A PROJECT ON

Airline Reservation System

SUBMITTED IN

PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR THE COURSE OF

DIPLOMA IN ADVANCE COMPUTING

FROM CDAC

SUNBEAM INSTITITE OF INFORMATION TECHNOLOGY, PUNE

HINJEWADI

SUBMITTED BY:

Pratik Banarase, Nikita Deshmukh, Aradhya Gudadhe, Aishwarya Dhatrak

UNDER THE GUIDANCE OF:

Miss. Pooja Jaiswal

Project Guide

Sunbeam Institute of Information Technology, Pune

ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We would like to express our sincere gratitude to our project guide Miss. Pooja Jaiswal for her guidance and support throughout this project. Without her invaluable insights and suggestions, this project would not have been possible.

We would also like to thank and express our gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mr. Yogesh Kolhe (Course Coordinator, SIIT, Pune).

We extend our appreciation to them who generously provided us with information and feedback for this project. Their cooperation was instrumental in the successful completion of this project.

We would like to acknowledge the contributions of our classmates and friends who provided their assistance and support throughout the project. Their feedback and encouragement helped us stay motivated and on track.

Last but not the least we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Table of Contents

1. INTRODUCTION

The web based "Airline Reservation System" project is an attempt to stimulate the basic concepts of airline reservation system. The system enables the customer to do the things such as search for airline flights on a specified date and choose a flight based on the details and reservation of flight.

The system provides you Quick Search facility that provides you details about flights without login. But if user want to book ticket then it must require login into our account.

The system allows the airline passenger to search for flights that are available between the two travel cities, namely the "Departure city" and "Destination city" for a particular departure and arrival dates. The system displays all the flight's details such as flight no, name, price, time of journey etc.

Here we provided quick search facility which displays list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight. If the seats are available then the system allows the passenger to book a seat. Otherwise it asks the user to choose another flight.

To book a flight the system asks the customer to enter his details such as name, address, city, state, credit card number and contact number. Then it checks the validity of card and book the flight and update the airline database and user database.

1.1 Objective of Project

Airline reservation systems were first introduced in the late 1950s as relatively simple standalone systems to control flight inventory, maintain flight schedules, seat assignments and aircraft loading. The modern airline reservation system is comprehensive suite of products to provide a system that assists with a variety of airline management tasks and service customer needs from the time of initial reservation through completion of the flight.

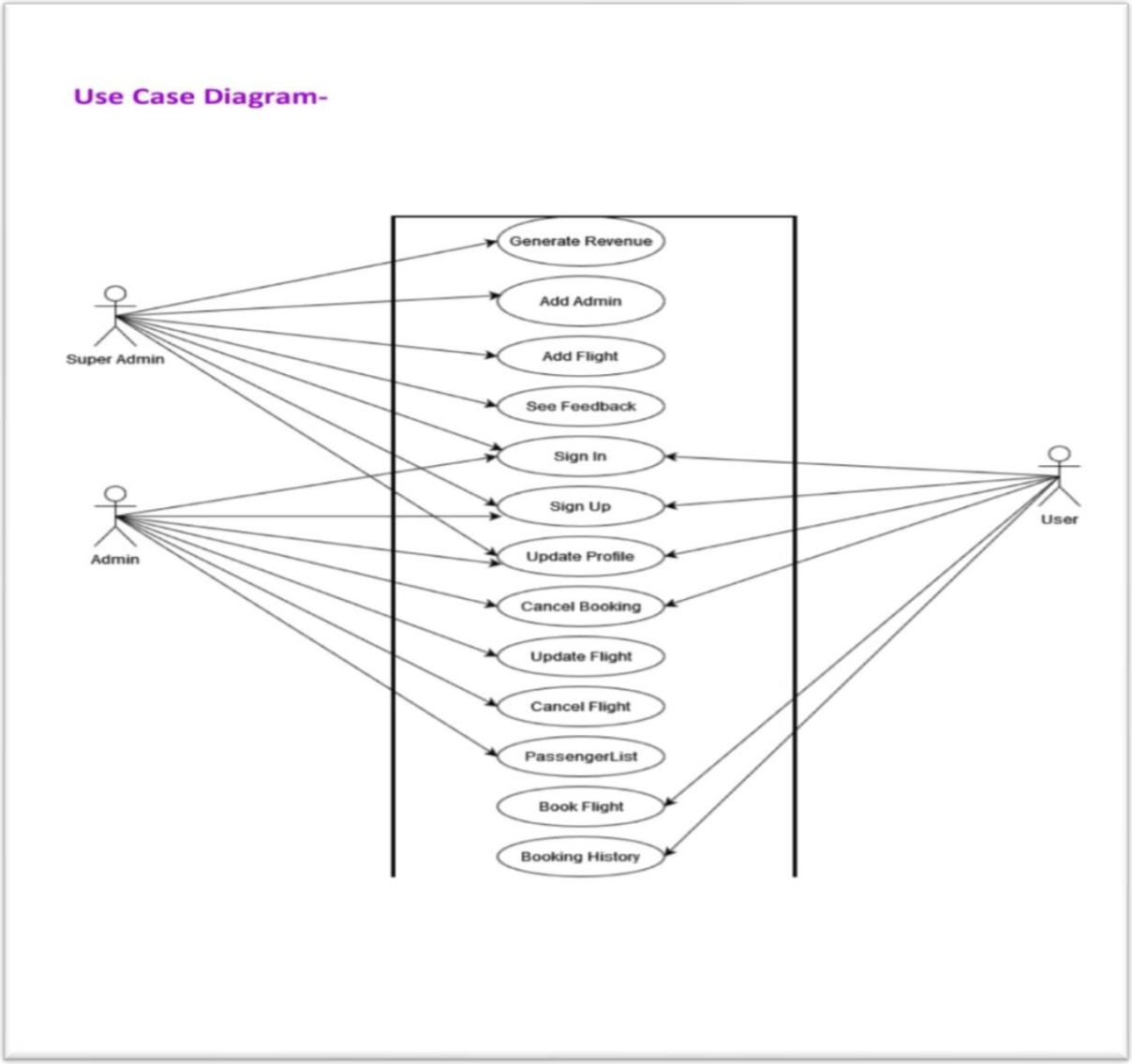
One of the most common modes of travel is traveling by air. Customers who wish to travel by air nowadays have a wide variety of airlines and a range of timings to choose from. Nowadays competition is so fierce between airlines that there are lot of discounts and a lot of luxuries given to customers that will give an edge to that particular airline.

The World Wide Web has become tremendously popular over the last four years, and currently most of the airlines have made provision for online reservation of their flights. The Internet has become a major resource for people looking for making reservations online without the hassle of meeting travel agents. My project intends to serve these purposes. It intends to check all the available airline databases and return a string of results, which can help them in their travel plans.

The objective of this project is to create an airline reservation system where a traveler can request all flight information as per their journey dates. They can get information regarding time, cost, etc all at the same time and place. When the customer calls the Counter Assistant for his/her travel needs, the counter assistant will enter the customer's details (flight requirements) in the system. The system displays all the available airlines, schedules and prices. This system would help the airline to better serve its customers by catering to their needs. The site would use a Database to hold this information as well as the latest pricing and availability information for the airlines.

2. REQUIREMENTS

2.1 Functional Requirements



2.1.1 User Account

The passenger, who will henceforth be called the 'user', will be presented with 3 choices by the reservation system, as the first step in the interaction between them. A user can choose one of these and his choice would be governed by whether he is a guest or a registered user and whether he wants to check the availability of tickets or also block/buy them. The terms 'registered user' and 'guest' are described below.

A user who has traveled by the airline earlier would have been given a user id and a password. This 'personal information' would be henceforth referred to as 'profile'. Such a user with a profile in database shall be called a 'registered user'. A registered user will be able to check the availability of tickets as well as block/buy a ticket by logging into the system.

A new user, on the other hand, would either have to

a) register himself with the system by providing personal information or

b) log into the system as a guest.

In case of 'a', the new user becomes a registered user and in case of 'b', the new user would remain a guest.

A guest can only check the availability of tickets and cannot block or buy tickets. But a registered user can also act as a guest if he only wants to check the availability of tickets. 'Availability of tickets' always refers to viewing the flight schedule for given days, the price of tickets and discount offers if any. The system shall present the user with an option to exit from the system at any time during the following processes.

2.1.2 Registration and creation of user profile

The system shall require a user to register, in order to carry out any transactions with it except for checking the availability of tickets. It will ask the user for the following information at the least – a user id, a password, first name, last name, address, phone number, email address, sex, age, preferred credit card number.

2.1.3 Quick Search

Here we provided Quick Search facility for any user to search flight schedule without login into account. This will provide user an option for searching flight and comparing their prices of all companies.

After logging in a user (either a registered user or a guest), the system shall request him to enter the following details – origin city and destination city. "City' is a generic term and refers to a city or town as the case may be. The original destination cities would be entered as text.

After the origin and destination cities are ascertained, the system shall now access the flight schedule database, referred to as 'Flight Administrator', and checks if there is a direct operational service between the two cities.

The system shall now ask the user to enter the following details - class, one-way or round trip, departure date and the number of adult passengers, children and senior citizens. 'Class' refers to Business class/Economy class. This choice shall be made by the user through a drop down menu indicating all the possible combinations of choices.

One-way/round trip shall be button selection. 'Departure date' refers to either a single date or a range of dates, entered through text box. In case, the trip is a round trip, the system shall also ask the user to enter the return date

Having taken all the above input from the user, the system checks for any false entries like the departure date on the return trip being earlier than the departure date on the onward trip. In case of incompatibility, the system will not display any flights available.

