

PROJECT REPORT FORMAT

1. Introduction

1.1 Project Overview

FlightFinder is a web application designed to help users search, compare, and book flights across multiple airlines in one seamless platform.

1.2 Purpose

The project aims to simplify air travel planning by providing a user-friendly interface, real-time flight data, and secure booking capabilities.

2. Ideation Phase

2.1 Problem Statement

Current flight booking methods are often fragmented. Users must visit multiple airline websites, manually compare prices, and manage bookings in separate portals.

2.2 Empathy Map Canvas

Think & Feel: Users want peace of mind—easy search, simple booking flow.

See & Hear: Overwhelmed by airline deals and pop-ups.

Say & Do: “I wish this were easier to compare at once.”

Pain: Switching tabs, hidden fees, slow checkout.

Gain: Instant comparisons, upfront cost visibility, one-click booking.

2.3 Brainstorming

Aggregate flight data via API

Filters: date, passenger count, class, price

Authentication for personalized features

Contact form for support

Multi-screen UI for user journey

3. Requirement Analysis

3.1 Customer Journey Map

Users: Signup → Login → Search flights → Compare → Enter details → Confirm booking → Contact support

3.2 Solution Requirement

User authentication

Real-time flight search and filter

Payment integration

Contact form

Responsive design

3.3 Data Flow Diagram

Client ↔ Server: endpoints for login, search, booking, support

DB: Users, Flights, Bookings, Messages

3.4 Technology Stack

Frontend: React.js

Backend: Node.js/Express

Database: MongoDB or PostgreSQL

APIs: Third-party flight data, authentication

Payment Gateway: Stripe/PayPal

4. Project Design

4.1 Problem-Solution Fit

Aggregates flight info from multiple airlines and streamlines booking—solving fragmentation and usability issues.

4.2 Proposed Solution

A React frontend with user authentication, searching/filtering flights, adding passengers, secure payment, and support contact.

4.3 Solution Architecture

Frontend (React) ↔ Backend (Express, API endpoints) ↔ Database

Third-party flight data & payment gateway integrations

5. Project Planning & Scheduling

5.1 Project Planning

1. Week 1: Requirements gathering, UI mockups

2. Week 2–3: Authentication, flight search & filtering

3. Week 4: Passenger form, contact form

4. Week 5: Payment integration + booking flow

5. Week 6: Testing, bug fixes, deployment

6. Functional and Performance Testing

6.1 Performance Testing

Verified search speed and pagination

Payment flow tested with test cards

Form validation and error handling

7. Results

7.1 Output Screenshots (in order)

1. Homepage (logged in)

Buttons for Round-Trip/One-Way/Multi-City, date pickers, passenger count, class selector.

2. Flight Listings

Shows available flights with prices, airlines, duration, booking buttons.

3. Contact Form Submission

Shows confirmation message and summary after sending a request.

4. Passenger Details Form

Fields for multiple passengers, pre-filled user info and ability to remove passengers.

