Testing.txt

For the unit tests for the frontend, I divided the frontend into the major components that comprise of the homepage, view meals page, and the grocery list page.

AddMeal.test.js

- Tests the error handling when adding a recipe to the database
  - Ensures all invalid inputs don't result in a new recipe and result in an error instead
- Tests a valid recipe to ensure valid recipes are accepted and created

Equivalence Classes:
Keypresses VS Mouse Clicks

Keypress:

- Adding the ingredient name/meal name

Mouse Click: - Selecting units - Selecting images - Selecting diets

Filter.test.js

- Tests removing a filter from the home page via the dropdown menu

Homepage.test.js

- Tests the ability to change the name of the meal plan
  - From a name → a non-empty name
  - From a name → empty string (sets the name to the default name)
- Tests the ability to add a meal from the menu to the current meal plan
  - The test checks to make sure that the calendar that represents the current meal plan is updated properly

HomepageNewMeal.test.js

- Tests the ability to change the name of the meal plan

Login.test.js

- Tests an invalid login
  - Checks to make sure we remain on the login page
- Tests a valid login
  - Checks to make sure we make it to the home page

Equivalence Classes:
Keypresses VS Mouse Clicks

Keypress:

- Adding the email name/password

Mouse Click: - Clicking the login button

Menu.test.js

- Tests the menu by scrolling the menu to ensure the images are scrollable
- Tests the search bar of the menu
  - Checks to ensure only valid matches are found from the search
- Tests selecting a meal twice
  - Tests the deselection of an item
- Tests the homepage button to redirect to the checklist
- Tests the homepage button to redirect to the view plans page

Equivalence Classes:
Keypresses VS Mouse Clicks

Keypress:

- Search for recipes

Mouse Click: - Clicking a recipe for the menu - Clicking the buttons to redirect to other pages

Registration.test.js

- Tests an invalid registration (invalid password - at least 1 symbol, 1 uppercase, 1 lowercase)
  - Checks to make sure we remain on the signup page
- Tests a valid registration
  - Checks to make sure we make it to the login page

Equivalence Classes:
Keypresses VS Mouse Clicks

Keypress:

- Adding the email name/password/name

Mouse Click: - Clicking the signup button

Sidebar.test.js

- Tests opening/closing the sidebar
- Tests the redirections to other pages via the sidebar
    - Checks to make sure we make it to the other pages
- Tests the settings button on the sidebar
    - Checks to see if the settings dialog shows up
- Tests the logout button on the sidebar
    - Checks that we go back to the login page

Tags.test.js

- Tests updating the tags alignment
    - Ensures the choose all buttons work
- Tests the ability to open and close the tags drawer

Tools.test.js

- Tests the search bar and the ability to clear the search bar

Viewplans.test.js

- Tests the ability to view previous meal plans
- Tests the ability to search for meal plans
- Tests the ability to filter by public/private meal plans
- Tests the ability to copy a meal plan to the current meal plan