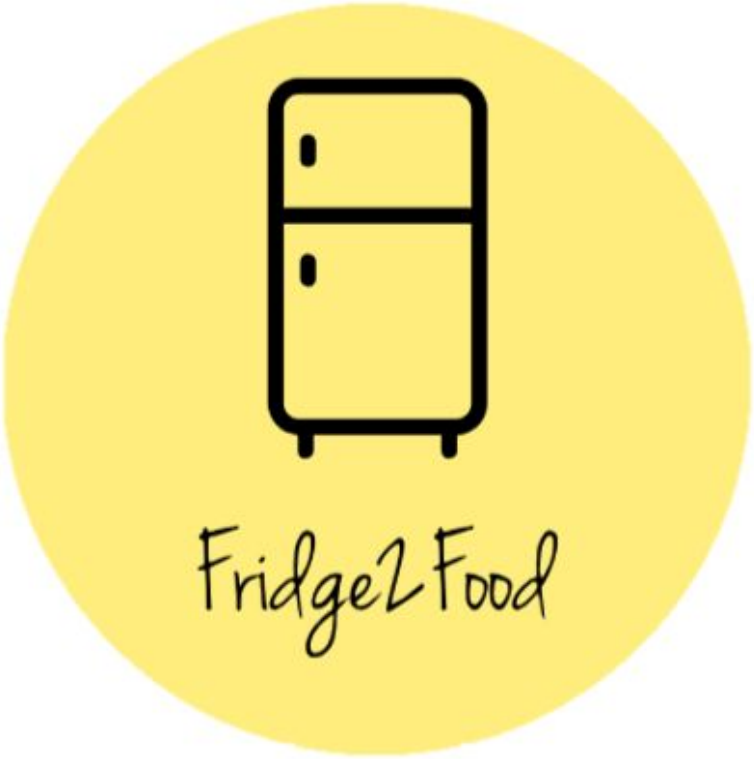


CS 30700

Design Document

1.28.2019

Team 18:

Vivian Guo, Matt Retoff, Aditya Sankaran, Zach Bridwell, Thomas Schroeder, Anthony Kiselev

Index

Design Document **1**

Index **2**

Purpose **3**

Functional Requirements 3

User Account 3

Recipes 3

Ingredients 4

Comments 4

Sorting 4

Accessibility 5

Non Functional Requirements: 5

Architecture and Performance 5

Usability 5

Hosting/Deployment 5

Design Outline: **6**

High Level Overview 6

Activity Diagram 7

Sequence of Events 8

Making an account 8

Input Ingredients 9

Search for Recipes 10

Design Issues: **11**

Functional Issues: 11

Non-Functional Issues: 12

Design Detail: **14**

Class Diagram: 14

Description of Data Classes 15

User 15

Recipe 15

Purpose

Every day, families across America are faced with the question: “What is for dinner?”. Many people have a wide variety of seemingly random ingredients in their refrigerators, but nothing concrete. People can rely on recipes from the internet, but they often do not have all of the needed ingredients. Even though there are websites that provide recipes based on ingredients, the websites do not cater to specific allergies, or diet and caloric needs.

Our project hopes to remedy this problem by allowing users to input a list of ingredients they have on hand, and provide a list of recipes which the users can use to (hopefully) produce something edible, through creating an account on a website where they can personalize their criteria. Their personalizations will be saved to their account so the next time they would like to find a recipe, their allergies and information will already be in the system. While there are existing solutions to this problem, many of them do not allow a lot of personalization to input allergies and other limitations that you don't want showing up in your recipe list.

Functional Requirements

1. **User Account**

   As a user,

   a. I would like to be able to create an account on Fridge2Food.
   b. I would like to be able to login and manage my Fridge2Food account.
   c. I would like my password to be reset if I forget it.
   d. I would like to create a profile picture.
   e. I would like to be able to change my username.
   f. I would like to be able to input my allergies as a list of ingredients in the database.
   g. I would like to be able to favorite recipes so that I may find them again.

