

---

# **Software Requirements Specification**

**for**

# **RecipeBuddy**

**Version 1.2**

**Prepared by Michael Prevost, Michael Pierce, Taylor Manley,  
Randeep Singh**

**PiercePrevostSinghManley**

**10/06/2021**

# Table of Contents

<b>0. Revision History</b>	<b>4</b>
<b>1. Introduction</b>	<b>5</b>
1.1 Purpose	5
1.2 Document Conventions	5
1.3 Intended Audience and Reading Suggestions	5
1.4 Project Scope	5
<b>2. Overall Description</b>	<b>6</b>
2.1 Product Perspective	6
2.2 Product Features	6
2.3 User Classes and Characteristics	7
2.4 Operating Environment	8
2.5 Design and Implementation Constraints	8
2.6 User Documentation	8
2.7 Assumptions and Dependencies	8
<b>3. System Features</b>	<b>9</b>
3.1 User Registration	9
3.2 Recipe Creation	9
3.3 Pantry and Shopping List	10
3.4 Search Function	10
3.5 Trending Page	11
3.6 User Interfaces	12
3.7 Hardware Interfaces	16
3.8 Software Interfaces	16
3.9 Communications Interfaces	17
3.10 Performance Requirements	17
3.11 Safety Requirements	17
3.12 Security Requirements	17
3.13 Software Quality Attributes	17
<b>4. Key Milestones</b>	<b>18</b>
<b>5. Key Resource Requirements</b>	<b>19</b>
<b>6. Requirement Change Management</b>	<b>20</b>

<b>a. Glossary</b>	<b>20</b>
<b>b. Analysis Models</b>	<b>20</b>
<b>c. Issues List</b>	<b>20</b>

**Revision History**

<b>Name</b>	<b>Date</b>	<b>Reason For Changes</b>	<b>Version</b>
Pierce, Prevost, Singh, Manley	10/03/21	Initial Version	1.0
Pierce, Prevost, Singh	10/06/21	Continuing Initial Version	1.1
Pierce, Prevost, Singh, Manley	10/6/21	First Document Submission	1.2
Pierce	11/14/21	Includes Case Diagrams for Each Function	1.3

# **1. Introduction**

## **Purpose**

The purpose of the Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of RecipeBuddy. This document will cover each of the system's intended features, as well as provide a preliminary glimpse of RecipeBuddy's User Interface (UI). The document will also cover hardware, software, and other dependencies. The main priorities for RecipeBuddy will be functions to allow creating, deleting, and sharing recipes. This also includes having an ingredient and appliance list. A function search functionality will be a priority.

## **Document Conventions**

This document features some terminology that readers may be unfamiliar with. See Appendix A (Glossary) for a list of items and their definitions.

## **Intended Audience and Reading Suggestions**

This document is intended for anyone overseeing the RecipeBuddy project. Readers who are interested in an overview of the project should focus on Part 1, the Introduction, as well as Part 2, the Overall Description). These readers may also be interested in Part 6, Key Milestones, to obtain a concise timeline of the project to its current state.

Readers who wish to delve deeper into the features of RecipeBuddy should continue onto Part 3, System Features, which expands upon Part 2. Beyond this, readers should continue onto Part 4, External Interface Requirements, which offers technical details ranging from details about the user interface to the hardware and software platforms RecipeBuddy runs on.

Readers not interested in technical details should read Part 5, which covers performance, safety, security, and other aspects of the project that will be important to users. Readers who have not found the specific information desired should seek Part 8, Other Requirements, for any additional information that may not be appropriate for other sections of this document.

## **Project Scope**

The RecipeBuddy system is an internet website designed to facilitate recipe creation and storage. Some potential scenarios include creating and sharing recipes, utilizing a cooking timer, converting units, etc.

## **2. Overall Description**

### **Product Perspective**

RecipeBuddy is a product intended for use on a web browser. There is a server-side database of user accounts and user information, including recipes and ingredients. This server will be responsible for all information and client connections/functionalities.

### **Product Features**

1. User Registration
  - Appears on every page allowing easy accessibility
  - Allows the user to register with the RecipeBuddy server
  - Enables the user to customize his/her account and preferences upon creation
2. Recipe Creation
  - Provides a number of ways users can customize their recipes
  - Allows users to import and export recipes
  - Supports the deletion and editing of recipes after creation
3. Pantry & Shopping List
  - Provides users with a way to store what appliances and ingredients they currently have
  - Allows user to add missing ingredients to a shopping list
  - Prompts user to add ingredients to the shopping list when trying to create recipes without the required ingredients
4. Viewing Recipes
  - Allows the user to view their recipes at any time by choosing the “My Recipes” tab
  - Supports numerous ways the user can sort his/her recipes (recency, tags, pantry, etc.)
  - Provides the user the option to favorite any recipe on the website so they do not lose it and allows quick navigation later
5. Search Function
  - Enables user to search recipes globally with a number of toggleable options
  - Provides the option to only search with the current items in your pantry
  - Searches the server quickly and efficiently

## 6. Trending Page

- Displays the most popular recipes searched
- Updated daily to match current trends
- During special events (holidays, anniversaries, birthday, etc.) layout and color scheme is thematically appropriate

## 7. Member-to-Member actions

- Offers the option to review other users' recipes
- Allows the user to favorite and save other users' recipes
- Provides the option for the user to follow other users

## 8. Settings Menu

- Allows the user to customize features of his/her account
- Enables the user to modify certain website features and functionality
- Can be accessed at any time by clicking on the account information button on any page

# User Classes and Characteristics

### 1. Constant use (once a day)

- Key Functions:
  - Track ingredients in a user's pantry
  - Track recipes cooked and frequency of cooking
  - Offer helpful suggestions to a user's pantry
- Requirements:
  - User accounts
  - Method for adding ingredients to the pantry
  - Method for tracking user-cooked recipes
  - Automatically find recipes in the system based on a user's pantry

### 2. Mild use (twice a week)

- Key Functions:
  - Make recipes easy to find
  - Suggest recipes based on time of day
  - A user-friendly interface to quickly and easily find recipes
- Requirements:
  - A smart search function that takes a user's pantry into account
  - A server clock to suggest recipes based on time of day
  - A simple interface that is very straightforward and intuitive

### 3. One time use (Guest account)

- Key Functions:
  - Quickly find relevant recipes for the date and time
  - Easy to sign up and cook

- Allow people to cook without filling out a pantry
- Requirements:
  - Simple design that suggests relevant trending recipes
  - Easy and simple sign-up page that takes not much time
  - Allow people to skip the pantry function.

These are only different examples of users and situations users might find themselves in. Some people may need all of these functions and some may not.

