

# **Deliverable 4**

## **Group 2**

Members:

Gilbert Martinez

Jon Brooks

## a). Requirements

### Frontend

- Feed displaying recipes
  - Shows all of the recipes listed in the sqlite3 database with a button on each recipe card linking to the recipe individually.
- Sort and search function
  - Sorts through the feed of recipes either by recently added or highest rated and allows users to filter by key words in the searchbar.
- Individual recipe pages
  - Displays all information of an individual recipe.
- User login
  - Where the user can login, create a account, or continue using the application as a guest.
- User dashboard
  - Displays all the users recipes, comments, and ratings. It all so allows the user to delete their recipes, comments, and ratings individually.
- Post recipe
  - Where the user can post their own recipe.
- Rating system
  - User can rate other users' recipes on a scale of 1 to 5 stars
- Comment system
  - User can comment on recipes
- Moderation tool
  - A page where the admin can see all the recipes and comments posted and can delete them individually if they violate the terms of service.

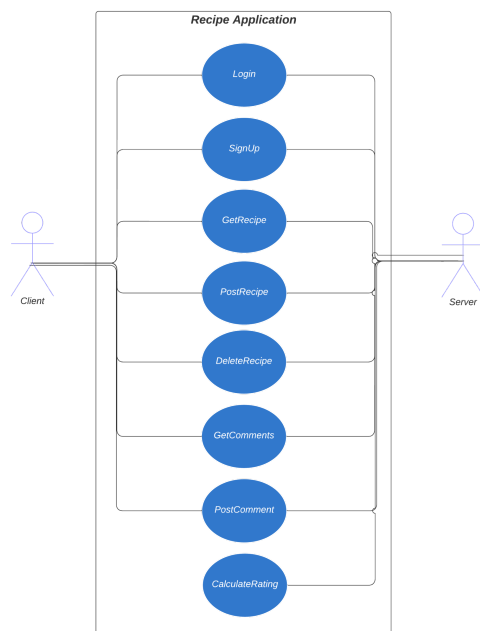
### Backend

- **Uploads**
  - Accept uploaded content
  - Serve uploaded content
- **Sessions**
  - The backend will be responsible for handling user sessions
- **Users**
  - Create
    - Being a user is required for posting recipes, comments and ratings
    - Any connection can create a user in order to achieve higher functionality
  - Delete
    - Users may need to be removed
- **Recipes**
  - Get all available recipes
    - Integrates with the frontend requirement of a dashboard
  - Get recipe pertaining to a specific id
    - Integrates with the frontend recipe page

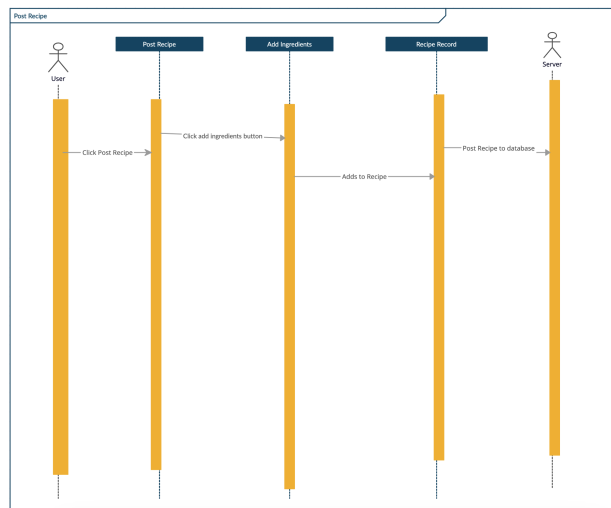
- Create
  - The foundational function of this app
  - Having a logged in session is required to create a recipe
  - Integrates with the users and uploads requirement
- Delete
  - Recipes may need to be deleted
- **Ratings**
  - View average rating for a recipe
    - We don't want to just return all the individual ratings, so instead we have the database return the average rating of each recipe for us
  - Rate
    - A user can rate a recipe
- **Comments**
  - View all comments for a specific recipe post
    - On the recipe page on the front end. Comments on the recipe will be displayed. So we offer an endpoint to retrieve all comments for a specific recipe
  - View all comments for a specific user
    - On the front end the user account page will have the users own comments displayed. We offer an interface to retrieve that
  - Post comment
    - A user can post a comment on a recipe
  - Delete comment
    - A user can delete their own comments