2. **Recipes**

   As a user,

   a. I would like to be able to view recipes.
   b. I would like to see a list of ingredients for each recipe.
   c. I would like to see instructions for each recipe.
   d. I would like to see the total number of recipes I can make with the ingredients I have.
   e. I would like to search through the recipes I can make using a search bar.
   f. I would like to have recipes that I have every ingredient for at the top of the list and recipes where I’m missing a couple ingredients near the bottom.

g. I would like to be able to view ingredient substitutions for recipes.
h. I would like to be able to input my own recipes.
i. I would like to be able to access my user created recipes.
j. I would like to insert pictures of recipes I have created.
k. I would like to click on the recipe name to see the full recipe.
l. If time allows, I would like to see the utensils and equipments needed for the recipe.
m. If time allows, I would like to know the nutrition facts about the recipes.

3. Ingredients

As a user,

a. I would like to be able to input the ingredients I currently have so that I can have recipes catered to my needs.
b. I would like to remove all currently selected ingredients.
c. I would like to add ingredients to the database.
d. I would like to see my most frequent ingredients appear in a dropdown when I click on the search bar.
e. I would like to add ingredients by clicking on options.
f. I would like to add ingredients by typing.
g. I would like to see the ingredients I have inputted.
h. I would like to know how many ingredients I have inputted.
i. I would like to see pictures of the ingredients when clicking on options.
j. If time allows, I would like to know the price of an ingredient I do not have at a store nearby.

4. Comments

As a user,

a. I would like to see all comments made on a recipe.
b. I would like to reply and like other user's comments.
c. I would like to rate and comment on recipes I have created so recipes can be reviewed.
d. As a user, I would like to receive comments on personal recipes so that I may improve them.

5. Sorting

As a user,

a. I would like to be able to sort recipes by cooking time, calories, rating, etc.
b. I would also like to sort recipes by meal type (ex. Breakfast, Dinner, Dessert, etc).
c. I would like to input dietary restrictions (ex. Vegan, vegetarian, pescetarian, etc).

d. If time allows, I would like to specify the type of cuisine I would like to cook.

6. **Accessibility**

   As a user,

   a. I would like to be able to access Fridge2Food from all of my devices on a mobile-friendly website.
   b. I would like to be able to allow users to fork my recipes so they can make personal alterations
   c. I would like to use the site without logging in.
   d. I would like to be able to convert measurements.
   e. I would like to be able to convert between Fahrenheit and Celsius
   f. I would like to see my most popular recipes.
   g. I would like to see the website's most popular recipes.

Non Functional Requirements:

1. **Architecture and Performance**

   The front end web application will be developed using React JS, and the backend will be a combination of Flask and Firebase. Our list of known recipes will be gathered using an API.

2. **Usability**

   The website should be visually appealing, be easily accessible, and categorized as a food and drink application. For us, this means making the website intuitive and simple enough that anyone can come to it and not be confused while using it. We will also want our website to be accessible from any platform and browser so as to reach the largest audience possible.

3. **Hosting/Deployment**

   For our website we will be looking to use Firebase to host the website. Firebase will allow us to easily deploy updates and to handle user management more easily than it would be otherwise. This ease will allow us to focus on other aspects of the project.

Design Outline:

High Level Overview

This project will be a web application that allows users to input ingredients and find recipes based on the ingredients based on their dietary needs and preferences. This application will use the client-server model where many clients will be connected to a server. The clients will send requests to the server which will then query the database for the correct information. The server will send appropriate data back to the client. When the client sends information to the server, the server will write the data to the database.

Client:

The client will be the the point where users interact with the program. It will display relevant information to the user and allow them to use all implemented features. The client will interact with the server primarily through a REST API. The client receives responses which update the interface.