The system queries the flights database 'Flight Administrator' to check which of the flights on the schedule have seats available. The system displays the results in a suitable form (a tabular form) with the following information depicted – for each airlineId, flight number, departure time in origin city, arrival time in destination city, departure city, arrival city, Ticket price and the number of seats available on that flight.

There can be several flights of different airlines between two cities and all of them will be listed for the particular date that the user wants to depart from the Origin City. In case, the user has entered a range of dates, the system shall display all the flights for all those dates in the range. There will be a Book button in front of every row displayed in the table of flights searched.

The system will then ask for personal information of all passengers i.e. one registered user can book for multiple users. So, all users will be added in the table.

The system shall now display the price of the ticket for the trip. This will be the sum of the prices for all the members of the travel party being represented by the user.

2.1.4 Making Reservations/ Blocking/ Confirmation

After having taken the user through the step 2.1.2, checking availability, the system will now ask the user if he wishes to block/buy the ticket. If yes, and

a) if the user has been a guest, he will have to first register and become a registered user and then log onto the system.

b) if the user is already a registered user, and if he has logged on already, he can block/buy the ticket, but if he has been acting as a guest, he will have to log on.

Having ensured that the user is logged on validly, the system compares the departure date with the system date. If the departure date falls within 2 weeks of the system date, the system informs the user that he has no option to block the ticket and asks him if he would like to buy it.

If the difference between the departure date and system date is more than 2 weeks, the system asks the user if he would like to block or buy the ticket. The system informs the user that he can block the ticket at no cost now. It also informs him that if he chooses to block the ticket, he should make a final decision before 2 weeks of the departure date. The system shall send an email to the user.

Having taken the input from the user, the system shall now proceed to update the reservation database. It will decrement the number of available seats on the particular flight for the particular class by the number of travelers being represented by the user.

In case the user buys the ticket, the system asks for entering his or her bank information i.e. debit card or credit card information and then charges the price of the ticket to his debit card number.

2.1.5 View Booking History

The system shall allow a user to view all information about his previous bookings. After logging him on, it asks for his blocking number or his confirmation number. It accesses bookings table and retrieves the details of the trip and presents them to the user in a tabular format.

Admin will be able to Sign in, Sign up, Update the profile, Cancel the booking, Update Flight, Cancel flight and can also see the Passengers list.

Flight Administrator will be able to Sign in, Sign up, Update the profile, can see feedback, Add flight, Add admin and can generate the revenue.

2.2 Non-Functional Requirements

Following Non-Functional Requirements will be there in the insurance to the internet:

(i) Secure access to user's confidential data.

(ii) 24X7 availability.

(iii) Better component design to get better performance at peak time.

(iv) Flexible service-based architecture will be highly desirable for future extension. Non-Functional requirements define system properties and constraints.

Various other Non-Functional Requirements are:

- Security
- Reliability
- Maintainability
- Portability
- Extensibility
- Reusability
- Compatibility

2.2.1 Hardware & Software Requirements :

HARDWARE REQUIREMENTS

RAM	4 GB
Hard Disk	320 GB
Processor	Dual Core

SOFTWARE REQUIREMENTS

Web Browser	Google Chrome or any compatible browser
Operating System	Windows or any equivalent OS

3. DESIGN

3.1 Database Design

The following table structures depict the database design.

Table 1: users

Key Type	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
PRI	id	int	4	0
UNI	card_number	varchar	255	1
	expiry_date	date	6	1
	name_on_card	varchar	255	1
UNI	email	varchar	30	0
	first_name	varchar	30	1
	last_name	varchar	30	1
	mobile_no	varchar	15	1
	password	varchar	30	0
	status	int	4	0
	user_role	varchar	255	1

Table 2: passengers

Key Type	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
PRI	id	int	4	0
	gender	varchar	255	1
	passenger_age	int	4	0
	passenger_name	varchar	30	1
	passenger_type	varchar	255	1
	seat_number	int	4	0
MUL	booking_id	int	4	1

Table 3: bookings

Key Type	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
PRI	id	int	4	0
	airline_id	int	4	0
	arrival_date	date	6	1
	arrival_time	time	8	1
	booking_date	date	4	1
	depature_time	time	8	1
	feedback	varchar	1000	1
	journey_date	date	6	1
	seat_type	varchar	255	1
	status	int	4	0
	total_fare	double	8	0
MUL	user_id	int	4	1

Table 4: airline

Key Type	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
PRI	id	int	4	0
	airline_name	varchar	30	1
UNI	airline_no	varchar	30	1
	arrival_date	date	6	1
	arrival_time	time	8	1
	available_seats	int	4	0
	business fare	double	8	0
	capacity	int	4	0
	depature_date	date	6	1
	depature_time	time	8	1
	economy_fare	double	8	0
	from_city	varchar	30	1
	to_city	varchar	30	1

Table 5: cities

Key Type	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
PRI	id	int	4	0
	city	varchar	255	1

4. CODING STANDARD IMPLEMENTED

Naming and Capitalization :

Below summarize the naming recommendations, like identifiers in Pascal casing is mainly used (i.e. capitalize first letter of each word) with camel casing (capitalize each word except for the first one) being used in certain circumstances.

Identifier	Case	Examples	Additional Notes
Class	Pascal	Airline, Booking, CardDetails, etc	Class names should be based on "objects" or "real things" and should generally be nouns. No '_' signs allowed. Do not use type prefixes like 'C' for class.
Method	Camel	cancelFlight, addAirline, etc	Methods should use verbs or verb phrases.
Parameter	Camel	airlineDTO, userDTO, etc	Use descriptive parameter names. Parameter names should be descriptive enough that the name of the parameter and its type can be used to determine its meaning in most scenarios.
Interface	Pascal with "I" prefix	IAirlineRepo, IUserRepo, etc	Do not use the '_' sign
Property	Pascal		Use a noun or noun phrase to name properties.
Associated private member variable	CamelCase	cardDetails, userId, etc	Use underscore camel casing for the private member variables.
Exception class	Pascal with "Exception" suffix	GlobalExceptionHandler	

Comments :

- Comment each type, each non-public type member, and each region declaration.
- Use end-line comments only on variable declaration lines. End-line comments are comments that follow code on a single line.
- Separate comments from comment delimiters (apostrophe) or // with one space.
- Begin the comment text with an uppercase letter.
- End the comment with a period.
- Explain the code; do not repeat it.

