Midterm Documentation

Alejandro Martinez

East Los Angeles College

CIS 112: Javascript Programming

Professor Rodriguez

Introduction

The goal of project Nutrientry was to make an easy to use and information website where a user could input their physical statistics and get information to help them reach their physical goals. Physical strength is important for providing an appropriate vessel for the mind to flourish. The design for the website takes elements from different competitors already on the market which include, MyFitnessPal, chronometer and LoseIt! The project consists of 4 major parts that are as follows. The route logic, the forms and submission logic, the database models using sql alchemy, and finally the static files which are CSS and Javascript.

The user flow goes as follow:

1. Landing Page
2. Register or Log In
3. Input health information to get a detailed overview of health goals
4. Learn about different dieting and lifestyle changes
5. Search and input foods using an API search feature
6. Log and track calories based on date and current health data

There are currently 8 major pages where the user can use and see information for their profile. The 8 pages are listed below:

1. Landing Page
2. Learning Page
3. Sign Up Form
4. Log In Form
5. Metrics Form
6. Dashboard
7. Search Foods Form
8. Display Food Choices (not yet a form)

There is currently no contact page but information for reaching me is displayed on the footer which persists in the HTML since I am using Jinja templates and a base to get persistent navbar and footers. The source code will be attached to the file. But the file structure is pictured below:

![Text

Description automatically generated]()

The main components and features of the website consist of being able to get information about the current state of a user’s health using their body measurements and then also being able to track and suggest the ideal diet breakdown for their meals. Currently the calculations are all being done behind the scenes and the api call is not apparent during the search. I am using the users to fill out my own food database that I can later turn into an API as well. By ensuring that a search is never done twice for the same foods, then eventually I will be able to fill out my database with the most common foods.

Before a lot of changes were implemented I was using a very basic boot strap layout and hardcoding the data. Now data is being generated dynamically and a lot of styling has been added but not finalized. Styling I will be leaving until the end so I can make everything look cohesive with just minor tweaks towards the end rather than building each page as I go since some pages may change layouts or be removed completely before the completion of the project.

Regarding failures there are many currently known existing bugs that need to be addressed. The site breaks very easily if a user searches for a previously searched item. I will be addressing that next by using if-statement logic to check if a food that a user is attempting to search for has already been extracted and added to the database with a matching query term. If a specific phrase or word has already been searched then the database should respond with results without making an API call. In addition, I currently need to deal with the data coming back from searches. I have the entries table set up for the database but I need to be able to populate a select form or think of a better way to display the data for a user so that they are able to submit and I can use the submission date in conjunction with a user id and food data to make an entry for that date. A user should be able to make multiple entries for a single day.

The Nutrientry project will be reaching a conclusion fairly soon and should only require some data handling and no more major functionality will need to be added. I have a list of ideas of how I can handle the currently known errors and once I deploy and am able to capture some user data I will have a better understanding of how I should be handling certain routes.

# References

(n.d.). Retrieved from Bootstrap4 Docs: https://www.getbootstrap.com/