# Search and Serve Group Design

Alex Wilcox - Product Owner

Damien Danelson

Elizabeth Hurteau

Ariel Sievers

Connor Wendolek

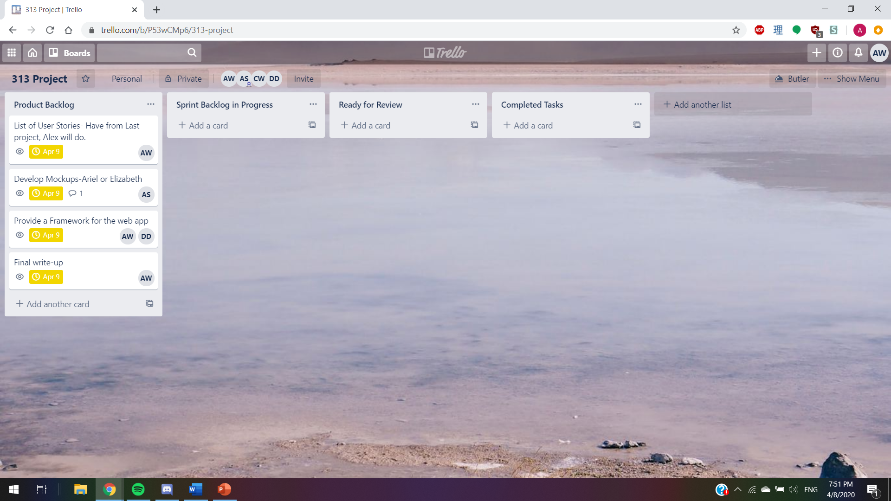
Project Description

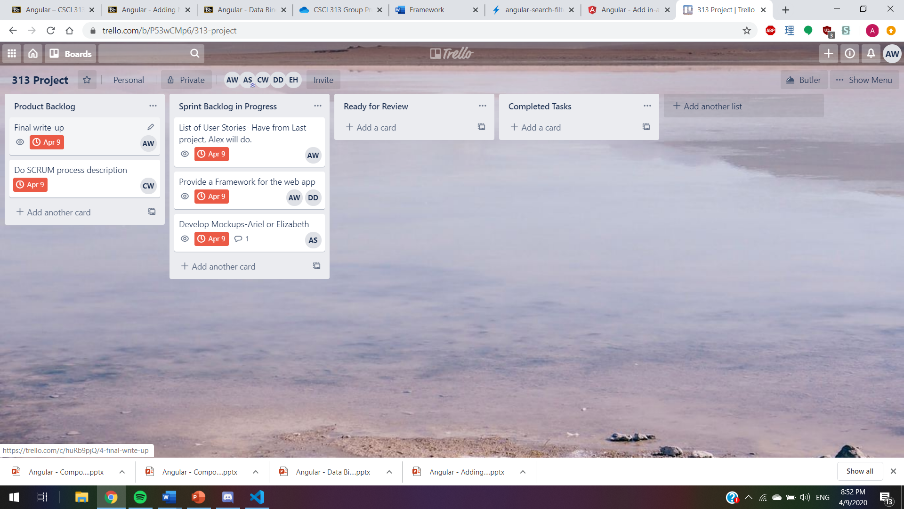
The application we are planning on making for this project is a simple and easy-to-use tool that allows users to find recipes for meals or desserts based on the ingredients they have. Developed as a web application with Angular and Bootstrap, the application should look and feel nice, as well as be mobile friendly. This way, a user can comfortably access the app’s features from their phone or tablet, making cooking in the kitchen even easier. Ingredients that the user already has will be provided as input as they search for a recipe. This information will then be queried to look for a recipe that does not require too many ingredients that the user does not have. However, if an item is needed, an option to buy the food online may be provided.

