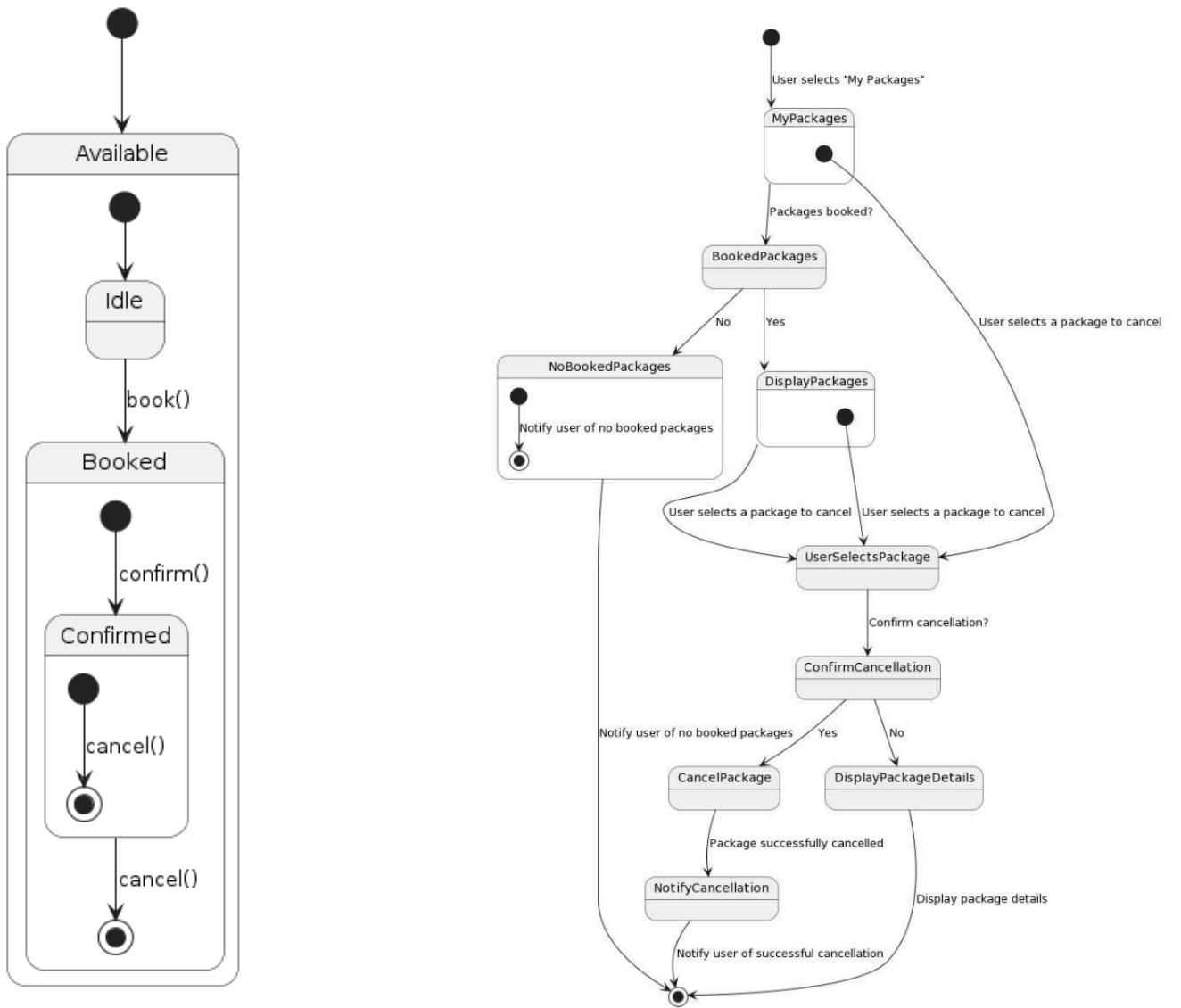
ACTIVITY DIAGRAM:

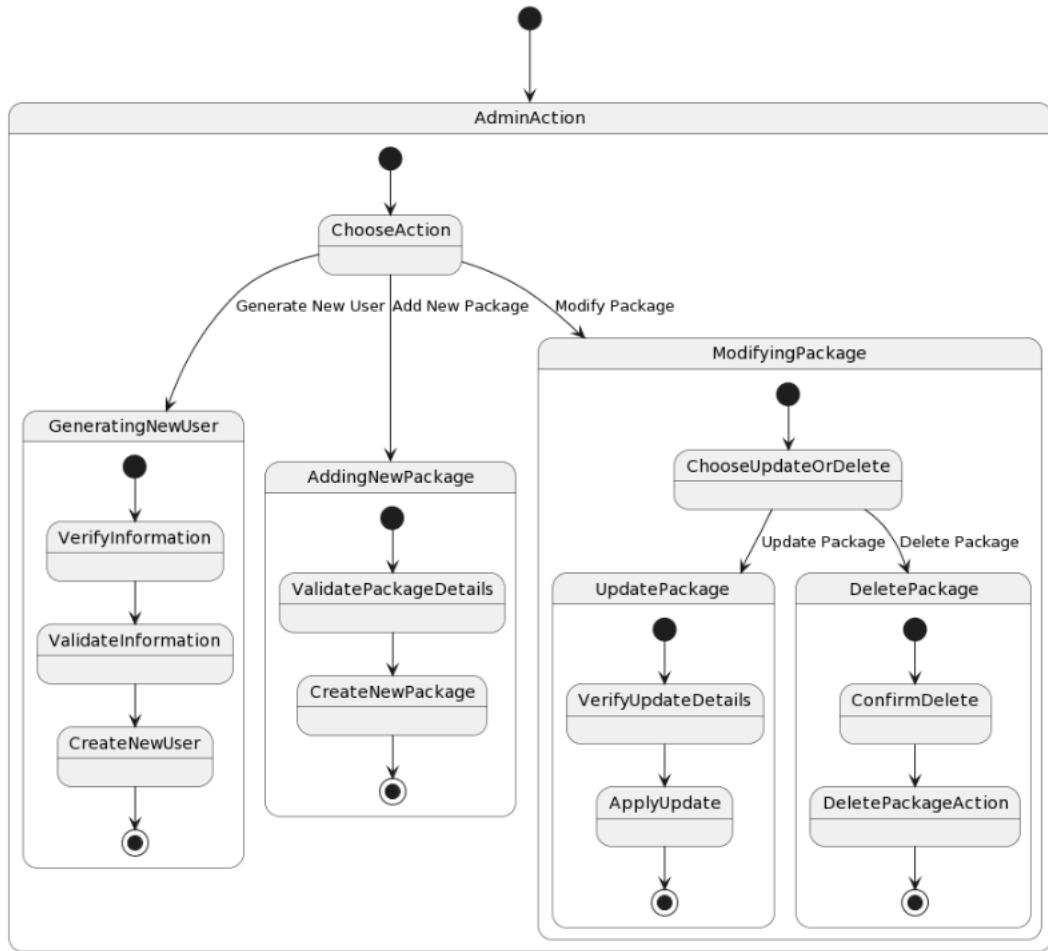




STATE DIAGRAM:







ARCHITECTURE PATTERNS

The architecture patterns used in the project are:

1) Model-View-Controller (MVC):

The system follows the MVC architectural pattern in its overall structure. Entities such as Flight, Person and Bookings represent the model layer, encapsulating data logic. Services such as FlightService, UserService, and BookingService along with FlightController, UserController and BookingController serve as the controller layer, handling application logic and coordinating interactions between the model and view layers. While the view layer is not explicitly represented in the provided code, it would typically include user interfaces such as web pages, API endpoints for interacting with the system.

2) Service-Oriented Architecture (SOA):

The system adopts a service-oriented architecture by organising functionality into individual services responsible for specific business capabilities. The UserService, FlightService and BookingService encapsulate distinct sets of functionalities related to user management, flight operations and booking management, respectively. This architectural approach promotes modularity, scalability and reusability by decoupling components and enabling them to communicate via well-defined interfaces

DESIGN PATTERNS

The design patterns used in the project are:

1) Facade Pattern:

The BookingFacade interface acts as a simplified interface for interacting with the booking system. It hides the complexities of the underlying services such as flight booking, cancellation and retrieval behind a complete interface. Clients interact with the facade rather than directly with the individual services, promoting simplicity and reducing coupling.

Users will use the UI components, utilise the BookingFacade to perform booking related operations without needing to understand the complete details of how those operations are implemented.

2) Strategy Pattern:

The Discount interface, along with its concrete implementations FixedDiscount and PercentageDiscount, embodies the Strategy pattern. Each discount strategy is encapsulated within its own class, allowing the system to easily switch between different discount algorithms at runtime.

By employing the Strategy pattern, the system achieves flexibility in applying different discount strategies based on certain conditions. For instance, it applies a fixed discount for bookings over a certain price threshold and a percentage discount for bookings below that threshold.

DESIGN PRINCIPLES

The design principles used in the project are:

1) Single Responsibility Principle (SRP):

Each class and method within the system is designed to have a single responsibility. Like, the BookingService is responsible for managing bookings, while the FlightService handles flight related operations like booking, cancelling, modifying date. This principle helps in maintaining code that is easier to understand, modify and test.