## Operating Environment

The main component of the RecipeBuddy project is the software application, which is limited to only Web use. The application is not resource- or graphics-intensive, so there are practical hardware constraints.

The application will, however, frequently interact with the RecipeBuddy server, which is hosted locally (TBD). The RecipeBuddy database will be stored on the server using MySQL and will be interfaced with JavaScript and other software languages when appropriate.

## Design and Implementation Constraints

RecipeBuddy is designated for web use, which includes uses on mobile platforms. Screen size and resolution will be important, but other constraints will include limited processing power and memory. Since RecipeBuddy is intended to be quick and responsive, as well as access to any user; each feature must be designed and implemented with efficiency in mind.

## User Documentation

The primary goal of RecipeBuddy is to ensure the process of creating, editing, and sharing recipes. As a result, the application will be designed to be as easy to use as possible. Users may still require additional information which will be found in the RecipeBuddy Tutorial.

The RecipeBuddy Tutorial takes the primary topics and condenses them into a simple-to-follow, step-by-step demonstration that the user can access at any time. This tutorial is meant to quickly teach new users the functionality of RecipeBuddy.

# 3. System Features

## User Registration

### 3.1.1 Description and Priority

When the webpage is first loaded, the user will be presented with the homepage screen. The user will select the log-in icon in the top right to continue account creation/log-in. This screen will prompt the user to create an account, log in, or continue as a guest.



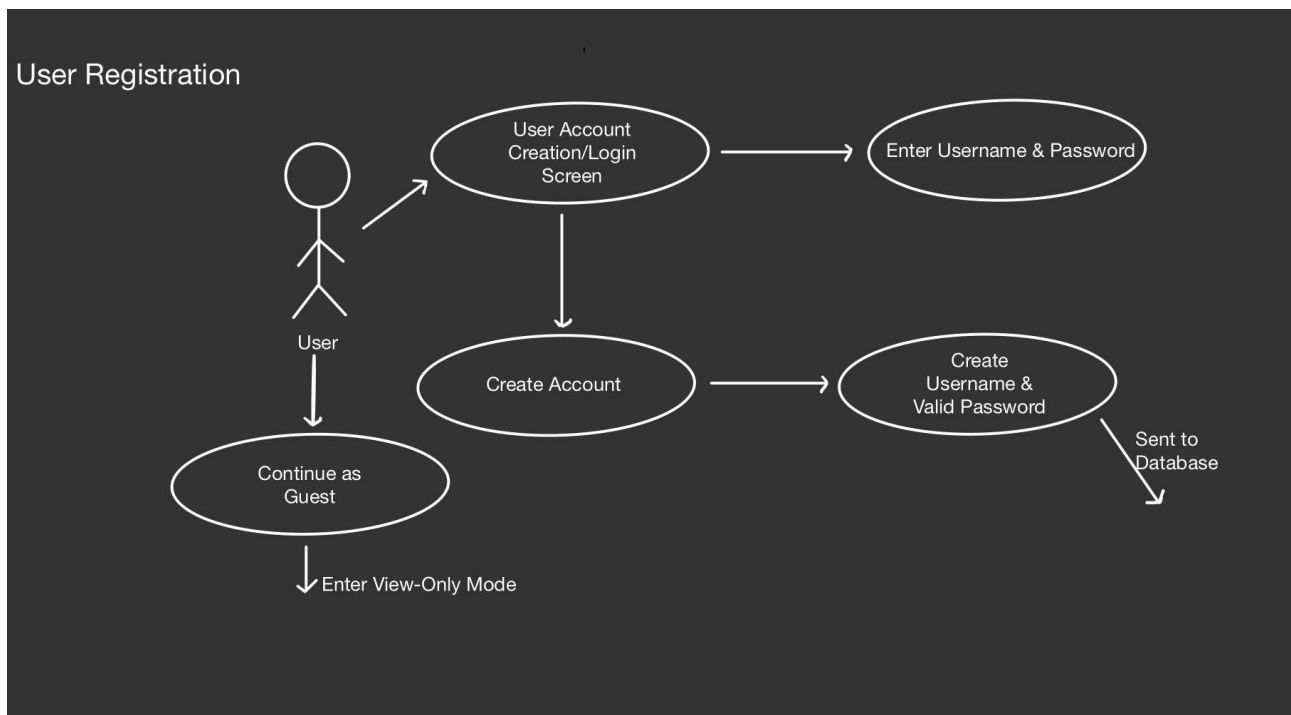
The user will enter a display name that will be used as their handle, as well as a password to access their account. Completing this process will create and store an account for the user on the RecipeBuddy server.

### 3.1.2 Stimulus/Response Sequences

- Step 1** The RecipeBuddy website is launched in the user's browser  
**Step 2** The user selects the log-in option located in the top right  
**Step 3** The user is prompted to enter a display name and password  
**Step 4** The information is sent to the server and is stored in the database  
**Step 5** Registration is completed and the user continues to the RecipeBuddy homepage

### 3.1.3 Functional Requirements

- REQ-3.1.1: The user must enter a valid display name and password.  
REQ-3.1.2: There must be a secure database to store the user's information.  
REQ-3.1.3: Users can not register with existing username.  
REQ-3.1.4: Users can not create accounts with weak passwords.



## Recipe Creation

### 3.2.1 Description and Priority

Throughout the website, there will be an Add Recipe button located on every page. When the user selects it, they will be brought to a page that allows them to create a new recipe by including all necessary ingredients and amounts, step-by-step instructions, and has the ability to tag recipes as satisfying dietary restrictions. Completing this process will create and store the recipe on the RecipeBuddy server.

### 3.2.2 Stimulus/Response Sequences

- Step 1** The RecipeBuddy website is launched in the user's browser  
**Step 2** The user selects the Add Recipe button  
**Step 3** The user is prompted to enter ingredients and amounts, steps, and restrictions  
**Step 4** The information is sent to the server and is stored in the database  
**Step 5** The recipe creation is finalized and is published on the website

### 3.2.3 Functional Requirements

REQ-3.2.1: The user must select at least 1 ingredient and list at least 1 step.

