

Hydronium: Dua Baig, Aidan Wong, Qianjun Zhou

SoftDev

POI: ArRESTed Development

2024-II-25

Time Spent: 2

TARGET SHIP DATE: 2024-12-17

DESIGN DOCUMENT (VERSION 2)

I. Description

This project is a multifunctional cooking website that contains recipes for every occasion. Users can search a collection of recipes drawn from an API and other users and contribute and save recipes from this collection. Users also have access to a holiday calendar, which is used to explore recipes for various holidays, and a random recipe generator for days they are unsure what to cook. This website can be viewed without an account.

A. Program Components

- a. User Accounts:
 - i. Creation of accounts and login/logout functionality
 - ii. Sessions
- b. Routes to different pages of the website using Flask and Python
- c. APIs:
 - i. Spoonacular: Provides food-related information (recipes, nutrition, etc)
 - ii. Calendarific: Generate holidays and related information for a calendar
 - iii. SearchAPI: Search for recipes for a given holiday
 - iv. Giphy API: Generating related gifs for each recipe
- d. SQLite3 Database: Stores data of the user and recipes
- e. Jinja Templates:
 - i. User dashboard:
 - 1. Logged-In State: Contains a profile image, a logout button, a button to a calendar, and a search bar that allows the user to search for recipes (Holidays and otherwise specified recipes)
 - 2. Logged Out State: Contains a search bar that allows the user to search for recipes, a login button, a calendar button, and a logout button
 - ii. User Settings: Allows the user to edit their username, and password
 - iii. Recipe Viewer: Displays recipe with corresponding image/gif and the user who created it (if it is a user-contributed recipe)

- iv. Search Result Page: Displays the results of a user search for recipes
- v. Holiday Calendar Page: Displays all the holidays from an API and allows the user to click the holiday to be brought to a page with related recipes
 - I. Holiday Recipe Results Page: Displays the related recipes to a holiday

B. Program APIs

- a. Spoonacular: Provides food-related information (recipes, nutrition, etc)
- b. Calendarific: Generate holidays and related information for a calendar
- c. SearchAPI: Search for recipes for a given holiday
- d. GiphyAPI: Generating related gifs for each recipe

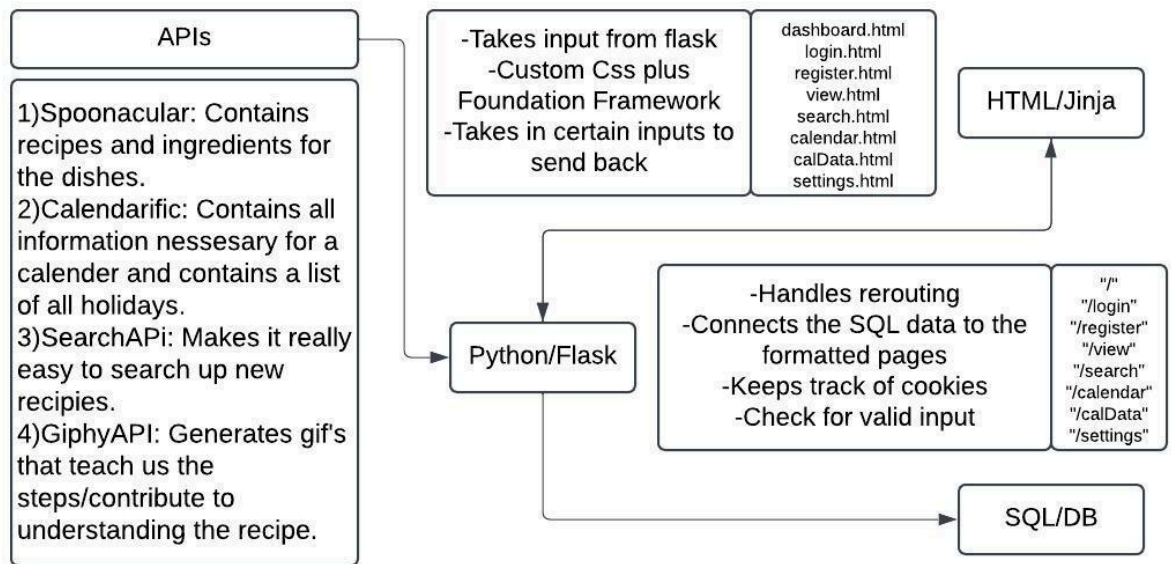
C. Frontend Framework: Foundation

- 1. Why Foundation?
 - a. Easy-to-use and easy-to-follow tutorials
 - b. Vast amount of CSS components and JS ones that will make things look and feel good
- 2. How Foundation?
 - a. Grid System: Structure layout of pages like dashboard, recipe pages, and search results (Website will be able to adapt to size changes)
 - b. Pre-designed Components: Buttons, Forms, etc
 - c. JS Plugins: Modal Windows for confirmation messages

