YumBoard

Table of Contents

Who

Members:

- Jackson Vondemkamp
- Aryaman Singh
- Jason Gao
- Dresden Friar
- José Camacho Guadamuz

Project Description

YumBoard is a social cooking app made for college students and young adults who want to cook more often but need tools that make it easier and quicker. The app helps people share and discover recipes with their friends. The main goal is to make cooking feel more social and less overwhelming. Users can post recipes with pictures, ingredients, and instructions, and they can also check out what their friends are cooking to get inspiration. It works like a personalized recipe feed where everyone contributes.

One of the key features is the built-in grocery list. When someone likes a recipe, they can add all the ingredients to their list in one click. YumBoard connects to the Kroger API to pull real-time prices, which means it can estimate the total cost of groceries based on the items selected. This makes budgeting and planning much easier. The list also updates automatically whenever users add or remove recipes.

The backend runs on Node.js and Express, with PostgreSQL handling the data. The frontend uses Handlebars for templating, Bootstrap for layout, and custom CSS for extra styling. YumBoard makes it simple to find new recipes, track ingredients, and stay connected through food.

Project Tracker - GitHub Project Board

Link to project tracker : https://github.com/orgs/YumBoard-Dev/projects/2

Video

Link to Video : https://youtu.be/BL0UH2rHkKI

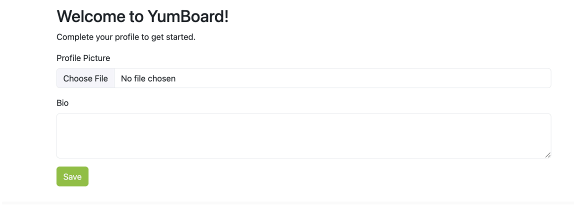
VCS

Link to Git Repository : https://github.com/YumBoard-Dev/YumBoard

Contributions

Jackson Vondemkamp

I developed the backend functionality of the login and register (Including RegEx requirements). Then, I developed the full functionality of the profile page (minus the recipe view), and the onboarding process where you pick a profile picture and a bio. I also fixed small bugs and issues, for example changing Universal Time to local time in comments.



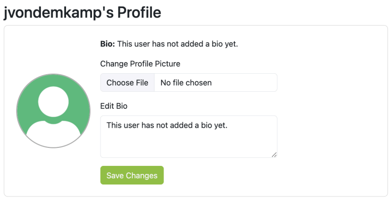
Welcome to YumBoard!

Complete your profile to get started.

Profile Picture

Choose File No file chosen

Bio



jvondemkamp's Profile

Bio: This user has not added a bio yet.

Change Profile Picture

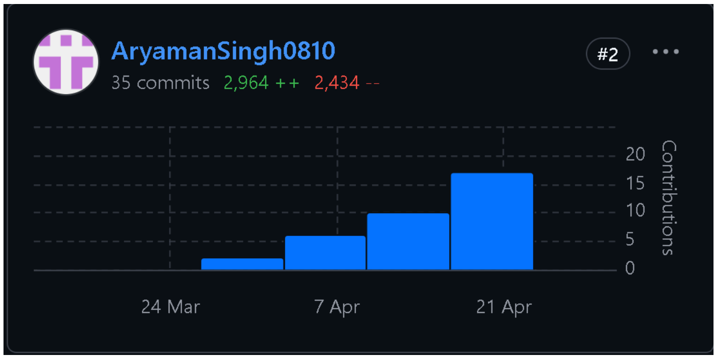
Choose File No file chosen

Edit Bio

This user has not added a bio yet.

Aryaman Singh

I made the post recipes pages and the, my recipes part of the profile page. I also checked the code for bugs before completion. I also made the sorting function that was used on my recipes page.I also wrote the UATs as well.

