

# **FLIGHT BOOKING APP DOCUMENTATION**

## **1. INTRODUCTION**

**Project Title:** Flight Booking App

**Team Members and Roles:**

**Abdulla Tippu Sultan I** : Project Manager

**Joshua Jared Jebaraj R** : Backend Developer

**Dinesh Kumar S** : Frontend Developer

**Giftson Jebaz P** : Database Administrator

### **PURPOSE OF THE PROJECT:**

To create a user-friendly web application that streamlines the flight booking process for travelers by providing secure and efficient booking capabilities.

## **2. PROJECT OVERVIEW**

**Objectives:**

Simplify the process of booking flight tickets.

Ensure secure and seamless transactions.

Provide a responsive design for compatibility across devices.

### **Features in Detail:**

#### **Flight Search:**

Users can search for flights based on destination, departure date, price range, and airline preferences.

#### **User Accounts:**

Create an account to save preferences.

Manage bookings with ease.

#### **Secure Payments:**

Integrate popular payment gateways for a smooth and safe payment process.

#### **Real-time Updates:**

Notifications for flight delays, cancellations, and price drops.

### **3. ARCHITECTURE**

## **System Design:**

**The application uses a three-tier architecture:**

- 1. Presentation Layer:** React.js frontend.
- 2. Business Logic Layer:** Node.js server with RESTful APIs.
- 3. Data Layer:** MongoDB for efficient storage and retrieval.

## **Technology Stack:**

**Frontend:** React.js, HTML, CSS, Bootstrap.

**Backend:** Node.js, Express.js.

**Database:** MongoDB (NoSQL).

## **4. SETUP INSTRUCTIONS**

### **Prerequisites:**

Install Node.js and MongoDB on your local machine.

Install npm for managing dependencies.

## **Steps to Install:**

1. Clone the repository from GitHub.
2. Install dependencies:  
Navigate to the root folder and run npm install.
3. Set up the environment variables in a .env file:
4. Start the application:

**Frontend:** Run npm start inside the client folder.

**Backend:** Run npm start inside the server folder.

## 5. Folder Structure

Client Directory:

Contains React components, CSS files, and static assets.

**Server Directory:**

API routes, middleware, and business logic.

**Database Schema:**

**User collection:** Stores user information, preferences, and booking history.

**Flight collection:** Details of available flights, including schedules and prices.

## 6. RUNNING THE APPLICATION

**Commands to Start Locally:**

**Frontend:** npm start in the client/ directory.

**Backend:** npm start in the server/ directory.

## 7. API DOCUMENTATION

**Endpoints:**

1. GET /api/flights

Fetch a list of available flights.

2. POST /api/bookings

Book a flight.

3. GET /api/users/{id}

Fetch user details and booking history.

### **Authentication:**

Users authenticate using JSON Web Tokens (JWT).

## **8. AUTHENTICATION**

### **User Registration:**

New users can sign up with their email and password.

### **Login Mechanism:**

Secure login with encrypted passwords (bcrypt).

JWT tokens generated on successful authentication.

## **9. USER INTERFACE**

### **Homepage:**

Displays search bar and featured flight deals.

### **Search Results Page:**

Displays flight options with filters for price, airlines, and layovers.

### **Booking Page:**

Form to input passenger details and payment information.

## **10. TESTING**

### **Types of Testing:**

#### **1. Unit Testing:**

Testing individual components such as APIs and UI forms.

#### **2. Integration Testing:**

Ensure the backend integrates seamlessly with the frontend.

