

Created by: Bryan Loo Chun Wai and 4 others

Agile Software Projects

Final Report Assignment

(NutriTrack)

Table of Contents

Introduction and outcomes

Github Repository Link.....
Video Demo Link.....
Abstract.....
Outcomes.....
Introduction.....
Background.....

Research on health trends

Discussion of literature.....
Technical Considerations.....
Problem analysis and argumentation.....

Planning

Planning and research.....

Project Team & Stakeholders.....
Project Manager.....
UI/UX Designer.....
Backend Developer.....
Project End Users.....

Backend processes.....
Iteration and deployment plan.....

Prototyping and iteration.....
Login page.....
Home page.....
Social page.....
Feedback page.....
Search page.....
Article page.....

Design.....

Requirements definition.....

Functional requirements.....

Functional complexity.....	
Functionality.....	
Complexity.....	
Time consumption.....	
System development.....	

Development issues and problems

Evaluation.....	
Analysis.....	
Evidence of collaboration and teamwork.....	
Change Request Form.....	
Git Commit Log.....	
Team Communication History.....	

Testing

Login page.....	
Home page.....	
Feedback page.....	
Articles page.....	
Social page.....	
Search page.....	

Technical Challenge faced during development phase

Login page.....	
Home page.....	
Feedback page.....	
Articles page.....	
Social page.....	
Search page.....	

Methodologies Used for Project Development

Summary

Discussion of Future Work.....	
Conclusion and summary.....	
References.....	

Github Repository Link:

https://github.com/BryanLoooo/ASP_NutriTrack_Team89

Video Demo Link:

[NutriTrack Agile Software Projects Team 89](#)

Introduction and outcomes

Abstract

The purpose of the project is to create a comprehensive application, with features developed to address the rising health concerns in Singapore (and hopefully more countries in the future). The app aims to empower users to make informed lifestyle choices through a multifaceted approach to health management such as detailed nutritional tracking and community support to allow users to join interest groups to share their experiences and motivate each other.

The features of the application were carefully selected based on rigorous survey and testing, and developed following the Agile software development methodologies. This allows our application to make continuous improvements throughout the course of this project by adopting iterative development and customer involvement through multiple user surveys to know which features will benefit them the most.

Outcomes

The goals for the project ties in with the key features for our application, this includes and is not limited to developing a user-friendly interface that is not only easy to use but also comprehensive enough to allow users to be able to track their nutritional and calorie intake at a glance. Additionally, implementing persistent data storage and personalization functionality, via the ability to create an account, helps users save their data so that they can come back to the application anytime without losing their progress. The application also features a curated news section that not only keeps them informed with the latest health trends but also broadens their understanding of various health topics fostering a more comprehensive approach to wellness. Lastly, the dynamic social networking feature in our application, facilitates the creation of support groups to allow users to connect with like-minded individuals, providing them with a platform to share their experiences and motivate each other. We hope that this feature will not only combat the isolation associated while making lifestyle changes but also harness the power of peer support to boost user motivation and accountability. With the aforementioned features, we aim to create an application that separates us from preexisting health related applications on the market, while also being effective in empowering our users to lead healthier and better lives.

Introduction

The project aims to create a mobile application adopting the software development approach utilized in the Agile methodology. The team is divided into 2 front-end and 3 back-end developers, where the back-end team will focus on the application functionality such as curating and integrating the specific APIs to power the app features. The front-end team, on the other hand, will focus on creating a visually appealing, intuitive and responsive UI so that users can navigate and utilize the application with ease.

The premise of the report will highlight the team's journey in developing the application, which will include a critical analysis on the technologies that were used during development as well as a breakdown of the development lifecycle from conceptualization to its implementation phase. The report will also showcase how the team adopted an iterative design approach (as taught in the module), where we used results from user research and survey to find ways to improve the application's user interface and experience.

Background

NutriTrack. Your one stop shop for reliable daily caloric tracking and motivation. The application is designed to help improve the quality of life of Singaporeans. In today's digital landscape, such applications play a crucial role in promoting healthy lifestyles and also to help to foster community engagement. Through research on current health trends, has highlighted the need for such tools and their potential impact on users. Especially in today's fast paced living environment in Singapore, maintaining a healthy lifestyle is paramount. This has inspired us to create a caloric tracking application that enables users to stay motivated and avoid overeating habits. The lack of accessible and reliable tools for tracking food and calories intake and finding motivation further exacerbates the problems.

The primary users of this application include individuals looking to lose weight, manage specific health conditions or anyone who is interested in adopting a healthier lifestyle. Stakeholders such as dietitians, fitness trainers, healthcare providers can also use this application to support their clients.

Despite the availability of these calories tracking applications and platforms, several limitations and gaps remain. These include a lack of user engagement, comprehensive information and user interface. Thus, with this mobile application that we developed we want to be able to tackle these areas and provide an easy to use and high value application that will allow users to track calories on a daily basis and be significantly encouraged to live a better lifestyle and share experiences.

Research on health trends

Discussion of literature

Before diving into development, it's crucial to understand the specific needs and preferences of Singaporeans when it comes to calorie tracking.

Dietary Habits

1) Typical Diet

Diagram 1

Daily intakes of nutrients (total and per 4184 kJ (1000 kcal)) by fast-food consumption status, adjusted for age, gender, ethnicity, education and income level: Singapore residents (*n* 1627) aged 18–69 years, 2010 National Nutrition Survey

	Fast-food consumption status						
	Non-consumer (<i>n</i> 557)		Occasional consumer (<i>n</i> 720)		Regular consumer (<i>n</i> 350)		
	Mean	SE	Mean	SE	Mean	SE	P for trend
Energy (kJ)	10 234	39.62	11 175	34.52	11 941	67.20	<0.001
Energy (kcal)	2446	9.47	2671	8.25	2854	16.06	<0.001
Carbohydrate (g)	320	1.17	341	1.03	362	1.91	<0.001
Carbohydrate (g/4184 kJ)	133	0.18	129	0.11	128	0.17	<0.001
Protein (g)	93	0.38	102	0.33	109	0.63	<0.001
Protein (g/4184 kJ)	38	0.07	38	0.06	38	0.09	0.627
Total fat (g)	84	0.38	96	0.34	104	0.72	<0.001
Total fat (g/4184 kJ)	34	0.06	35	0.04	36	0.06	<0.001
SFA (g)	31	0.15	36	0.14	40	0.30	<0.001
SFA (g/4184 kJ)	13	0.03	13	0.02	14	0.03	<0.001
MUFA (g)	31	0.15	36	0.14	39	0.27	<0.001
MUFA (g/4184 kJ)	12	0.03	13	0.02	13	0.03	<0.001
PUFA (g)	15	0.07	17	0.05	19	0.12	<0.001
PUFA (g/4184 kJ)	6	0.02	6	0.01	6	0.02	<0.001
Trans fat (g)	0	0.00	0	0.00	1	0.00	<0.001
Trans fat (g/4184 kJ)	0	0.00	0	0.00	0	0.00	<0.001
Cholesterol (mg)	305	1.77	353	1.65	382	2.70	<0.001
Cholesterol (mg/4184 kJ)	122	0.51	130	0.37	132	0.50	<0.001
Dietary fibre (g)	23	0.09	24	0.07	25	0.12	<0.001

About

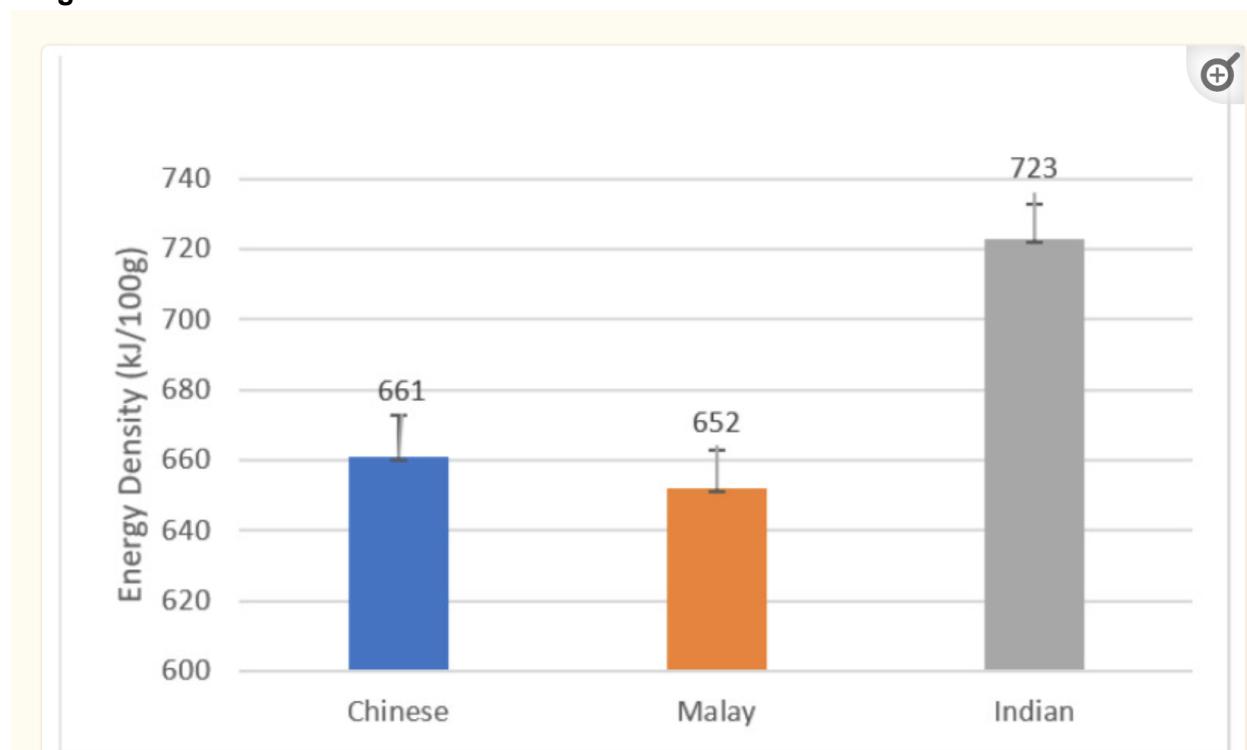
Singaporeans have a diverse diet influenced by the country's multicultural society, including Chinese, Malay, Indian, and Peranakan cuisines. Research shows a strong preference for rice-based dishes, noodles, and a variety of meats and seafood.

Literature Reference

In diagram 1, a survey done by National Library of Medicine revealed that 63% of adult Singapore residents consumed fast food monthly, with 20% eating it at least once a week. Fast-food consumption was more common among younger individuals, higher-income groups, and those with middle education levels. Frequent consumption was linked to higher intakes of energy, fat, and saturated fat, while lowering the intake of whole grains, fruits, and vegetables. This behavior also increased the risk of exceeding recommended daily allowances for energy and fats.

2) Nutritional Challenges

Diagram 2



About

High consumption of oil-rich foods, refined carbohydrates such as white rice, and sugar-laden drinks are common in the diet of a Singaporean. The prevalence of hawker centers and food courts offering affordable but calorie-dense meals poses a challenge for healthy eating.

Literature Reference

In diagram 2, the foods were listed from the highest to the lowest energy density for each ethnic group. The average energy density of Chinese, Malay, and Indian cuisines was 661, 652, and 723 kJ/100 g.

An article done by Channel News Asia has documented the nutritional content of typical hawker foods, indicating that they are often high in fat and calories. From the article, “HPB’s survey also found that Singapore residents are eating more, with their mean calorie intake going up from 2,360kcal in 2019 to 2,410kcal in 2022. Around 61% of residents exceeded what is recommended, which is estimated based on an individual’s age, gender, weight and level of physical activity. This was up from 55% in 2019.” This concludes a significant increase in calorie intake across all the ethnic groups, thus raising the issue of the lack of knowledge and discipline to practice healthy eating habits.