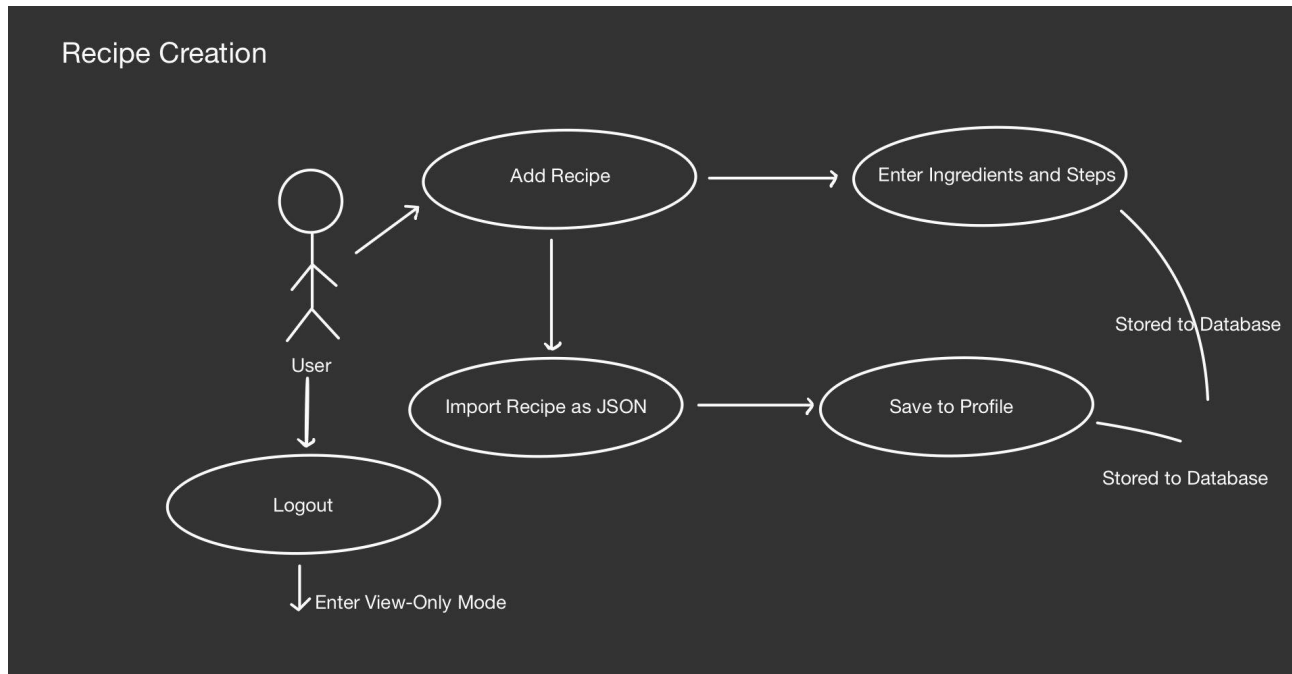
REQ-3.2.2: There must be a database to store the information contained in each recipe.

REQ-3.2.3: There should be step-by-step instructions on the recipe page.

REQ-3.2.4: Users can add recipes with already existing recipe names.

REQ-3.2.5: Users should be able to delete a recipe.

REQ-3.2.6: Users can not leave forms fields blank.



## Pantry and Shopping List

### 3.3.1 Description and Priority

The user will have quick access to their shopping list and pantry from the main page by selecting the tabs at the top of the screen. The user will be able to add any ingredients or appliances they have to their pantry, which will allow them to search by the

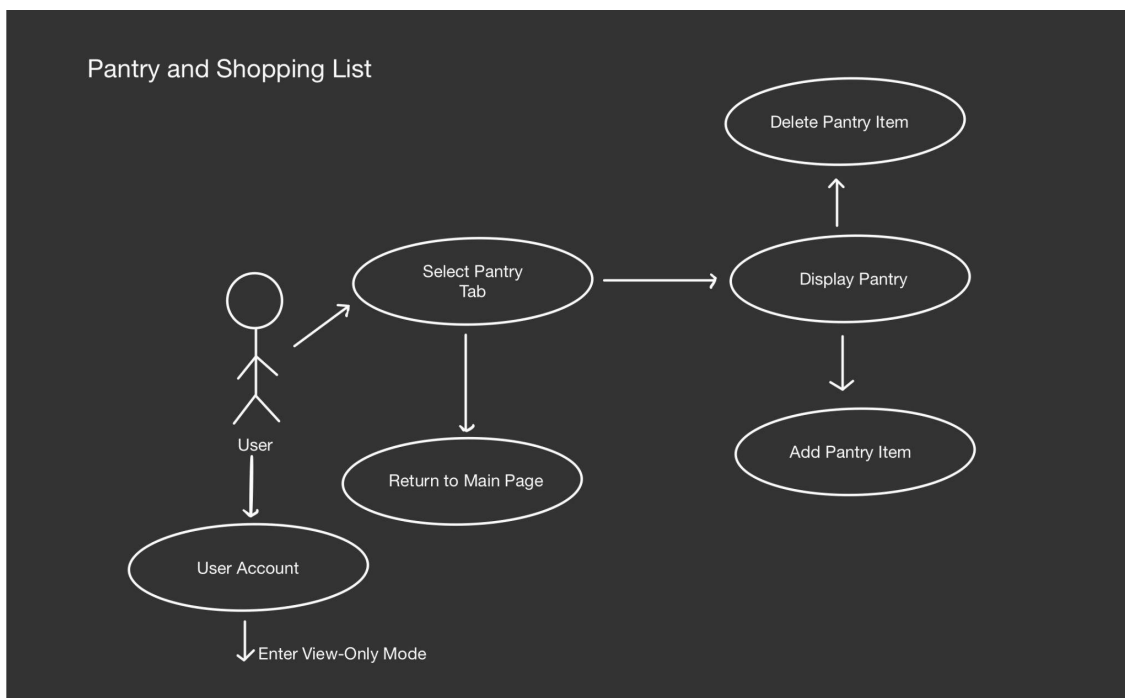
ingredients and appliances they already have. Users can also add ingredients they don't have to their shopping list and they will be organized by date added.

### 3.3.2 Stimulus/Response Sequences

- Step 1** The user selects the pantry tab
- Step 2** The user selects the edit button that allows modifications to their pantry
- Step 3** The user is prompted to either add or remove an ingredient
- Step 4** The information is sent to the server and is stored in the database
- Step 5** The pantry is updated and the user is returned to the pantry

### 3.3.3 Functional Requirements

REQ-3.3.1: There must be a database to store the pantry information.



## Search Function

### 3.4.1 Description and Priority

The user will be able to search for recipes at the top of every web page by selecting the search bar that spans the top of the screen. After selecting the search, there will be an advanced search option that will allow the user to filter the results by allergies or preferences, as well as search by pantry and appliances the user already has. To utilize those advanced search features, the user will be required to have an account.

