Food For All
Software Requirements Specification

Revision 1.0

Todd Anderson, Thomas Rollin Burr, Whitney Chase, Peyton Dunnaway, Ben Dzado, Janzen Ferrin, Jose Jaime Gamero, Bryan Heder, Nicholas Hendricks, Caleb Hensley, Austin Kelly, Shawn Lilly, Scott Malin, Jonathan Mancia, Tanner Meade,

Dylan Miessner, Paul Owens, Sara Reece, John Reiley, Scott Robison, Nathan Rowley, Koby Sanchez Campbell, Stephen Sharp, Jonathan Stutz, Jordon Thompson,

Adam Tipton, Tanner Wilson and Joshua Isom

Table of Contents

List of Figures

List of Tables

1 Introduction

This document serves as the Software Requirements Specification (SRS) for **Food For All**, a web-based group cookbook application.

1.1 Purpose

The purpose of Food For All is to provide a platform where a number of people can collaboratively manage recipes, menu planning, and food shopping lists.

1.2 Scope

Food For All will provide a web application based interface, designed to be accessible via traditional as well as mobile web browsers, to several tools:

1. A recipe manager that allows a number of people to collaboratively create, customize, and share recipes with one another.
2. A menu planner that allows a number of people to participate in the creation of menus (ie, recipe selection) for a user selectable period of time (ie, daily or weekly).
3. A shopping list manager that will support automatic generation of a list of ingredients necessary to prepare an individual recipe or menu of multiple recipes.(RR-50, RR-51)
4. User and group management to control who has access to specific instances of recipes, menu planning, and shopping lists.

1.3 Product Overview

1.3.1 Product Perspective

The product, Food for All, will be the main part of the system as well as the interface for the rest of the system as illustrated in Fig. 1. The product will interact with external products such as the calendar and database.

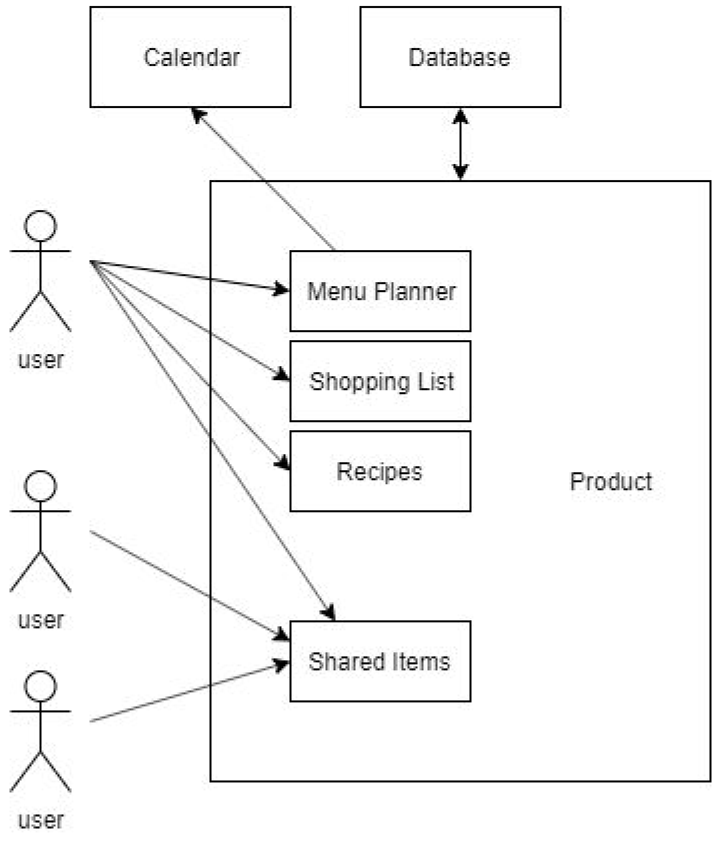


Figure 1: Diagram of the product and its relationship to the system as a whole

1.3.1.1 System Interfaces

Database	Allows for mass storage of user data (user accounts, groups, recipes, shopping lists, menu plans) and support of many simultaneous users

Table 1

1.3.1.2 User Interfaces and Operations

Specific user interfaces and operations shall include a search bar, user group formation, user recipe creation, shopping list generation, meal planning, and database access. Refer to section 3.1.1 User Interfaces for more details.

1.3.1.3 Hardware Interfaces

The application shall work on many platforms but the hardware will be abstracted by the use of web browsers.

a) Desktop Operating Systems
   - Windows
   - MacOS
   - Linux
   - etc

b) Mobile Operating Systems
   - Android
   - iOS
   - Windows

The primary interface output shall be the ability to share recipes. Refer to section 3.1.2 Hardware Interfaces for more details.

At this point in time, a specific implementation is not yet required for the server side platform. This document does not specify requirements or interfacing for such.

1.3.1.4 Software Interfaces

The software interfaces that the application will utilize are web page based interfaces that include the following:

a) HTML
b) CSS
c) JavaScript
d) Google Chrome
e) Firefox
f) Microsoft Edge
g) Safari

Other software interfaces shall include the following APIs:

Recipe Ingredient Parser API	Allows the user to give the app a website url

	from which to import a recipe (RR-7)
Recipe nutrition API	Give the user health/nutritional information on recipes and/or food items (RR-40)
Calendar integration	Allows the user to share and add menu plans to calendar events (RR-48)

Table 2

1.3.1.5 Communication Interfaces

The primary communication interface shall be interaction with the user's device calendar. Refer to section 3.1.4 for more details.

1.3.1.6 Memory Constraints

Primary memory constraints will vary depending on the device accessing the web application.

1.3.2 Product Functions

The Web Application's primary function shall allow users to create private groups, and to import or create recipes to share with other users, while also allowing users to search for, rate, favorite, and share other users' recipes. Recipes shall include various attributes set forth in "Database Entities and Their Relationships"(3.5.4) and be optimized for search, sorting, and filtering so that the user can view recipes according to their needs. To extend the utility of the application, it shall also provide functions to adjust portion sizes when making a recipe, set a daily/weekly meal plan and form a grocery list of the items needed for those meals. To protect against abuse of the application and of its users, the application's algorithms shall also prevent or delete inappropriate or duplicate content. (RR-0)

1.3.3 User Characteristics

Users are expected to have a high school education or higher. Users are expected to be proficient with navigating a web application on desktop and mobile browsers. Users are expected to have an interest in cooking or baking. Most users will not have any disabilities that would make the web application difficult to use.

1.3.4 Limitations

Due to the nature of this application, there are not too many limitations. Most limitations we must consider are related to a Terms of Service (ToS) and copyright law.

1.3.4.1 **Regulatory policies**

The application shall indicate to the user at the time that they sign up that there is a ToS that they must accept. This ToS shall include the copy-right agreement and information as well as an indication that the user agrees to the terms of service. Additionally for those users in the UK and the rest of the EU, the app shall conform to all GDPR regulations that have been put in place.

1.3.4.2 **Interfaces to other applications**

The application shall handle user's uploading information such as photographs, and the way they would interface and interact with other applications. Some considerations are as follows:

- Nutrition API
- Recipe API
- Camera Applications
- Photo Roll Applications

1.3.4.3 **Physical/mental considerations**

The application shall have colorblind settings to account for any users who are colorblind.

1.4 Definitions

Affiliate income - A source of money provided by a group or organization with a vested interest in the success of the app.

Algorithm - A step-by-step way to solve a problem, process or manipulate data, or perform calculations or other operations in a computer system.

Collaborative - When multiple users can participate at the same time, seeing each other's actions or changes.

Cookbook - A set of recipes collected together in a single, cohesive location, such as a file.

Follow - A feature that allows a user to pay close attention to a particular user, recipe, menu, meal plan, or group.

Group - A number of users assembled together that will have access to one or more collaborative works such as menus or meal plans.

Group Owner - An admin or moderator user who has assembled other users together into a group.

HTTPS - Hypertext Transfer Protocol Secure is the standard web transfer protocol with a layer of security for the data in transit to be transmitted through securely.

Ingredient - A single food item that can be purchased at a grocery store.

Interface - Typically a visual element that a user interacts with to accomplish some task.

Meal Plan - A set of recipes assigned to particular days or time of day, as in a scheduled family menu.

Personal Recipe - A recipe that is created by an individual user. It can only be viewed by the creator of the recipe unless the user specifically shares it.

Private Group - A group restricted to invited users only whose content and collaborative work is unseen by public or outside groups.

Quantized Ingredient - An ingredient with a specification of the quantity (mass, volume, or count) required for a specific recipe.