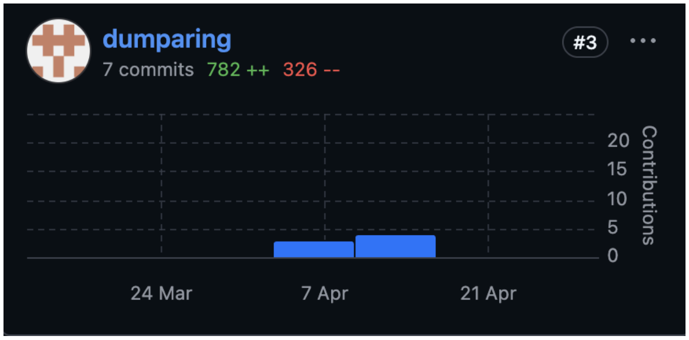


AryamanSingh0810

35 commits 2,964 ++ 2,434 --

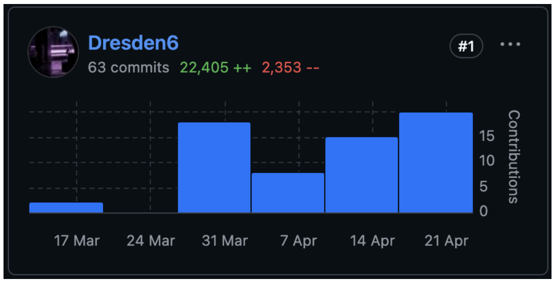
Jason Gao

I created like liking and commenting feature. I also made the feature that calculates a recipe's total price from the Kroger API. Additionally, I fixed some errors in the database and just some styling. I made the handlebars for the recipe page as well.



Dresden Friar

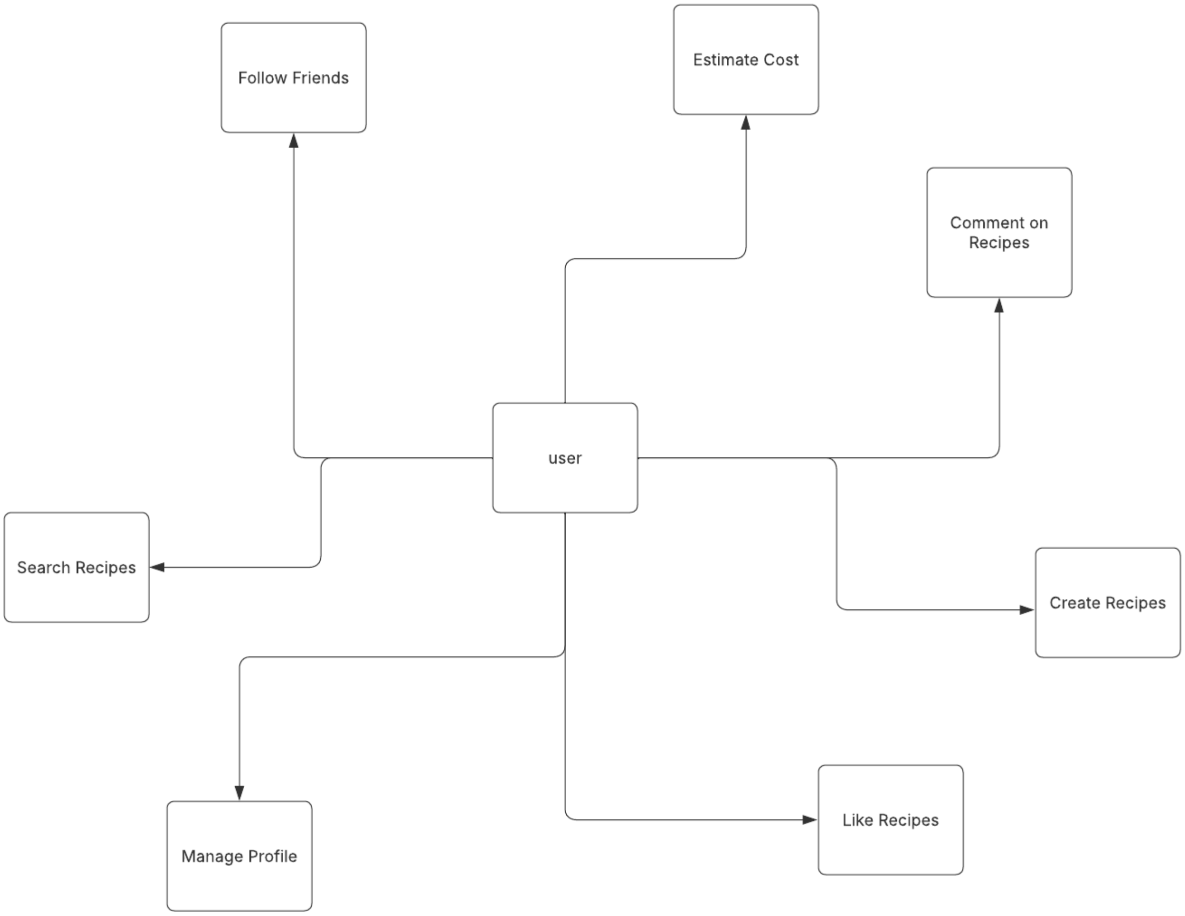
I made the navbar and home page, including the filter/search feature on the home page. I also made the login and registration frontend, and worked on some of the backend (session handling). I implemented dark mode, and set up bootstrap theming with Sass. I wrote all of the Mocha/Chai tests, and hosted the page on Render. I also fixed various little bugs and made small additions to many pages, like the delete button for recipes.