5. Payment Summary Page

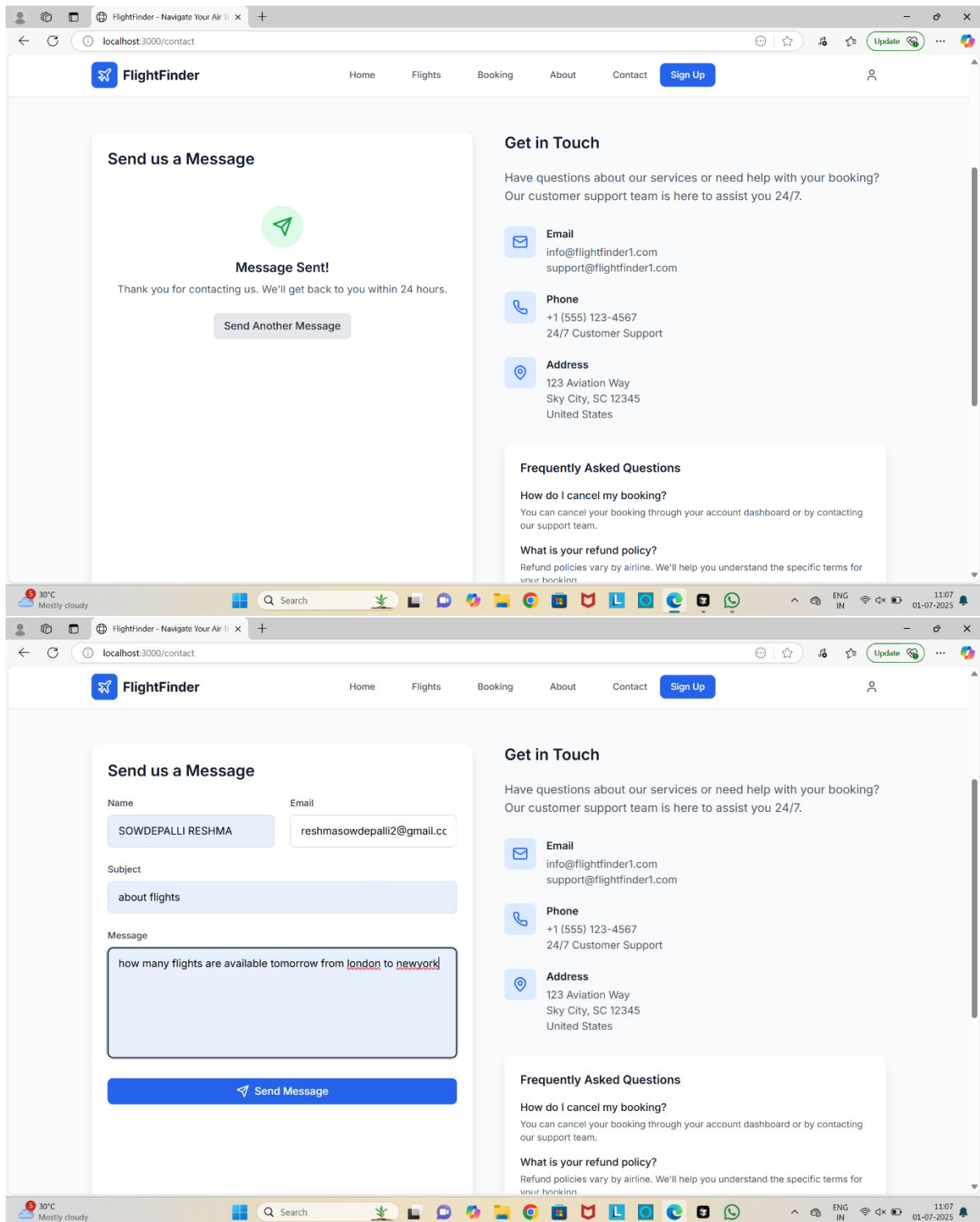
Input for card details and shows total summary including taxes and fees.