Rating - A scale of 1-5 that shows user satisfaction for a certain recipe, menu, or menu plan.

Recipe - A recipe is a plan for food preparation. In the application, a recipe comprises a title, a list of quantized ingredients, preparation steps, and a photograph.

Review - A users critical assessment of a recipe, menu, or meal plan.

Shopping List - A list of quantized ingredients required to prepare one or more recipes.

Tags - Short, descriptive word or reference by which whatever contains it can be grouped or filtered by, like a custom category or social media hash-tag.

TLS - Transport Layer Security is a protocol that provides communication securely between the client and database over the internet.

User - A person who utilizes the provided software application.

2 References

[1] J. Murray, “How Copyright Works with Social Media,” The Balance Small Business, 31 Jan 2019 [Online]. Available: https://www.thebalancesmb.com/copyrights-and-social-media-issues-397821

[2] J. De Groot, “What is the General Data Protection Regulation? Understanding & Complying with GDPR Requirements in 2019,” Digital Guardian, 15 July 2019 [Online]. Available: https://digitalguardian.com/blog/what-gdpr-general-data-protection-regulation-understanding-and-complying-gdpr-data-protection

[3] U.S. Copyright Office “Compendium of U. S. Copyright Office practices 3rd Edition,” 29 Sept 2017 [Online]. Available: https://www.copyright.gov/comp3/docs/compendium.pdf

[4] Color Blind Awareness (n.d.) [Online]. Available: http://www.colourblindawareness.org/colour-blindness/types-of-colour-blindness/

3 Specific Requirements

3.1 External Interfaces

3.1.1 User Interfaces

3.1.1.1 User Interface Inputs

3.1.1.1.1 Search bar

3.1.1.1.2 User Group Formation (RR-0, RR-60 - RR-61)

Refer to section (3.2.2 Creating Groups) for details.

3.1.1.1.3 User Recipe Creation (RR-0, RR-56, RR-57)

Refer to section (3.2.3 Creating Recipes) for details.

3.1.1.1.4 Generate Shopping List (RR-50 - RR-51)

Refer to section (3.2.13 Shopping List) for details.

3.1.1.1.5 Meal Planning (RR-46, RR-47)

Refer to section (3.2.12 Menu Planner) for details.

3.1.1.1.1.1 Functionality

The search bar shall allow users to search database of recipes and return results based upon multiple SEO methods to grab any/all relevant data. (RR-19 - RR 21, RR-23)

3.1.1.1.1.2 Convenience Features

The search bar shall use autofill and quick key searches. (RR-19 - RR 21, RR-23)

3.1.1.2 User Interface Outputs

3.1.1.2.1 Recipe Database

3.1.1.2.1.1 Access

The User shall be able to access theirs, shared or public recipe information stored on the database indirectly through the application. (RR-9 - RR-16)

3.1.1.2.1.2 Add and Update

The database shall allow recipes to be added and updated as requested by the user.

3.1.1.2.1.3 Sections

The database shall have sections available for customization for the user to store favorite, edited, or customized recipes.

3.1.2 Hardware Interfaces

3.1.2.1 Hardware Interface Outputs

3.1.2.1.1 Shareable Recipes (RR-56 - RR 58)

The application shall allow exporting of recipes in multiple formats(.pdf, .docx, .jpeg, .txt) in order to share or print the recipe that the user has retrieved from the application database.

3.1.3 Software Interfaces

3.1.3.1 Software Interface Inputs

3.1.3.1.1 List of Used API's (actual API undecided) (RR-6)

3.1.3.1.1.1 United States Recommended Daily Allowance API (RR-40)

The United States Recommended Daily Allowance API shall be used to provide nutritional analysis and recommended amounts of ingredients for recipes. This API can be found at https://www.ers.usda.gov/developer/api-terms-of-service/

3.1.3.1.1.2 Recipe Card API (RR-6)

The Recipe Card API shall be able to sort and retrieve recipes based on the search requirements defined in (RR-19 through RR-26)

3.1.3.1.1.3 Authentication/Security API (RR-6)

The Authentication/Security API shall be able to store, encrypt, and retrieve personal user information such as name, email, and password using best security protocols. Auth0 can be a potential API used for this requirement. https://auth0.com/docs/api/info

3.1.3.1.1.4 Ingredients parsing API (RR-21)

The Ingredient Parsing API shall be able to store and manipulate ingredients for user saved recipes. The user shall also be able to search this API for ingredients and or measurements defined in their recipes. Zestful can be a potential API used for this requirement. https://zestfuldata.com/

3.1.3.1.1.5 Google Calendar API (RR-48)

Google's API shall be used to create application calendar and shared calendar interfacing of menus and meal plans. This API can be found at https://developers.google.com/calendar

3.1.3.2 Software Interface Outputs

3.1.3.2.1 Application Interaction with Server Database (RR-1)

3.1.3.2.1.1 The Application shall be able to store users personal recipes and account information such as name, email, and password on server database using third-party authentication service (i.e. Auth0). (RR-1)

3.1.3.2.1.2 The Application shall be able to retrieve users recipes stored on the server database. (RR-1)

3.1.3.2.1.3 The Cookbook Application users shall be able to modify and delete their own recipe data as well as their own personal information stored on the server database. (RR-1)

3.1.4 Communications Interfaces

3.1.4.1 Application Interaction with the Users Device Calendar (RR-48)

3.1.4.1.1 The Application shall be able to create events using Google Calendar's API.(RR-48)

3.1.4.1.2 The Application shall be able to share events created through the Google Calendar's API with other Cookbook application users. (RR-48)

3.2 Functions

3.2.1 Platform

3.2.1.1 The Application shall be internet/web based.(RR-1)

3.2.1.2 The Application shall utilize best practices of responsive web design.(RR-1)

3.2.1.3 The Application shall utilize a cross-browser framework/library which supports at a minimum the last two versions of the following browsers: Google Chrome, Safari, Firefox, Opera, and Edge, and shall support at a minimum Internet Explorer 11. (RR-1)

3.2.2 Creating Groups

3.2.2.1 Users shall be able to follow other users. (RR-0, RR-61)

3.2.2.2 Users shall be able to collaborate with other users. (RR-0, RR-60)

3.2.2.3 Users shall be able to create private groups. (RR-0, RR-57)

3.2.2.3.1 The creator of a group shall be the group owner and shall have control over the group users, having the ability to add and delete users to and from the group, and control over the functionality of the group, including user menu and meal plan creation, editing, and sharing permissions. (RR-0)

3.2.2.3.2 Group owners shall be able to have a view only or an edit option for each added user Group owner shall assign permissions to invited users which shall control viewable content and levels of editing authority. (RR-0)

3.2.3 Creating Recipes

3.2.3.1 Users shall be able to create online recipes that can be posted and shared with other users. (RR-0, RR-56, RR-57)

3.2.3.1.1 A newly created recipe (including twists on an existing recipe) shall be added to a user’s personal cookbook page.

3.2.3.2 The recipe publishing interface shall require the user to create the recipe by inputting values for recipe attributes as noted in "Data Entities and Their Relationships" (3.5.4). (RR-6, RR-7, RR-9, RR-10, RR-12, RR-14, RR-17)

3.2.3.2.1 When a user provides a URL for a recipe on a third-party website, the application shall retrieve the content from the URL, parse the recipe title and quantized ingredients, and create a new recipe with the title and quantized ingredients obtained from the third-party website. (RR-5, RR-7)

3.2.3.2.2 To protect against abuse of the application and other users, the algorithm developed from the "Moderation" (3.2.9) requirement shall be called upon recipe attribute input by the user. (RR-7, RR-8, RR-35)

3.2.4 Sharing Recipes

3.2.4.1 Users shall share recipes with other users via a contextual share button which will provide the user with public, private, and group sharing options. (RR-0, RR-56, RR-57)

3.2.4.2 In a group/collaborative setting, all participants shall have viewing access to recipes shared in their group/page, with edit and delete access (on by default) set by the group owner. (RR-0)

3.2.4.2.1 Recipes in a group page shall retain their public privacy setting. If public, it doesn't matter if it's in a group: everyone can see it. If private, only group members can see it. (So groups grant special, customized access). (RR-57)

3.2.5 Editing Recipes

3.2.5.1 Users shall be able to edit/delete their own uploaded recipes. (RR-10)

3.2.5.2 Users shall be able to customize another user's recipe and save it as their own. (RR-9)

3.2.5.3 In a group/collaborative setting, all participants shall be able to see and edit recipes shared with them at the same time (here, edit means to edit the original recipe, as if the group members were the same user updating one of their own recipes). (RR-0)

3.2.6 Calories/Nutrition Information

3.2.6.1 Application shall provide caloric and nutritional information per recipe based on the ingredients and quantities provided. (3.1.3.1.1.1, RR-40)

3.2.6.1.1 Nutritional information shall include that which is standard on all nutrition labels and shall be displayed in each individual recipe on a per-serving basis.