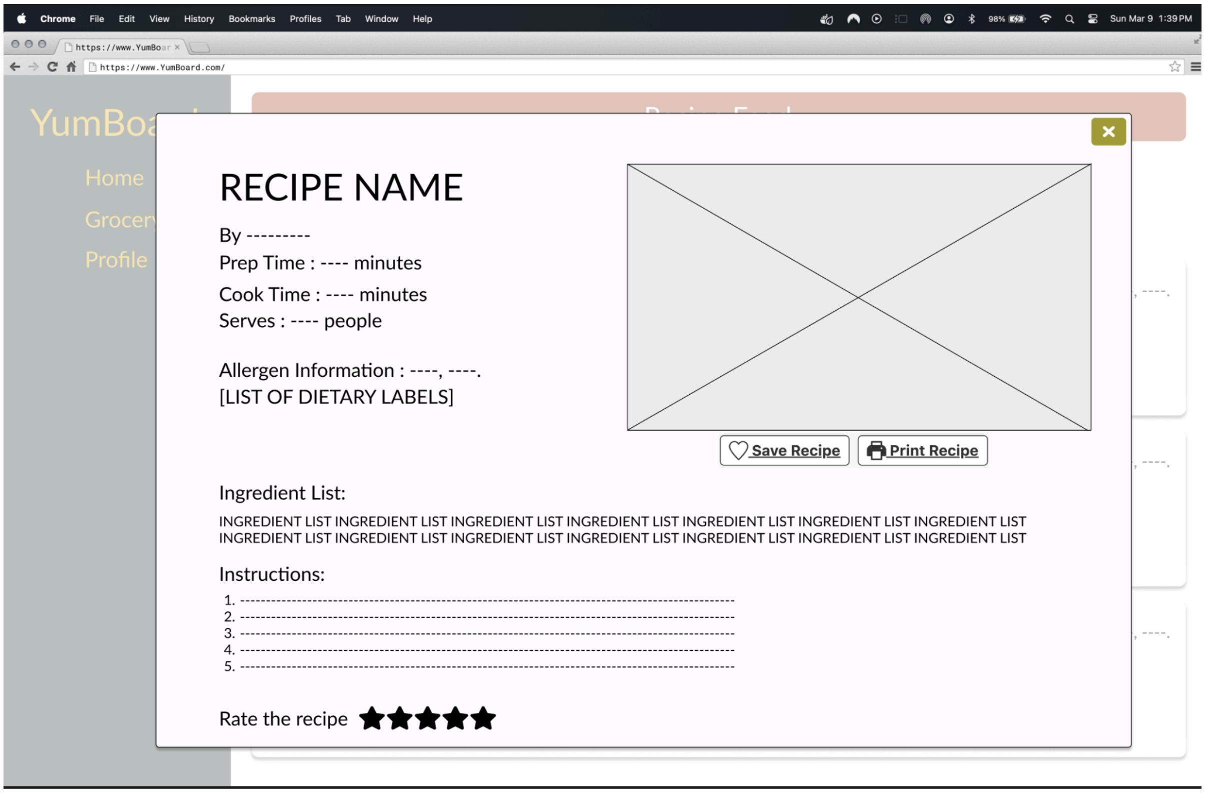
José Camacho Guadamuz

I made the .sql file that creates the tables for the database which is where all the data gets stored. I also worked on the user profile page that shows the users information and allows them to change it and on the grocery. As well as making some changes to enhance the UI.

