

Project Report: FlightFinder – Navigating Your Air Travel Options

1. INTRODUCTION

1.1 Project Overview

FlightFinder is a full-stack web application designed to streamline and enhance the flight booking experience. Built with the MERN stack (MongoDB, Express.js, React.js, and Node.js), it allows users to search, book, and manage flight tickets conveniently, while providing admin functionalities for managing flights and bookings.

1.2 Purpose

- Simplify the flight booking process
 - Provide real-time search and filter options
 - Ensure secure and smooth user authentication
 - Enable admin-level control over flights and bookings
 - Create an intuitive and user-friendly UI
-

2. IDEATION PHASE

2.1 Problem Statement

Travelers often face inconvenience with outdated or slow flight booking platforms. Issues such as lack of real-time updates, complex UIs, and limited admin control hinder the user experience. FlightFinder solves this with a responsive, secure, and scalable solution.

2.2 Empathy Map Canvas

- **Think & Feel:** Need for quick, secure, and reliable booking experience
- **See:** Multiple flight options, prices, filters
- **Hear:** Recommendations from friends or digital platforms
- **Say & Do:** Book based on timings, prices, preferences
- **Pain:** Booking errors, complex interface, poor support

- **Gain:** Easy booking, trusted system, efficient management

2.3 Brainstorming

- Real-time seat availability
 - Admin dashboard with analytics
 - Role-based access (User/Admin/Airline Operator)
 - Intuitive UI/UX
-

3. REQUIREMENT ANALYSIS

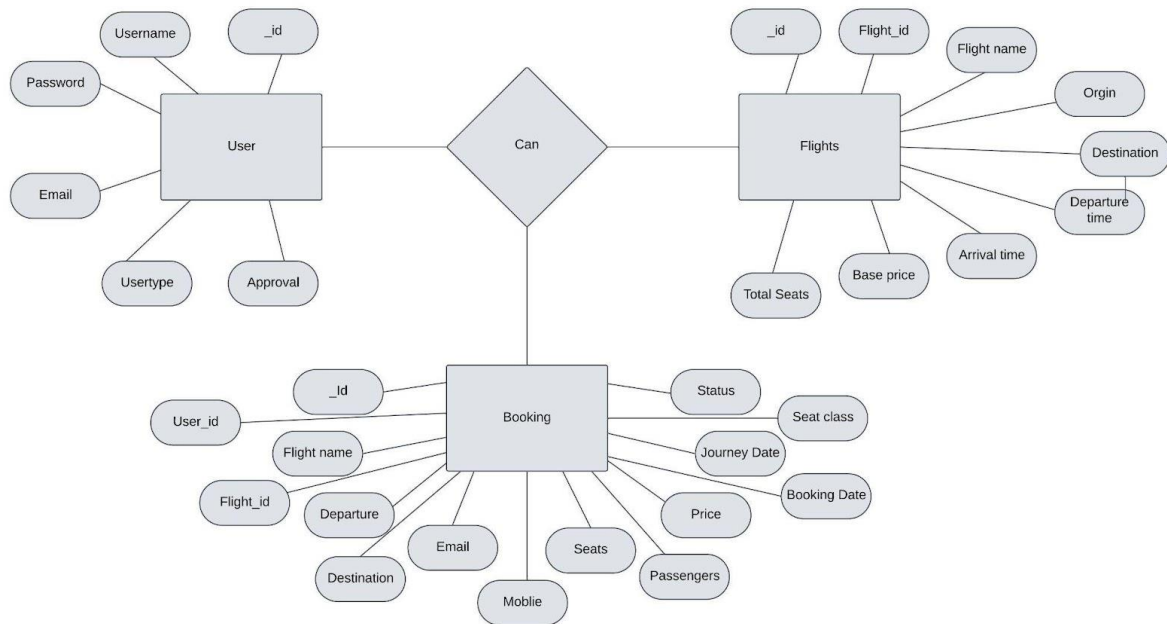
3.1 Customer Journey Map

1. User visits the site
2. Registers/Login
3. Searches flights
4. Applies filters and books tickets
5. Views bookings in dashboard
6. Cancels/Edits booking if needed

