CS480 Final Project Requirements Specification

**Task Force:** Nathaniel Krueper, Benjamin Krueper, Divya Ramayampet, Joshua Camacho, Thomas Vermeersch

**Client:** Prof. Edwin Rodriguez

**Introduction:**

For the final project of the Summer ‘17 Cal Poly Pomona course CS480, the instructor (and client) has requested a recipe sharing application that suggests recipes to make based on available ingredients.

**Background:**

Home cooking is a big part of many people’s lives, but because they only have a limited amount of ingredients at any one time, making certain dishes often requires an inconvenient trip to the supermarket. They may just want to cook something with what they have available, or just lack inspiration for what to cook. In these situations, software that determines what dishes can be prepared with available ingredients would be useful to help people decide what to make.

**Project Description:**

This project is a service that users can log on to in order to share, store, and look up cooking recipes. Users will have the ability to submit their own recipes to the server by providing the dish’s name and a list of ingredients needed to make it. They could optionally provide instructions on how to prepare the dish and give their dishes various tags to help other users find them.

Users will also have the ability to view recipes made by other users, and are able to search for recipes based on:

* Ingredients used
* The dish’s meal category
* Style
* What is in the user’s personal recipe list
* What is in the user’s “favorite” list

Users can input a list of ingredients they have available, and the service will provide them with a list of possible dishes they can make with those ingredients. This list can be filtered and sorted based on the above parameters.

If possible and time permitting, the following features can also be implemented:

* User rating system for recipes, a numerical voting system
* Keeps track of recently popular (trending) recipes and a corresponding search parameter.
* Additional parameters to store and search for, such as popularity and calorie count.

**Specific Requirements:**

Primary:

* Users can log in to the service.
* Service has a database of dishes as well as the recipes for that dish
* Service must initially start with at least 1,000 recipes available.
* Users can submit dishes for other users to access, which includes the name, recipe,and optionally, the preparation instructions, meal category, and style of the dish.
* Users can input a list of ingredients and how much they have of them to receive a list of what dishes in the system’s database can be made with those ingredients.
* Users can search the services’ collection of dishes and well as filter and sort the results of their search based on the dish’s style, meal category, and what ingredients are used to make it.
* The service keeps track of how much of each ingredient is required for each recipe and how much of an ingredient the user has.

Secondary:

* Users can view a list of recipes they looked at recently.
* Users can rate each dish based on how much they liked it.
* Users can view the calorie count for each dish and its style and meal category, if provided.
* Users can view and submit instructions for how to prepare the dish
* Users can view a list of the dishes they have submitted.
* Ease of use requirements:
  + The user’s username and password can be remembered if the user accesses the service from the same device.
  + Ingredient names should be auto-filled when the user is typing them in to speed up the process of inputting ingredients.

Technical:

* Service must be runnable from the client’s phone

**Non Functional Requirements:**

The service must be quick and easy to access and use. The user may be accessing this service 1-4 times per day with the intent of quickly getting an idea on what to make.

* Connecting to the service should take less than 4 seconds, even on slow networks.
* Transitioning between the menus should be very fast. The speed at which the user traverses the service should not be limited by transition animations or loading times.
* The service needs to be accessible at all times of the day. Maintenance should be rare and brief.

**Assumptions:**

There exists public APIs that our service can use to get raw data for the recipes.

**Deliver By:** August 31, 2017

**Project Constraints:**

Task force cannot enlist the direct help of other individuals in the creation of this project

Progress towards completion of the project must be made every day.

**What will not be implemented:**

This project is not a social media service. Users will not be notified when other users submit or rate dishes. Additionally, users will not add or have followers or post any kind of text based messages.

A terms of service contract, moderators, and rule enforcement system to stop users from submitting inappropriate content (i.e. “troll” recipes) will not be implemented.

**Communication Plan:**

The team will be using the SCRUM development model. Daily meetings between all developers will occur on weekdays at 9:00 a.m.,except for Tuesdays and Thursdays, where they will occur when the client’s class ends. All developers will report on what they did, what problems they encountered, and what they plan on doing for that day. Anything blocking the team member from completing a task will also be mentioned and a solution can be identified. A report will be made to summarize each meeting and will be made available to all developers and the client. The team will also use weekly sprint planning meetings which will divide out the project’s tasks into small, manageable pieces and ensure everyone is under an equal workload.

**Challenges:**

This project requires that the service is always available to users, and that it remembers all the users and recipes they submit. As such, building a web service with a database will be required. The team has some experience in this field, but will have to learn how to do many things on the fly. The project must also be completed in only a month, which will put stress on the developers.

Obtaining the initial 1,000 recipes for the service is also an issue, as it would take too much effort for the developers to provide them themselves. A third-party database will have to be used to obtain the required information, and that information will need to be converted into a usable format.

**Glossary:**

* Dish: a meal that can be made with certain ingredients
* Recipe: a list of ingredients required to make a dish
* Meal Category: when in the day the dish is usually consumed (breakfast, dinner, snack…)
* Style: the culture that the dish is most often associated with (Mexican, Italian…)