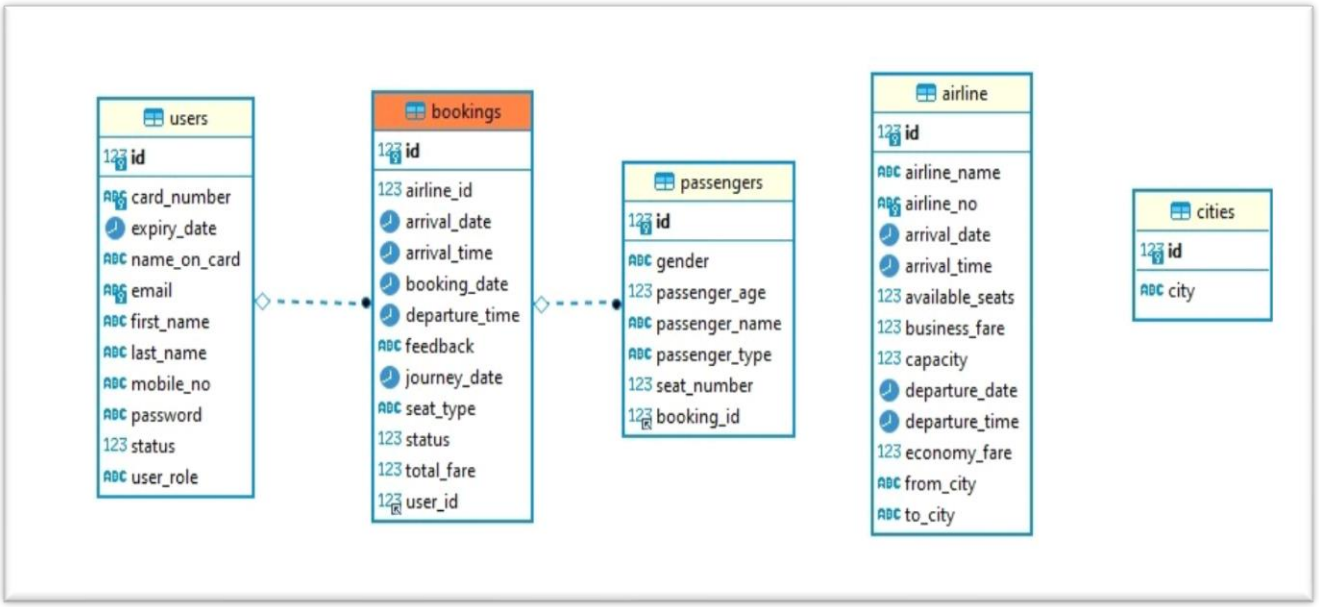
5. Test Report

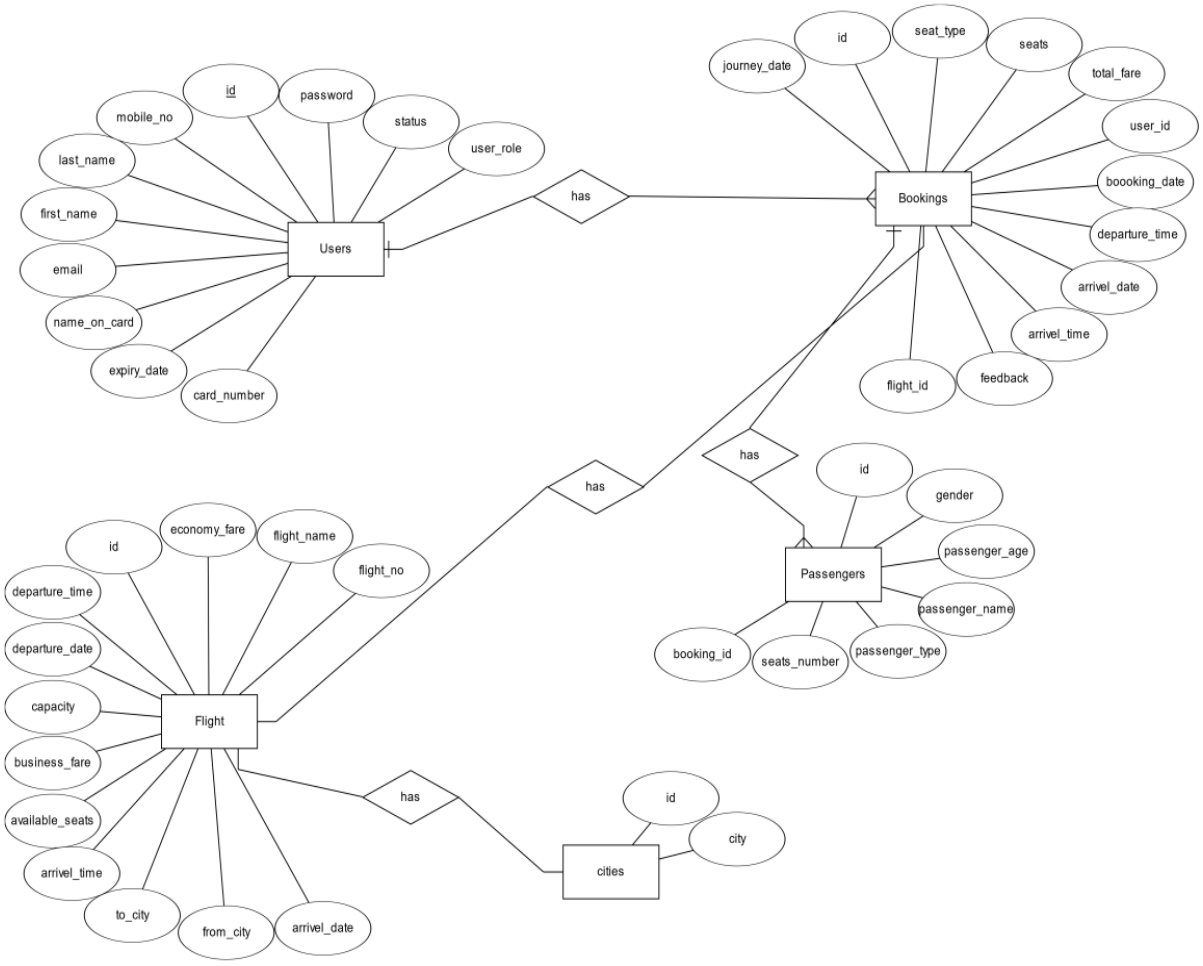
General Testing:

Sr. No	Test Case	Expected Result	Actual Result	Error Message
1	Register Page	Redirected to Next page	Ok	Nothing
2	Login Page	Redirected to Next page	Ok	Please enter username and password again
3	Reset login	Only user password will be reset	Ok	Nothing
4	Quick search flight	Gives all flight details	Ok	Nothing
5	Booking Ticket	All the fields should be filled for submission	Ok	Nothing
6	Checking login or not	User is logged in or not	Ok	Nothing
7	Add person details for tickets	Add information according to no of seats allocated	Ok	Nothing
8	Go to ticket page	Set added information about person	Ok	Nothing
9	Add information in booking table	Save this all data into booking table	Ok	Nothing
10	Transaction	On back it should be reverted to previous page	Ok	Nothing
11	View transaction done	It shows you all transactions done previously	Ok	Nothing
12	Logout	It will logout from user profile.	Ok	Nothing