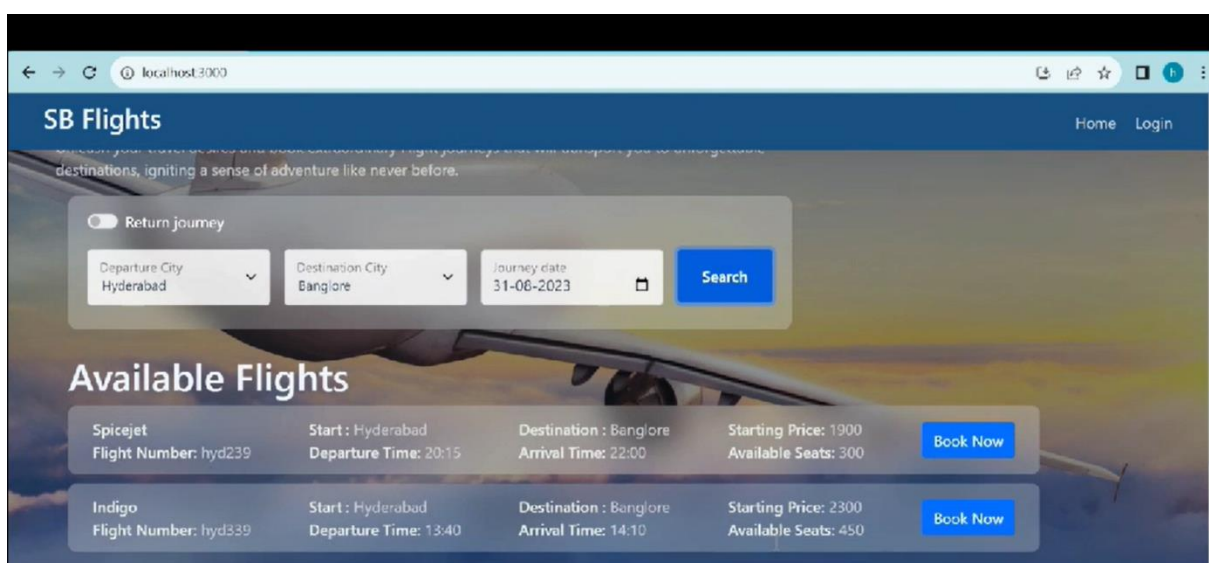
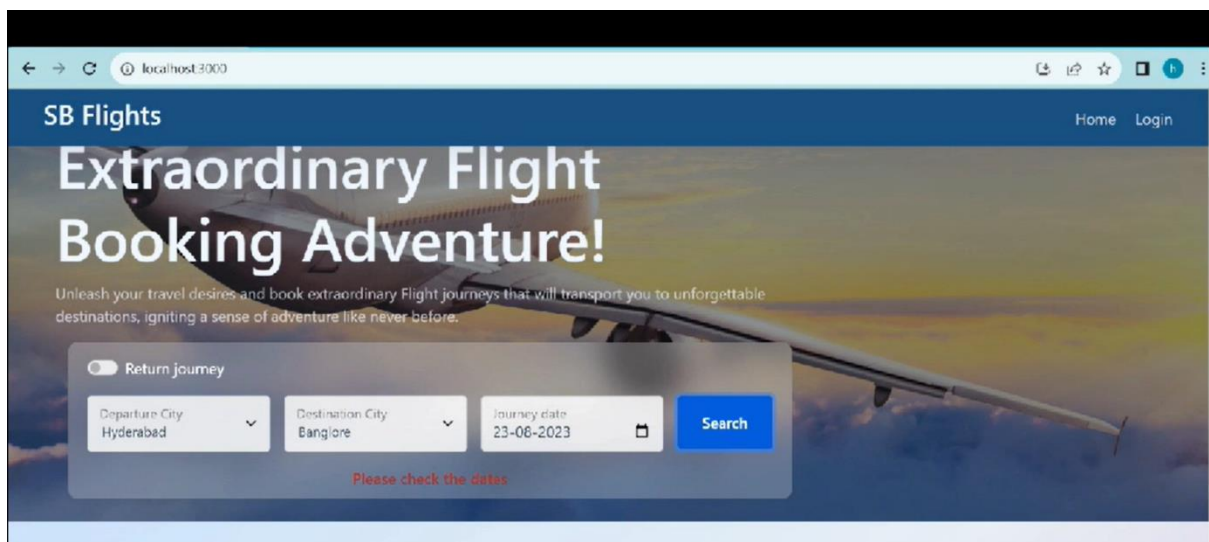
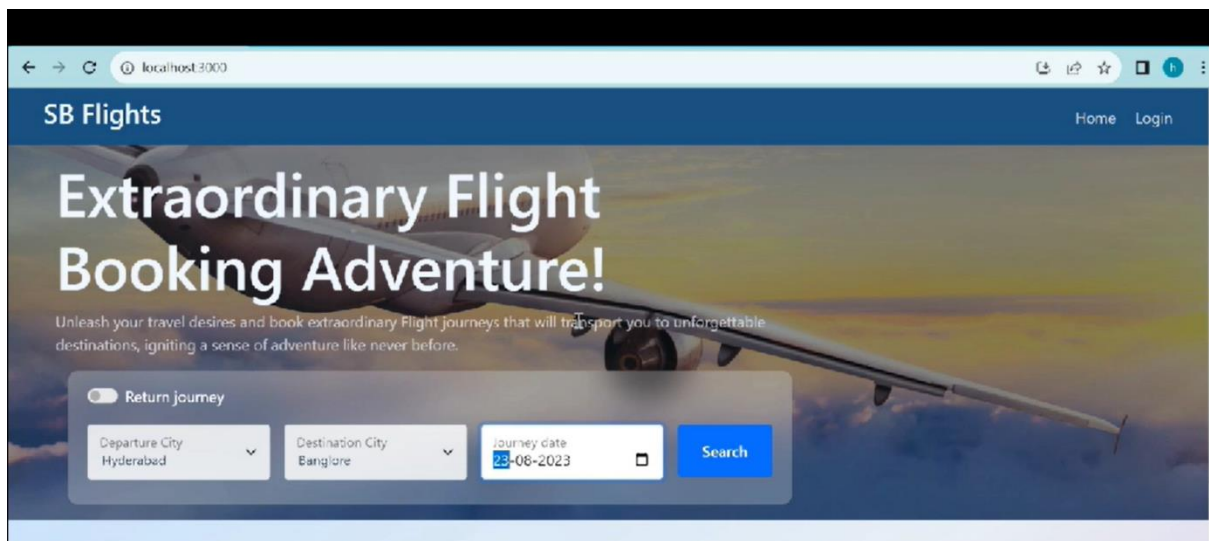
#### **3. End-to-End Testing:**

