FIight Booking App using MERN

1. Abstract

This project report presents a **Flight Booking Platform – Myflight**, designed to simplify and enhance the flight booking experience for users while providing airlines and administrators with tools to manage flights, bookings, and user interactions. The platform supports functionalities such as flight search, booking, cancellation, and user profile management. Developed with React, Bootstrap, Express, and Node, Myflight incorporates secure user access, dynamic seat management, and notifications using Nodemailer. It ensures a seamless experience through responsive design and efficient backend processing.

---

2. Introduction

- **Purpose of the Platform**: The goal of Myflight is to promote a seamless and user-friendly flight booking experience for passengers, airlines, and administrators by leveraging modern web technologies.
- **Problem Statement**: The platform addresses the need for an integrated solution where passengers can book and manage flights securely, airlines can oversee bookings and schedules, and administrators can maintain platform integrity and compliance.

---

3. Key Features of the Flight Booking Platform

1. User Roles and Access Control

- **Passenger**: Can search flights, book tickets, view booking history, and manage their profiles.
- **Airline Operator**: Can manage flight schedules, oversee bookings, and update flight details.
- **Admin**: Can monitor user and flight data, handle disputes, and remove outdated or non-compliant flights.

2. Passenger Functionalities

- **Flight Search and Booking**:
    - Passengers can search for flights based on origin, destination, date, and seat class.
    - Real-time seat availability is displayed for transparency.
- **Booking Management**:
    - View, modify, or cancel bookings.
    - Receive email confirmations for successful bookings and cancellations.
-
- **Profile Management**:
    - Passengers can update personal details such as name, email, and contact number.
- **Booking History**:
    - View a log of past bookings and payment details.

3. Airline Operator Functionalities

- **Flight Management**:
    - Add, edit, or remove flights, specifying details like origin, destination, timings, seat capacity, and pricing.
- **Booking Insights**:
    - View detailed data on passenger bookings for each flight.
- **Seat Map Management**:
    - Update seat availability dynamically based on cancellations or changes.

4. Admin Console

- **User Oversight**:
    - View detailed profiles of passengers and airline operators.
- **Flight Monitoring**:
    - Oversee all flights listed on the platform, remove non-compliant entries, and handle user disputes.
- **Audit Logs**:
    - Maintain a history of significant platform activities for compliance purposes.

---

4. Technical Architecture

1. Frontend

- **Framework**: React is used for developing an interactive and responsive user interface.
- **UI Components**: Bootstrap and custom CSS create a visually appealing and user-friendly interface.
- **Protected Routes**: JWT tokens ensure sensitive pages are securely accessible to authorized users.

2. Backend

- **Framework**: Node.js and Express power server-side operations and API integration.
- **Authentication and Security**:
    - JWT tokens for session management.
    - Password hashing using bcrypt for secure credential storage.
- **Dynamic Seat Allocation**:
    - Efficient algorithms ensure real-time seat allocation and conflict prevention.
- **Notifications**:
    - Nodemailer sends booking confirmations, cancellations, and reminders.

3. Database

- **Database Choice**: MongoDB Atlas stores user data, flight schedules, and booking information.
- **Data Structure**:
    - Models for passengers, airline operators, and admins, each with specific permissions and functionalities.
    - Flight schema to manage schedules, pricing, and seat availability.
    - Booking schema to track passenger details and seat assignments.

---

5. Workflow

1. Passenger Workflow

- Search Flights -> Select Flight -> Book Ticket -> Receive Confirmation Email -> View Booking History.

2. Airline Operator Workflow

- Login -> Add/Edit/Remove Flights -> Monitor Bookings -> Update Seat Maps.

3. Admin Workflow

- Login -> Monitor Flights and User Activity -> Remove Non-compliant Content -> View Audit Logs.

---

6. Security Measures

- **JWT Tokens**: Used to secure session management and protect sensitive routes.
- **Password Hashing**: All user passwords are securely hashed with bcrypt before storage.
- **Role-Based Access Control (RBAC)**: Ensures users only access functionalities permitted for their roles.