Appendix A

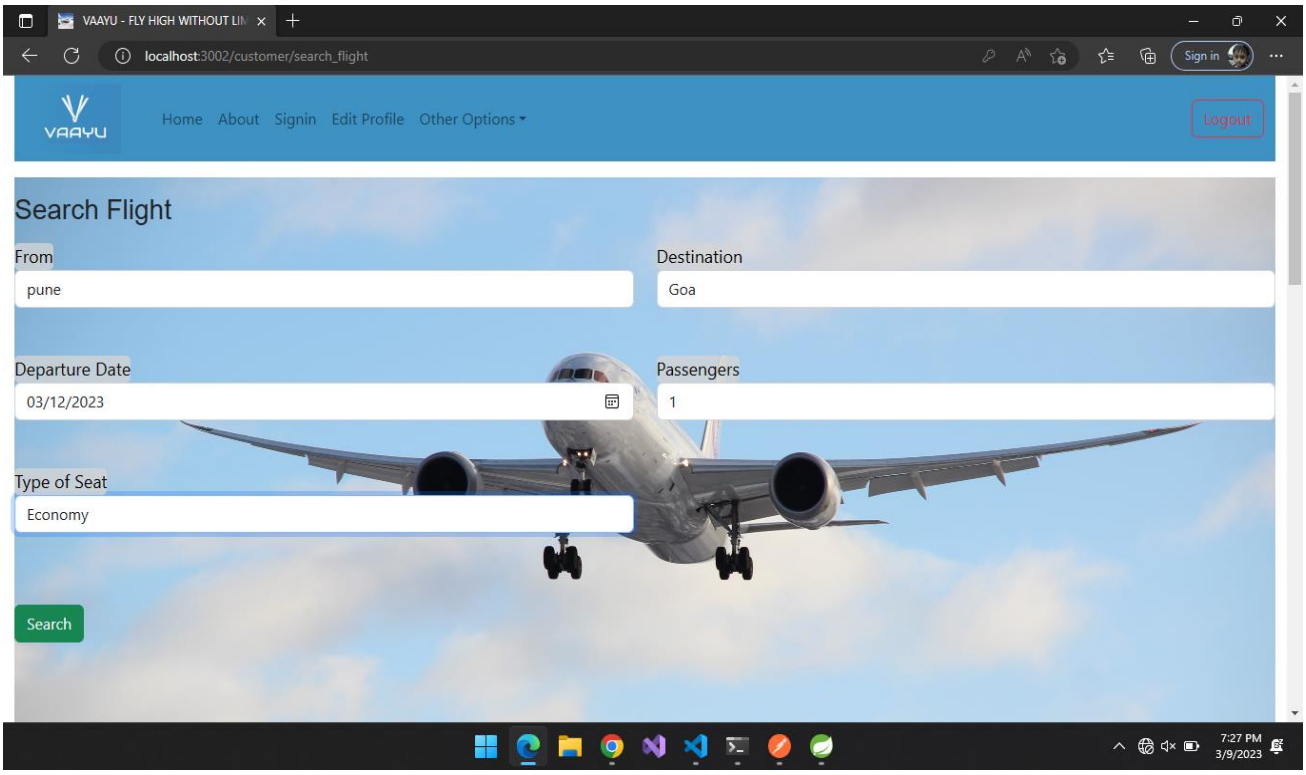
Entity Relationship Diagram



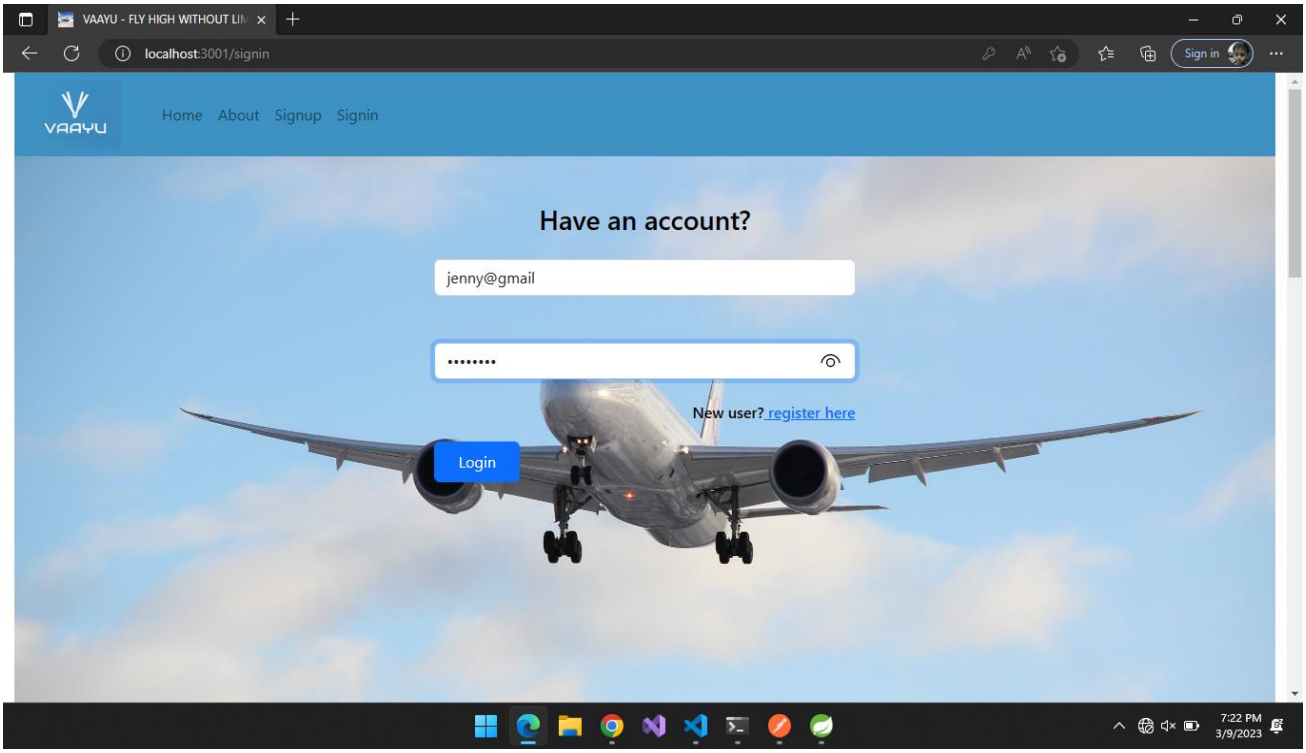


Appendix B

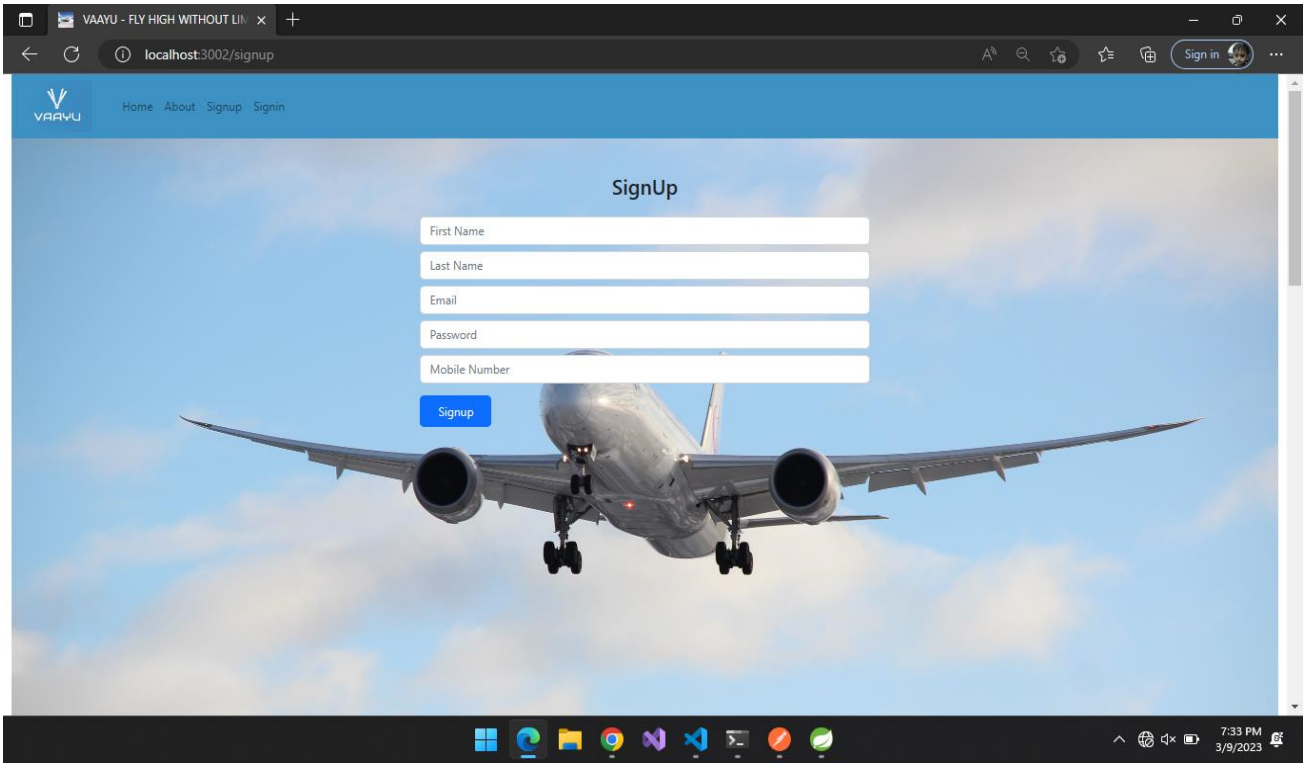
Search Flight :



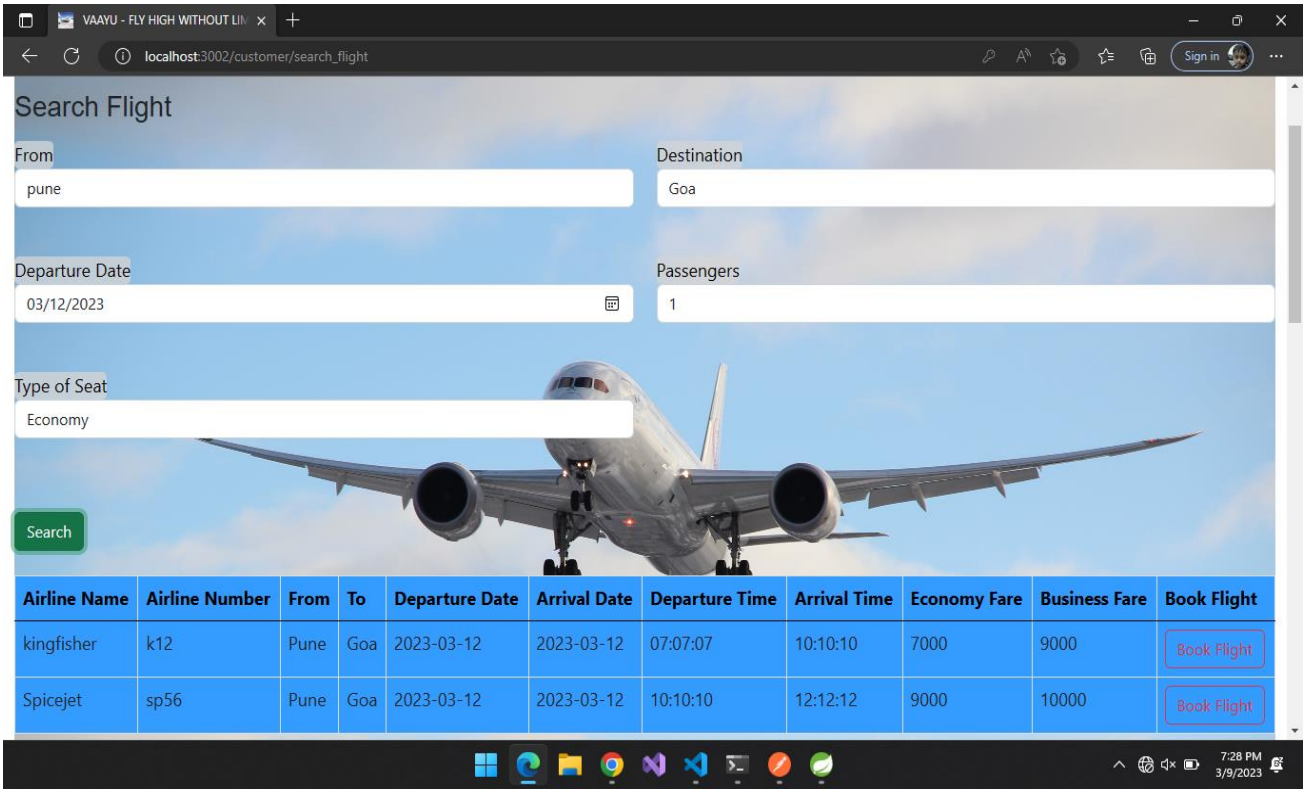
Sign-In :



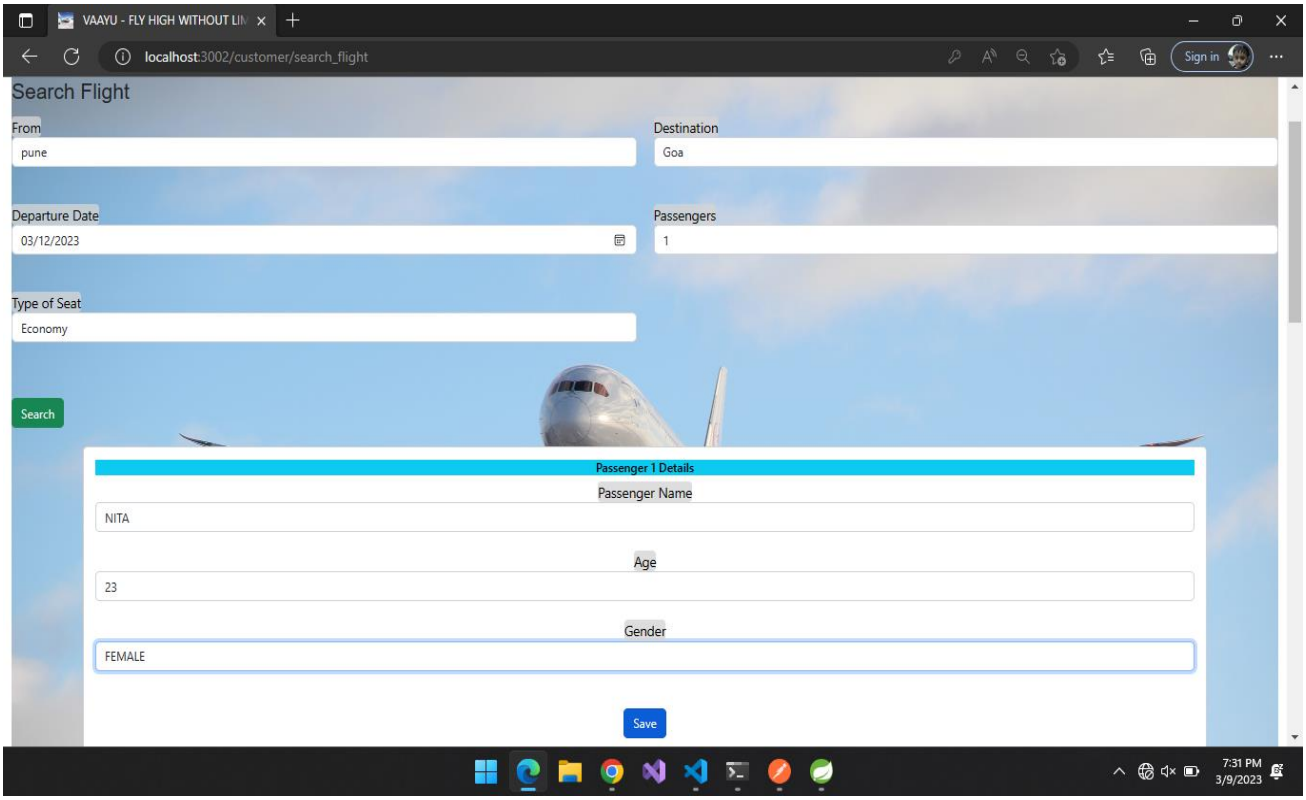
Sign-Up :