3.2 Solution Requirement

- Secure authentication
- CRUD operations on bookings and flights
- Admin approval of flights
- User-friendly interface

3.3 Data Flow Diagram



1. User actions (search/book) trigger frontend events
2. React calls backend APIs using Axios
3. Express routes handle business logic
4. MongoDB stores and retrieves data

3.4 Technology Stack

- **Frontend:** React.js, Bootstrap
- **Backend:** Node.js, Express.js
- **Database:** MongoDB (Atlas)
- **Tools:** GitHub, VS Code, Postman

4. PROJECT DESIGN

4.1 Problem Solution Fit

FlightFinder bridges the gap between user needs and current offerings by delivering real-time availability, intuitive UI, and secure booking.

4.2 Proposed Solution

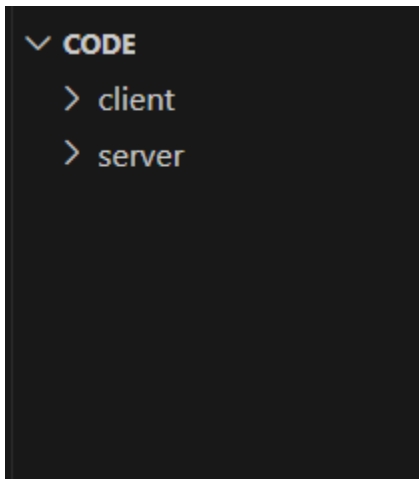
A web application where users can:

- Search and book flights

- View/manage bookings
Admins can:
- Add/manage flights
- Approve bookings
- View analytics

4.3 Solution Architecture

- **Frontend:** Components for search, booking, dashboard



- **Backend:** REST APIs with secure routes
- **Database:** User, Flight, and Booking schemas

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- Week 1: Requirement gathering and setup
- Week 2: Backend API and MongoDB integration
- Week 3: Frontend development
- Week 4: Testing and UI improvements
- Week 5: Deployment and documentation

6. FUNCTIONAL AND PERFORMANCE TESTING

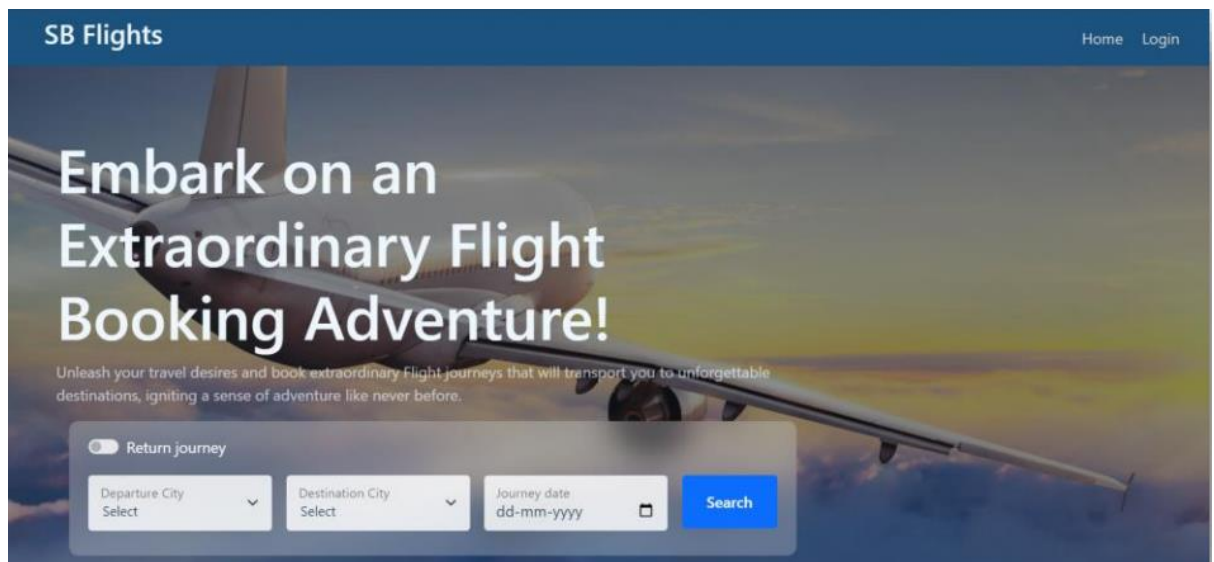
6.1 Performance Testing

- Load tested APIs for flight search and booking
 - Validated MongoDB query performance
 - Frontend load time under 3 seconds
 - Secure login under concurrent sessions
-