## b). Diagrams

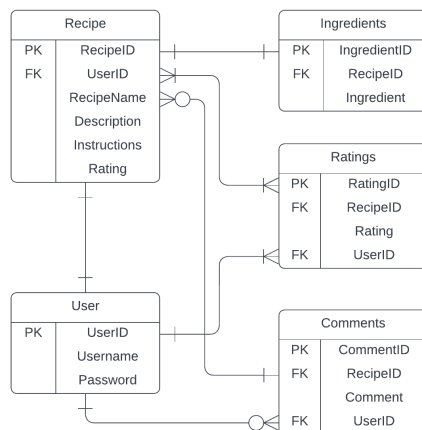
### Use Case Diagram



## Sequence Diagram



## Entry Relationship Diagram



### c). Test Cases

#### Post Recipe Form

- Case 1: Form is empty, Result: Gives user error to complete the form.
- Case 2: Some fields of the form is empty, Result: Gives user error to complete the form.
- Case 3: All field of form are completed, Result: Posts recipe to data and notifies the user the recipe has been posted.

#### Post Comment

- Case 1: comment input is empty, Result: Gives user error to write something in the comment input before posting comment.
- Case 2: comment input is field with text, Result: Post comment to data base and notifies the user that their comment has been posted.

## Create Account Form

- Case 1: form is empty, Result: Gives user a error letting them know they need to complete the form to create an account.
- Case 2: form is partly empty, Result: Gives user a error letting them know they need to complete the form to create an account.
- Case 3: Password and Verify Password input forms dont match, Result: Gives the user a error that the passwords do not match.

## d). User Manual

When using this application user have an option to browse as a guest or login to use the rating, commenting, and posting recipe features. Once a user has login they are then brought to the main page when you can sort and search through all the recipes. You can then click on the view recipe to go to a specific recipe page. There you can view the recipe and then rate it after you have used it. You also give the poster feedback on their recipe in the comment section of the page. To post a recipe users can click the post recipe button in the navigation bar and then click post recipe in form popup after all the fields in the form are completed. The delete recipes and edit user data the user can go click on the My Account button in the navigation bar. This bring you to your personal page where you can delete your recipes you have posted as well as any comments or rating you posted on other user's recipes. To delete a recipe you can toggle the trash can button then click the X button over the recipe you would like to delete.

## e). Instructions

### Frontend Application

#### Dependencies

- Node.js

#### Instructions

- Go to recipe-application/frontend
- npm install
- npm start
- In browser go to localhost:3000 to view application

### Backend Application

#### Dependencies

- Python3

#### Instructions to run dev server

- change working directory to "recipe-application/backend"
- > python3 -m venv venv
- > source ./venv/bin/activate (> .\venv\Scripts\Activate.ps1 on windows)
- (venv) > python3 -m pip install flask
- (venv) > flask --app api initdb
- (venv) > flask --app api run

## f). Code Inpection

Added a moderation page and a comment system based on the feedback given.

**g). Accomplishments**

Name	Contributions
Gilbert Martinez	<ul style="list-style-type: none"><li>- Login page</li><li>- Recipe Feed</li><li>- Recipe Sorting function</li><li>- Individual Recipe Page</li><li>- Client side Post Rating function</li><li>- Comment Section</li><li>- Account Page</li><li>- Client side Delete Recipe function</li><li>- Moderation Page</li></ul>
Jon Brooks	<ul style="list-style-type: none"><li>- User sessions</li><li>- Database structure and initialization</li><li>- User create/delete/query</li><li>- Recipe create/delete/query</li><li>- Rating create/delete</li><li>- Comment create/delete</li><li>- Rating calculation</li><li>- Image file storage</li></ul>