

Project Manual: Wanderlust.ai

This project manual provides a comprehensive guide to setting up, running, and using the "Wanderlust.ai" travel planning website.

1. Prerequisites

Before you start, ensure you have the following installed on your system:

- Python 3.x
- Flask and other necessary Python packages
- Google Generative AI SDK
- OpenWeatherMap API Key

2. Project Setup

2.1. Clone or Download the Project

Clone the project from your repository or download and extract it to your desired location.

2.2. Install Dependencies

Navigate to the project directory and install the required Python packages:

```
bash
pip install -r requirements.txt
```

2.3. Configure API Keys

Ensure you have the following API keys:

- Google API Key for Google Generative AI
- OpenWeather API Key for fetching weather and air quality data

Place these API keys in app.py:

```
python
GOOGLE_API_KEY = "YOUR_GOOGLE_API_KEY"
OPEN_WEATHER_API_KEY = "YOUR_OPENWEATHER_API_KEY"
```

3. Running the Project

3.1. Start the Flask Application

In the terminal, navigate to the project directory and run:

```
bash
python app.py
```

The Flask development server should start, typically accessible at <http://127.0.0.1:5000/>.

3.2. Access the Website

Open a web browser and go to <http://127.0.0.1:5000/> to view the website.

4. Features and Usage

4.1. Home Page

- Video Section: View the introductory video with an overview of the website.
- Feature Boxes: Learn about the key features offered by the site.
- Itinerary Form: Enter travel details (destination, start date, end date, interests) to generate a personalized itinerary.

4.2. Itinerary Generation

- After filling in the itinerary form, click "Get Itinerary." The generated itinerary will be displayed below the form.

4.3. Real-Time Updates

- The site can provide real-time weather and air quality updates based on the user's current location.

4.4. Chatbot

- Use the chatbot feature to ask travel-related questions or seek assistance.

4.5. Admin Dashboard

- Admin Login: Access the admin section using the provided credentials (admin for both username and password).
- Data Viewing: After logging in, admins can view and search data in the `places.csv` and `cities.csv` files.

5. Troubleshooting

5.1. Common Issues

- API Key Errors: Ensure your API keys are valid and have the correct permissions.
- CSV File Issues: Verify that your CSV files are correctly named and placed in the `data/` directory.

5.2. Debugging

- Use Flask's built-in debugging features by running the app in debug mode:

```
bash  
python app.py --debug
```

- Check the console for error messages and stack traces.

6. Additional Information

- Documentation: For more details on the Flask framework, visit the [Flask Documentation](<https://flask.palletsprojects.com/>).
- API Documentation: Refer to the respective API documentation for Google Generative AI and OpenWeatherMap for detailed usage instructions and limits.