Use Cypress for automated end-to-end testing.

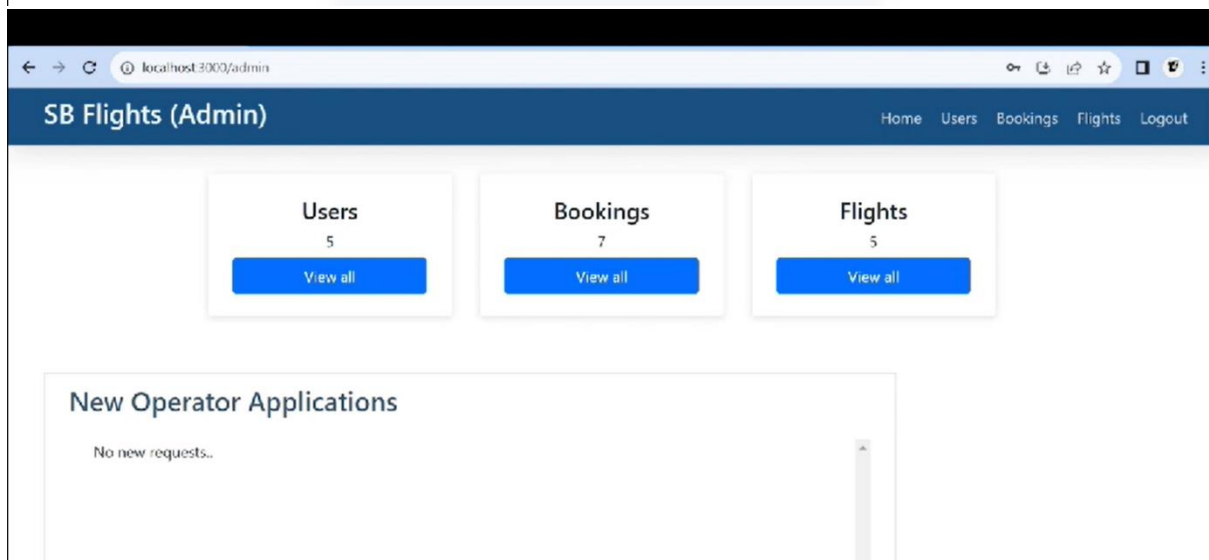