3.2.6.1.2 Nutritional information of each ingredient shall be obtained through a nutrition data API as explained in 3.2.14.

3.2.7 Ratings

3.2.7.1 Recipes shall allow users to give a written review (including a 5-star rating) to rate the recipe. (RR-18, RR-28)

3.2.7.1.1 Users shall be able to rate a given recipe by means of a 5-star UI element in the review submission interface. (RR-18)

3.2.7.1.1.1 The UI element shall display the average star rating next to it. (RR-18)

3.2.7.1.1.1.1 The 5-star UI element shall also be displayed on the recipe list UI element. (RR-18)

3.2.7.1.2 Database shall reflect that an individual recipe can have a collection of reviews submitted by users, along with an average star rating. (RR-18)

3.2.7.1.3 This function shall be a filter by which recipes can be viewed/sorted/searched. (RR-28)

3.2.8 Portion Adjustment

3.2.8.1 Application shall provide a calculator to each recipe to adjust portion size and count based on provided ingredient quantities. (RR-39, RR-46, RR-47)

3.2.8.1.1 The user shall input a number of people being served, and the quantity/measures of each ingredient will automatically be calculated and adjusted in the recipe. (RR-39)

3.2.8.1.1.1 Changing the portion (of a public recipe) shall not sync across any other browser instances. It changes only for the purposes of the individual viewer in-browser. Otherwise, the setting (and ingredient quantities) shall be displayed initially as the default input by the recipe publisher.

3.2.8.1.2 In a shared list or collaborative group setting, the portion adjuster element shall update for every person participating so that all participants see the updated ingredient quantities. (RR-0)

3.2.9 Moderation

3.2.9.1 Application shall implement an algorithm to prevent or delete recipes that use inappropriate language or ingredients, as well as prevent duplicate recipes. (RR-11, RR-34, RR-35)

3.2.9.1.1 By means of a “black list” of predetermined prohibited words and ingredients, the Algorithm shall recognize inappropriate language and ingredients that are prohibited when entered by user. (RR-34)

3.2.9.1.1.1 Algorithm shall also recognize recipes that attempt to use the exact same ingredients and quantities as duplicates.(RR-35)

3.2.9.1.1.2 Similar to a “username already taken” error when trying to create a username for a given web service, this function shall inform the user when they have duplicated an existing recipe or have input prohibited content, then prompt them to alter their recipe in order to post it. (RR-34, RR-35)

3.2.9.1.1.3 The user shall then decide to revise or discard the recipe draft. (RR-9, RR-10, RR-11)

3.2.10 Search Optimization

3.2.10.1 Application shall provide various methods by which lists of recipes can be filtered, sorted, and searched. (RR-19, RR-20, RR-21, RR-22, RR-23, RR-24, RR-27)

3.2.10.2 Essential methods include filtering/sorting/searching by:

3.2.10.2.1 Title (RR-19)

3.2.10.2.2 Author (RR-20)

3.2.10.2.3 Ingredients(RR-21)

3.2.10.2.4 Keyword (RR-23)

3.2.10.2.5 Preparation Time/Cooking Time(RR-24)

3.2.10.2.6 Tags (RR-27)

3.2.10.2.7 Rating (RR-28)

3.2.11 Customization

3.2.11.1 Application shall provide preferential controls, including:

3.2.11.1.1 Recipe list view filtering/sorting. (RR-19 - RR-32)

3.2.11.1.2 Settings to set the privacy of each recipe to either public, followers, or private. (RR-56, RR-57)

3.2.11.1.2.1 These privacy settings do not apply to group members. Members of the same group can see everything in the group.

3.2.11.1.3 Adding tags to recipes for custom category creation. (RR-27)

3.2.11.1.4 Settings for Group/Shared Recipe List controls including add, delete, and edit recipe privileges, as well as group member adding/removing privileges. (RR-0)

3.2.11.2 User's cookbook (saved/created recipes page) shall allow the user to sort/view recipes in various ways set forth in 3.2.10.2. (RR-30, RR-31, RR-32)3.2.11.3 User shall be able to customize discovered recipes through the recipe "editing" interface. (RR-9, RR-10)

3.2.11.3.1 This function shall save the "new" recipe to the user's cookbook, but shall not alter the recipe belonging to another user. (RR-9)

3.2.11.4 Notes - Users shall be able to attach personal notes to saved recipes. These notes can be comments, adjustments, conversions, warnings, etc. (RR-14, RR-15, RR-16)

3.2.11.4.1 The app shall allow the user to create notes in the app.(RR-14)

3.2.11.4.2 The app shall allow the user to edit notes in the app. (RR-15)

3.2.11.4.3 The app shall allow the user to delete notes in the app. (RR-16)

3.2.11.5 Tags - User shall be able to add tags or hashtag-style references in a recipe that allows for searching according to tags.(RR-17)

3.2.11.5.1 Users shall be able to enter a tag in a search input to view recipes with that tag.(RR-27)

3.2.11.6 Media - Users shall be able to attach media, such as videos or images to recipes. (RR-12)

3.2.11.6.1 The app shall allow the user to create media in the app. (RR-12)

3.2.11.6.2 The app shall allow the user to delete media in the app. (RR-13)

3.2.12 Menu Planner

3.2.12.1 Application shall allow users to add a recipe to daily, weekly, and monthly meal plans and shall also have the option to add the meal plans ingredients to a shopping list. (RR-46, RR-47)

3.2.12.1.1 The daily/weekly/monthly menu/meal plan can be created, edited and deleted.

3.2.12.2 Menu and Meal Plan items shall have Google Calendar API date data assigned to them and shall be exportable to a shared Google Calendar. (RR-0, RR-48, 3.2.12.1.1, 3.1.3.1.1.6)

3.2.12.3 In a group/collaborative setting, multiple users shall be able to add recipes to a shared menu plan and see all recipes added to the plan by others.(RR-0)

3.2.13 Shopping List

3.2.13.1 Application shall form a grocery list to fulfill the requisite ingredients from a given recipe when the user presses a button such as: “Add to shopping list!” (RR-50)

3.2.13.1.1 Shopping List items shall have a delete/remove function to check off or remove items that have been purchased.

3.2.13.2 This function shall also be available from the Menu Planner interface, creating a grocery list from the whole day’s/week’s menu options. (RR-51)

3.2.13.3 In a group/collaborative setting, all participants shall have real-time access to the shared shopping list and be able to: (RR-0)

a) Check off items

b) Uncheck items

c) Add items that are not associated with a meal plan or recipe

d) Delete items (If given privileges by group owner to remove items)

3.2.14 Nutrition Data API

3.2.14.1 Upon the addition of an ingredient by the user into their recipe “draft”, an API call shall be made that returns the nutrition data (daily value items & calories) to the user’s browser. Once the user publishes their recipe, it’s nutrition section shall be populated with this data. (RR-40)

Different nutrition APIs can be used. This is discussed in 3.1.3.1.1.1.

3.3 Usability Requirements

3.3.1 Understandable Purpose

From a focus group of first time users, 95% of them shall understand the purpose of the app within 20 minutes. (RR-1)

3.3.2 Satisfaction

From a focus group of users that have tried the app at least once, 70% of them shall find it useful and pleasant. (RR-1)

3.3.3 Ease of Learning for New Users

From a focus group of first time users, 85% of them shall be able to find and understand individual features within the app. (RR-1)

3.3.4 Ease of Learning for Experienced Users

From a focus group of users with previous cooking app experience, 95% of them shall be able to find and understand individual features within the app.(RR-1)

3.3.5 Rememberability

From a focus group of users that have not used the app for 7 days, the user shall be able to perform specific tasks with 95% accuracy.(RR-1)

3.3.6 Efficiency Over Time

From a focus group of users who have used the app at least once weekly for three weeks, the user shall be able to perform specific tasks with 98% accuracy.(RR-1)

3.3.7 Efficiency in Finding Recipes

From a focus group of users, 85% of them shall be able to find a specific recipe in under 3 minutes. (RR-19) (RR-20)

3.3.8 Efficiency in Meal Planning

From a focus group of users, 85% of them shall be able to complete a weekly meal planning session in under 20 minutes. (RR-63)

3.3.9 Efficiency in Editing Recipes

From a focus group of users, each of them shall be able to edit(add, edit, delete) a specific recipe with 90% accuracy.(RR-10)