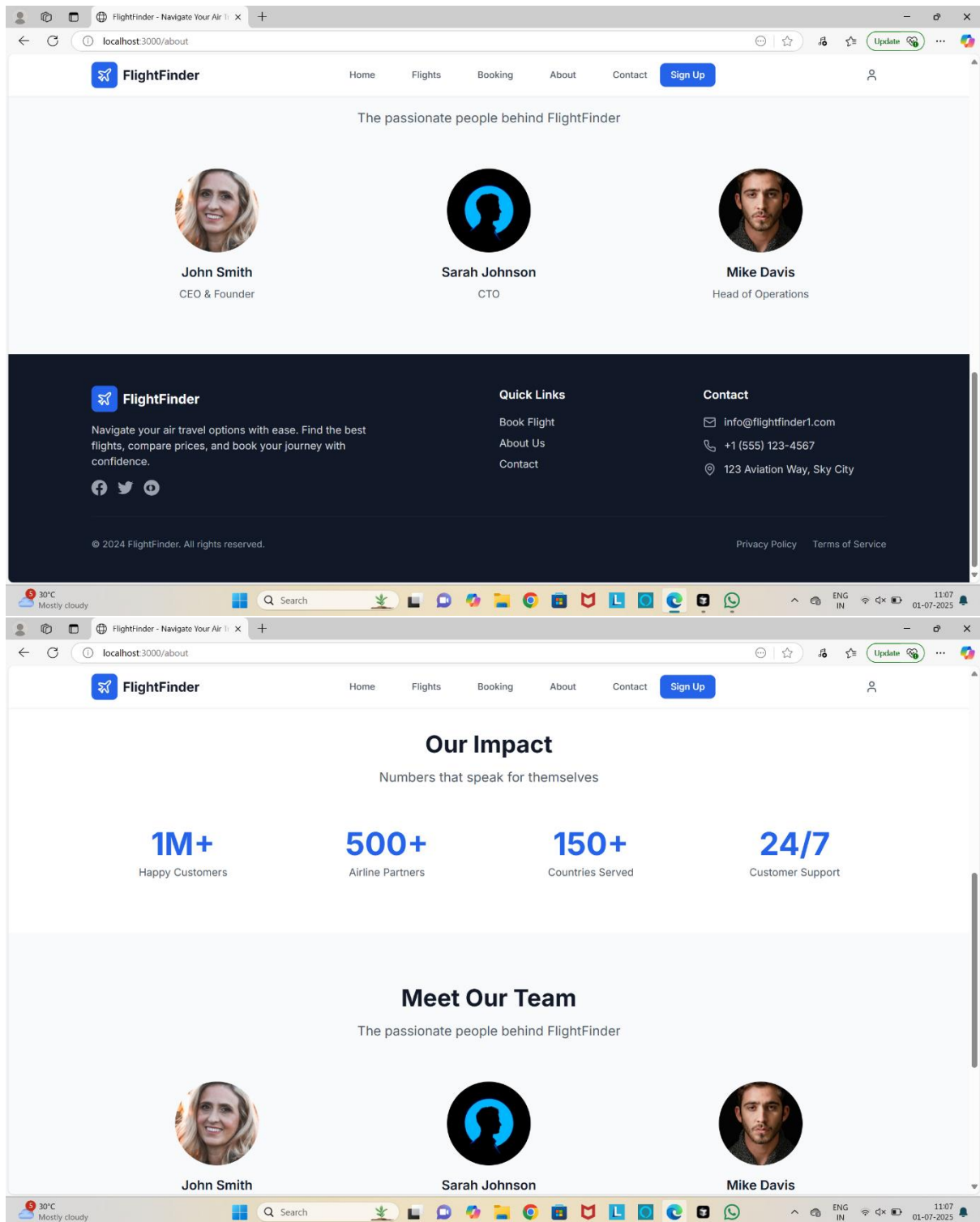
6. Contact Form Pre-Submission

