

# 1. Positioning

## 1.1 Problem statement

The problem of the inefficient organization of recipes affects those looking to plan their meals or cook; the impact of which is inefficient time spent looking for recipes and money spent on unnecessary ingredients.

## 1.2 Product Position Statement

For those who are actively looking for new recipes and cook, FoodWeb is a meal organization app that allows the user to find, organize, and share recipes unlike cookbooks, Google, or social media, our product focuses on an ever-expanding library of recipes as the main content the user interacts with.

## 1.3 Value proposition and consumer segment

Value Proposition: FoodWeb is a meal organization app that allows those looking to plan their meal and cook to find, organize, and share recipes, by providing an ever-expanding library of recipes as the main content the user interacts with.

Consumer Segment: People who actively look for new recipes and cook.

# 2. Stakeholders

Users, clients, competitors, detractors, developers, etc.

Project Manager: Nazhem Jones

Responsible for assigning duties, handling deadlines, and communication between developers.

Application Developers: Tyler Lockwood / Dylan Anderson / Ethan Green / Eduardo Martinez

Develop, plan, implement, and debug specifications needed for the software.

Testers/Users: Zachary Dehaven

Measure and approve the quality of the final product. They will give us demo and test runs

to ensure the product is functional.

Competitors: cookbooks, search engines (google), social media ( instagram / facebook )

Clients: People involved in fitness, people who like cooking, people looking to plan their meals

### 3. Functional requirements (features)

- Database
- Search recipes
- Save recipes
- View/explore recipes
- Create recipes
- Manage recipes
- Remove recipes
- Create account
- Manage account

### 4. Non-functional requirements

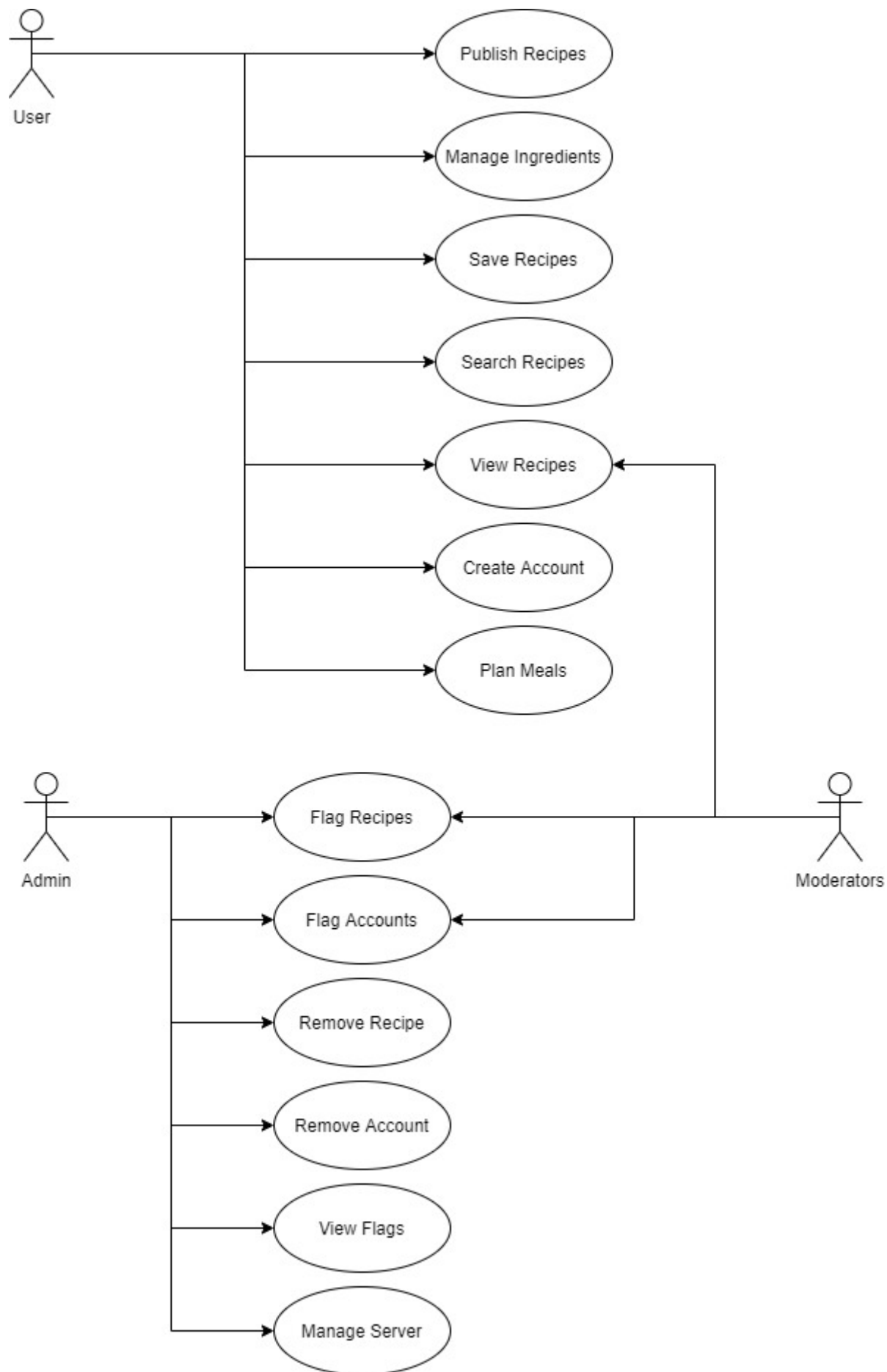
- Usability: users are able to publish and search through recipes without developer help
- Data integrity: security measures to avoid outside use of the system.
- Maintainability: app is able to remain secure and in a good performance as it evolves
- Scalability: assume an increase in data without affecting its speed or performance
- Availability: the system meets a 99.99% uptime
- Performance: ensuring the speed of searches, uploads, and feed in a given time span
- Satisfaction: system performs with a certain precision

