# Final Report: FlightFinder (SB Flights)

---

## 1. INTRODUCTION

### 1.1 Project Overview

SB Flights (FlightFinder) is a full-stack flight booking platform designed to make air travel planning easy, transparent, and accessible. The web application allows users to search, compare, and book flights with real-time information and offers an admin dashboard for managing flights and bookings.

### 1.2 Purpose

The purpose of this project is to provide a seamless experience for both users and flight service administrators through a single platform, reducing booking complexity and enhancing overall user satisfaction.

---

## 2. IDEATION PHASE

### 2.1 Problem Statement

Travelers often face challenges such as unclear pricing, overwhelming options, lack of personalization, and difficulty managing bookings. Existing platforms either clutter the interface or do not offer sufficient backend control to flight administrators.

### 2.2 Empathy Map Canvas

- **Says**: “I want to book quickly without hidden charges.” □ **Thinks**: “Will I get the best price? Is this platform secure?” □ **Does**: Compares multiple sites, checks reviews, delays booking.
- **Feels**: Overwhelmed, uncertain, time-constrained.

### 2.3 Brainstorming

Ideas generated included real-time filtering, price transparency, admin-managed dashboards, email confirmations, booking history, and support for multi-city routes.

---

# 3. REQUIREMENT ANALYSIS

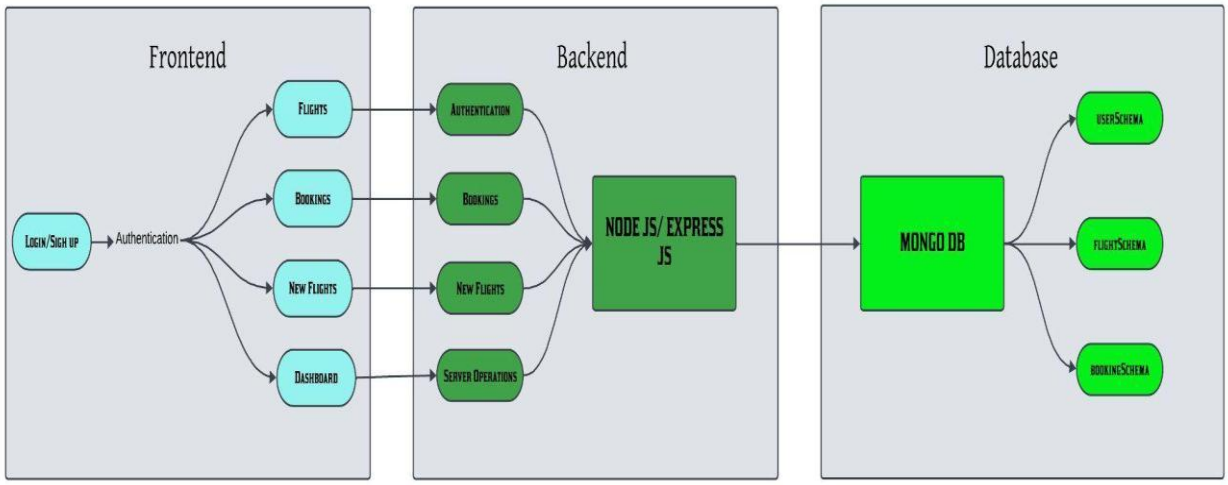
## 3.1 Customer Journey Map

1. User visits website
2. Registers/Login
3. Searches for flights
4. Filters and selects flights
5. Books ticket
6. Receives confirmation
7. Views past bookings
8. Admin manages flights/bookings

## 3.2 Solution Requirement

- User-friendly UI
- Secure authentication
- Booking and management dashboard
- Admin and operator views

## 3.3 Data Flow Diagram




## 3.4 Technology Stack

- **Frontend**: React.js
- **Backend**: Node.js with Express
- **Database**: MongoDB
- **Version Control**: Git/GitHub

---

# 4. PROJECT DESIGN

## 4.1 Problem Solution Fit

FlightFinder directly addresses user pain points through real-time search, transparency, and admin control.

## 4.2 Proposed Solution

An interactive platform where users can search and book flights, while admins can manage listings and monitor bookings.

## 4.3 Solution Architecture

- **Frontend**: React Components
- **Backend**: RESTful APIs
- **Database**: MongoDB with Mongoose models
- **Authentication**: JWT-based login system for users/admins

---

# 5. PROJECT PLANNING & SCHEDULING

## 5.1 Project Planning

- Week 1: Setup and research
- Week 2-3: Backend development
- Week 4-5: Frontend integration
- Week 6: Testing and documentation

---

# 6. FUNCTIONAL AND PERFORMANCE TESTING

## 6.1 Performance Testing

 All major functions tested using Postman and browser tools. 
Search and booking API respond within 2.5s under load.

