Team Mango: Aryaman Goenka, Sadid Ethun, Haotian Gan
SoftDev
P01 -- Design Doc
12/8/21
Time Spent: 5 hours

Concept: Food Network Website

- Discover new recipes and restaurants
    - Recipe search bar (**Spoonacular API**)
    - Save **Spoonacular API**'s recipes to your personal library, your "cookbook".
    - Each recipe comes with ingredient nutritional information hover-overs (**USDA, and OpenFoodFacts API**)
        - Detailed Nutritional Breakdown: **(USDA API)**
        - Images for recipe ingredients are provided by the **OpenFoodFacts API**
    - Social Media-esque site: your favorite recipes are publicly viewable, you can make posts about what you've made recently and receive likes and comments. Each user has a page similar to Instagram's setup.
    - Restaurant suggestions based on cuisine or location using the **Documenu API** (Based off of recipes in your library and your search history)

All APIs Used:

- **Spoonacular API (Recipe search)**
- **OpenFoodFacts (Images for ingredients)**
- **USDA API (Nutritional breakdown for ingredients)**
- **Documenu (Used for restaurant recommendation system and restaurant search)**

Site Components

- Login
    - Verify user exists using flask form or confirm new username and password
    - Give cookies to keep a session running
    - Client-side verification that username/password meet constraints
    - After, redirect to Main/Landing Page
- Main Page
    - Nav Bar
        - Search page link
        - Personal user posts link
        - Link to page with user's favorited recipes, restaurants, and posts
        - Personal recipe book link
    - A feed with user posts (Will look the same as Instagram)
- Personal User Page
        - Each user has a page that looks like an Instagram page
        - Three sections:
            - User Posts

            - Posts require an image and a description, just like on Instagram.
            - Posts have a comment section.
            - Posts are sortable by date created, cuisine, and full-text searchable.
            - Posts can link to a recipe in the user's recipe book
            - Posts can be "favorited" by other users (heart icon)
        - User recipe book
            - The user's "cookbook". Everything that the user has deliberately chosen to save.
            - Recipes that the user has created themselves
        - Favorited posts, restaurants, and recipes
            - A list of the user's favorited posts and restaurants
- Search Page (Three sections)
    - Find Recipes (Spoonacular API)
        - Search by:
            - Diet Restrictions
            - Cuisine
            - Arbitrary word query
    - Find Food Facts (Spoonacular, USDA)
        - Search by:
            - Keyword search string
        - Display nutritional info as well as the image of the food item
    - Find Restaurants (Documenu API)
        - Restaurants are automatically suggested to the user based on their saved, searched and posted recipes based on the cuisine.
        - Manual search functionality
            - Cuisine
            - Menu Item
            - Location
                - Key words and cities

DBs (SQLite3)

- User DB
    - One Entry per User
    - Values: Username, Password
- User Personal Information
    - Foreign Keys Referencing to User DB
    - DB for user personal info
- User's favorite restaurants, recipes, and posts

Database Organization

User DB

UserID (INTEGER PRIMARY KEY)	Username (TEXT NOT NULL)	Password (TEXT NOT NULL)
0	user	pass
1	user2	pass

User Personal Information

Foreign Key (a reference to user DB UserID)	Preferred Cuisine	Dietary Restrictions	Display Name	Personal Page Description
0 (user)	Italian	Pescatarian	FeedMeCheese	Avid chef interested in Italian cuisine Columbia 22’

User Favorite Restaurants

Foreign Key (a reference to user DB userID)	Cuisine	Location	Other Documenu API data…
0 (user)	Fusion	345 Chambers St, Stuyvesant High School	…

User Favorite Recipes

Foreign Key (a reference to user DB userID)	Cuisine	Description	Ingredients	Instructions
0 (user)	Indian	Delicious Pasta	[array of ingredients]	[array of instructions]

User Favorite Posts

Foreign Key (a reference to user DB userID)	Foreign Key (a reference to Post DB’s postID)
0 (user)	1239973

User Posts:

Foreign Key (a reference to user DB's userID)	postID	Image Link	Description
0	12321398	https://imgur/…	Made some apple pie :)

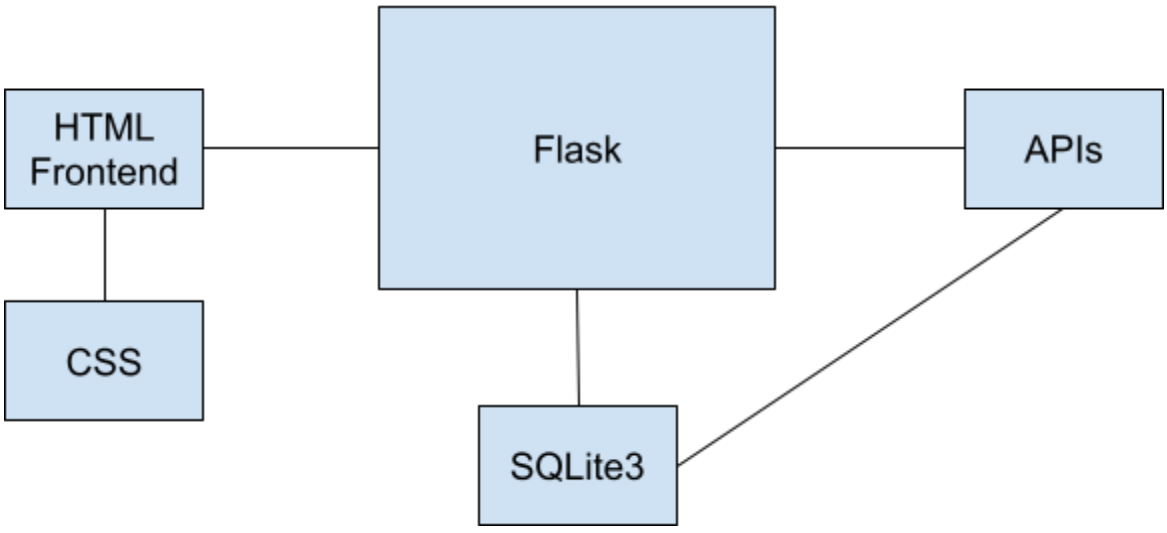
Post Comments:

Foreign Key (a reference to user DB's user ID) (The commenter)	Foreign Key (postID) (the post being commented on)	Comment
0	12321398	This is such a great recipe!

Code Components

1. Flask
    a. Hosts and routes the website.
    b. Interacts with APIs on the backend, allowing for the functionality of the API.
2. Sqlite3
    a. Stores results from API to minimize calls to API (storing data on the recipes users have favorited)
    b. Stores User Data
3. CSS/JS (Bootstrap)
    a. Used for styling and interactive elements on the frontend.

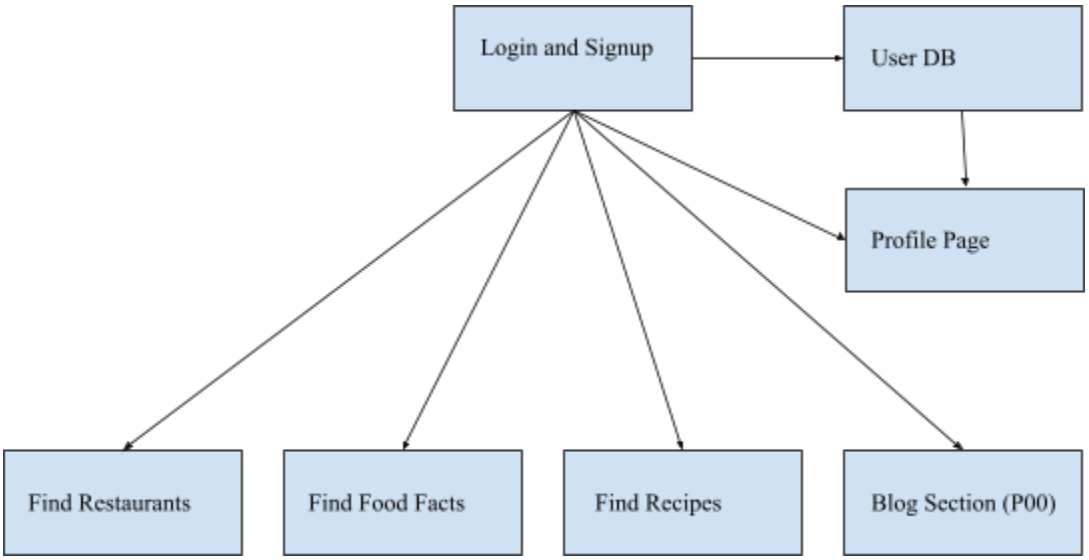
Component Map



Roles

- Aryaman Goenka (PM): DB, Flask Routing, Find food facts function
- Sadid Ethun: Html, Bootstrap, Find restaurants function
- Haotian Gan: User pages, restaurant suggestions, recipe search page

Site Map:



Routes

Every page comes with a Navbar

- / - Home page
    - Post Feed
    - Suggested users to check out (Randomly picked)
- /recipes/<search_query> - Search Results Page
    - Results that appear after submitting a search query to Spoonacular's API for recipes
- /recipes/<spoonacular_recipe_id> - Recipe Page
    - Displays information on selected recipe
- /facts/<search_query> - Search Results Page
    - Results that appear after submitting a search query to OpenFoodFactsAPI and USDA API for more information about an ingredient
- /facts/<ingredient_item_facts> - Ingredient info page
    - Displays information on selected ingredient item
- /restaurants/<search_query> - Search Results Page
    - Results that appear after submitting a search query for restaurants
- /restaurants/<restaurant> - Restaurant Page

    - Displays information on selected restaurant
- /user/<username>
    - User page
- /user/<username>/favorites
    - User's favorite recipes, restaurants, and posts.
    - **Send a post request to this endpoint to create and delete entries**
- /user/<username>/cookbook
    - User's personal cookbook
    - **Send a post request to this endpoint to create and delete recipes in the cookbook**
- /user/<username>/posts
    - List of user's posts
    - **Send a post request to this endpoint to create, update, and delete posts**
- /posts/<search string>
    - Results from full-text search on the user post database.
- /posts/<postID>
    - Information on the specific post
    - Edit the post if it is the user's post
    - Comment on the post
    - **Send a post request to this endpoint delete and edit comments**

Future features (if time allows)

- Ability for users to create their own custom recipes.

Target Ship Date - 1/03/22