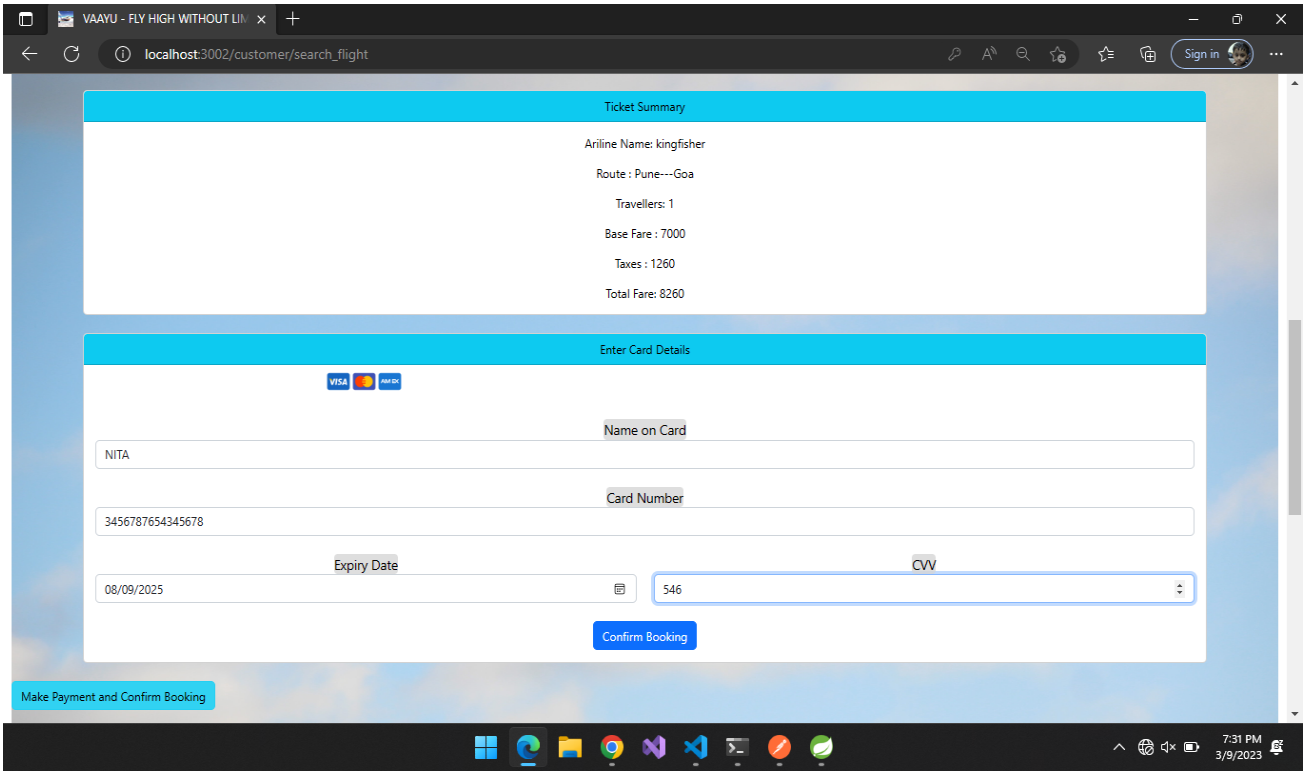
Flight Details :



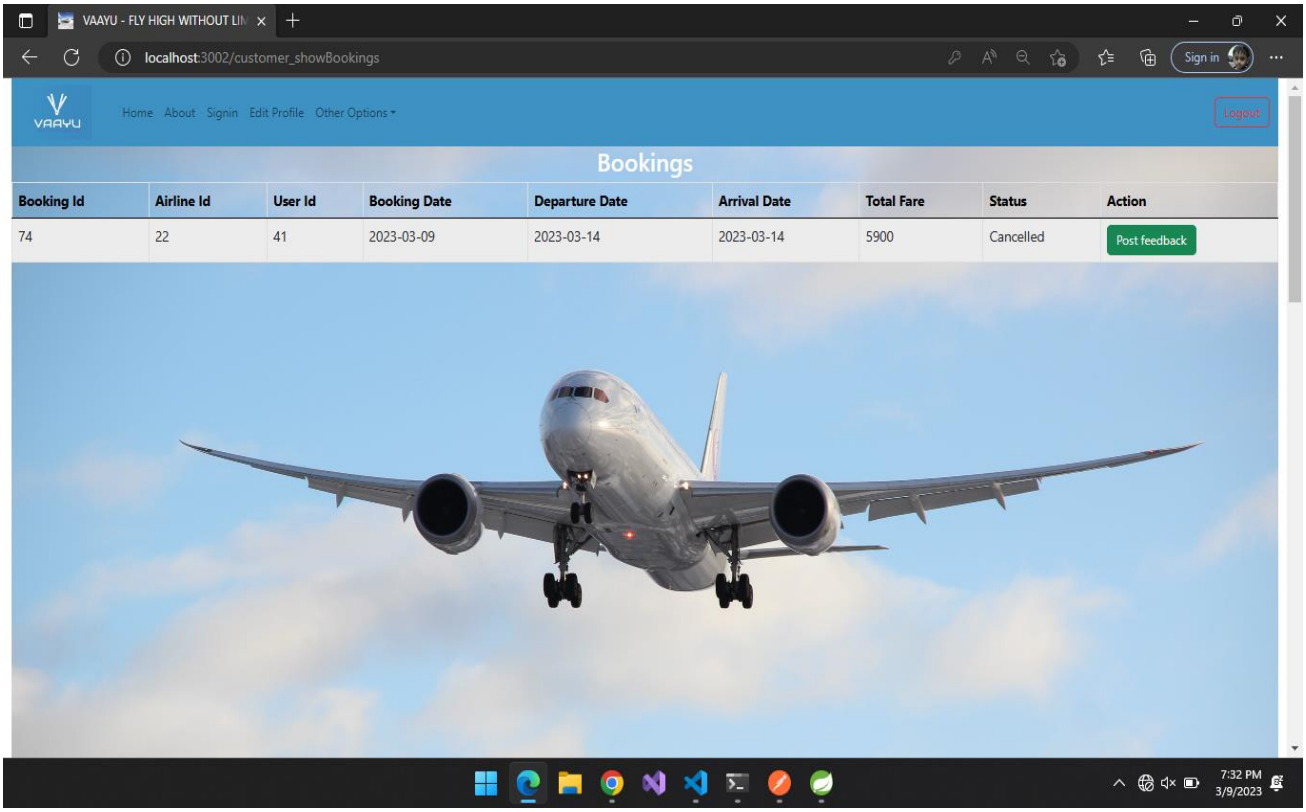
Flight Booking :



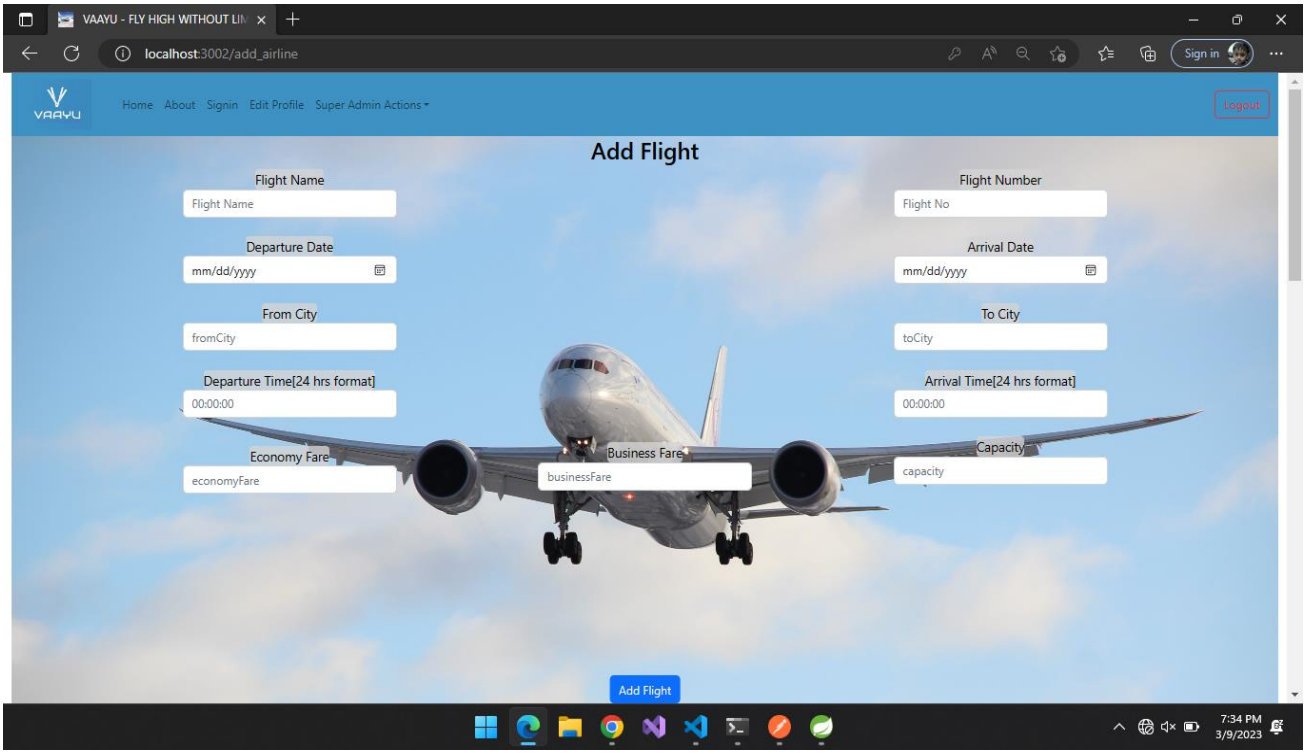
Payment Page :



