ONLINE AIRLINES RESERVATION SYSTEM

A MINI PROJECT REPORT

SUBMITTED BY

ROSHINI VS 220701229

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

(AUTONOMOUS) THANDALAM

CHENNAI-602105

2023 - 24

TABLE OF CONTENTS

1. INTRODUCTION

1. INTRODUCTION :

The Online Airline Reservation System is designed to provide users with an efficient and user-friendly way to book airline tickets. The system aims to automate and simplify booking processes, offering an interface to search flights, select seats, make secure payments, and receive instant confirmations. This project covers key features of the airline reservation process, benefiting both the customers and the airline's management by streamlining operations.

2. OBJECTIVE:

The primary objectives of this project include:

- **Ease of Access**: Users can easily search for flights, view availability, and make reservations.
- **Automation**: Automate ticket booking, reducing manual work.
- **Security**: Implement a secure transaction process for payment handling.
- **Admin Management**: Allow the airline's administration to add, update, and monitor flight schedules.nh

3. SYSTEM REQUIREMENTS:

Software Requirements:

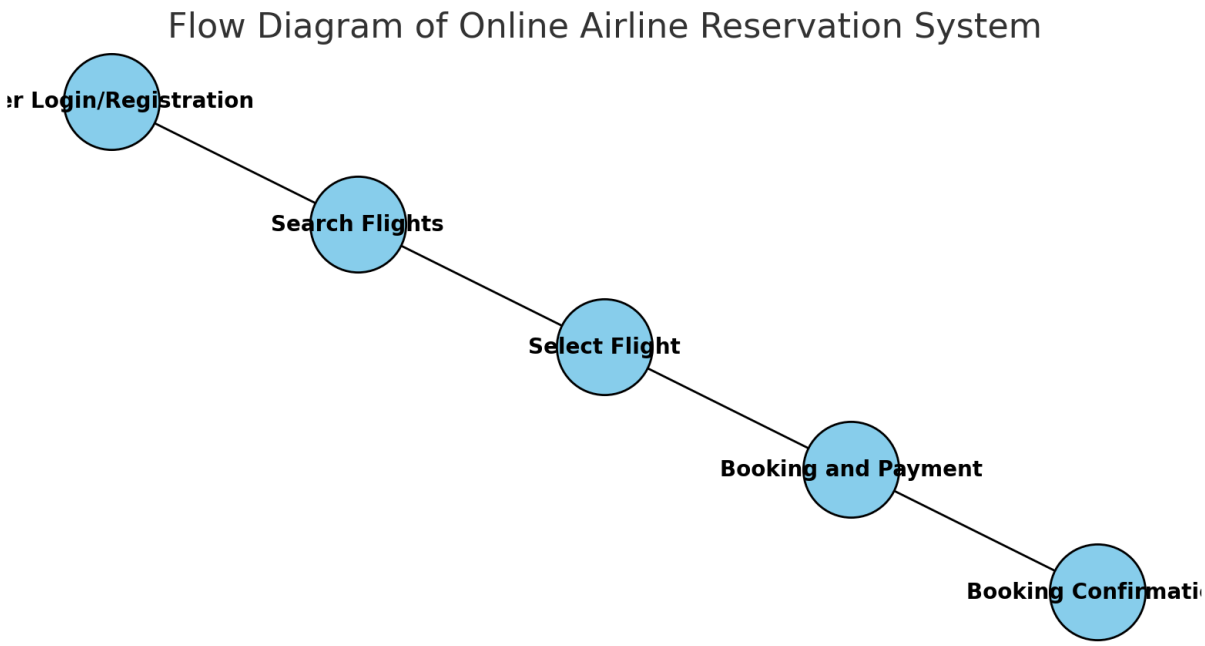
- **PHP**: For server-side scripting and handling backend processes.
- **MySQL**: Database to store information like user data, flights, and transactions.
- **Apache Server**: Required for hosting and testing the application locally.
- **HTML, CSS, JavaScript**: Frontend development for interactive user interfaces.

Hardware Requirements:

- **Standard PC** with enough memory and processor capacity to host a server environment.

4. SYSTEM DESIGN:

Flow Diagram:



A flow diagram represents the system process:

1. **User Login/Registration**: Allows users to log in or register.
2. **Flight Search**: Users search flights based on dates, destinations, and preferences.
3. **Flight Selection**: Users select flights and seats.
4. **Payment Processing**: Payment confirmation and transaction recording.
5. **Booking Confirmation**: System confirms and displays the booking.

