Final Report

Rohandra Macolm

Dobrian Zaprianov

Diamond Zetty

Dorrell Zimmerman

CMSC 495 6380

9 July 2024

Table of Contents

1. Introduction………………………………………………………………………………………………3
2. Overview……………………………………………………………………………………………………..3
3. Requirements Specification…………………………………………………………………………..4
4. Project Design……………………………………………………………………………………………..4
5. Project Evaluation……………………………………………………………………………………….6
6. Design and Alternate Designs……………………………………………………………………….6
7. Development History…………………………………………………………………………………..8
8. Discussion………………………………………………………………………………………………….12
9. Conclusions……………………………………………………………………………………………….13
10. References………………………………………………………………………………………………..14
11. Appendix…………………………………………………………………………………………………..15

Introduction

The Diet and Nutrition Application project aimed to create a user-friendly application to help individuals calculate metrics for general dietary health and track dietary and nutrition choices. The motivation behind this project was to provide an accessible tool to support healthier lifestyle choices. Our application provides users with a method to calculate Body Mass Index (BMI), Basal Metabolic Rate (BMR), and a Food and Calorie tracker.

Overview

Our approach when developing this application involved developing a collaborative environment using the agile development process. Various tools were used in the creation of this application including:

* Project Management Tools: GitHub for version control
* Development Tools: Python Flask and Bootstrap for front-end development and Python Flask for back-end development.
* Design Tools: Bootstrap for, UI/UX Design
* Testing Tools: Manual testing and UnitTest for automated testing.

This application was developed with a four-person team working together, each with specific roles and responsibilities to ensure that the application was developed effectively.  Below are the list of the team members along with their role for the development of the application:

* Dorrell Zimmerman-Team Leader
* Rohandra Macolm-Back End Developer
* Dobrian Zaprianov-Front End Developer
* Diamond Zetty-Test Planner and Technical Writer

Requirements Specification

Functional requirements for the application whereas follow:

* Body Mass Index (BMI) Calculator
* Basal Metabolic Rate (BMR) Calculator
* Food and Calorie Diary

Non-functional requirements for the application were as follows:

* User-friendly interface
* Informative infographics

Project Design

Architecture

This application used Python Flask for both server and client- side logic. The server built with Python Flask, the server processes requests, serves HTML and CSS files, and interacts with user inputs (Welcome to Flask — Flask Documentation (3.0.X), n.d.). Bootstrap was used for the user interface for styling and providing a responsive design.  As Bootstrap is a free open source tool for front-end framework it was helpful for the design and implementation of the user interface (Zola, 2022) .

Implementation

The application starts with a general home screen welcoming the user providing options that link the user to the BMI Calculator, BMR Calculator, and the Food Calorie Diet.

BMI Calculator

The implementation of the BMI Calculator allows the user to input their height, weight, and gender. The server then calculates the BMI using the BMI formula and displays the results to the user. An infographic is located under the calculation so that user can reference their calculation to correct category which would indicate if the result fell under underweight, normal weight, overweight, or obesity. An additional infographic is present to show the user how BMI is calculated.

BMR Calculator

The BMR Calculator allows the user to know their height, weight, age, and gender. The server then calculates the BMR using the Harris Bendict equation. The result is displayed, indicating the number of calories needed per day to maintain the user's current weight. An infographic is present to show the BMR calculation and how it is attained.