Use Case Diagram



Wireframes


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Sun Mar 9 1:39 PM
https://www.YumBoard.com/
YumBoard
Home
Grocery
Profile
Save Recipe
Print Recipe

RECIPE NAME

By ---------

Prep Time : ---- minutes

Cook Time : ---- minutes

Serves : ---- people

Allergen Information : ----, ----.

[LIST OF DIETARY LABELS]

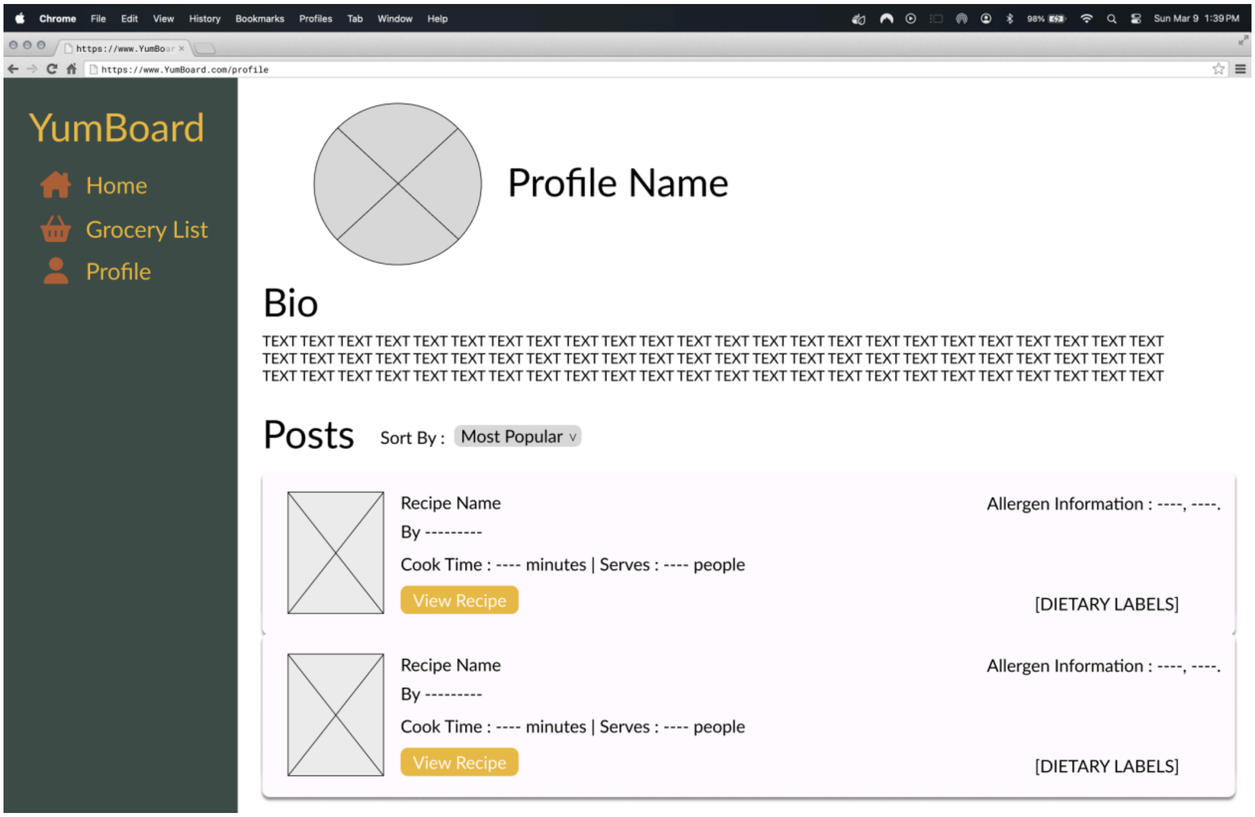
Ingredient List:

INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST INGREDIENT LIST

Instructions:

1. --
2. --
3. --
4. --
5. --

Rate the recipe ★★★★★


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Sun Mar 9 1:39 PM
https://www.YumBoard.com/profile
YumBoard
Home
Grocery List
Profile
View Recipe
View Recipe

Profile Name

Bio

TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT TEXT

Posts Sort By : Most Popular v

Recipe Name

By ---------

Cook Time : ---- minutes | Serves : ---- people

Allergen Information : ----, ----.

[DIETARY LABELS]

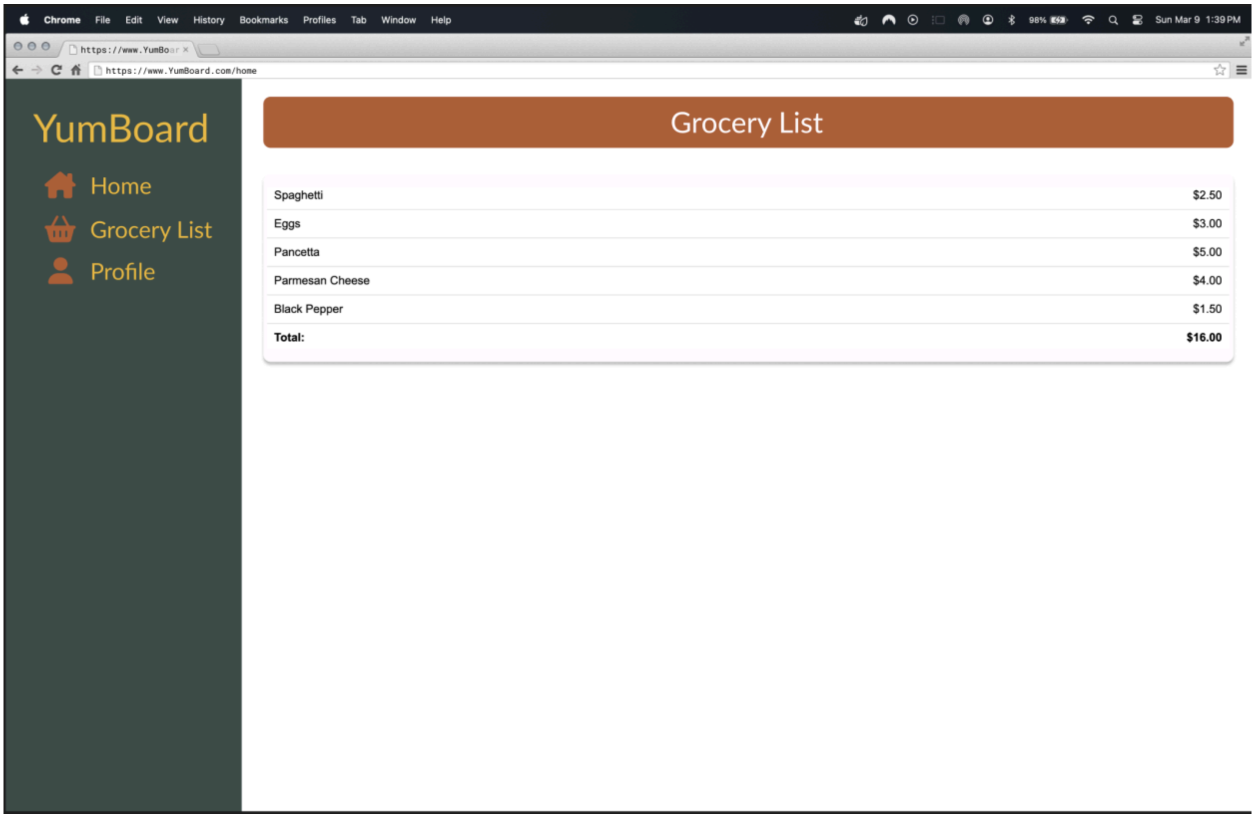
Recipe Name

By ---------

Cook Time : ---- minutes | Serves : ---- people

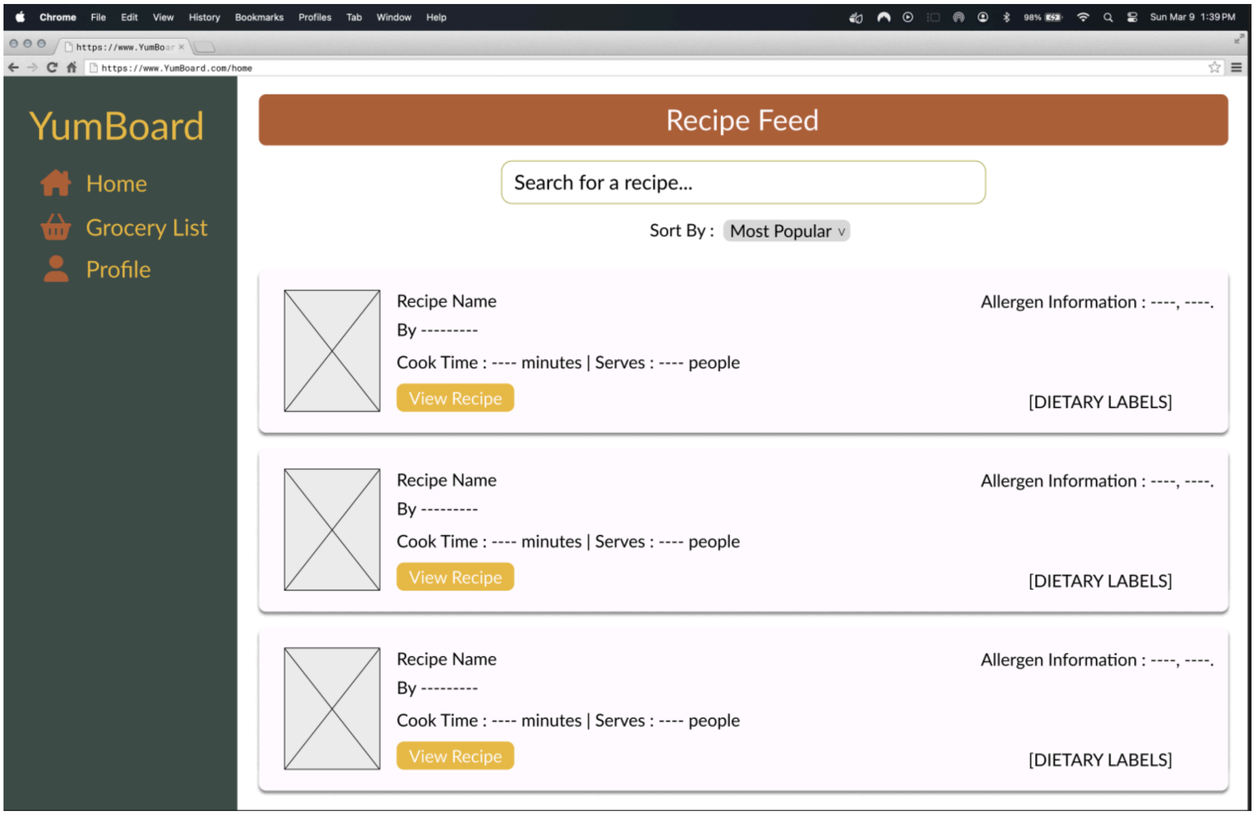
Allergen Information : ----, ----.

[DIETARY LABELS]


Chrome File Edit View History Bookmarks Profiles Tab Window Help
98%
Sun Mar 9 1:39 PM
https://www.YumBoar
https://www.YumBoard.com/home
YumBoard
Home
Grocery List
Profile

Grocery List

Spaghetti	$2.50
Eggs	$3.00
Pancetta	$5.00
Parmesan Cheese	$4.00
Black Pepper	$1.50
Total:	**$16.00**



Recipe Feed

Search for a recipe...