3.3.10 Efficiency in Sharing Recipes

From a focus group of users, 85% of them shall be able to share a specific recipe in under 1 minute. (RR-56) (RR-57)

3.4 Performance Requirements

3.4.1 Server-side

3.4.1.1 Number of Users

3.4.1.1.1 Number of Users per Group

The application shall support at least 5 users per recipe group (RR-62)

3.4.1.1.2 Number of Users in Total

The application shall support at least 1,000 users (RR-6)

3.4.1.2 Number of Groups

3.4.1.2.1 Number of Groups per User

The application shall support up to 50 groups per user (RR-0)

3.4.1.3 Number of Recipes

3.4.1.3.1 Recipes Created per User

The application shall support a user to create up to 50 recipes (RR-0)

3.4.1.4 Application Transactions

3.4.1.4.1 Simultaneous Transactions

The application shall support up to any 125 simultaneous (25 users x 5 transactions per user per minute) transactions per minute per group (RR-62)

3.4.2 User Authentication

3.4.2.1 Response Time

3.4.2.1.1 The application shall take 5 seconds or less of server time to authenticate users as measured from the time the authentication request is received by the server until the time the application has generated a response for the client. (RR-0) (RR-62)

3.4.3 Website (RR-1) (RR-62)

3.4.3.1 Application Load Time

3.4.3.1.1 Loading Time for Text

The application shall load text within 5 seconds or less from the time the request has been made to the server until the time the request for text has been received. (RR-62)

3.4.3.1.2 Loading Time for Media

The application pages shall load media assets in 30 seconds or less during peak traffic times. (RR-12) (RR-62)

The application pages shall load media assets in 5 seconds or less during low traffic times. (RR-12) (RR-62)

3.4.3.2 Video Playback

3.4.3.2.1 Video Play Streaming

A video playback shall stream with less than 10% buffer time at 1080p resolution with a stable internet connection. (RR-12) (RR-62)

3.4.4 Searching (RR-30)

3.4.4.1 Response Time

3.4.4.1.1 Response Time for Search by Recipes

When a search query of recipes request is received by the server, the results response shall take 2 seconds or less to generate for the client. (RR-62)

3.4.4.1.2 Response Time for Search by Ingredients

When a search query of recipes by ingredient(s) request is received, the results response shall take 2 seconds or less to generate for the client. (RR-6) (RR-62)

3.4.4.1.3 Response Time for Search by Allergy/Diet

When a search query of recipes by allergy and/or diet preference(s) is received, the results response from the server shall take 2 seconds or less to resolve. (RR-62) (RR-63)

3.4.4.1.4 Response Time for Search by Username

When a search query of users by username is received, the results response from the server shall take 2 seconds or less to resolve. (RR-62) (RR-20)

3.4.5 Loading (RR-1)

3.4.5.1 Response Time

3.4.5.1.1 Response Time for Loading Recipe from Search

The loading of a selected recipe from search shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client. (RR-62) (RR-19)

3.4.5.1.2 Response Time for Loading User Profile from Search

The loading of a user profile from search shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client. (RR-62) (RR-20)

3.4.5.1.3 Response Time for Loading Bookmarked Recipe

The loading of a bookmarked recipe shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client. (RR-62) (RR-0)

3.4.5.1.4 Response Time for Loading Selected Interface

The loading of a selected interface shall take 1 second or less as measured from the time the request is received by the server until the response is ready for the client. (RR-62)

3.4.6 Saving (RR-1)

3.4.6.1 Response Time

3.4.6.1.1 Response Time for Saving Recipe

The saving of a recipe shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client. (RR-62) (RR-06)

3.4.6.1.2 Response Time for Saving Media

The saving of media shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client. (RR-12) (RR-62)

3.4.6.1.3 Response Time for Saving Bookmark

The saving of a bookmark for a given recipe shall take 2 seconds or less as measured from the time the request is received by the server until the response is ready for the client.(RR-12)

3.4.7 Sharing

3.4.7.1 Response Time

3.4.7.1.1 Response Time for Sharing Recipe

The sharing functionality of a recipe to group members shall take 2 seconds or less to resolve. (RR-57) (RR-62)

3.4.8 Grocery List Generation

3.4.8.1 Response Time

The generation of a grocery list shall take 10 seconds or less. (RR-57) (RR-62)

3.4.9 Planning

75% of users shall complete their meal planning for a week in 20 minutes or less.(RR-63) (RR-62)

3.5 Logical Database Requirements

3.5.1 Types of Information Used by Various Functions

3.5.1.1 Data should be stored in the following formats:

3.5.1.1.1 Varchar (Strings)

3.5.1.1.2 Int (Whole numbers)

3.5.1.1.3 Float (Decimals)

3.5.1.1.4 Media types (links to videos and images)

3.5.1.2 The database shall be built using a CRUD relational database approach.

3.5.1.2.1 The database shall allow the ability to CREATE the following:

3.5.1.2.1.1 Create Recipes (RR-6)

3.5.1.2.1.2 Import Online Recipes (RR-7)

3.5.1.2.1.3 Create Media (RR-12)

3.5.1.2.1.4 Create Notes (RR-14)

3.5.1.2.1.5 Daily Menu (RR-46)

3.5.1.2.1.6 Weekly Menu (RR-47)

3.5.1.2.1.7 Shopping List – Generate from Recipe (RR-50)

3.5.1.2.1.8 Shopping List – Generate from Menu (RR-51)

3.5.1.2.2 The database shall allow the user to READ/SEARCH for the following:

3.5.1.2.2.1 Search by Title (RR-19)

3.5.1.2.2.2 Search by Author (RR-20)

3.5.1.2.2.3 Search by Ingredients (RR-21)

3.5.1.2.2.4 Search by Keyword (RR-23)

3.5.1.2.2.5 Filter by Tag (RR-27)

3.5.1.2.2.6 Filter by Rating (RR-28)

3.5.1.2.2.6 Daily Menu (RR-46)

3.5.1.2.2.7 Weekly Menu (RR-47)

3.5.1.2.3 The database shall allow the user to update/customize the following:

3.5.1.2.3.1 Customize Recipes (RR-9)

3.5.1.2.3.2 Update Recipes (RR-10)

3.5.1.2.3.3 Update Notes (RR-15)

3.5.1.2.3.4 Tag Recipe (RR-27)

3.5.1.2.3.5 Scale Rating (RR-28)

3.5.1.2.3.6 Daily Menu (RR-46)

3.5.1.2.3.7 Weekly Menu (RR-47)

3.5.1.2.4 The database shall allow the user to DELETE the following:

3.5.1.2.4.1 Delete Recipes (RR-11)

3.5.1.2.4.2 Delete Media (RR-13)

3.5.1.2.4.3 Delete Notes (RR-16)

3.5.1.2.4.4 Daily Menu (RR-46)

3.5.1.2.4.5 Weekly Menu (RR-47)

3.5.2 Frequency of Use

3.5.2.1 The application shall allow on demand use of the following functions:

3.5.2.1.1 Create Recipes (RR-6)

3.5.2.1.2 Import Online Recipes (RR-7)

3.5.2.1.3 Customize Recipes (RR-9)

3.5.2.1.4 Update Recipes (RR-10)

3.5.2.1.5 Delete Recipes (RR-11)

3.5.2.1.6 Create Media (RR-12)

3.5.2.1.7 Delete Media (RR-13)

3.5.2.1.8 Tag Recipe (RR-27)

3.5.2.1.9 Search by Title (RR-19)

3.5.2.1.10 Search by Author (RR-20)

3.5.2.1.11 Search by Ingredients (RR-21)

3.5.2.1.12 Search by Keyword (RR-23)

3.5.2.1.13 Filter by Tag (RR-27)

3.5.2.1.14 Filter by Rating (RR-28)

3.5.2.1.15 Sort by Title (RR-30)

3.5.2.1.16 Sort by Author (RR-31)

3.5.2.1.17 Sort by Ingredients (RR-32)

3.5.2.1.18 Monitor Duplicate Recipes (RR-35)

3.5.2.1.19 Estimated Calories/ Nutritional Information (RR-40)

3.5.2.1.20 Daily Menu (RR-46)

3.5.2.1.21 Weekly Menu (RR-47)

3.5.2.1.22 Shopping List – Generate from Recipe (RR-50)

3.5.2.1.23 Shopping List – Generate from Menu (RR-51)

3.5.2.1.24 Share Recipes Publicly (RR-56)

3.5.2.1.25 Share Recipes Privately (RR-57)

3.5.2.1.26 Create Notes (RR-14)

3.5.2.1.27 Update Notes (RR-15)

3.5.2.1.28 Delete Notes (RR-16)

3.5.2.1.29 Scale Rating (RR-28)

3.5.2.1.30 Adjust for Serving Size (RR-39)

3.5.2.1.31 Publish to Calendar (RR-48)

3.5.3 Accessing Capabilities

3.5.3.1 Private User and Private Recipe

3.5.3.1.1 Create Recipe

Users shall be able to create a new recipe from within the app itself through the publishing interface and enter the necessary information such as the following: author, title, ingredients, and directions (RR-6).

