

Calorie Tracker App

Concept and Inspiration

Creating an app that resonated with our collective passion was the most challenging aspect of this project. Through numerous meetings and brainstorming sessions, we generated a range of ideas, ranging from addressing recycling issues to developing social apps exclusively for UMD students, homework management tools, and task manager applications. However, it was during this long process that we stumbled upon an idea that we were all collectively interested in. A team member had mentioned they were heading to the gym after a meeting and our app discussion quickly shifted to a gym related concept. Coincidentally, we had all recently started going to the gym and we noticed that we were neglecting to monitor our dietary intake and lacked the knowledge to tailor our diets to align with our gym goals. This common concern, which we could all identify with, proved seemed like something we could explore further. Although the app idea was initially inspired by diets being very ambitious, we decided to plan out additional features to address various fitness-related issues. Within a matter of minutes we arrived at the name for our first version of the app, "Mobile Fitness Tracker," an app that would allow users to track their fitness goals and monitor their progress anytime, anywhere. Its core functionalities were user profile creation, goal setting, workout logging, and meal tracking. The app would also provide personalized recommendations for workouts and meals based on individual fitness objectives and preferences. We also quickly set out a stretch goal to leverage the device's motion sensors and GPS capabilities to track users' activities more accurately, enhancing their overall experience.

However, as we delved deeper into the app's design and outlined its features, our goals underwent a significant shift. It felt like we were trying to build the perfect app and some of our goals were unrealistic for the scope of the project. We decided to narrow our focus to the aspect we deemed most crucial: calorie tracking. Going back to our original concept for the app, we decided to change the name to the "Calorie Tracker App". Our new app aimed to enable users to monitor their daily caloric intake and log their meals with ease. The app would not only focus on calorie tracking but also include an option for notification reminders to prompt users to eat at regular intervals. Along with the ability to log meals, the app provided the functionality to analyze past days' calorie progress, enabling users to go back and see what patterns worked best for them and produce their desired results. Throughout the development process, we had to make several changes and challenges but being flexible and open-minded allowed us to navigate through the process. We are happy with the app's UI and functionality and truly believe that the app represents our collective desire to live a more healthy lifestyle and it addresses a common issue faced by a lot of gym enthusiasts and even individuals who simply want to track their calories in order to get to a desired weight. By focusing on calorie tracking and meal logging, the app allows users to make informed dietary choices and progress toward their weight/body goals. The app's evolution from the initial "Mobile Fitness Tracker" concept to the final "Calorie Tracker App" also taught us a great lesson on app development and being able to adapt in order to create a practical solution. With its user-friendly interface and comprehensive features, the app is a creation we are all proud of and truly believe can be helpful in assisting individuals in making positive changes to their lifestyles.

Goals

Our goal for the app was to create a space for users to keep track of their daily calorie intake, allowing users to analyze their statistics compared to a goal, keep track of daily habits and meals and compare their progress from day to day. Specifically, users would be able to store different types of meals in the app that they may consume throughout the day. Here we discuss some of the goals we had set.

Set-Up Calorie Calculator

Our first goal was to set up the calorie calculator. We wanted users to have some way of setting a goal for how many calories they wanted to intake daily.

Create a Meal Book

We want users to be able to create their own meal book where they can add meals they regularly consume. This way when they track their calories they can add a meal from the predefined meal book. These meals would store information such as calories, protein, carbohydrate, and fat levels for the user to keep track of and reference. Users should also be able to add meal descriptions to include any other notes they want to record about their meal.

Logging Meals

Another goal is for users to be able to log the meals that they are consuming on a daily basis. We want users to be able to reference prior days of logging and retrieve old calorie information. We also want users to enter new information and log meals for any day.

Show Daily Progress

We want users to be able to see their daily progress in the logs. This will help inform them about how many more calories they need to eat.

Enable Notifications

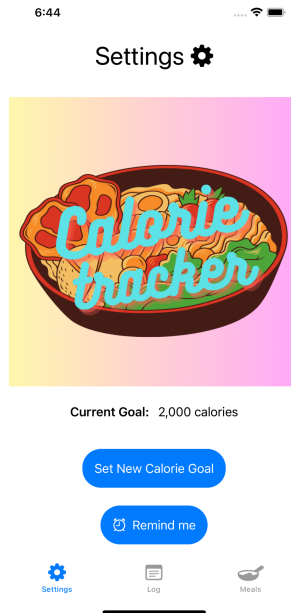
For our app, we also wanted to incorporate more interactive features so that the user is engaged with our app, by allowing users to receive notifications from the app to promote daily use. We want users to be able to enable notifications which will remind them throughout the day to log their meals and keep track of their consumption.

Walk-through

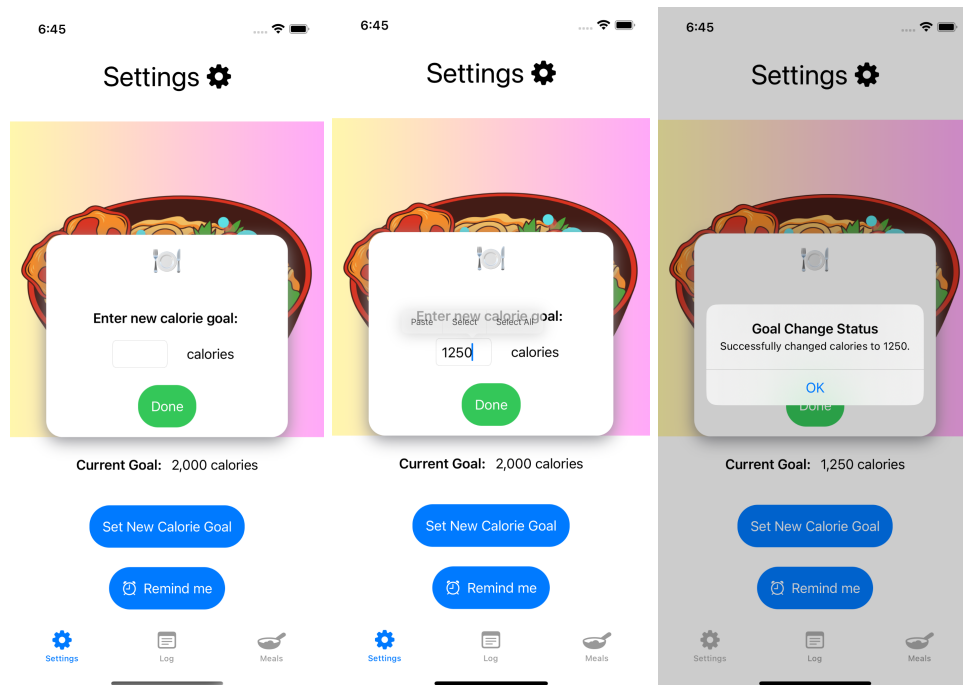
Our App contains 3 basic views, SettingsView, LogView, and MealView.

Settings View

When the user first opens the app, they come across the settings page. This page has our app logo displayed in order to show our brand. This helps add an identity to our app.

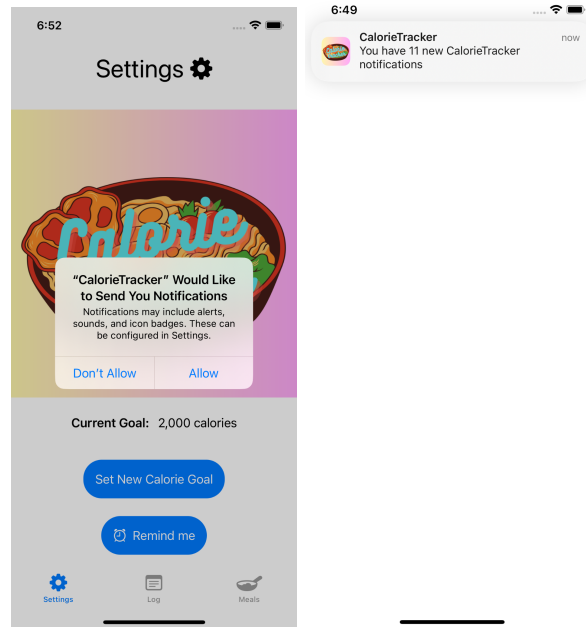


On this page, the user can view their current calorie goal. The app has a default calorie goal of 2,000 calories. The goal is displayed right under our logo. The user can also set their own preferred calorie goal they wish to meet by pressing on the button “Set New Calorie Goal”