Server:

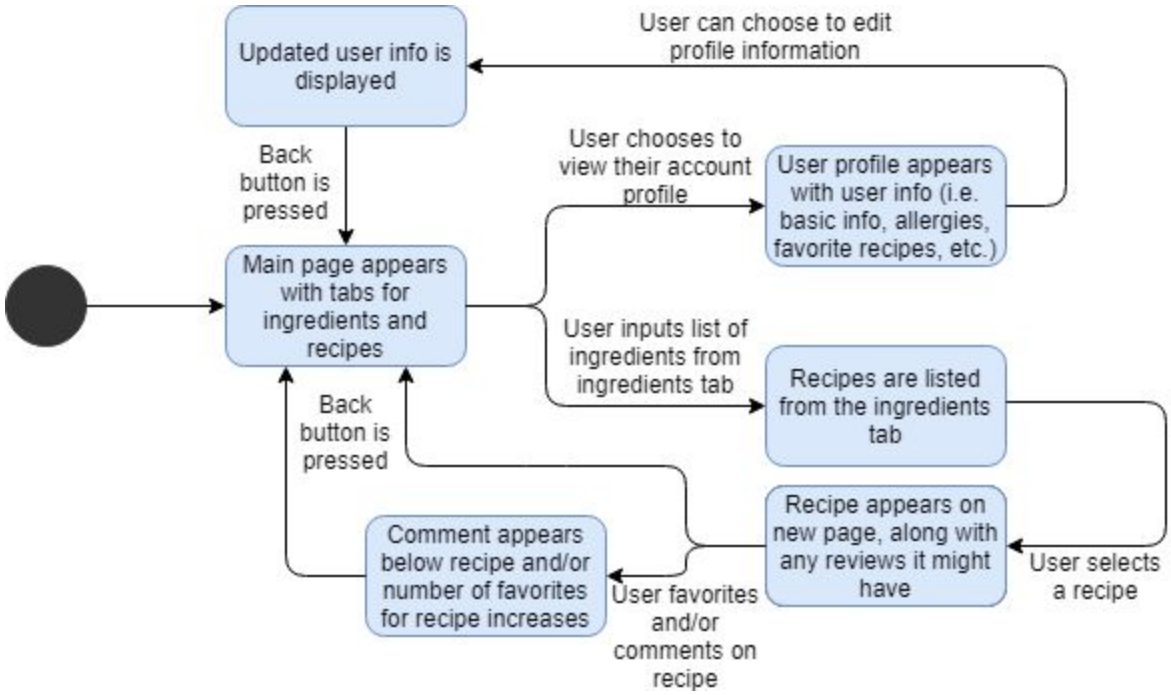
The server will handle all interactions between the client and the database. It will also handle all heavy algorithmic work involved in the application, for example, finding recipes that the user can make with the ingredients they have. The server will interact with the database through SQL queries, and with the client by responding to HTTP get/post/put requests.

Database:

The database will store all application data, such as user information, recipe information, and ingredients. The database interacts with the server by executing and responding to queries.