### 5. MVP

Our MVP will be an app that will allow users to search, save/favorite recipes, download and upload recipes, and create user profiles in a timely manner. To test upload one person will download the app onto our devices and see if we can upload a recipe. While someone else will try to download that recipe that was uploaded by another team member. To test the effectiveness of the search feature we will look to see how accurate it is while also making it easy to find. To implement and test users profiles we will require an email address, username, and a password. To test we will make sure that we can log back into the system with entered user information. We will try to make all these features easy to use and easy to find.

### 6. Use Cases

#### 6.1 Use case diagram



## 6.2 Use case descriptions and interface sketch

### 1. Remove Account

FoodWeb	Tyler Lockwood
Use-case Specification | Remove Account  
Date | 09/23/2021

**Brief Description:**

the ability to remove accounts whether they be an admin, or user accounts

**Actor Brief Descriptions**

- \* Member: foodWeb user with a foodWeb account
- \* Admin: developer, or specially privileged user who has system access

**Preconditions**

- \* When an account is/must be(ing) deleted

**Post-conditions**

- \* The system deletes the account

**Basic Flow of Events**

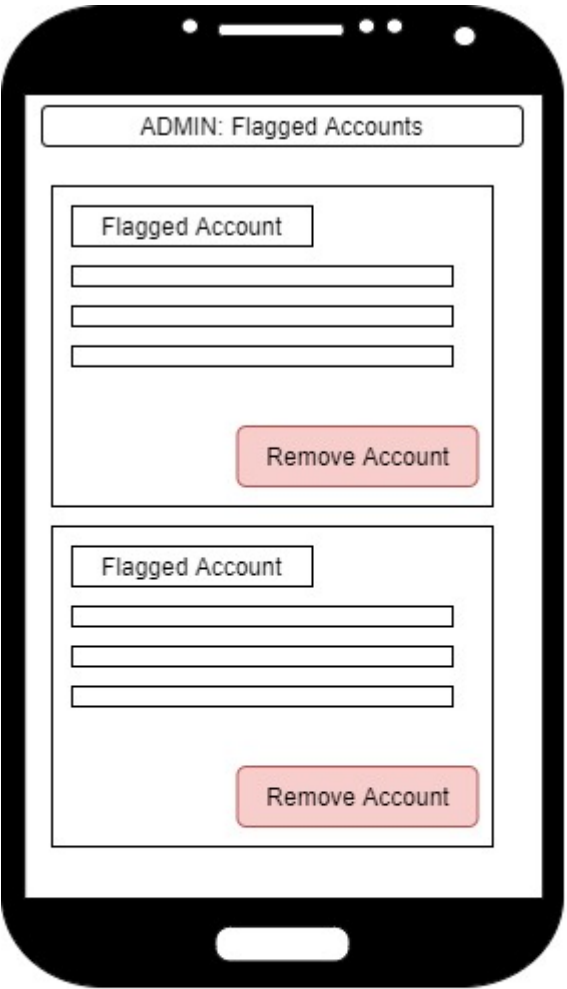
1. The use case begins when a user, or an admin goes to delete their account
2. System gets the notice to delete
3. System asks the user, moderator, or an admin to confirm deletion
4. System gets a notice of confirmation
5. The use case ends