### 3.4.2 Stimulus/Response Sequences

- Step 1** The user logs in to their account
- Step 2** The user selects the search bar and “Advanced Search”
- Step 3** The user enters their filters

**Step 4** The information is sent to the server returns all the matching results matching the criteria provided

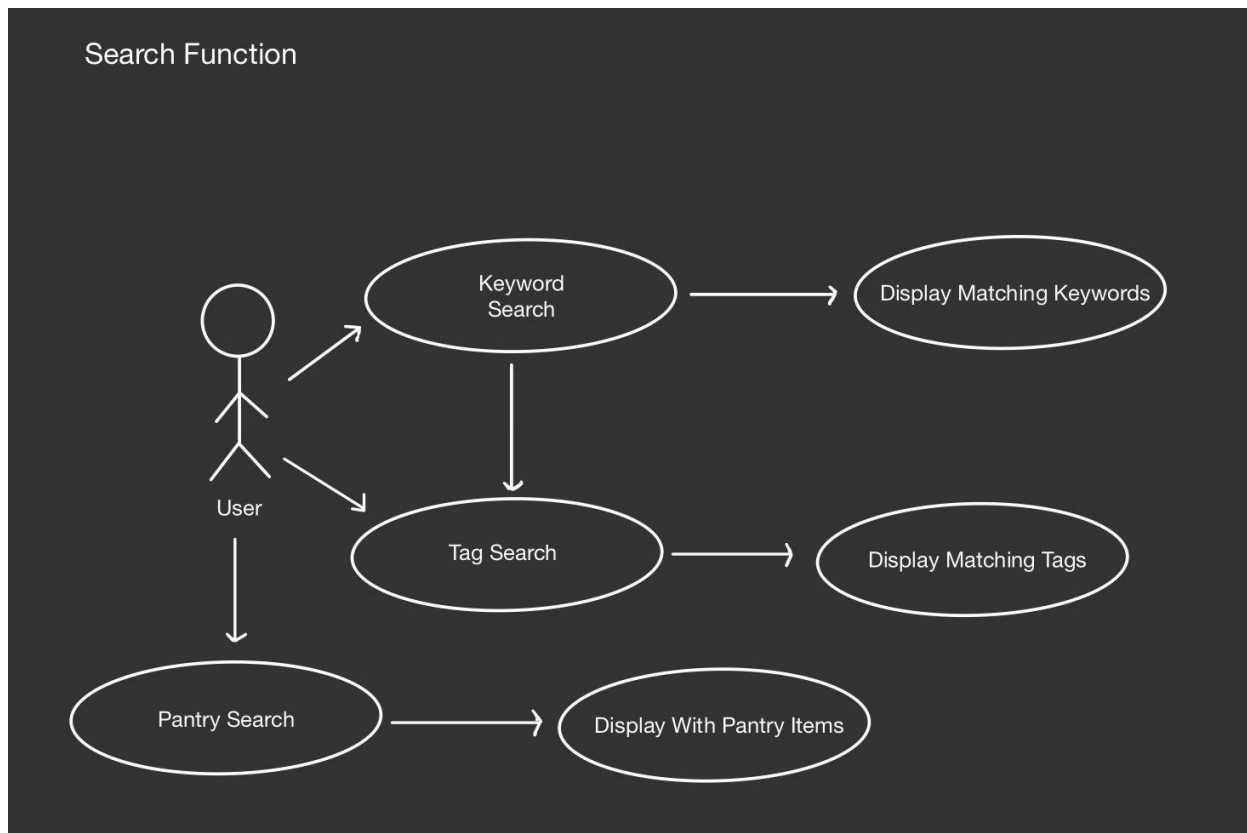
**Step 5** The results are displayed in a list to the user

### 3.4.3 Functional Requirements

REQ-3.4.1: The user must type a valid character into the search

REQ-3.4.2: The user must be logged in to an account to use advanced features

REQ-3.4.3: There must be a database to store lists of recipes



## Trending Page

### 3.5.1 Description and Priority

The top of the main webpage will hold a rotating banner that displays trending recipes or featured recipes. These can be themed around time-of-day, upcoming holidays, or current events. When a user selects one of the banners, they will be presented with a search results screen of the recipes that were included in that showcase.

### 3.5.2 Stimulus/Response Sequences

**Step 1** The RecipeBuddy website is launched in the user's browser

**Step 2** The user selects one of the rotating banners

**Step 3** The information is sent to the server and it returns a showcase of recipes

**Step 4** The recipes are shown to the user in a search results screen

### 3.5.3 Functional Requirements

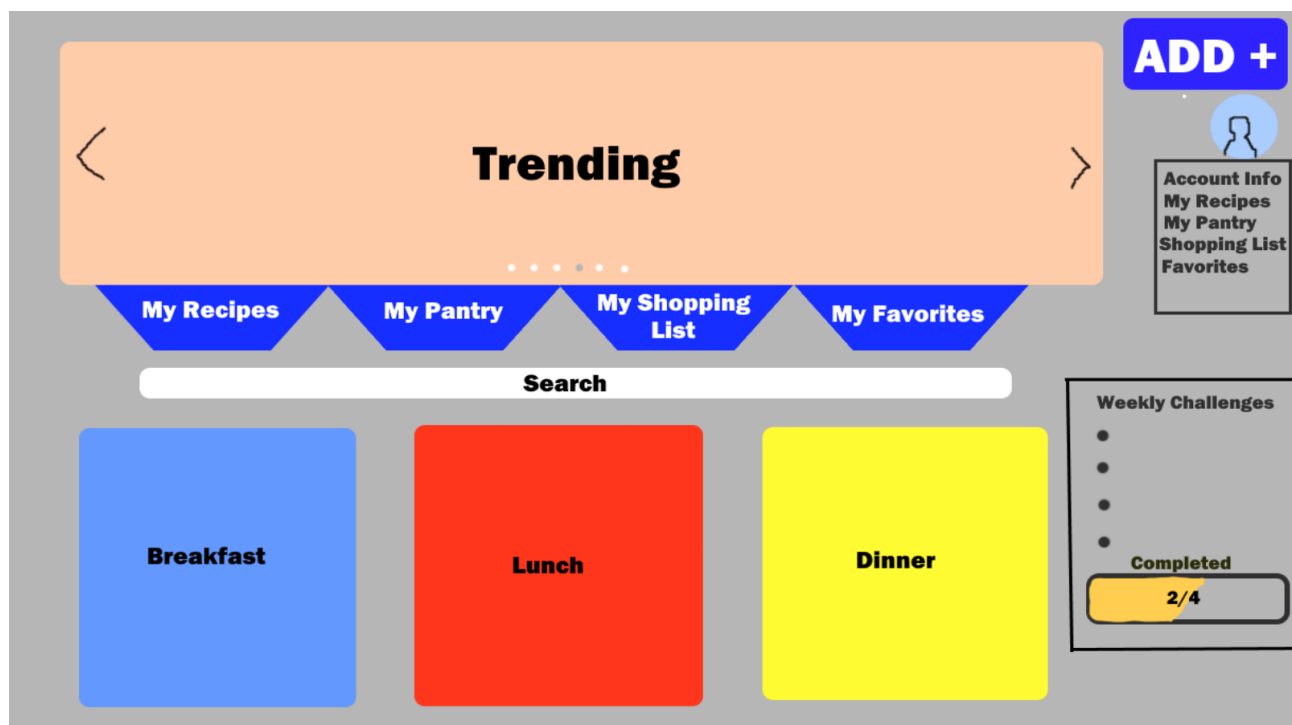
REQ-3.5.1: Banner should rotate with popular recipes.