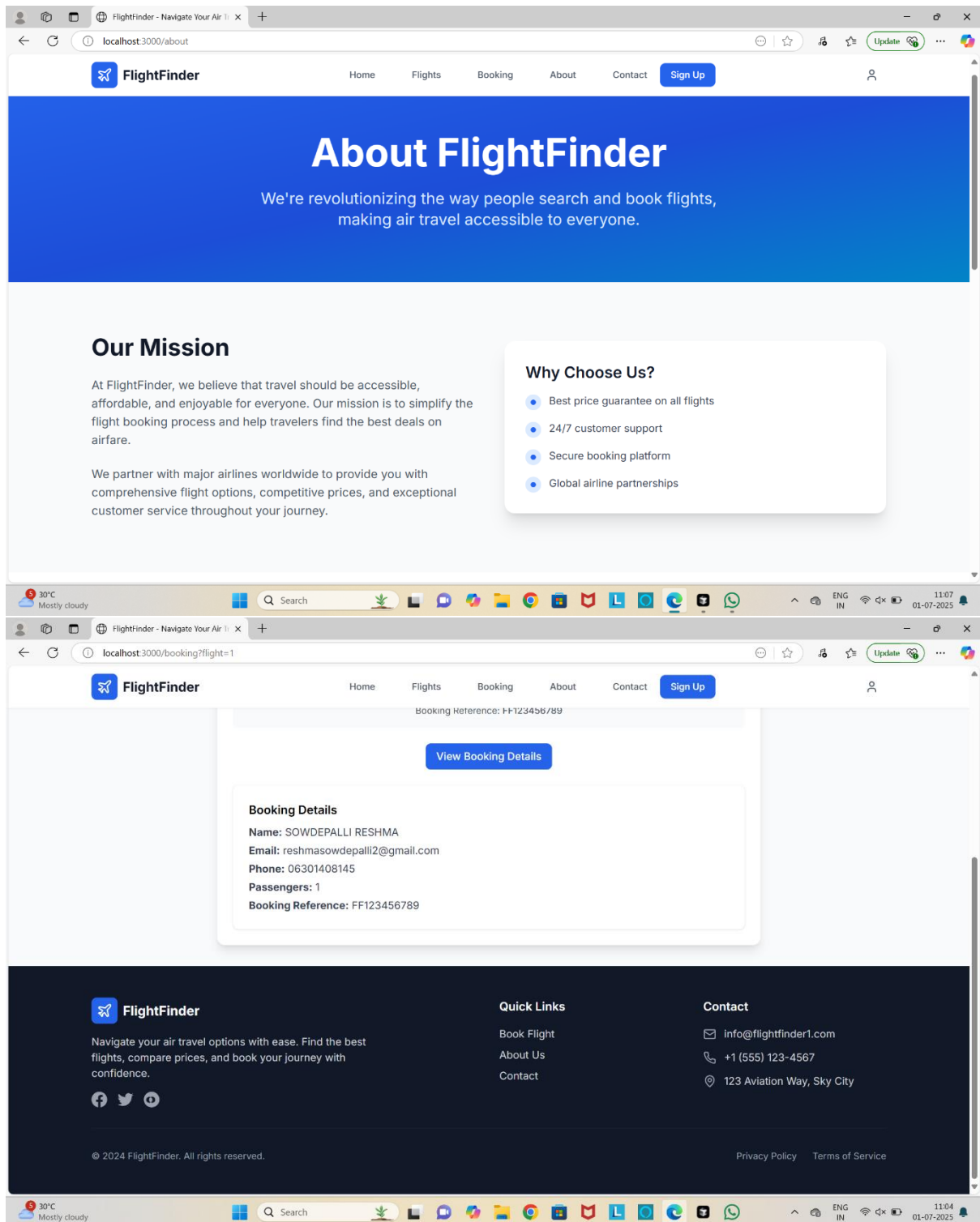
Input fields filled with user's details ready to send a query.