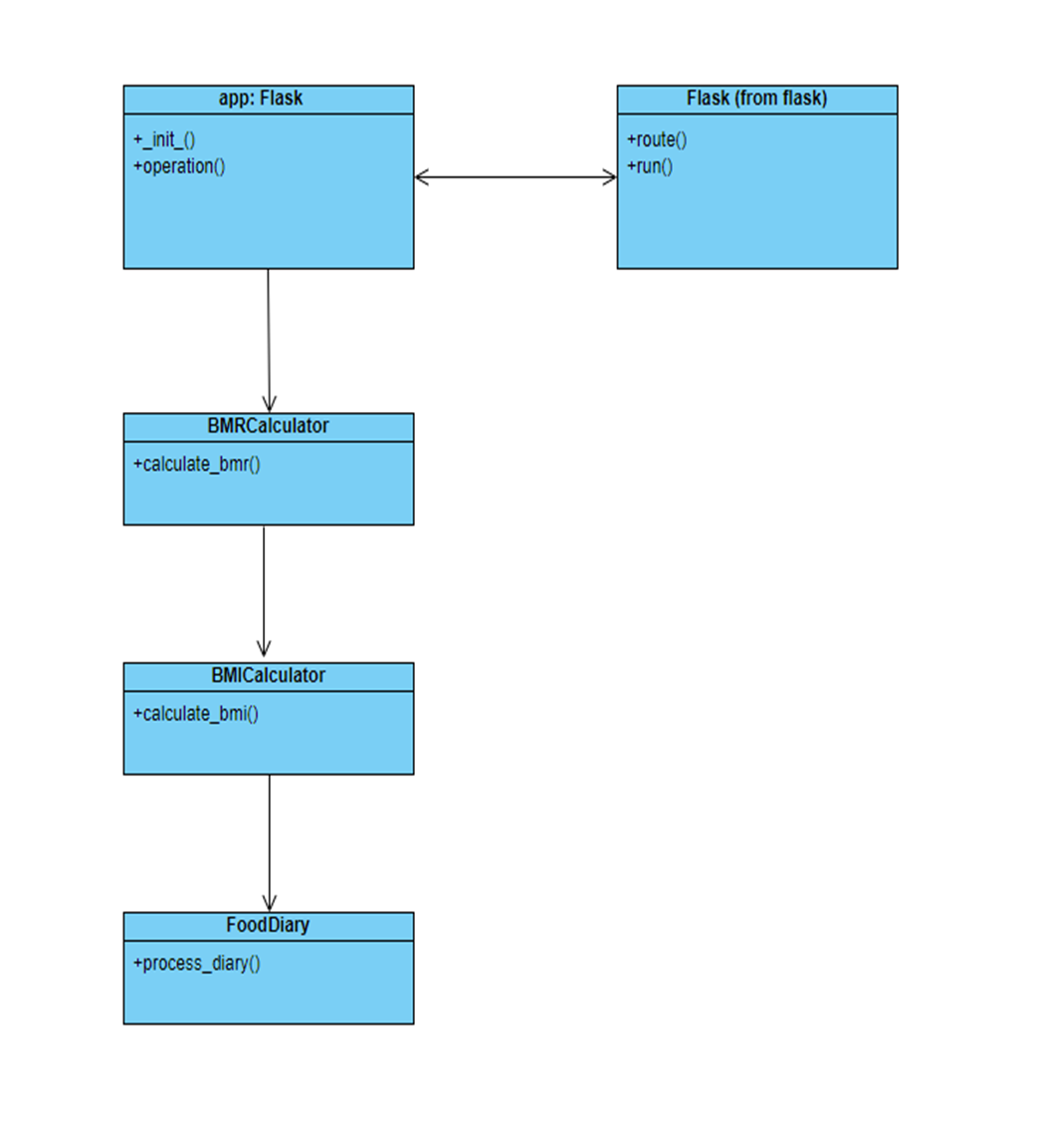
Food and Calorie Diary

The Food and Calorie Diary allows the user to input food items and the number of calories for each entry. The total number of calories is then calculated at the bottom. An infographic and link for MyPlate.gov is at the bottom to give users more information for nutritional guidance.

Project Evaluation

The testing strategy for our application included unit testing and end-to-end testing. Unit Testing was done using unittest, which covered testing individual components and functions in our framework (GeeksforGeeks, 2024). End-to-end testing was done to simulate real user scenarios and interactions for the application. The application validates and updates until all tests are passed. Below are the updated test plan and unittest that were used for the application.

Design and Alternate Designs

The original application UML is shown below for the design of the application. :  Figure 1: Application UML

Based on the test results done with the end-to-end testing, UI components were improved for errors to be correctly addressed. Infographics were added to provide a better understanding of calculations and enhance user experience of what data provided meant. Automated testing was done during every code modification to ensure that all routes and functions worked correctly.

Development History

The application's development history can be noted with the timeline developed during the project planning process.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Duration (days) | Start Date | End Date | Personnel |
| Project Requirements   * Writing * Review * Revision | 6    3  2  1 | 20 May 2024 | 27 May 2024 | Rohandra Macolm  Dobrian Zaprianov  Diamond Zetty  Dorrell Zimmerman |
| Project Plan   * Writing * Review * Revision | 6  3  2  1 | 20 May 2024 | 27 May 2024 | Rohandra Macolm  Dobrian Zaprianov  Diamond Zetty  Dorrell Zimmerman |
| Project Design   * Writing * Review * Revision | 6  3  2  1 | 27 May 2024 | 4 June 2024 | Rohandra Macolm  Dobrian Zaprianov  Diamond Zetty  Dorrell Zimmerman |
| Phase 1 Source   * Development * Test * Review | 6  3  2  1 | 4 June 2024 | 10 June 2024 | Rohandra Macolm  Diamond Zetty |
| Project Test Plan & ICD   * Writing * Review * Revision | 6    3  2  1 | 11 Jun 2024 | 17 June 2024 | Dobrian Zaprianov |
| Phase 2 Source   * Revise * Test * Finalize | 6  3  2  1 | 18 June 2024 | 24 June 2024 | Rohandra Macolm  Diamond Zetty |
| User Guide   * Writing * Review * Revision | 6  3  2  1 | 25 June 2024 | 1 July 2024 | Dorrell Zimmerman  Dobrian Zaprianov |
| Final Deliveries   * Writing * Review * Revision | 6  3  2  1 | 2 July 2024 | 9 July 2024 | Rohandra Macolm  Dobrian Zaprianov  Diamond Zetty  Dorrell Zimmerman |

