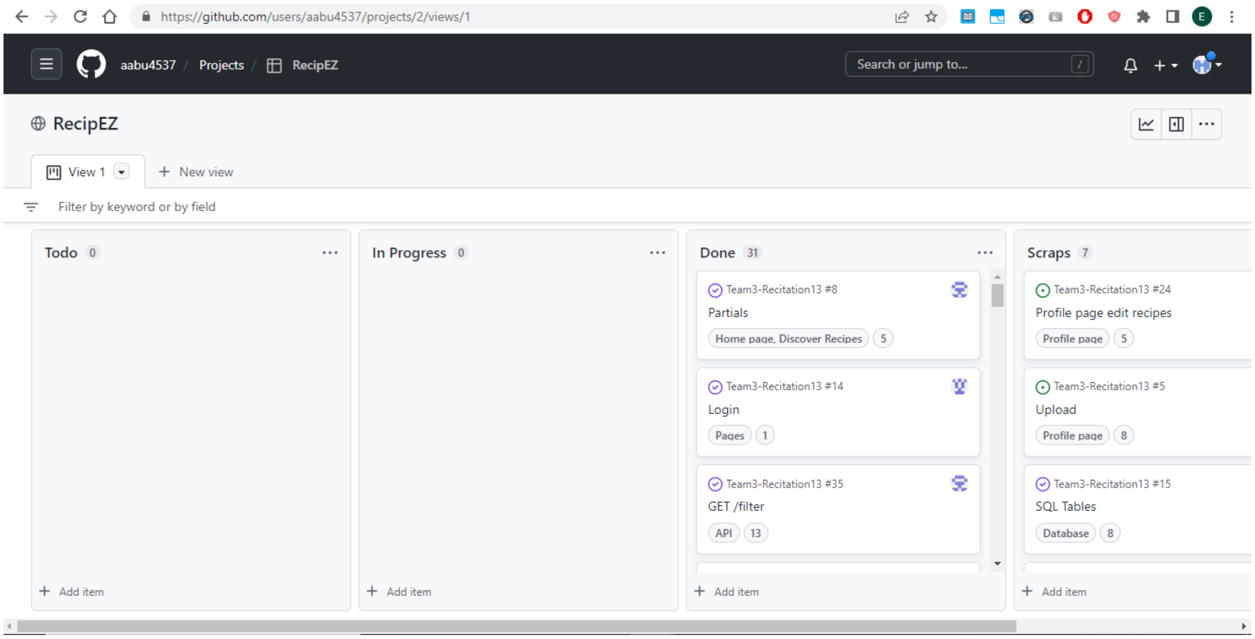
RepicEZ

Kayla Zhong
Batool Hani
Alam Bushara
Drew Boynton
Eric Graham

Recipeasy is a recipe sharing website for elderly people. This is precisely why the name is a bad pun. Recipeasy is designed to replace physical cookbooks entirely, meaning you can find and save new recipes online rather than having to buy and store physical cookbooks. The main features are discovering recipes through filtering and sorting and the ability to curate a personal cookbook (list of favorite recipes). The personal cookbook of course requires a sign-in function, but the rest of the website is accessible without a sign-in. The value of the website mainly comes from ease-of-use, so we used big font sizes, and did not clutter the pages. Users are able to upload their own recipes in order to be able to keep everything in one place.

[Recipeasy Project Board]


https://github.com/users/aabu4537/projects/2/views/1
aabu4537 / Projects / RecipEZ
Search or jump to...
RecipEZ
View 1
New view
Filter by keyword or by field
Add item
Add item
Add item
Add item