Diagram Explanation :

Each step shows user interaction with the system, starting from login to booking confirmation. The flow diagram aids in visualizing the process from beginning to end.

5. MODULES AND FUNCTIONALITIES:

- **User Module**: Manages user registration, login, and viewing booking history.
- **Search and Booking Module**: Users search flights, filter by date and location, and view details.
- **Payment Gateway**: Simulates payment processing; future versions could integrate real payment platforms.
- **Admin Module**: Enables admins to add/remove flights and view booking statistics.
- **Other Features**: Options like seat selection, booking cancellation, and alerts for bookings.

2. BACKEND AND DATABASE DESIGN:

Database Structure :

- ***Users Table***: Stores user data (user_id, name, email, password).
- ***Flights Table***: Stores flight information (flight_id, airline, destination, date, time, seats).
- ***Bookings Table***: Stores booking details (booking_id, user_id, flight_id, booking date, seat number).
- ***Transactions Table***: Stores transaction details (transaction_id, booking_id, amount, status).

Users Table :

Column Name	Data Type	Description
user_id	INT	Primary key, unique user ID
name	VARCHAR	Full name of the user
email	VARCHAR	Email address of the user
password	VARCHAR	Hashed password for security

Flights Table:

Column Name	Data Type	Description
flight_id	INT	Primary key, unique flight ID
airline	VARCHAR	Name of the airline
destination	VARCHAR	Flight destination location
date	DATE	Date of the flight
time	TIME	Time of departure
seats	INT	Number of seats available

Bookings Table:

Column Name	Data Type	Description
booking_id	INT	Primary key, unique booking ID
user_id	INT	Foreign key, references Users table
flight_id	INT	Foreign key, references Flights table
booking_date	DATE	Date when the booking was made
seat_number	VARCHAR	Seat number assigned to the booking

Transactions Table :

Column Name	Data Type	Description
transaction_id	INT	Primary key, unique transaction ID
booking_id	INT	Foreign key, references Bookings table
amount	DECIMAL	Amount paid for the booking
status	VARCHAR	Status of the transaction (e.g., Paid)

Explanation:

Each table connects through primary keys, allowing data to be stored and retrieved efficiently. For example, a booking links to a user and flight, creating a relational structure.

7. FRONTEND DESIGN:

User Interface:

Home Page: Introduces the system and has fields for searching flights.

Booking Page: Allows seat selection and viewing detailed flight information.

Payment Page: Displays payment details and processes payment.

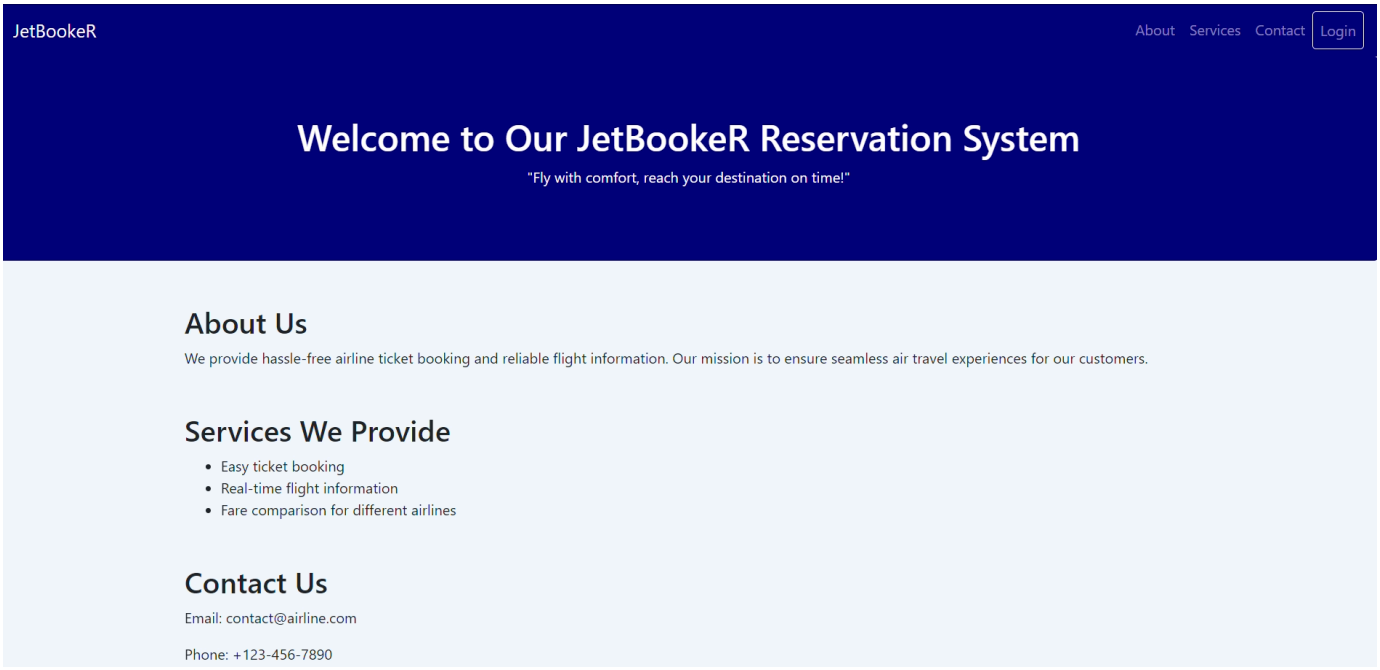
Confirmation Page: Shows booking confirmation and allows ticket download.