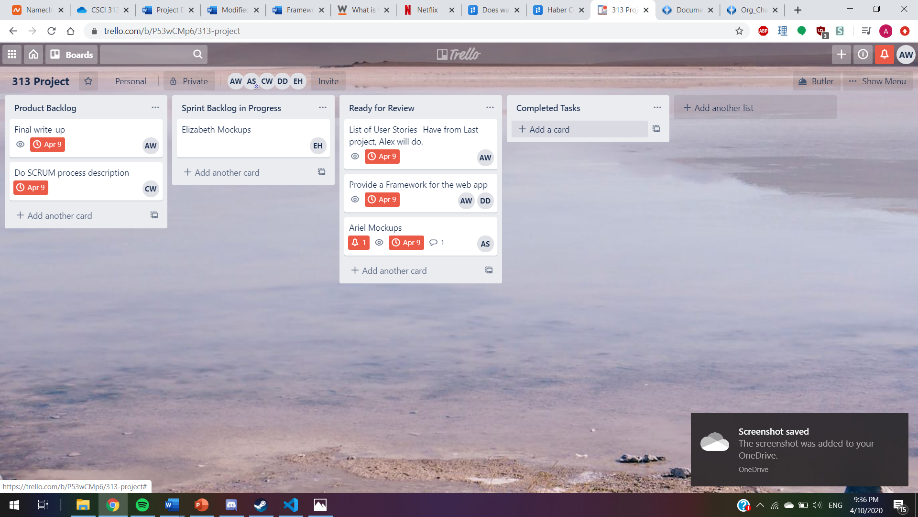
# Scrum Process:

* We decided that there was not enough work for two sprints, and instead used one sprint to complete the project. We met on Discord to discuss any possible changes we wanted to make and what we wanted the project to look like when it was completed, and identified three major areas that we would have to work on for our project- the mockups, creating the framework, and completing a final write-up. We divided the work between three teams: Connor and Alex worked on the framework, Elizabeth and Ariel worked on the mock-ups, and Damien would work on the write-up. Elizabeth and Ariel split the mock-ups in half between them, and Connor and Alex worked in tandem on the framework.
* The team met twice, at the beginning of the Sprint and at the end, for final review. Alex was available throughout to answer questions and keep tabs on the team.
* The team communicated primarily through Discord, and sometimes Alex would utilize texting to get in direct contact with certain team members.

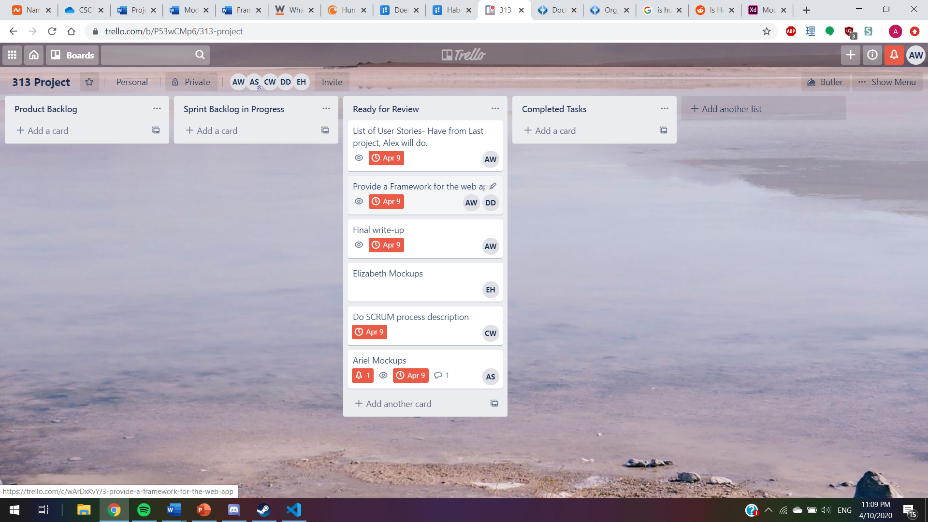
Scrum Screenshots:

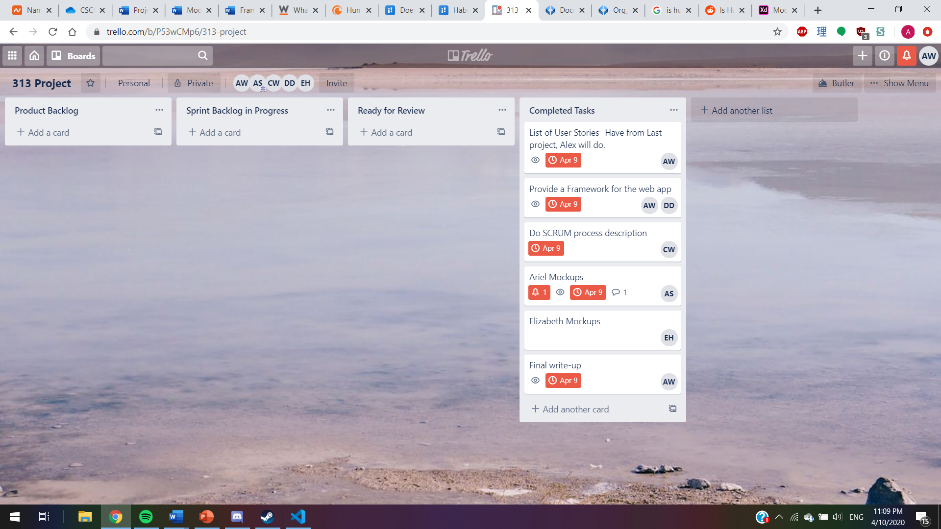






Scrum Screenshots Continued:





Epic and User Stories:

Epic:

* As a cook I can find recipes in ways pertinent to my situation and preferences to make cooking easier.

User Stories:

* As a new cook I can filter recipes by what I have available in my kitchen so that I can cook without having to run to the store.
* As a new cook I want to find recipes so that I can cook for myself.
* As a user I can sort recipes by their rating so I can choose higher-quality recipes.

Stretch User Stories:

* As someone with diet restrictions I can limit the recipes to whatever diet I am on
* As a parent I can sort recipes by preparation time so that I can cook something in a pinch.
* As a user I can sort by types of recipes, such as soups, deserts, or appetizers, so that I can find a recipe specific for what I am making.
* As someone on a budget I can filter recipes based on their price so that I can keep my meals within budget.

Epic:

* As a cook I have a system of saved content to make using the application in the future easier.

User Stories:

* As a home-cook I can register for an account so that I can have user data saved session to session.
* As a user I can log in so that I can keep my user data and preferences saved.
* As a user my supplies are stored from session to session so that I do not have to repeatedly enter them upon using the service.
* As a user I can leave reviews of recipes I have cooked myself so that I can make the experience easier for other users.
* As a user I can submit a recipe for others to use

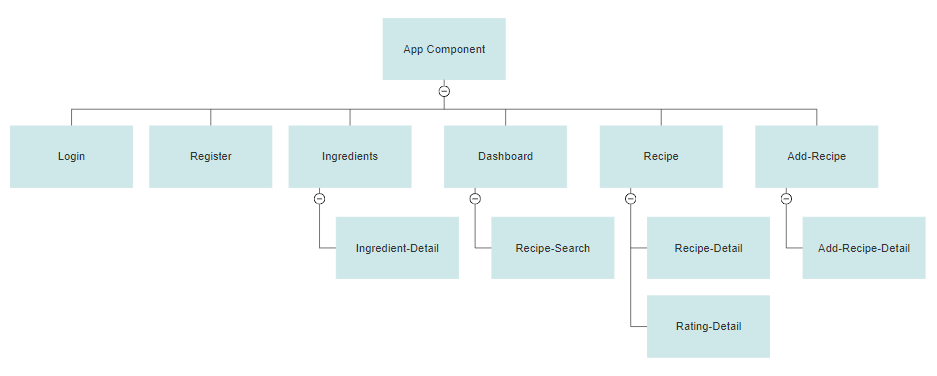
Stretch User Stories:

* As a user I can see past recipes I have used so that I can reuse them.
* As a user, I can add recipes to the list of available recipes.