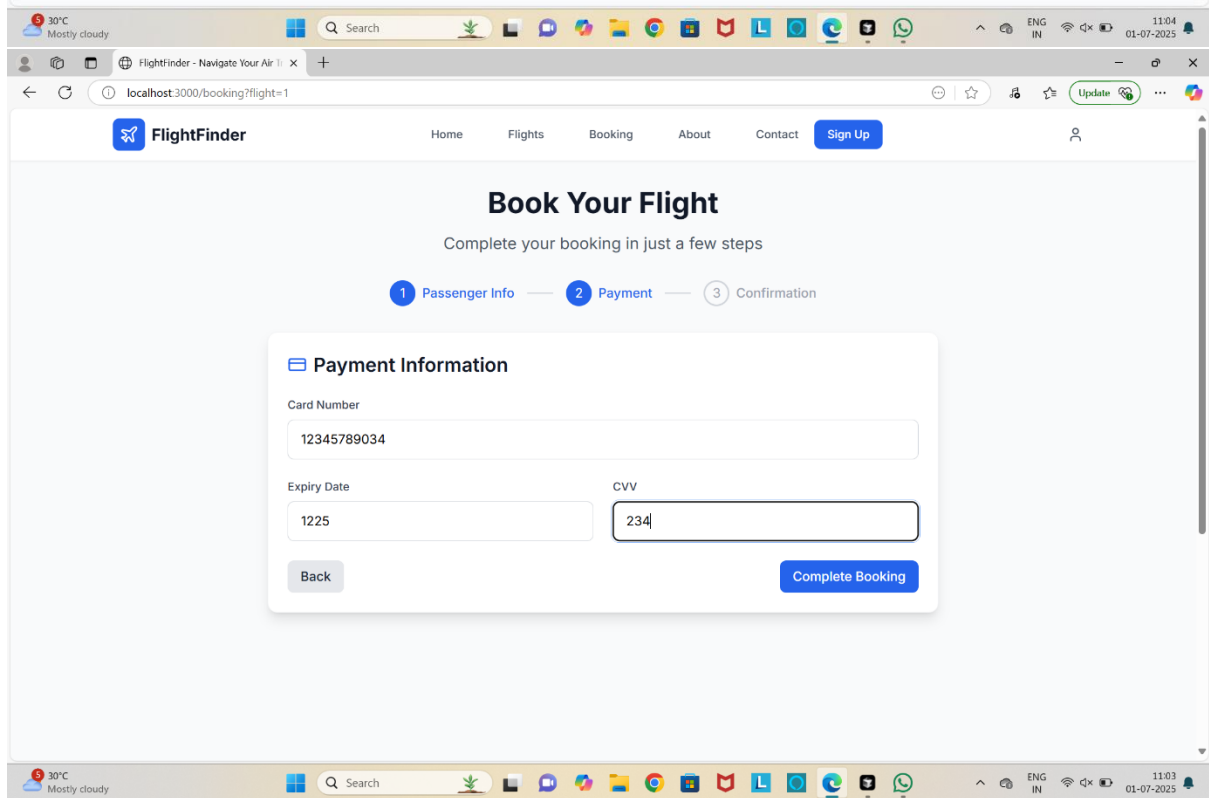
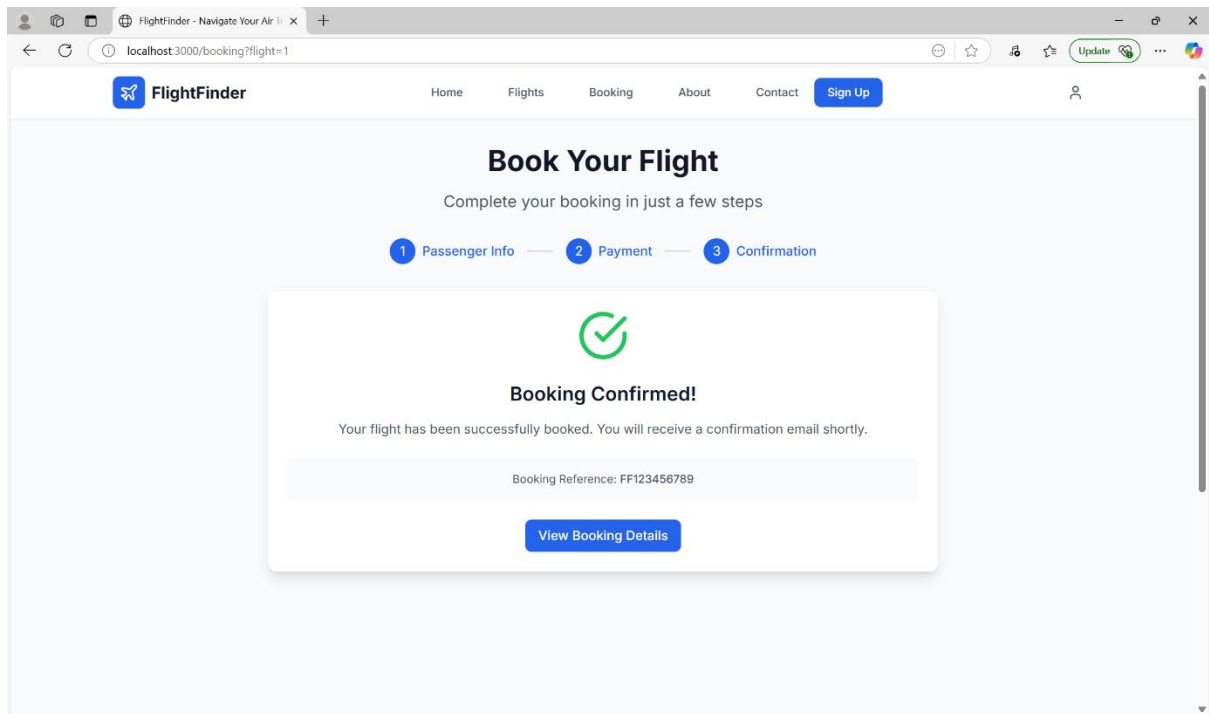
7. About Us Page

Overview of FlightFinder's mission and features.