REQ-3.5.2: Banner items should redirect to the appropriate recipe.

## External Interface Requirements

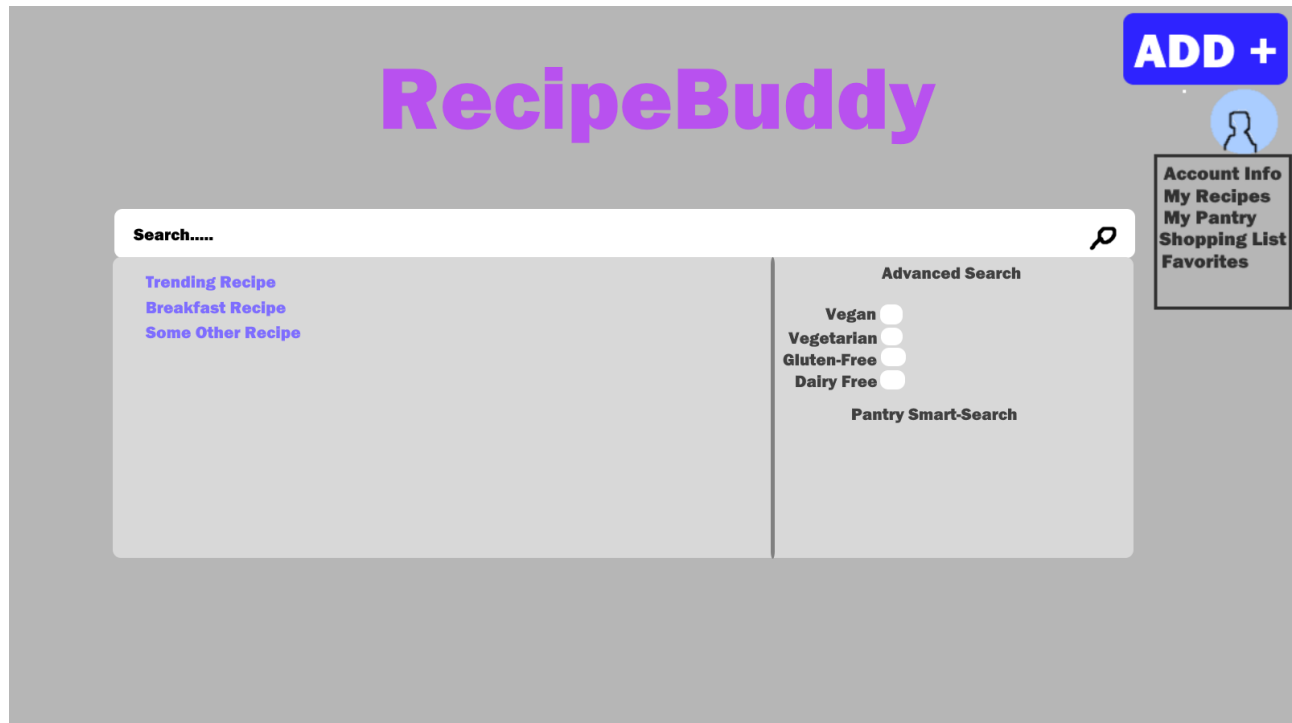
### User Interfaces

This section will include mockups of our current vision. The final project may look entirely different.



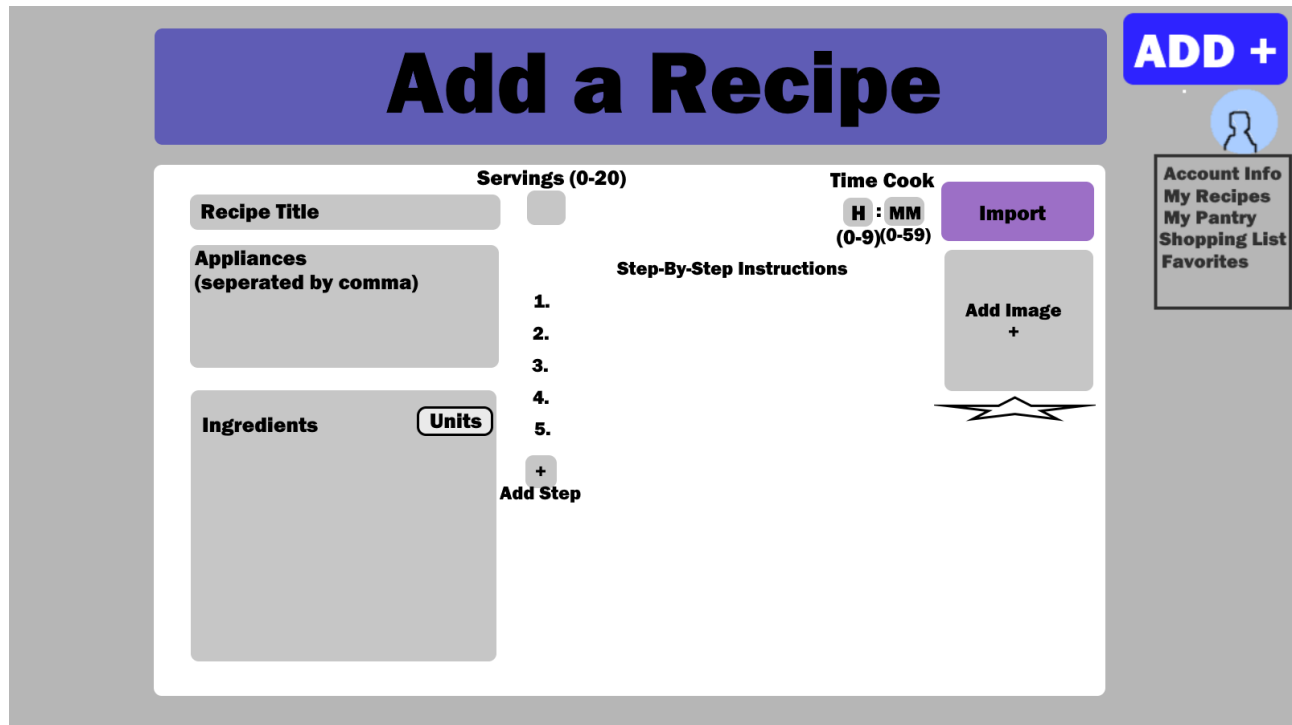
**Figure 1** - Mockup for the homepage

- This is the main page that a user will see when they first enter the website.
  - Users have the ability to log in to their accounts from all pages
  - Users have the ability to add a recipe from all pages
  - There will be a carousel of trending recipes covering the top  $\frac{1}{3}$  of the screen with related recipes around the bottom half
  - Clearly labeled tabs will be used to guide users through the website
  - Clicking on the search bar will bring users to a search screen



