

1. Introduction

Description (paragraph describing the system): The system will let a user choose to track their budget and their health goals. For the budget section, a user can input their overall budget over a certain period (week/month). For the health goals, a user can input which goal they want to pursue, whether that is fat loss, weight gain, or maintenance. The system will recognize the input from the user, and then the system will update the different respective counters such as calories and macronutrients. As users add their foods and their respective prices, the system will automatically update the budget and even keep track of the meals that the users create. Value proposition: "Keep track of your finances and health without worrying about your college classes."

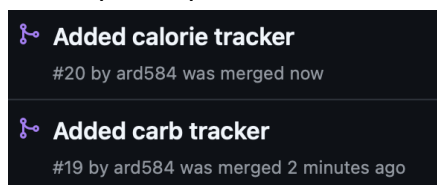
MVP(main features of the MVP): Budget calculator, meal tracker, health goals (eg. weight, calories, macronutrients, exercise)

Repository link: <https://github.com/loganb7869/CS386-Project-Repo.git>

2. Implemented Requirements

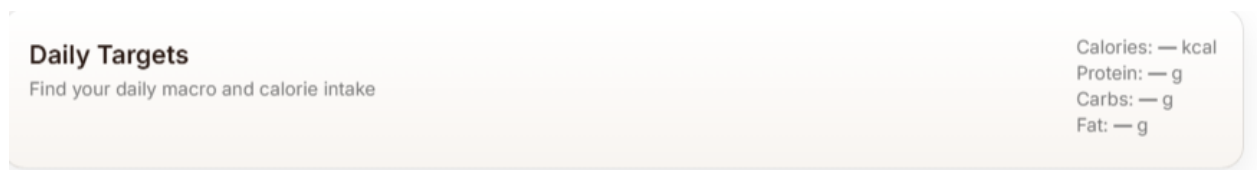
2.1 Logan Bankert

Github pull request link for:



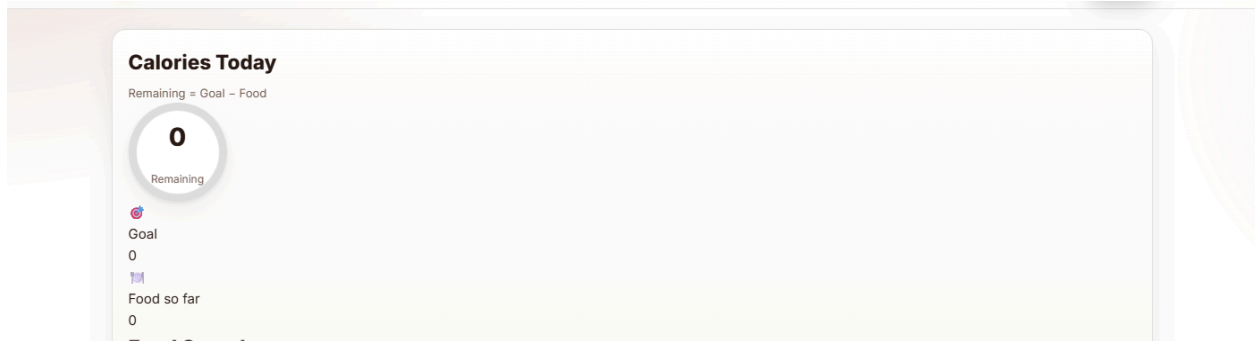
Requirement 1

1. Requirement(User story from Repo): **As a student athlete, I want to keep track of my carb intake so that I can carb-load more effectively.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/3>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/19>
4. Implemented by: Logan Bankert
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/main/Unittest_2.1.A
7. Visual evidence:



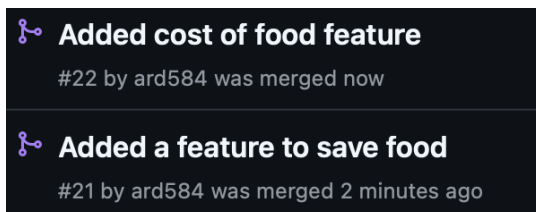
Requirement 2

1. Requirement(User story from Repo): **As a busy student, I want to keep track of my calorie intake so that I eat enough to not be tired during class.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/4>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/20>
4. Implemented by: Logan Bankert
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.1.B
7. Visual evidence:



2.2 Rita Bolanos

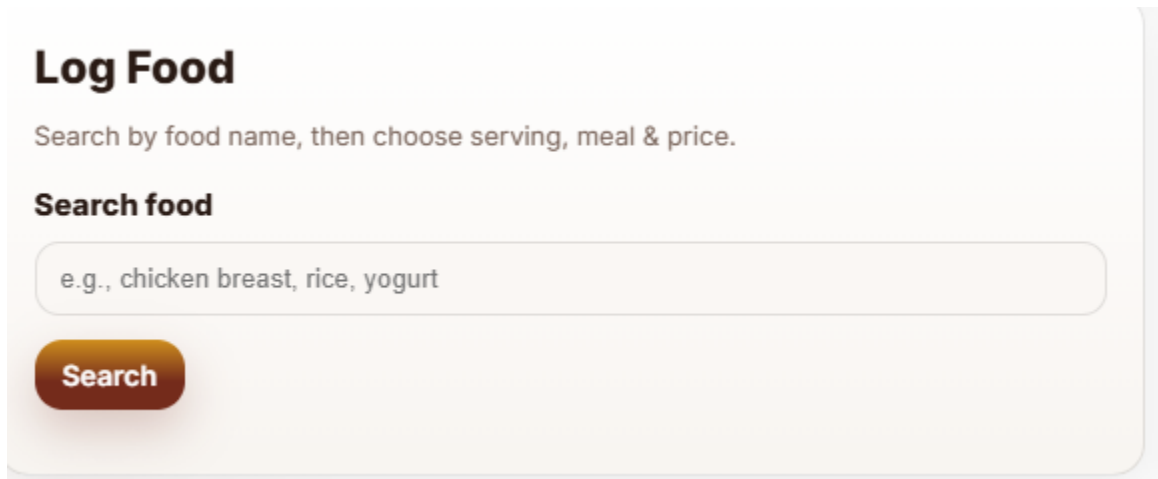
Github pull request link for:



Requirement 1

1. Requirement(User story from Repo): **As a student, I want to save frequently eaten foods so that I can easily access them.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/5>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/21>
4. Implemented by: Rita Bolanos
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.2.A

7. Visual evidence:



The screenshot shows a web interface titled "Log Food". Below the title is a subtitle: "Search by food name, then choose serving, meal & price." Underneath is a section labeled "Search food" which contains a text input field with the placeholder text "e.g., chicken breast, rice, yogurt". Below the input field is a prominent orange "Search" button.

Requirement 2

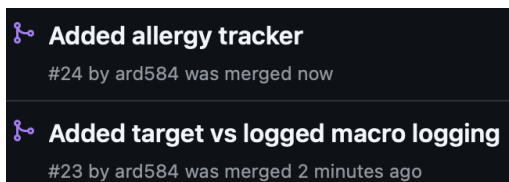
1. Requirement(User story from Repo): **As a budget conscious student, I want to see the cost per food item so that I can stay within my budget.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/6>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/22>
4. Implemented by: Rita Bolanos
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.2.B

7. Visual evidence:

The screenshot displays a budget tracking application interface. At the top, there are three summary sections: 'Month-to-date spend' showing '\$0.00' with a note 'From groceries, meals, etc.', 'Avg per day this month' showing '\$0.00 / day', and 'Today's budget status' showing '\$0.00 spent today' and '\$0.00 left'. Below these is a section titled 'Add a purchase' with the instruction 'Log what you just bought.' This section contains a form with the following fields: 'Date' (placeholder 'mm/dd/yyyy'), 'Item' (placeholder 'e.g. Chicken Breast Bowl'), 'Qty' (placeholder '1'), 'Unit cost (USD)' (placeholder '5.00'), 'Store' (placeholder 'e.g. Walmart / Campus Dining / Chipotle'), and 'Category' (placeholder 'e.g. Protein / Meal / Snack'). A brown 'Add to log' button is positioned below the form. At the bottom, a box labeled 'Estimated spend this month' shows '\$0.00'.