Todo 0

In Progress 0

Done 31

Team3-Recitation13 #8
Partials
Home page, Discover Recipes 5

Team3-Recitation13 #14
Login
Pages 1

Team3-Recitation13 #35
GET /filter
API 13

Scraps 7

Team3-Recitation13 #24
Profile page edit recipes
Profile page 5

Team3-Recitation13 #5
Upload
Profile page 8

Team3-Recitation13 #15
SQL Tables
Database 8

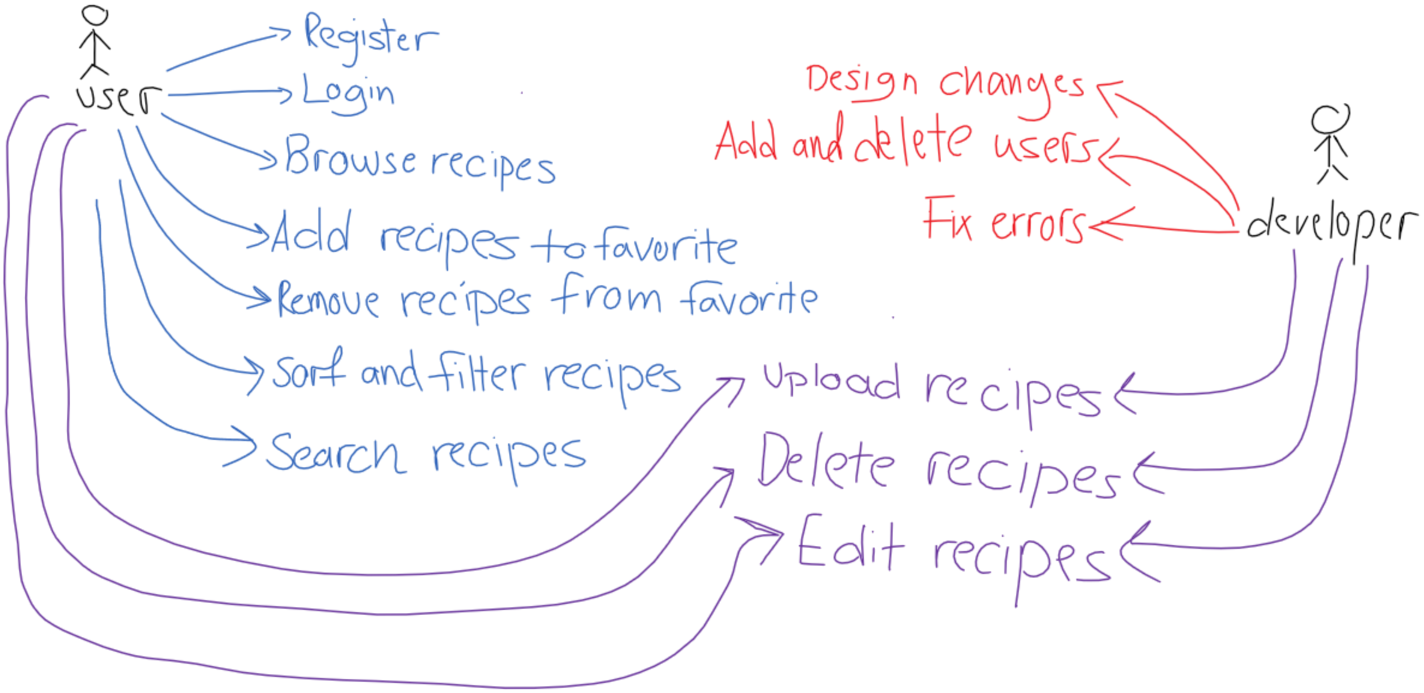
Video:
https://drive.google.com/file/d/1iftWI_fFXMfu6o0q8vTCYqEbyL5C2vep/view?usp=share_link

Version Control: [RecipEZ GitHub]

Contributions:

- Kayla: I took care of the release notes, the Discover page, and any related functionalities. This includes designing the Discover page and recipe cards, helping set up the database to contain information for recipe details, and debugging API calls for the display as well as the filter, search, and sort features.
- Batool: I worked on the profile page and everything related to it. I implemented multiple functions for the API calls in and I worked on ejs for everything else. The profile page included the following features: Displaying name and email to each specific user, adding recipes, seeing your favorite recipes and viewing the details, seeing your added recipe and viewing the details, removing recipes from favorite, deleting uploaded recipes, and editing any entries in your uploaded recipes. I fixed some samll issues in the haeder and footer code. I also helped with testing and debugging errors as well as merging and merge conflicts. I helped with identifying and distributing weekly tasks for the team.
- Alam: I worked with Eric to create the login, register, and logout functions. As well as creating a ‘View recipe’ page, where it retrieves the recipe_id of each recipe and displays its instructions, ingredients, and rating. I also handled the API routes of this specific page. I also handled most of the inserts for the pre-set recipes in the discover page.
- Drew: I set up the file directory at the start and made sure the structure was fine from top to bottom. Created partials and made sure all the html files had those as a base skeleton. Wrote API calls sort and filter that take all the recipes in the database, and sort/filter them based on the user’s needs, then display those select recipes in the order they are supposed to be displayed.
- Eric: I worked with Kayla to design our database. I also wrote the API routes for adding a favorite recipe, uploading a recipe, for retrieving the signed-in user’s favorite recipes for displaying on the profile page, and retrieving all recipes from the database for the home page. I worked with Alam on the login, register, and logout functions.