---

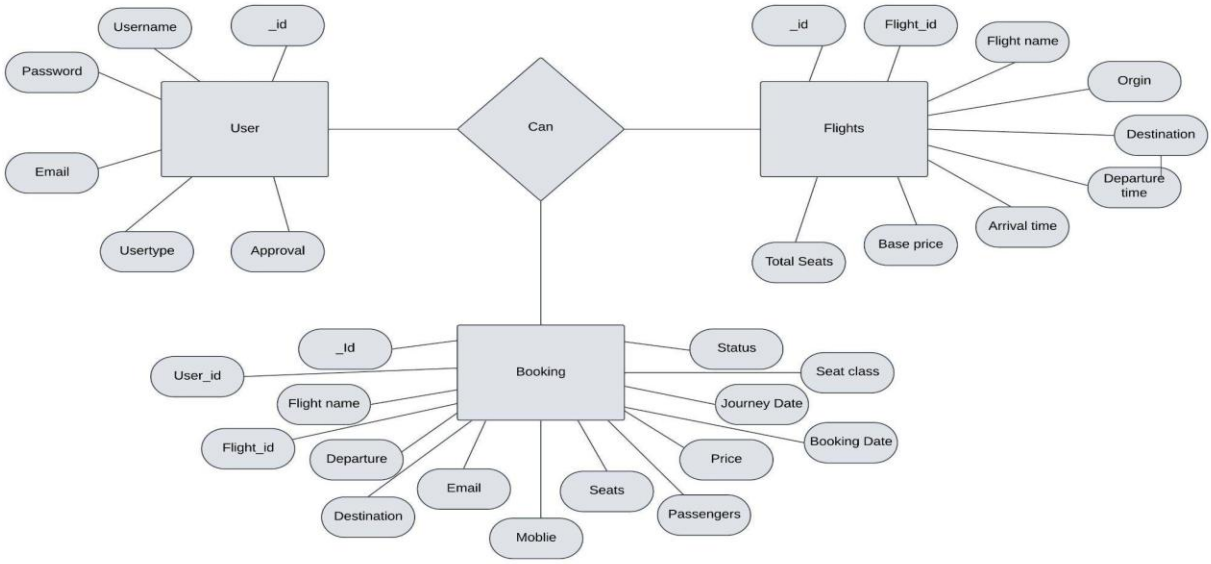
# 7. RESULTS

## 7.1 Output Screenshots

- Landing Page
- Registration/Login
- Flight Listings
- Booking Modal

- Admin Dashboard (Refer to demo link for complete visuals)

## 7.2 ER Diagram




The flight booking ER-diagram represents the entities and relationships involved in a flight booking system. It illustrates how users, bookings, flights, passengers, and payments are interconnected. Here is a breakdown of the entities and their relationships:

**USER:** Represents the individuals or entities who book flights. A customer can place multiple bookings and make multiple payments.

**BOOKING:** Represents a specific flight booking made by a customer. A booking includes a particular flight details and passenger information. A customer can have multiple bookings.

**FLIGHT**: Represents a flight that is available for booking. Here, the details of flight will be provided and the users can book them as much as the available seats.

**ADMIN**: Admin is responsible for all the backend activities. Admin manages all the bookings, adds new flights,etc.

---

# 8. ADVANTAGES & DISADVANTAGES

## Advantages

- Real-time flight search
- Admin and operator roles

- Simple and clean UI

### Disadvantages

- Limited multi-language support
- Lacks integrated payment gateway

---

## 9. CONCLUSION

FlightFinder successfully delivers a streamlined solution for booking and managing flights. It caters to both users and admins, ensuring a smooth and secure experience across the platform.

---

## 10. FUTURE SCOPE

- Integration with payment gateway (e.g., Razorpay)
- Mobile application
- Predictive pricing algorithm
- Support for international flights

---

## 11. APPENDIX

- **GitHub Repo**: https://github.com/Ashik1029/FlightFinder-Navigating-Your-AirTravel-Options
- **Demo Video**: https://drive.google.com/drive/folders/1HxqGxNo84_xKrTyA9Ecu3HWaa0QllRZI
  **Source Code**: Included in GitHub repository