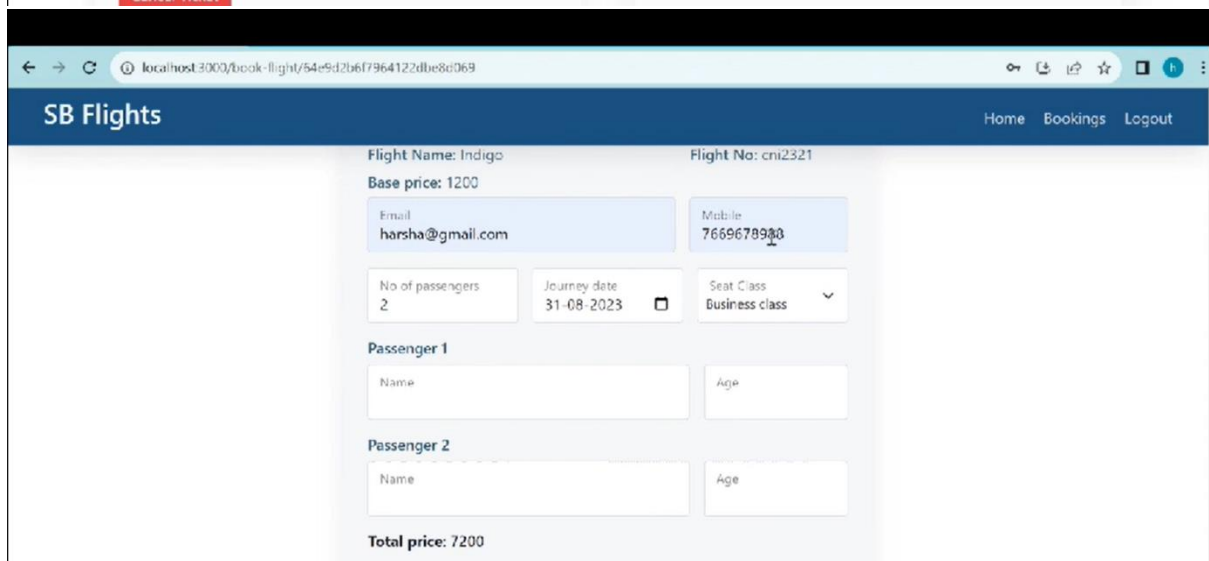
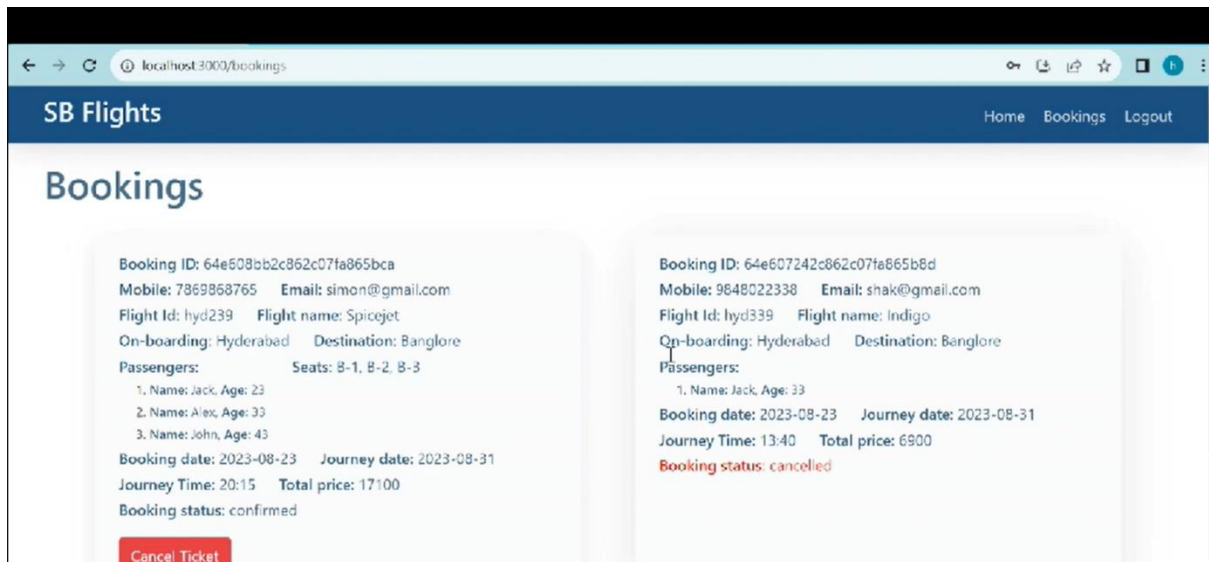
## **11. SCREENSHOTS AND DEMO**