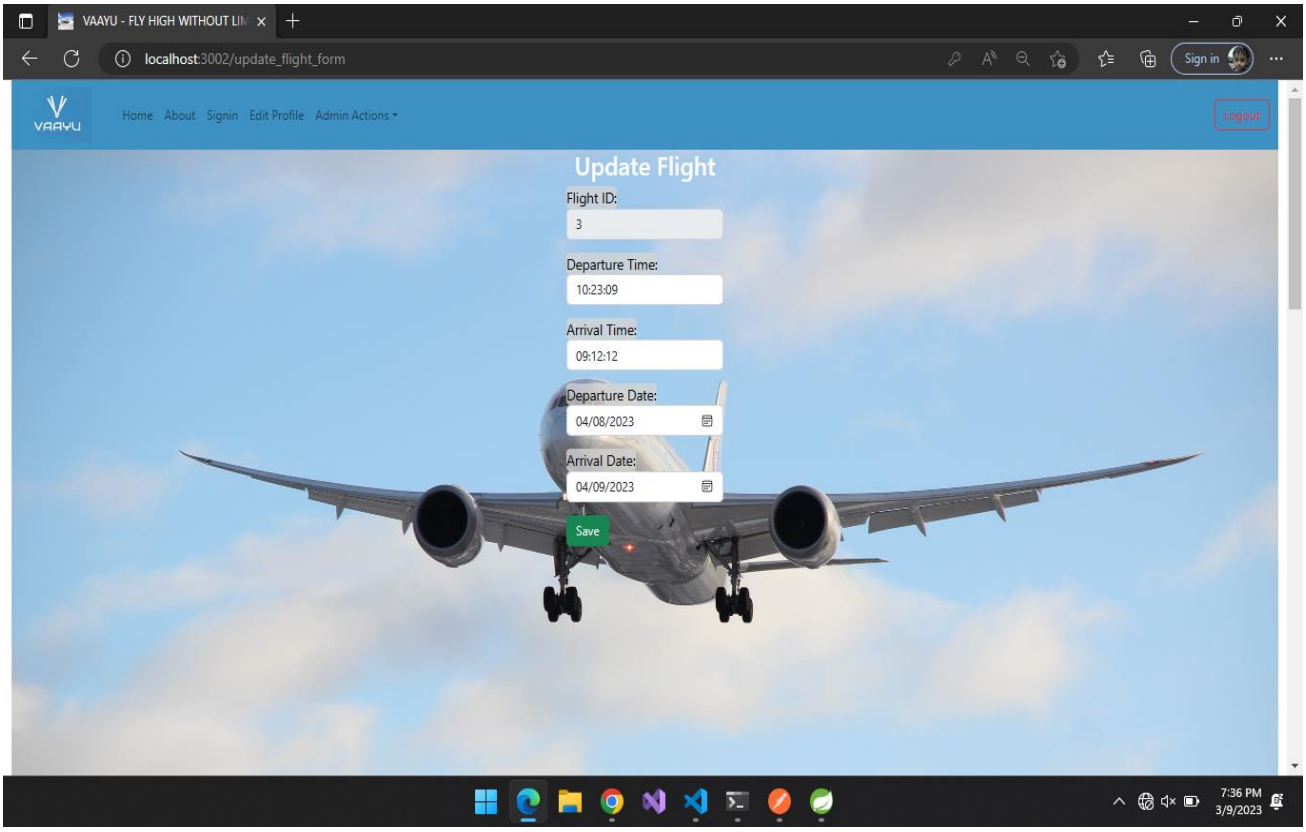
Booking Details :



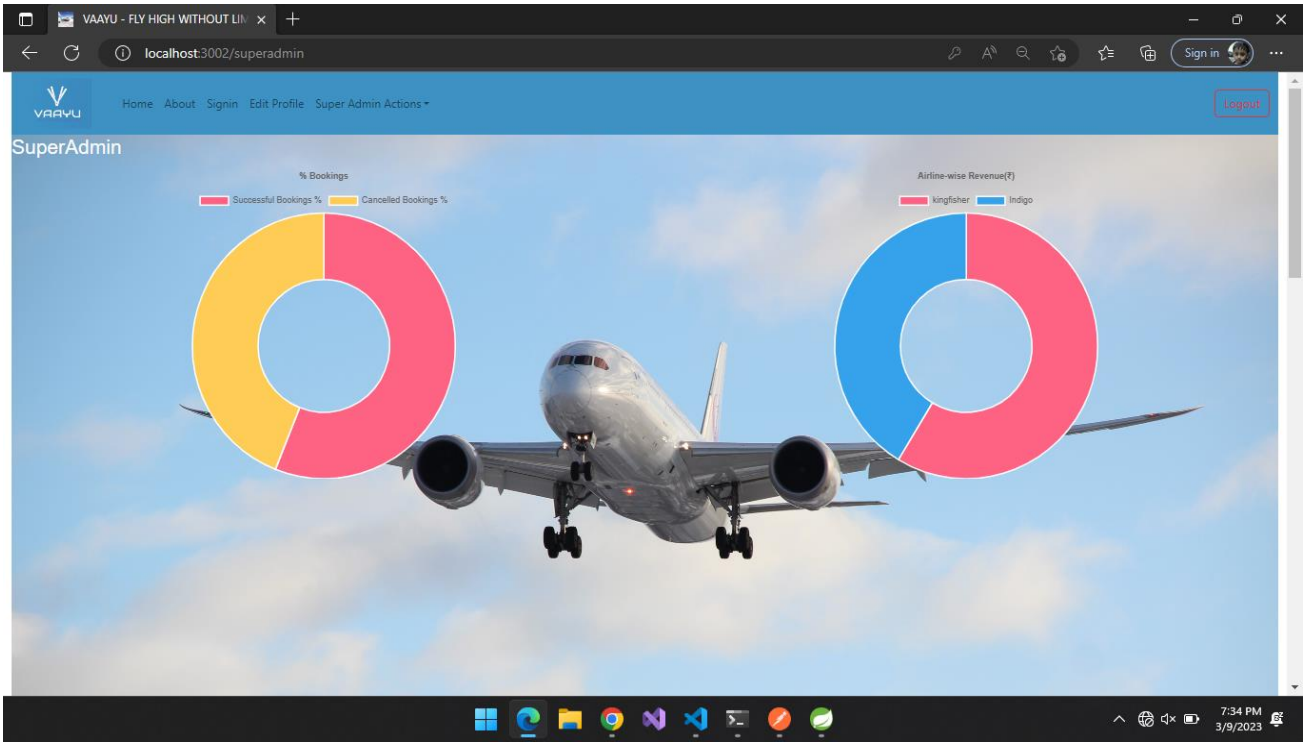
Add Flight :