The timeline shows a breakdown of length all team members must provide initial deliverables during each week. This provides time for work to be reviewed and finalized during weekly meetings to ensure all team members have a stake in development process and what is turned in. All items were done under their estimates allowing plenty of time for revisions and reviews to provide a well-developed application for delivery.

An overview of the two phases of this development process where as follows:

Phase I

Rohandra Macolm and Diamond Zetty began the initial build of the application together. Due to both being knowledgeable on their given tasks and the coding practices required completion of the initial version of the application met the initial timeline. Dorrell Zimmerman ran unittest to ensure that all pathways and functions interacted correctly.

Phase II

During Phase 2 of application development Ms. Macolm and Ms. Zetty began the process of updating issues that were found during Mr. Zaprianov’s testing. Ms. Macolm updated the code to alter the calculation to shorten the number of decimal places displayed during the calculation result. Also, the code was modified to prevent the ability of a number zero or less being used during the calculation. Ms. Zetty began updating the user experience by providing slight alterations to the design and additional infographics to better increase the user experience. Mr. Zaprianov and Mr. Zimmerman then initiated testing again not to ensure all issues were fixed and that application still worked as appropriate and modified code to fix any still underlying issues.

Discussion

Lessons Learned

Lessons learned through this process are the importance of setting up thorough requirements and an adequate plan to assist in the development of an application. Having these things already thought out assisted in the development of the applications as it provided a path to completion. The importance of flexibility and communication is an important item during development. With all individuals being in different time zones and having different work hours, being able to remain open to what issues might come up and constant communication assisted in ensuring that the project was completed with little to no issue.

Design Strengths

The strength of this application is that in its current form it is very user friendly. It provides all information for the calculation and its meaning for the user to be able to use the application effectively. On the back end the code is very robust and is easy to follow. This allows any future modifications to the application to be easier if the same coding practices are followed when adding any new feature to the application.

Limitations

Current limitations for the application are that users must have one that runs a flask application. This does limit the user base as it would require the knowledge to be able to deploy it for widespread use. Another limitation that is obvious is that the food and calorie diary require the user to input all food and calorie information requiring an additional item that the user would have to provide to user this function of the application.

Suggestions for Future Improvements

Future improvements to this application would comprise of many additional functions being added. The first substantial change would be to include a food database, this would provide an easier way for users to calculate calories used and not need them to already know the number of calories that each food item consisted of. Secondly, would be the incorporation of AI to provide personalized food recommendations based on BMR and BMI results. Another item would be adding social features to the application to add community support for the exchange of health-related ideas and motivation. Finally, providing a method that can be deployed to increase the number of users that have access to use it.

Conclusions

The Diet and Nutrition App successfully addresses the need for a comprehensive diet tracking tool, offering BMI and BMR calculations along with food and calorie tracking features. Through a collaborative development process and rigorous testing, we delivered a robust and user-friendly application. Future improvements will focus on enhancing personalization and expanding the app's capabilities to provide even greater value to users.

References

GeeksforGeeks. (2024, June 12). Unit testing software testing. GeeksforGeeks. https://www.geeksforgeeks.org/unit-testing-software-testing/

Welcome to Flask — Flask Documentation (3.0.X). (n.d.). https://flask.palletsprojects.com/en/3.0.x/#user-s-guide

Zola, A. (2022, August 3). Bootstrap. WhatIs. https://www.techtarget.com/whatis/definition/bootstrap

Appendix

System specification:

Developmental Platform:  Intel Core i5 1.5 GHz, 2 Gigabytes memory, 250 GB hard disk, Windows 10, or Mac OS X 10.15 or higher, PyCharm

Operating Platform: Any system able to run a localhost environment

User Guide

User Guide on how to use application can be accessed below: