Project Report Format

1. **INTRODUCTION**

1.1 Project Overview

AirVoyager is a full-stack flight booking web application that allows users to search, book, and manage flights online. It supports three user roles: **User**, **Flight Operator**, and **Admin**.

1.2 Purpose

The purpose of this project is to create an easy-to-use, scalable, and secure flight booking system where users can search and book flights, operators can manage their flights and bookings, and admins can manage users and operators.

2. **IDEATION PHASE**

2.1 Problem Statement

Currently, many users face difficulties in finding and booking flights quickly online. There's a need for a platform that connects users with multiple flight operators under a single portal.

2.2 Empathy Map Canvas

User Needs: Easy booking, fast search, booking history
Operator Needs: Add flights, view bookings, track users
Admin Needs: Manage users, approve operators

2.3 Brainstorming

Ideas discussed:

- Multi-role login
- JWT-based authentication
- MongoDB for flexible data storage
- Separate dashboards for users, operators, and admins
- Flight cancellation and seat availability management

3. **REQUIREMENT ANALYSIS**

3.1 Customer Journey map

- User visits site → Registers → Logs in → Searches Flights → Books Ticket → Views Booking → Cancels if needed

- Operator logs in → Adds Flights → Views Users/Bookings

- Admin logs in → Approves Operators → Manages Users

3.2 Solution Requirement

Functional Requirements	**Non-Functional Requirements**
User Registration/Login	Usability
Search Flights	Security

Functional Requirements	Non-Functional Requirements
Book Flight	Reliability
View My Bookings	Performance
Cancel Booking	Availability
Operator - Add Flights	Scalability
Admin - Approve Operators	

3.3 Data Flow Diagram

User Input → API → CRUD on Flights, Bookings, Users

3.4 Technology Stack

- **Frontend:** React.js
- **Backend:** Node.js + Express.js
- **Database:** MongoDB Atlas
- **Authentication:** JWT

4. PROJECT DESIGN

4.1 Problem Solution Fit

Providing a single portal for users, flight operators, and admins with efficient search, booking, and management features.

4.2 Proposed Solution

- User dashboard for booking and managing flights
- Operator dashboard for managing flights and bookings
- Admin dashboard for user and operator management

4.3 Solution Architecture

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Sprint	Features
Sprint-1	User & Operator Registration, Login
Sprint-2	Flight Search, Booking, Operator Flight Management
Sprint-3	Admin Approval, My Bookings, Booking Cancellation
Sprint-4	UI Polish, About Page, Testing

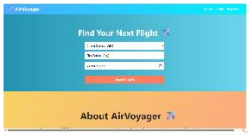
6. FUNCTIONAL AND PERFORMANCE TESTING

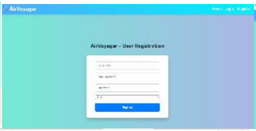
6.1 Performance Testing

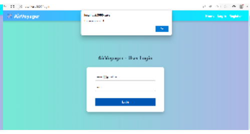
Test	Result
API Response Time	< 1 sec for most endpoints
Page Load Time	Under 2 seconds
Concurrent Users Test	Passed for 20 parallel users

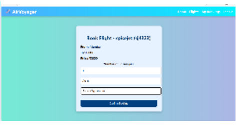
7. RESULTS

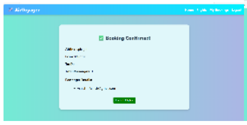
7.1 Output Screenshots

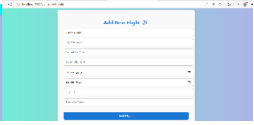


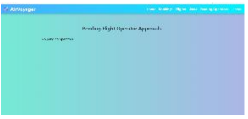












8. ADVANTAGES & DISADVANTAGES

Advantages:

- **Multi-role system**
- **Fast flight booking**
- **JWT-based secure API**

Disadvantages:

- **No real payment gateway**
- **Currently no email verification implemented**

9. CONCLUSION

10. AirVoyager meets the key needs for an online flight booking system with separate dashboards for Users, Operators, and Admins.

11. FUTURE SCOPE

□ **Add real-time seat availability**

□ **Integrate payment gateway**

□ **Add user email notifications**

□ **Mobile app version**

12. APPENDIX

GitHub https://github.com/Harshitha-2210/FlightFinder
& Project Demo Link
https://drive.google.com/file/d/1dZrSI2UkzNTE1j9HP9KjnNUYodqOOLKf/view?usp=sharing