Lifestyle

1) Activity Levels

Diagram 3

Top 10 activities for teens (aged 13-19)

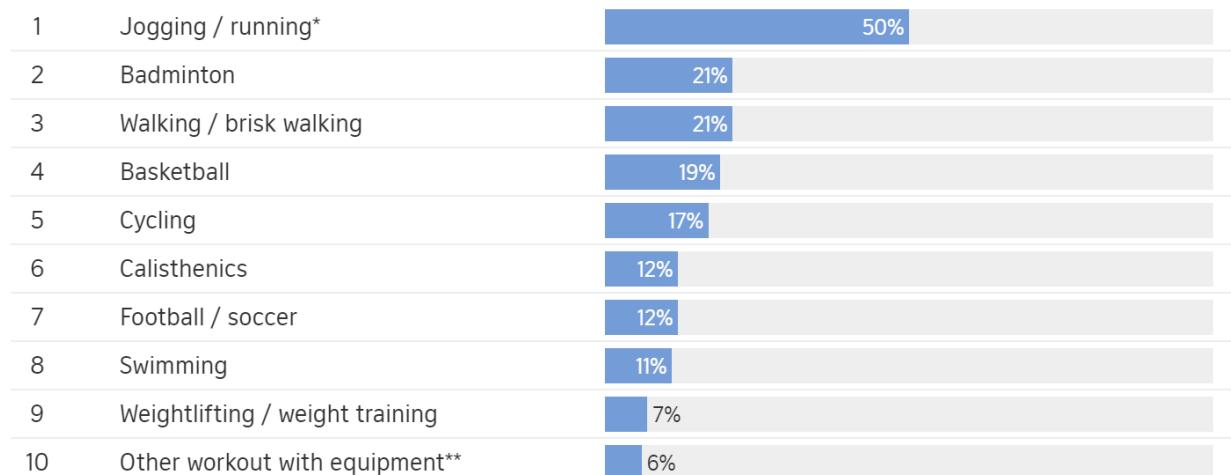
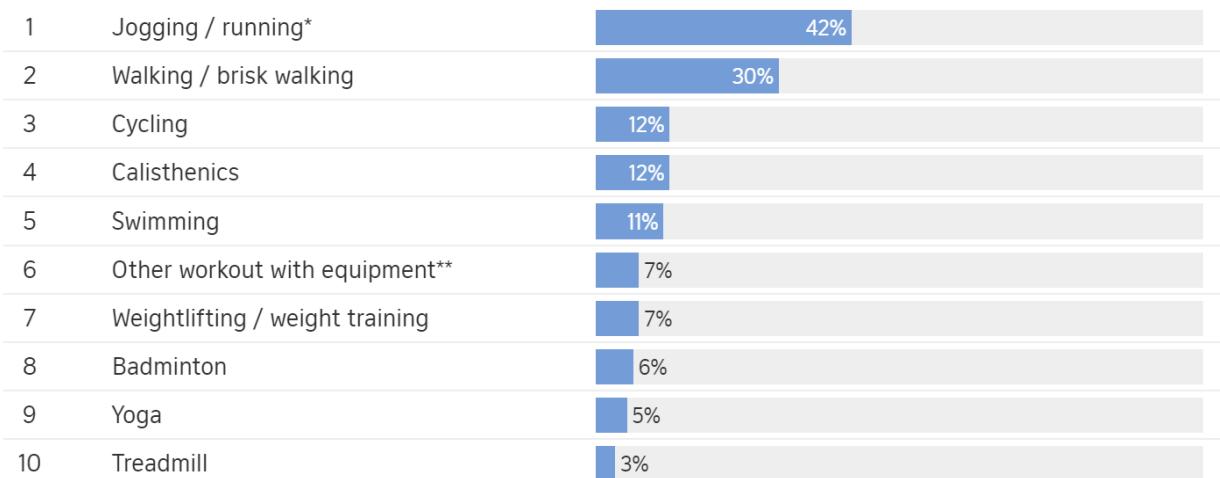


Diagram 4

Top 10 activities for young adults (aged 20-39)



About

Singaporeans generally lead busy urban lives with a mix of sedentary office work and varying levels of physical activity. While a significant portion of the population remains sedentary, there's also a growing fitness culture, particularly among younger adults who participate in gym activities, running, and other forms of exercise.

Literature Reference

A study done by National agency Sports Singapore highlights the increasing participation in sports and fitness programs, though it also notes a rise in sedentary behavior among certain demographics. From the article done by Straits Times mentioned that, "Released on Tuesday by national agency Sport Singapore (SportSG), the national survey found that 74% of 4,500 respondents aged 13 and above took part in sporting activities at least once a week." From the above diagram 3 and 4 shows that the highest percentage activity is jogging and running among other sports activities. From this research, it can be concluded that there has been an upward trend for sports activities. However this may also mean that there might be a platform to promote a community of people with similar interests.

2) Daily Routines

Diagram 5

	Total (N = 33)	Men (N = 15)	Women (N = 18)
<i>N (%)</i> , all such values			
Response to 'How often do you feel rushed or pressed for time?'			
Almost always	6 (18.2)	2 (13.3)	4 (22.2)
Often	7 (21.2)	4 (26.7)	3 (16.7)
Sometimes	14 (42.4)	5 (33.3)	9 (50.0)
Rarely	3 (9.1)	2 (13.3)	1 (5.6)
Never	3 (9.1)	2 (13.3)	1 (5.6)
Work satisfaction rating			
1–Extremely dissatisfied	0 (0.0)	0 (0.0)	0 (0.0)
2	0 (0.0)	0 (0.0)	0 (0.0)
3	0 (0.0)	0 (0.0)	0 (0.0)
4	5 (15.2)	2 (13.3)	3 (16.7)
5	14 (42.4)	6 (40.0)	8 (44.4)
6	9 (27.3)	4 (26.7)	5 (27.8)
7–Extremely Satisfied	5 (15.2)	3 (20.0)	2 (11.1)

About

Daily routines are often structured around work, with many Singaporeans eating out for lunch and dinner due to long working hours. The typical workday includes a mix of sedentary time in the office and some physical activity during commute or exercise sessions.

Literature Reference

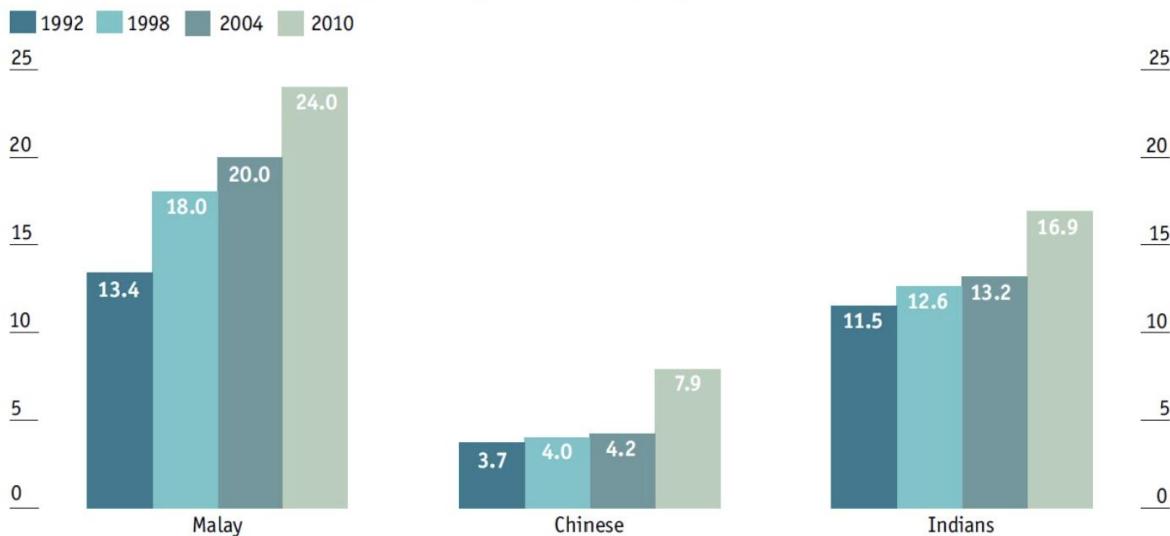
In diagram 5, we can see that there is a trend in Singaporeans who feel that they are pressed for time. A survey conducted by the National Library of Medicine reveals insights into the daily routines of Singaporeans, showing a correlation between long working hours and reliance on convenient food options. They concluded from the study that "For young working adults, long work hours are normalized, even though there is a barrier to healthy diets and physical activity. Existing social and institutional norms support a culture that values commitment to work and encourages young adults to devote long hours to building a sound financial future and achieving personal and cultural aspirations. These findings have implications for long-term population health and should be considered in health promotion activities targeting young adults and barriers." From this study it can be concluded that due to long working hours and cultural pressures, Singaporeans may not be able to exercise as often and practice healthy eating habits. With a lack of time and energy they will likely resort to fast food and instant meals.

Health Concerns

1) Common Health Issues

Diagram 6

Figure 11: Age-standardised adjusted prevalence of obese respondents (BMI equal or more than 30 kg/m²) in Singapore by ethnic group and year of survey (%)



About

The prevalence of chronic conditions like diabetes, hypertension, and obesity is increasing in Singapore, driven by diet and lifestyle factors.

Literature Reference

The Ministry of Health mentioned in an article that "Across all ages, our obesity prevalence has risen from 8.6% in 2017 to 10.5% in 2020. We have also returned to the previous level seen in 2010, after having decreased over the past decade." In diagram 6, shows a study done by Food Industry Asia of Singaporeans across the different ethnic groups in 2017. It shows that there is an increase in obesity cases, but has been concluded that the malay ethnic group has the highest number of cases. With this research, it can be concluded that there has been an upward trend of obesity cases due to a lack of discipline and accountability to eat healthy and choose better food options.

Dietary Impact on Health

Diagram 7

Dietary intakes

Dietary intakes of key foods and nutrients in adults aged 20 years and over compared against minimum and maximum targets

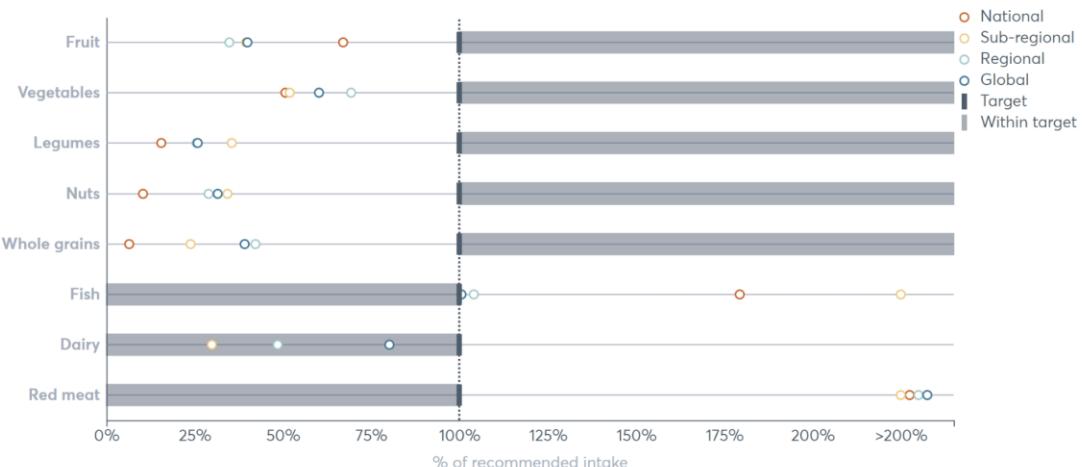


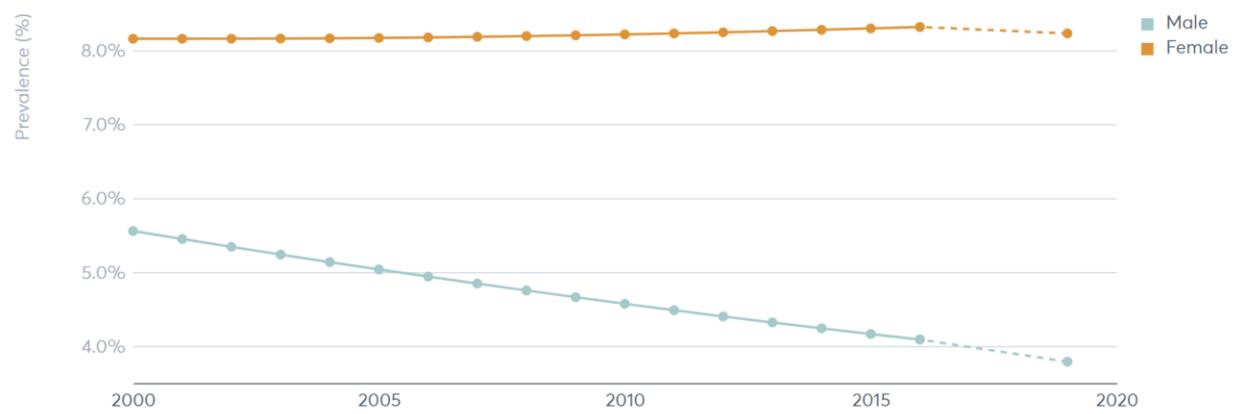
Diagram 8

Adult nutrition status and disease

Prevalence of underweight, overweight and obesity in adults aged 18 years and over

Select indicator:

Underweight Overweight Obesity



About

Research indicates that the consumption of high-calorie, nutrient-poor foods is a major risk factor for these health concerns. Singapore's adult population has also faced a lower nutritional intake over the years.

Literature Reference

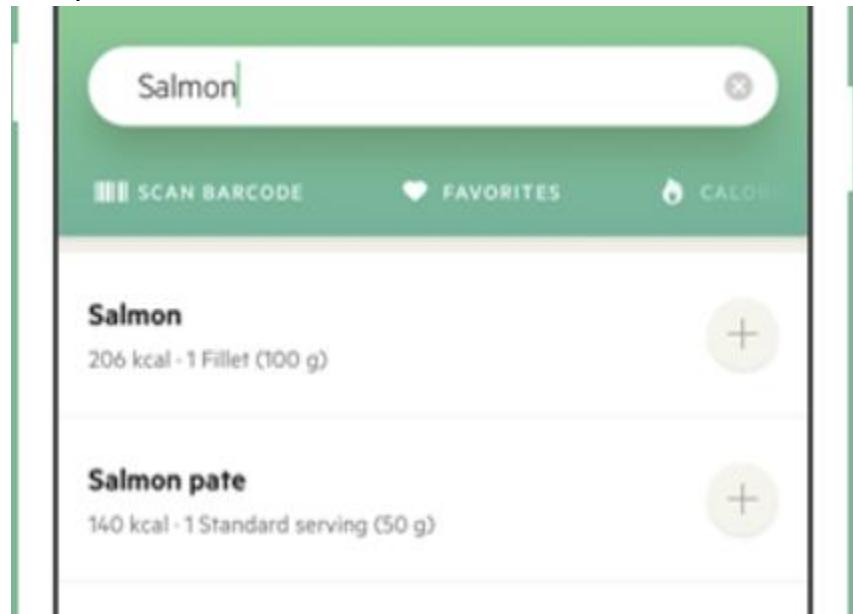
In diagrams 7 and 8 it shows the trends in the nutritional intake breakdown of Singaporeans as well as the trend of adult female and male percentage of low nutritional intake. From the diagram 7 we can see that there are more Singaporeans eating fruits and vegetables but do not eat as much whole grains. From the study, "6.9% of adult (aged 18 years and over) women and 6.7% of adult men are living with obesity. Singapore's obesity prevalence is lower than the regional average of 10.3% for women and 7.5% for men. The country is 'on course' to meet the target for diabetes when looking at adult women (6.2% affected), but they are 'off course' to meet the target for diabetes when looking at adult men (9.4% affected)." From this research, we can conclude that despite the higher calorie intake, this does not reflect a higher nutritional intake from the food that Singaporeans are consuming. This could be due to the lack of knowledge of nutritional information. This raises the need to have a quick way to search for food and be informed of macronutrients and calorie data.