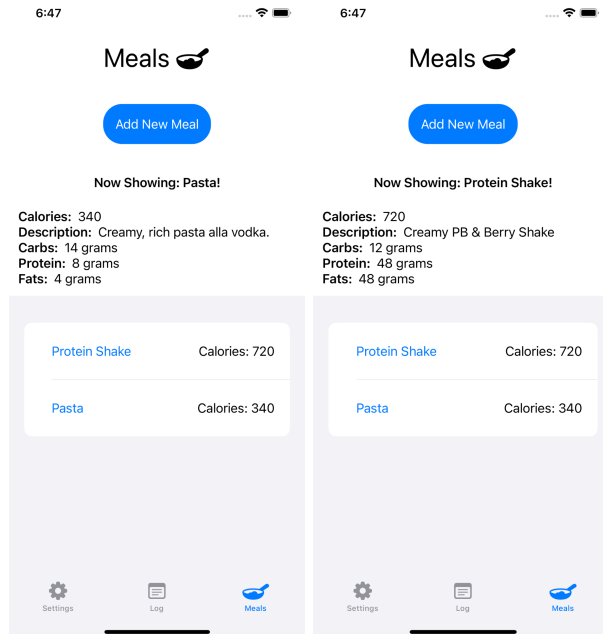
This will prompt the user to enter in a new calorie goal. When they finish they can press done and an alert window will appear indicating whether or not the change was successful. If the user inputs incorrect fields such as string or no value in the text box, the alert window that appears will display how the change has failed. Upon updating the calorie goal, the current goal will then be updated in the settings page.

In the settings page, a user can enable notifications for their device by clicking the “Remind me” button. This meets our goal of setting up reminders for our users to encourage them to frequently use the app to log their calories. Upon clicking this button an alert window will pop up, asking the user for permission to send notifications. Once notifications are enabled periodic reminders will pop up as the user uses their phone.

