

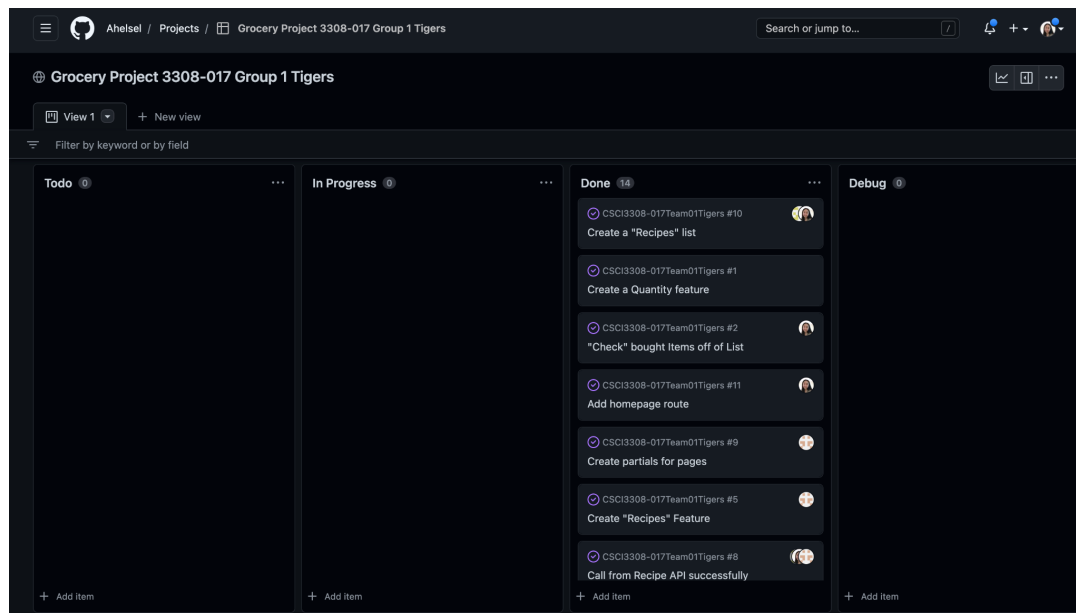
Project Report: Team 017-01

Title: Go Groceries!

Members: Kaitlyn Huynh, Sam Sherman, Evan Thompson, Anthony Helsel

Project Description: Go Groceries is an easy to use application targeted towards college students to help keep track of ingredients that need to be purchased by having a grocery list feature, a separate tab of recipes and ability to view recipe details. This application includes a login/logout feature where grocery lists are tied to a user's account and will be saved upon logging out. In the grocery list, users add items into the list and are also able to add a quantity attached to the item. The grocery list also has a feature in that users are able to cross out an item that's been obtained. If users want to create a certain recipe, they can go to the recipes tab and choose the 'Add ingredients' button which will grab the ingredients of the recipe and add it to the existing grocery list. Users also have the ability to view recipe details, clicking on the button 'View details' will take the user to another page where the recipe instructions and recipes can be viewed.

Project Tracker:



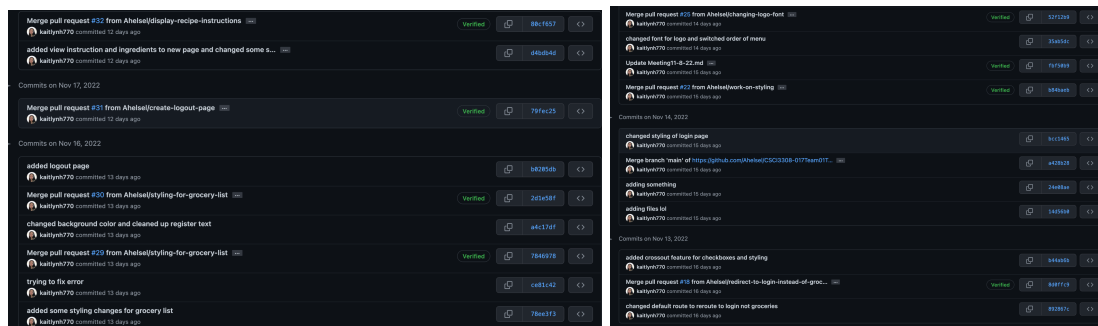
Video:

<https://youtu.be/bPInxKjZVGo>

VCS Link

Contributions:

Kaitlyn: Focused primarily for styling and front-end for the project. Created styling for various pages in the application such as login page, logout page, grocery list, and recipe details list. Also create some routes in the backend such as logout and recipedetails and groceries/checked (when an ingredient is obtained the user can delete the item). Helped develop logic for recipe details by calling the correct route and having it dynamically appear in the front-end. Primarily worked with EJS, Node.js and CSS.

