Brief

What is required?

To create a web application for an online cookbook.

Users need to be able to search for recipes in a database and add new ones.

Who will use the application?

The application will be used by any person wishing to find a recipe or store their own.

What technologies will be used?

The application will require a database which can be created as a relational database using mySQL, or as a document-based database using MongoDB.

The front end will use HTML, CSS and JavaScript to create a good User Experience.

The logic will be written in Python using the Flask framework.

How will users interact with the application?

The application will be web based and accessible through all browsers on both desktop and mobile devices.

User stories

- As a vegan, I would like to be able to search through a list of recipes based on their type, in order to avoid going through all ingredient lists.
- As a cook, I would like to be able to store any recipes I create, in order to access and share with others.

Planning

Layout

Landing Page:

The home page of the application will present the user with three options: 'Search', 'Add a Recipe' and 'Browse All Recipes".

Search Page (may be unnecessary if search is implemented into home page):

The search page will offer the user a form to input their search requirements. They will be offered results based on their search and the ability to filter these results. UPDATE: to avoid multiple pages doing similar tasks, the search option is to be integrated into the home page and will take the user to a results page where they can filter with checkboxes.

Add recipe:

The Add page will allow the user to input a new recipe using a form. When a recipe has been submitted the user will be taken to that recipe's page.

Add category:

The user will also be able to add categories with a simple page or modal.

Recipe:

The recipe page will show the user their selected recipe. The user will then have the option to edit or delete the recipe from this page.

Edit:

The edit page will offer a form for the user to update the chosen recipe. After changes are made the user will be redirected back to the recipe page.

Database

Given the complexity and major differences between recipes and the need for multiple options, a document-based MongoDB database has been deemed the most suitable choice for the data store.

The database will contain two collections, Recipes and Categories. Categories will be fairly basic, just showing category name and type in each document. Recipes will require various levels of nested data to accommodate the potentially vast level of detail needed.

The recipes will need the following entries: Name, Ingredients (nested with multiple items), Method (again, nested due to multiple steps required), Serves (number of people), Author, Categories (multiple levels), Rating. Others may need to be added as the need presents itself.

Design

Wireframe mockups can be found as PDFs in the 'planning' directory.