2.3 Annaliese Dedmore

Github pull request link for:



Requirement 1

1. Requirement(User story from Repo): **As a lazy student, I want to be able to track my total targeted macros vs. logged macros.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/7>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/23>
4. Implemented by: Annaliese Dedmore
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/main/tests/Unittest_2.3.A

7. Visual evidence:

Daily Targets
Find your daily macro and calorie intake

Calories: — kcal
Protein: — g
Carbs: — g
Fat: — g

Requirement 2

1. Requirement(User story from Repo): **As a health conscious student, I want to be able to track all my allergies to make sure I remember what to avoid.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/8>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/24>
4. Implemented by: Annaliese Dedmore
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.3.B
7. Visual evidence:

Active allergies

0

Why this matters

When you add food from the campus menu or USDA search, we'll scan for common allergens (milk, soy, nuts, etc). If we detect a match, we'll flag it.

Add an allergy / intolerance
e.g. milk, peanuts, gluten, shellfish, soy, egg...

Allergen / ingredient
e.g. milk, peanuts

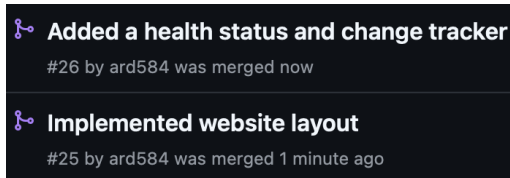
Add to list

Your list
Tap to remove something if you no longer want warnings for it.

Recent conflict warnings
We'll look at what you logged in Calories and Campus and highlight risky items.
We'll scan only the last ~7 diary entries to keep this lightweight.

2.4 Isidro Marquez

Github pull request link for:



Requirement 1

1. Requirement(User story from Repo): **As a student committed to improving my personal health, I want to be able to view all of my health details and goals all in one place.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/9>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/25>
4. Implemented by: Isidro Marquez
5. Approved by: Logan Bankert
6. Automated Tests: N/A
7. Visual evidence:

Your Goals

Current weight Target weight Daily budget Spend today

— — \$— \$—

Track weight targets and budget targets over time.

Add / Update Goal

Goal type **Target value**

Weight e.g. 75

Unit / currency **Deadline**

kg or USD mm / dd / yyyy

Save Goal

All Goals

Newest first

Type	Target	Unit	By
------	--------	------	----

Requirement 2

1. Requirement(User story from Repo): **As a newly-independent student, I want to be able to track all of my health changes as I adapt to college life.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/10>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/26>
4. Implemented by: Isidro Marquez
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.4.B

7. Visual evidence:

The image displays three screenshots of the CalPal application interface. The top screenshot shows a user profile section with a placeholder for a name and a unit selector (kg, cm). To the right, there are input fields for daily targets for Calories (kcal), Protein (g), Carbs (g), and Fat (g), with a 'Recalculate' button. The middle screenshot shows the 'Profile Settings' section, including a 'Display name' field, an 'Avatar URL' field, and a 'Units' dropdown menu. A 'Save changes' button is at the bottom. The bottom screenshot shows the 'Your Daily Targets' section, which displays the current targets for Calories, Protein, Carbs, and Fat, along with a button to 'Open Macros Calculator'.

2.5 Nile Ham

Github pull request link for:

[Added a goal versus eating habit tracker](#)

#28 by ard584 was merged now

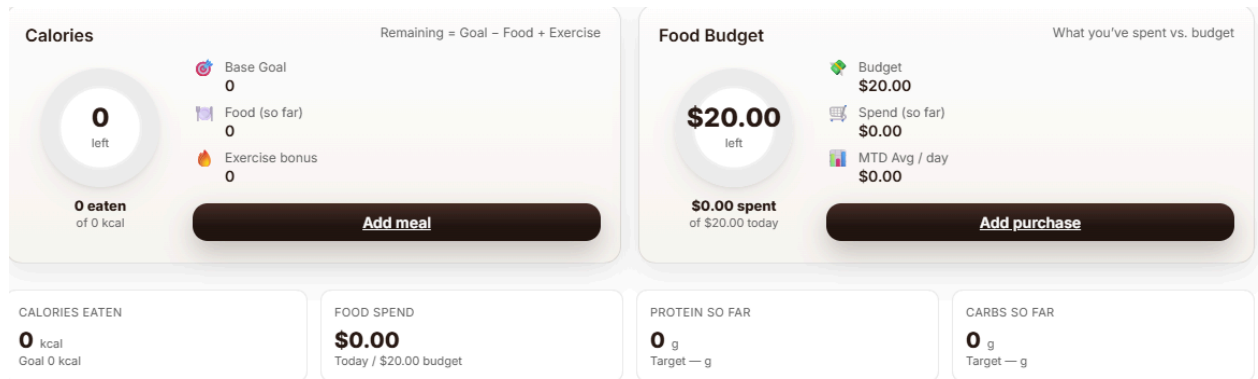
[Added money/cost tracker](#)

