CFGDegree Spring | Software Engineering

Software 2
Group 6

Recipea APP

TEAM

Database Development:
- Danielle Lott

Back-end development.

Endpoint Development and implementation:
- Isolda Liborio, Ella Myton, Raluca Teslovan, Lidiia Kliuchna

Console App Implementation:
- Ella Myton

Testing and Quality Assurance:
- Danielle Lott

Front-end:
- Elizabeth Turay
- Raluca Teslovan

Project Management and Documentation:
- Isolda Liborio

ABOUT

Ignite Your Culinary Passion!

Welcome to Recipea, the app for cooking lovers! Add your own recipes or family favourites and explore recipes by categories. Connect with like-minded cooks who share your love for food.

Personalized Recipe Collection

Curate your own collection of recipes. Add your favourite family secrets and culinary masterpieces for easy access.

Easy Recipe Discovery:

Discover new recipes effortlessly. Explore categories like desserts, main courses, and vegetarian options.

Connect with Like-Minded Cooks:

Join a community of food enthusiasts. Exchange tips, techniques, and build lasting connections with fellow foodies.

Recipea is the go-to app for cooking enthusiasts. Store, explore, and share recipes seamlessly. Revolutionise your cooking experience with Recipea!

TECH STACK

- Front end - to be defined
- Back-end - Python
- API - Flask
- DB - SQL

FEATURES:

First stage

Recipe Management:

- Create
- Update
- Delete recipes

Recipe Search:

- Search recipes by Ingredient, health, cuisine type,
    - Show recipes info from the app Database
    - Shows recipes from the Edamam API
- Research for recipes that gather Edamam API recipes and recipes from the app API Database. (to be implemented - on backlog tasks)

Next stages:

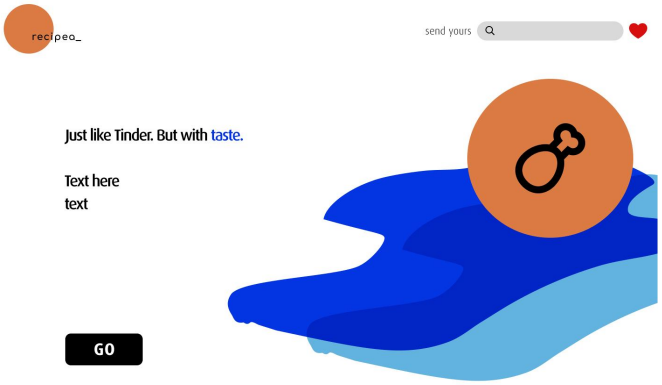
User management:

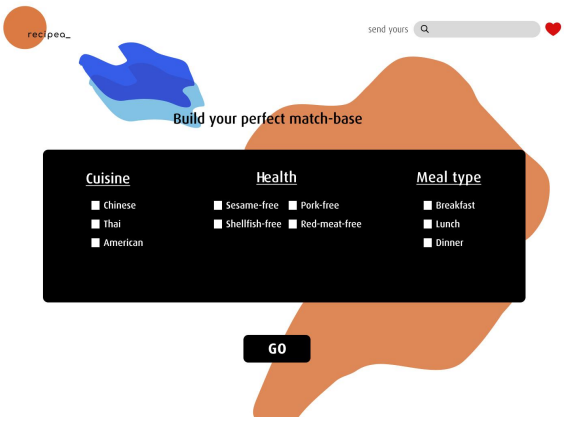
- Create an account, login, and logout
- Manage their profile, including updating their personal information and changing their password.
- Manage your recipe collection

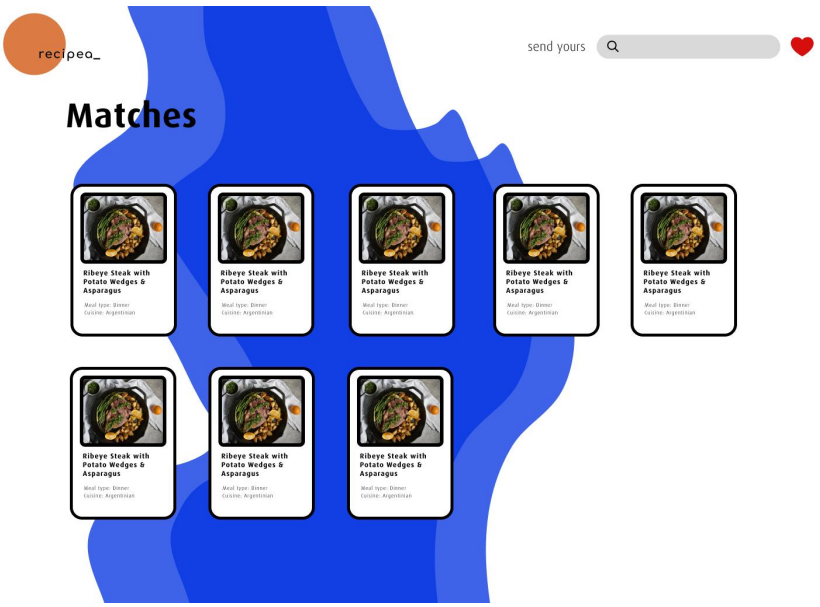
Match:

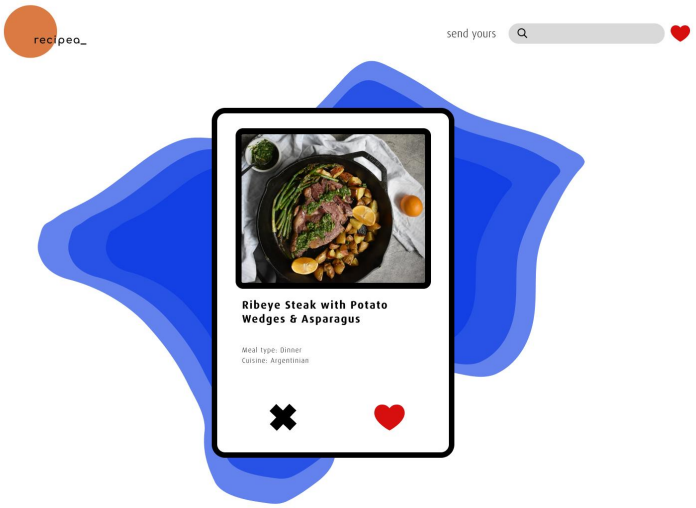
- Follow users
- Like recipes
- Connect with fellow foodies your profiles match.

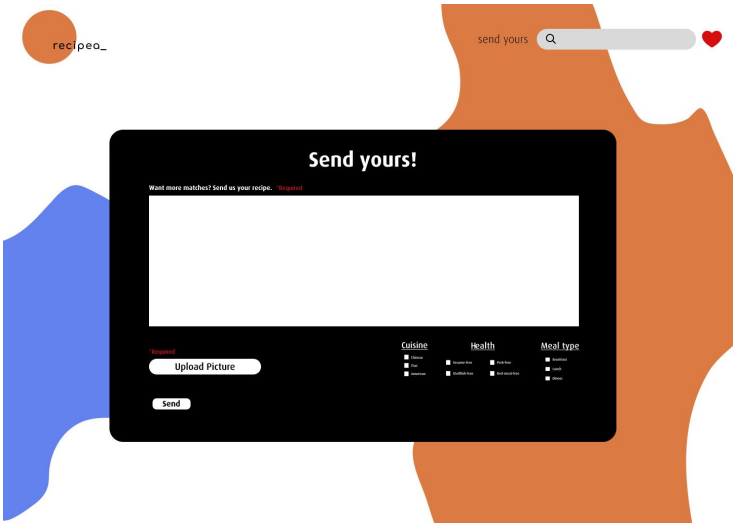
WIREFRAME SKETCH:



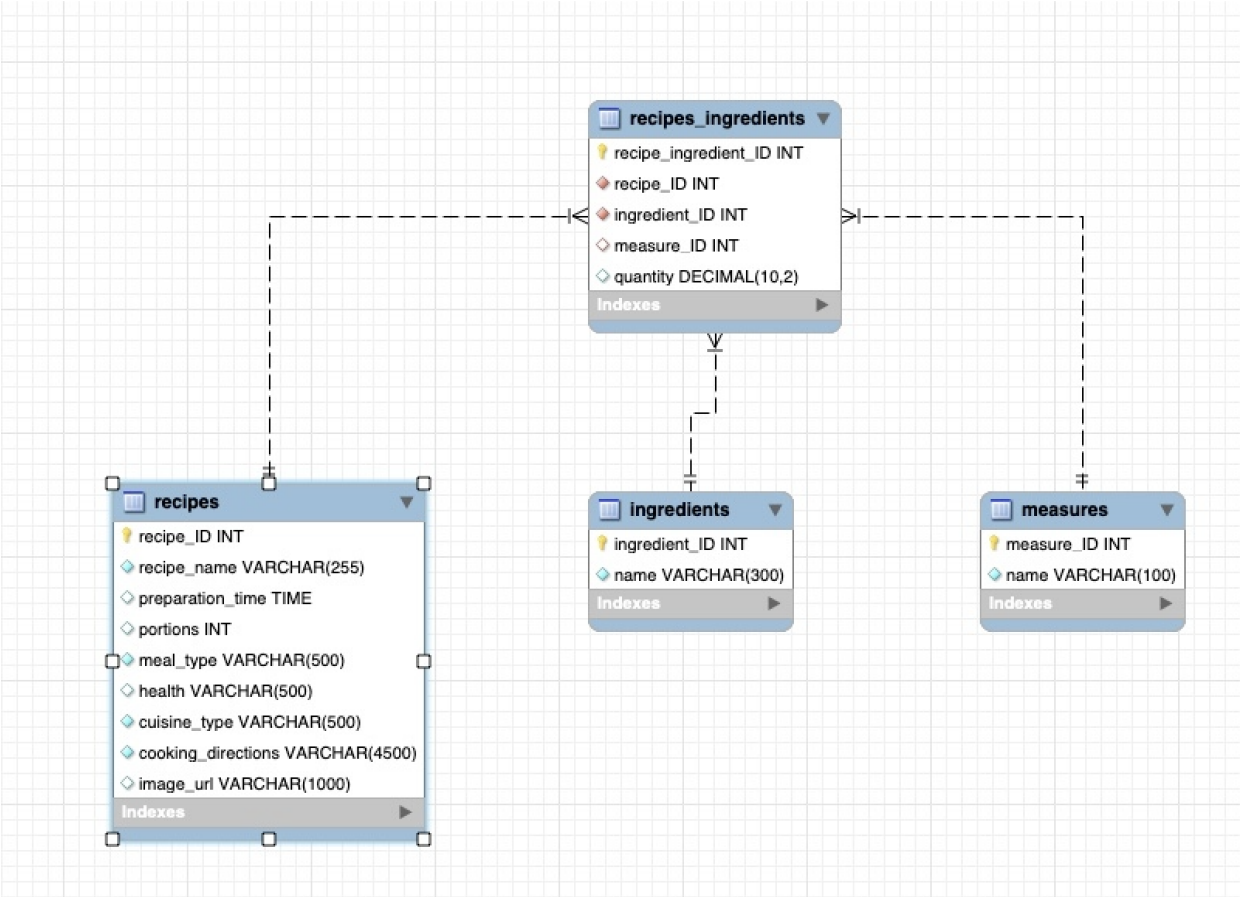



recipea_
send yours
Matches
Ribeye Steak with Potato Wedges & Asparagus
Meal type: Dinner
Cuisine: Argentinian


recipea_
send yours
Ribeye Steak with Potato Wedges & Asparagus
Meal type: Dinner
Cuisine: Argentinian


recipea_
send yours
Send yours!
Want more matches? Send us your recipe. *Required
*Required
Upload Picture
Send
Cuisine
Health
Meal type

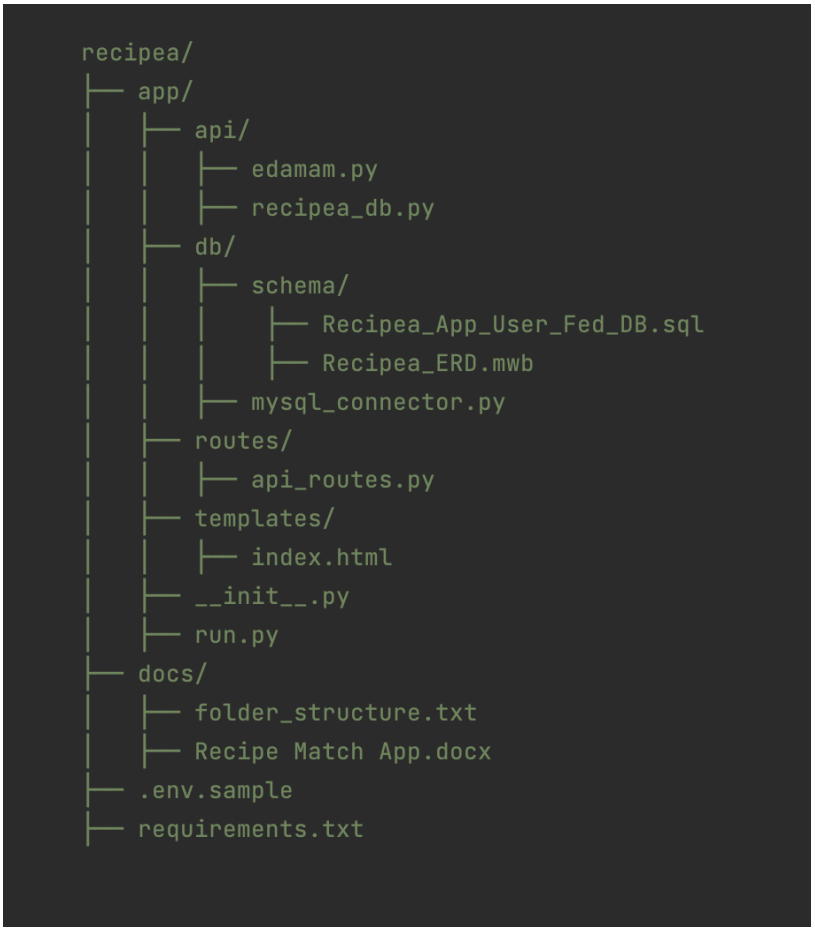
DATABASE SCHEMA:



FOLDER STRUCTURE:

We implemented a folder structure with the following goals in mind:

- Safeguarding sensitive data
- Maintaining clear and organised information
- Modularization of the code, making the code easier to maintain



TEST:

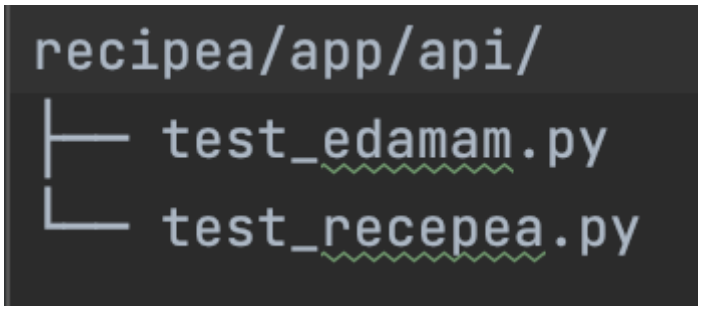
This document provides an overview of the tests implemented for the Python API project. The tests ensure the correctness and reliability of the API functionality and help maintain the expected behavior as the project evolves.

Test Framework

The test suite for this project is implemented using the unittest module, which is a built-in testing framework in Python. It provides a rich set of assertion methods and test discovery mechanisms.

Test Structure

The test suite is organized into two modules, each focusing on specific API features or components. The structure of the test suite is as follows:



README - instructions:

We built a very detailed README including all the information to run the app properly, test the features, and explaining the folder structure

ROADMAP :

Roadmap Week 1 - 5

Project detail

- Define the idea to develop
- Define the features
- Describe the feature's behaviour
- Pic the key features and functionalities to develop

Database Development

- Data structure and database requirements design
- Design the database schema
- Create the necessary tables and relationships

Endpoint Development

- - Design the API endpoints.
    - - Edaman API
        - search Edaman API
    - - APP Database
        - search_db
        - Add_recipe
        - Update_recipe (Raluca)
        - Delete_recipe
    - - Other routes - for the future or if we have time
        - Gather view from Edaman API view_recipe by ID
        - like_recipe
- Implement the necessary CRUD Functions (Create, Read, Update, Delete) operations.
    - Add_recipe (Create)

    - Search Edaman API (read)
    - Search_db (read)
    - Update_recipe (update)
    - Delete_recipe (delete)

- Integrate the database functionality into the endpoints
- Ensure proper error handling and validation

Console App Implementation

- Develop the user interface for the console app to
    - add recipe
- Integrate the console with the API

Back log tasks to do:

- Develop the user interface for the console app to
    - delete recipe
    - Update recipe
    - Search recipe

Testing and Quality Assurance

- Conduct unit tests for the individual components
- Perform integration tests to ensure seamless communication
- Conduct system tests to validate the overall functionality
- Identify and fix any bugs or issues

Documentation

- Document the project, including installation instructions and usage guidelines.
- Add the guidelines to the read me with the documentation
- Project design
    - Role definition
    - Features
    - Tech stack
    - Road Map

Roadmap for Future Implementation:

In addition to the initial stage, we have identified several other features that we would like to implement. Here is an overview of these upcoming stages:

User Management:

- Implement account creation, login, and logout functionality.
- Enable users to manage their profiles, including updating personal information and changing passwords.

Recipe Collection Management:

- Allow users to manage their recipe collections, including adding, editing, and deleting recipes.

Matches:

- Introduce a matching system where users can find recipes that align with their preferences and dietary restrictions.

User Interactions:

- Enable users to follow other users and receive updates on their activities.
- Implement a "like" feature for recipes, allowing users to show appreciation for recipes they enjoy.

Social Integration:

- Enhance user engagement by providing connections with like-minded food enthusiasts whose profiles match theirs.

Front-end and Back-end Integration:

- Integrate the front-end and back-end components of the application seamlessly for a cohesive user experience.

Performance and Scalability:

- Optimise the application for performance, ensuring fast response times and low latency.
- Prepare the system to handle a large number of users and recipes efficiently.

Scalable Infrastructure:

- Host the application on a scalable infrastructure that can handle fluctuations in traffic and accommodate future growth.

By following this roadmap, we aim to continuously enhance the app's functionality, user experience, and performance while meeting the needs of our growing user base.

.*Final Considerations*

We worked in a collaborative manner , and we supported each other throughout the project. We challenged ourselves trying to go further on the app development.
As developers we embraced the opportunity to learn and expand our skills.
Together, we overcame the obstacles and achieved our objectives.