**DEMO LINK :** [https://drive.google.com/file/d/1F-w6Mc1gz17Z9Gu\\_nbuWlAbQB-GT0TpF/view?usp=drivesdk](https://drive.google.com/file/d/1F-w6Mc1gz17Z9Gu_nbuWlAbQB-GT0TpF/view?usp=drivesdk)

## SCREENSHOTS :







← → ↻ localhost:3000/auth

SB Flights

Home

Login

Email address

Password

Sign In

Not registered? Register

← → ↻ localhost:3000/flight-admin

SB Flights (Operator)

Home Bookings Flights Add Flight Logout

Approval Required!!

Your application is under processing. It needs an approval from the administrator. Kindly please be patience!!

SB Flights (Admin)

Home Users Bookings Flights Logout

Users

5

View all

Bookings

7

View all

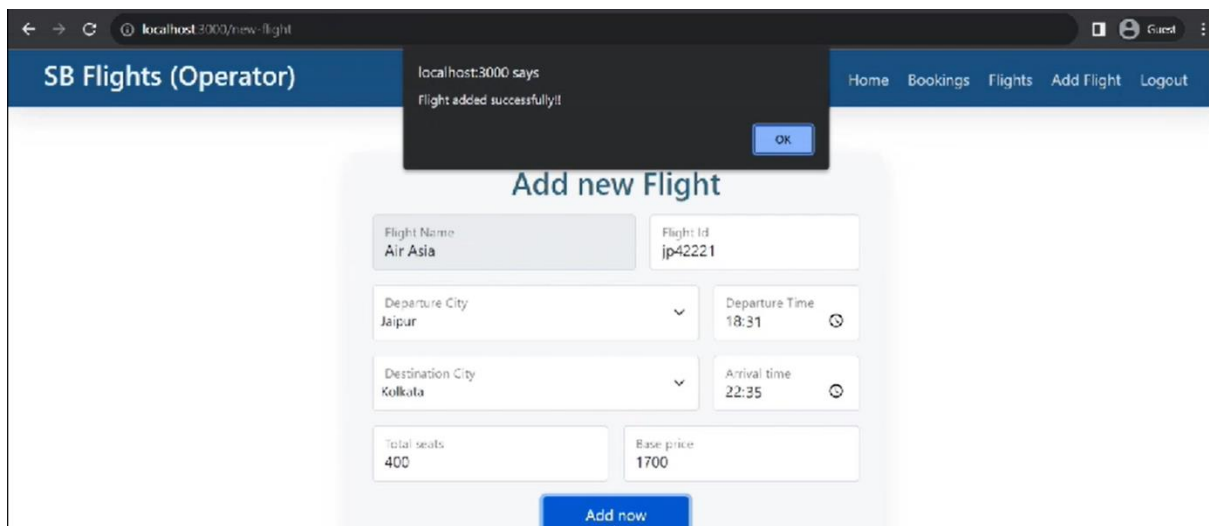
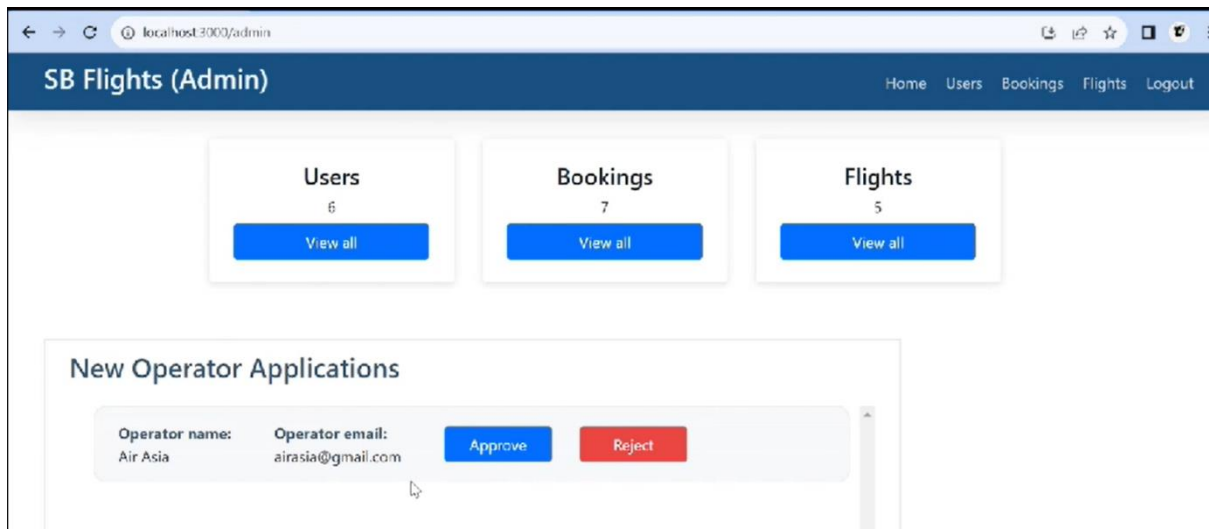
Flights

5

View all

New Operator Applications

No new requests...



## 12. KNOWN ISSUES

### API Limitation:

Currently, flight data is not sourced in real time from airlines.

### Mobile Optimization:

Minor inconsistencies in some older devices.

## **13. FUTURE ENHANCEMENTS**

### **Real-time Flight Data:**

Integration with airline APIs for live updates.

### **Multi-language Support:**

Translate the interface to support global users.

### **AI-Powered Recommendations:**

Use machine learning to recommend flights based on user behavior.

## **15. CONCLUSION**

The Flight Booking App is designed to simplify the travel booking process, enhance user experience, and provide a reliable platform for flight management. With its robust architecture and potential for future enhancements, it aims to be a leading tool in the travel industry.