Conclusion

About

These are the 5 components that we would like to include in our application that would aim to try to solve the issues faced by Singaporeans. These 5 components include a reliable food database source, community features, daily calorie tracking, insightful articles for browsing and an interactive feedback system.

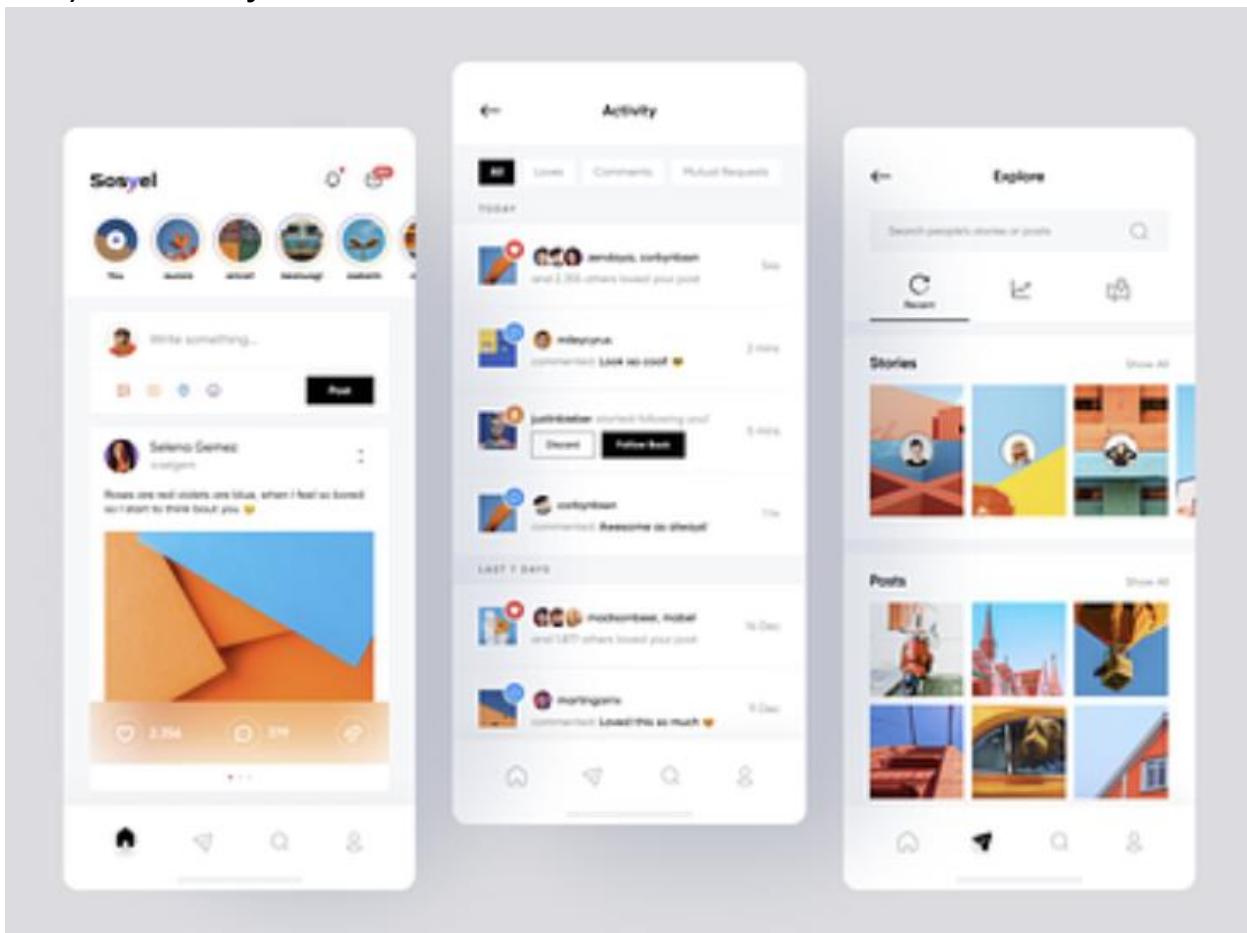
1) Food Database



About

The app should include a comprehensive database of local dishes, accounting for their high calorie and fat content. This aligns with the dietary habits identified in Singapore. The data that is included in the database should include macronutrients information such as fats, protein and carbohydrates. Which can be useful for the user to visualize their daily nutrient intake as well.

2) Community Features



About

By providing a community driven environment we aim to reflect the achievements and special moments from users that share similar fitness interests. This promotes inclusivity and encourages users who have lower motivation to eat healthier be more inclined to showcase their efforts and celebrate their achievements with the people around them.

Insightful articles for browsing



Buying Electric Bikes online through our BikeShop

23.06.2017, 04:00

Rapido BikeShop is proud to introduce our new electric bikes. The bikes are produced in UK,



Family vacation bikes

22.06.2017, 00:00

Rent cheap bikes for the entire family and experience the outdoor nature.



Vitoroni Foundation

22.06.2017, 00:00

The Vitoroni foundation is named after the great biker Sascha Vitoroni. He started a foundation to send a message to the young bikers about road safety.



Have the proper equipment when fixing a flat tire

19.06.2017, 00:00

You will get a flat tire in your lifetime. So why not be prepared? Better know what to do, when you experience this situation!



Top 3 best mountain bikes in 2018

16.06.2017, 12:00

We take a look on the top best mountain bikes in 2018, not looking price but on the looks and its performance through different terrains

About

Articles provide the user an opportunity to absorb interesting facts and events that happen in the fitness realm. This will motivate the user to use the app to learn new things and also share articles with the people around them. And not only forms good communication practices but also a stronger community with similar interests.

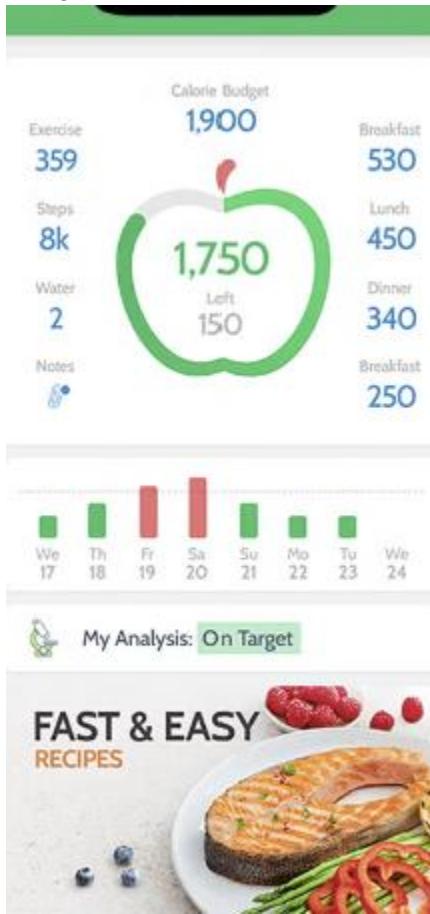
Interactive feedback



About

By providing a feedback option, it allows the user to input insightful comments and improvements for the system. Improvement is a big part in Agile Software Development that utilizes iterative project development where there are many versions of the app that constantly improves to meet the needs of its stakeholders.

Daily calorie tracker



About

A daily tracker provides the user a visual pleasing and dynamic interface. Where the user is able to input daily calorie consumption and improve each day. A daily tracker keeps the user accountable to his goals and encourages the user to eat less calorie dense foods. With a cleaner diet, it leads to a better mental health and immune system.

Technical Considerations

1) Platform & Tools

Target Android operating systems to reach the tech-savvy Singaporean audience. With the assumption that most Singaporeans have a smartphone with them, we want to create a convenient and straightforward app that caters to the busy lives of Singaporeans.

Android was chosen due to the easy integration and an easy execution of the app on emulators in Android studio. Android studio provides readily available emulators that the app can be hosted on. It provides developers with a quick way to visualize the app at a glance and change the user interface when needed.

2) Data Storage

Consider the use of databases to store information such as calorie data. Using a SQL database will allow for easy insertion and retrieval of macronutrients information.

Efficient data storage is also important as it affects the speed of execution of the function or algorithm. By using a structured database it promotes a faster app that will reflect as a more reliable app overall for Singaporeans.

3) API Integration

The accuracy of information is also an important factor for this app. Using APIs that have reliable and accurate data contributes to more goals achieved. APIs in this app can be integrated into the articles page and food search page.

APIs also provide a level of trust from users as these information are retrieved from trusted sources that carry out in depth studies and calculations on macronutrients. And as for articles, provides a wider range of articles that the user is able to browse and view. Thus keeping the user interested and interacting with the app more frequently.

4) User experience

Designing a user interface that reflects modern calorie tracking apps, increases the user experience level. With a modern and clean look, it allows tech savvy Singaporeans to navigate easily and utilize the features in the app.

By integrating this research into the development plan, the app can better meet the specific needs and preferences of Singaporeans, increasing its relevance and effectiveness.

Problem Analysis

1) Lack of Awareness

Many Singaporeans may not have a clear understanding of their daily calorie intake, especially when consuming local foods like hawker center dishes, which can be high in calories. Studies suggest that even health-conscious individuals often underestimate the calorie content of their meals.

Impact

Without accurate knowledge, users are likely to consume more calories than they intend, leading to weight management issues and other health concerns.

2) Complex Calculations

Calculating daily calorie intake manually is difficult, particularly when it comes to mixed dishes with varying ingredients, which is common in Singapore. For instance, tracking the exact calories in a dish like chicken rice.

Impact

The complexity and time required to do this manually can discourage users from tracking their diet consistently.

3) Limited Access to Nutritional Information

While international foods often have clear nutritional labels, many local dishes in Singapore do not. This makes it hard for users to accurately track their calorie intake, especially when eating at hawker stalls or smaller eateries that don't provide nutritional information.

Impact

This lack of transparency leads to inaccurate calorie tracking, hindering users from making informed dietary choices.

4) Lack of Motivation

Consistently tracking calories requires discipline and long-term commitment. Users may initially be motivated but might lose interest over time due to the perceived complexity or lack of visible progress. Without engaging features, many users abandon their health goals.

Impact

Users often fail to stick to their calorie tracking efforts, making it difficult for them to achieve their health and fitness goals.

5) Integration with Lifestyle

Singaporeans lead fast-paced lives with long working hours, making it challenging to balance health and fitness with daily routines. The app should integrate seamlessly into their busy lives, enabling easy tracking even on the go.

Impact

If the app doesn't fit naturally into their daily routines, users are less likely to consistently use it, limiting its effectiveness in helping them manage their diet.

Argumentation

1) Convenience and Efficiency

This app simplifies the process of tracking calories by automating calorie calculations. Users can search for foods by name, or select common Singaporean dishes from the database. This saves time and effort, making calorie tracking a seamless part of their daily routine.

Supporting Feature

Automated records of calorie intake as well as a dynamic progress tracker resolves this problem by providing visual and quick reference information on daily calorie count as well as macronutrients data.

2) Comprehensive Food Database

A key strength of your app is its extensive food database, which includes both international and local dishes like chicken rice. This ensures that users have access to accurate nutritional information, helping them make informed dietary choices.

Supporting Feature

Users can also add ingredients, making the app adaptable to different diets.

3) Motivation and Engagement

To keep users engaged, the app includes features like progress tracking, visual history of past calorie records. Users can reach daily limits and will be encouraged each day with pop ups with encouraging messages. Past records will also be color coded with either red or green. With more greens in the record, it indicates to the user a streak of calories for the week.

Supporting Feature

Integration with social features and community forums helps users stay motivated by sharing their progress and learning from others.

Conclusion

By addressing these problems with your app's features, you demonstrate a clear value proposition for Singaporean users. It simplifies calorie tracking, provides local food data, keeps users motivated, and fits seamlessly into their fast-paced lifestyles. The combination of convenience, personalization, and engagement will help differentiate your app from competitors, making it a compelling choice for health-conscious users in Singapore.

Planning

Planning and research

Hint: Planning and logging time for each member and what each member has done over the time period)

This section details the planning process, including time logs and the allocation of tasks to each team member. Over the project timeline, (member names) have undertaken backend development, user experience and user interface design. Research was conducted on relevant research topics such as best practices in user interface design and efficient application navigation to inform the project's development.

Project Team & Stakeholders

Before starting on the project, the team has decided to delegate roles based on the strengths of the team members. This not only significantly increases the project's chances of success but also lets the team produce a higher quality of work. With each team member being aware of their specific tasks, it greatly reduced confusion and overlap during the development phase and made it easier to track individual contributions while allowing the team to address any performance issues promptly.

