FlightWise Project Overview

# 1. Project Overview

The FlightWise platform is an advanced flight booking system that aims to simplify the process for users by offering AI-powered features such as flight search, price prediction, route optimization, booking management, and travel information. This platform leverages machine learning, optimization algorithms, and third-party integrations to provide users with an enhanced flight booking experience. The primary objectives of this project are:

1. To develop a seamless and efficient flight search system that matches user preferences.
2. To implement AI-driven price prediction techniques to help users book flights at the best possible rates.
3. To optimize flight routes using advanced algorithms for better travel experience based on user preferences.
4. To streamline booking management and integrate third-party services for easy flight booking and payment.
5. To provide up-to-date travel information, including visa requirements, to ensure users have all the necessary details for their trips.
6. To offer a user-friendly platform that combines flight search, prediction, and booking management for a comprehensive travel solution.

# 2. User Management

## Key Features

- Registration and authentication via JWT token for secure logins.  
 - Profile management to update user preferences.

## Endpoints

- POST /register: User registration.  
- POST /login: User authentication.  
- POST /logout: User logout.  
- GET /profile: View profile.  
- PUT /profile/data: Update user preferences.

## Models

- User: Stores user details.  
- User Preferences: Manages user preferences for recommendations.

# 3. Flight Search

## Key Features

- Allows users to search for flights based on origin, destination, dates, budget, etc.  
 - Saves search history for easy access to previous searches.

## Endpoints

- POST /search: Submit flight search criteria.

## Models

- Flight: Contains flight data.  
- Search History: Logs user searches.

# 4. AI/ML Prediction

## Key Features

- Price prediction to determine the best time to book flights.  
 - Notifies users when prices drop or when it's optimal to book.

## Endpoints

- POST /set/prediction/:route\_id: Predict flight prices for specific routes.

## Models

- Price Prediction Model: Predicts booking times based on historical data.  
- Price History: Stores historical flight prices.

# 5. Route Optimization

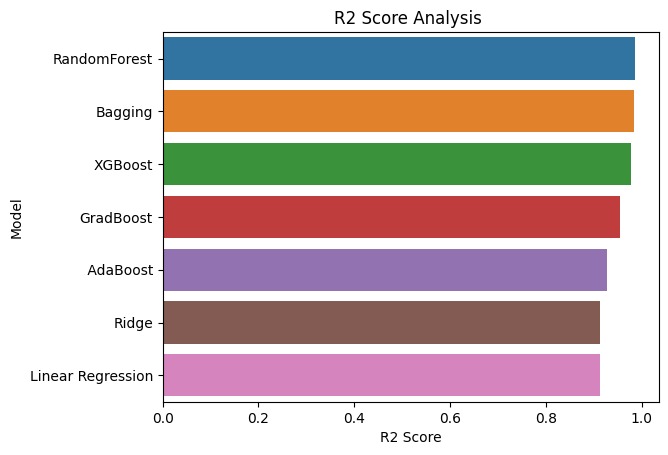
## Key Features

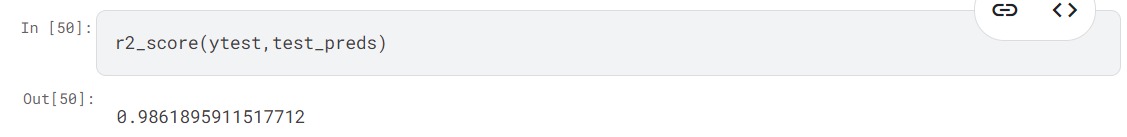
- Optimizes flight routes based on user preferences (e.g., shortest duration, least layovers).  
 - Utilizes algorithm Dijkstra’s for efficient route suggestions.

## Models

- Flight Route: Stores flight routes.  
- Route Optimization Model: Applies algorithms for route optimization.

- Accuracy:





# 6. Booking Management

## Key Features

- Allows users to book flights and manage bookings.  
 - Payment details integrated with third-party services for processing.