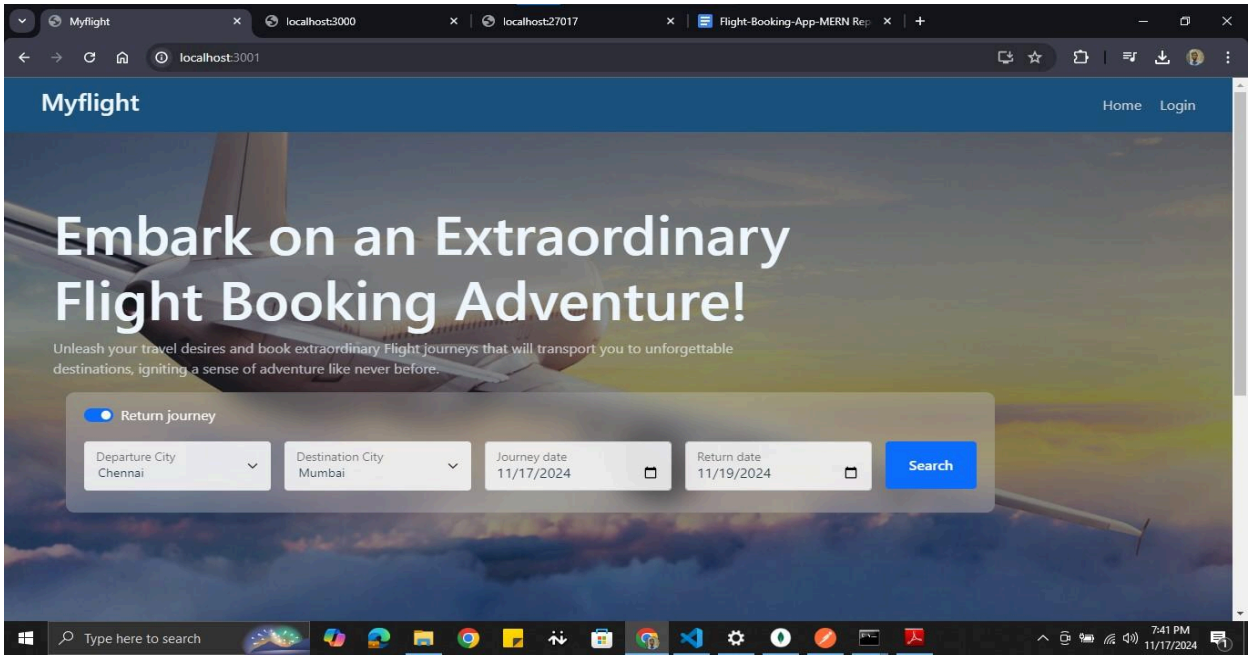
---

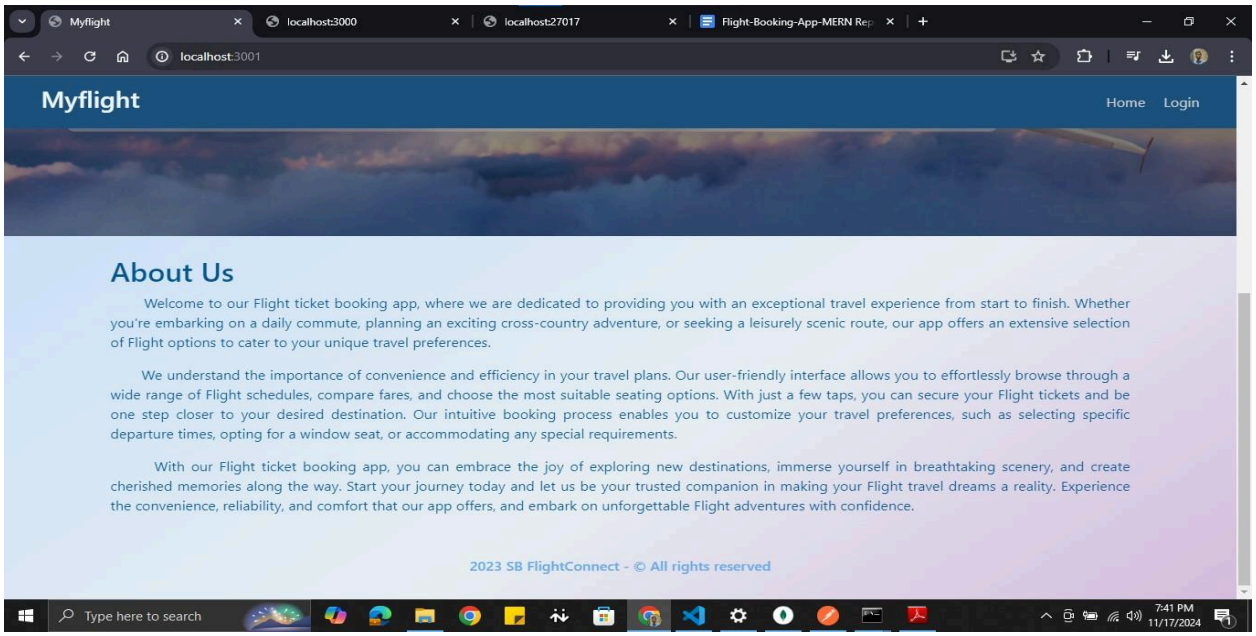
7. Future Scope

- Integration of a payment gateway for ticket purchases.
- Implementation of dynamic pricing based on demand and availability.
- Introduction of loyalty programs to reward frequent users.
- Real-time flight tracking with notifications for delays or cancellations.
- Enhanced analytics for operators and admins to monitor platform performance.

