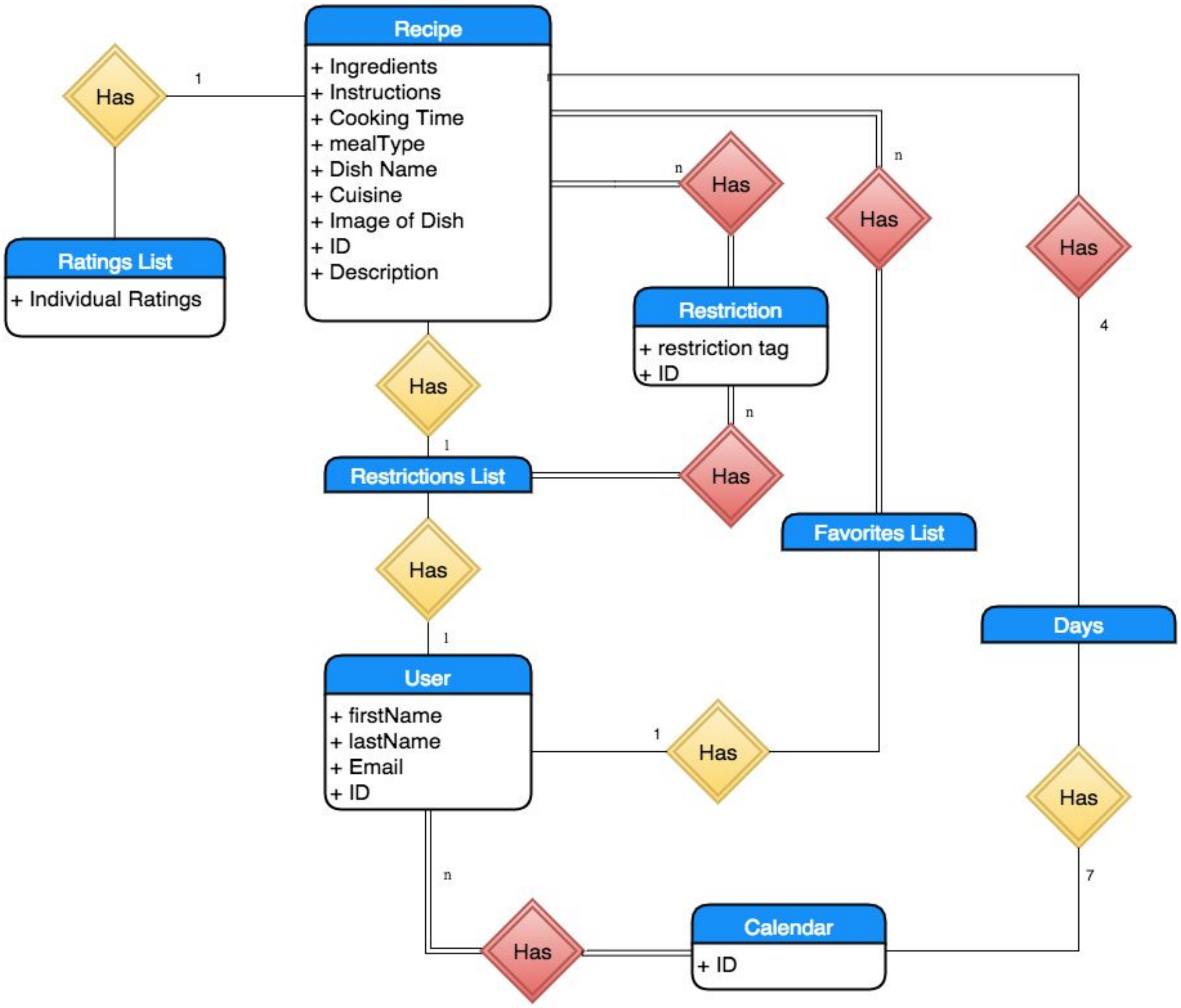
Startup Submission 04: Dynamic UI Mockups

Startup: kuk
Group Members: Stephanie Xie, Emma Stephenson, Syeda Zainab Rizvi, Karen Chau, Jennifer Zhu, Eleanor Ewing, Farida Sabry

New ER Diagram:



Stephanie Xie (Results Page + Navbar Search)

Bugs:

- Average ratings of each recipe is a bit buggy: they all display the correct ratio of filled to unfilled stars, but only “Brownies” has the correct total number of stars (5) - the other recipes all have 6 stars
    - Attempted to fix this by subtracting 1 from findAverageRating in the render method, but while this did subtract 1 star from all the total stars, “Brownies” now only displays 4 stars in all. Will need to fiddle around with how the stars are displayed in Render.
- Searched text is not staying in the search bar after the results are pulled up
    - Tried initializing the value to the query after passing in the query via a prop, but so far only the initial searched text stays in the results page’s search bar - and only if it was searched from the navbar’s search bar, not the results page’s search bar (the text doesn’t stay in the navbar’s search bar, for some reason).
    - May need to use a componentWillReceiveProps method, similar to the one in ResultsFeed, so that the search bar will update with the query accordingly instead of going blank after the first search.
- Need to order the results in terms of relevance. Right now the results display all the recipes that have a word in their titles that matches only the FIRST WORD of query.
    - Need to fix the “Encountered two children with the same key” error that pops up if the query contains multiple words.
    - To sort the recipes: loop through all the words in the results titles and compare them to all the words in the query; the title with the most matches is displayed at the top of all the results. If it happens all the recipes only have one word in the title that matches the query, then display the results in ABC order.

React Components:

- ResultsFeed: displays all the recipes that match the search query, calls the other components
- ResultsItem: each recipe on the results page
- Searchbar: search bar on the results page
- Navbar: displays a navbar containing links to the other pages and a separate search bar

Server Functions:

- findRecipe(searchText, cb): returns an array of the recipes whose names match the searched keyword

Pages:

- Results page: displays all the recipes that match the search query

Emma Stephenson (Recipe Page)

Bugs:

- Supports the favoriting of multiple recipes (can fix this with a server method, just have to come to office hours to learn how to do callbacks within callbacks)
- Need more detail on the calendar - currently only supports adding to the dinner of the current week (need to figure out a week by week pop-up)
    - Should probably also make the alert box prettier
- Users can't yet select their own rating for the recipe (though the calculation of the average rating of the recipe does work, and allowing users to select their own rating is just going to be a matter of figuring out how to get at parent html elements)

React Components:

Recipe: displays a recipe according to the recipe id passed in via the url

Server Functions:

- addFavorite(recipeId, userId, cb): when given a user id and a recipe id, adds the recipe's id into the current list of favorites and returns a resolved user item
- removeFavorite(recipeId, userId, cb): checks to see if the recipe is in the user's list of favorites, if it is, remove it from the list
- checkUserFavorites(recipeId, userId, cb): checks the user's favorites list to see if the given recipe is on the list. Returns a boolean (recipe is/is not a favorite). This is the one that, when tested with console.logs, returns the right thing, but I can't get JS to wait for it to return before moving on to the next thing.
- addRecipeToCalendar(recipeId, userId, day, cb): takes in a day of the week (from the dropdown menu) and the current user and recipe. Adds the recipe to the "dinner" slot of the current week.

Syeda Zainab Rizvi (Calendar Page)

Bugs:

- Even though the right recipe (for the right meal) is deleted in the database, once the page is refreshed, the deletions are no longer in order. This can be fixed by displaying the recipes in the right order. Right now, I am just mapping the array, regardless of meal type.
- The next week and previous week buttons need to be clicked twice in order to work. This is because I could not figure out how to counter the lag from setState.

React Components:

- Calendar: page that users see and has all calendar entries arranged
- CalendarEntry: each entry in the calendar according to day and meal

Server Functions:

- getCalendarData: returns the Calendar for the specific user. This is used by the Profile page
- removeRecipefromCalendar: this removes the recipe by the specific index. This method is working correctly but it is not being displayed by the React component as I am simply mapping recipes, regardless of meal.
- getProfileCalendarData: returns the specific meals according to the day and also writes those meals in the user's state array so they can be displayed by the Calendar React Component

Pages:

- Calendar page
- Recipe page (displays any recipe)

Karen Chau (Favorites Page)

Bugs:

- The right recipes are selected when the user clicks on a checkbox on the filter sidebar, but the integration of updating the favorited recipes according to the filter tags still needs to be implemented so the filter panel and apply filter button is not complete.
    - Will work on returning filtered recipes from the Filter component to Favorites component through a callback function
    - Will work on multiple filtering functionality so will have to cross-check the filtered lists
- The sorting of the favorited items according to their properties needs to be implemented. The default is the order in which the items are added to the favorites' list.

React Components:

- Favorites: displays all the items that the user has favorited
- FavoritesItem: contains the thumbnail that represents each recipe that has been favorited along with its name as a link to its recipe page.