## Models

- Booking: Stores booking and transaction details.

# 7. VISA and Travel Info

## Key Features

- Provides relevant VISA and travel information using third-party APIs based on destination.

# 8. Database Schema (PostgreSQL)

1. Users: User details (ID, email, username, password, preferences).  
2. Flights: Flight data (ID, origin, destination, price, dates, layovers).  
3. Search History: User search logs (ID, user ID, search criteria, timestamp).  
4. Price History: Historical price data (ID, flight ID, date, price).  
5. Bookings: Transaction and booking data (ID, user ID, flight ID, transaction details).

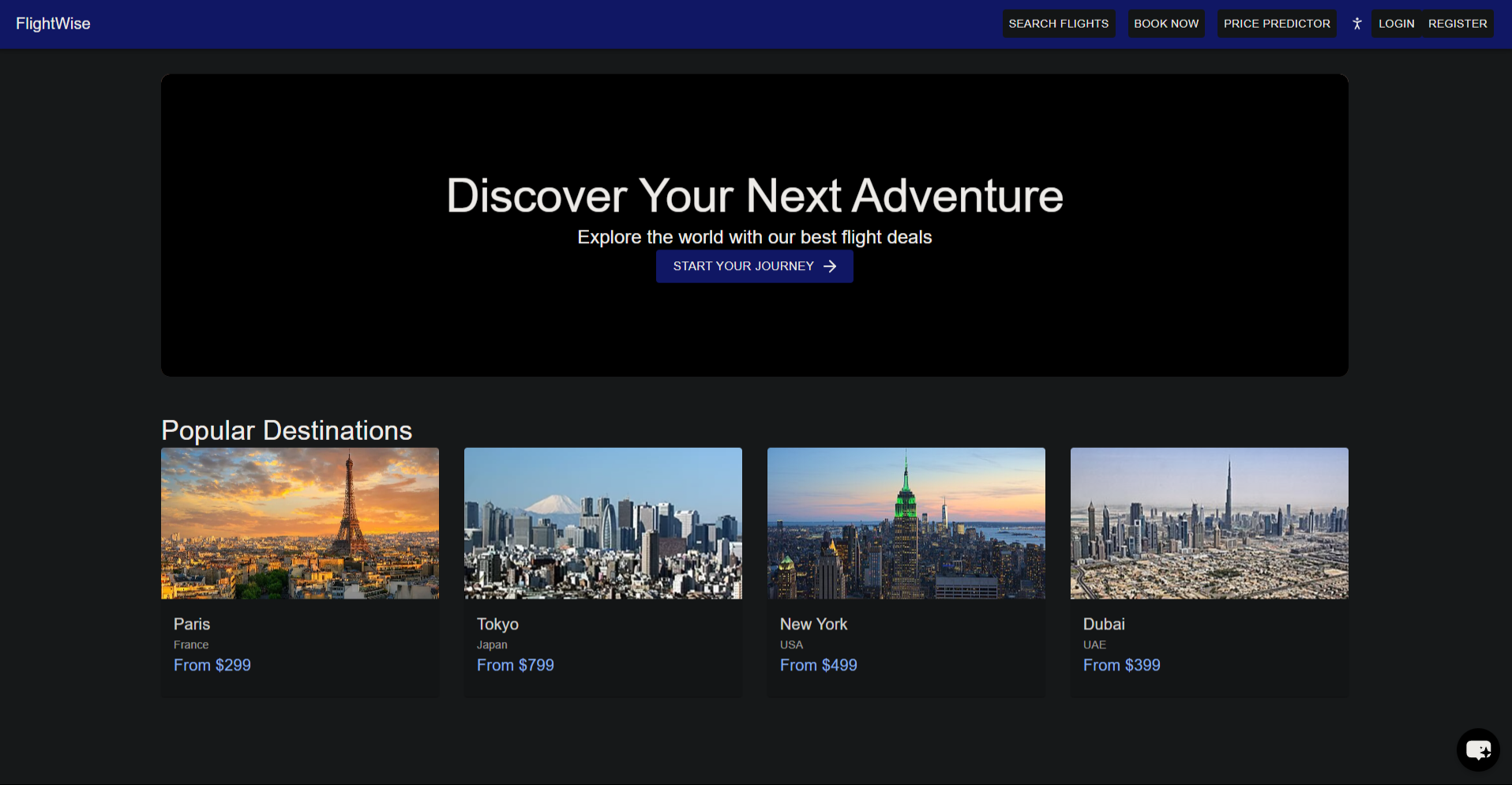
# 9. Backend API (Django + PostgreSQL)

