

# Travel Planner

---

A personalized travel recommendation application built with Ruby on Rails and JavaScript. Users can create profiles with their travel preferences, budget, and passport country to receive AI-powered destination suggestions tailored specifically for them.

<https://travel-planner-cs397-9396d2cb2102.herokuapp.com/>

## MVP Overview

A single integrated travel platform that combines **OpenAI**, **Google Maps/Flights**, and **TripAdvisor** APIs to provide a **feasible, safety-aware**, and **visa-friendly** travel experience.

---

## Core Features

- **User Input** – Collect user data such as nationality, budget, travel dates, and interests.
- **Filtering** – Automatically filter destinations based on **visa requirements**, **safety**, and **seasonal conditions**.
- **AI Recommendations** – Use **OpenAI API** with prompt engineering to generate **personalized**, **accessible** travel suggestions.
- **Data Integration** –
  - *Google Maps/Flights*: distance, flight options, hotels
  - *TripAdvisor*: attractions and local activities
- **Output** – Display a list of feasible destinations with **visa info**, **safety level**, and **top activities**.

## Features

- **User Authentication**: Secure signup/login with password encryption
- **Personalized Profiles**: Store travel preferences, budget range, passport country, and safety preferences
- **AI-Powered Recommendations**: Get destination suggestions based on your profile
- **Destination Management**: Browse and manage travel destinations
- **Travel Planning**: Create and manage travel plans
- **Responsive Design**: Beautiful, mobile-friendly interface with Bootstrap
- **API Support**: RESTful API endpoints for all resources

## Technology Stack

- **Backend**: Ruby on Rails 8.0.3
- **Frontend**: HTML, CSS, JavaScript, Bootstrap 5.3
- **Database**: SQLite (development), PostgreSQL (production ready)
- **Authentication**: bcrypt for secure password hashing
- **Styling**: Bootstrap 5 with custom CSS gradients
- **API**: RESTful JSON APIs with jbuilder

## Prerequisites

Before running this application, make sure you have the following installed:

- **Ruby**: Version 3.4.x (tested with 3.4.6)
- **Rails**: Version 8.0.3 or higher
- **Bundler**: For managing Ruby gems
- **Node.js**: For asset compilation (if needed)
- **Git**: For version control

## Installation & Setup

### 1. Clone the Repository

```
git clone https://github.com/NU-CS-Software-Studio-Fall-25/project-travel-planner.git  
cd project-travel-planner
```

### 2. Install Dependencies

```
# Install Ruby gems  
bundle install  
  
# If you encounter any gem conflicts, try:  
bundle update
```

### 3. Database Setup

```
# Create the database  
rails db:create  
  
# Run migrations  
rails db:migrate  
  
# Seed the database with sample data  
rails db:seed
```

### 4. Start the Application

```
# Start the Rails server  
rails server  
  
# Or use the shorthand  
rails s
```

The application will be available at <http://localhost:3000>

## 🎯 Usage

For New Users:

1. Visit <http://localhost:3000> - you'll see the landing page
2. Click "Create Account" or "Sign Up"
3. Fill out your travel profile including:
  - Name and email
  - Password (minimum 6 characters)
  - Passport country
  - Budget range (min/max)
  - Preferred travel season
  - Safety preference (1-10 scale)
4. After signup, you'll be redirected to your profile page
5. Click "🚀 Get Travel Recommendations" to start planning!

For Existing Users:

1. Visit <http://localhost:3000>
2. Click "Login" and enter your credentials
3. You'll be redirected to your profile page (your personal hub)

Test Accounts:

The seed data includes these test accounts (password: [password123](#)):

- [alex@example.com](#)
- [maria@example.com](#)
- [john@example.com](#)

## 📁 Project Structure

```
app/
  └── controllers/          # Request handling and business logic
    ├── application_controller.rb  # Authentication helpers
    ├── sessions_controller.rb    # Login/logout
    ├── users_controller.rb      # User management
    └── home_controller.rb      # Landing page
    └── api/v1/                # API endpoints
  └── models/                # Data models and validations
    ├── user.rb                # User authentication & preferences
    ├── destination.rb         # Travel destinations
    ├── travel_plan.rb         # User travel plans
    └── recommendation.rb     # Travel recommendations
  └── views/                 # HTML templates
    ├── layouts/application.html.erb # Main layout with navigation
    ├── users/                  # User profile pages
    └── sessions/               # Login forms
```

```
|   └── home/                      # Landing page
public/
└── index.html                   # Static landing page (alternative entry point)
config/
├── routes.rb                    # URL routing
└── database.yml                 # Database configuration
db/
└── migrate/                     # Database migrations
    └── seeds.rb                  # Sample data
```

## 🔒 Authentication Flow

1. **Landing Page** (/) - Static page or Rails home
2. **Signup** (/signup) - User registration with travel preferences
3. **Login** (/login) - User authentication
4. **Profile** (/users/:id) - User's personal hub after login
5. **Logout** - Destroys session, returns to landing page

## 🎨 Styling & UI

- **Bootstrap 5.3:** Responsive grid system and components
- **Custom CSS:** Beautiful gradients and hover effects
- **Color Scheme:** Purple/blue gradients with green accents
- **Mobile-First:** Responsive design that works on all devices

## 🛡 Security Features

- **Password Encryption:** Using bcrypt for secure password hashing
- **Session Management:** Rails session-based authentication
- **CSRF Protection:** Cross-site request forgery protection enabled
- **User Authorization:** Users can only access/edit their own profiles
- **Input Validation:** Comprehensive model validations

## 🧪 Testing

```
# Run the test suite
rails test

# For specific test files
rails test test/models/user_test.rb
```

## 🚀 Deployment

The application is configured for deployment on platforms like Heroku:

```
# For Heroku deployment
git push heroku main
```

```
heroku run rails db:migrate  
heroku run rails db:seed
```

## Troubleshooting

Common Issues:

1. **Gem conflicts:** Run `bundle update` or delete `Gemfile.lock` and run `bundle install`
2. **Database issues:** Try `rails db:reset` to recreate and reseed
3. **Port conflicts:** Use `rails s -p 3001` to run on a different port
4. **Asset issues:** Run `rails assets:precompile` if needed

Error Messages:

- "**Password can't be blank**": Make sure you're using the web form at `/signup`, not the API
- "**Email already exists**": Use a different email or login with existing account

## API Documentation

The application provides RESTful API endpoints:

- `GET /api/v1/users` - List all users
- `POST /api/v1/users` - Create new user (include password fields)
- `GET /api/v1/destinations` - List destinations
- `POST /api/v1/travel_recommendations` - Get recommendations

## Contributing

1. Fork the repository
2. Create a feature branch (`git checkout -b feature/amazing-feature`)
3. Commit your changes (`git commit -m 'Add amazing feature'`)
4. Push to the branch (`git push origin feature/amazing-feature`)
5. Open a Pull Request

## Development Team

Built by: **Sirui Chen, Ailin Chu, Jerome Bizimana, Rohun Gargya**

## License

This project is part of the Northwestern University CS Software Studio course (Fall 2025).

---

Happy Traveling! 