Project Manager

Chen Soon Keat oversees the entire project, ensuring smooth execution and timely delivery. His responsibilities include:

- Project planning and timeline management
- Facilitating communication between team members and stakeholders
- Identifying and mitigating project risks
- Organizing regular team meetings and progress reviews
- Ensuring the project aligns with user needs

UI/UX Designer

Boovana and Bryan Loo are responsible for creating an intuitive and visually appealing user interface. Their tasks include:

- Conducting user research and creating user personas
- Designing wireframes and prototypes
- Creating a cohesive visual design and style guide
- Collaborating with the developer to ensure design feasibility
- Conducting usability tests and iterating on designs based on feedback

Backend Developer

Yuxiang and Bryan Tan will focus on server-side logic and database management. These tasks include:

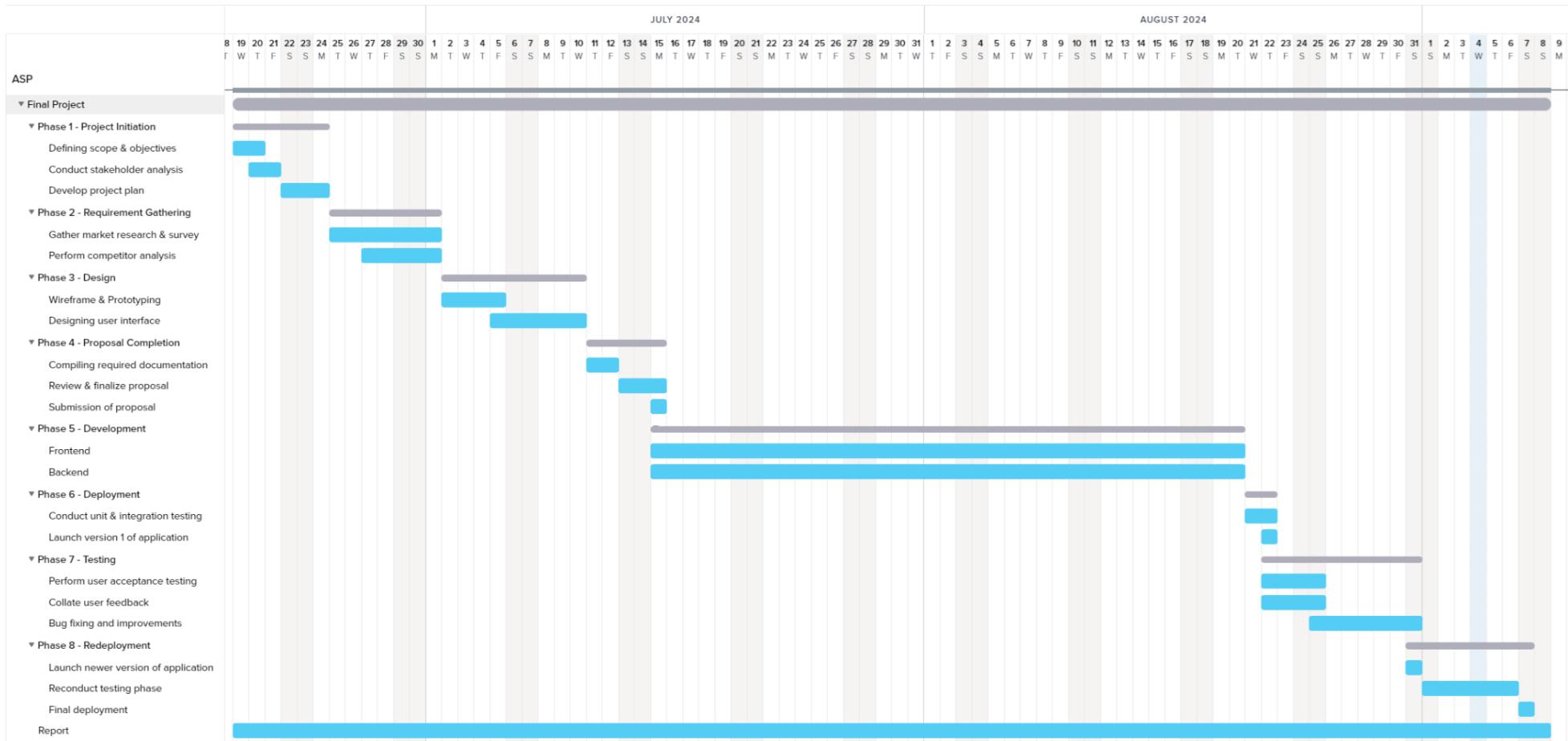
- Designing and implementing the database schema
- Curating APIs for the mobile app
- Ensuring data security and user privacy
- Implementing user authentication and authorization
- Optimizing backend performance and scalability

Project End Users

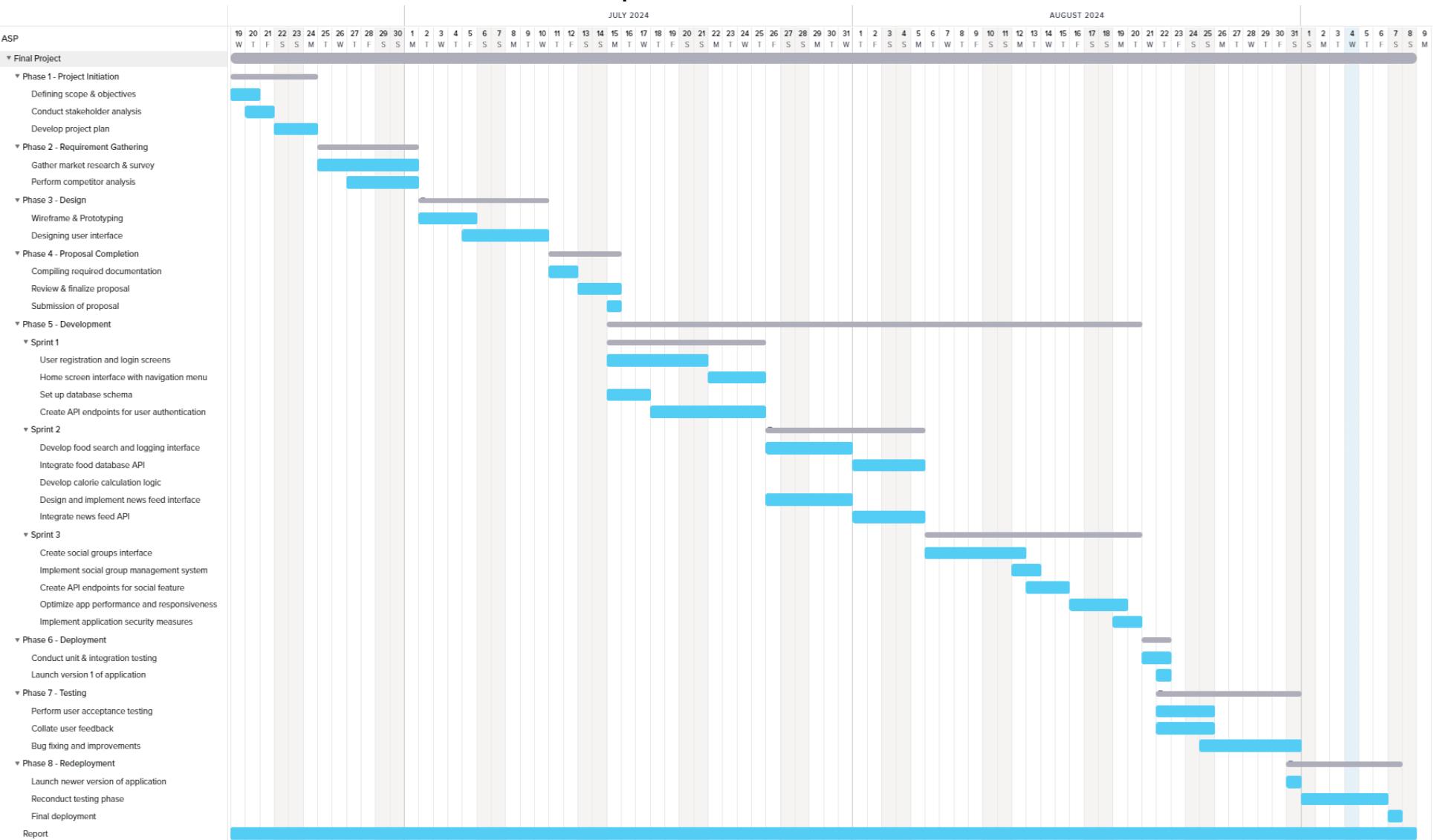
The targeted end users will be any individuals that would like a better insight of the calories and different macro nutrients from the food they consume on a day to day basis. This is specifically useful for people who want to take that first step into making a lifestyle change or even those with medical conditions such as diabetics that want to monitor their carbohydrate intakes. All in all, the app serves a multitude of purpose and can be used flexibly.

Gantt Chart

Initial Gantt Chart



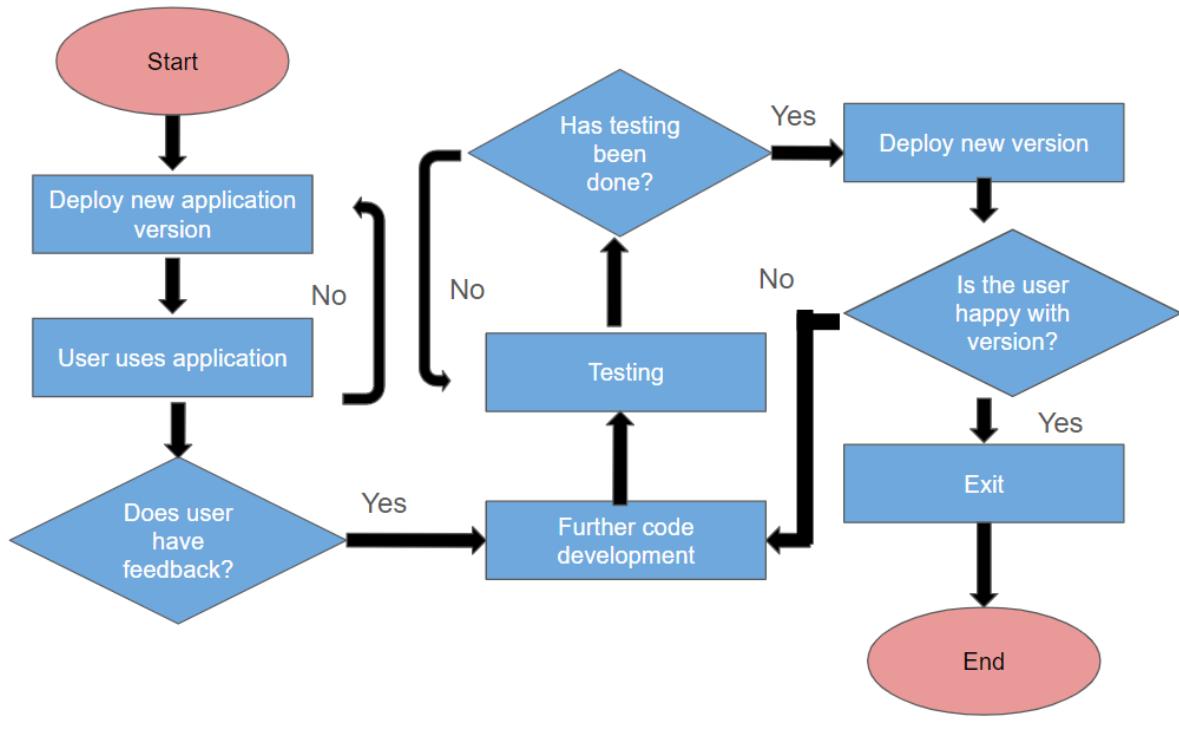
Updated Gantt Chart



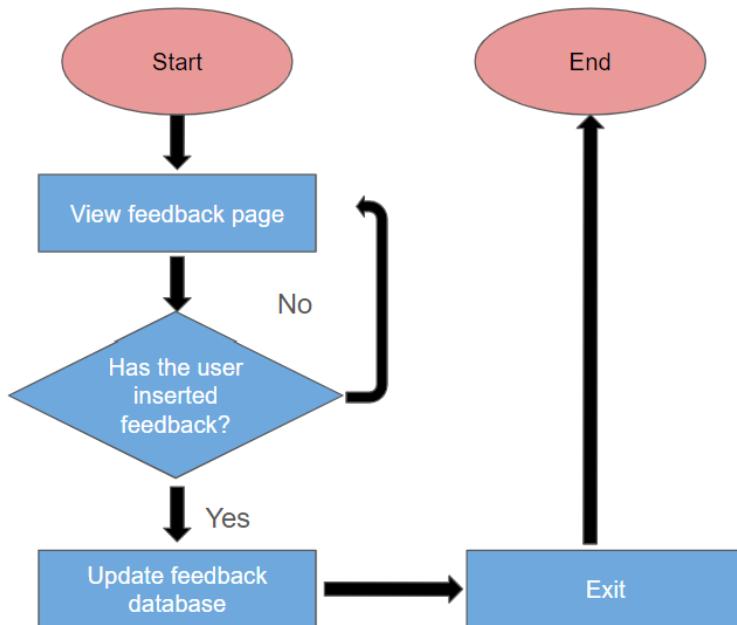
The planning and execution of this project was guided by a well-structured Gantt Chart, which gives us a rough timeline and key milestones we needed to hit throughout the development phase. The Gantt Chart was pivotal in the application's development as it allowed us to monitor progress and keep the team on track by ensuring tasks were completed on time. In the case that the team is unable to hit the deadline for the specific milestone, adjustments can be made to the Gantt Chart to reallocate resources and revise the timeline so as to minimize delay for the project to be completed on time.

Fortunately, the team was able to adhere to the initial time allocation, successfully meeting all milestones as planned. However during the development phase, we thought it would be more beneficial if we broke it up into *sprints*. This allowed us to better manage our workload as it gave us a clearer idea of the features that needed to be completed by a specific date, while also letting us ship high quality work faster and more frequently. The team took about a week and a half for each sprint, with the last sprint spanning 2 weeks to give us additional time to optimize and work on the security aspect of the application.