D. Program Component Connections

- a. User accounts: Give access to contributing and favoriting recipes (Created through registration, and can be logged in to or out of).
- b. Routes + Python: Routes allow users to traverse the website. They connect the different pages (HTML documents) of the website. Python also interacts with APIs and the database.
- c. Database: Stores information related to the user (ID, Password, etc)
 - i. One of the main factors for information exchange between components (has all the data)
- d. APIs: Provide information that is stored in the database and displayed on the HTML pages
- e. Templates: Allow for dynamic web pages (as they need to update when new stories are created/edited)

E. Component Map



F. Database Organization

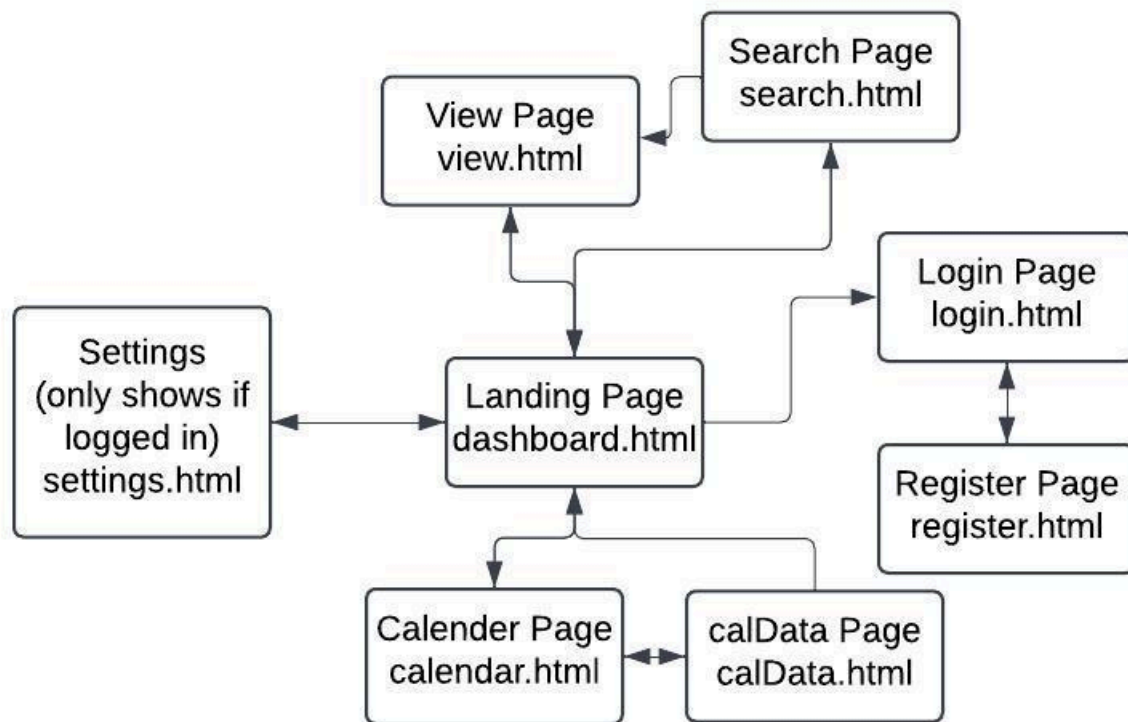
Table to store userData

userName (string)	userPassword (string)
----------------------	--------------------------

- a. User Table
 - i. Username (PK)
 - ii. Password

Note: PK for primary key (each row value must be unique)

G. Site Map + Descriptions



1. Landing page/Dashboard (/): Dashboard
2. Settings page (/settings): Allows user to change username, password, and profile picture
3. Login (/login): Allows the user to sign in to an account
4. Registration Page (/register): Allows the user to register for an account and logs the user in upon successful account creation
5. Recipe Viewer (/view): Displays recipe information with additional information like nutrition facts, holiday if applicable, etc
6. Search result page (/search): Shows the result of a recipe search by a user
 - a. Search parameters (as dropdowns): diet type (vegetarian), intolerances (no gluten), protein, calories, fat, carbs
7. Calendar Page (/calendar): Displays a calendar with holidays filled in on their respective dates
8. Calendar Data Page (/calData/<name>): Displays information on a specified holiday after clicking it on the calendar page, and also includes a link to a recipe for that day.
- 9.

H. Task Breakdown

- I. Dua Baig: Frontend
 - a. Create HTML pages with Jinja templating - Includes any forms required for logging in/signing up
 - b. Design site style with CSS and front-end framework (Foundation)
2. Aidan Wong: Backend (Database and API functions)
 - a. Create SQLite3 database schema
 - b. Work on database interaction modules (python) → General operations (ex. Inserting data, creating tables, etc)
 - c. Work on API interaction modules for Spoonacular and Calenderific (python) → General operations (ex. Retrieving data based on function input, etc)
3. Qianjun Zhou: Backend (Python and API functions)
 - a. Routing and logic between pages + User session management
 - b. Linking database and API functions with site features
 - c. Work on API interaction modules for SearchAPI and GiphyAPI (python) → General operations (ex. Retrieving data based on function input, etc)