Final Report - Submission 6, Adding a Database

Special Server Setup Procedure

As in the previous submission, if your server has any advanced features that require additional setup besides npm install and node src/server.js, include a section in the report that describes how to set up these features. Include details on how to tell if the feature is working properly.

Our server does not require additional setup besides npm install and node src/server.js.

Individual Contributions

Include a section that describes each startup founder's contribution. Each startup founder should be responsible for at least one product feature and its HTTP route(s). Name the feature and the HTTP routes that the founder updated to use the database. Each founder *must* be responsible for converting at least one HTTP route.

Browse Page (Farida Sabyr):
- <u>Feature:</u> The Browse page, where users can browse recipes and filter them according to dietary restrictions.
- <u>Updated HTTP Routes that use the database:</u> PUT /feed
- <u>Bugs:</u> Filtering isn't quite working yet

Instamode Page (Jennifer Zhu):
- <u>Feature:</u> The Instamode page, where users can enter in ingredients and search for recipes that contain those ingredients.
- <u>Updated HTTP Routes that use the database:</u> GET /instaresults/:ingredientString, GET /instaresults/ingredientsONLY/:ingredientString

Results Page (Search) (Stephanie Xie):
- <u>Feature:</u> The results page, which contains the result items (recipes) from the search query.
- <u>Updated HTTP Routes that use the database:</u> POST /results
- <u>Note:</u> search server route in the client now handles errors

Profile Page (Eleanor Ewing):
- <u>Feature:</u> The profile page, which contains a user's meals for that day (hard coded to Monday), and allows a user to set/adjust their dietary restrictions.
- <u>Updated HTTP Routes that use the database:</u> GET /user/:userid, GET /user/:userid/restriction, PUT/DELETE /user/:userid/restriction/:restrictionid

Recipe Page (Emma Stephenson):
- <u>Feature:</u> The recipe page, which displays any given recipe. Users can add to their favorites and to their calendar directly from this page.
- <u>Updated HTTP Routes that use the database:</u> PUT /recipe/:recipeid/user/:userid/calendar/:dayid, PUT /recipe/:recipeid/favorites/check/user/:userid, DELETE /recipe/:recipeid/favorites/user/:userid, PUT /recipe/:recipeid/favorites/user/:userid, GET /recipe/:recipeid/

Favorites Page (Karen Chau):
- ● <u>Feature:</u> The favorites page, which contains the favorited items of the user. Users can remove the favorited items from this page.
- ● <u>Updated HTTP Routes that use the database:</u> GET /user/:userid/favorites/

Calendar Page (Syeda Zainab Rizvi):
- ● <u>Received an extension till Sunday 6 pm.</u>

Lingering Bugs / Issues / Dropped Features

If your application has any lingering bugs, issues, or missing/dropped features, include a section listing these.Similarly, if any particular features or components are not using the database, please mention those features here.

- ● <u>Yummly API integration:</u> The Yummly API will only return ingredients, the time it takes to prepare the dish, meal type, and cuisine. It does not return instructions for making the dish, which is crucial for our site. We are currently looking into other APIs that will also return instructions, but are not having much luck. We may not be able to incorporate communication with an API for populating our database.
- ● <u>Search:</u> Query text does not disappear from the search bar after navigating away from the results page.
- ● <u>Filter:</u> The filter on the browse page isn't fully working yet, so I left out unfinished code as not to produce errors.