Use Case Diagram:



Test Cases and Results

Feature 1: Register, Login, and Logout

1. Register
    a. Description: Users can register into RecipEZ
    b. Expected Input: User clicks the “Register” link at the bottom of the Login page, fills out the form, and then clicks the “Register” button
    c. Output:
        - If a username and password is provided, the user’s username, email (optional), and password is stored in the ‘users’ table in the ‘recipes’ database with a unique user_id.
        - User is redirected to the Login page.
        - If successful, users should be able to login with their registered information and unable to otherwise.
    d. Observations/Results:
        - User Test 1: User took longer than expected to find the register link because they were focused on the login form. They filled out the register form, clicked “Register”, and then successfully logged in with the same credentials. They were confused on why they had to include a purpose because it wasn’t something they usually registered with.

        - User Test 2: User was able to register with not problem, and even wrote their purpose of the account. However, they didn't understand the point of username and email. They preferred one or the other.
        - User Test 3: It took the user a few seconds to see the register button and go to the register page. The register is straightforward and easy and doesn't have many requirements which the user liked. The purpose part was vague and didnt know wether to write a word, a sentence, or an article.

2. Login
    a. Description: Users can log in to RecipEZ by filling out the login form with the correct credentials and pressing "Login"
    b. Expected Input: User's username, email (optional), and password, clicks "Login" button
    c. Output:
        - If the correct username, email, and password is entered, the user should be logged in and navigated to their unique profile page.
        - If the correct username and password is entered, and no email is provided, the user should be logged in and navigated to their unique profile page.
        - If the email entered is incorrect, the login page is reloaded.
        - If the username/password entered are empty or incorrect, the login page is reloaded.
    d. Observations/Results:
        - User Test 1: User was confused initially because they didn't have any login credentials. After navigating to the registration page and registering, they successfully filled out the form and logged in with no issues. It did take the user longer than expected to find the register link.
        - User Test 2: User read the instructions and clicked on the registration link and made an account. It then directed him to the login page, where he was able to successfully login to his account.
        - User Test 3: The user like that the login page appeared right after registration, which made it go smoothly instead of looking for the login page again. The Login worked successfully with no errors or difficulties. The user would have preferred to login with either email or username to make things go faster, otherwise, everything else was fine.

3. Logout
    a. Description: Users can log out of RecipEZ by clicking "Logout"
    b. Expected Input: User clicks "Logout" at the top of any page
    c. Output:
        - User is navigated to the Login page with the message "Logged out Successfully" displayed at the top.

d. Observations/Results:
   - User Test 1: User looked first at the top right of the page for a logout feature. After some deliberation, they found the “Logout” button, clicked it, and successfully logged out. They explained that they’re used to having the logout feature at the top right instead of the top center.
   - User Test 2: User did not log in to his account to begin with. So the logout button took him to the login page. He explained that the location of the button was perfect.
   - User Test 3: User logged out successfully without issues and the message confirming it made things easier to understand.

Feature 2: Discover Recipes

1. Filter Recipes
   a. Description: Users can filter recipes using the provided form in the Discover page
   b. Expected Input: User clicks on the filters they want to apply (checkboxes or dropdown menu) and then clicks the “Apply Filter” button
   c. Output:
      - After “Apply Filter” is clicked, the Discover page is reloaded showing all recipes whose details match the filters.
      - If unsuccessful, the Discover page is reloaded with an error message and blank recipes.