#27 by ard584 was merged 1 minute ago

Requirement 1

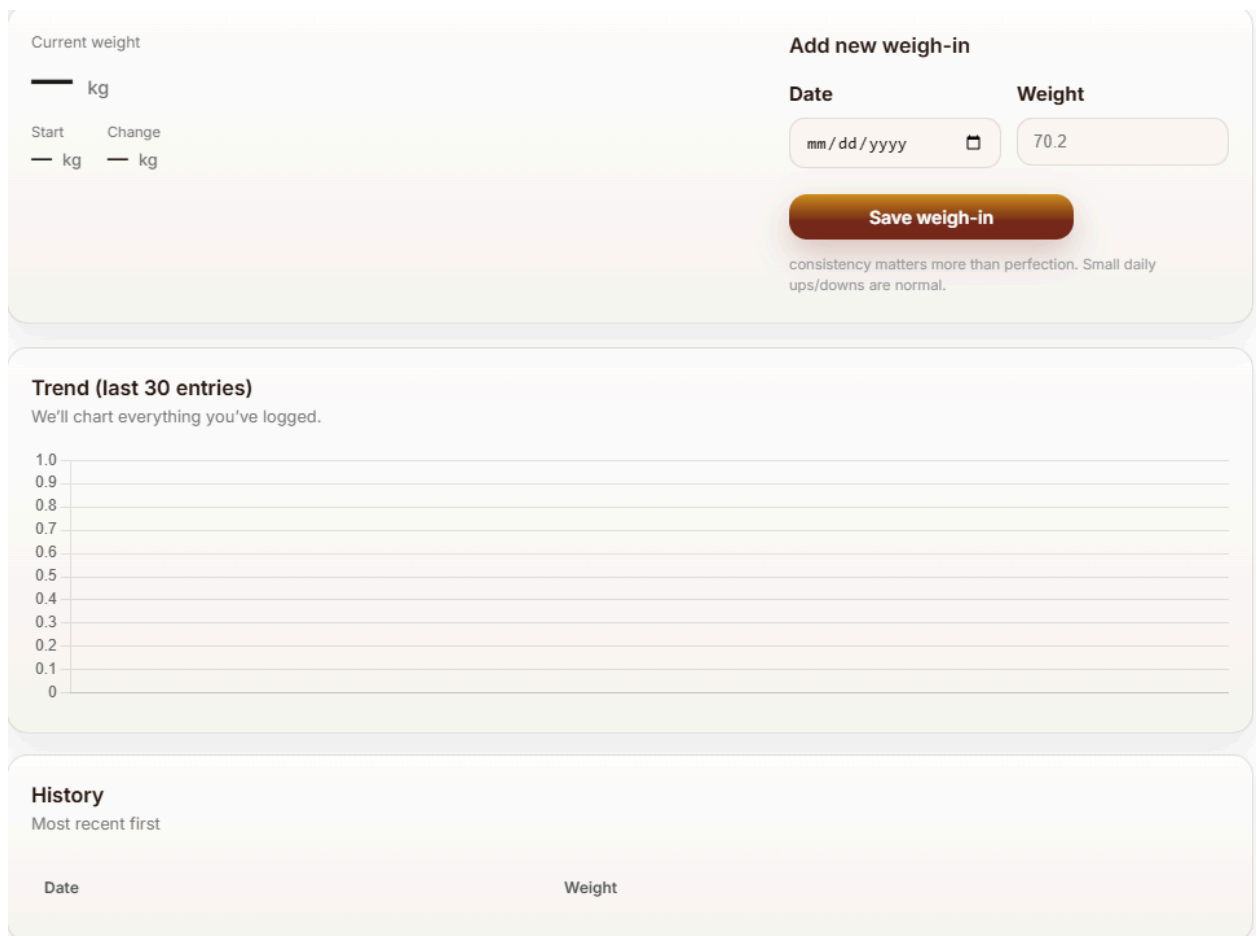
1. Requirement(User story from Repo): **As someone who enjoys eating healthy and saving money, I want to be able to track what I eat and spend at the same time.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/11>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/27>
4. Implemented by: Nile Ham
5. Approved by: Logan Bankert
6. Automated Tests: N/A

