**Wonderlust Tours Documentation**

**Table of Contents**

1. Introduction

2. Project Setup

1. Prerequisites

2. Installation

3. Configuration

3. Project Structure

4. Database Schema

5. Application Routes

6. Templates

7. Styling

8. Application Features

9. Deployment

10. Contributing

11. License

12. Acknowledgements

**Introduction**

Wonderlust Tours is a web application for discovering and booking tours and travel packages. The application is built using Flask, a lightweight web framework in Python. This documentation provides a comprehensive guide on how to set up, use, and contribute to the Wonderlust Tours project.

**Project Setup**

**Prerequisites**

Before setting up the project, ensure you have the following installed:

- Python 3.8+

- pip (Python package installer)

- Virtual environment tool (e.g., venv or virtualenv)

- SQLite (for the database)

Installation

1. Clone the Repository:

git clone https://github.com/yourusername/WonderlustTours.git

cd Wonderlust Tours

2. Create a Virtual Environment:

python -m venv venv

source venv/bin/activate On Windows, use `venv\Scripts\activate

3. Install Dependencies:

pip install -r requirements.txt

Configuration

1. Environment Variables:

Create a `.env` file in the root directory and add the following variables:

FLASK\_APP=run.py

FLASK\_ENV=development

DATABASE\_URL=sqlite:///Wonderlust Tours.db

SECRET\_KEY=your\_secret\_key

2. Initialize the Database:

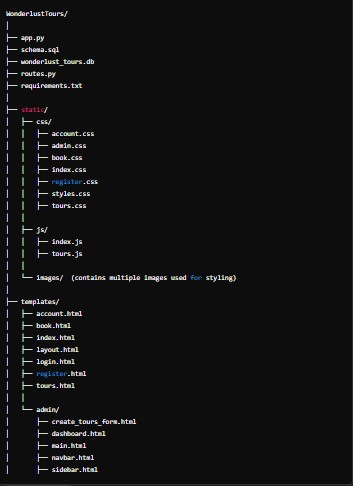
flask db init

flask db migrate -m "Initial migration"

flask db upgrade

Project Structure

The project structure is as follows:



Database Schema

The application uses the following database schema:

1. User:

- `id` (Integer, Primary Key)

- `username` (String)

- `email` (String)

- `password\_hash` (String)

2. Tour:

- `id` (Integer, Primary Key)

- `title` (String)

- `description` (Text)

- `price` (Float)

- `image\_url` (String)

3. Booking:

- `id` (Integer, Primary Key)

- `user\_id` (Foreign Key to User)

- `tour\_id` (Foreign Key to Tour)

- `date` (Date)

Application Routes

The application defines the following routes:

- Home Page: `/`

- Renders the homepage.

- Tour Listings: `/tours`

- Displays a list of available tours.

- Book Tour: `/book/<int:tour\_id>`

- Allows users to book a tour.

- User Account: `/account`

- Displays user account information and upcoming trips.

Templates

The application uses the following HTML templates:

- `layout.html`: Base template with common structure.

- `index.html`: Home page template.

- `tours.html`: Tour listings page.

- `tour\_detail.html`: Tour details page.

- `book.html`: Booking page.

- `account.html`: User account management page.

Styling

The application uses Bootstrap for styling. Custom styles can be added in `static/css/styles.css`.

Application Features

Wonderlust Tours offers the following key features:

Home Page

The home page welcomes users and provides an overview of the services offered. It includes a call-to-action button to explore available tours.

Tour Listings

The tour listings page displays all available tours in a grid layout. Each tour card includes an image, title, brief description, and a link to view more details.

Tour Details

The tour details page provides comprehensive information about a specific tour, including images, full description, and pricing. Users can also book the tour from this page.

Booking

The booking page allows users to book a tour by filling out a form with their name, email, and preferred date. This information is saved to the database.

User Account Management

The user account management page displays the user's information and a list of their upcoming trips. Users can view their booked tours and relevant details.

Responsive Design

The application is fully responsive, ensuring a seamless experience across various devices, including desktops, tablets, and mobile phones.

Deployment

To deploy the application, follow these steps:

1. Set Up a Production Server:

- Choose a server (e.g., Heroku, AWS, DigitalOcean).

2. Configure Environment Variables:

- Set the necessary environment variables for production.

3. Install Dependencies:

- Ensure all dependencies are installed on the server.

4. Run the Application:

- Use a production-ready server like Gunicorn to run the application.

gunicorn -w 4 run:app

Contributing

To contribute to this project, follow these steps:

1. Fork the repository.

2. Create a new branch (`git checkout -b feature-branch`).

3. Make your changes.

4. Commit your changes (`git commit -m 'Add new feature'`).

5. Push to the branch (`git push origin feature-branch`).

6. Create a pull request.

License

This project is licensed under the MIT License. See the LICENSE(LICENSE) file for more details.

Acknowledgements

- Flask: The web framework used.

- Bootstrap: The front-end framework used for styling.

- All contributors who have helped improve this project.

---

This structure ensures that all aspects of the Wonderlust Tours project are well-documented, from setup and configuration to detailed explanations of the application's features and functionalities.