Framework:

List of Components:

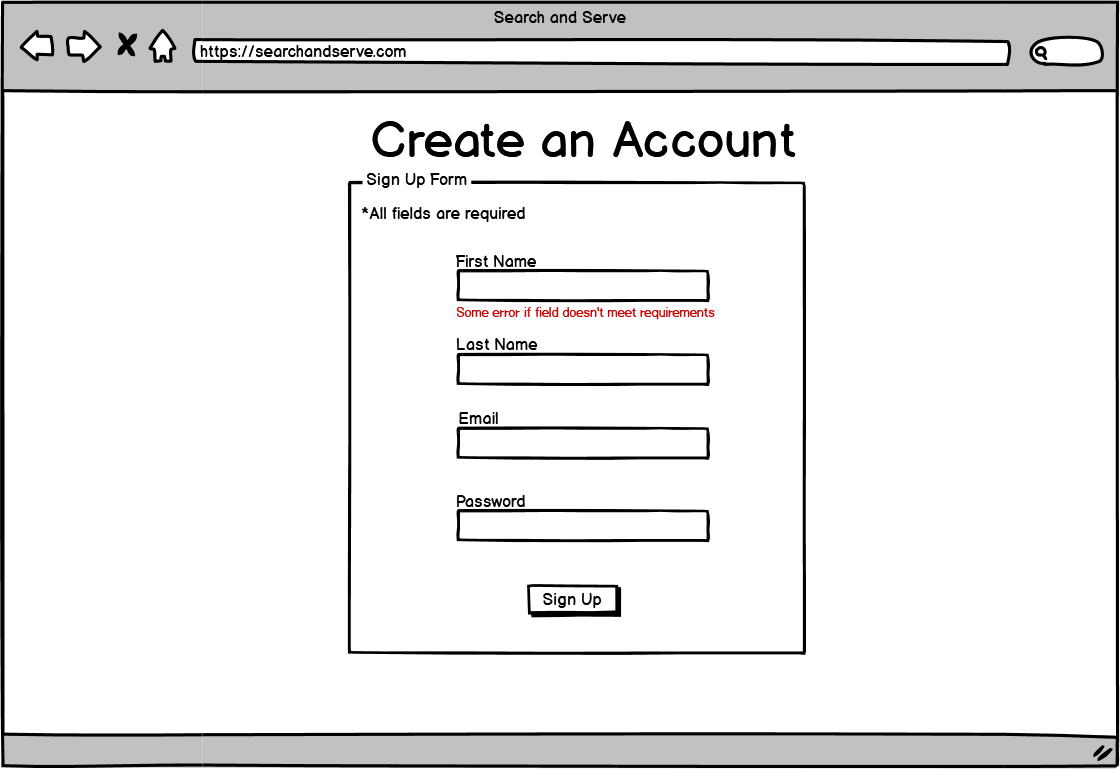
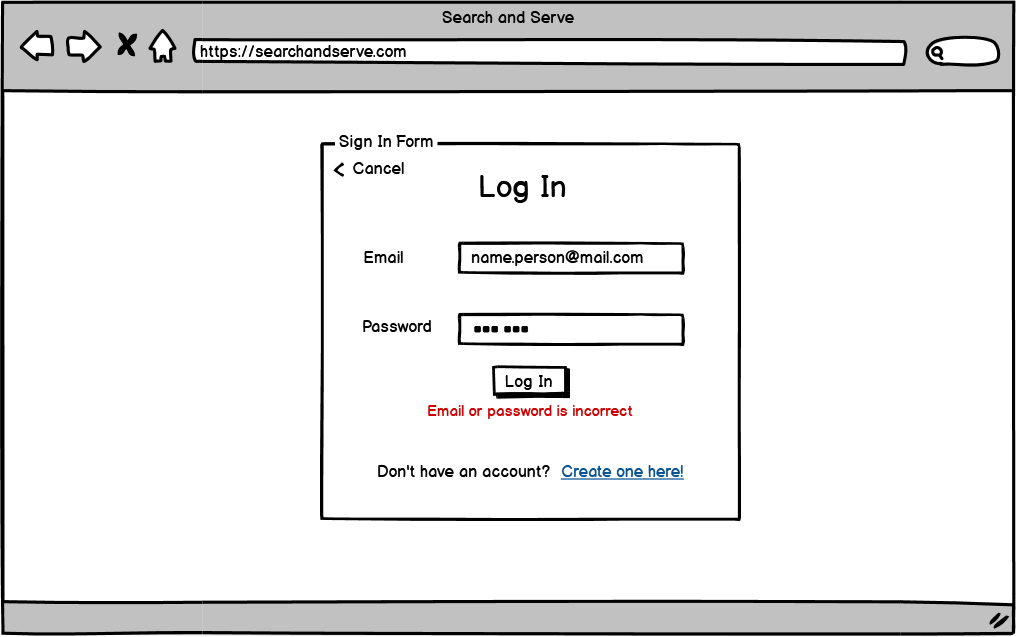
* App-component
* Login
* Register
* Dashboard
  + Recipe-search (feature to search the available recipes based on your ingredients/list view of recipes)
* Ingredient (List view of your available ingredients, with add ingredient feature)
  + Ingredient-detail (data of a recipe)
* Recipes (individual view of the recipes)
  + Recipe Detail (data of a recipe)
  + Rating-Detail (rating of a recipe)
* Add Recipe (Let’s you add a recipe)
  + Add-Recipe-Detail (Support for adding a recipe)