7. Visual evidence:



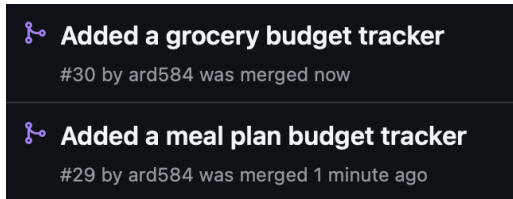
Requirement 2

1. Requirement (User story from Repo): **As someone who frequently does intense workouts, I want to be able to track what I eat throughout the year depending on what my goals are.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/12>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/28>
4. Implemented by: Nile Ham
5. Approved by: Logan Bankert
6. Automated Tests: https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.5.B
7. Visual evidence:



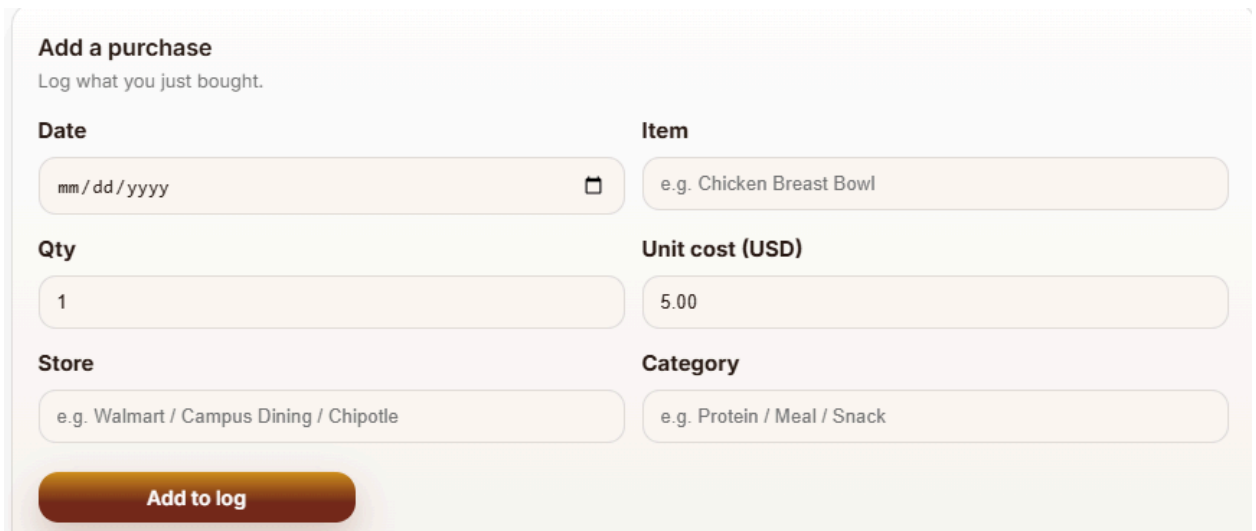
2.6 Luke Flaker

Github pull request link for:



Requirement 1

1. Requirement(User story from Repo): **As a freshman college student with a limited dining plan, I want to be able to juggle my spending on other meals alongside using my dining plan to ensure I have enough until the semester ends.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/13>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/29>
4. Implemented by: Luke Flaker
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.6.A
7. Visual evidence:



Add a purchase
Log what you just bought.

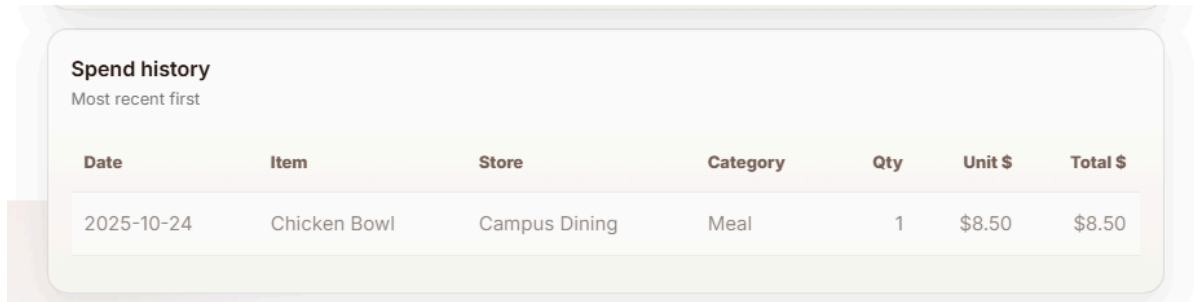
Date	Item
<input type="text" value="mm / dd / yyyy"/>	<input type="text" value="e.g. Chicken Breast Bowl"/>
Qty	Unit cost (USD)
<input type="text" value="1"/>	<input type="text" value="5.00"/>
Store	Category
<input type="text" value="e.g. Walmart / Campus Dining / Chipotle"/>	<input type="text" value="e.g. Protein / Meal / Snack"/>