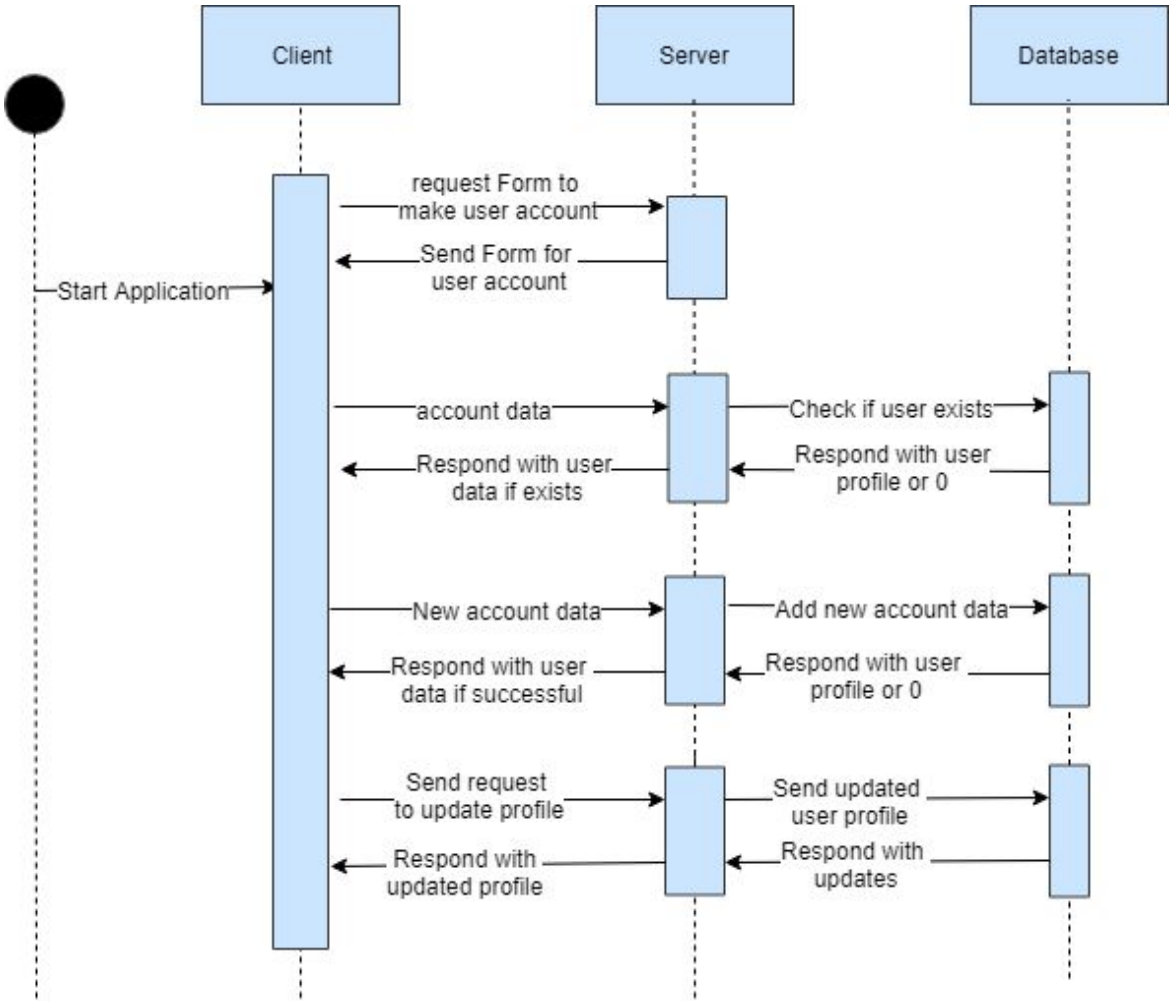
Activity Diagram

This diagram shows the different states that the application can be in. When the user opens the website, the main UI is present. The user is presented with a tab for ingredients and a tab for recipes. Once the user inputs all of the desired ingredients, the recipe tab will display all possible recipes. After clicking on a recipe, the user is redirected to a page that displays the full recipe, along with any comments that the recipe might have. If the user wants to create a profile or login, they can click on the login button on the main page. Once logged in, the user can view and edit their profile as necessary by adding any allergies or dietary restrictions they may have so that recipes conflicting with those restrictions won't be shown. The user can also see their favorite recipes from their profile.