**Alternative Flows**

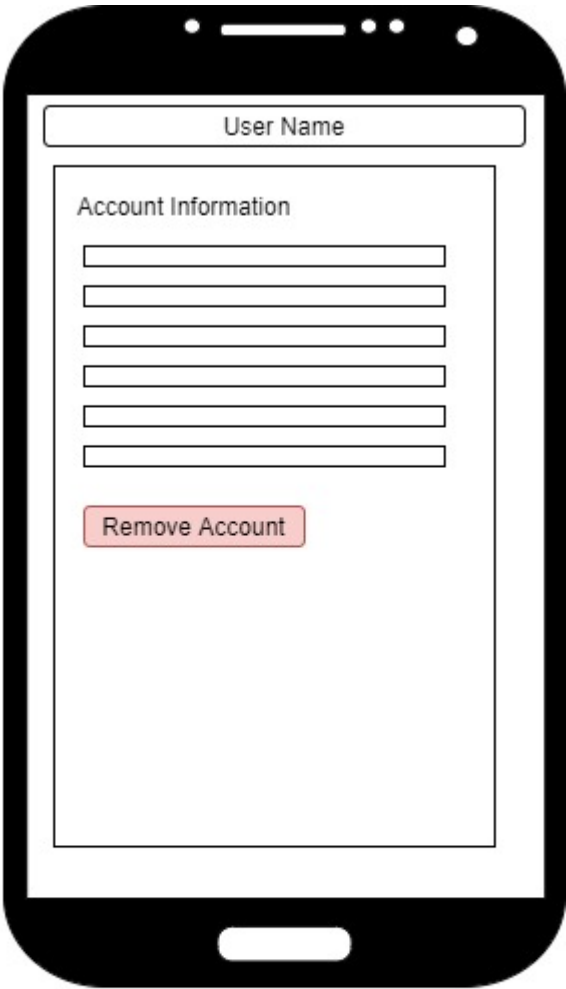
- If in step 3 of the basic flow the user, or an admin does not confirm account for deletion then the system will not get notice of confirmation
1. System will confirm that the user, or an admin is trying to leave account

- deletion
- 2. User, or an admin will confirm if they are leaving account deletion
  - 3. System will get confirmation, and leave account deletion
  - 5. System will not get confirmation, and the use case resumes at basic flow 3

Use Case Sketch



Removing an account as an Admin



Removing an account as a User

2. Publish Recipes

FoodWeb	Eduardo Martinez
Use-case Specification	Publish Recipes
Date	09/27/2021

Brief Description:

the user publishes a recipe to store or share.

#### Actor Brief Descriptions

\* Member: a user who has an account registered within the app

#### Preconditions

\* The user is already registered and logged into the app.

#### Post-conditions

\* The system deletes the account

#### Basic Flow of Events

1. The user selects the option to post a recipe
2. The app displays a template to input recipe information
3. The user fills out the template with related recipe information
4. The user selects an option to publish the recipe
5. The app publishes the recipes and saves it

#### Alternative Flows

1. At any time the user decided to not publish the recipe
  1. The user selects cancel
  2. The app asks to confirm the cancellation
  3. The user confirms
  4. The use case ends
2. At any time the user closes the app while performing a publication
  1. The app clears the template
  2. The app closes the template
  3. the app returns to the home page.