**Figure 2** - Mockup of the Search Functionality

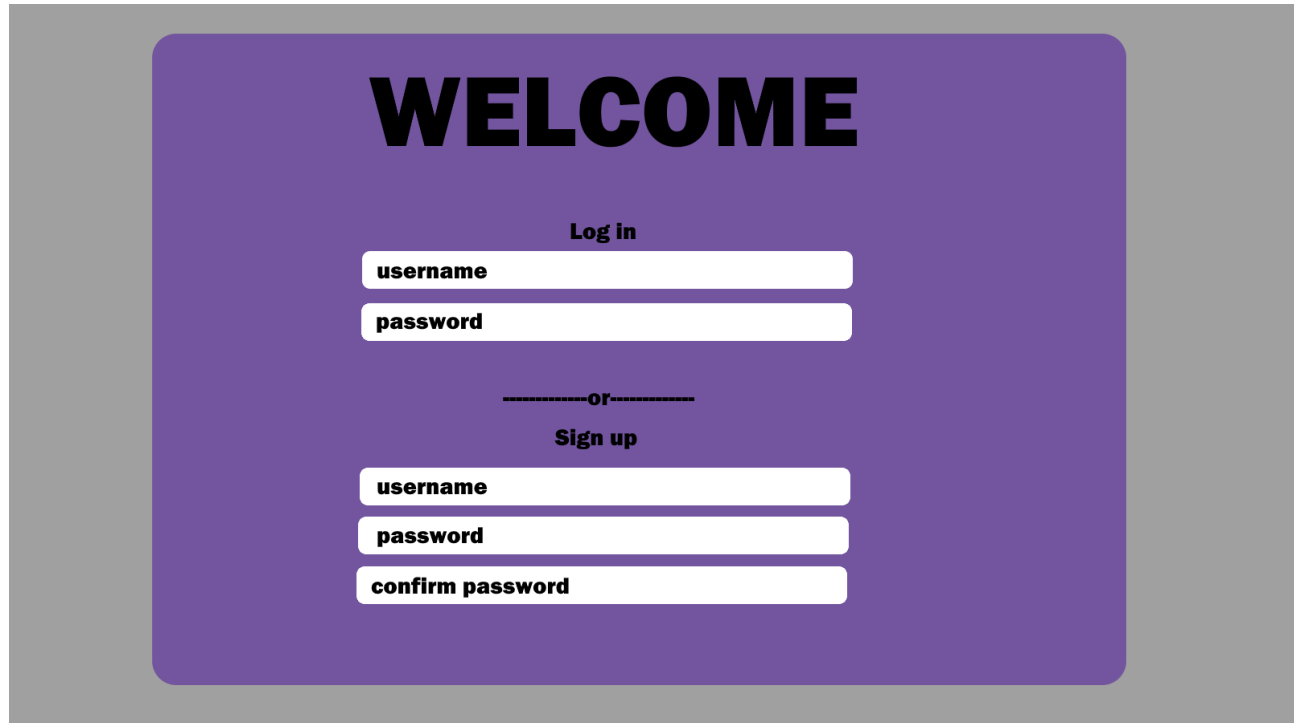
- This is an example of the search screen
  - The search bar will have an area underneath where RecipeBuddy will suggest trending recipes based on the time of day or year.
  - Advanced search options to exclude certain ingredients or allergies will be available.



The mockup shows a web interface for adding a recipe. At the top is a large purple header with the text "Add a Recipe". To the right of the header is a blue button labeled "ADD +" and a user profile icon. Below the header, the form is organized into several sections. On the left, there is a "Recipe Title" text field, an "Appliances (seperated by comma)" text area, and an "Ingredients" text area with a "Units" button next to it. To the right of these is a "Servings (0-20)" dropdown menu. Further right is a "Time Cook" section with "H : MM" input fields and a range "(0-9)(0-59)". An "Import" button is located to the right of the "Time Cook" section. Below the "Time Cook" section is a "Step-By-Step Instructions" area with a numbered list from 1 to 5 and an "Add Step" button. To the right of the instructions is an "Add Image" button with a plus sign and a star icon. On the far right, there is a sidebar with a user profile icon and a list of links: "Account Info", "My Recipes", "My Pantry", "Shopping List", and "Favorites".