3.5.3.1.2 Update Recipe

Users that are authors of a recipe shall be able to select the editing interface and edit and save changes to their recipes as they deem necessary (RR-10).

3.5.3.1.3 Delete Recipe

Users that are authors of a recipe shall be able to delete the recipe in its entirety by selecting the editing interface (RR-11).

3.5.3.1.4 Delete Media

Users shall be able to delete or unattach media for recipes they have authored by selecting the editing interface (RR-13).

3.5.3.2 Private user and Shared Recipes

3.5.3.2.1 Customize Recipe

Users shall be able to create a version of a recipe they have authored or a recipe that is shared with them be selecting the editing interface of a recipe, so they are able to customize recipes to their needs (RR-9).

3.5.3.2.2 Create Note

Users shall be able to add notes to recipes they have authored or that are shared with them by selecting the editing interface (RR-14).

3.5.3.2.3 Update Note

Users shall be able to update or change existing notes on recipes they have authored, or are shared with them by selecting the editing interface (RR-15).

3.5.3.2.4 Delete Note

Users shall be able to delete a note for recipes they have authored or recipes that are shared with them through the editing interface of that recipe (RR-16).

3.5.3.2.5 Search and Filter

Users shall be able to Search or filter recipes by recipe title, author, keyword, tag, or rating etc. When they are searching or viewing recipes that are shared with them (RR-19, RR-20, RR-21, RR-23, RR-27, RR-28).

3.5.3.2.6 Sort

Users shall be able to sort recipes by recipe title, author, and main ingredient ect. When they are viewing recipes that are shared with them (RR-30, RR-31, RR-32).

3.5.3.2.7 Adjust for Serving Size and Count

Users shall be able to adjust the recipe's quanity of ingredients according to serving size or count of a recipe. This may be done via the recipe interface (RR-39).

3.5.3.3 Shared Users and Shared Recipes

3.5.3.3.1 Attach Media

Users may attach media in forms of links to video and images stored at other publicly accessible locations (such as YouTube or other similar video hosting services) for recipes they have authored and recipes that are shared with them through the editing interface of that recipe (RR-12).

3.5.3.3.2 Tag Recipe

Users shall be able to add tags to recipes they have authored and recipes that are shared with them via the recipe interface(RR-17).

3.5.3.3.3 Create Daily and Weekly Menu

Users through a menu interface users shall be able to create daily and weekly menu plans by selecting the recipes that are shared with them(RR-46).

3.5.3.3.4 View Daily and Weekly Menu

Users shall be able to view the recipe title and link to the recipe interface when selecting a daily or weekly menu viewing options in the menu interface (RR-46, RR-47).

3.5.3.3.5 Scale Rating

Users shall be able to rate recipes that they have authored or that are shared with them on a number scale via the recipe interface (RR-18).

3.5.3.3.6 Moderation

Users content shall be subjected to a moderation system to block duplicated and inappropriate content in recipes before they are published (RR-35 RR-34).

3.5.3.3.7 Nutritional Facts

User shall be able to view nutritional facts for a recipe they have authored and recipes that are shared with them via the recipe interface (RR-40).

3.5.3.3.8 Create Shopping List From Recipe

Users shall be able to create a shopping list of ingredients from a recipe that they have authored or that are shared with them via the recipe interface (RR-50).

3.5.3.3.9 Create Shopping List From Menu

Users shall be able to create a shopping list from a private or shared menu of recipes via the menu interface (RR-51).

3.5.3.4 Access to a third-party site or feature (Calendar)

3.5.3.4.1 Import Online Recipe

Users shall be able to import a recipe by providing a link to the third-party recipe via the publishing interface(RR-7).

3.5.3.4.2 Export Menus

Users shall be able to contribute to a private or a shared menu and publish them on third-party private or shared calendar via the menu interface (RR-48).

3.5.4 Data Entities and Their Relationships

3.5.4.1 Recipes

A user shall have a one to many relationships with recipes allowing them to be an author and many recipes. (RR-6)

3.5.4.2 Shopping list

A user shall have a one to many relationships with shopping lists. Users should have a private shopping list, a shared shopping list, and a menu shopping list. (RR-50, RR-51)

3.5.4.3 Meal Plans

3.5.4.3.1 Daily Menu Plans

A user shall have a one to many relationships with a daily menu, which should be private daily menu plan and shared daily menus. (RR-46)

3.5.4.3.2 Weekly Menu Plans

A user shall have a one to many relationships with a weekly menu, which should be private weekly menu and shared weekly menus. (RR-47)

3.5.4.4 Groups

3.5.4.4.1 Group Owners

Group owners shall have a one to many relationships of users which should contribute and view recipes with the group (RR-0).

3.5.5 Integrity Constraints

3.5.5.1 Encryption In Motion

Data shall use TLS encryption via the HTTPS protocol when data is being transferred from the client to the database and visa-versa. (RR-34)

3.6 Design Constraints

3.6.1 Development Platform

3.6.1.1 The app shall be developed as a web app capable of working on web browsers. (RR-1)(See section 3.2.1.3 for target browsers.)

3.6.1.2 The app shall use HTML, CSS, and JavaScript for the client side of the app. (RR-1)

3.6.2 Time Constraints

3.6.2.1 For 75% of users, the app shall take no longer than 20 minutes to complete meal planning for a week for a family of eight or fewer people. (RR-63)

3.6.3 Copyright Law

3.6.3.1 The user shall accept an agreement that allows the app to have worldwide, non-exclusive, royalty-free, sub-licensable, transferable license to use, copy, reproduce, process, adapt, modify, publish, transmit, display and distribute recipes in any and all media or distribution methods (now known or later developed).[1]

3.6.3.2 The application shall parse recipe titles and ingredient lists from websites (third-party recipe titles and ingredient lists are not copyrightable under US law). The application shall import preparation instructions, photographs, or other media, because this is copyrightable content.

3.6.4 Budget

3.6.4.1 Parsing recipe ingredients shall cost no more than $0.02 per ingredient.

3.6.4.2 The cost to obtain nutritional information shall be no more than $0.00 per call.

3.6.4.3 The cost to store a single recipe shall be no more than $0.0012 per month.

3.6.4.4 User authentication shall cost no more than $0.00 per user.

3.6.4.5 An application hosting service shall cost no more than $1.00 per day.

3.6.5 Regulatory Requirements

3.6.5.1 The app shall conform to the European General Data Privacy Regulation.[2]

3.6.6 Visuals

3.6.6.1 The app shall have four different visual options (see Figure 2) that cover the three major color blindness deficiencies and a normal option [4].

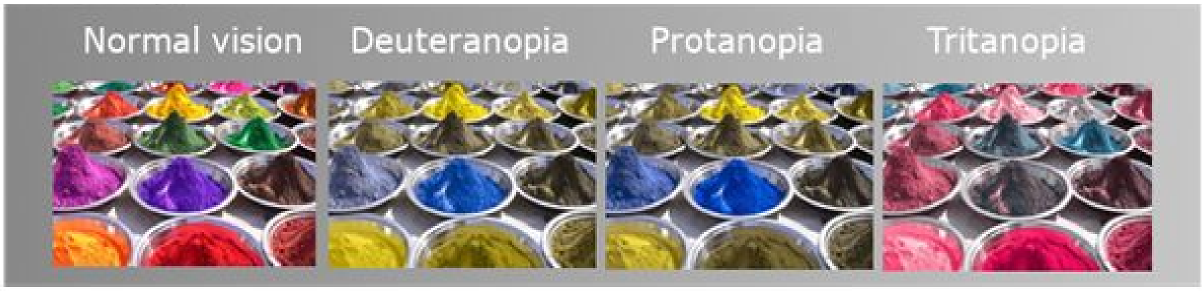


Fig. 2 - Color Deficiencies

3.7 Software System Attributes

3.7.1 Reliability

3.7.1.1 The app shall verify, using JavaScript, that when a user is uploading a recipe, no input fields are left blank. (i.e., The recipes being uploaded have a title, ingredients, and instructions.) (RR-6)

3.7.1.2 The app shall provide an in app notification immediately after the user attempts to upload/share a recipe to verify if the upload/share was successful or not. (RR-56)

3.7.1.3 The app shall open external links in a new tab as not to interrupt the user's progress. (RR-12)

3.7.2 Security

3.7.2.1 Users shall be required to login in order to view personal recipes or upload/share recipes. (RR-1)

3.7.2.2 The app shall use HTTPS to ensure user data confidentiality. (RR-1)

3.7.2.3 The app shall use TLS/SSL to secure communication between the server and the web browser. (RR-1)

3.7.3 Maintainability

3.7.3.1 The app shall provide users with a button on each page where they can submit error reports to the development team by filling out a bug report form. (RR-34)

3.7.3.2 The app shall provide users with a button on each page where they can submit duplicate recipe reports to the development team by filling out a duplicate recipe report form. (RR-35)

3.7.4 Portability

3.7.4.1 Portability concerns for a web application include a list of browsers that will be supported. As this impacts functionality, please refer to section 3.2.1.3.