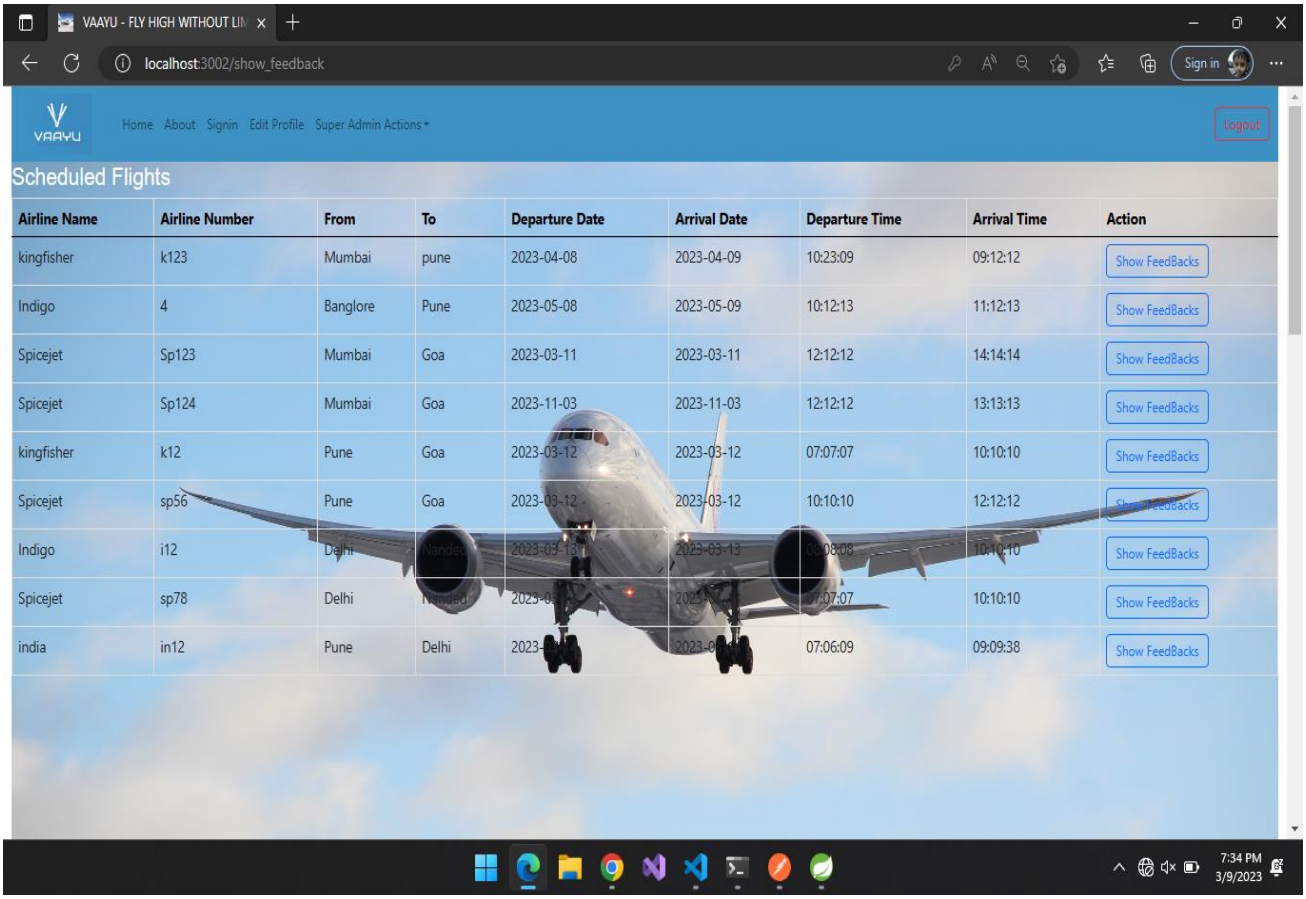
Update Flight :



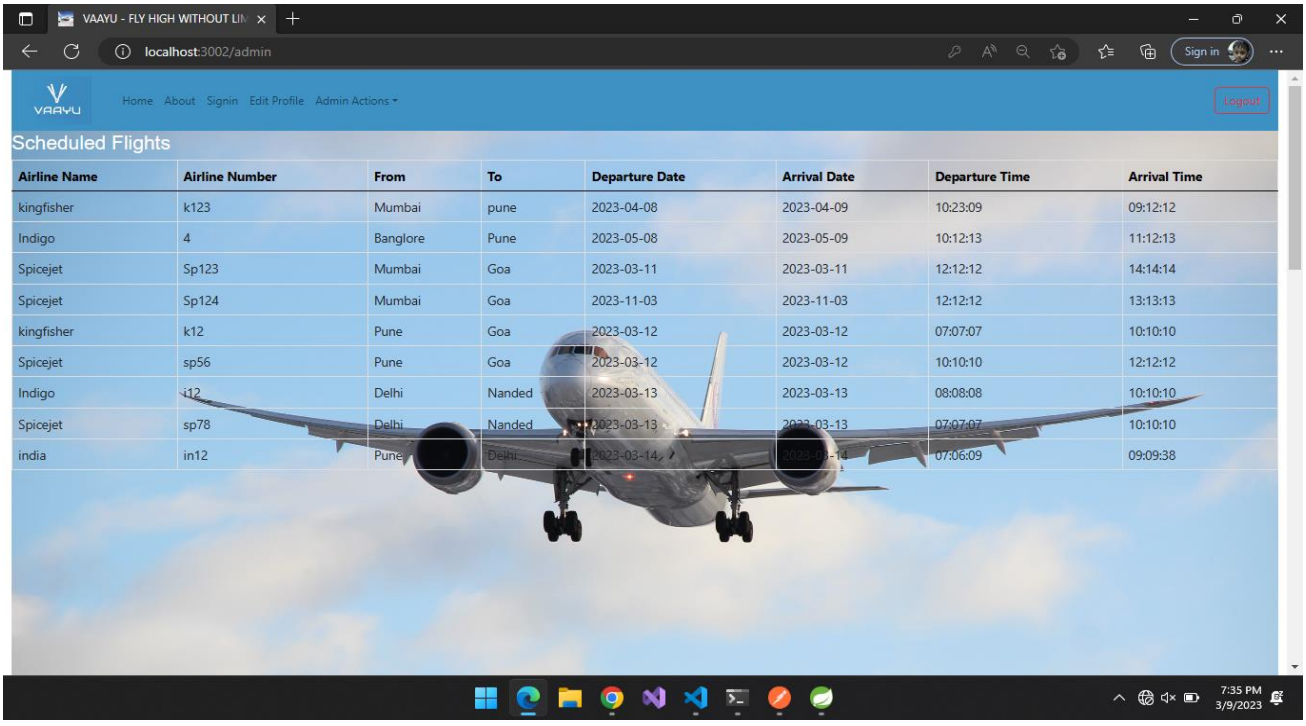
Revenue :



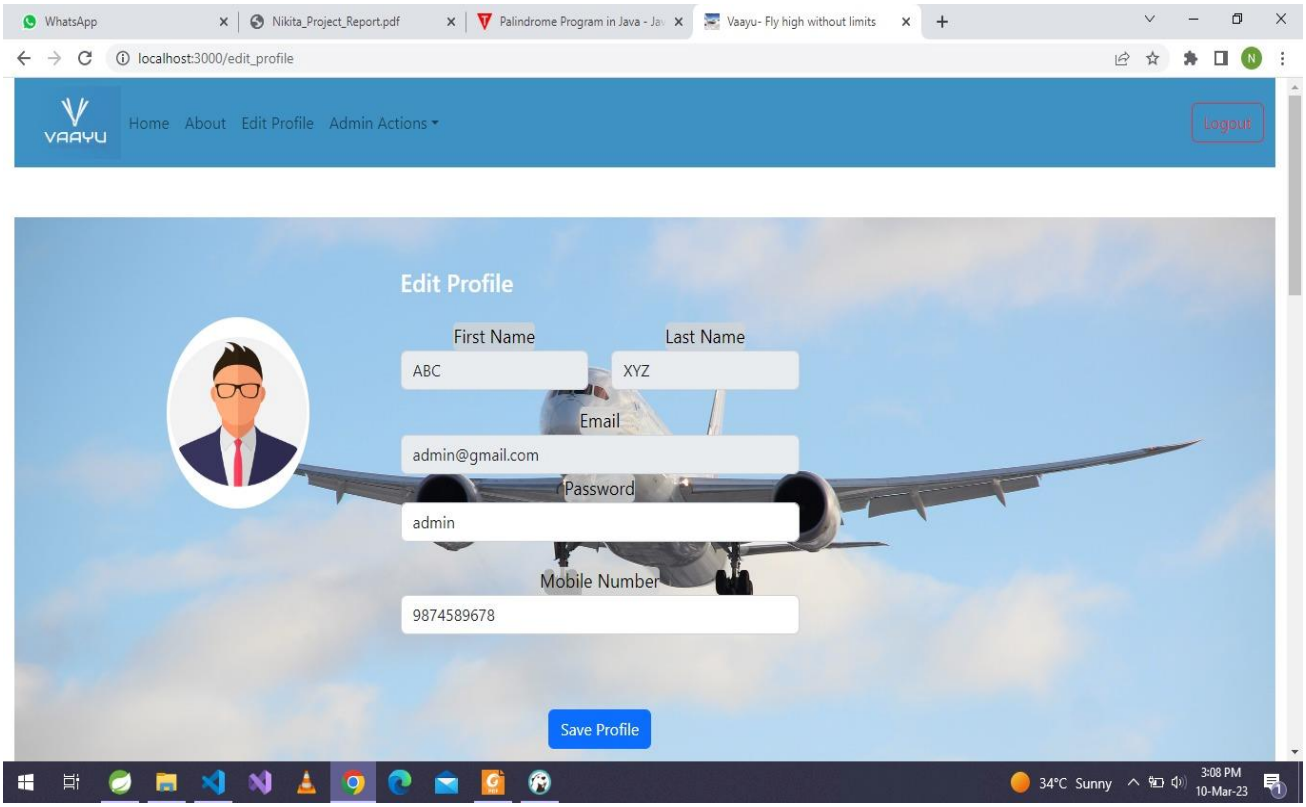
Show Feedbacks :