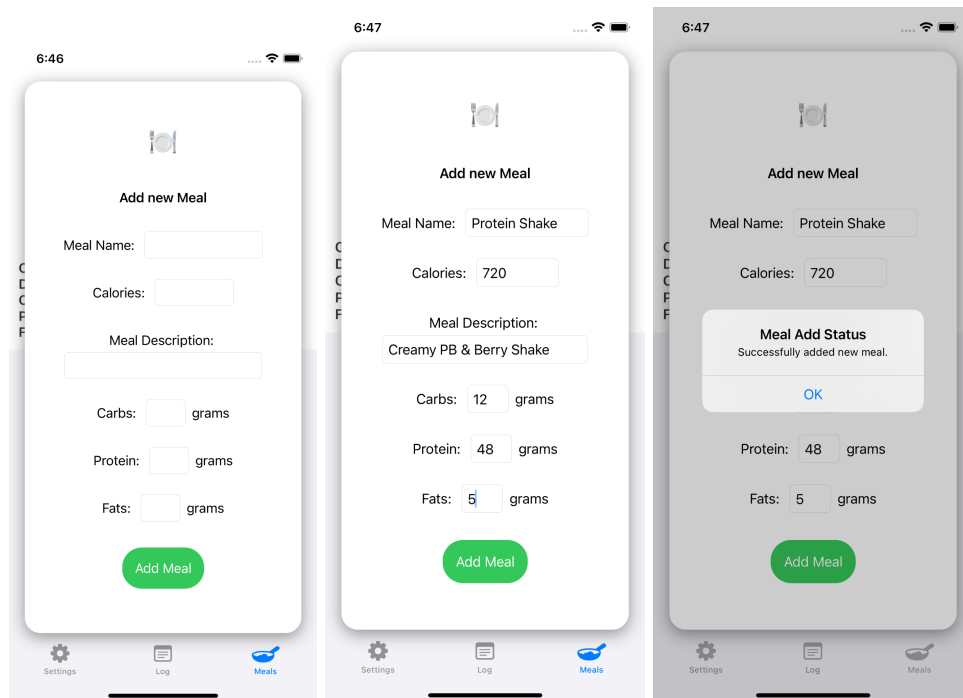


Meals View

In order to access the meals on file, the user can access the meals tab. In this view a user can view all meals they have on file. It opens to a default meal of Pasta. The user can see this meal's description and scroll through any meals they have previously added to their meal book. The user can click on the meal name to update the view to show the details of the meal.



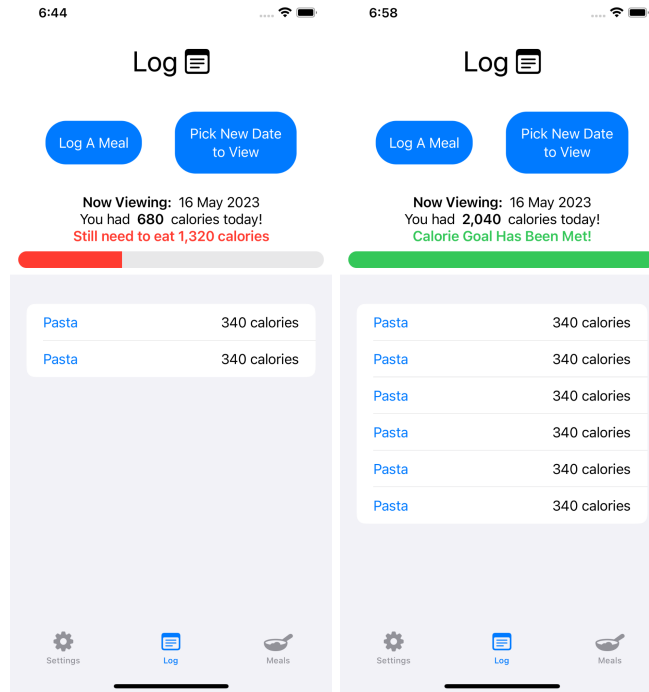
The user can also add new meals to their meal book by clicking on the “Add New Meal” button. Upon clicking this, a window pops up and shows the user various fields about the meal to fill out. The user can enter the meal name, calories, description, and any other macros they want to keep track of.



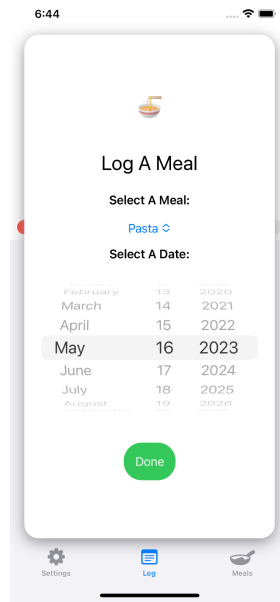
Once the user fills out all the fields they can press the “Add Meal” button. If the meal was successfully added to the meal book, an alert pops up indicating the success. If the user incorrectly filled out fields (put strings where numbers are required or didn't fill out all fields) the alert will indicate failure. If the meal added was a success the meal view will update to show the newly added meal.