3.7.4.2 The app shall use CSS media queries in order to adjust the appearance of the web application depending on the device; i.e. mobile phone, or computer. (RR-1)

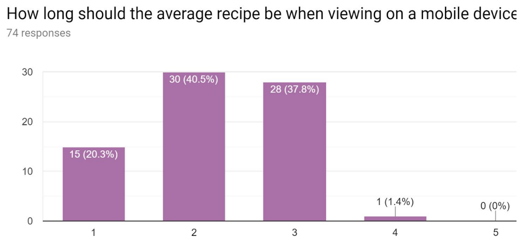
3.8 Supporting Information

3.8.1 General Info

3.8.1.1 The app shall focus on search optimization to allow users to quickly find recipes that meet their desired criteria.

3.8.2 General Questions

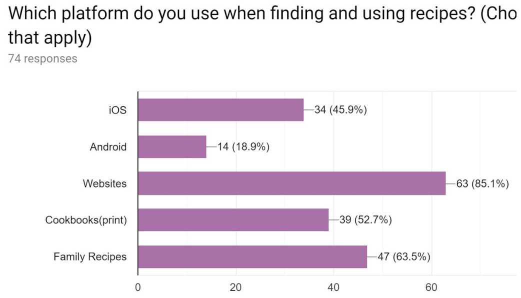
3.8.2.1 A recipe should be 1 to 1.5 pages (of scrolling).(RR-6)



How long should the average recipe be when viewing on a mobile device

74 responses

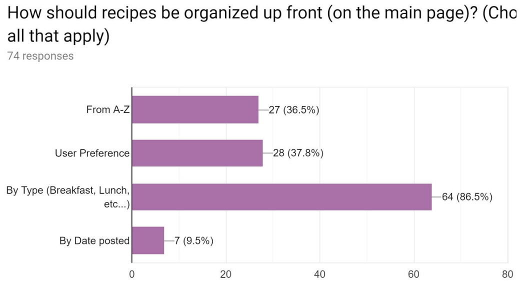
3.8.2.2 Most respondents use web-based platforms to find recipes. (RR-1)



Which platform do you use when finding and using recipes? (Cho that apply)

74 responses

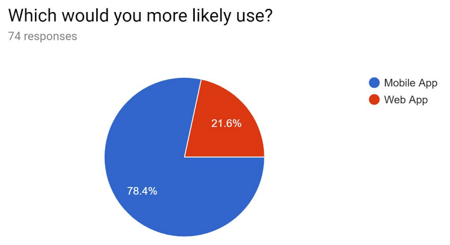
3.8.2.3 Most respondents want recipes organized by meal type, although the organization by user preference is likely preferred. (RR-27)



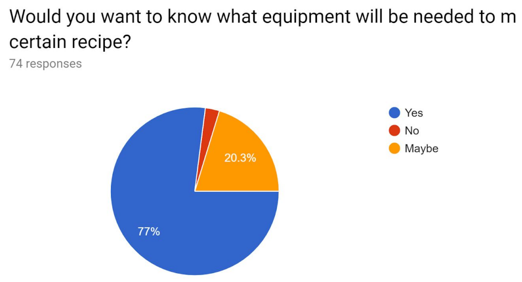
How should recipes be organized up front (on the main page)? (Cho all that apply)

74 responses

3.8.2.4 Most users are more likely to use a mobile app.(RR-2, RR-3)

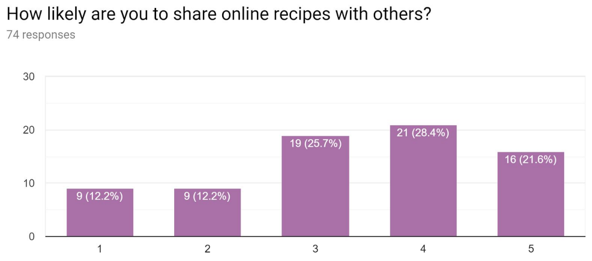


3.8.2.5 Most users want to know the equipment needed for a recipe.(RR-27)

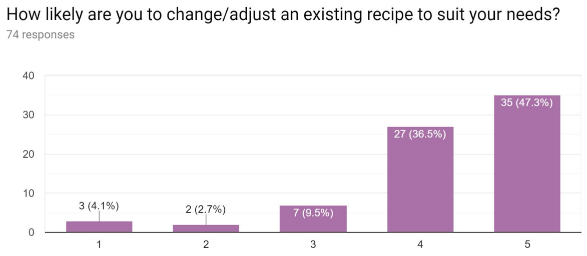


3.8.3 Sharing Questions

3.8.3.1 Most users would like to share recipes online with others. (RR-56, RR-57)



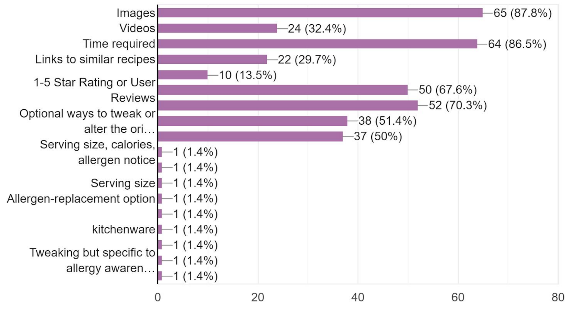
3.8.3.2 Most users would like to tweak and adjust a recipe to suit their own needs. (2-9, 2-10)



3.8.3.3 Respondents thought the top 5 things each recipe should include were: images, time required, ratings/likes, optional ways to adjust the recipe, and a way to share the recipe.(RR-9, RR-12, RR-56)

Which of the following should be included with a recipe? (Select all that apply)

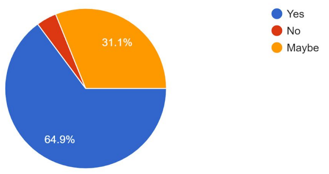
74 responses



3.8.3.4 Most respondents want a digital cookbook (online recipes) to contribute to and share with others.(RR-6, RR-57)

Would you like a digital cookbook to share with your family and friends that you could contribute to?

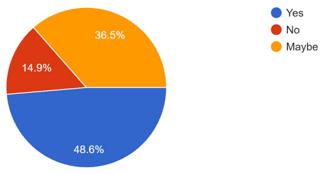
74 responses



3.8.3.5 Most respondents think the developers should censor/moderate/control inappropriate or duplicate recipe postings.(RR-34, RR-35)

Should a recipe site/app be censored/moderated to control inappropriate/duplicate recipe postings?

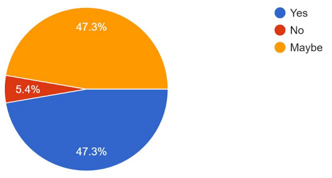
74 responses



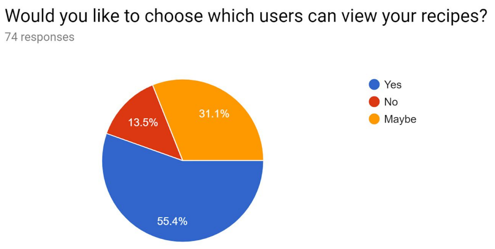
3.8.3.6 Some users are less likely to want others openly allowed to view their recipes.(RR-56, RR-57)

Would you like others to be able to view your recipes?