Add to log

Requirement 2

1. Requirement(User story from Repo): **As a busy college student, I want to be able to keep track of what I spend at the grocery so as to not go over my weekly budget.**
2. Issue: <https://github.com/loganb7869/CS386-Project-Repo/issues/14>
3. Pull request: <https://github.com/loganb7869/CS386-Project-Repo/pull/30>

4. Implemented by: Luke Flaker
5. Approved by: Logan Bankert
6. Automated Tests:
https://github.com/loganb7869/CS386-Project-Repo/blob/ard584-unittests/Unittest_2.6.B
7. Visual evidence:



The screenshot shows a web interface with a 'Spend history' section. Below the title is the text 'Most recent first'. A table follows with the following data:

Date	Item	Store	Category	Qty	Unit \$	Total \$
2025-10-24	Chicken Bowl	Campus Dining	Meal	1	\$8.50	\$8.50

3. Automated Testing

1. Testing Framework: Python unittest
2. Test Location: <https://github.com/loganb7869/CS386-Project-Repo/tree/main/tests>
3. Test Example:
 - a. https://github.com/loganb7869/CS386-Project-Repo/blob/main/tests/Unittest_2.3.A
 - b. [https://github.com/loganb7869/CS386-Project-Repo/blob/main/tests/2.3.A Source](https://github.com/loganb7869/CS386-Project-Repo/blob/main/tests/2.3.A_Source)
 - c. The test validates that the macro values from the user's input are added correctly and the remaining macros properly reflect the changes made against the user's goal.
4. Test Results:

```
anna@ihd26 Desktop % python3 -m unittest macro_test.py
.
-----
Ran 1 test in 0.000s

OK
```

4. Technology Stack

HTML

Description: The standard language for most websites that gives the basic structure for content on a web page.

We used HTML because many of our team members are either already experienced with HTML or are currently learning it. HTML is also simply the easiest to use language for website design since it is the most widely used.

CSS

Description: The standard styling language for most websites that gives more in depth structuring and stylization to a website.

We used CSS because many of our team members are either already experienced with CSS or are currently learning it. CSS is also generally the standard for website styling which makes it the most compatible with our website.

JavaScript

Description: The standard functionality implementation language for most websites that gives a website its core coded functions.

We used JavaScript because many of our team members are either already experienced with JavaScript or are currently learning it. JavaScript is also generally the standard for website functionality implementation which made it the natural choice for our website.

5. Learning Strategy

Prior to the project all of our team members have taken a variety of coding courses that covered C-programming, Python, HTML, CSS and Java. Additionally some of the team members have experience in digital design and illustration which enabled them to create unique digital assets for our project. On top of our current knowledge base, team members have invested time into learning languages and development strategies both in and out of the classroom such as unit testing, creating technical documentation and diagrams, etc.

6. Deployment

Live System Link: <https://calpalwebsite.z1.web.core.windows.net/calpal/index.html>

Deployment Method: Microsoft Azure's Static website option offered the possibility to host our website. After providing the necessary files for the website and linking everything together, Azure provided the team with a static website link that can be accessed by anyone that has the link.

Platform Justification: The team has previously used Microsoft Azure to deploy our own personal websites before. After looking at other sites for hosting, we decided to use Microsoft Azure as it provides a free service to deploy our website and manage the files as they are updated.

7. Licensing

The license we chose for our project was the unlicense

We chose this license because as a team we feel that contributing to the public domain is important. Especially since this project is supposed to be a simple learning tool for CS 386 and not a fully fledged commercial product, allowing others to use the code to build their own experimental projects is a benefit in our eyes.