Log View

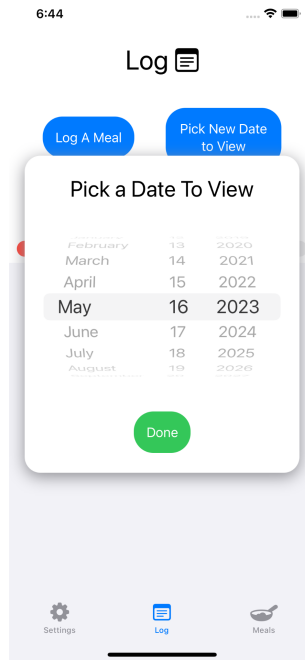
In the log view tab, the user can log a new meal a desired day and see their calorie intake process for any day. There is a sliding progress bar which shows a day's current calorie intake. And updates when meals are added.



The user can press the “Log A Meal” button in order to log a new meal eaten for any desired day. Upon pressing the button a pop up appears asking the user to select a meal from the meal book and pick a desired date. By default, the date is set to show the current date being viewed. Once the user presses done, the meal is stored.



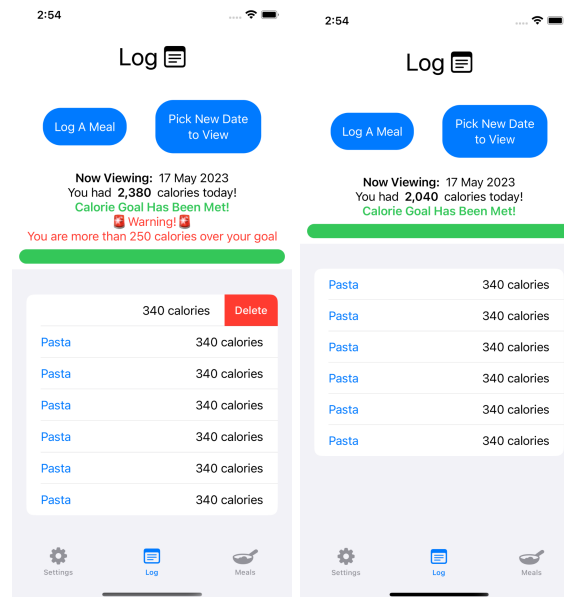
The user can also press the button, “Pick New Date to View” in order to see the meals eaten on any specific day. Upon pressing this button, a pop up appears and asks the user to select a date. When the user presses “Done” the selected date is then used to update the view to show the meals eaten on that day.



The user can also click on any of the meals logged in order to see the meal description. When they are done viewing the description they can press done to make the view go away.



Additionally the user can delete meals that they have accidentally added. To do this they need to swipe left on a meal and a red delete button will appear. Upon clicking, the meal is deleted.



If the user exceeds daily calorie intake by more than 250 calories, a warning message appears on the log page to indicate this information to the user.

Development process

Changes Made From Initial Goals

Prior to writing any code we decided that it would be a great idea to draw out the app UI and decide how we wanted to design the model. We began drawing out the app design for what was supposed to be our app, mobile fitness tracker. As we began drawing and trying to figure out where each component would go we became overwhelmed and soon realized that we were overshooting and trying to implement too many different things into the app which wasn't ideal given the time constraint of the course.

It was here we began to scratch out ideas such as individual user accounts, authentication, logging workouts and tracking runs. Given the time constraint we decided that it would be smarter to begin with focusing strictly on calorie tracking and expand further depending on time available. Over the course of our project, many of our minimal goals evolved because we decided to change the focus of the app to be strictly calorie tracking and to not include fitness.

We did not create our own calorie calculator because we do not have proven algorithms which can recommend to users calories based on their fitness goals. We are not dieticians and we lack the knowledge for properly recommending dietary needs. Instead, we decided to let the users pick their own calorie goals since everyone has their own preferences. This is better than incorrectly recommending users with diet choices.

Because of this, we decided to scrap logging and keep track of workouts. This would be a great future addition, however for an initial model, we wanted to focus on only calories and diet. Workouts could be later used to keep track of overall fitness. Since we lacked the knowledge of how workouts could impact calories to consume and we changed the focus of the project, we decided this was not essential.

We also decided that implementing a camera feature is not essential to the app. It seemed that having images of the meals or workouts has no additional value to the user or their calorie goals. Images of meals cannot be used to determine nutritional status nor will they benefit users when it comes to tracking calories. For this reason, we decided it was not important to implement the camera feature since it wasn't necessary.

Instead we opted to create reminders. As people carry on throughout the day and consume meals, it is easy to forget to log your calories. To combat this, we decided focusing on getting notifications for our users was a better alternative. This way users would be encouraged to use the app more. This feature adds more value to a user using the app.

We used goodnotes on an ipad device to draw up the UI and then used notion to come up with the pseudo code for our model and delegate tasks to each team member.

Implementation and Challenges

CalorieTracker was used as an observable object and represents the instance of our app, storing the calorie goal, a mealbook, logged calories, and logged meals. We decided to disregard individual user accounts and authentication from our app because it is not necessary for the utility of what we are creating. Each instance of our app on the users phones will reflect and store their own data. When beginning work on our project we had to decide on the structure for our model, how we would store and organize the different aspects of data from our users. Our model class creates a struct called Meal, which is used to store Meal objects which include information such as calories, fat, protein, carbohydrates, all in grams, and a description of the meal. The model stores a hashmap with string key values which reference the name of a meal and a value corresponding to a Meal object created by the user. The user can log their meals by day and reference them to output daily stats on their intake, including comparing their total calorie intake to a daily goal that the user can set themselves.

We implemented the different views for our app which include UI for the different functional parts of our app. The views consist of pages for adding a meal, viewing the collective meal book, logging meals eaten from the meal book, logging and displaying the user's calorie intake goal, calculating their calorie intake, as well as displaying them based on date. Our UI hierarchy consists of a ContentView page which references three main views, Settings, Log, and Meal. The Settings view includes functionality for viewing by accessing the EnvironmentObject variable tracker, an instance of CalorieTracker that is used and updated by our different views. In order to change the calorie goal, a new ChangeCalorieGoal view is opened, allowing the user to update the calorie goal in the tracker. Using the built in Swift object, UNUserNotificationCenter, we were able to create a feature for users to get notifications from the app, repeated over a set time interval, to remind them to engage and log their meals. The MealView references two other views, a view for displaying the meal book as well as a new view for adding a meal, both of which reference and update our tracker as well. The LogView displays logged meals and calorie intake, much of this page uses functions created for our CalorieTracker including retrieving our calorie goal, displaying calorie intake for a specific day by referencing a given date and searching through our logged meals.

We aimed to expand the usability of our app by including a feature to allow users to directly open SMS and send their statistics via text, through the use of Apple's MessageUI. However, our project structure was further developed and integrating the protocol and delegation for the Swift library into our purely View based structure proved difficult, along with XCode's restrictions on accessing messages through the phone simulator. In order to maintain user engagement outside of the app we decided to implement the Notification Center to send alerts.

Potential future directions

Some future changes we would like to add to our app is to include our own algorithm which can help recommend users how many calories they should be consuming based on their height, weight, activity levels, and desired physique. Doing this will help users who aren't as educated to be able to make healthy choices. Artificial Intelligence is a powerful tool which can be used to achieve this.

Another feature we would love to add is meal recommendations. If we integrate our app with Firebase we can create a pre-existing set of meals and their recipes and recommend meals to users based on their dietary needs or goals.

We can also allow users to create other goals for macros. Many fitness enthusiasts love to keep track of their daily protein intake so they can adequately build muscle. We can add a feature where a user can set a daily protein goal. They should also be able to view the progress for this goal in a page.

Another direction we can take is camera integration. We can add a camera feature so users can take pictures of their meals and keep them on file with their meals. We can use Firebase to store these images for the app.

Social media sharing is a unique feature. This way users can share their progress or meals with their friends in order to encourage healthy competition or showcase what they have done.

Another thing we can add to the app is integration with fitness tracking devices. People can log meals on apple watches. Apple watches can also collect motion data and keep track of user activity. This data can then be used to create a fitness page and update the users recommended calorie intake based on their activity. This way the user can maintain optimal physique. Users can use the GPS and motion sensors to track their activity.