Meal Planner

Our Group

- Nikhil Chakka
- Trajan Pei
- Kashyap Challapalli
- Tien Vu
- Jason Pabelico
- Angela Nguyen

Description of the Project

This product allows you to track your meals on a day to day basis and get an idea of your daily intake of calories, carbohydrates, protein, etc. It encourages people to organize their meals and become more healthy.

Tools

- Project Tracker = Github Project Board
- VCS repository = Github
- Database = PostgreSQL
- IDE = vsCode
- UI Tools = HTML, CSS, EJS
- Application Server = NodeJS
- Deployment environment = localhost and CU Boulder host

Project Tracker (Github)

The project tracker on github was really helpful for general project management. With things like prioritizing what we want to get done first, breaking problems down into smaller tasks, splitting those tasks up, opening issues and making decisions about our project. We gave this tool 4 stars because while it is useful for the reasons listed above, we still used other methods of communication like messaging to give more specifics about the project.

VCS Repository (Github)

Github was amazing in the sense that it allowed all of us to work collaboratively on single project. At times, the merge conflicts were confusing to solve but in the end, it was very convenient to use. Likewise, the existence of branches helped us organize code by individual and allowed us the convenience of keeping it separate from main until it completely done. Though there were some annoying times with Github, it was overall a great VSC repository for this project.

Database (PostgreSQL)

PostgreSQL was a great tool for our project, as it was a great way to store any sort of data we needed in a database. There weren't many problems we ran into, but just looking back at times when we didn't have SQL, like the calendar JavaScript lab, it would've been terrible and very frustrating to use a javascript file for our data storage. We gave this tool 4 stars because of the utility and efficiency that this tool provided, as PostgreSQL is an integral part of our project, and we couldn't have done it nearly as well without it.

IDE (vsCode)

We used visual studio for our integrated development environment and it worked great. It seamlessly allowed us to work with different types of coding languages along with debugging and syntax highlighting. Overall it has a great simple UI and does everything we needed it to and since it’s what we have used all year we were all familiar with it and were able to work with it without any problems.

UI Tools (HTML, CSS, EJS, JS)

HTML, CSS, EJS, and JS were great tools for what we were trying to accomplish. Although they are the most barebones UI tools, they were able to get everything we needed done. We gave this tool 4 stars since there are more efficient and intuitive tools available nowadays, and raw HTML, CSS coding can be frustrating at times but it does get the job done eventually.

Angela's Slide

Application Server (NodeJS)

- Allows users to personalize their meal calendar
  - Add
- Allows users to add their own unique meals to database
  - Nutrition facts of each meal
  - Create Meal Names