Technology Used:

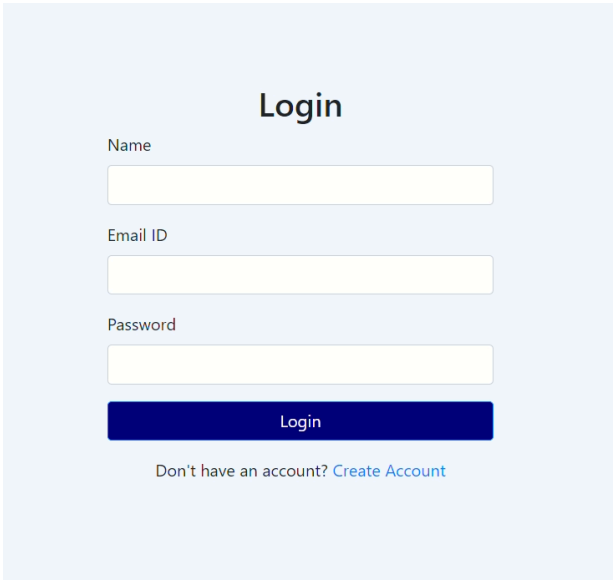
- **HTML and CSS**: For structure and styling.
- **JavaScript**: For validation and dynamic interactions (e.g., available seats).

RESULT:

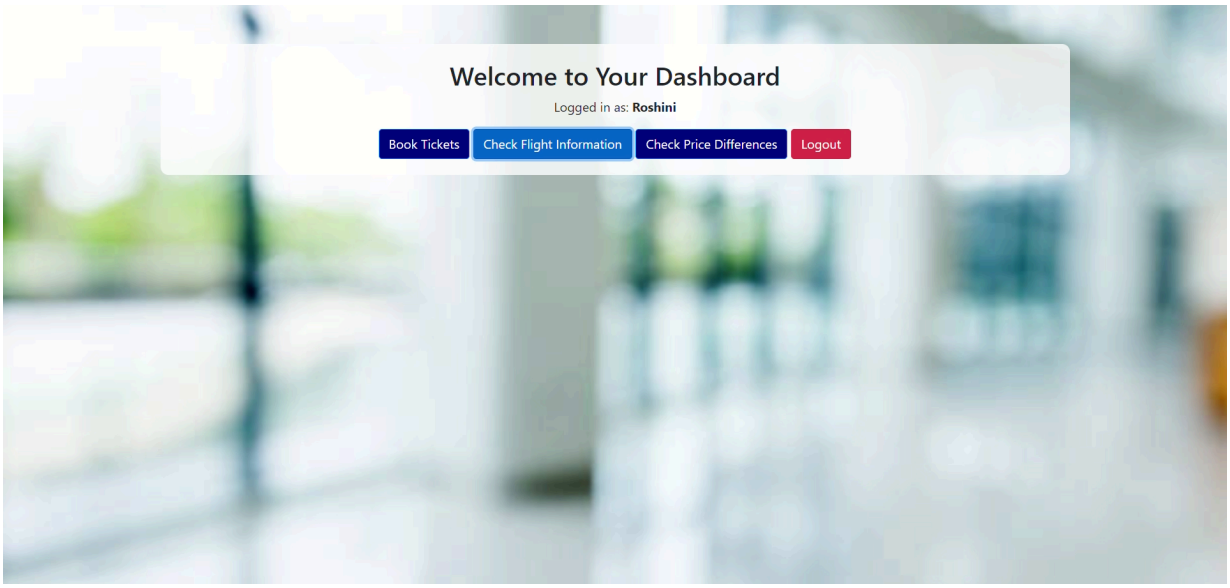
Front page :



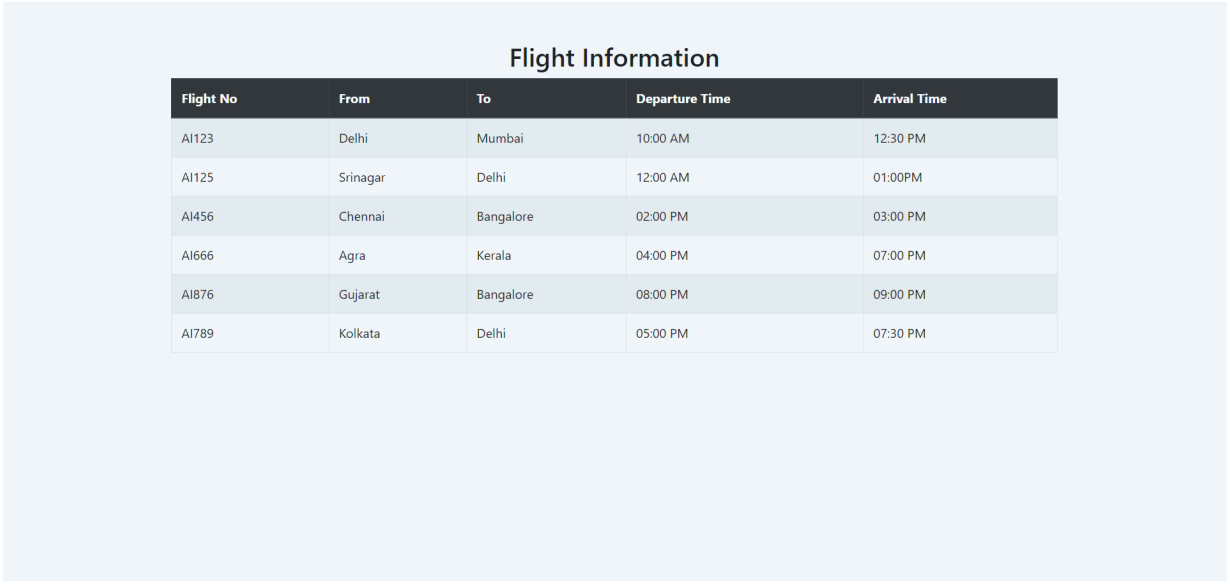
Login page :



Dashboard :



Flight Information :



Flight Price Information :

Flight Price Differences

Flight No	From	To	Class	Price
AI123	Delhi	Mumbai	Economy	₹4,500
AI666	Agra	Kerala	Economy	₹13,000
AI456	Chennai	Bangalore	Business	₹7,800
AI125	Srinagar	Delhi	Business	₹15,ooo
AI789	Kolkata	Delhi	First Class	₹12,500

Thank you for visiting our website!

Ticket Booking :

Book Your Ticket

From

Agra

To

Date of Travel

dd-mm-yyyy

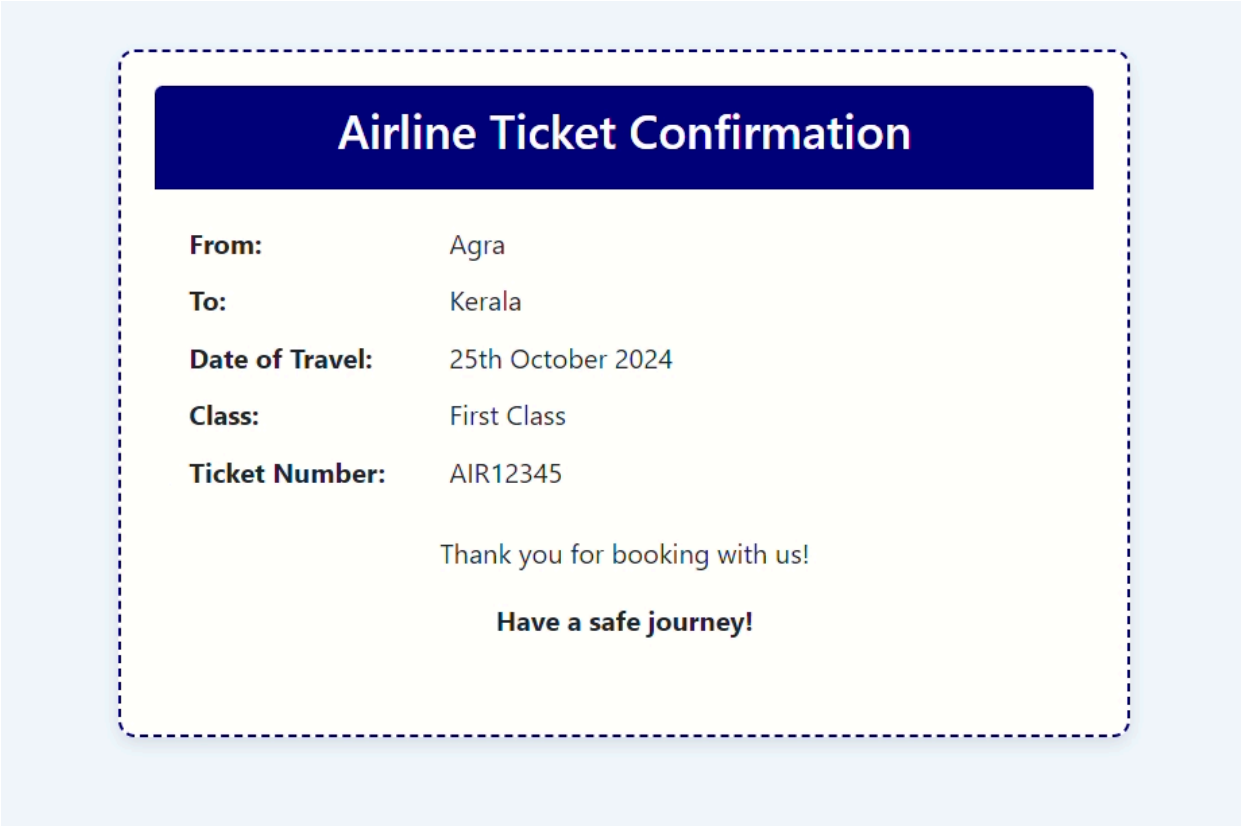
Number of Passengers

Class

Economy

Get Ticket

Ticket Confirmation :



8. TESTING AND IMPLEMENTATION:

Testing Strategy:

Unit Testing: Tested each module independently, such as the booking page and login.

Integration Testing: Checked how modules like search and payment work together.

User Acceptance Testing (UAT): Verified that the system meets user requirements and is intuitive.

Sample Test Cases:

1. **Flight Search Test**: Ensure accurate flight results.
2. **Booking Confirmation Test**: Verify all booking data appears correctly.
3. **Payment Test**: Simulate payment and verify transaction recording.

Implementation Details:

The system was deployed on a local server, testing functionality end-to-end, from the registration page to booking confirmation.

9. Challenges and Solutions

Challenges Faced:

- **Database Security**: Protecting user information.
- **User Authentication**: Managing secure sessions.
- **Payment Gateway**: Handling payment failures effectively.

Solutions:

- **Encryption**: Encrypt sensitive information in the database.
- **Session Management**: Secure sessions with user authentication.
- **Error Handling**: Robust error handling to manage payment processing issues.

10. Conclusion and Future Scope

Conclusion:

This project successfully implements a basic online airline reservation system, streamlining flight search, booking, and management for users and admins. The project meets the objectives of ease of use, secure transactions, and automated ticket management.

Future Scope:

Possible future improvements include:

- **Real-time Flight Status**: Integrating with APIs to provide flight tracking.
- **Multi-Currency Support**: Supporting international payments.

- **Personalized Features**: Offer personalized recommendations and loyalty programs.