8. Repository Documentation

Prepare your repository for future contributors.

Required Files:

- **README.md** (primary focus)
 - Project description and setup instructions
 - Current version number following Semantic VersioningLinks to an external site.
 - Installation and usage instructions
 - Contribution guidelines
 - Git tag matching the version number

Link:

<https://github.com/loganb7869/CS386-Project-Repo/blob/main/README.md>

- **LICENSE**


Link:

<https://github.com/loganb7869/CS386-Project-Repo/blob/main/CalPal%20License.md>

9. User Interface Design

When it came to designing the UI the first step we did after discussing the basic functionality of the website was to draw several mockups of screens that could be present in the final website. These mockups were done live in a meeting to allow for immediate feedback from other team members and were added to the repo to serve as a guide for designing the final website. After the mockup phase HTML and CSS code were written to mimic the mockups and then creative liberties were taken to make the UI look cleaner and be easier to implement.

Visual Examples: **(Images created by Rita Bolanos)**

MENU | NAME OF WEBSITE HERE 

Macro

Macro

Current Goals

Cost Calc

Allergy List

Weight Tracker

Calories Tracker

Carbs	Protein	Fiber	Fats
Curr level:n Curr Sugar:n Curr starch:n Goal:n	Curr level:n DV level:n% Goal:n	Curr level:n DV level:n% Goal:n	Curr level:n Sat level:n Trans level:n Goal:n
Recommended Foods:	Recommended Foods:	Recommended Foods:	Recommended Foods:
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

MENU | NAME OF WEBSITE HERE 

Macro

Goals

Current Goals


Cost Calc


Allergy List


Weight Tracker





Calories Tracker

- | | |
|--|--|
| <input type="checkbox"/> Toggleable Goal | <input type="checkbox"/> Toggleable Goal |
| <input type="checkbox"/> Toggleable Goal | <input type="checkbox"/> Toggleable Goal |
| <input type="checkbox"/> Toggleable Goal | <input type="checkbox"/> Toggleable Goal |
| <input type="checkbox"/> Toggleable Goal | <input type="checkbox"/> Toggleable Goal |

Menu	NAME OF Website HERE 																																																							
Macro	Cost																																																							
Current Goals	<table border="1"> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> <tr><td>Item</td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td>Cost</td></tr> </table>		Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost	Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost	Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost	Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost	Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost	Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Item	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Cost																																																
Cost Calc																																																								
Allergy List																																																								
Weight Tracker																																																								
Calories Tracker																																																								
		Total:																																																						

Menu	NAME OF Website HERE 									
Macro	Allergies									
Current Goals	<table border="1"> <tr> <td><input type="checkbox"/> Toggleable Allergy</td> <td><input type="checkbox"/> Toggleable Allergy</td> </tr> <tr> <td><input type="checkbox"/> Toggleable Allergy</td> <td><input type="checkbox"/> Toggleable Allergy</td> </tr> <tr> <td><input type="checkbox"/> Toggleable Allergy</td> <td><input type="checkbox"/> Toggleable Allergy</td> </tr> <tr> <td><input type="checkbox"/> Toggleable Allergy</td> <td><input type="checkbox"/> Other <input type="text" value="Text box"/></td> </tr> </table>		<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Other <input type="text" value="Text box"/>
<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy									
<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy									
<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Toggleable Allergy									
<input type="checkbox"/> Toggleable Allergy	<input type="checkbox"/> Other <input type="text" value="Text box"/>									
Cost Calc										
Allergy List										
Weight Tracker										
Calories Tracker										

menu	NAME OF website HERE 															
Macro	<div>Weight Tracker</div> <table border="1"> <tr> <td>Summary</td> <td> Update Height <input type="text"/> Update weight <input type="text"/> Update activity <input type="text"/> </td> </tr> <tr> <td>Weight up/down</td> <td></td> </tr> <tr> <td>Weekly goal hit $\frac{n}{n}$</td> <td></td> </tr> <tr> <td>Last recorded:</td> <td></td> </tr> <tr> <td>Height</td> <td></td> </tr> <tr> <td>Weight</td> <td></td> </tr> <tr> <td>activity level</td> <td></td> </tr> </table>		Summary	Update Height <input type="text"/> Update weight <input type="text"/> Update activity <input type="text"/>	Weight up/down		Weekly goal hit $\frac{n}{n}$		Last recorded:		Height		Weight		activity level	
Summary			Update Height <input type="text"/> Update weight <input type="text"/> Update activity <input type="text"/>													
Weight up/down																
Weekly goal hit $\frac{n}{n}$																
Last recorded:																
Height																
Weight																
activity level																
Current Goals																
Cost Calc																
Allergy List																
Weight Tracker																
Calories Tracker																