7. RESULTS

7.1 Output Screenshots

- Landing Page
- Authentication (Login/Register)
- Flight Search with Filters
- Seat Booking
- Admin Dashboard (Manage Flights/Bookings)
- User Dashboard



Login

[Sign in](#)[Not registered? Register](#)

Bookings

Booking ID: 64ec8c3c4622709484005484
Mobile: 7669678988 Email: harsha@gmail.com
Flight Id: cni2321 Flight name: Indigo
On-boarding: Chennai Destination: Bangalore
Passengers: Seats: B-1, B-2
1. Name: Alex, Age: 44
2. Name: Snyder, Age: 55
Booking date: 2023-08-28 Journey date: 2023-08-31
Journey Time: 18:40 Total price: 7200
Booking status: confirmed

[Cancel Ticket](#)

Booking ID: 64e608bb2c862c07fa865bca
Mobile: 7869868765 Email: simon@gmail.com
Flight Id: hyd239 Flight name: Spicejet
On-boarding: Hyderabad Destination: Bangalore
Passengers:
1. Name: Jack, Age: 23
2. Name: Alex, Age: 33
3. Name: John, Age: 43
Booking date: 2023-08-23 Journey date: 2023-08-31
Journey Time: 20:15 Total price: 17100
Booking status: cancelled

Booking ID: 64e607242c862c07fa865b8d

Booking ID: 64e604fa1b698133e1a38d19

SB Flights (Admin)

HomeUsersBookingsFlightsLogout

Users

6

View all

Bookings

7

View all

Flights

6

View all

New Operator Applications

No new requests...

SB Flights (Admin)