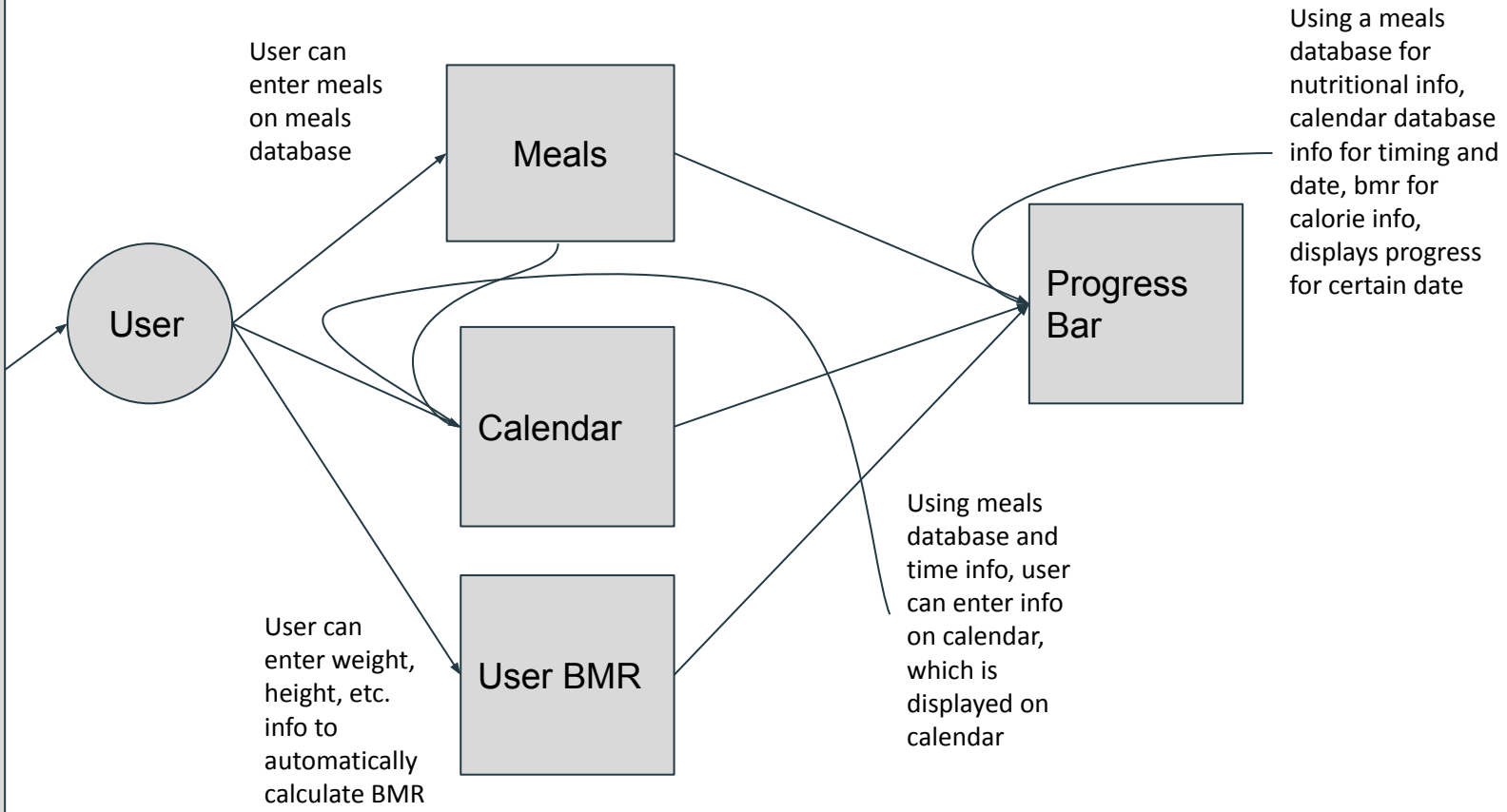
NodeJs was a great tool for us to allow users to modify their individual accounts and unique meals. For instance, our calendar feature made itself available to add certain meals to a day. In addition, we allowed users to create their own database of meals to make the database more flexible and convenient to add. NodeJS let us create the perfect personalised meal planner for customers which was the main point of our project (5 Stars).

Deployment Environment (localhost and CU Boulder)

- The development and testing for our website was mostly done on local host.
  - Uploaded through branches
- After the final version of our project was completed, we deployed our code to the CU Boulder Cloud.

Architecture Diagram

Auth



Challenges we Faced

The main challenge we faced was proper communication. This was a problem because at times we were unable to properly assign tasks to one another to help split up the workload. We overcame this by assigning people tasks and holding them accountable for it, and reaching out personally to people. More communication may also have allowed us to meet more as a team and the knicks and knacks of our project could have been improved.

We did also face some challenges when it came to git and github, there were some accidental changes to main that had to be reverted and some other minor conflicts. Mainly with inexperience with git. We overcame this by talking to each other about our conflicts or simply googling the problem. This however, did not have too much of an effect on our original project plans.

Other challenges were very minor and we were able to overcome with little effort.

Project Demo