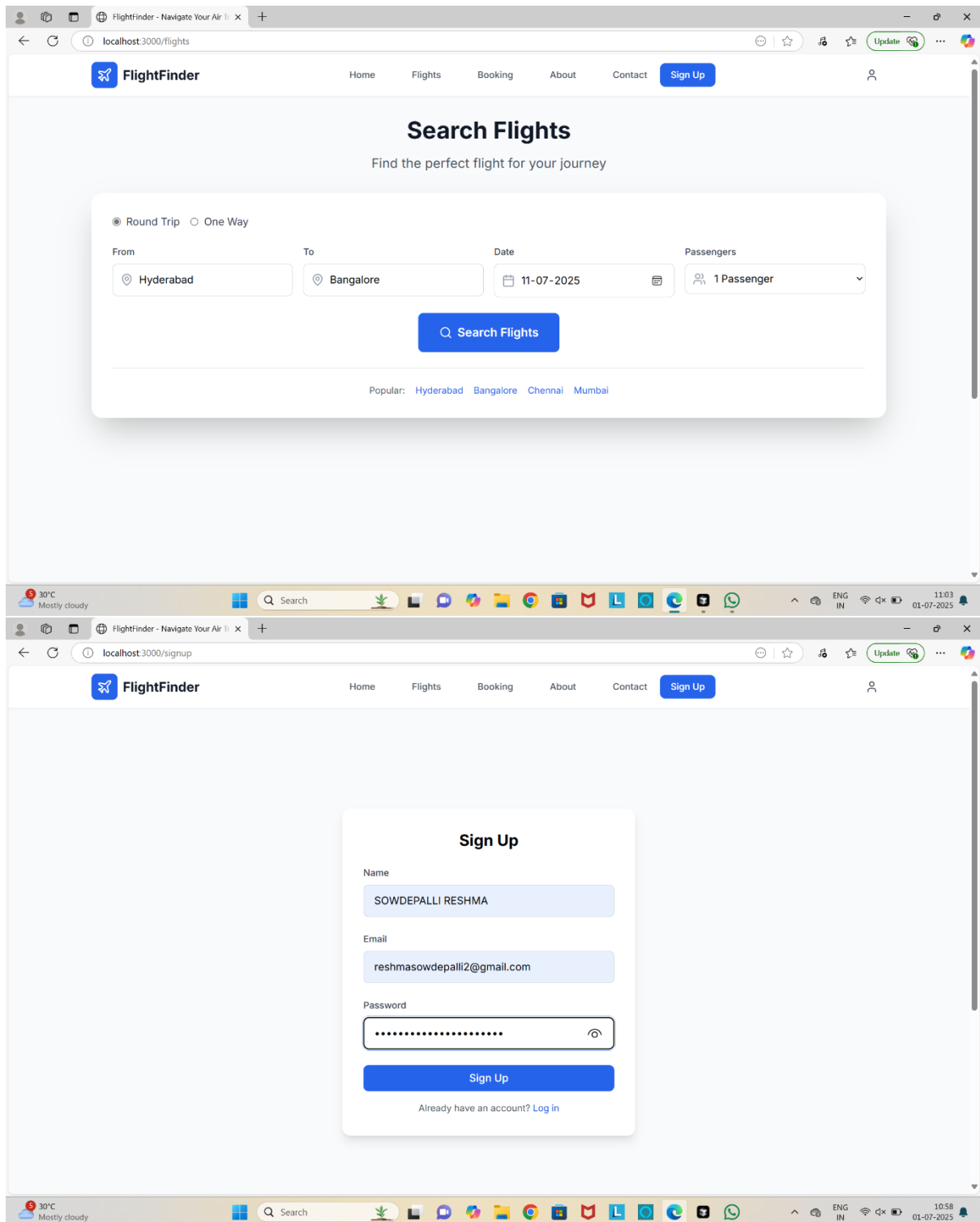


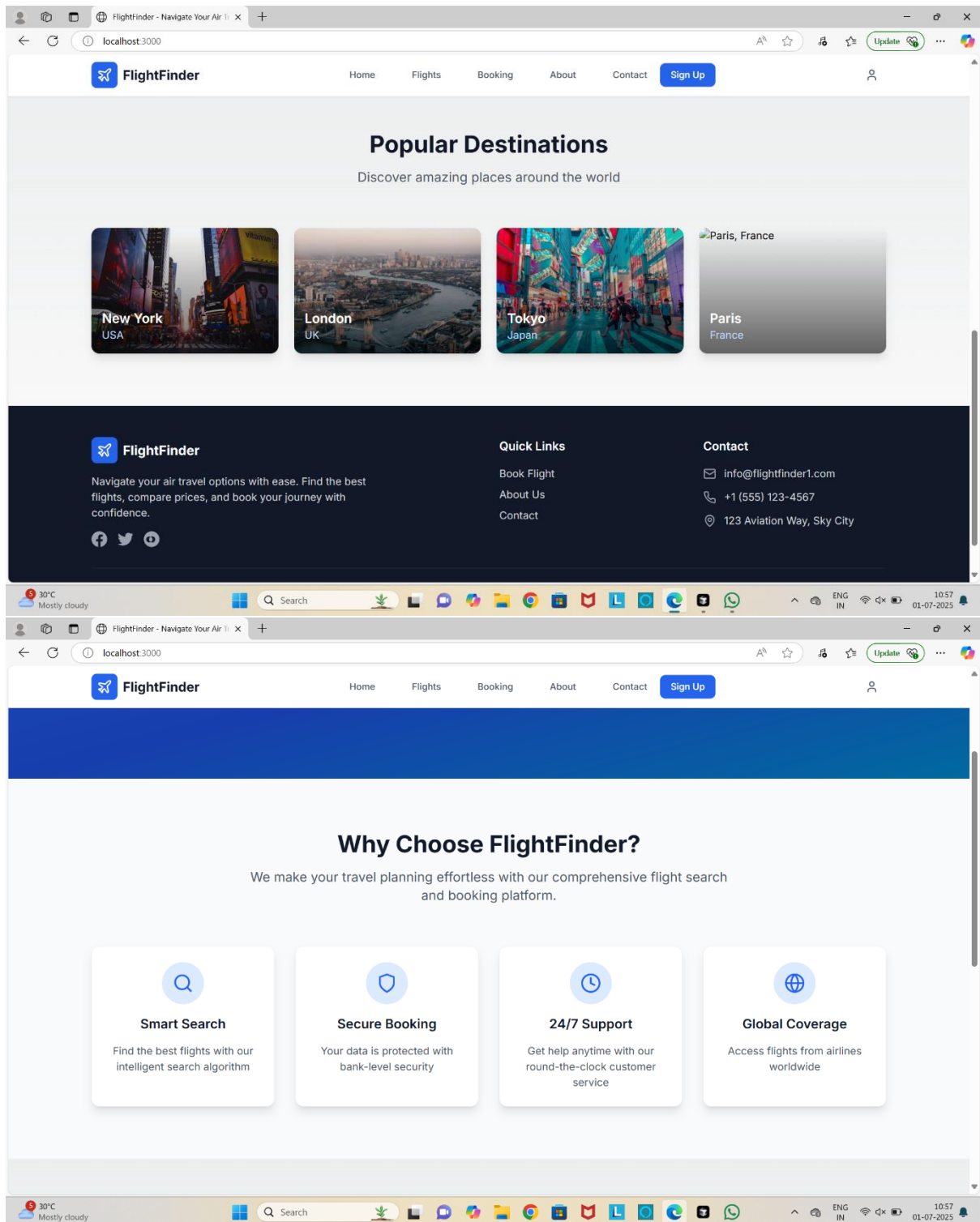


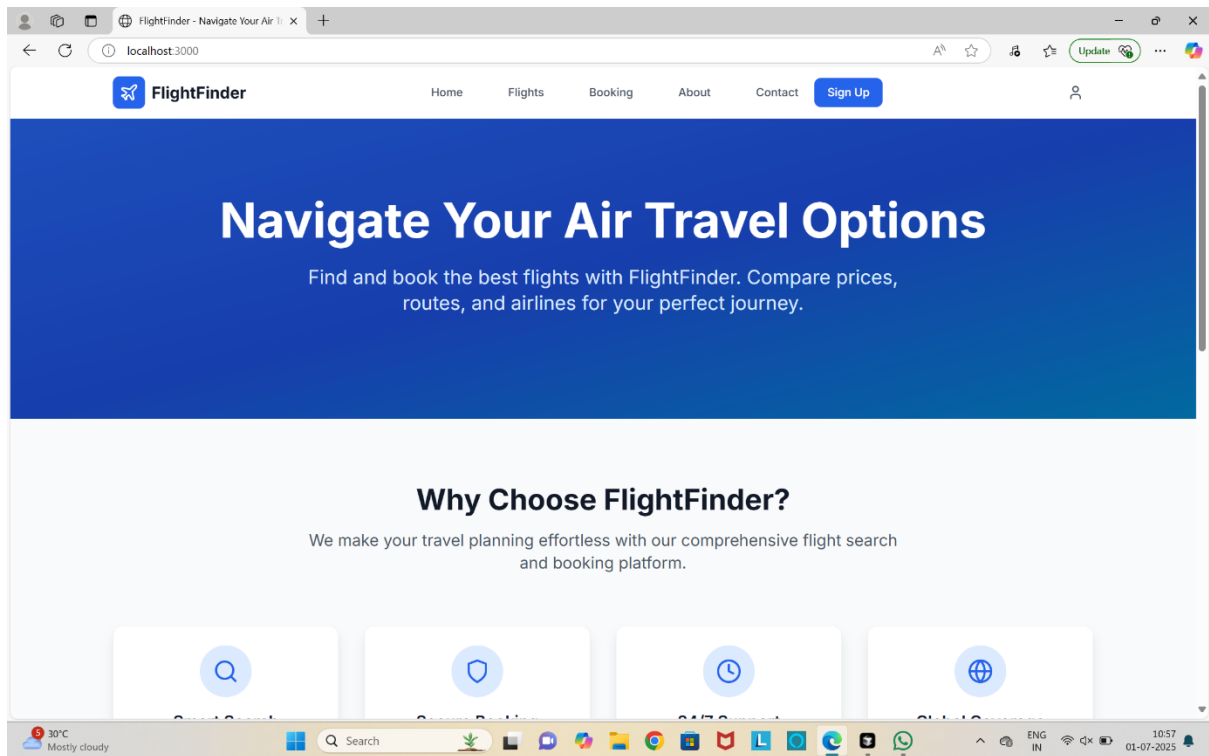




A screenshot of a web browser displaying the FlightFinder application. The browser's address bar shows 'localhost:3000/flights'. The application has a navigation bar with links for Home, Flights, Booking, About, and Contact, along with a 'Sign Up' button and a user profile icon. The main heading is 'Available Flights', with a subtext '3 flights found'. Three flight cards are listed: Delta Airlines (Flight DL123), British Airways (Flight BA456), and American Airlines (Flight AA789). Each card shows the origin 'New York' with a departure