**Figure 3** - Mockup of the Recipe Creation page

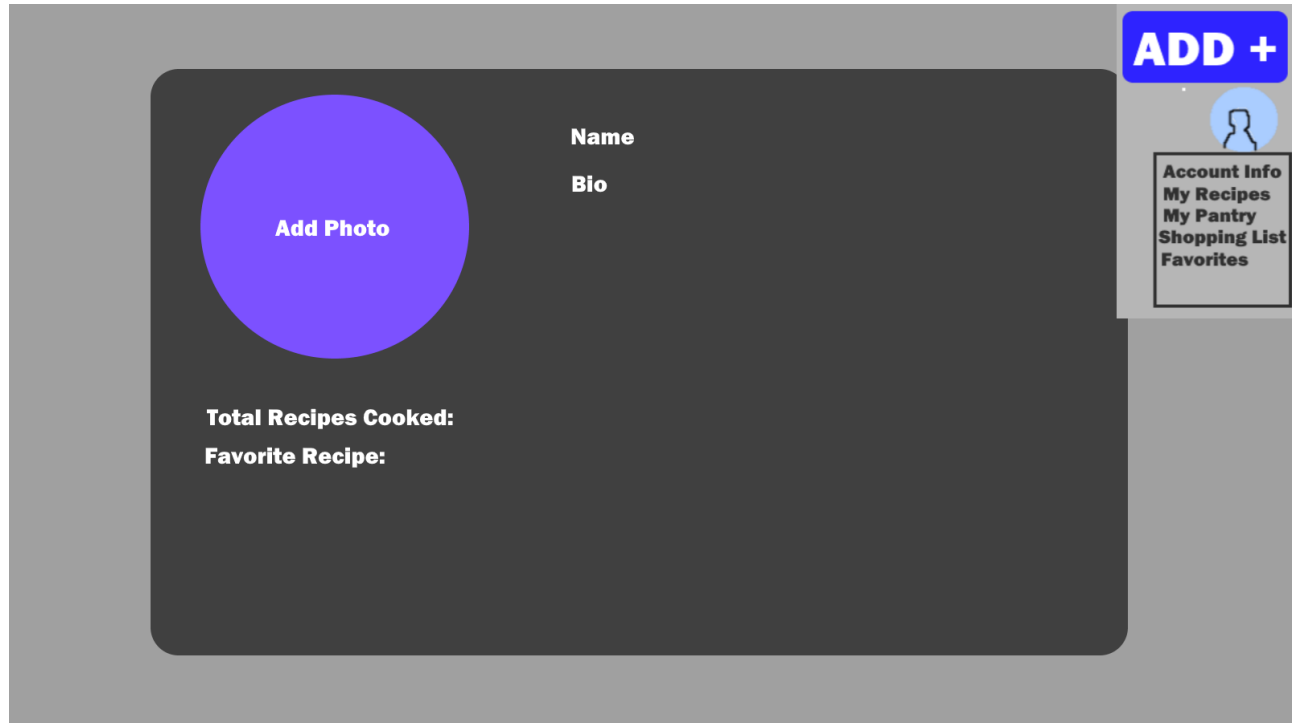
- When users go to add (or edit) a recipe, they will be met with this page
  - Multiple text areas where users can add a title, steps, ingredients, required appliances
  - Users will be able to set a serving size (drop-down menu 0-20) and a set timer for cooks to use.
  - Users can import recipes from JSON file formats



**Figure 4** - Welcome and login page mockup

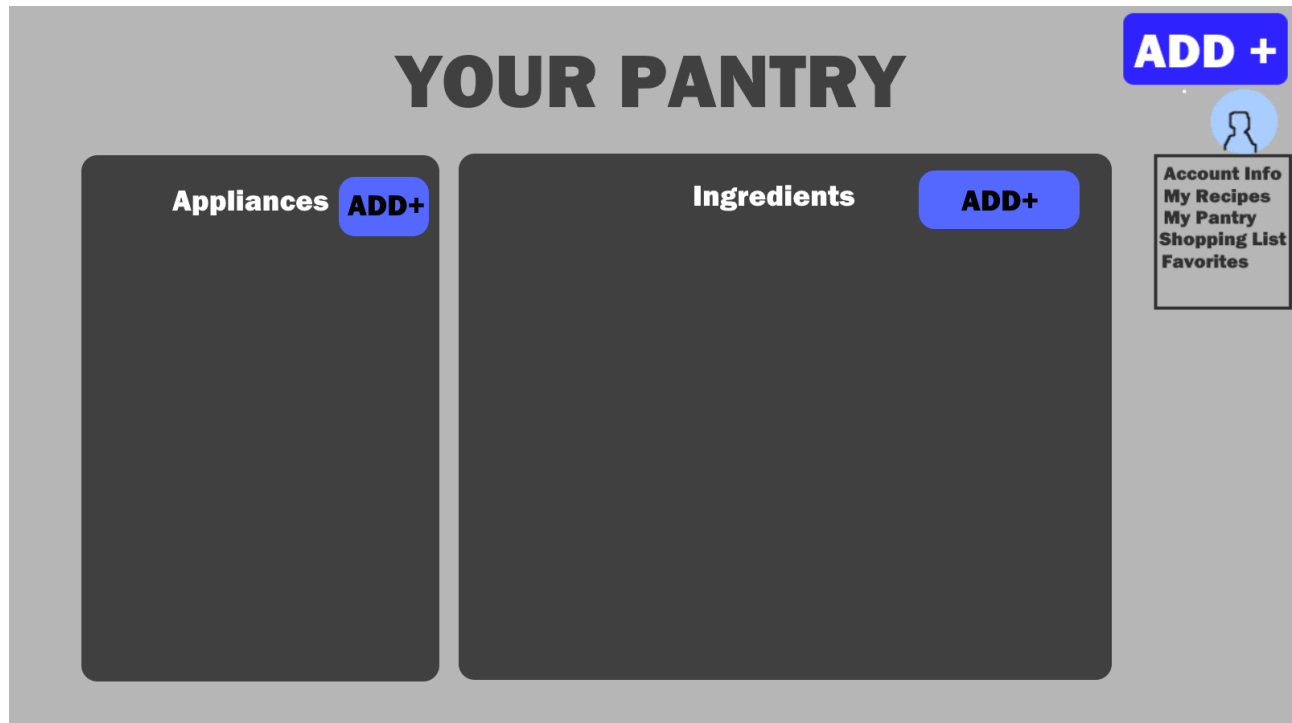
- Users will be met with this page when logging in
  - Existing users will be able to log in and be sent back to the home page
  - New users after account creation will be prompted to add their appliances and be sent to their profile screen to add a picture and/or bio





**Figure 5** - Account Information page

- This is the profile screen where users can add/change their information
  - Users can add their name, photo, favorite recipe, and a short bio of themselves
  - A total number of recipes cooked will be tracked for each user



**Figure 6** - Mockup of the user pantry system

- Each user will have their own private pantry
  - Users can add/remove ingredients
  - Users can add/remove appliances

## Hardware Interfaces

RecipeBuddy is intended as a website to be used in modern browsers. Recipes, account management, and data exchange are handled by the RecipeBuddy Server. This produces the illusion of peer-to-peer interactions between users, however, this is not the case as all interactions run through the server first.

## Software Interfaces

RecipeBuddy is being developed specifically with HTML5 and CSS3. As such, some functionality native to these platforms may not work on older browsers as intended. JavaScript is used to handle communication between the front-end and back-end, in addition to Python, wherever required. The backend will be using MySQL and handling the database systems.

### 4.3.1 Incoming and Outgoing Items

- Outgoing data consists of account information, recipes, and all subsequent information (ingredients, user ratings, comments, times cooked, difficulty rating), search queries, shopping lists.
- Incoming data consists of updates from the server regarding account information as well as trending recipes, and server updated recipe information (such as ratings).

#### **4.3.2 Services and Communications**

- RecipeBuddy relies on server push-and-pull functionalities to allow the website to work as intended.
- Communication will occur in occasional, small bursts, between the website and the server in the following situations:
  - Whenever a user creates an account
  - Whenever a user creates/edits/deletes a recipe
  - Whenever a user uses a recipe and cooks it
  - Which recipes have been searched in the past 24 hours to allow the functionality of the trending feature to function as intended
  - Whenever the server finished searching for a recipe upon request from the server
  - Whenever a user leaves a rating and/or comment on a recipe.
  - Whenever a user requests a change to be made on another user's recipe.