#### Subflows

1. The user selects publish recipe but does not have an account
  1. The user is prompted to create an account
  2. The user creates an account
  3. The user is redirected to publish a recipe

### Key Scenarios

1. The user successfully publishes a recipe
  1. The user gets displayed a recipe information template
  2. The user fills the template out
  3. The user publishes it
2. The user decides not to publish a recipe
  1. The user cancels the operation
  2. The user confirms the cancellation
  3. The system disregards the template
3. The user closes the app while publishing a recipe
  1. The system clears the template
  2. The system closes the template
  3. The system returns to the home page

### Post-conditions

- \* The recipe is published and can be accessed by other users

### Use Case Sketch



3. Create Account

FoodWeb	Zach DeHaven
Use-case Specification:	Create Account
Date:	09/25/2021

Brief Description

The user creates an account with a username and password, to hold their personal information.

Actor Brief Descriptions

\* User: someone using the foodWeb app, who does not have an account

Preconditions



- \* User selects 'Create Account' on landing page

### Basic Flow of Events

1. The user selects 'create account'
2. The user enter a username/e-mail
3. The user enters a password
4. The user confirms password
5. The account is created and saved by the system
6. The user is logged in and able to access the application
7. The use case ends

### Alternative Flows

1. If the user enters a password with weak security quality at step 3
  1. The system notifies the user what the password is missing
  2. The user enters a different password
  3. Continues to step 4
2. The user wants to log-in instead
  1. The user selects "Log In"
  2. The user enters their username/e-mail and password
  3. The systems connects them to their account
  4. The use case ends

### Subflows

1. The user enters an e-mail that's already in use
  1. The user is notified that the e-mail is already in use
  2. The system directs the user to log-in
  3. The use case ends