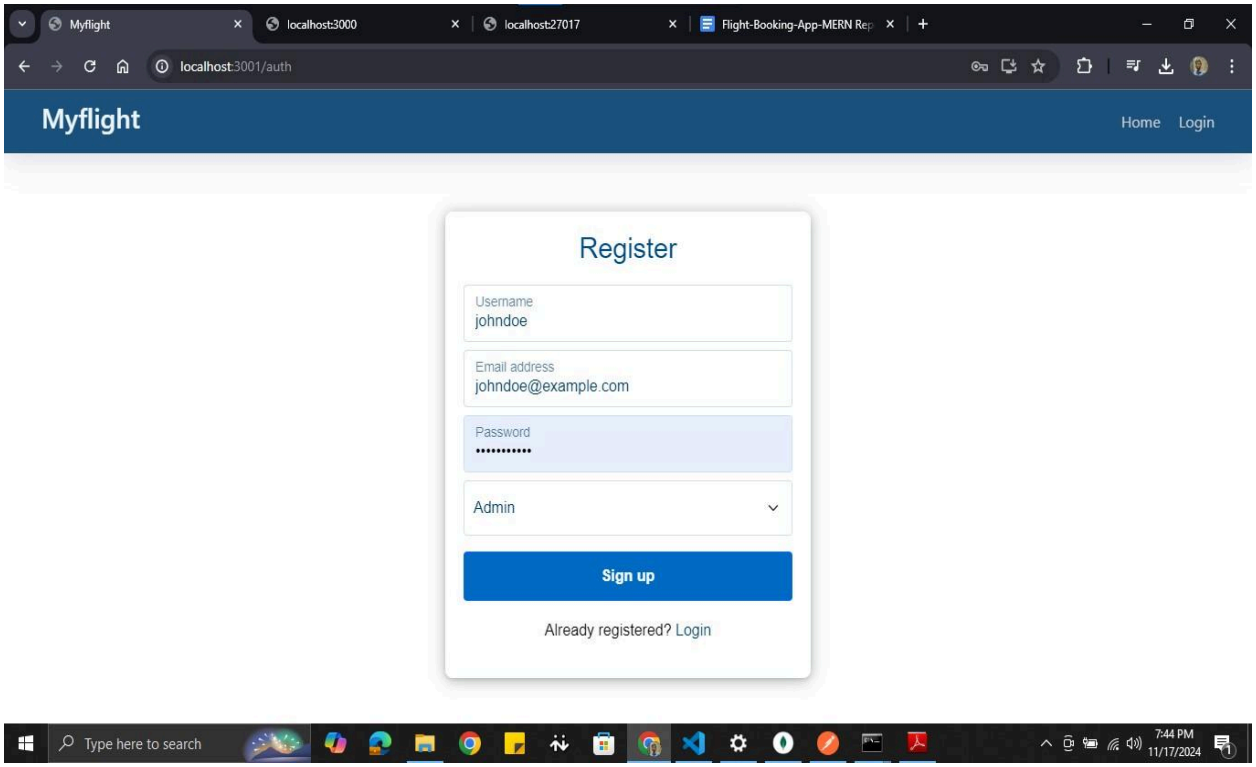
Output Screenshots

1. Landing Page

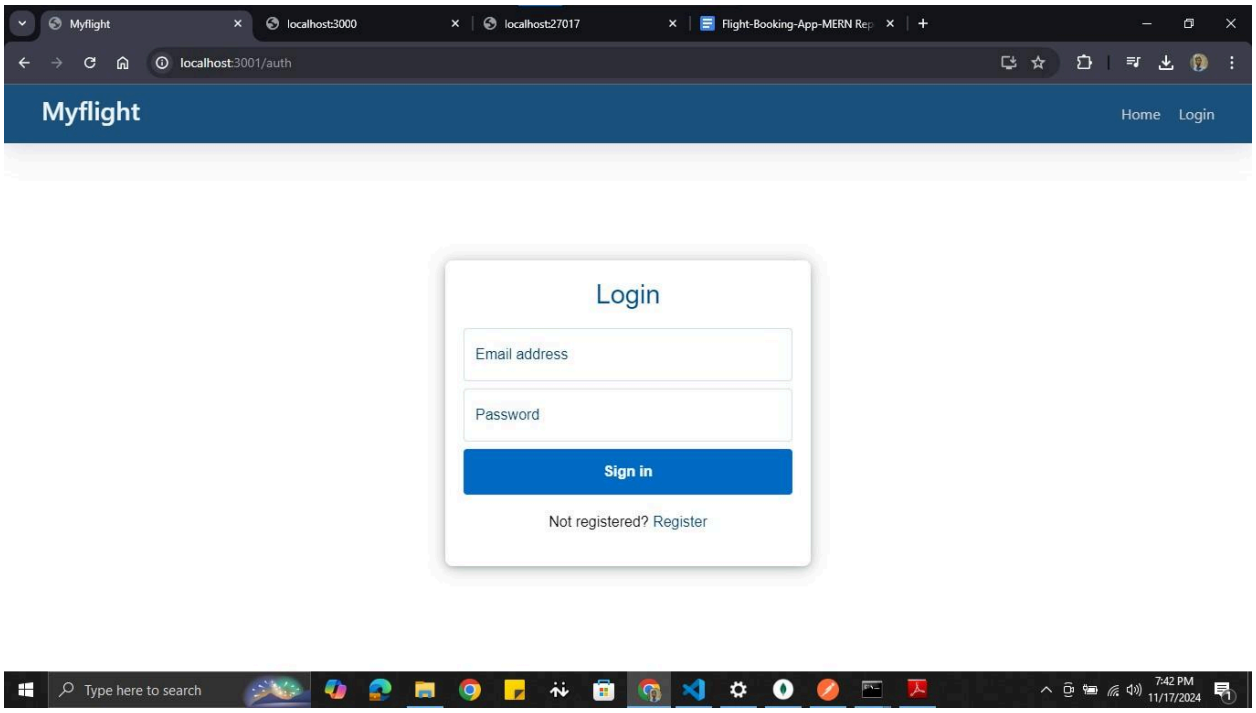