1. User Management: register, login, logout, profile, profile data.  
2. Flight Search: search.  
3. AI/ML Prediction: set, prediction, route id.  
4. Booking Management: booking.  
5. Travel Info: travel info, destination.

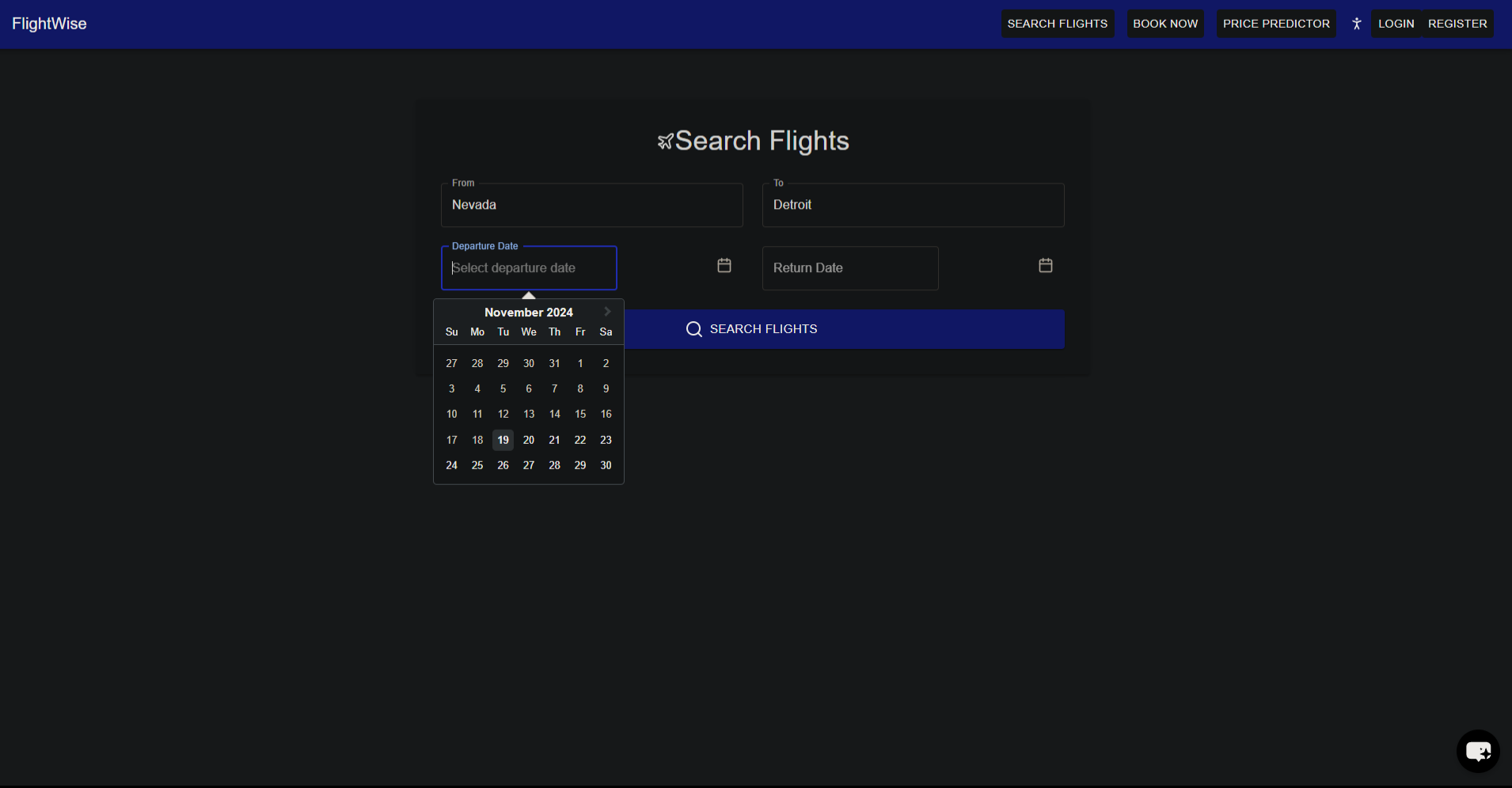
# 10. Frontend UI/UX (React & React Native)