Sort By : Most Popular v

Recipe Name
By ---------
Cook Time : ---- minutes | Serves : ---- people
Allergen Information : ----, ----.
[DIETARY LABELS]

Recipe Name
By ---------
Cook Time : ---- minutes | Serves : ---- people
Allergen Information : ----, ----.
[DIETARY LABELS]

Recipe Name
By ---------
Cook Time : ---- minutes | Serves : ---- people
Allergen Information : ----, ----.
[DIETARY LABELS]

Test Results

Register

Testing Data: None. A blank database is required.
Environment: Localhost
User Acceptance Testers: Someone who did not work on the app, so it is a fair test of how intuitive the interface is.
Steps:

1. Navigate to the register page from the home page (using buttons in the top right)
2. Fill out user information with a valid username and password
3. Click register button

Verify:

- Entry added to the users table.
- User automatically logged in and redirected to home page.
- Session visible in browser dev tools.

Test Results:

- ☑ Correct navigation to register page.
- ☑ Error messages shown when required fields missing or password invalid.
- ☑ Successful registration with valid data.
- ☑ Auto-redirected to home and session set correctly.

Observations:

- What are the users doing?
  - Users attempted to fill in fields immediately and submit. Some initially skipped fields and immediately tried to submit.
- What is the user's reasoning?
  - They expected real-time validation feedback to catch missing fields without needing to submit.
- Is their behavior consistent with the use case?
  - Yes, filling and submitting is expected.
- Deviations from expected actions?
  - Users tried submitting forms with missing fields, possibly assuming there would be in-line validation instead of just after submission.
- Changes made?
  - Added real-time field validation to display errors before form submission (e.g., if password doesn't match regex or if a field is missing).

Posting a Recipe

Testing Data: A registered user in the database, and some recipe tags in the tags table.
Environment: Localhost
User Acceptance Testers: Someone who did not work on the app.

Steps:

1. Login with registered user.
2. Navigate to "Post Recipe."
3. Fill out form fields and upload images.
4. Submit the recipe.

Verify:

- Recipe is added to database.
- User is redirected to the recipe page with correct display.

Test Results:

☑ Logged in correctly.
☑ Could not submit without required fields.
☑ Submitted recipe with and without image (default image used when needed).
☑ Correct redirection and display.

Observations:

- What are the users doing?
  - Users filled in fields in order and attempted uploads.
- What is the user's reasoning?
  - Expected image upload to be optional because of form labeling; assumed all fields marked required needed to be filled.
- Is their behavior consistent with the use case?
  - Yes.
- Deviations from expected actions?
  - Some users expected the image upload button to specify that it's optional.
- Changes made?
  - Clarified labeling near the image upload input ("optional").

Liking and Commenting a Recipe

Testing Data: A valid recipe and a valid user exist in the database.
Environment: Cloud Deployment
User Acceptance Testers: Someone who did not work on the app.

Steps:

1. Login.

2. Navigate to a recipe page.
3. Like a recipe.
4. Comment on a recipe.

Verify:

- Entries added to likes and comments tables.
- Like and comment appear visibly after posting.

Test Results:

- ☑ Like button updated count instantly.
- ☑ Comment appeared immediately.
- ☑ Alert correctly shown when comment box was empty.
- ☑ Like/comment persisted when checked with another account.

Observations:

- What are the users doing?
  - Users quickly found and clicked the heart icon and comment box.
- What is the user's reasoning?
  - Expected immediate feedback visually after like/comment action.
- Is their behavior consistent with the use case?
  - Yes.
- Deviations from expected actions?
  - A few users thought double-clicking the recipe would like/unlike (like Instagram behavior), but ours is single-click on the heart only.
- Changes made?
  - Decided not to add double-click functionality for now, but made the heart briefly animate on click to visually reinforce that the action was successful.

Viewing & Sorting Liked Recipes

Testing Data: User with multiple liked posts, each with varying metadata.
Environment: Localhost
User Acceptance Testers: Someone who did not work on the app.

Steps:

1. Login.
2. Navigate to "Liked" page.
3. Use sort dropdown to reorder liked recipes.

Verify:

- Recipes load correctly.
- Sorting updates list dynamically.

- Error messages on server failure.
- Empty state message if no liked recipes.