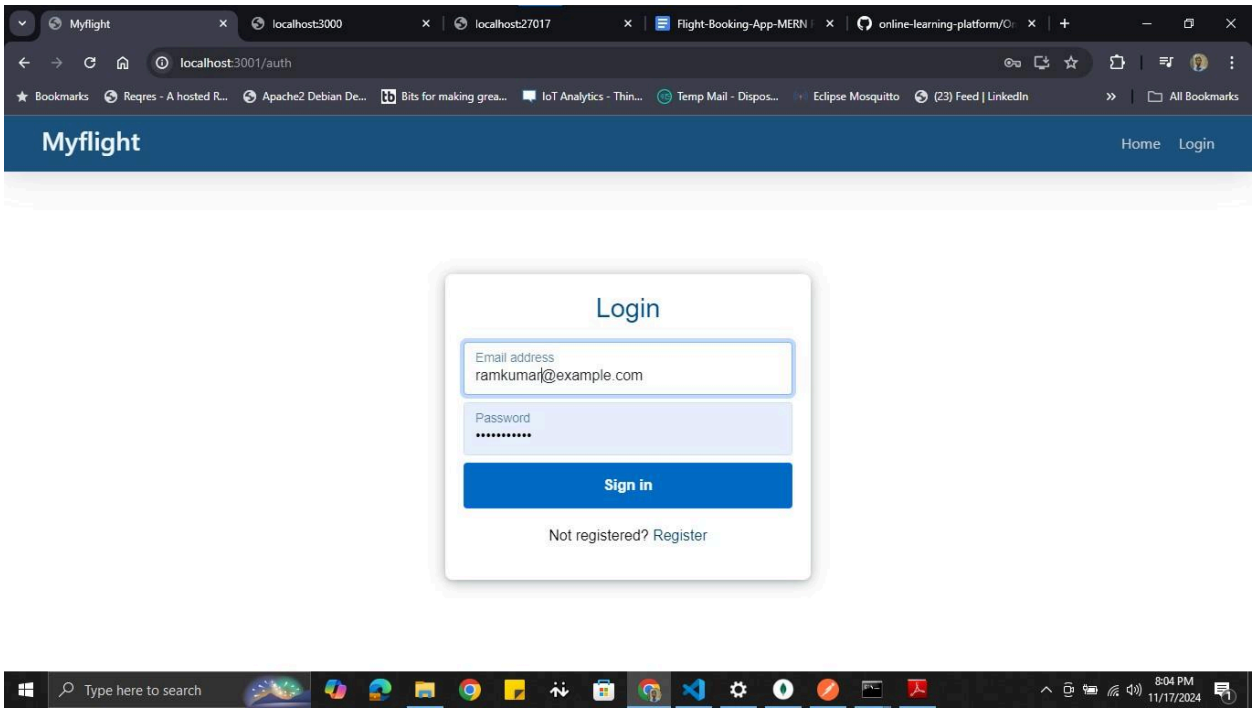
2. Register Page



3. Admin Login Page



4. Customer Login



5. Flight-Operator Register