**Flask Recipe Application**

**Overview**

This project is a Flask-based web application designed to manage and share recipes. It includes functionalities such as user authentication, recipe management, and integration with third-party services.

**Features**

* User authentication (registration, login, password reset)
* Recipe management (add, view, edit, delete recipes)
* Token management for authentication
* Integration with external APIs (if any)

**Technologies Used**

* Python with Flask
* HTML/CSS for templates
* SQLite/PostgreSQL for the database
* Various Python libraries (Flask-Login, Flask-WTF, etc.)

**Getting Started**

**Prerequisites**

Ensure you have the following installed:

* Python 3.x
* Virtualenv
* SQLite/PostgreSQL (depending on your configuration)

**Installation**

1. **Clone the repository:**

bash

Copy code

git clone https://github.com/yourusername/your-repo-name.git

cd your-repo-name

1. **Create a virtual environment:**

bash

Copy code

python -m venv venv

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

1. **Install the dependencies:**

bash

Copy code

pip install -r requirements.txt

1. **Set up environment variables:**

Create a .env file in the project root and add your configuration details:

makefile

Copy code

SECRET\_KEY=your\_secret\_key

SQLALCHEMY\_DATABASE\_URI=your\_database\_uri

1. **Run the application:**

bash

Copy code

flask run

**Project Structure**

arduino

Copy code

.

├── app/

│ ├── \_\_init\_\_.py

│ ├── recipe.py

│ ├── token.py

│ ├── models.py

│ ├── forms.py

│ ├── templates/

│ └── static/

├── migrations/

├── .env

├── config.py

├── requirements.txt

└── run.py

* app/: Contains the main application code (models, forms, recipes, tokens, templates, and static files).
* migrations/: Database migration scripts.
* .env: Environment variables file.
* config.py: Configuration settings for the app.
* requirements.txt: Python dependencies.
* run.py: Entry point to start the Flask application.

**Routes**

**Public Routes**

* /register: User registration page.
* /login: User login page.
* /logout: User logout route.
* /recipes: View all recipes.
* /recipe/<id>: View a specific recipe.

**Protected Routes (require login)**

* /add\_recipe: Add a new recipe.
* /edit\_recipe/<id>: Edit an existing recipe.
* /delete\_recipe/<id>: Delete a recipe.

**Contribution**

Contributions are welcome! Please fork the repository and create a pull request with your changes.

**License**

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

**Flask Recipe Application**

**Overview**

This project is a Flask-based web application designed to manage and share recipes. It includes functionalities such as user authentication, recipe management, and integration with third-party services.

**Features**

* User authentication (registration, login, password reset)
* Recipe management (add, view, edit, delete recipes)
* Token management for authentication
* Integration with external APIs (if any)

**Technologies Used**

* Python with Flask
* HTML/CSS for templates
* SQLite/PostgreSQL for the database
* Various Python libraries (Flask-Login, Flask-WTF, etc.)

**Getting Started**

**Prerequisites**

Ensure you have the following installed:

* Python 3.x
* Virtualenv
* SQLite/PostgreSQL (depending on your configuration)

**Installation**

1. **Clone the repository:**

bash

Copy code

git clone https://github.com/yourusername/your-repo-name.git

cd your-repo-name

1. **Create a virtual environment:**

bash

Copy code

python -m venv venv

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

1. **Install the dependencies:**

bash

Copy code

pip install -r requirements.txt

1. **Set up environment variables:**

Create a .env file in the project root and add your configuration details:

makefile

Copy code

SECRET\_KEY=your\_secret\_key

SQLALCHEMY\_DATABASE\_URI=your\_database\_uri

1. **Run the application:**

bash

Copy code

flask run

**Project Structure**

arduino

Copy code

.

├── app/

│ ├── \_\_init\_\_.py

│ ├── recipe.py

│ ├── token.py

│ ├── models.py

│ ├── forms.py

│ ├── templates/

│ └── static/

├── migrations/

├── .env

├── config.py

├── requirements.txt

└── run.py

* app/: Contains the main application code (models, forms, recipes, tokens, templates, and static files).
* migrations/: Database migration scripts.
* .env: Environment variables file.
* config.py: Configuration settings for the app.
* requirements.txt: Python dependencies.
* run.py: Entry point to start the Flask application.

**Routes**

**Public Routes**

* /register: User registration page.
* /login: User login page.
* /logout: User logout route.
* /recipes: View all recipes.
* /recipe/<id>: View a specific recipe.

**Protected Routes (require login)**

* /add\_recipe: Add a new recipe.
* /edit\_recipe/<id>: Edit an existing recipe.
* /delete\_recipe/<id>: Delete a recipe.

**Contribution**

Contributions are welcome! Please fork the repository and create a pull request with your changes.

**License**

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