Test Results:

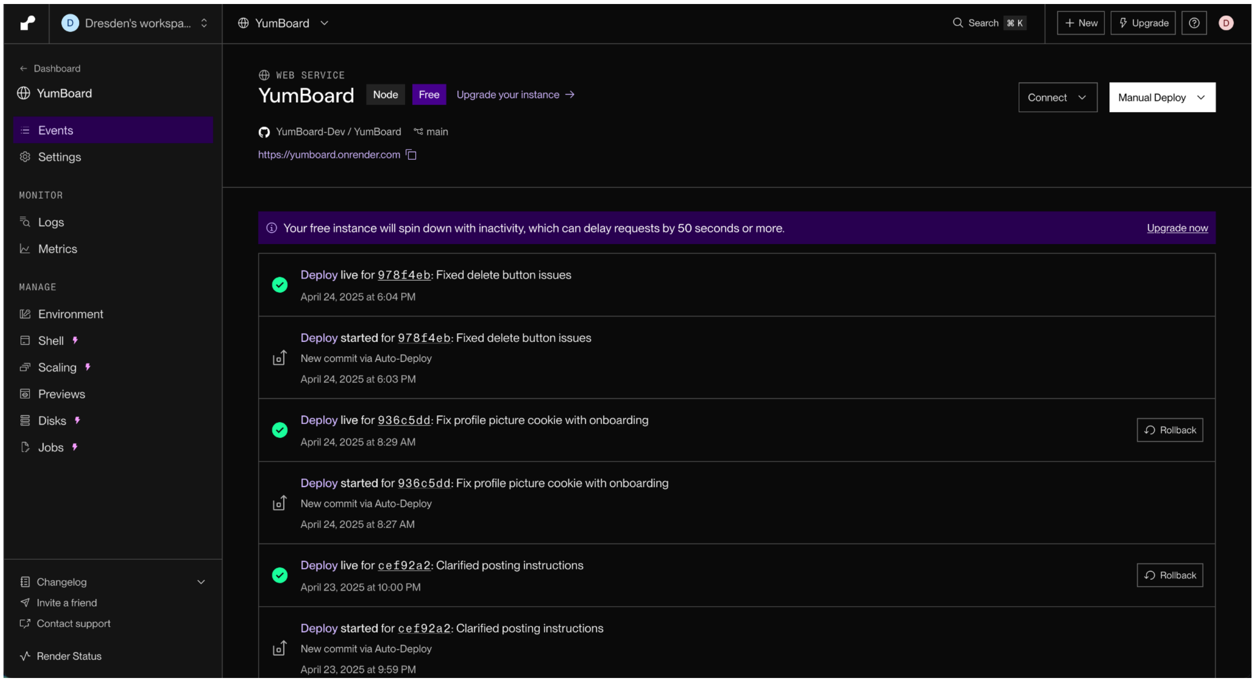
☑ Default load was by newest first.
☑ Correct dynamic reordering for each sort selection.
☑ Error message displayed correctly when server down.
☑ Empty state handled gracefully.

Observations:

- What are the users doing?
  - Users immediately tested multiple sort options without needing any guidance.
- What is the user's reasoning?
  - They expected changes to happen immediately after sort option selected.
- Is their behavior consistent with the use case?
  - Yes.
- Deviations from expected actions?
  - None really
- Changes made?
  - Nothing

Deployment

Render Link: https://yumboard.onrender.com/



Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more. Upgrade now

Deploy live for 978f4eb: Fixed delete button issues
April 24, 2025 at 6:04 PM

Deploy started for 978f4eb: Fixed delete button issues
New commit via Auto-Deploy
April 24, 2025 at 6:03 PM

Deploy live for 936c5dd: Fix profile picture cookie with onboarding
April 24, 2025 at 8:29 AM

Deploy started for 936c5dd: Fix profile picture cookie with onboarding
New commit via Auto-Deploy
April 24, 2025 at 8:27 AM

Deploy live for cef92a2: Clarified posting instructions
April 23, 2025 at 10:00 PM

Deploy started for cef92a2: Clarified posting instructions
New commit via Auto-Deploy
April 23, 2025 at 9:59 PM