Server Functions:

- findRecipesFromId: takes in a list of recipe ids and a callback function and returns an array of the recipes whose ids match the list of recipe ids

Pages:

- Favorites page: page that displays the user's favorited items, arranged in the order that they have been favorited. On this page, the user can also favorite/unfavorite the recipes by hovering over the recipes and clicking on the button. Refreshing the page will show these changes.

Jennifer Zhu (Instant Recipes Page)

Bugs:

- Clear the search does not get rid of the ingredients user has entered.
  - Resolve by getting the individual react components to communicate with each other upon pressing the Clear button
- Implementing the recipes with only these ingredients functionality
  - Resolve by adding in another server method that will check if all the ingredients the user entered are in the recipe.
- Having the results stay in the feed after going to another page (e.g. a recipe) and then clicking the "Back" button on the browse (can resolve by storing it temporarily)
- User entered a two word ingredient, will only match on the word individually (i.e. sesame oil will match ingredients of sesame or oil solely)
  - Will need to rewrite the algorithm to find by ingredients, perhaps even take up your previous suggestion of adding ingredients as entities in the database

React Components:

- Instamode: page that users see, calls each of the other components
- InstaIngredient: each individual ingredient the user enters, manages removing an ingredient
- InstaIngredientsList: display for all of the ingredients, from when they are entered into the form and appear in the ingredients box
- enterInstaIngredient: form for entering an ingredient, handles submitting the ingredient when pressing enter
- instaResults: will load in the recipes from the server method and then pass them into resultItem

Server Functions:

- findRecipeByIngredient(ingredientsList, cb): returns an array of the recipes who contain at least one of the ingredients listed (will improve to have only the ingredients listed by the next assignment)

Pages:

- Instamode Page: page to enter in the ingredients the user has in their refrigerator and directs them to a recipe suited to those ingredients.

Eleanor Ewing (Profile Page + Navbar except search)

React Components:

- Navbar: displays the navbar and contains link to various parts of the website.
- ProfileFeed: displays the entire profile page, calling each of the following components
- ProfileMeal: represents a single meal in the "upcoming meals" portion of the favorites page
- ProfileRestrictions: represents the entire profile restrictions part of the profile. Sets a user's dietary restrictions in the database

Server Functions:

- getRecipeSync: returns a resolved recipe object
- getUpcomingMeals: gets the next 4 meals for a particular user. As of now, Monday is the default current day.
- getRestrictionStrings: takes in an array of restriction ids and returns an array holding the restriction tag names
- getProfileData: takes in the user id and a callback function and calls the callback function on an object that holds the profile of the specified user and another field that holds the user's next four upcoming meals
- getUserRestrictions: takes in the id of a user and a callback function and passes into the callback function an array of the tags of the user's dietary restrictions
- addUserRestriction: takes in a checkbox DOM element, a user's id, and a callback function. Adds the given restriction id (identified in the checkbox) to the user's dietary restrictions and passes to the callback function an object that holds user's new restriction references and the checkbox element
- removeUserRestriction: takes in a checkbox DOM element, a user's id, and a callback function. Removes the given restriction id (identified in the checkbox) from the user's dietary restrictions and passes to the callback function an object that holds user's new restriction references and the checkbox element

Pages:

- Profile Page

Farida Sabry (Browse Page + Filter Sidebar)

Bugs:

- I am able to get the right recipes when the user clicks a specific checkbox on the filter sidebar using the server methods I wrote, but I still have not quite figured out how to get the filter to talk to the feed and update it.

React Components:

- BrowseFeed: This is the first page that users see and are able to navigate to other pages from. It displays a feed of recipes and a filter sidebar to filter out certain dietary restrictions.
- FeedItem: essentially the hexagons in the feed that represent recipes
- Filter: filters the recipes being shown in the feed by having users check different dietary restrictions

Server Functions:

- getFeedData: takes in an integer, amountofRecipes, and callback function and passes into the callback function an array of recipes of length that matches that integer.

- getFilteredRecipes: takes in an array of recipe ids, and callback function and checks the recipe ids against the recipe collection for matches that don’t have those ids.
- getRestriction: takes in a checkbox DOM element, and a callback function and passes into the callback function an array of recipes listed under that specific restriction (which is extracted from the checkbox)

Pages:

- Browse Page: This is our home page where you can browse recipes and filter them according to your cravings, desires, and diet restrictions.