   d. Observations/Results:
      - User test 1: User looked for a recipe that was 5 stars and vegetarian. Nothing shows up as a result and she believes that our website is broken. She suggests having an error message saying that there are none that fit in the filter.
      - User test 2: Tested out multiple different filters. Also noted that a no results message would be useful. Said with multiple filters enabled there are too many instances where there is no results.
2. Search Recipes
   a. Description: Users can search for a recipe by name or cuisine type in the Discover page.
   b. Expected Input: User types a recipe name or cuisine type in the appropriate text box at the top of the Discover page and then clicks the corresponding “Search” button.
   c. Output:
      - If “Search” is clicked for recipe name, the Discover page is reloaded showing all recipes whose name matches the entry.
      - If “Search” is clicked for cuisine type, the Discover page is reloaded showing all recipes whose cuisine type matches the entry.
      - If unsuccessful, the Discover page is reloaded with an error message and blank recipes.

   d. Observations/Results:

        - User test 1: User inputs the name ‘Banana Pancakes’ into the search and it returns the Banana pancakes recipe card.
        - User test 2: User inputs rice and and mexican rice and beans show up. Noted that more recipes will make the website better.

3. Sort Recipes
    a. Description: Users can sort all recipes on the Discover page according to their selected option
    b. Expected Input: User selects an option to sort in the dropdown menu found at the top of the Discover page and then clicks the “Sort” button
    c. Output:
        - After “Sort” is clicked, the Discover page is reloaded showing all recipes sorted according to the option chosen.
        - If unsuccessful, the Discover page is reloaded with an error message and blank recipes.
    d. Observations/Results:
        - User test 1: User selects the dropdown menu and chooses to filter by rating and the first recipe is Poutine. She clicks on ‘view recipe’ to confirm that it is 5 stars.
        - User test 2: User selects dropdown menu and test various sort options. Makes sure highest to lowest for rating is opposite of lowest to highest. Makes sure the rating system actually makes sense and is being displayed properly. Very content with this feature.
4. View Recipes
    a. Description: Users can view a page showing all the recipe details for any recipe
    b. Expected Input: User clicks on the “View Recipe” button in the recipe card for the recipe they’re interested in
    c. Output:
        - After “View Recipe” in a recipe card is clicked, users are navigated to another page displaying the recipe’s image, name, ingredients, instructions, and rating.
        - If unsuccessful, the Discover page is reloaded with an error message and blank recipes.
    d. Observations/Results:
        - User test 1: The user clicks on view recipe for the ‘Mexican rice and beans’ recipe and is taken to a new page where it lists the instructions and ingredients of the recipe.
        - User test 2: User views recipe for a already existing recipe and is please by the layout of the ingredients and other information. Then makes own custom recipe and makes sure this page is working for that.

Feature 3: Favorites

1. Favorite Recipe
    a. Description: Users can add a recipe from the Discover page into their favorites
    b. Expected Input: User clicks on the “Favorite” button in the recipe card for the recipe they’re interested in

c. Output:
    - After “Favorite” is clicked, the Discover page is reloaded. The user can click on “My Profile” at the top of any page and see the recipe card in their profile under “My favorite recipes”. The user’s id and recipe’s id is stored in the ‘favorites’ table in the ‘recipes’ database.
    - If a recipe is already favorited, the Discover page is reloaded with no changes in the profile page and no new additions in the ‘favorites’ table.
    - If unsuccessful, the Discover page is reloaded.

d. Observations/Results:
    - User Test 1: User immediately liked the “Pancit” recipe upon seeing it. After viewing the recipe, they wanted to favorite it. They pressed “Favorite” and successfully favorited the recipe. The user was confused on how to view all of their favorites, but after some explanation, they understood and began favoriting more recipes.
    - User Test 2: User sorted by several different options, including rating from lowest to highest then the opposite as well. All options worked. Tested out most of sort options, noted that more recipes would make these options even more convenient.
    - User Test 3: When the user found a recipe, they clicked on “view recipe” to find out more, and then they said they were ready to add it to their favorites, expecting to be able to from the view page. They realized they couldn’t and then navigated back to the discover page and favorited it from there.

2. Unfavorite Recipe

    a. Description: Users can remove a recipe from their favorites

    b. Expected Input: User clicks on the “Remove favorite” button in the recipe card for the recipe they’re interested in

    c. Output:
        - After “Remove favorite” is clicked, the Profile page is reloaded with the recipe card no longer seen under “My favorite recipes” in the user’s profile. The row in the ‘favorites’ table containing the user’s id and the recipe’s id is deleted.
        - If unsuccessful, the Profile page is reloaded.

    d. Observations/Results:
        - User Test 1: User navigated to the Profile page, pressed “Remove favorite”, and saw the recipe removed after the Profile page reloaded. They had no trouble with this task after playing around with the favorite feature first.
        - User Test 2: User started on the discover page, favorited a few recipes. Went to profile page to make sure they showed up. Checked specific recipe pages to make sure they worked from the profile page. Tested unfavorite option, was happy with this feature.