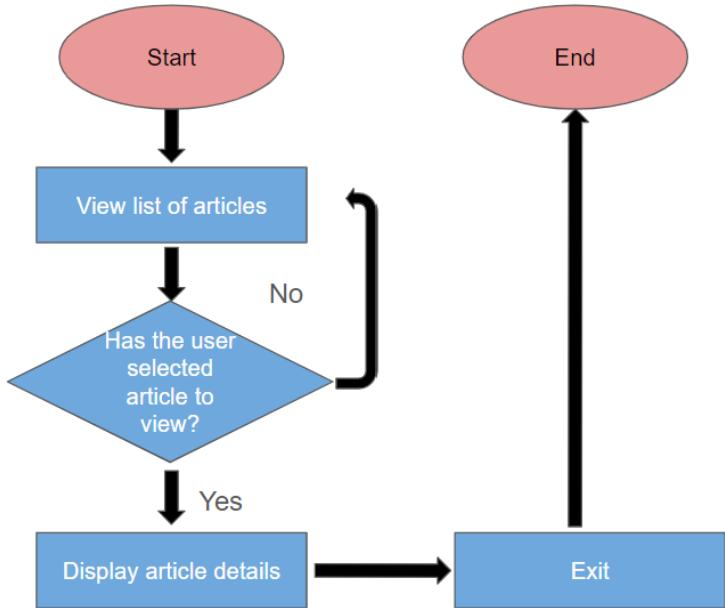
Backend processes
Iteration and deployment plan



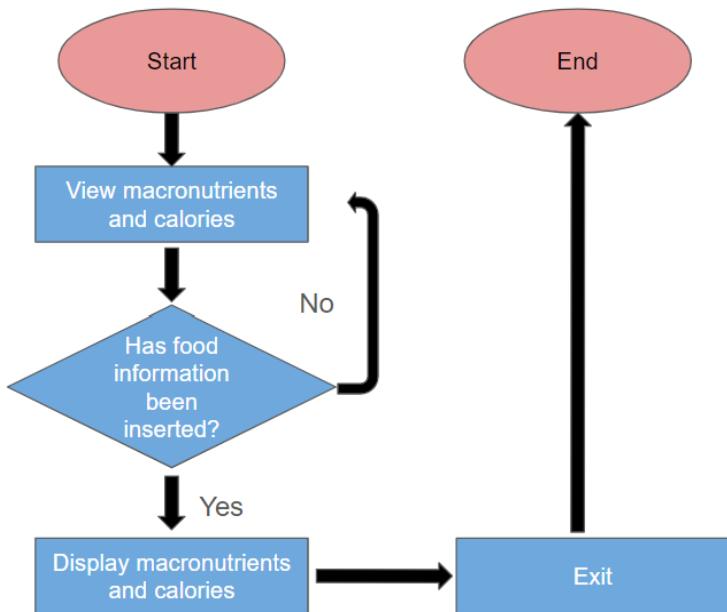
Feedback page - Submit feedback



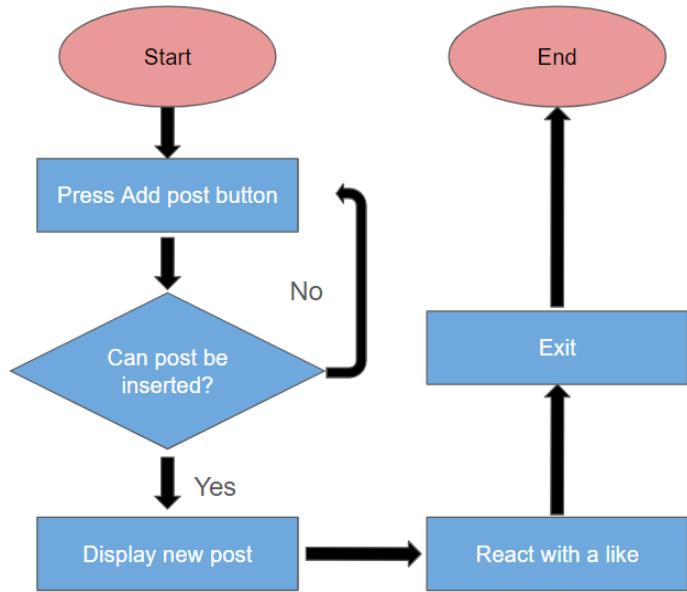
Article page - view article



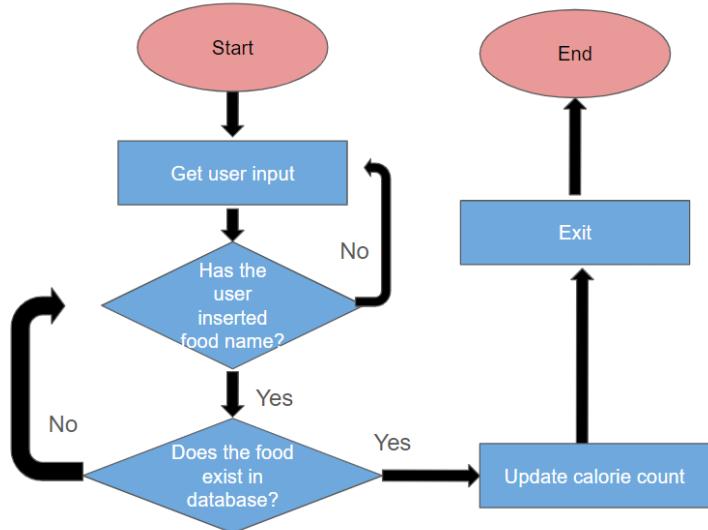
Home screen page - display macro nutrients and calories



Social screen page - Add post



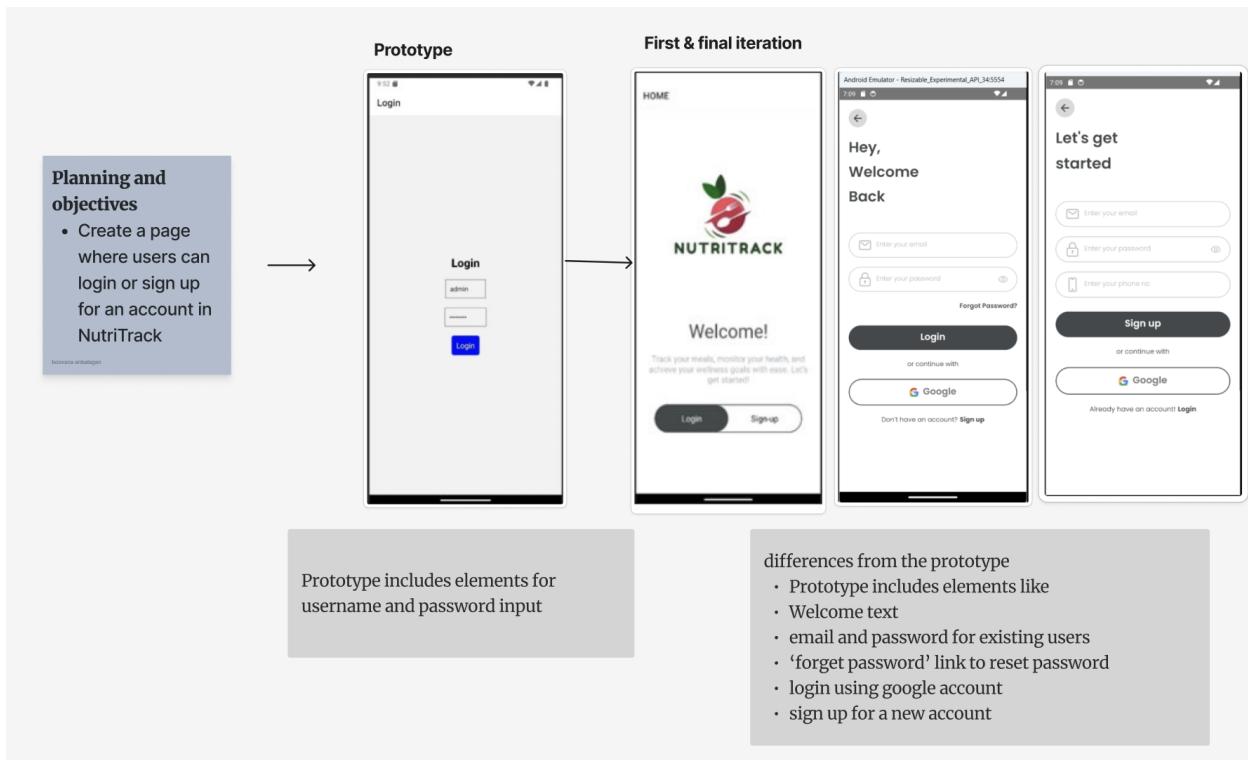
Search food page - Update calories



Prototyping and iteration

Login page

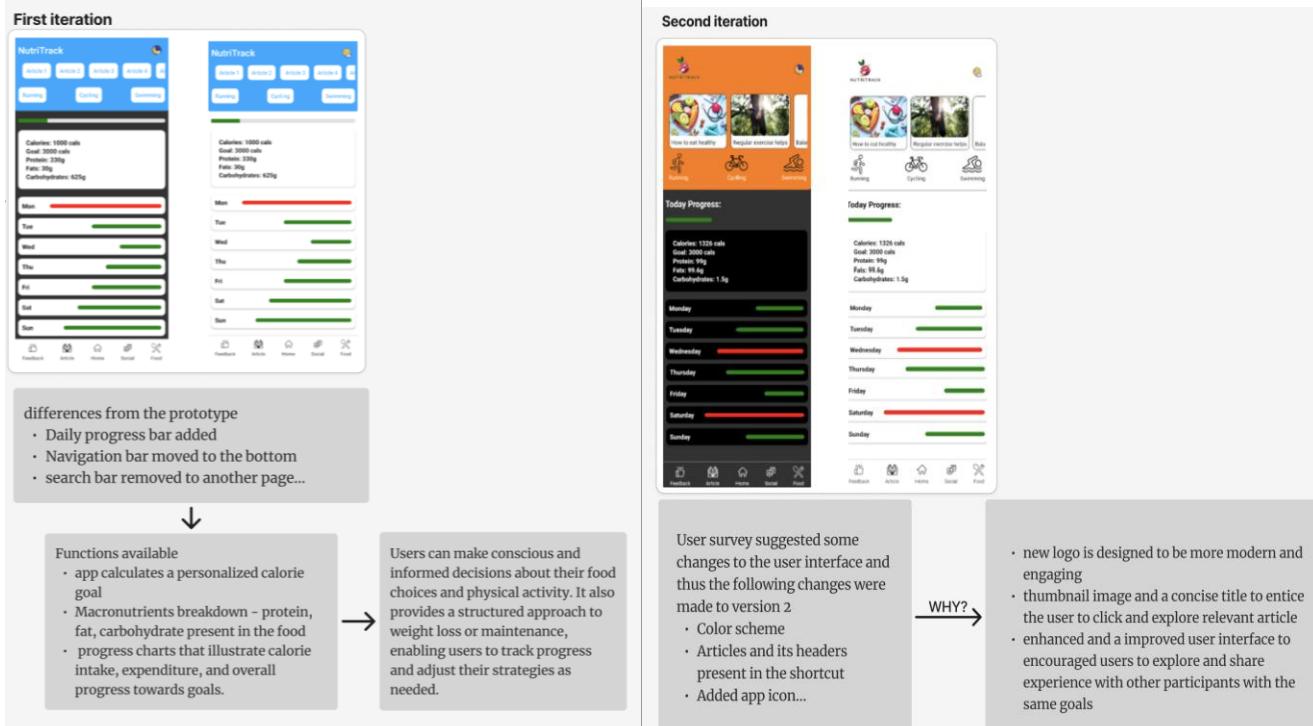
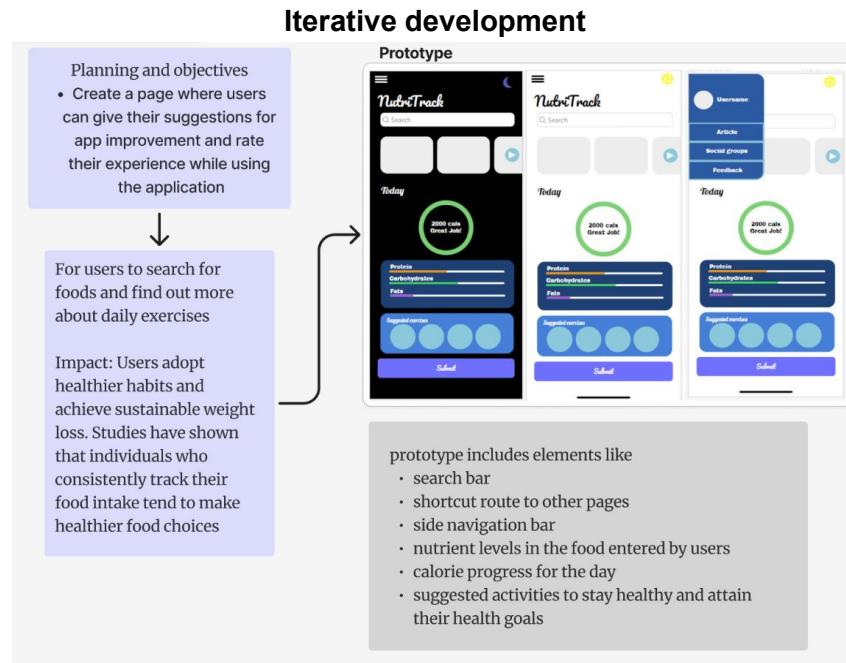
Iterative development



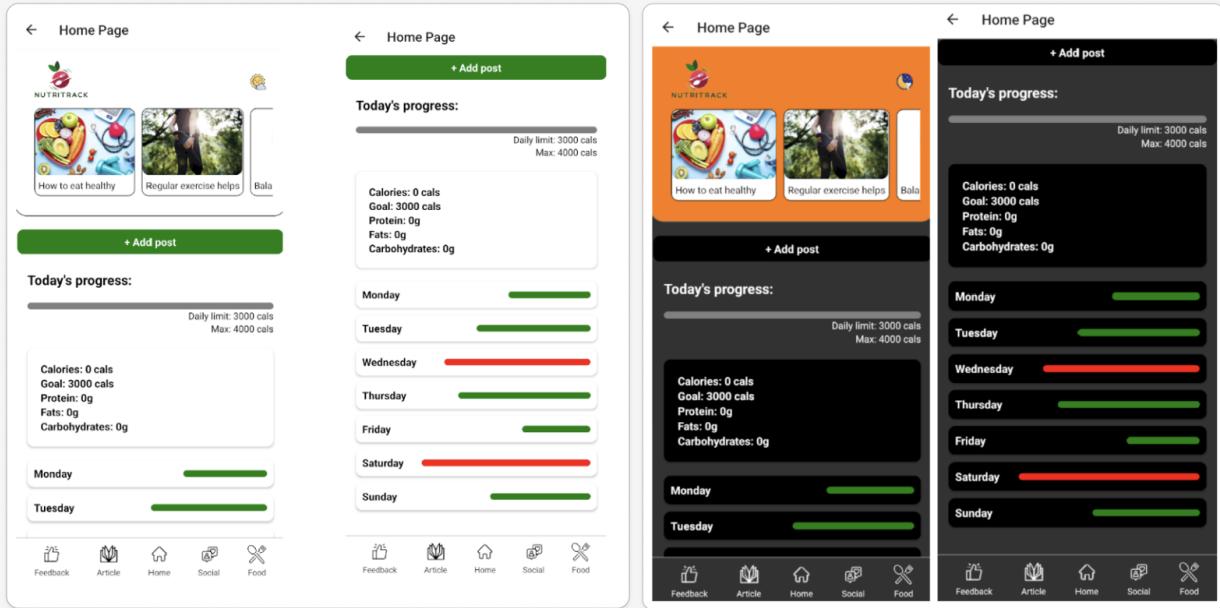
Evaluation of work

The prototype is a simple design of a typical login page and it has been enhanced in the first iteration where multiple elements have been added to make the login process more convenient. The final iteration of the page includes email and password input for existing members to login in, as well as a forget password link which is pertinent to access their account. New users can either sign up for a new account completely or may continue with their existing google account, saving them the hassle of entering the particulars.

