

Completed Tasks:

1. Created Index.html and frontend functioning search bar (Abel Theodros, Michelle Dozal)
2. Frontend Star Rating System so the user can rate the recipe (Michelle Dozal)
3. Frontend Comment Section/Submit & Cancel button to Post comments (Michelle Dozal)
4. Created the savedrecipes.html,searchhistory.html,top_recipes.html,under100cal.html (Kinjal Mugatwala)
5. Created a drop-down menu that has the list of other pages the user can visit (Kinjal Mugatwala)
6. Using Python and Flask for the backend instead of NodeJS as NodeJS was becoming more difficult to incorporate (Terry Jung)
 - a. Files created:
 - i. Server-side = server.py (utilizing libraries from Flask)
 - ii. Client-side = index.html (contains html, css, and js implementation), results.html (used to display the search results for now)
7. Created a parser() function using the csv library that reads in each value using the reader() function and tries to separate the values based on the delimiter (Terry Jung, Steven Joseph)
8. Passing in values that user inputted in the search bar to server.py to create some output, searched input value can be seen in the url (Kinjal Mugatwala)
9. Added the ability to see the comments on the page (Michelle Dozal)
10. Added separate “get” functions(parsers) for specific data on different csv files(Terry Jung)
11. Wrote implementation to save recipes based on the user inputting the recipe_id value (Kinjal Mugatwala)
12. Enabled a button that lets user clear all of their saved recipes → clears saved.csv content (Kinjal Mugatwala)
13. Added 3 graphs for analytics - scatterplot, bar graph, and boxplot (Kinjal Mugatwala, Terry Jung)
14. Search features lets user view several entries of recipes that match the searched category (Abel Theodros, Michelle Dozal)
 - a. Created edge cases in case of empty category and search value inputs (Abel Theodros)
 - b. Helper functions for traversing .csv files by column and row
15. Added the last 3 graphs to finish our analytics (scatterplot, histogram, histogram) (Kinjal Mugatwala)
16. updated sprint 4’s first three analytics to incorporate the entire csv files like RAW_recipes.csv and interactions_test.csv. Because of dimension errors, the first three analytics are now histogram, bar graph, box plot (Kinjal Mugatwala)
17. Added functions into the back end that help to find and update certain entries specified by inputs, and edit, append, or delete them as the user sees fit. (Steven Joseph, Terry Jung)
18. Improved performance of analytics that use RAW_recipes.csv file as they were the slowest of all the analytics (Kinjal Mugatwala)
 - a. Created attributes in main() function so that the RAW_recipes.csv file is only read once when the server begins to run
19. Completed the update, append, delete, writer functions to make edits to interactions_test.csv file (Terry Jung, Steven Joseph)