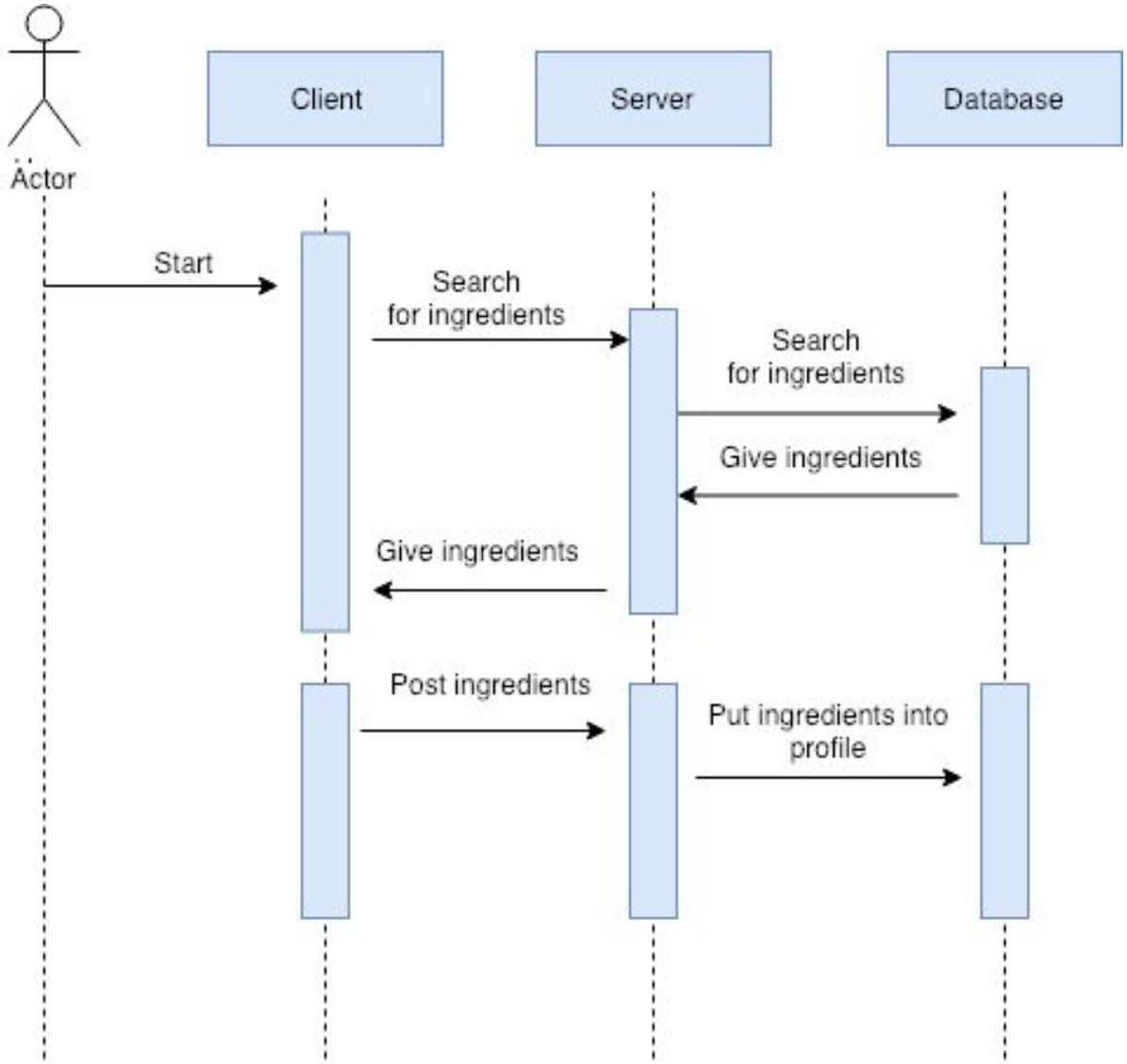


Sequence of Events

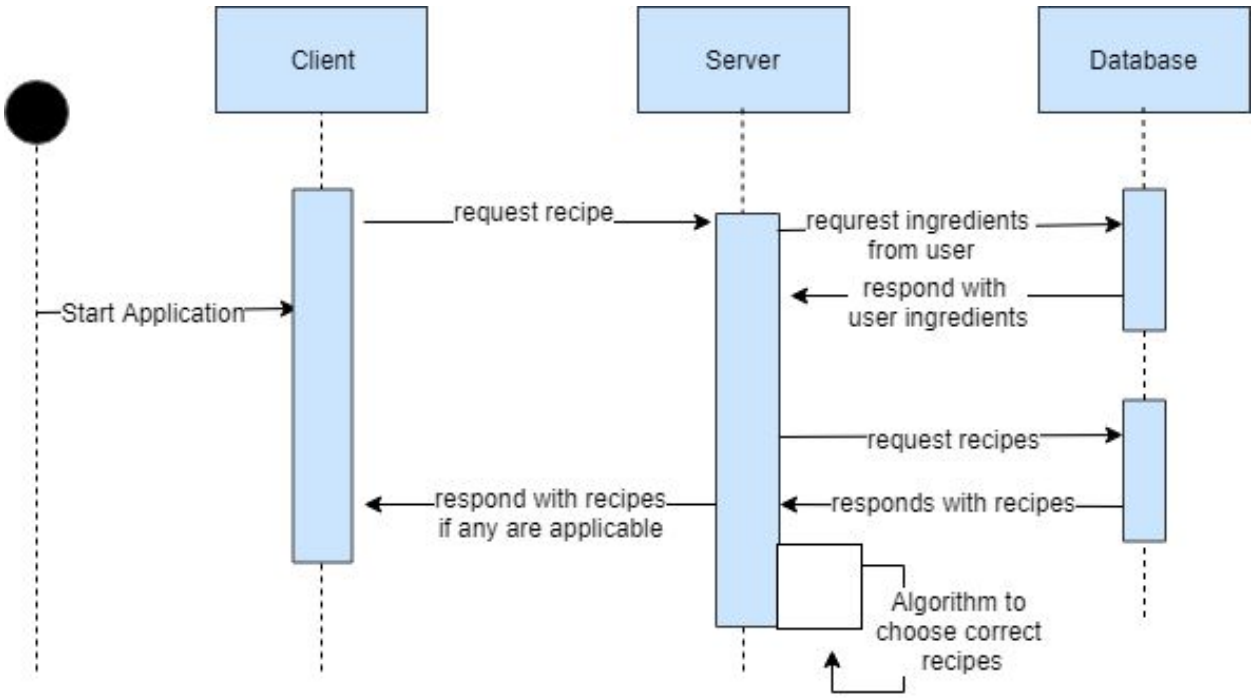
Making an account



Input Ingredients



Search for Recipes



Design Issues:

Functional Issues:

1. What information is needed for making an account?
    - Option 1: Username
    - Option 2: Username and Password
    - Option 3: Username, password, and email address

    Choice 3

    To set up an account on Fridge2Food a username, password, and email address will be needed. This is necessary because it allows users to have their own unique account with some level of protection through email verification. The email not only provides protection of the account, but it also allows us to let the user receive an email in the event that they wish to reset a password or if they even forget their password. The email will let us communicate with the user when their account is out of reach. Further protection isn't needed for Fridge2Food because the information held within the account isn't of the highest degree thankfully.

2. Is an account needed to access the website?
    - Option 1: Allow guest users
    - Option 2: Require an account to use the website

    Choice 1

    We will allow guest users as to create an environment that is welcoming to everyone that comes to the website. Allowing guest users will make it so that the website is more accessible to everyone while giving the choice to also sign up for an account. The account will provide people more options when it comes to customizing their searches and profiles that won't be available to those without an account.

3. How will ingredients be inputted by the user?
    - Option 1: Input ingredients manually by typing them out.
    - Option 2: Using a checklist to select ingredients.
    - Option 3: Allowing users to type ingredients and select from a checklist.
    - Option 4: Allowing users to search for ingredients from the database with the option to add new ingredients if they are not found in the database.

    Choice 4

    We can not be sure that our database contains every possible ingredient. Therefore, we want to allow users to add new ingredients to the database as needed for a smoother user

experience. Additionally, this will decrease the possibility of users misspelling ingredients to prevent multiple spellings of the same ingredient.

4. How should recipes be displayed to the user?
    - Option 1: All recipes are displayed on one page
    - Option 2: Allowing users to see all of the possible recipe titles from the database, with the option to add new recipes if they are not found in the database.

    Choice 2

    Because the database will contain many of recipes. Displaying all possible recipes with the details all on one page would be overwhelming. To help clarify and provide better organization, allowing only the recipe titles to be initially displayed would help decrease confusion and increase clarity for the user. Additionally, this will allow users to add their own recipes to share.

5. What information should be displayed for each recipe?
    - Option 1: The ingredients and instructions
    - Option 2: The ingredients, instructions, and cooking time
    - Option 3: The ingredients, instructions, ingredient substitutions, calories, ratings
    - Option 4: The ingredients, instruction, cooking time, calories, ratings, dietary restrictions,

    Choice 4

    This choice will allow users to look back at certain characteristics they may have inputted or wanted when searching for recipes. Additionally, ingredients and instructions must be displayed so the user can produce the final product. However, providing the user with ingredient substitutions with the recipe may cause additional confusion. Instead of providing the ingredient substitutions with the recipe, the ingredient substitutions should come earlier when inputting ingredients to allow all possible recipes to be displayed.

Non-Functional Issues:

1. What frontend frameworks/libraries should we use?
    - Option 1: React JS and Bootstrap
    - Option 2: Angular JS
    - Option 3: Node JS

    Choice 1

    We’re choosing to use React JS and Bootstrap for a few reasons. Most of us have at least limited experience with React so we have a starting point which is better than none at all. On top of that, React and its learning curve is lower than those of Angular and Node

which is good since we have to learn most of it as we work on it. Bootstrap will also be used in order to make the website look much nicer than with React alone.

2. What backend language should we use?
    - Option 1: Flask
    - Option 2: SQL
    - Option 3: Django

    Choice 1

    We are going to use Flask since it is more easier to learn than Django. Flask has the most lenient learning curve among all the options.

3. What API should we use to access food recipes if we so choose to?
    - Option 1: Spoonacular
    - Option 2: Food2Fork
    - Option 3: Edamam
    - Option 4: BigOven
    - Option 5: Yummly

    Choice 1

    This choice will allow us to access thousands of ingredients, recipes, food products, and menu items at a reasonable price along with up to 5000 requests per day. Other options either have very limited access or are very costly.

4. How are we going to host our backend services?
    - Option 1: Firebase
    - Option 2: Heroku

    Choice 1

    We are going to use Firebase to implement the login aspect and creation of a user account. It will be fairly easy using Firebase to accomplish this.