Home page



Final iteration

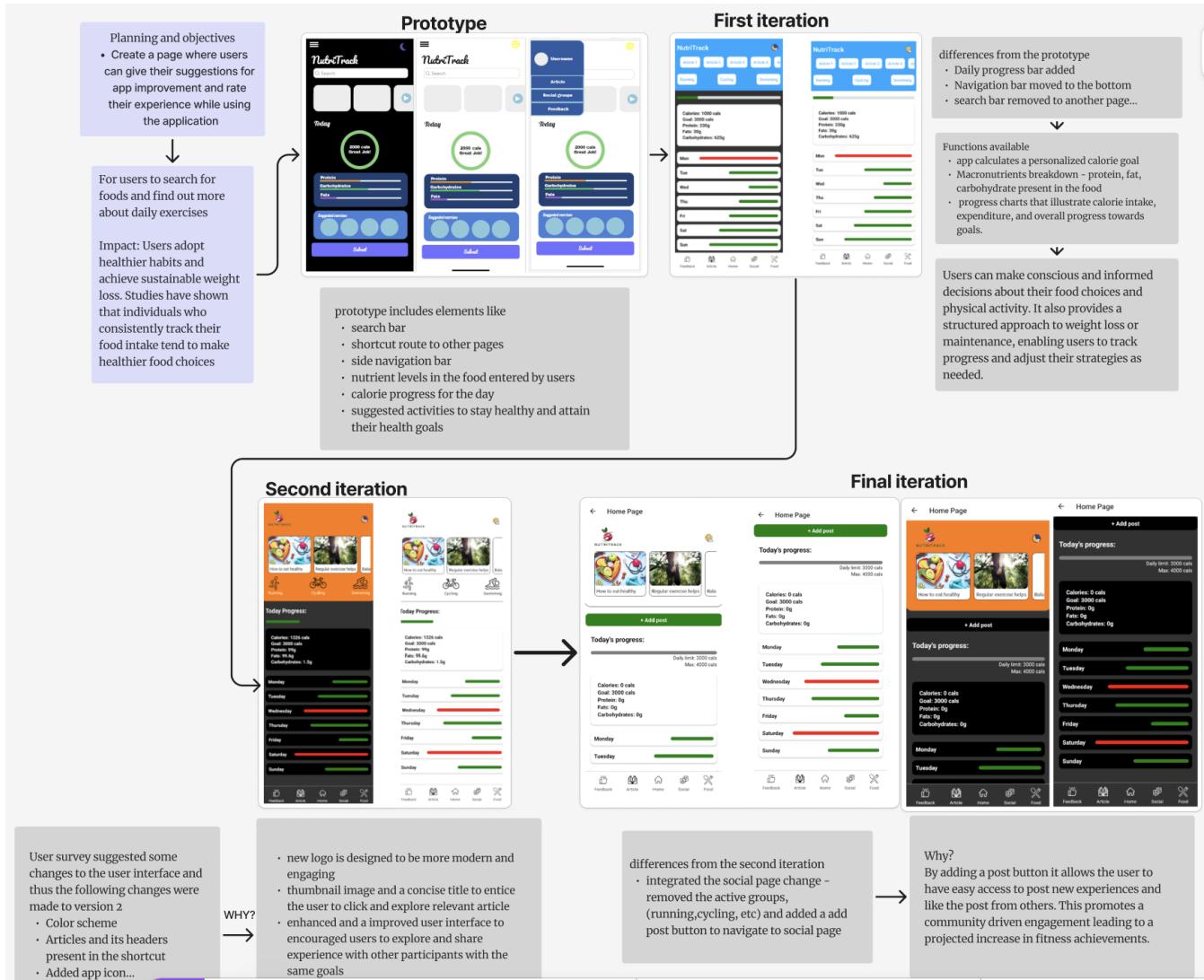


differences from the second iteration

- integrated the social page change - removed the active groups, (running,cycling, etc) and added a add post button to navigate to social page

Why?

By adding a post button it allows the user to have easy access to post new experiences and like the post from others. This promotes a community driven engagement leading to a projected increase in fitness achievements.

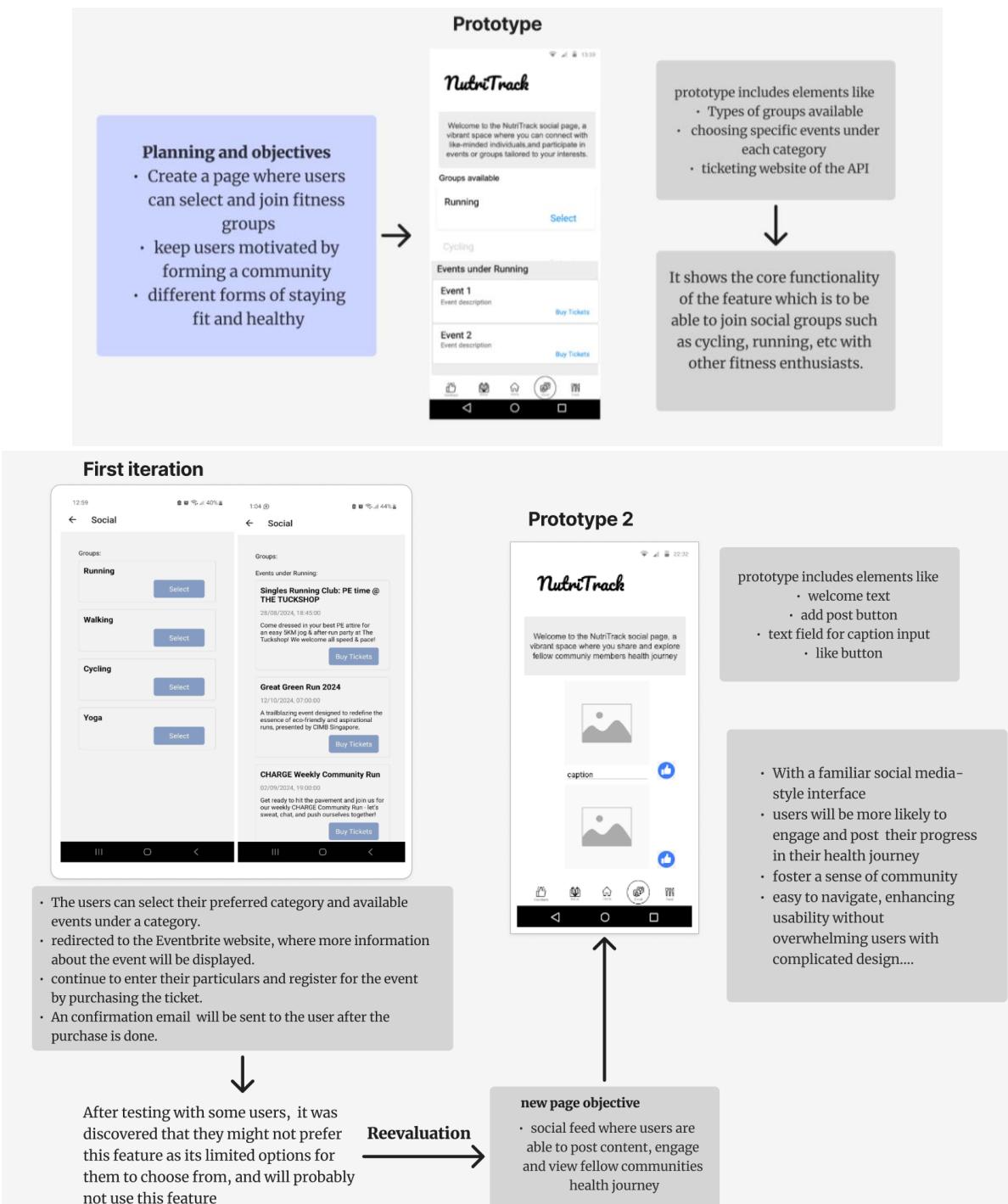


Evaluation of work

The first version and prototype included suggested exercises and also a search food function. However through continuous improvement and research. The final version of the home page provides a seamless, clean and modern look. The home page now mainly displays daily calorie count, past calorie records, list of articles and an interactive post button. With this design, it provides the user with a more visually appealing and higher usability overall.

Social page

Iterative development



Second iteration

→

Pass testing

Final iteration

Designed to create a community-centered platform within the NutriTrack app. Users can engage with each other by sharing images, posts, and thoughts related to their health, fitness, or nutrition journey.

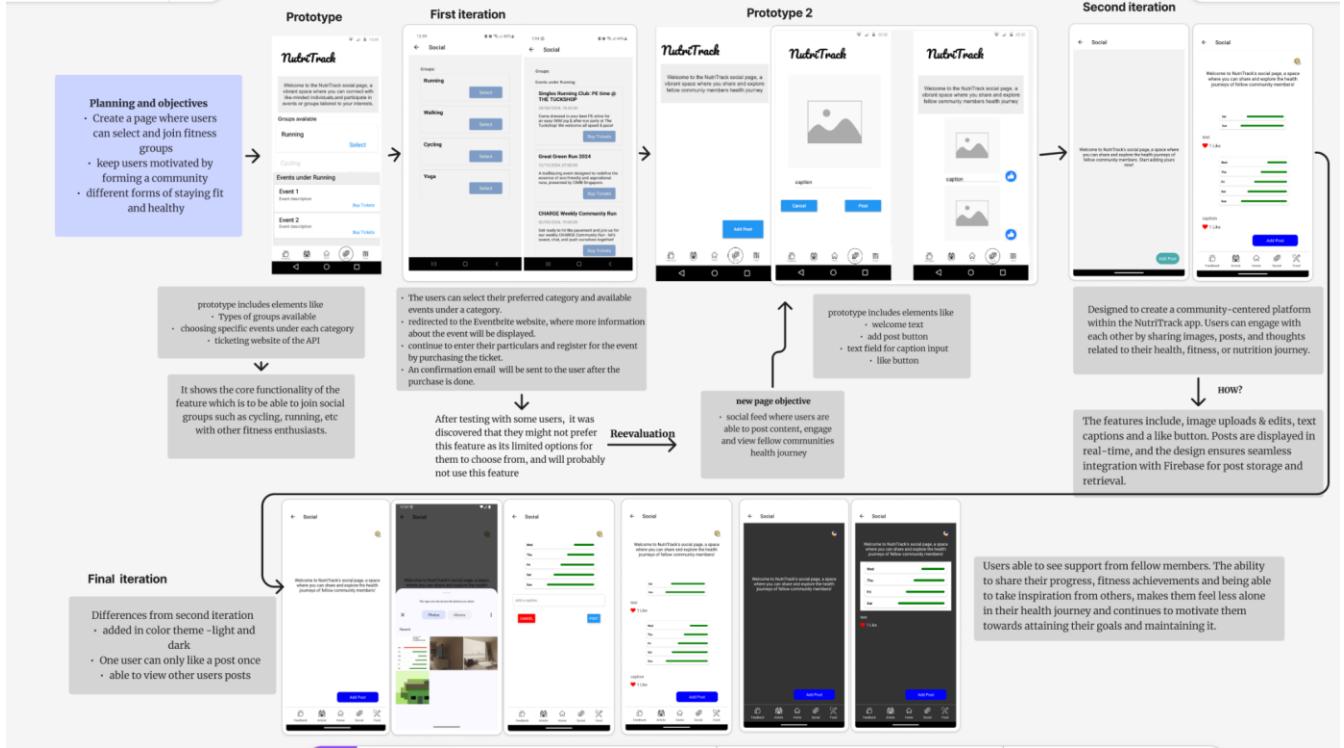
▼ HOW?

The features include, image uploads & edits, text captions and a like button. Posts are displayed in real-time, and the design ensures seamless integration with Firebase for post storage and retrieval.

Differences from second iteration

- added in color theme - light and dark
- One user can only like a post once
- able to view other users posts

Users able to see support from fellow members. The ability to share their progress, fitness achievements and being able to take inspiration from others, makes them feel less alone in their health journey and continues to motivate them towards attaining their goals and maintaining it.

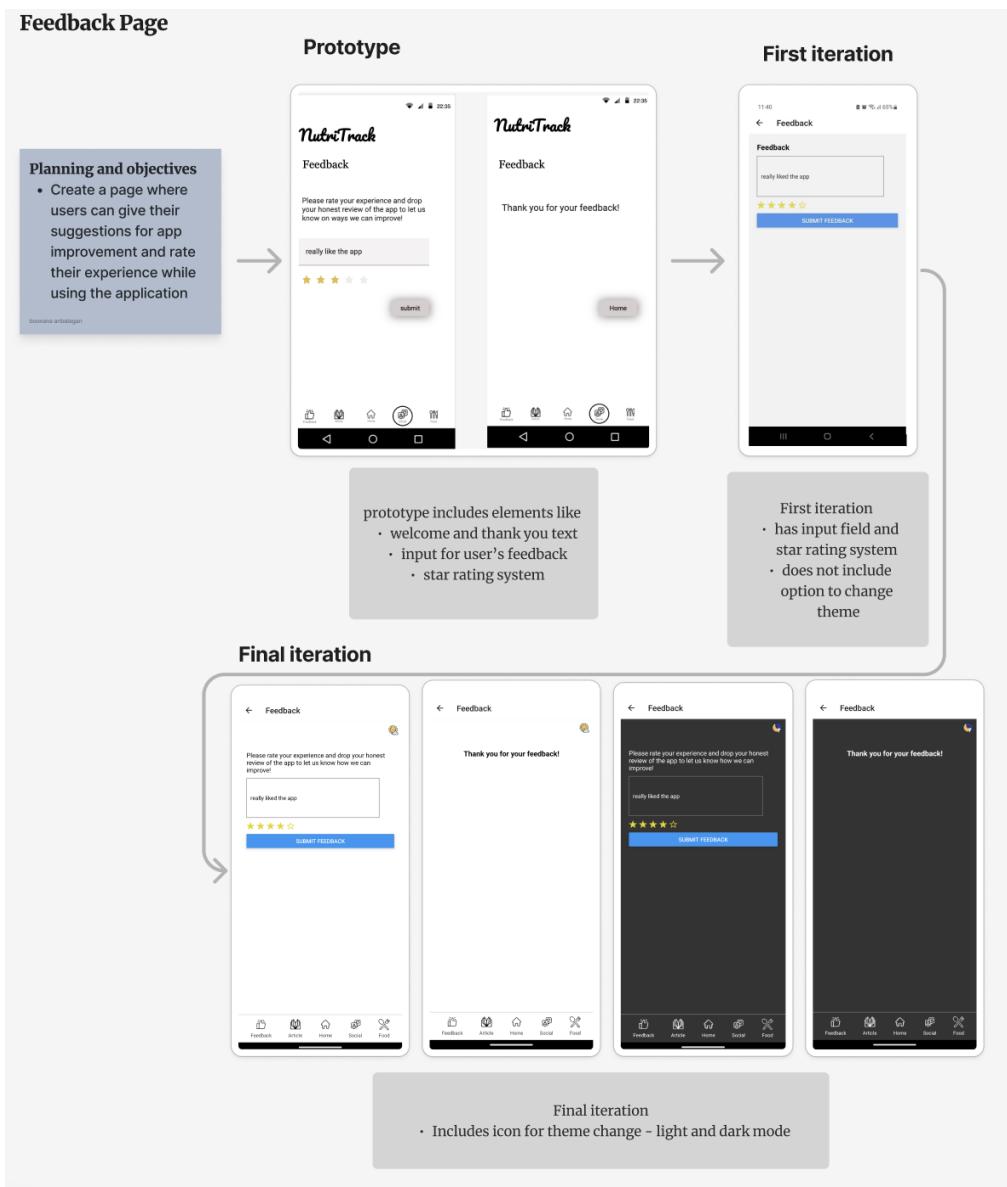


Evaluation of work

The first version, the social page started as an activities page consisting mainly of 4 exercises, running, walking, cycling and yoga. Where the user would be able to click into and explore sports activities to participate in. But through research on current calorie tracking apps we decided that through shared experiences and interconnectedness over social media style type pages, that Singaporeans would be more inclined to use a social media sharing platform. Thus in the final version, it provides a sharing platform, where users are able to react to post of other users and as well as share their own achievements. Posting is straightforward and easy to learn.

Feedback page

Iterative development



Evaluation of work

The first version provided a simple but effective feedback page, where users are able to provide improvements. Users can share their concerns and enquiries there as well. The overall design did not change much over time. With the final version of the page, there was a more elaborate theme change. This allows the user to view the page in two different modes. Hence promoting customisation and personalized user experience.

Search page

Iterative development

Planning and objectives

- Create a page where the users can search for the nutrients and calories in a particular food item
- Able to save and view the food on the homepage

Prototype 1

First iteration

prototype includes elements like

- side navigation bar
- Search bar for users to input the food
- separate input field to adjust the grams
- Gives the nutritional value present in the food searched - Carbs, fats, protein and calories

Second iteration

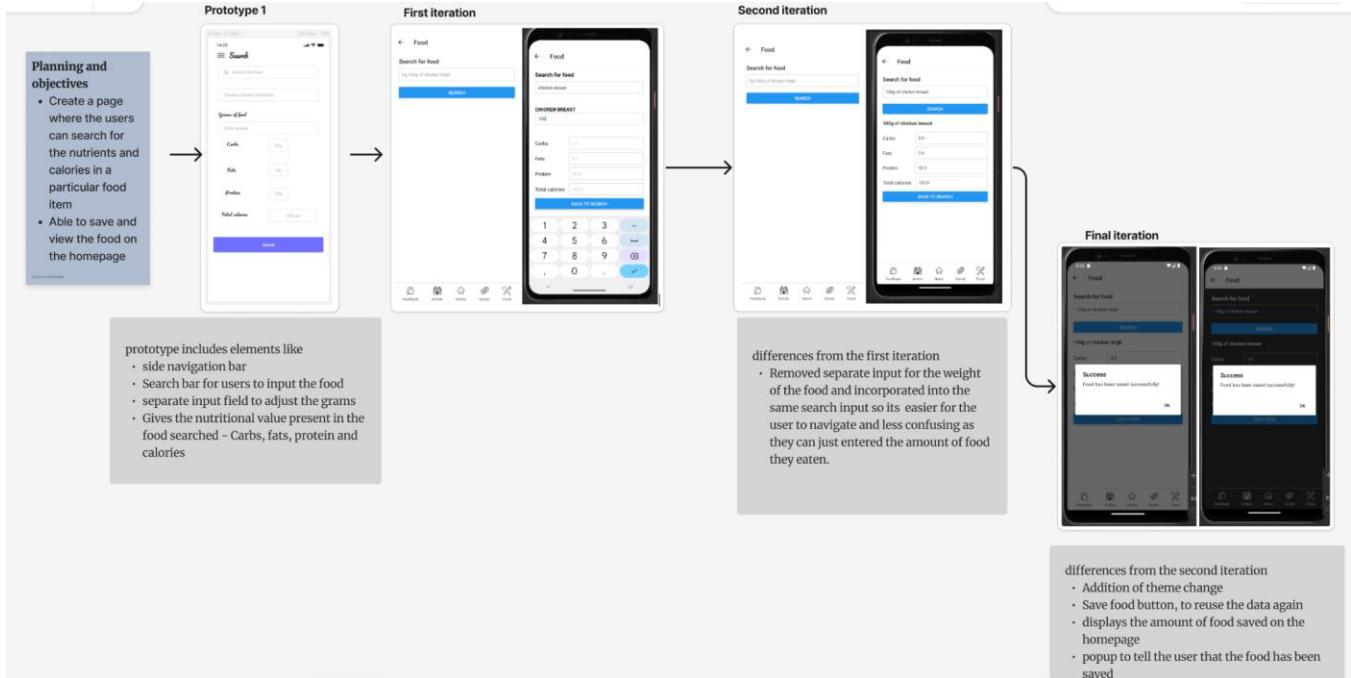
Final iteration

differences from the first iteration

- Removed separate input for the weight of the food and incorporated into the same search input so its easier for the user to navigate and less confusing as they can just enter the amount of food they eaten.

differences from the second iteration

- Addition of theme change
- Save food button, to reuse the data again
- displays the amount of food saved on the homepage
- popup to tell the user that the food has been saved