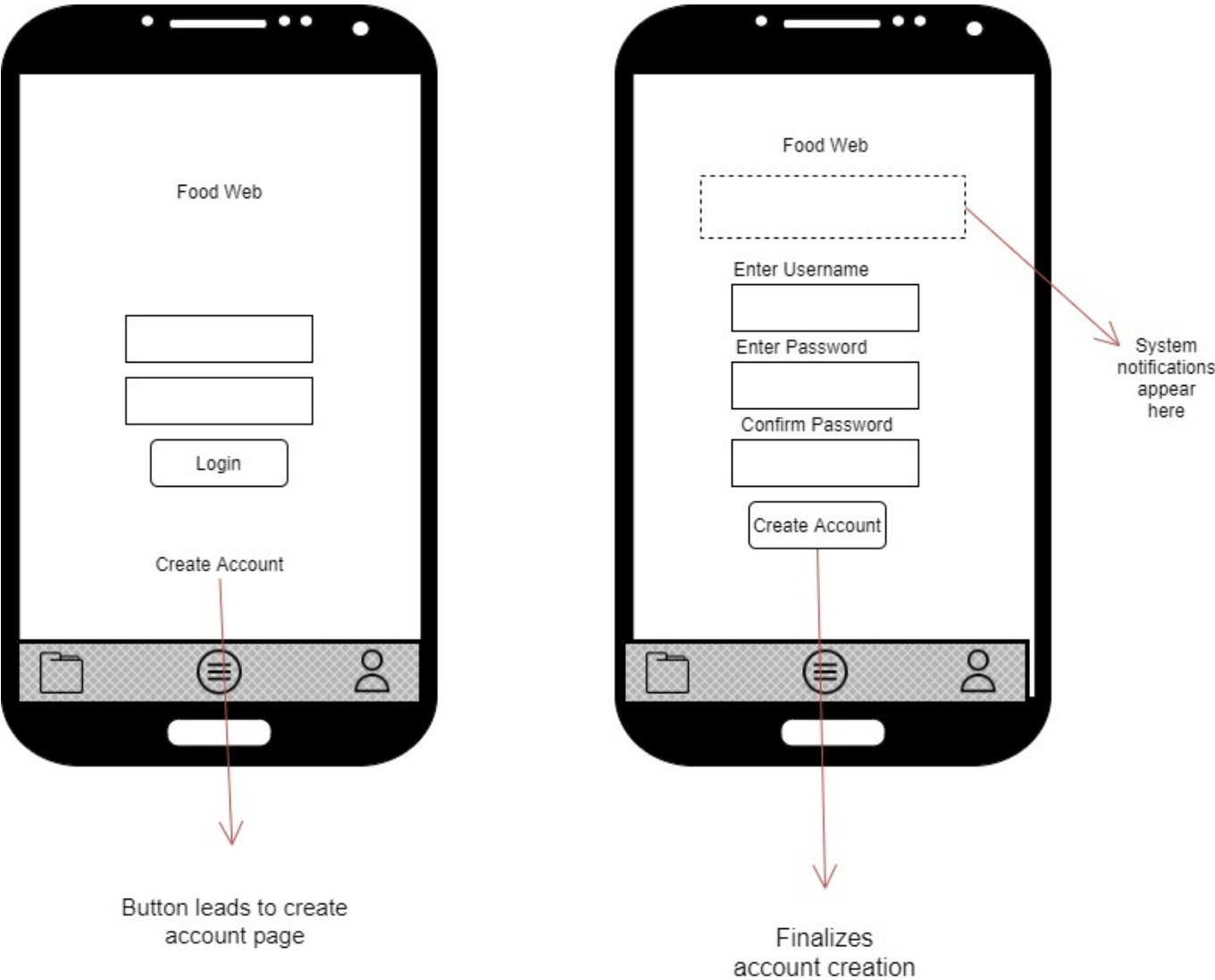
### Key Scenarios

1. The user selects 'create account'
  1. The user enters login information
  2. An account is created for the user

### Post-conditions

\* A new user account is created and stored in the system

Use Case Sketch



4. Search for Recipes

FoodWeb	Nazhem Jones
Use-case Specification:	Search for Recipes
Date:	09/25/2021

Brief Description:

the user searches for recipes on foodWeb to either use or save

Actor Brief Descriptions

\* Member: A foodWeb user with a foodWeb account

### Preconditions

\* The user is logged into their account and the foodWeb app is open.

### Basic Flow of Events

1. The use case begins when the foodweb user selects the search for recipes option.
2. The system displays the search bar in response to the user selection of "search".
3. User inputs information, in the "search bar", about the types of recipes he/she is looking for.
4. A list of recipes, fitting the users specifications, is displayed by the system.
5. The user can select recipes and inspect them.
6. The user can choose to save the recipe or continue searching.

### Alternative Flows

1. If in step 3 of the basic flow the user searches for a word or ingredient that is not found in the recipe library, the system displays "no recipes found".
  1. The user can then select "clear search"
  2. The use case resumes at basic flow step 2.

### Subflows

1. The user finds a recipe and gains interest in seeing more of that account's recipes.
  1. User selects "view account" on the recipe they like.
  2. System displays that account and the recipes the account has posted.
  3. The User is now free to view those recipes and save them.