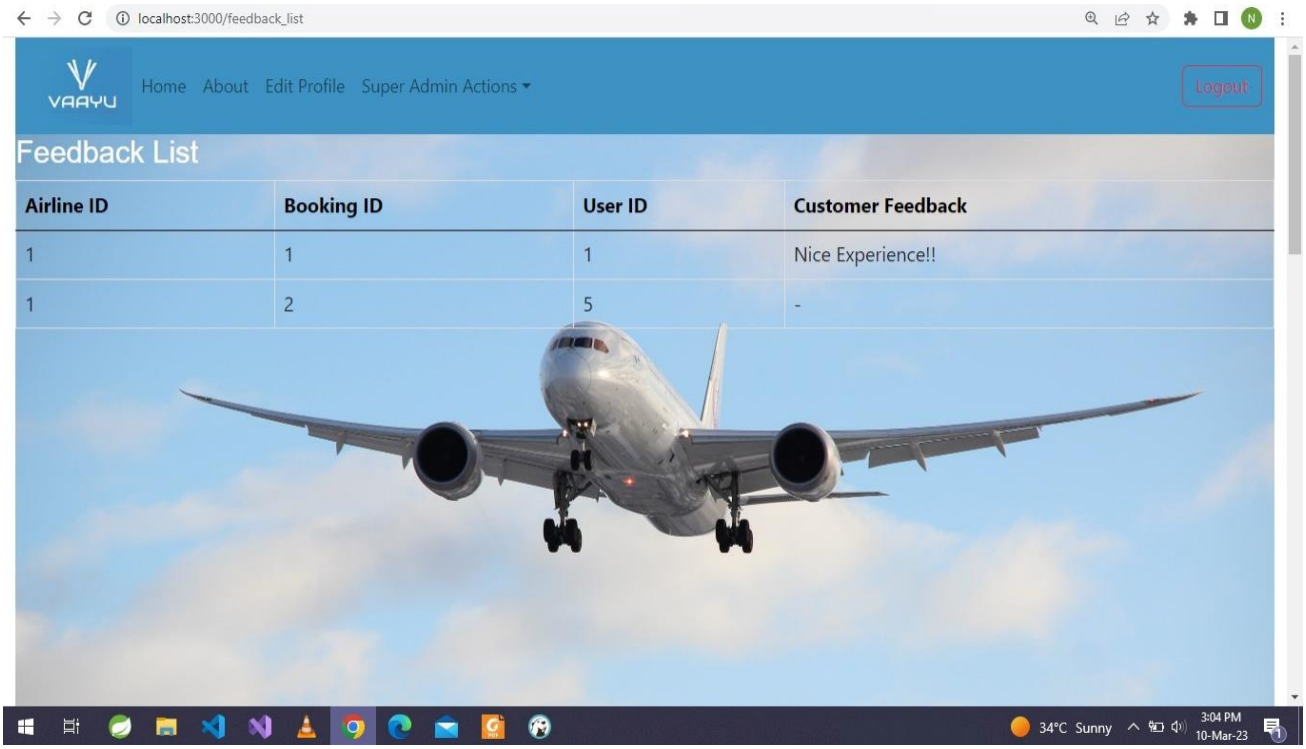
Scheduled Flights :



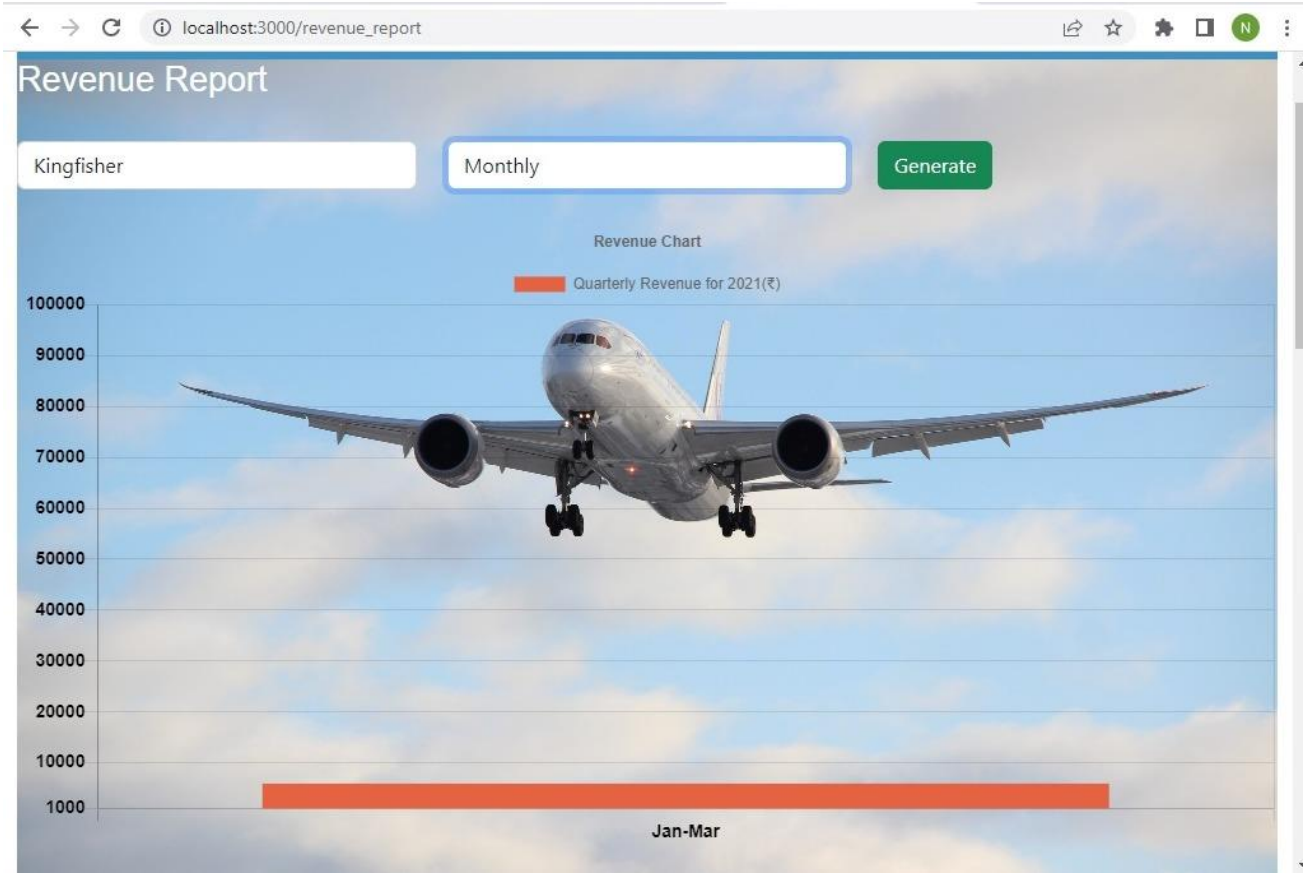
Edit Profile :



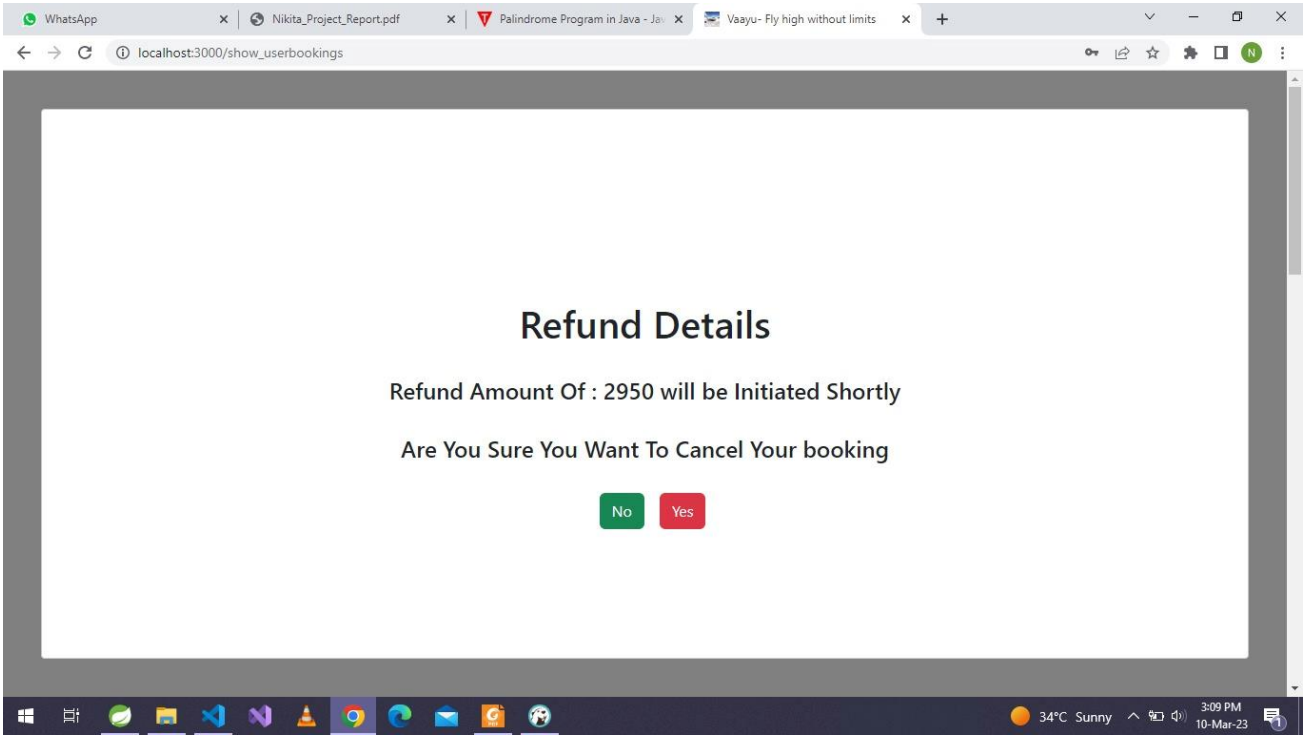
Feedback :



Revenue Report :



Refund Page :



8. REFERENCES

https://www.google.com

https://www.airindia.in

https://www.qatarairways.com

https://www.spicejet.com

https://www.wikipedia.org

CERTIFICATE

This is to certify that the project work under the title "*Airline Reservation System*" is done by Pratik Banarase, Nikita Deshmukh, Aradhya Gudadhe & Aishwarya Dhatrak in partial fulfillment of the requirement for award of Diploma in Advanced Computing Course.

Miss. Pooja Jaiswal
Project Mentor

Mr. Yogesh Kolhe
Course Co-ordinator

Date : 10/ 03/ 2023