## Components

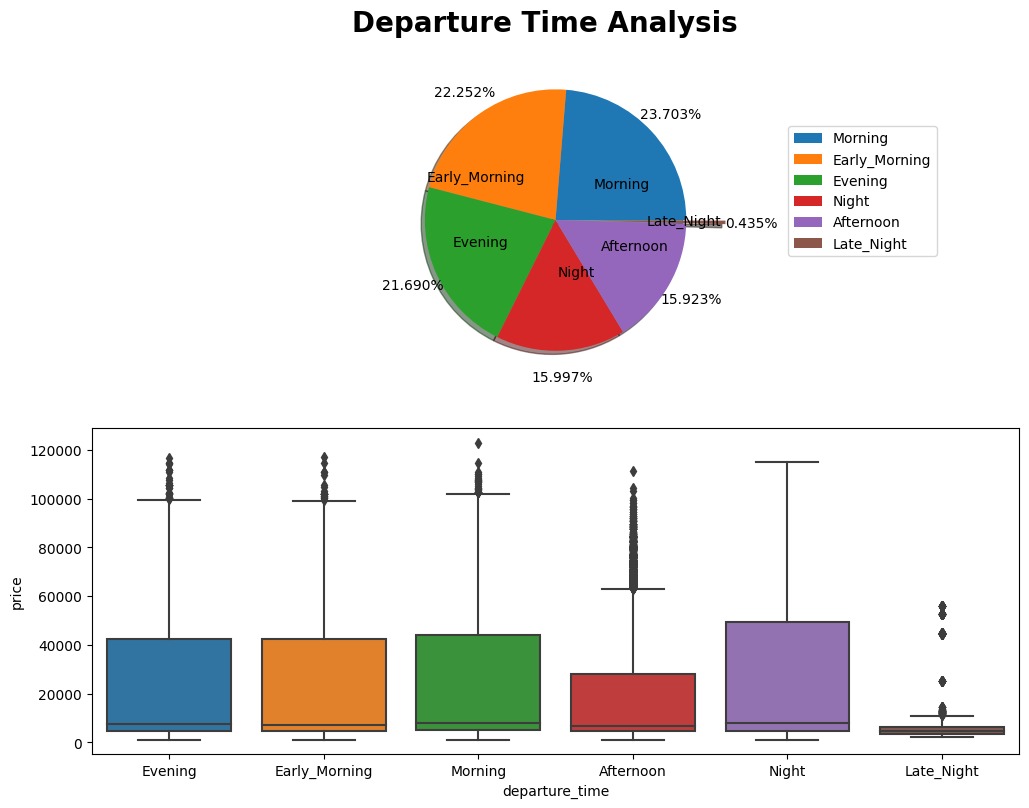
1. User Dashboard: Displays user data.