### Key Scenarios

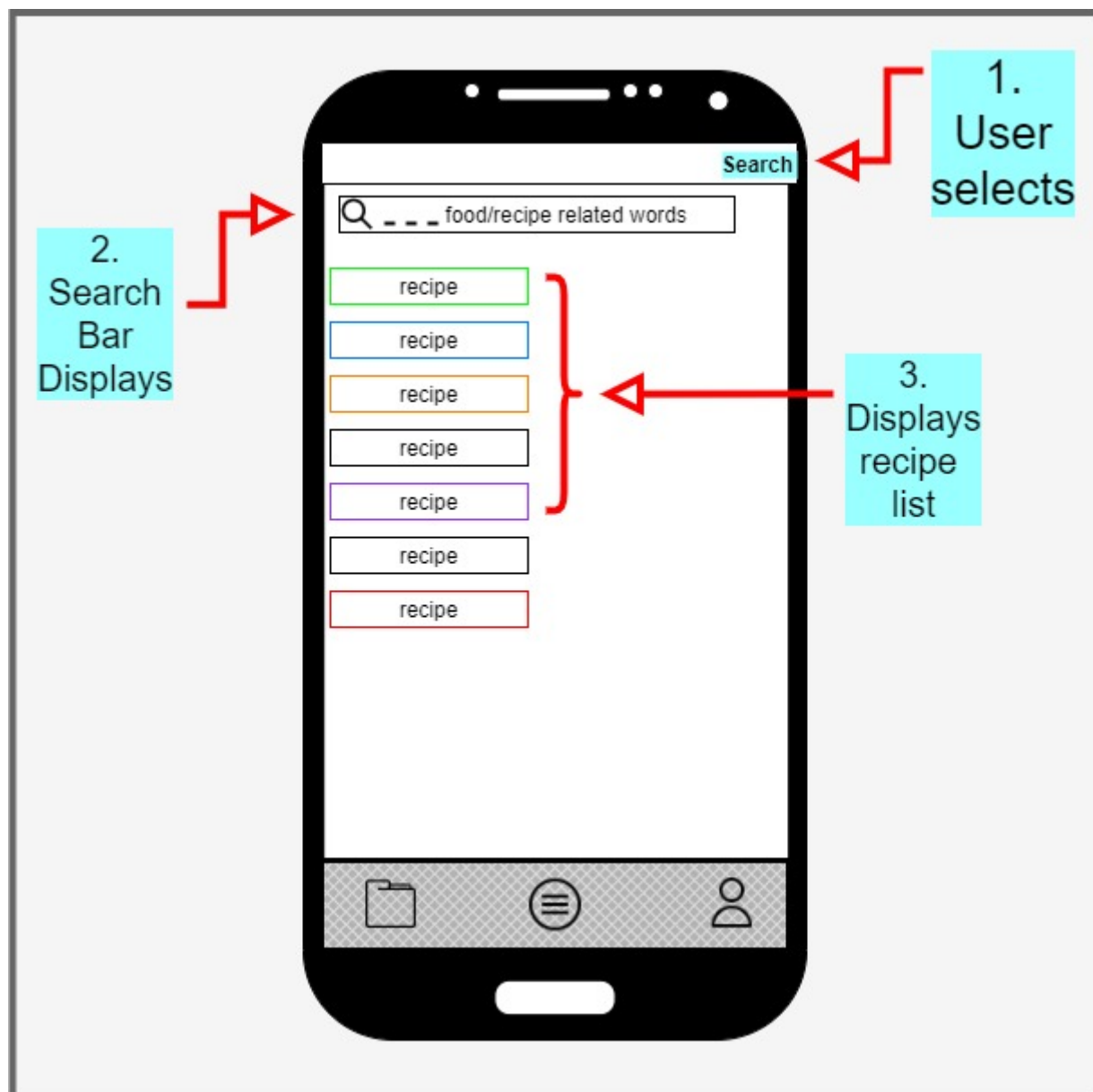
1. User searches for a recipe and finds it.
  1. User inputs search info. into the search bar.

2. System displays a list of recipes matching specifications set by the user.
3. User finds the recipe they want.
4. User uses the recipe or saves it.

#### Post-conditions

- \* The user finds a recipe to use now or save.

#### User Case Sketch



#### 5. Manage Ingredients

**FoodWeb**

**Dylan Anderson**

Use-case Specification: Manage Ingredients

**FoodWeb****Dylan Anderson**

Date: 09/23/2021

**Brief Description:**

the ability to add, remove, and change ingredients for a recipe

**Actor Brief Descriptions**

\* Member: A foodWeb user with a foodWeb account

**Preconditions**

\* The user is logged into their account

**Basic Flow of Events**

1. The use case begins when a user, wants to change an ingredient
2. System gets notice to add, remove or change ingredient
3. System asks user to confirm change
4. System completes the change
5. The use case ends.

**Alternative Flows**

1. If in step 3, user does not confirm change or system does not receive a confirmation, then
  1. System does not complete the change
  2. The use case resumes at step 1

