FLIGHT BOOKING SYSTEM

PROJECT DOCUMENTATION

INTRODUCTION

Project Title: Quick Flights – Flight Booking System
Team Members: Individual Project (Single Person)

The Flight Booking System is a web-based application designed to simplify and automate the process of searching and booking flight tickets. The system provides a smooth and user-friendly interface for users to select their departure city, destination city, journey date, and book flights efficiently.

PROJECT OVERVIEW

The Quick Flights Flight Booking System aims to provide a seamless and hassle-free flight booking experience. It allows users to search for flights, manage bookings, and view booking details through an intuitive interface. This project is developed using the MERN stack to ensure scalability, performance, and security.

OBJECTIVES

- To provide an easy and fast flight booking platform
- To reduce manual booking processes
- To offer a clean and responsive user interface
- To securely manage user and booking data

KEY FEATURES

- Flight search using departure city, destination city, and journey date
- One-way and return journey option
- Secure user authentication (Login & Logout)
- Booking management and booking history
- Responsive and modern UI design
- Secure handling of user data

SYSTEM ARCHITECTURE

Frontend (React JS)

The frontend of the application is built using React JS. It includes components for the home page, flight search form, booking pages, and user dashboard. React Router is used for navigation, and React Hooks manage the application state.

Backend (Node.js + Express.js)

The backend is developed using Node.js and Express.js. It provides RESTful APIs for user authentication, flight search, and booking operations. Business logic and request handling are managed efficiently using middleware.

Database (MongoDB)

MongoDB is used to store user details, flight information, and booking records. Mongoose is used for schema design and database interaction.

TECHNOLOGIES USED

- Frontend: React JS, HTML, CSS, JavaScript
- Backend: Node.js, Express.js
- Database: MongoDB
- Tools: VS Code, Postman, Git

SETUP INSTRUCTIONS

Prerequisites

- Node.js
- MongoDB (Local or MongoDB Atlas)
- Git

Installation Steps

1. Clone the project repository
2. Install backend dependencies using npm install
3. Install frontend dependencies using npm install
4. Create a .env file and configure environment variables
5. Start backend and frontend servers

FOLDER STRUCTURE

Client (React Frontend)

```
client/
 ├── src/
```

```
|   ├── components/
|   ├── pages/
|   ├── styles/
|   ├── App.js
|   └── index.js
```

Server (Node.js + Express.js)

```
server/
 ├── controllers/
 ├── models/
 ├── routes/
 ├── middleware/
 ├── server.js
└── .env
```

API DOCUMENTATION

User APIs

- POST /api/auth/register – Register new user
- POST /api/auth/login – User login

Flight APIs

- GET /api/flights – Fetch available flights
- POST /api/flights/book – Book a flight

Booking APIs

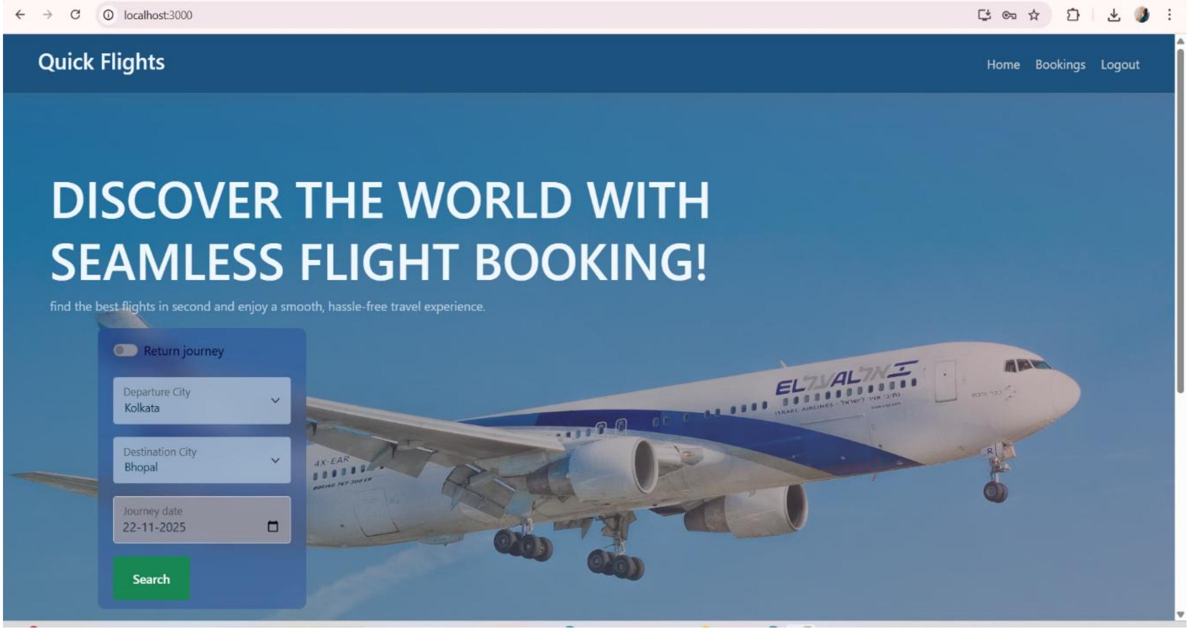
- GET /api/bookings/:userId – View user bookings

USER INTERFACE

Home Page

The home page allows users to search for flights by selecting the departure city, destination city, and journey date. A simple and clean UI ensures ease of use.

📷 Flight Booking UI Screenshot here



DISCOVER THE WORLD WITH SEAMLESS FLIGHT BOOKING!

find the best flights in second and enjoy a smooth, hassle-free travel experience.

Return journey

Departure City: Kolkata

Destination City: Bhopal

Journey date: 22-11-2025

TESTING

- Manual UI testing was performed to verify layout and navigation
- Backend APIs were tested using Postman
- Error handling and validation were checked

KNOWN ISSUES

- Limited flight data (demo purpose)
- No real-time flight availability
- Payment gateway not integrated

FUTURE ENHANCEMENTS

- Online payment gateway integration
- Real-time flight availability updates
- Email and SMS booking notifications
- Advanced search and filtering options
- Improved mobile responsiveness

CONCLUSION

The Quick Flights Flight Booking System successfully demonstrates the implementation of a modern web application using the MERN stack. The system provides an efficient and user-friendly platform for flight booking and can be further enhanced with real-time data and payment integrations.