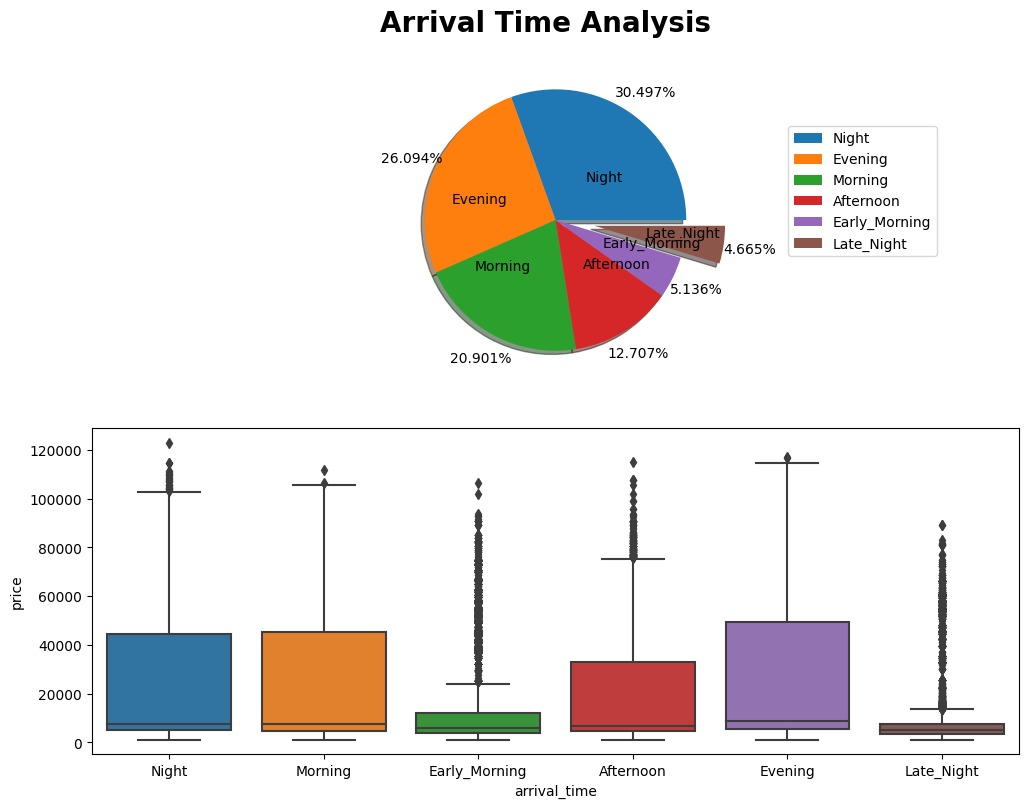
2. Flight Search: Search form and results.



3. Price Prediction: Displays predicted flight prices.



4. Booking Management: Transaction overview.



## State Management

- React Context or Redux for managing user data, flight searches, and bookings.  
 - React Query for data fetching.