        - User Test 3: Starting on the profile page, the user located a recipe they wanted to remove, and without hesitation clicked the remove button. This successfully prevented the recipe from showing up on their profile.

Feature 4: Personal Recipes

1. Add Recipe
    a. Description: Users can create and upload a recipe of their own into RecipEZ
    b. Expected Input: User clicks on the "Add a recipe" button in their profile, fills out the form with their recipe details, and then clicks either the "Submit" button or the "Close" button
    c. Output:
        - If "Submit" is chosen, the recipe's details and user's id is stored in the 'recipes' table in the 'recipes' database with a unique recipe_id. The Profile page is reloaded with the recipe card added under "My recipes". Users can click on the Discover page and see their recipe added to the bottom of the page.
        - If "Close" is chosen, the form is closed and no additions or changes are made to the 'recipes' table, Profile page, or Discover page.
        - If unsuccessful, the Profile page is reloaded and no additions or changes are made to the 'recipes' table, Profile page, or Discover page.
    d. Observations/Results:
        - User Test 1: The user added a recipe with filling everything and submitted it. The recipe showed up in the profile and the discover recipes pages which made the user happy to be able to see it immediately in their profile and also the discover recipes page. The user would have preferred to be able to upload images from the device directly instead of a link. Also teh user suggested to have teh possibility to upload vidoes to teh recipe as well or multiple pictures.
        - User Test 2: When the user attempted to add a recipe, they entered a mix of text and numbers in the cook and prep time fields ("30 minutes"), which was unexpected. They then clicked on the button to add the recipe, however due to their unexpected input, this recipe did not get added to the database and did not show up on the website.
2. Edit Recipe
    a. Description: Users can edit recipes they've created
    b. Expected Input: In their profile, under "My recipes", the user clicks on the "Edit recipe" button on the recipe card for the recipe they want to edit, fills out the form with the changed recipe details, and then clicks the "Save recipe" button
    c. Output:
        - After "Save recipe" is clicked, the Profile page is reloaded. Users can click "View Recipe" on the recipe card for the edited recipe and view all of the changes made. The row representing the recipe in the 'recipes' table is adjusted to match the entries by the user.

        - If unsuccessful, the Profile page is reloaded, no changes are visible on the associated recipe page, and no changes are made to the ‘recipes’ table.

    d. Observations/Results:

        - User Test 1: The user tried editing some stuff to test this feature and they immediately saw the results. The user was confused to not be able to see some stuff like the prep or cook time or others when viewing recipes which made it more complicated to check the changes.
        - User Test 2: The user went to the profile page, hesitated because of the color of the edit button, and then decided to click it. They were then pleased with the form that popped up because if its similarity to the add recipe form, and tweaked a couple of tags, and these edits were successfully saved.

3. Delete Recipe

    a. Description: Users can delete any recipe they’ve created

    b. Expected Input: In their profile, under “My recipes”, the user clicks on the “Delete recipe” button on the recipe card for the recipe they want to delete.

    c. Output:

        - After “Delete recipe” is clicked, the Profile page is reloaded with the recipe’s card no longer visible under “My recipes”. The row representing the recipe in the ‘recipes’ table is deleted.
        - If unsuccessful, the Profile page is reloaded, no changes are made to the user’s profile, and no deletions occurred in the ‘recipes’ table.

    d. Observations/Results:

        - User Test 1: Delete recipe worked successfully with no troubles and the button was easy to find. The user would have preferred to get a warning or something before deleting the recipe in case it was by mistake or something.
        - User Test 2: When tasked with deleting a recipe, the user was initially confused, as they were on the discover page. After going to the profile, the user saw one of the recipes that they had uploaded, clicked on the delete recipe button, and the recipe was successfully removed from the database and never showed up on the website again.

Deployment:

How to run locally

Download zipfile from github repository

Open VS code

Type “docker-compose up-d” in terminal

Open “localhost:3000” on your preferred browser