### **Communications Interfaces**

RecipeBuddy will be located on a web-based server and created using HTML, CSS, JavaScript, Python. The server will be used to retrieve information from the database and send it to the user. The server will store information about each user.

### **Other Nonfunctional Requirements**

### **Performance Requirements**

As a website, there should not be many performance issues with navigation from page to page. Functions that may take longer are searching for a recipe and double-checking that a user has all of the ingredients (and appliances) for a recipe in their pantry.

### **Safety Requirements**

RecipeBuddy will not and cannot cause damage to a user's device.

### **Security Requirements**

This website assumes that only the user will have access to his/her recipes. With that being said, only a username and password are required to verify account credentials. Other users should not be able to access other account information and/or stored recipes, shopping lists, or pantry items.

## **Software Quality Attributes**

The graphical user interface of RecipeBuddy is to be designed with ease of use as the primary priority. The website will be presented and organized in a manner that is functional and visually appealing.

## 4. Key Milestones

#	Milestone	Target Completion Date	Comments
1.	<b>RecipeBuddy</b>	<b>December 05 2021</b>	<b>Final Product</b>
2.	Homepage	October 31, 2021	Usable Shell
3.	Database	October 24, 2021	
4.	User Accounts	October 31, 2021	Base code to store data
5.	Add Recipe	October 31, 2021	Able to add and save to user accounts
6.	Search	November 13, 2021	Search at least by recipe title
7.	Pantry	November 13, 2021	Store user food/appliances
8.	Trending	November 13, 2021	Suggested recipes by time of day/year
9.	Tags	November 21, 2021	At least 1 visible tag per recipe
10.	Shopping List	November 21, 2021	Able to add/remove items to cart

## 5. Key Resource Requirements

Major Project Activities	Skill/Expertise Required	Internal Resource	External Resource	Issues/Constraints
<b>RecipeBuddy</b>				
Homepage	HTML/CSS/JS (UX)	Randeep, Michael Pierce, Michael Prevost have knowledge in this area	Internet	Learning JavaScript
Database	Django (Python) MySQL	The team has experience with Python but none with the Django frameworks.	Internet, Colleagues	Learning back-end technologies and how-to create a database Learning MySQL and Django
User Accounts	Security (Scrubbing input)		Internet	Needs functional database Learn account validation and account creation implementation
Add Recipe	Storing correctly, JSON, scrubbing input		Internet	Learning JSON
Search	Connecting to tags, scrubbing input, effective database searching algorithms		Internet	Finding out which algorithm is best for our use
Pantry	Storing and Saving in a database		Internet	Learning Django, MySQL
Trending	Accessing time of day and holidays to display appropriate recipes, scrubbing most cooked recipes in 24hr		Internet	Using time of day or holiday season to change what is shown on the website.
Tags	Data		Internet	Learning Django, MySQL
Shopping List	Creating lists that can be easily accessed, database storage		Internet	Learning Django, MySQL

## 6. Requirement Change Management

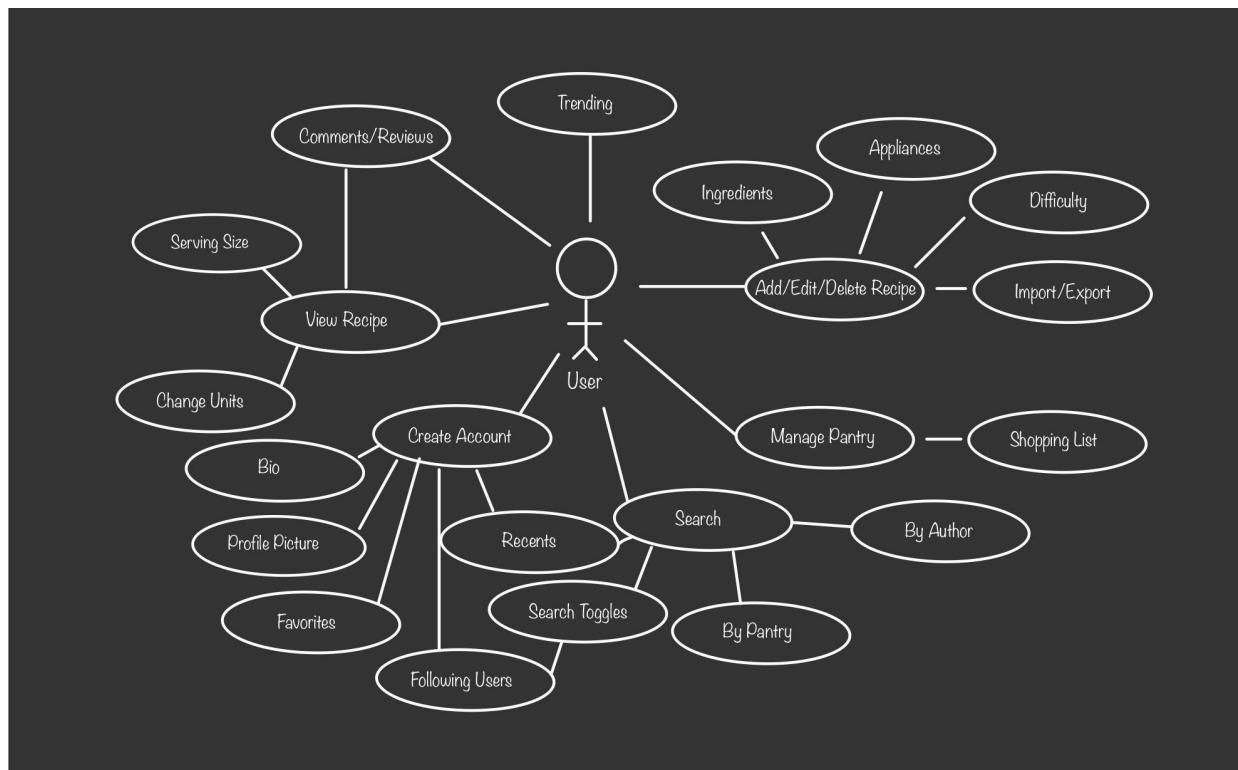
The customer can request requirements updates or changes through the team Slack workspace. They are able to request changes through the [#requirements-changes](#) channel.

## Appendix A: Glossary

None at this time.

## Appendix B: Analysis Models

Case Diagram:



## Appendix C: Issues List

None at this time.