# 11. Payment and Travel Integration

## Payment Gateways

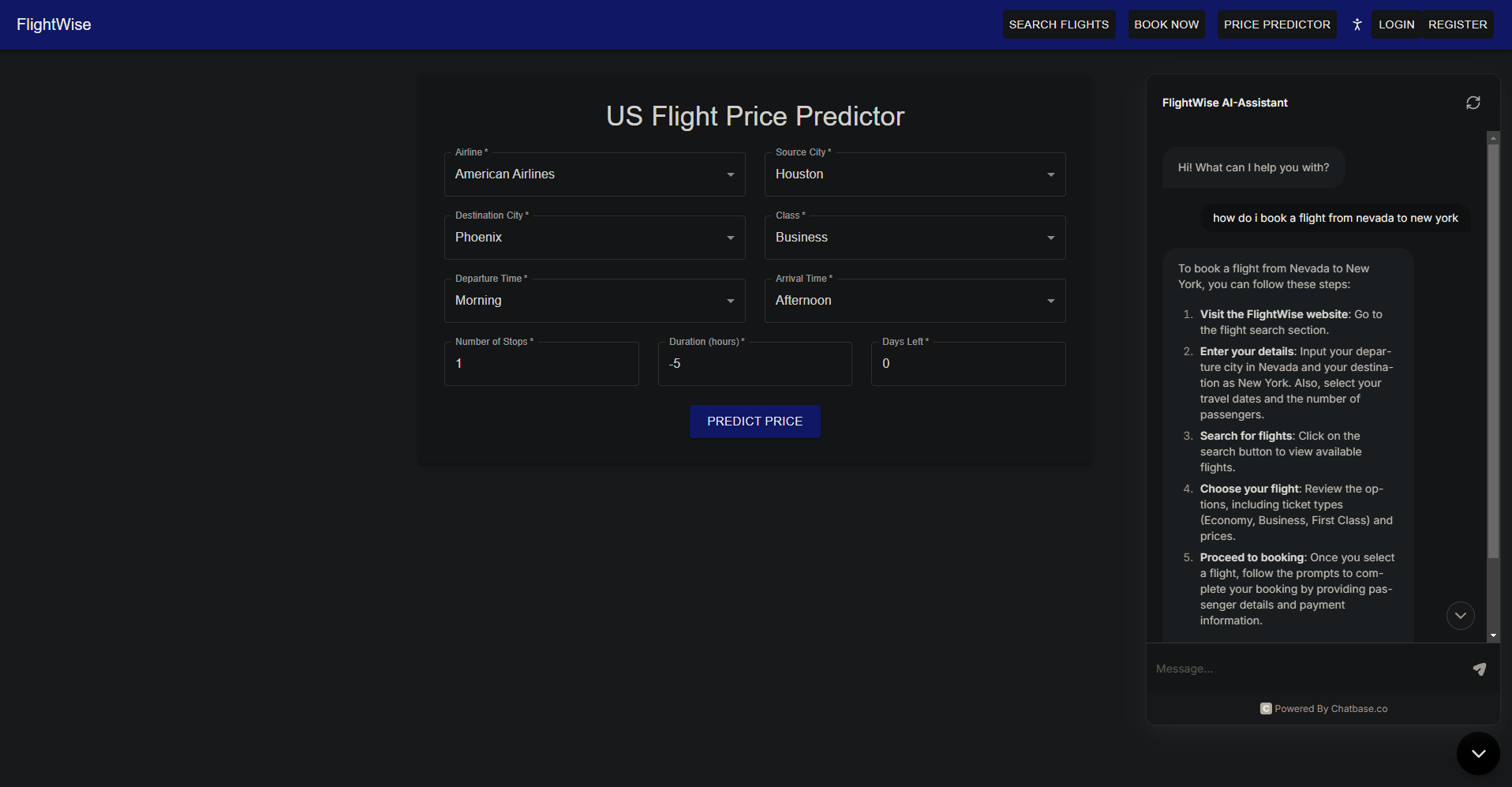
- Integrates with Stripe, Razorpay, or PayPal for booking transactions.

## Logistics and VISA Integration

- Uses third-party APIs for travel documentation and VISA requirements.

# 12. Chat Bot AI Powered Assistant

- We have implemented AI-powered chatbot platform that enables businesses to create custom, intelligent conversational agents for customer support, lead generation, and engagement.



# 13. Resources Used

1. Amadeus API: For flight search, pricing, and booking.  
 - Flight Offers Search: https://developers.amadeus.com/self-service/category/flights/api-doc/flight-offers-search  
 - Flight Offers Price: https://developers.amadeus.com/self-service/category/flights/api-doc/flight-offers-price  
 - Flight Create Orders: https://developers.amadeus.com/self-service/category/flights/api-doc/flight-create-orders  
2. Django and DRF Documentation:  
 - Django Documentation: https://docs.djangoproject.com/en/5.1/topics/  
 - Django REST Framework Quickstart: https://www.django-rest-framework.org/tutorial/quickstart/  
3. Libraries and Tools: amadeus, Django, Django REST framework, JWT for authentication, React/React Native for frontend, PostgreSQL for database, and machine learning tools like scikit-learn and pandas.