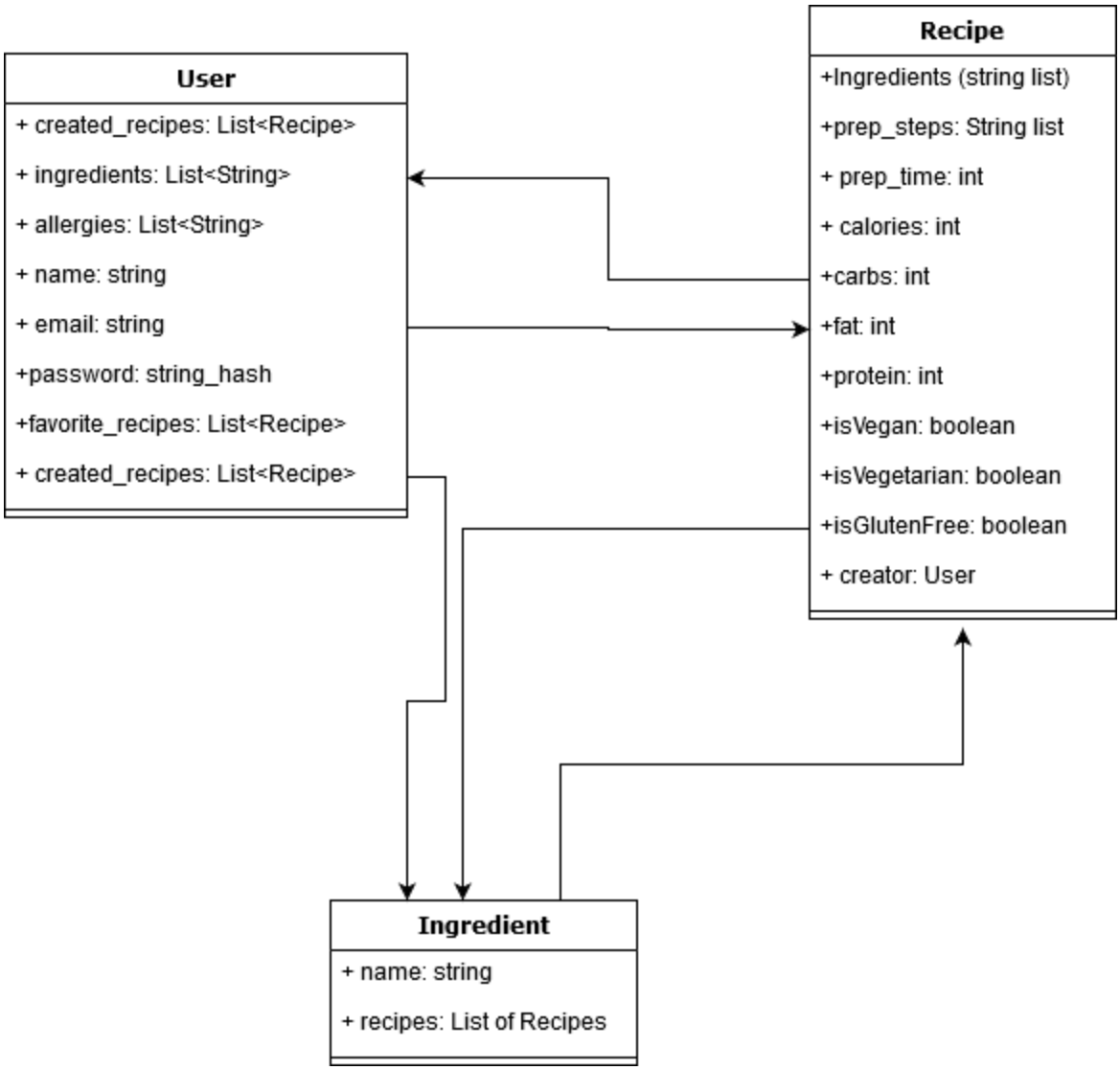
5. What type of database is most appropriate?
    - Option 1: Firebase
    - Option 2: MySQL

    Choice 2

    We are going to use MySQL as we believe it is the most appropriate for a project such as this. MySQL will allow us to easily add and get rid of ingredients and recipes which is one of the most important functions that we will have to perform. MySQL also lets us search for ingredients and recipes easily with its built in functions which makes MySQL easy and quick for us to use for such a website.

Design Detail:

Class Diagram:



Description of Data Classes

Our classes are generally just stores of data. They don't have functions or methods that can be performed on them, as all modification of data will be handled by backend server by modifying the database.

User

This class contains all information about a given user, including their name, password, email, and created recipes. This class is created by the server when the user registers for the site. It is updated whenever the user performs any interaction with the site.

Recipe

The recipe class contains all information about each recipe, as well as a list of ingredients.

Ingredient

The ingredient class contains simply the name of each ingredient, and a list of recipes that use it.