time of 10:30 AM, 2:15 PM, and 8:45 PM respectively, and a duration of 7h 15m. The destination is 'London' with arrival times of 10:45 PM, 6:30 PM, and 1:00 PM. The prices are \$450, \$520, and \$380 per passenger. Each card includes a 'Details' button and a 'Book Now' button. The browser's taskbar at the bottom shows the system clock as 11:03 on 01-07-2025, with a temperature of 30°C and 'Mostly cloudy' weather.







8. Advantages & Disadvantages

Advantages

- Unified flight search
- Quick comparisons and sorting
- Secure, user-friendly booking
- Responsive and intuitive UI

Disadvantages

- Dependent on third-party APIs
- No multi-city booking or seat selection yet
- Payment gateway simulated (needs real integration)

9. Conclusion

FlightFinder delivers a compelling experience for flight search and booking with essential features covered. The UI is clean, the booking flow is logical, and user support is built-in.

10. Future Scope

- Add multi-city booking
- Seat selection, booking history

Real payment gateway

Mobile app version

Detailed booking confirmations & email notifications

11. Appendix

Source Code: [Reshma-sowdepalli/Flightfinder-Navigating-Your-Air-Travel-Options](https://github.com/Reshma-sowdepalli/Flightfinder-Navigating-Your-Air-Travel-Options)

Project Demo: C:\Users\reshm\Videos\Captures