2) Open Closed Principle (OCP):

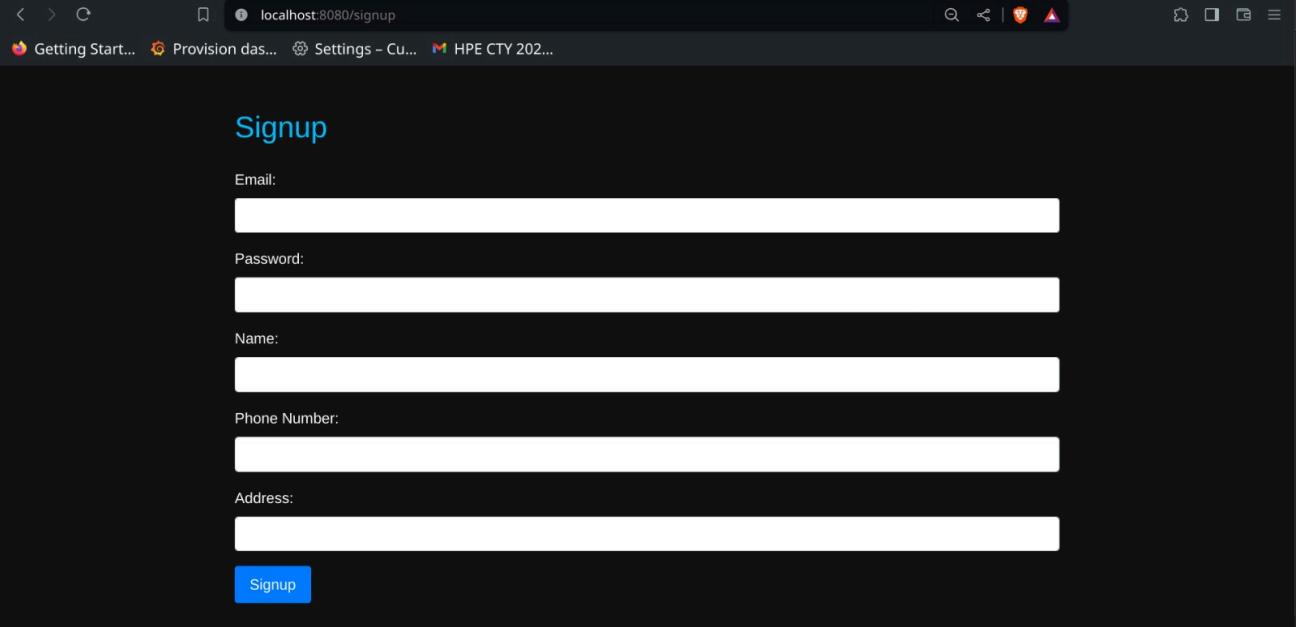
The project design adheres to the Open Closed Principle by favouring composition over inheritance and designing components to be easily extendable without modification. Like, the system can accommodate new discount strategies such as FixedDiscount, PercentageDiscount without altering existing code by implementing the Strategy pattern.

INDIVIDUAL CONTRIBUTIONS

- 1) Adithya TG -** Created the Admin html dynamic page and the related controller and the service java file and integrated it with the Postgres SQL database to do flight modification services.
- 2) Akshar S -** Created the Base framework and the setup the SpringBoot Java architecture and built the overall UI along with the integration of all the models, like the Flight model, Bookings model and the User model and established the Backend connection to the UI.
- 3) Adithya Mahesh -** Added the Discount feature to the Bookings model and connected it to the backend database as an extra feature called the discount cost.
- 4) Sai Amara Prasad -** Added the validation feature for login pages and setup the connection to the database to verify the kind of login (whether admin or client) and redirected based on the input.

SCREENSHOTS

1) Signup page



A screenshot of a web browser window showing a "Signup" form. The URL in the address bar is "localhost:8080/signup". The form consists of six input fields: "Email:", "Password:", "Name:", "Phone Number:", "Address:", and a blue "Signup" button at the bottom.

Signup

Email:

Password:

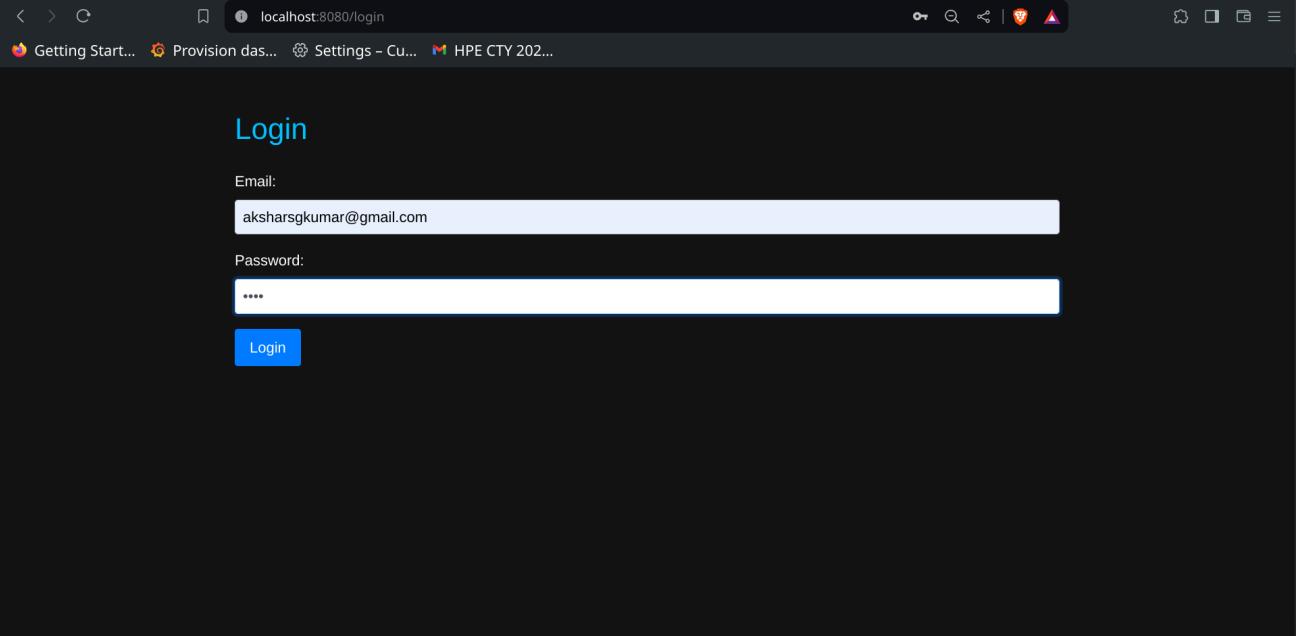
Name:

Phone Number:

Address:

Signup

2) Login page



A screenshot of a web browser window showing a "Login" form. The URL in the address bar is "localhost:8080/login". The form consists of two input fields: "Email:" containing "aksharsgkumar@gmail.com" and "Password:" containing "****", followed by a blue "Login" button.

Login

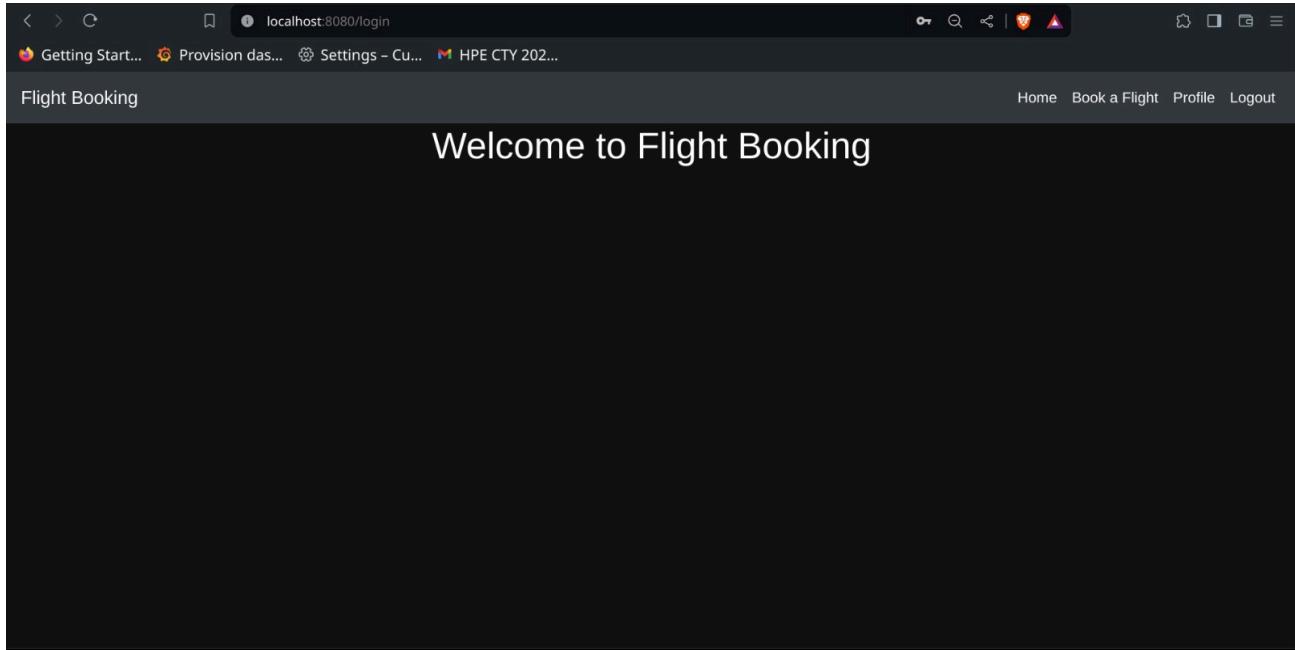
Email:

aksharsgkumar@gmail.com

Password:

Login

3) Welcome Page

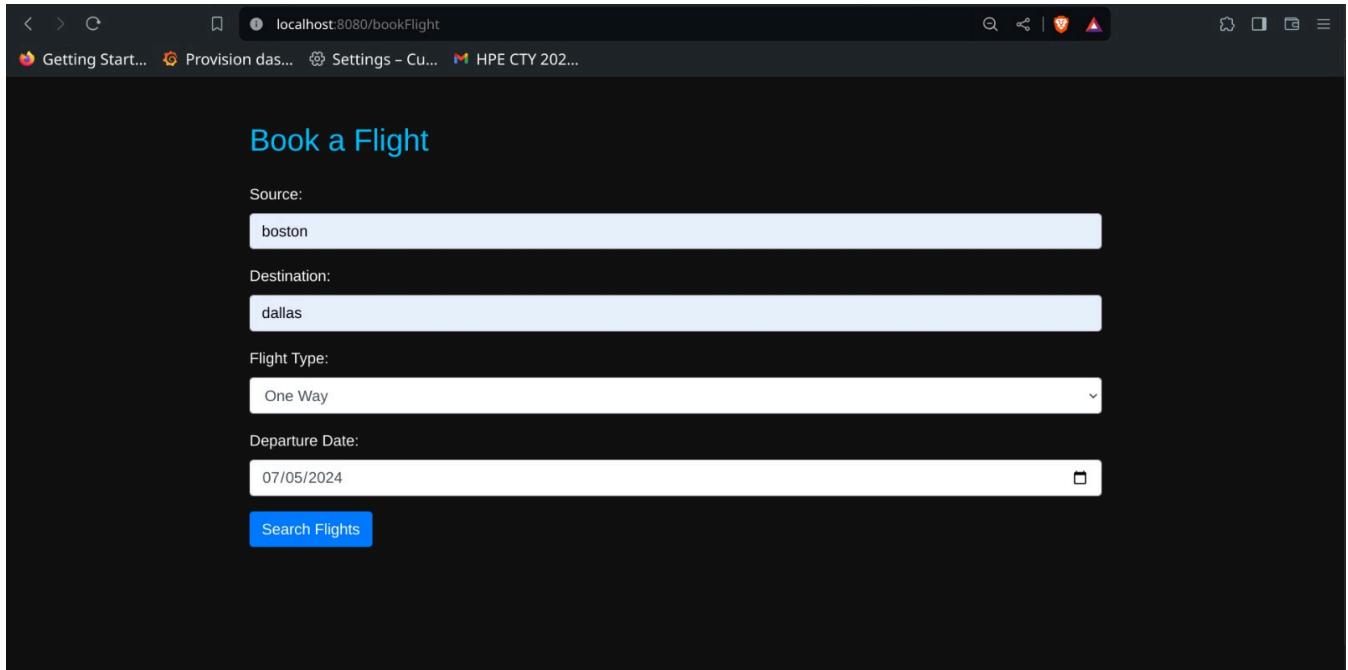


4) View Flight Options Tab:

A screenshot of a web browser window showing a table titled 'Flights'. The table lists various flight options with columns for Flight ID, Source, Destination, Departure Date, Return Date, Price, Available Seats, and Flight Type. The table has alternating row colors and a vertical scrollbar on the right side.

Flight ID	Source	Destination	Departure Date	Return Date	Price	Available Seats	Flight Type
35	Boston	Dallas	2024-08-01 00:00:00.0		450.0	208	OneWay
162	Boston	Dallas	2024-08-09 00:00:00.0		450.0	210	OneWay
161	Boston	Dallas	2024-08-05 00:00:00.0		450.0	206	OneWay
160	Boston	Dallas	2024-08-01 00:00:00.0		450.0	207	OneWay
37	Houston	Phoenix	2024-10-20 00:00:00.0		420.0	175	OneWay
42	St. Louis	Tampa	2025-03-15 00:00:00.0	2025-03-25 00:00:00.0	510.0	170	TwoWay
44	Salt Lake City	Cincinnati	2025-05-20 00:00:00.0	2025-05-30 00:00:00.0	540.0	165	TwoWay
46	Kansas City	Raleigh	2025-07-15 00:00:00.0	2025-07-25 00:00:00.0	490.0	170	TwoWay
47	Indianapolis	Nashville	2025-08-01 00:00:00.0		460.0	200	OneWay
48	Oklahoma City	Sacramento	2025-09-10 00:00:00.0	2025-09-20 00:00:00.0	530.0	175	TwoWay
49	El Paso	Albuquerque	2025-10-20 00:00:00.0		410.0	190	OneWay
50	Louisville	Milwaukee	2025-11-15 00:00:00.0	2025-11-25 00:00:00.0	490.0	165	TwoWay
51	Memphis	Tulsa	2025-12-05 00:00:00.0		500.0	210	OneWay
52	Baltimore	Virginia Beach	2026-01-10 00:00:00.0	2026-01-20 00:00:00.0	570.0	185	TwoWay
53	Atlanta	Charlotte	2026-02-20 00:00:00.0		450.0	175	OneWay

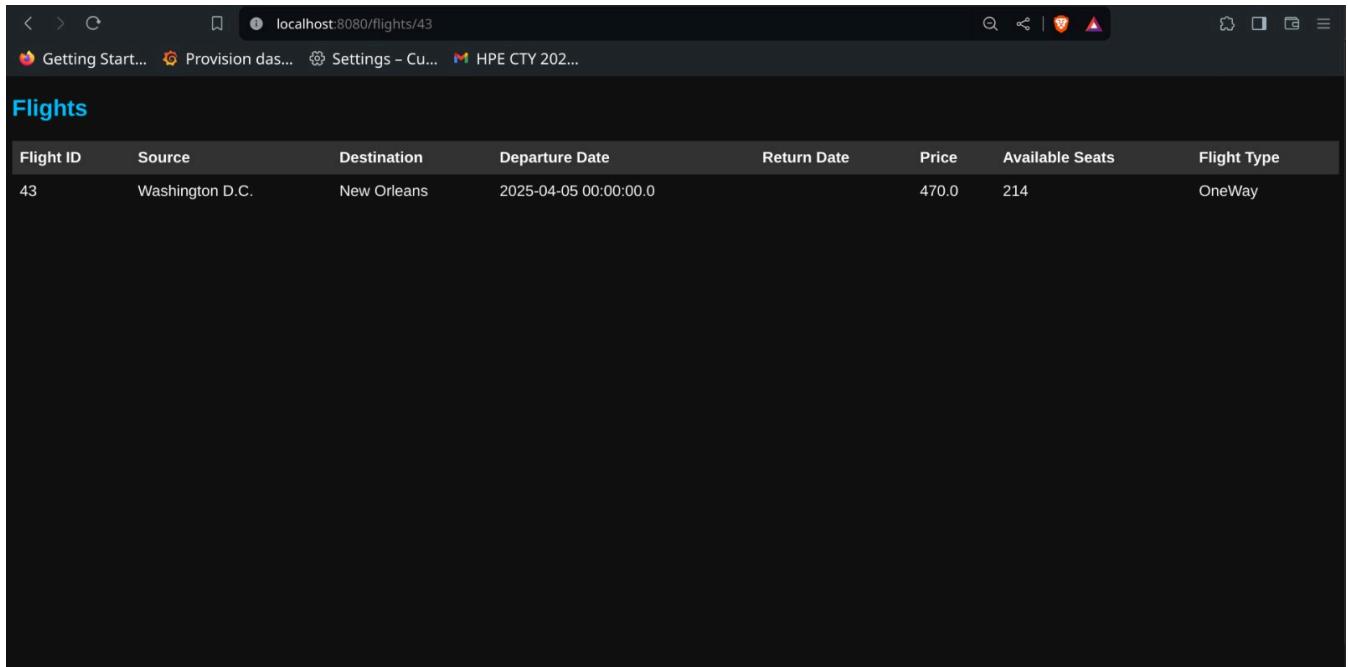
5) Book Flights Tab



A screenshot of a web browser showing a flight booking form titled "Book a Flight". The form has the following fields:

- Source: boston
- Destination: dallas
- Flight Type: One Way
- Departure Date: 07/05/2024
- Search Flights button

6) Search Result for Flight tab



A screenshot of a web browser showing a table of flight search results titled "Flights". The table has the following columns and data:

Flight ID	Source	Destination	Departure Date	Return Date	Price	Available Seats	Flight Type
43	Washington D.C.	New Orleans	2025-04-05 00:00:00.0		470.0	214	OneWay

7) View Flight Details page:

The screenshot shows a web browser window with the URL `localhost:8080/finalBook/35`. The title bar says "Flight Details". The main content area has a section titled "Flight Information" containing the following details:

- Flight ID: 35
- Source: Boston
- Destination: Dallas
- Departure Date: 2024-08-01 00:00:00.0
- Return Date:
- Price: 450.0
- Available Seats: 208
- Flight Type: OneWay

Below this section is a blue "Book" button.

8) User profile page:

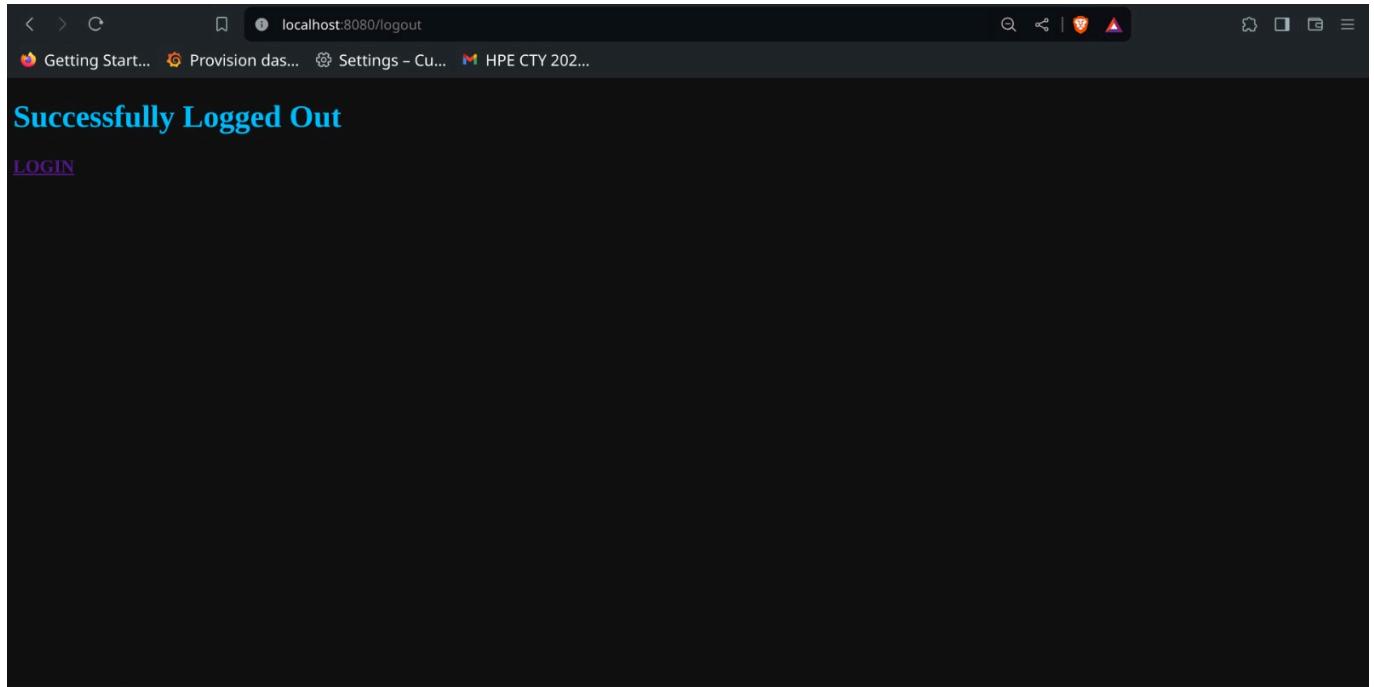
The screenshot shows a web browser window with the URL `localhost:8080/profile`. The title bar says "User Profile". The main content area has a section titled "User Information" containing the following details:

- User ID: 1
- Name: John Doe
- Email: johndoe@example.com
- Phone Number: 1234567890
- Address: 123 Main St

Below this section is a section titled "Bookings" with a table showing the following data:

Booking ID	View	Booking Date	Status	Discount	Discount Availed?
29	Flight Details	2024-04-15	confirmed	-50.0	true
30	Flight Details	2024-04-15	confirmed	-50.0	true
31	Flight Details	2024-04-15	confirmed	-50.0	true
32	Flight Details	2024-04-15	confirmed	-50.0	true

9) Logging out page



10) Admin Page

11) Admin Page deleting a flight

a) Page

The screenshot shows a web-based admin interface with two main sections: "Bookings" and "Flights".

Bookings:

Booking ID	User ID	Flight ID	Booking Date	Status	Delete
1	1	1	2024-04-05	Confirmed	1
2	1	3	2024-04-06	Pending	2
3	2	2	2024-04-07	Confirmed	3

Flights:

Flight ID	Source	Destination	Departure Date	Return Date	Price	Available Seats	Flight Type	Delete
1	New York	Los Angeles	2024-05-01 00:00:00.0		500.0	200	OneWay	1
2	Chicago	Miami	2024-06-10 00:00:00.0	2024-06-15 00:00:00.0	600.0	150	TwoWay	2
3	San Francisco	Seattle	2024-07-20 00:00:00.0		400.0	180	OneWay	3
4	Boston	Dallas	2024-08-01 00:00:00.0		450.0	210	OneWay	4
5	Denver	Atlanta	2024-09-10 00:00:00.0	2024-09-20 00:00:00.0	550.0	190	TwoWay	5
6	Houston	Phoenix	2024-10-20 00:00:00.0		420.0	175	OneWay	6
7	Las Vegas	Orlando	2024-11-15 00:00:00.0	2024-11-25 00:00:00.0	480.0	160	TwoWay	7
8	Detroit	Minneapolis	2024-12-05 00:00:00.0		530.0	205	OneWay	8
9	Philadelphia	San Diego	2025-01-10 00:00:00.0	2025-01-20 00:00:00.0	560.0	195	TwoWay	9
10	Portland	Charlotte	2025-02-20 00:00:00.0		440.0	185	OneWay	10
11	St. Louis	Tampa	2025-03-15 00:00:00.0	2025-03-25 00:00:00.0	510.0	170	TwoWay	11
12	Washington D.C.	New Orleans	2025-04-05 00:00:00.0		470.0	215	OneWay	12

b) Table before deleting

flight_id	source	destination	departure_date	return_date	price	available_seats	flight_type
1	New York	Los Angeles	2024-05-01		500	200	OneWay
2	Chicago	Miami	2024-06-10	2024-06-15	600	150	TwoWay
3	San Francisco	Seattle	2024-07-20		400	180	OneWay
4	Boston	Dallas	2024-08-01		450	210	OneWay
5	Denver	Atlanta	2024-09-10	2024-09-20	550	190	TwoWay
6	Houston	Phoenix	2024-10-20		420	175	OneWay
7	Las Vegas	Orlando	2024-11-15	2024-11-25	480	160	TwoWay
8	Detroit	Minneapolis	2024-12-05		530	205	OneWay
9	Philadelphia	San Diego	2025-01-10	2025-01-20	560	195	TwoWay
10	Portland	Charlotte	2025-02-20		:		

c) Page while deleting flight-3

The screenshot shows an 'Admin Page' interface with two main sections: 'Bookings' and 'Flights'. In the 'Bookings' section, there is a table with columns: Booking ID, User ID, Flight ID, Departure Date, Status, and Delete. Three rows are listed: row 1 (User ID 1, Flight ID 1, Departure Date 2024-04-05, Status Confirmed, Delete 1), row 2 (User ID 1, Flight ID 3, Departure Date 2024-04-06, Status Pending, Delete 2), and row 3 (User ID 2, Flight ID 2, Departure Date 2024-04-07, Status Confirmed, Delete 3). A modal dialog box is overlaid on the page, displaying the message "localhost:8080 says flight deleted successfully!" with an "OK" button. In the 'Flights' section, there is another table with similar columns: Flight ID, Source, Destination, Departure Date, Return Date, Price, Available Seats, Flight Type, and Delete. Thirteen rows are listed, numbered 1 through 13, representing various flight routes and details.

Flight ID	Source	Destination	Departure Date	Return Date	Price	Available Seats	Flight Type	Delete
1	New York	Los Angeles	2024-05-01 00:00:00.0		500.0	200	OneWay	1
2	Chicago	Miami	2024-06-10 00:00:00.0	2024-06-15 00:00:00.0	600.0	150	TwoWay	2
3	San Francisco	Seattle	2024-07-20 00:00:00.0		400.0	180	OneWay	3
4	Boston	Dallas	2024-08-01 00:00:00.0		450.0	210	OneWay	4
5	Denver	Atlanta	2024-09-10 00:00:00.0	2024-09-20 00:00:00.0	550.0	190	TwoWay	5
6	Houston	Phoenix	2024-10-20 00:00:00.0		420.0	175	OneWay	6
7	Las Vegas	Orlando	2024-11-15 00:00:00.0	2024-11-25 00:00:00.0	480.0	160	TwoWay	7
8	Detroit	Minneapolis	2024-12-05 00:00:00.0		530.0	205	OneWay	8
9	Philadelphia	San Diego	2025-01-10 00:00:00.0	2025-01-20 00:00:00.0	560.0	195	TwoWay	9
10	Portland	Charlotte	2025-02-20 00:00:00.0		440.0	185	OneWay	10
11	St. Louis	Tampa	2025-03-15 00:00:00.0	2025-03-25 00:00:00.0	510.0	170	TwoWay	11
12	Washington D.C.	New Orleans	2025-04-05 00:00:00.0		470.0	215	OneWay	12
13	Salt Lake City	Cincinnati	2025-05-20 00:00:00.0	2025-05-30 00:00:00.0	540.0	165	TwoWay	13

d) Table after deleting flight-3

```
[travel=# select * from flight;
```

flight_id	source	destination	departure_date	return_date	price	available_seats	flight_type
1	New York	Los Angeles	2024-05-01		500	200	OneWay
2	Chicago	Miami	2024-06-10	2024-06-15	600	150	TwoWay
4	Boston	Dallas	2024-08-01		450	210	OneWay
5	Denver	Atlanta	2024-09-10	2024-09-20	550	190	TwoWay
6	Houston	Phoenix	2024-10-20		420	175	OneWay
7	Las Vegas	Orlando	2024-11-15	2024-11-25	480	160	TwoWay
8	Detroit	Minneapolis	2024-12-05		530	205	OneWay
9	Philadelphia	San Diego	2025-01-10	2025-01-20	560	195	TwoWay
10	Portland	Charlotte	2025-02-20		440	185	OneWay
11	St. Louis	Tampa	2025-03-15	2025-03-25	510	170	TwoWay
12	Washington D.C.	New Orleans	2025-04-05		470	215	OneWay
13	Salt Lake City	Cincinnati	2025-05-20	2025-05-30	540	165	TwoWay

PROJECT GITHUB LINK

https://github.com/adithyatg/Travel_Management_System