menu	NAME OF website HERE 																														
Macro	<div>Calories Tracker</div> <table border="1"> <tr> <th colspan="2">Summary</th> <th colspan="2">Recommendations</th> </tr> <tr> <td>Total: n</td> <td>Goal amt: $\frac{n}{n}$</td> <td><u>Food</u></td> <td><u>Exercise</u></td> </tr> <tr> <td>Breakdown:</td> <td>Goal burned: $\frac{n}{n}$</td> <td>Food item</td> <td>Workout n hrs</td> </tr> <tr> <td>Total Fat: n</td> <td rowspan="4"> (Other Info Here)  </td> <td>Food item</td> <td>Workout n hrs</td> </tr> <tr> <td>Cholesterol:</td> <td>Food item</td> <td>Workout n hrs</td> </tr> <tr> <td>Total Carbs:</td> <td>Food item</td> <td>Workout n hrs</td> </tr> <tr> <td>Protein:</td> <td>Food item</td> <td>Workout n hrs</td> </tr> <tr> <td>Other:</td> <td></td> <td></td> <td></td> </tr> </table>		Summary		Recommendations		Total: n	Goal amt: $\frac{n}{n}$	<u>Food</u>	<u>Exercise</u>	Breakdown:	Goal burned: $\frac{n}{n}$	Food item	Workout n hrs	Total Fat: n	(Other Info Here) 	Food item	Workout n hrs	Cholesterol:	Food item	Workout n hrs	Total Carbs:	Food item	Workout n hrs	Protein:	Food item	Workout n hrs	Other:			
Summary			Recommendations																												
Total: n			Goal amt: $\frac{n}{n}$	<u>Food</u>	<u>Exercise</u>																										
Breakdown:			Goal burned: $\frac{n}{n}$	Food item	Workout n hrs																										
Total Fat: n			(Other Info Here) 	Food item	Workout n hrs																										
Cholesterol:				Food item	Workout n hrs																										
Total Carbs:	Food item	Workout n hrs																													
Protein:	Food item	Workout n hrs																													
Other:																															
Current Goals																															
Cost Calc																															
Allergy List																															
Weight Tracker																															
Calories Tracker																															

Usability Considerations: For the website, we wanted the user to be able to switch seamlessly between the different headers which leads them to the pages that hold each unique program/function. Each section of the website contains prompts that the user can input depending on their goals/budget/meals. Once the user actually inputs this information into the website, these values will then be displayed back to the user through the user of appealing

graphics, and the data would then be stored so that when the user comes back to the website later, they are able to view the information that they added. Also, the user can change their goals and edit their budget whenever they want to provide a more interactive user experience. The user interface is still under development as the project reaches the halfway point but a general MVP has been created for the anticipated final design.

10. Retrospective Analysis

Key Learnings: The team developed a more improved methodology to performing weekly meetings by assigning alternating roles to members every week, increased production efficiency thanks to our collective understandings of software development, and even improved our understandings of how teams operate under pressure in the real world to meet deadlines on time. Alongside this, the team has become more comfortable with using a code-hosting tool such as GitHub to make constant changes to our program by proposing changes to be made, which have to be approved by one or more other group members.

Challenges Overcome: During the course of the drafting process of the project, the group had little to no experience with working in a team environment. However, as time went on the team became more comfortable with working with each other and even understood how to operate effectively when under a future deadline. Some other notable problems that the team encountered include struggling with automated tests and the implementation of more accurate functions inside the website itself.

Future Improvements: In the future, the team will further our understanding of how to improve the functionality of our website by ultimately improving the accuracy of our functions which calculate key information about the user and contribute more to the repository of our code to highlight any noticeable changes needing to be made for the final product.

11. System Demonstration

Video Link(2-5 min):

https://drive.google.com/file/d/1YPH7AKvaiKY7_wToBbHCh0hx8-glehMm/view?usp=sharing