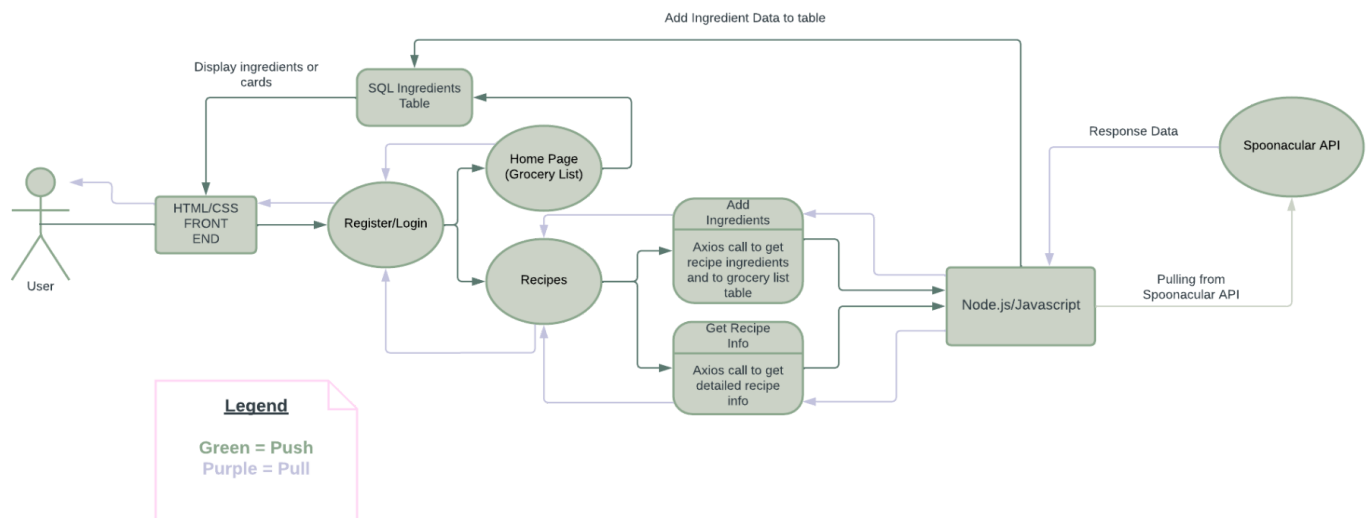


Anthony: Focused primarily on backend and API calls for the application. Created and debugged the recipes page (pulling 24 random recipes from the API) and created logic for adding recipe ingredients into the grocery list. Primarily worked with EJS and Node.js.

Sam: Focused primarily on helping with backend and database (created the Create.sql file) and worked on the register page. Created logic for creating a new user and adding it to the user's table. Also worked with RegEx in putting in password requirements. Worked on some styling with the register page. Primarily worked with EJS, Node.js and SQL.

Evan: Focused primarily on backend and the database. Added backend login, register, logout, and groceries functionality, including all queries to the database within those routes and hashing of passwords. Also created the initial frontend versions for login page, register page, and the partials. Created the database for the server, including a table for username and password, a table for grocery items and quantity, and a table for recipes and their names. Did a couple bug fixes, most notable being a bug where if quantity was inputted as a string the server would crash. Worked with EJS, Node.JS, and SQL.

Use Case Diagram:



Test Results:

Evan:

1. Login Page

Quickly noticed the register button, and clicked. Registered while taking into account the notice of how the password should be formatted. After completing registration the user was redirected back to the login page, where they were able to log in. The user reported that this was “good and rudimentary”.

2. Add and cross out an item from the list

Users had no issues adding groceries and changing the quantity around, but noticed that the groceries don't actually get deleted after they are crossed out.

3. Go to recipes page and add ingredients from a recipe

User did not like the recipes, but found that adding the recipes was really easy. User was quoted as saying, "woah", after the recipe was added to the grocery list.

4. Go to recipes page and view details of recipe

The user found it very easy to open up and read the details of the recipe. User was quoted as rating the experience, "10/10, very simple".

Kaitlyn:

1. Logging In/Logging Out/ Register

- a. Users typed in username just fine, but hesitated when typing in password
- b. One user got confused by the redirection of register and login since they were in the same format
- c. No issues with logging out, said instructions were clear
- d. Had to read the fine print for password requirements
- e. One user got confused by the register and login menu
- f. Changed styling of login and register menu; specifically changed the background color of the register menu for a more drastic change between the 2 menus

2. Adding and Crossing out an item:

- a. Sometimes they wouldn't bother with the quantity feature
- b. There was one instance where the user accidentally clicked enter on the text box and entered an empty item, but it was quickly resolved by clicking the delete button

- c. Made changes: When the user tries to insert an empty item it will not be added to the grocery list
- 3. Going to the recipes tab, viewing all recipes and adding ingredients into grocery list
 - a. Going to the recipes page, scrolling through the page and clicking on the 'add ingredients' button
 - b. Users also read most of the recipes to see which one she would like.
 - c. Interested in the styling, wanted to see more of the page and layout
 - d. Wanted to see what the add ingredients button would do
 - e. Wanted to find a recipe that they were interested in
- 4. Test Case 4: Viewing the details of a recipe
 - a. Wanted to view more information about the recipe, suggested adding measurements for the ingredients
 - b. No changes made due to how the API is structured there isn't a way to grab the measurements in the route that is called