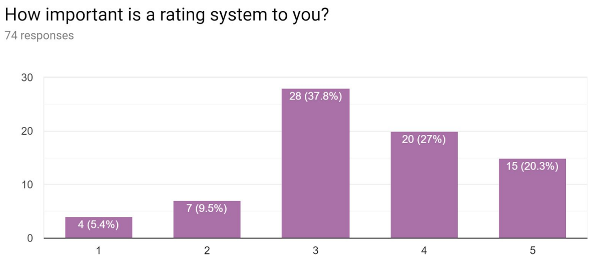
74 responses



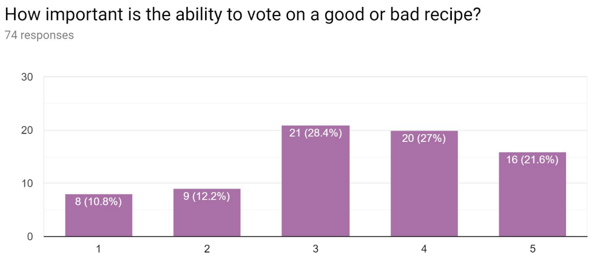
3.8.3.7 More users would like to choose who sees their recipes. (RR-57)



3.8.3.8 More users feel a rating system is important, but not very important.(RR-18)

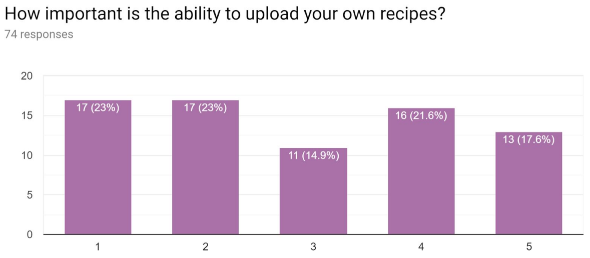


3.8.3.9 More users feel it's important to vote/rate recipes.(RR-18)

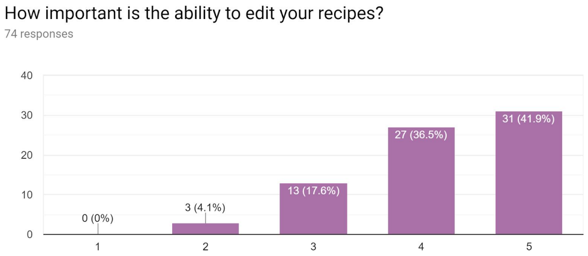


3.8.4 Customization

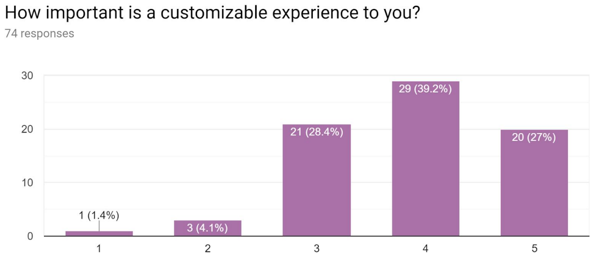
3.8.4.1 A surprising number of respondents actually didn't care about uploading their own recipes. We can infer that they care about "finding" recipes, but not always "making" them.(RR-6)



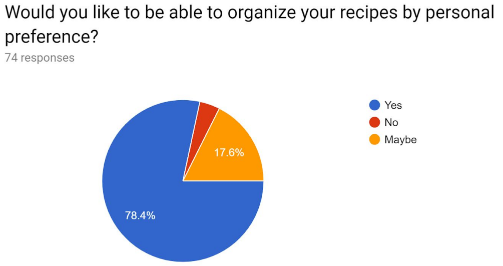
3.8.4.2 Most respondents want to edit their recipes after they've been posted.(RR-9, RR-10)



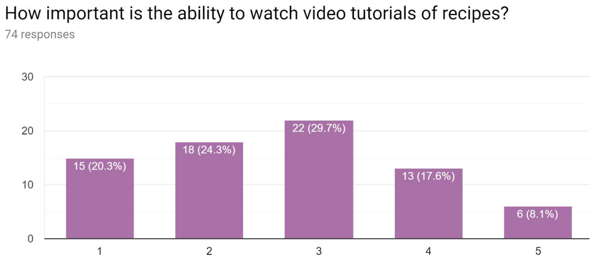
3.8.4.3 Most users want the app/site to be customizable to their needs.(RR-9)



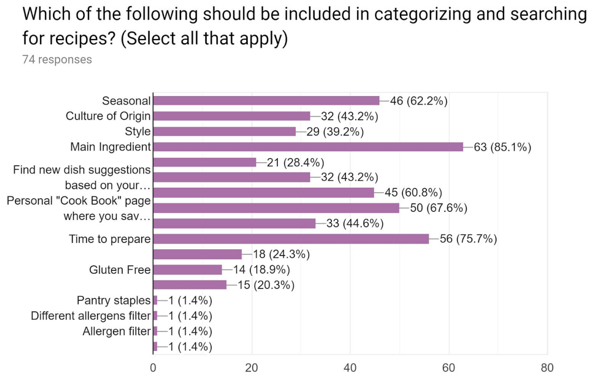
3.8.4.4 Most users would like to organize their recipes according to their preferences.(RR-18)



3.8.4.5 Users did not find a great need to see video tutorials of each recipe.(RR-12)



3.8.4.6 The top 5 things users want to categorize/search by are: main ingredient, preparation time, favorites, ratings, and seasonal.(RR-17, RR-18, RR-21, RR-24, RR-27, RR-28)



Which of the following should be included in categorizing and searching for recipes? (Select all that apply)

74 responses

4 Verification

4.1 Verifying External Interfaces

The verification process for 3.1 External Interfaces shall consist of the testing verification method. The tests shall be conducted by four members of the design team under the direction of the project manager. The Unit.js framework shall be utilized to write unit tests for the various external interface functions and shall insure that all of the APIs and developer-written functions work as intended. The tests will be written in a manner that will have at least 85% code coverage across the code that deals with any external interface. Functions will have many test cases written for them to deal with the multitude of scenarios that they could be subjected to. This rigorous testing will ensure the code written is robust enough to handle potential issues that may arise post-production.

To ensure the product is shipped as complete and bug free as possible, the tests shall be deemed successful if 100% of the unit tests are passed. The product will not be put into production if there is not 85% code coverage and a 100% passing rate among the unit tests.

If these testing constraints are not met, the software design team will be informed of the errors and will be given time to locate the bugs causing the tests not to pass and fix the issues accordingly. For quality control purposes, this process will be repeated until the testing constraints are met and the project manager deems the unit tests complete. This adjustment period shall take no longer than two weeks.

4.2 Verifying Functions

The verification process for 3.2 Functions Requirements shall consist of the Demonstration verification method. The Project Manager shall oversee the execution of the demonstration of all requirements listed in section 3.2 and record the demonstration via Loom. Before the demonstration, the website shall be prepared with mock-up data for the user to use in the demonstration. This preparation shall at least consist of the creation of groups and recipe data with rich detail.

The demonstration shall consist of the user using the website and demonstrating each requirement individually. The project sponsor and at least 200 stakeholders shall witness the demonstration by being emailed a link with the recording, done via Loom, and be given 72 hours to provide feedback on the demonstration. The feedback shall entail whether they felt the demonstration of each requirement listed in section 3.2 is satisfied. It shall also entail their notes on why or why not the requirement was or was not met to their satisfaction.

The demonstration shall be deemed successful if an average of 85 percent of the listed requirements are deemed satisfactory by the witnesses. The average cannot be computed with less than 150 responses and it cannot be computed until the end of the one week response window.

If the 85 percent satisfactory average is not reached the functionality team will have two weeks to make adjustments and the demonstration session will be repeated.

4.3 Verifying Usability Requirements

The verification process for 3.3 Usability Requirements shall consist of the analysis verification method. Usability requirement verification will take place concurrently with development as new features based upon user requirements have been completed and as coordinated by the project manager. When a new set of features has been completed, a simple random sample of 200 stakeholders will be selected to try out the newly implemented features via a test build of the application. The selected stakeholders will be given a survey that has both quantitative and qualitative data to gauge their impressions on the usability of the feature set. The quantitative data will be rankings as to how satisfied they are with given features, while the qualitative section will be their justifications for their rankings and possible suggestions for improvement.

We can be confident that a given feature is acceptable when the mean satisfaction for the feature is above 85% for the focus group as a whole.

If a feature within the set's mean satisfaction among the stakeholders is below 85%, then we know that the design and interface usability of that feature must be reworked. In these instances, the design team will analyze the feedback from the stakeholders as to why the feature was difficult to use and redesign the feature accordingly. The newly redesigned feature will then be included in the next subsequent feature sets until it reaches 85% satisfaction.