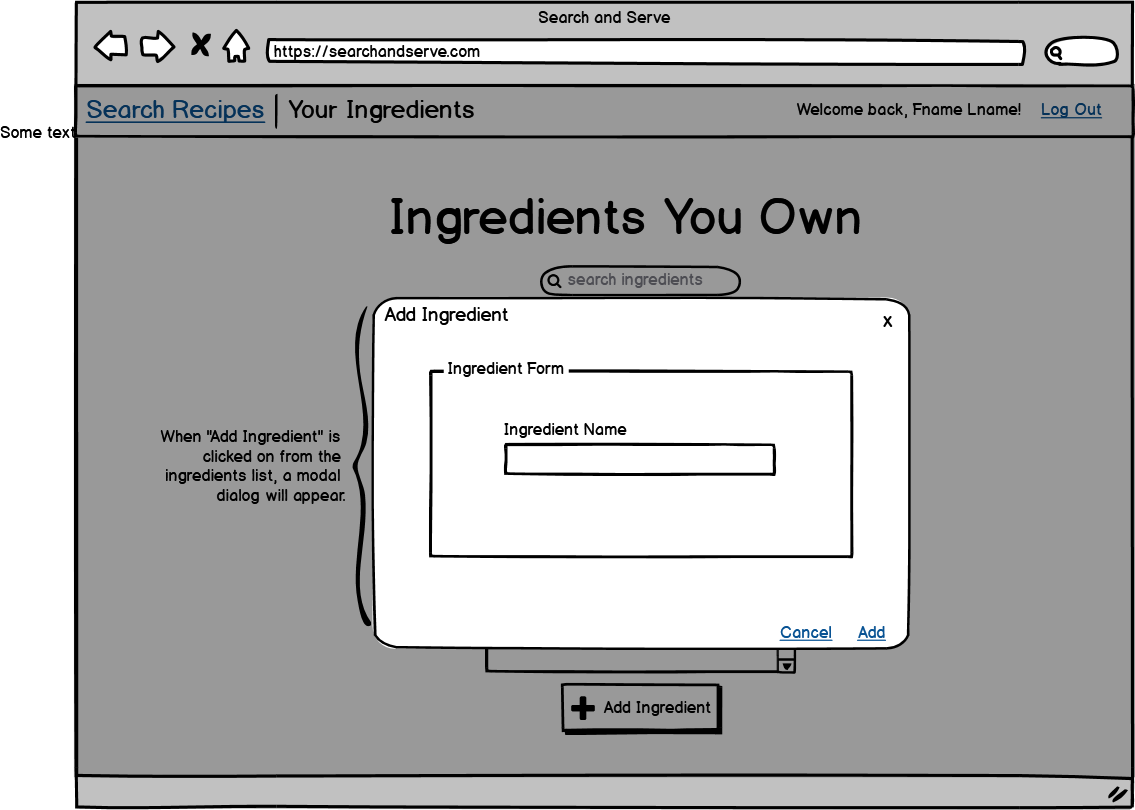
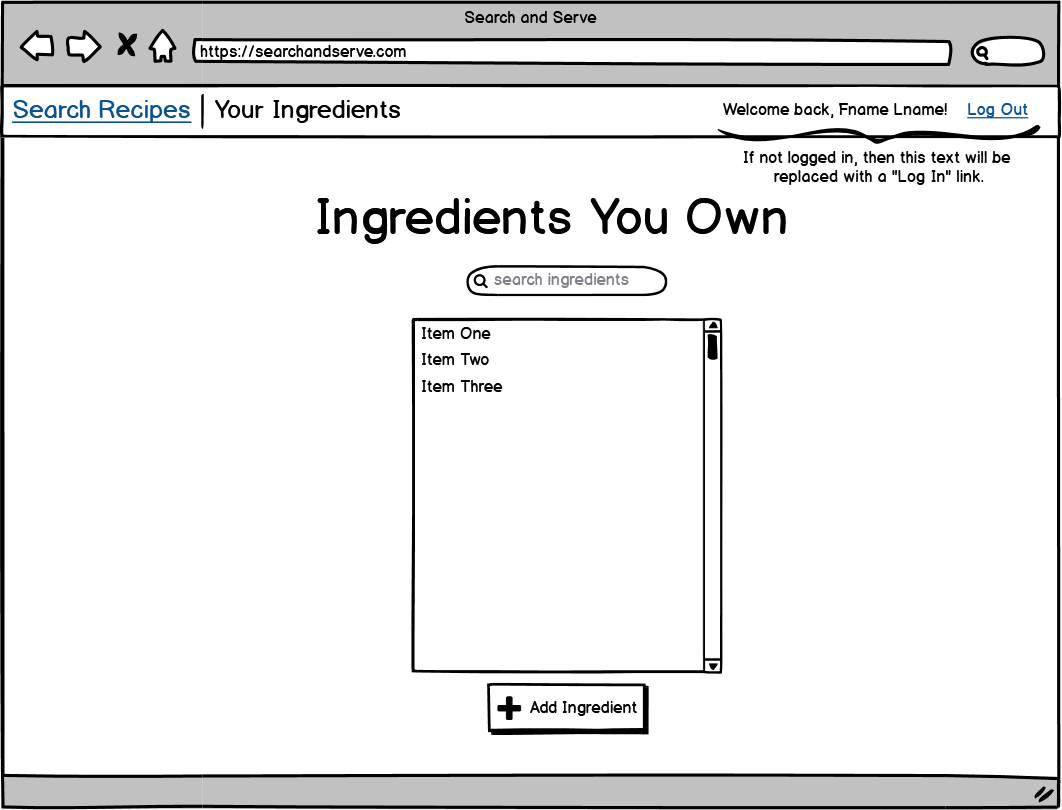
Services:

* user info Service (Shuttles user data to and from database, including when logging in)
* Search Service (Searches through the Recipe List, which is gotten by Recipe Service)
* Recipe Service (Gets the list of Recipes)
* Rating Service (Gets the data of all Ratings of each recipe)
* In-memory-data Service

Mockups:



Mockups Continued:



|  |
| --- |
| Inserting image... |

Mockups Continued:

|  |  |
| --- | --- |
| Inserting image... | |
|  |