**Key Scenarios**

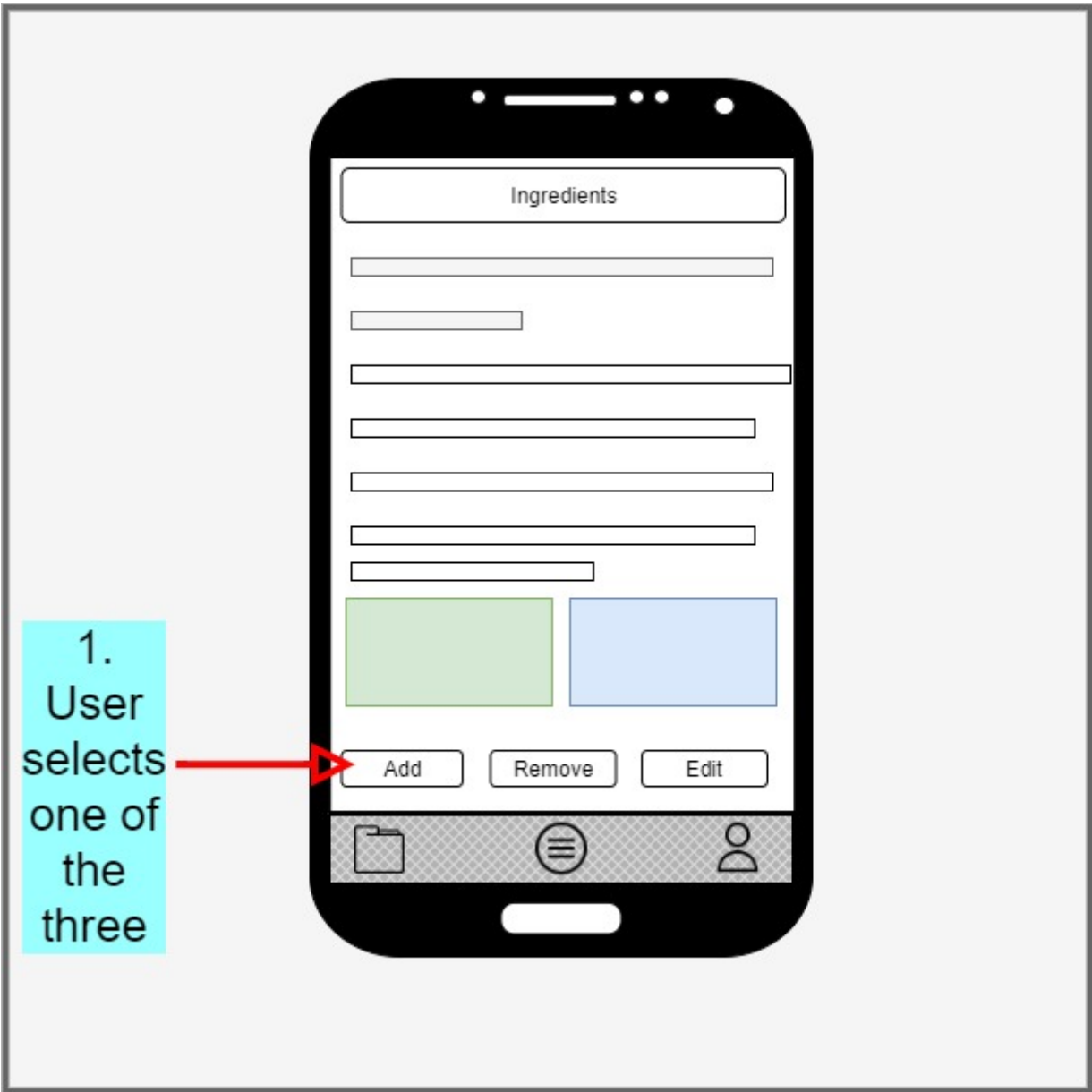
1. The ingredient is added, removed, or changed
  1. The user is prompted to provide the ingredient that is added, removed, or changed to

- 2. The user provides the ingredient
- 3. The user completes the change

Post-conditions

\* The ingredient is added, removed, or changed

Use Case Sketch



6. Save Recipes

FoodWeb	Ethan Green
Use-case Specification:	Save Recipes
Date:	09/23/2021

**Brief Description:**

the ability to store a recipe

**Actor Brief Descriptions**

\* Member: A foodWeb user with a foodWeb account

**Preconditions**

\* The user is logged into their account

**Basic Flow of Events**

1. The user selects the save option for a recipe
2. The system stores the recipe in a database associated with the user account
3. The recipe appears and available in the user's library
4. The use case ends

**Key Scenarios**

1. The user successfully saves a recipe

**Post-conditions**

\* The recipe is stored on the user's account