4.4 Verifying Performance Requirements

4.4.1

Performance requirements for Section 3.4 shall be verified via the Testing method. The verification shall happen when the performance team deems the database functional. The test shall be run by the project manager and via Loom. The recording will show each functionality described in section 3.4 performing to the specified standards. The project manager shall record the requirements that the system failed to meet and may add additional comments describing why the system may have failed.

The project manager shall calculate the percentage of requirements that the system meet compared to the total number of requirements in the section. The system will be deemed acceptable if this percentage meets or exceeds 85%.

If the percent of requirements that the system meets compared to the total number of requirements in the section does not meet or exceed 85% then the system shall not be deemed acceptable. In this case the performance team shall have two weeks to make the necessary changes for the test to be re-run.

4.4.2

Performance requirements for Section 3.4 shall be verified via the Demonstration method. The verification shall happen after 4.4.1. The test shall be run by the project manager and recorded via Loom. The recording, made with Loom, shall be emailed to the project sponsor and at least 200 stakeholders. A 72 hour response time will be allotted for stakeholders to review the test and provide their feedback. The feedback shall describe whether or not the stakeholders felt that the system meets each requirement specified in section 3.4 and where specifically the system failed to meet the requirements. The stakeholders will also include reasons why the stakeholders felt that the system does not meet the requirements.

An average of the number of requirements that the stakeholders felt the system met will be calculated after the 72 hour window. If the average number of requirements met meets or exceeds 85% of the total number of requirements in the section the system is satisfactory. The average may not be calculated until at least 150 stakeholders has evaluated the system and the 72 hour period has concluded.

If the average number of requirements that the stakeholders felt the system met does not meet or exceed 85%, then the system is not yet satisfactory. In this case the performance team shall have two weeks to make the necessary changes for the test to be re-run.

4.5 Verifying Logical Database Requirements

The verification process for 3.5 Logical Database requirements shall consist of two methods.

4.5.1

The first being the Demonstration Verification Method upon the database being deemed functional by the design team. The Project Manager shall oversee the execution of the demonstration of all requirements listed in section 3.5 and record the demonstration via Loom. The project sponsor and at least 200 stakeholders shall witness the demonstration by being emailed a link with the recording and be given 72 hours to provide feedback on the demonstration. Specifically, the feedback should entail whether they felt the demonstration of each requirement listed in section 3.5 has been met to their individual satisfaction. It shall also entail their notes on why or why not the requirement was met to their satisfaction.

The demonstration shall be deemed successful if an average of 85 percent of the listed requirements are deemed satisfactory by the witnesses. The average cannot be computed with less than 150 responses and it cannot be computed until the end of the 72 hour response window.

If the 85 percent satisfactory average is not reached the design team will have two weeks to make adjustments and the Demonstration will be repeated.

4.5.2

The second verification method will be the Inspection Verification Method, which will be completed simultaneously with the previous method above. For the inspection the development team will compare the behavior of the database directly with the described behavior from the SRS. They will utilize the performance reporting features of the chosen database management system to verify the behavior of the database matches the requirements listed in the SRS.

The verification will be deemed successful once the performance reports indicate the database’s behavior matches 100% with the descriptions in the SRS.

If the test is not successful the development team will be given two weeks to make adjustments before the next test.

4.6 Verification Design Constraints

The verification process for 3.6 Design Constraints shall consist of three different methods.

4.6.1

The first method to be used will the Inspection Verification Method. This method will be used requirements 3.6.1.2, 3.6.3.2, & 3.6.5.1. For requirements 3.6.1.2 and 3.6.3.2 the development team will review the code to ensure that it meets those requirements. For requirement 3.6.5.1, the development team shall consult the European General Data Privacy Regulations to ensure we are following their guidelines in the implementation of our application.

The inspection will be deemed successful when all three requirements are met with 100% completion.

If any of the requirements are not met then the development team will have up to two weeks to make the necessary changes to pass with a 100% completion rate.

4.6.2

The second verification method will be the Demonstration Method, and will be used on requirements 3.6.1.1, 3.6.2.1, 3.6.3.1, & 3.6.6.1. This will be completed after 4.6.1 has passed with a 100%, so that we eliminate the possible need to make changes after we have confirmed that the given features in section 3.6 work. The Project Manager shall oversee the execution of the demonstration of the previously specified requirements in section 3.6 and record the demonstration via Loom. The project sponsor and at least 200 stakeholders shall witness the demonstration by being emailed a link with the recording and be given 72 hours to provide feedback on the demonstration. Specifically, the feedback shall entail whether they felt the demonstration of each of the specified requirements from section 3.2 has been met to their individual satisfaction. It shall also entail their notes on why or why not the requirement was met to their satisfaction.

The demonstration shall be deemed successful if an average of 85 percent of the listed requirements are deemed satisfactory by the witnesses. The average cannot be computed with less than 150 responses and it cannot be computed until the end of the one week response window.

If the 85 percent satisfactory average is not reached the Development team will have two weeks to make adjustments and the demonstration session will be repeated.

4.6.3

The third verification method used will the Analysis Method. It will be used on the requirements listed in section 3.6.4. This method will be completed simultaneously with the first verification method, the inspection method. Using the information provided by the third-party services that are to be used to fulfill these requirements, at least three members of the development team will create a scenario of projected usage of the application to show theoretical compliance with the estimated budget for each requirement. The third-party information, the simulation, and the results shall be presented to at least 200 stakeholders who shall have 72 hours to provide feedback on the analysis. Specifically, the feedback shall entail whether they felt the analysis of each of the specified requirements from section 3.6.4 has been met to their individual satisfaction. It shall also entail their notes on why or why not the requirement was met to their satisfaction.

The analysis shall be deemed successful if an average of 85 percent of the listed requirements are deemed satisfactory by the witnesses. The average cannot be computed with less than 150 responses and it cannot be computed until the end of the one week response window.

If the 85 percent satisfactory average is not reached the Development team will have two weeks to make adjustments and the analysis will be repeated.

4.7 Verification Software System Attributes

The verification process for 3.7 Software System Attributes shall consist of the testing verification method. The test shall be carried out by the Project Manager and three members of the design team once the design team has deemed the requirements in section 3.7 functional. One member of the test team shall test all requirements in 3.7 on a dell desktop computer no older than a year with a standard size monitor. A second member shall test all requirements on Samsung Laptop that is no older than one year, with a standard sized screen to test all of the requirements. A third member shall test all of the requirements for section 3.7 on a 10.2inch Ipad that is no older than a year old. The fourth member shall test the requirements of section 3.7 on a Google Pixel 3 XL 64GB smartphone that is no older than a year.

This test has been specifically designed to ensure all requirements are working, especially, to make sure the design is truly responsive as requirement 3.7.4.2 states. All tests will be recorded via Loom. The links to the Loom recordings will be sent, via email, to the project sponsor and at least 200 stakeholders. They will be given one week to respond. Their responses shall be for each individual test to see if they feel the requirements have been met with detailed notes explaining why or why not they feel those requirements were met for each test.

The test shall be deemed successful if an average of 85 percent of the listed requirements are deemed satisfactory by the witnesses. The average cannot be computed with less than 150 responses and it cannot be computed until the end of the one week response window.

If the 85 percent satisfactory average is not reached the design team will have two weeks to make adjustments and the test will be repeated.

5 Appendices

5.1 Assumptions and Dependencies

5.1.1 Although there are many recipe and meal planning applications, there are few (if any) that provide the group functionality described herein. (3.2.4 3.2.7)

5.1.2 Application is free to users but can be monetized through affiliate income and advertising (3.6.4).

5.1.3 There exist third-party APIs which satisfy the functionality described herein (3.1.3.1.1).

5.1.4 The user will have reliable internet connection (3.6.1).

5.1.5 The user will have access to a computer and know how to use it (3.6.1).

5.1.6 The user will know how to navigate a web browser (3.6.1).

5.2 Acronyms and Abbreviations

5.2.1 API - Application Program Interface

5.2.2 CSS - Cascading Style Sheets

5.2.3 CRUD - Create, Read, Update, Delete

5.2.4 EU - European Union

5.2.5 GDPR - General Data Protection Regulation

5.2.6 HTML - Hypertext Markup Language

5.2.7 HTTPS - Hypertext Transfer Protocol Secure

5.2.8 RAM - Random Access Memory

5.2.9 SSL - Secure Sockets Layer

5.2.10 TCP - Transmission Control Protocol

5.2.11 TLS - Transport Layer Security

5.2.12 ToS - Terms of Service

5.2.13 UDP - User Datagram Protocol

5.2.14 UK - United Kingdom