HomeUsersBookingsFlightsLogout

All Users

Userid 64e5fcb298f1c5aa0a36c2a7

Username hola

Email hola@gmail.com

Userid 64e9d2e0f7964122dbe8d098

Username alex

Email alex@gmail.com

Flight Operators

Id 64e8ce302bb50798fe630779

Flight Name spicejet

Email spicejet@gmail.com

Id 64e8d11154e48a90d1c0f26b

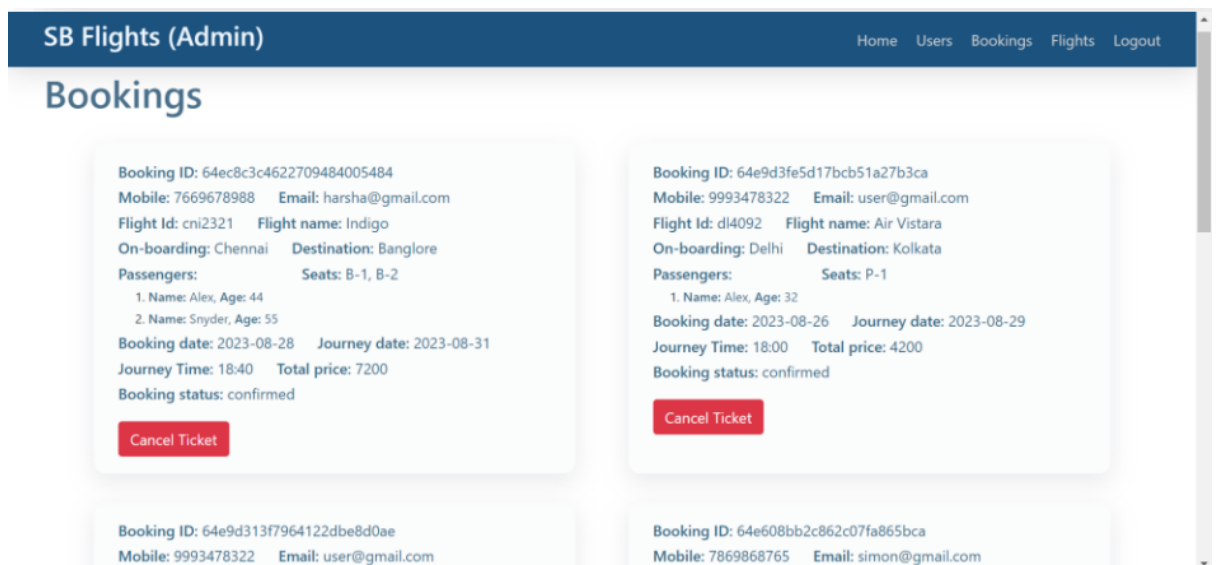
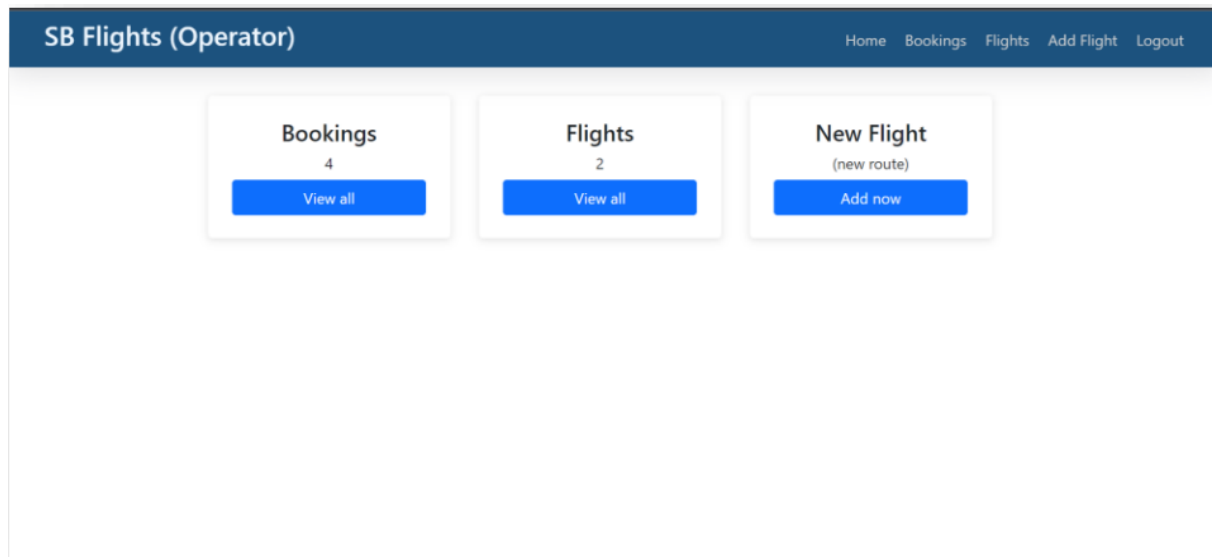
Flight Name Indigo

Email indigo@gmail.com

Id 64e9d38e5d17bcb51a27b36a

Flight Name Air Vistara

Email vistara@gmail.com



Demo Link: <https://drive.google.com/file/d/18yTPal-BYVDYp5VynWfCx5IEPzi0QHBk/view>

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Real-time booking system
- Easy-to-use interface
- Scalable backend

- Secure login and booking flow

Disadvantages:

- No payment gateway (only simulated)
 - No integration with live airline APIs
-

9. CONCLUSION

FlightFinder successfully demonstrates how a modern web application can simplify complex travel booking processes. It highlights full-stack development best practices, from secure authentication to real-time CRUD operations with a clean UI.

10. FUTURE SCOPE

- Integration with live airline APIs
 - Real payment gateway integration
 - Mobile app version
 - Advanced analytics for admins
 - User notifications (email/SMS)
-

11. APPENDIX

Source Code:

[Github Repo](#)

Dataset Link:

N/A (Data handled dynamically via MongoDB)

Demo Link:

<https://drive.google.com/file/d/18yTPal-BYVDYp5VynWfCx5IEPZi0QHBk/view>