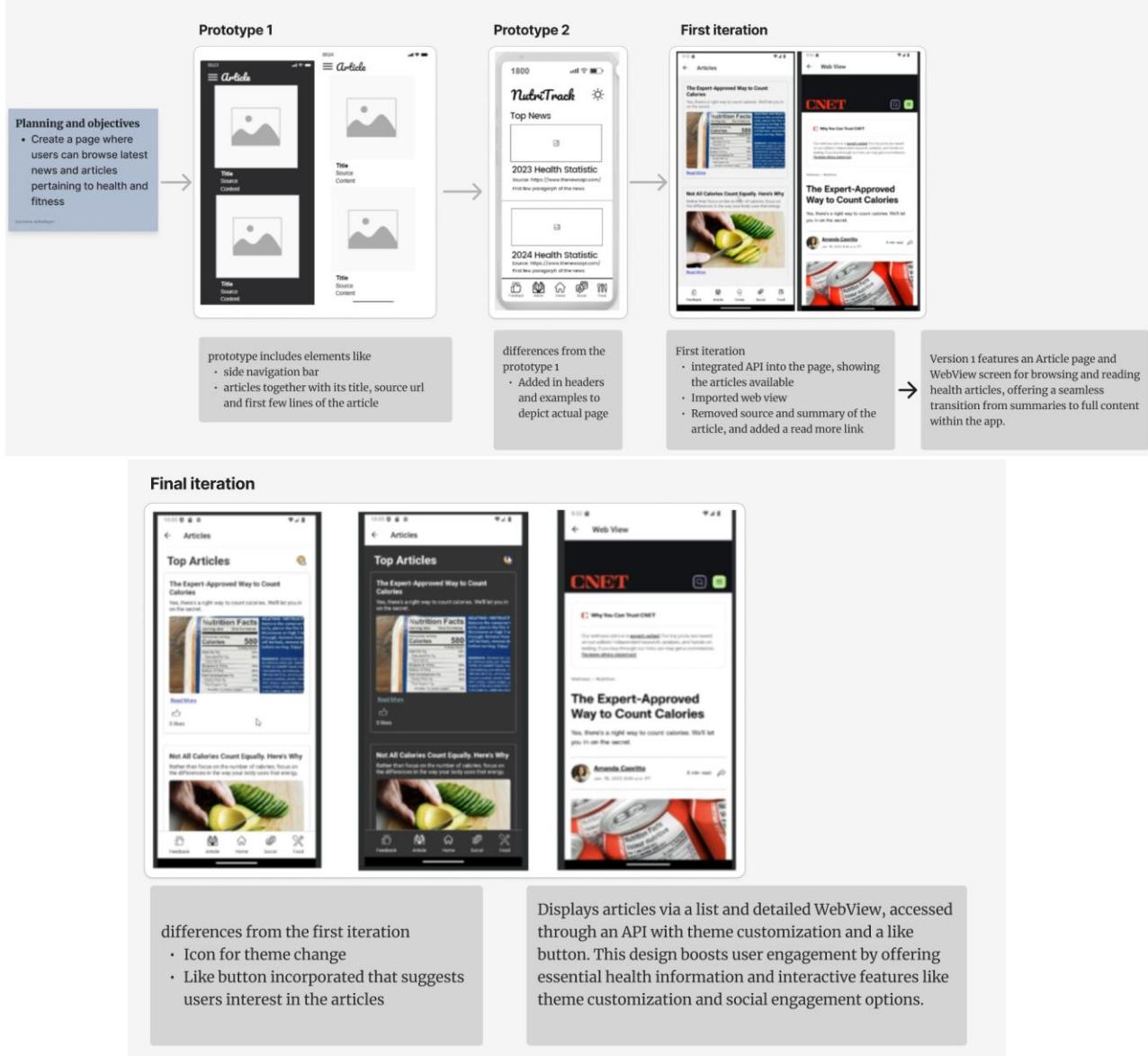


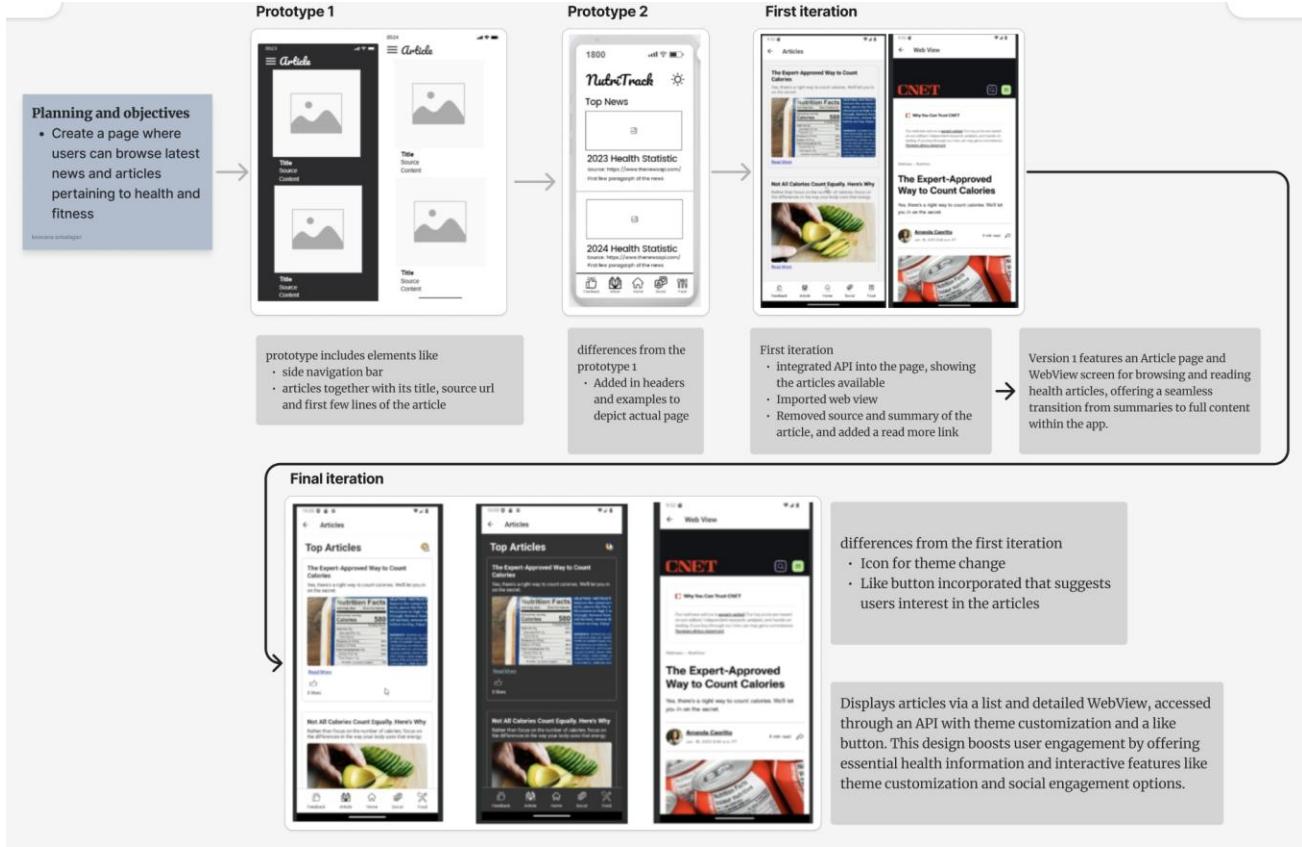
Evaluation of work

The first version of this page provides the user with a simple and seamless look, where users are able to search for food, view macronutrients information as well as add the food to the total daily calorie count. Through iteration and research, the final version provides an interactive and user-friendly experience. The page provides success message pop ups to indicate successful insertion of food and also a more modern user interface design to cater to tech savvy Singaporeans.

Article page

Iterative development





Evaluation of work

The first version of the page provides a simple and clean look, where users are able to browse through various articles of their interest as well as an interactive read more button to click into a read more about the topic. Through research and iterative development, the final version provides a more user-friendly experience. By providing a like button and an improved modern feel and style, the user is able to react and interact with the page with a higher level of interest and curiosity.

Design

The design of the system was informed by both user feedback and technical requirements. The specifications were continually refined through prototyping and evaluation. This section also details the design specifications and the approach taken to ensure a robust and scalable system.

Requirements definition

This project is an agile software prototype for daily caloric tracking. The main purpose of this application is to encourage users to insert calories on a daily basis and provide insightful nutritional information such as macronutrients.

The app also comes with other areas of interest such as an article page for the user to browse and explore, a page to post experience and react with posts from other users and a page to provide feedback and improvements.

Functional requirements

- 1) Feedback page
 - Add feedback text and star rating
 - Upon pressing submit and successful submission a success message is displayed.
 - If submission has failed, an error message is returned.
- 2) Articles page
 - Can browse articles
 - Can add likes to various articles
 - Click on read more button to read the selected article
- 3) Home page
 - Change theme of the app screen
 - Browse from a list of articles
 - Press on the post button and redirect to social page
 - Update daily caloric count and display progress on the progress bar
 - Display dynamic macronutrients inserted by the user
 - View the general calorie count for the past week
- 4) Social page
 - Add a new post
 - Post includes a photo and a description
 - Updates the list of posts as the posts are inserted
 - Add likes to various posts

- 5) Search food page
 - Search for food
 - View macronutrients information before adding to daily caloric tracker
 - Add food to daily caloric tracker
 - Upon successful insertion, a success message is returned

Functional complexity

To measure the functional complexity of this project we can visualize this in these three areas. These include functionality, complexity and time consumption.

Functionality

- This refers to how important it is to add the functions into the app. This means how much these proposed functions will impact the target users.

Complexity

- This refers to how difficult it is to write the code for the function. This means how complex the code for the function is going to be and whether it can be done.

Time consumption

- This refers to how time consuming it is to code the function out. This means how much time will it take to carry out the implementations for the functions.

Development Issues and Problems

UI Evaluation

Survey 1:

URL: https://docs.google.com/forms/d/e/1FAIpQLSdVolzS-b12wXjApkGzyEH6ei1qJZwUJZs8sVjXsEEo_ivrVg/viewform

Login Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Login Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
2	3
1	4
1	5
3	3
1	5
2	3
1	5
3	5
5	5
1	5
2	4
1	5
2	4
2	4
3	5
2	4.333333333

The average score users rated the UI of the login page is 2 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.33 out of 5.

Dark Mode:

Do you think having a dark mode is an important feature for an application?

Yes

Yes

Yes

Yes

Yes

Yes

Yes

No

No

Yes

No

Yes

Feedback Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Feedback Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
2	4
3	4
3	4
3	3
3	4
3	4
3	5
2	5
4	4
2	5
3	4
2	5
3	4
3	4
3	5
2.8	4.266666667

The average score users rated the UI of the feedback page is 2.8 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.26 out of 5.

Article Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Article Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
3	3
4	4
3	4
3	3
3	3
3	4
4	5
2	5
3	4
4	5
2	4
4	5
4	4
3	4
3	5
3.2	4.133333333

The average score users rated the UI of the article page is 3.2 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.13 out of 5.

Social Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Social Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
3	4
4	4
3	3
3	3
3	3
3	4
1	1
3	4
4	4
4	5
3	4
1	1
1	4
3	4
3	5
2.8	
3.5333333333	

The average score users rated the UI of the social page is 2.8 out of 5. Additionally, the average score for how easy it was to navigate through this page is 3.53 out of 5.

Search Food Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Search Food Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
2	4
4	4
3	4
3	4
3	4
3	4
4	5
4	5
3	4
2	5
3	4
4	5
4	4
3	4
3	5
3.2	
4.0666666667	

The average score users rated the UI of the search food page is 3.2 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.06 out of 5.

Survey 2:

URL:

https://docs.google.com/forms/d/e/1FAIpQLSfNMLecVnwEDMUnB7HK_M3VhQKpeDOaxEtNZ08p0_FRylakqg/viewform

Login Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Login Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
4	5
4	4
4	5
4	5
4	4
4	4
4	5
4	5
5	5
4	4
5	4
4	4
4	5
4	5
4	5
4.133333333	4.466666667

The average score users rated the UI of the login page is 4.13 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.46 out of 5.

Home Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Home Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
1	4
2	4
1	2
3	5
3	4
4	4
3	4
4	5
3	4
3	5
3	4
2	4
4	4
3	5
4	5
2.866666667	4.2

The average score users rated the UI of the home page is 2.86 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.2 out of 5.

Feedback Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Feedback Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
1	5
3	5
2	5
3	4
3	4
4	4
3	4
4	5
3	4
2	5
3	4
5	5
4	5
4	5
3	4
3.133333333	
4.533333333	

The average score users rated the UI of the feedback page is 3.1 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.53 out of 5.

Article Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Article Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
4	4
3	4
4	5
4	4
4	4
4	4
5	5
4	4
4	3
4	4
3	3
4	4
4	5
4	5
3.933333333	
4.133333333	

The average score users rated the UI of the article page is 3.93 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.13 out of 5.

Social Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Social Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
1	1
3	3
2	4
4	4
4	4
4	4
4	4
3	5
3	4
4	5
4	2
4	5
4	5
4	5
	3.466666667
	3.933333333

The average score users rated the UI of the social page is 3.46 out of 5. Additionally, the average score for how easy it was to navigate through this page is 3.93 out of 5.

Search Food Page:

On a scale of 1 to 5, how would you rate the user interface of this page? (Search Food Page)	How easy was it to navigate through this page? (1 = Very Difficult, 5 = Very Easy)
4	5
4	5
5	5
4	4
4	4
4	4
4	4
4	5
4	4
2	5
4	4
5	4
4	5
4	5
4	5
4	5
	4.533333333

The average score users rated the UI of the search food page is 4 out of 5. Additionally, the average score for how easy it was to navigate through this page is 4.53 out of 5.

Final user interface rating out of 5:

	Login	Home	Feedback	Article	Social	Search Food
Survey 1	2	2.46	2.8	3.2	2.8	3.2
Survey 2	4.13	2.86	3.13	3.93	3.46	4

The results show significant improvements in user interface rating across all pages between the two surveys. For example, the login page score increased from 2 to 4.13 and the search food

page improved from 3.2 to 4. As a result, the different version reflects positive changes in user experiences.

Final ease of navigation rating out of 5:

	Login	Home	Feedback	Article	Social	Search Food
Survey 1	4.33	4.06	4.26	4.13	3.53	4.06
Survey 2	4.46	4.2	4.53	4.13	3.93	4.53

The results show improvements in ease of navigation across all pages between the two surveys.

User Evaluation Involving Representative Stakeholders:

Formative Evaluation

During the early stages of developments, we conducted a formative user evaluation using google forms from the user for NutriTrack application. The goal of this survey was to gather user feedback on the User Interfaces of the current version and what we can improve on. Moreover, it also gathers feedback on the ease of navigation across pages such as login page, social page etc.

The feedback from the survey showed that the user interface particularly on the Login Page required more attention. It received the lowest score out of all the pages with an average rating of 2 out of 5. The user suggested a better design and more user-friendly aesthetics. Social page received similar feedback which also had room for improvement in terms of standardizing colors and design elements.

On the other hand, navigation through the whole application was rated positively across all pages as it is easy for users to travel across different pages. It received an average score of 4 to 5 for ease of usage. It shows that it is relatively straightforward, but the design and interface still needed enhancement.

Summative Evaluation

After implementing most of the changes based on the feedback from Version 1, we conducted a summative evaluation with Version 2 of the NutriTrack using google forms. The goal of this survey is the same as the previous survey, to gather more user feedback on the User Interfaces and what we can improve on. The same user who participated in the first survey will be the one who attends the second survey.

The feedback from the survey showed significant improvements in both user interface and ease of navigation across all the pages. For instance, the Login Page, which previously received an average rating of 2 out of 5, improved to 4.13. As a result, it shows the successful adjustment in the design satisfied the user requirement. Similarly, the Social Page also shows an improvement in user interface with positive feedback on its aesthetics.

On the ease of navigation parts, the whole application rated positive across all pages as it is relatively straightforward for the user.

Conclusion

The formative evaluation using Survey 1 helped us identify critical areas of improvement during the project development phase, especially the interface design. It provides constructive feedback on what we should be improved on. As a result, the summative evaluation using survey 2 demonstrated the effectiveness of iterative changes with the user expressing much better experience with the final product. These evaluations validate the importance of user feedback and meet up the user needs and expectations successfully.

Overall UI Design Justification:

The iterative design approach which is driven by user feedback from google forms ensured that the final user interface not only looked aesthetically pleasing, but it also supports ease of navigation through the whole application. The application is easy to use and able to navigate between pages easily. Based on the user's feedback, we enhance the design focusing the visual coherence of pages, standardizing elements and make the interfaces more approach and appealing to most of the users.

Evidence of Collaboration and Teamwork

Change Request Form

ASP - Change Request

<https://drive.google.com/drive/folders/1cNT1vGIGtizlCT8Ut3ccwCxpeePQqQ7I?usp=sharing>

We have streamlined a change request process to ensure efficiency and accountability within the team. The process starts when a requester submits a change request. Subsequently, the team reviews the request to determine its feasibility and alignment with project goals. Based on the review, the team either approves or rejects the request. This concise procedure ensures that all proposed changes undergo thorough scrutiny to optimize outcomes and resource allocation.

First Change Request:

CHANGE REQUEST FORM	
Change Description	
System Name: ASP_NutriTrack_Team89	
Change Title: Skeleton of all pages	Number: 001
Requested By: Bryan Loo	Date: 06/08/2024
Description of Change: Add skeleton structure (react framework) of all main pages of NutriTrack App to lay out the basic UI components and placeholders for future content and functionality.	
Reason for Change: To establish the foundational structure for the application pages, allowing the team to proceed with content integration and feature development.	
Priority [Circle One]: 1. High 2. Medium 3. Low	
Impact on Deliverables: This change will provide the base skeleton for all pages, which is necessary for integrating further development tasks such as adding functionalities and features. Without this change, developers cannot move forward with page-specific content or features.	
Impact of Not Responding to Change (and Reason Why): If not implemented promptly, it will delay the entire project timeline, as no subsequent development on page content, UI, or functionalities can begin without the basic structure.	
Schedule date and time duration:	2 days

Scope Of Work (SOW)
<ol style="list-style-type: none">1. Create skeleton structures (React frameworks) for all main pages: Home, Articles, Feedback, Food, Social2. Ensure that all navigational links and placeholders for key components are included.3. Provide basic styling to ensure a clean layout for developers to build upon.4. Coordinate with the team to ensure the page skeletons align with the overall UI/UX design.

Roll Back Plan
In case of any issues or design inconsistencies, revert the pages to their original state before the change. Backup all files and commits prior to implementing the skeleton structure to ensure recovery can be easily achieved.

Sign Offs	
[Circle One]: 1. Accepted 2. Deferred 3. Rejected 4. More Info Requested	
Requester Signature: Bryan Loo	Date: 06/08/2024
Reviewer Signature: All Team members	Date: 06/08/2024
Approver Signature: All Team members	Date: 06/08/2024

Git Commit Log

Commit log.xlsx

https://docs.google.com/spreadsheets/d/1_R1UzbFAdM-IQr5TrDtMV5MjfxInvkXd/edit?gid=1499541692#gid=1499541692

We use GitHub for version control, tracking changes through detailed commit logs. This allows us to monitor updates efficiently and ensure transparency in our development process.

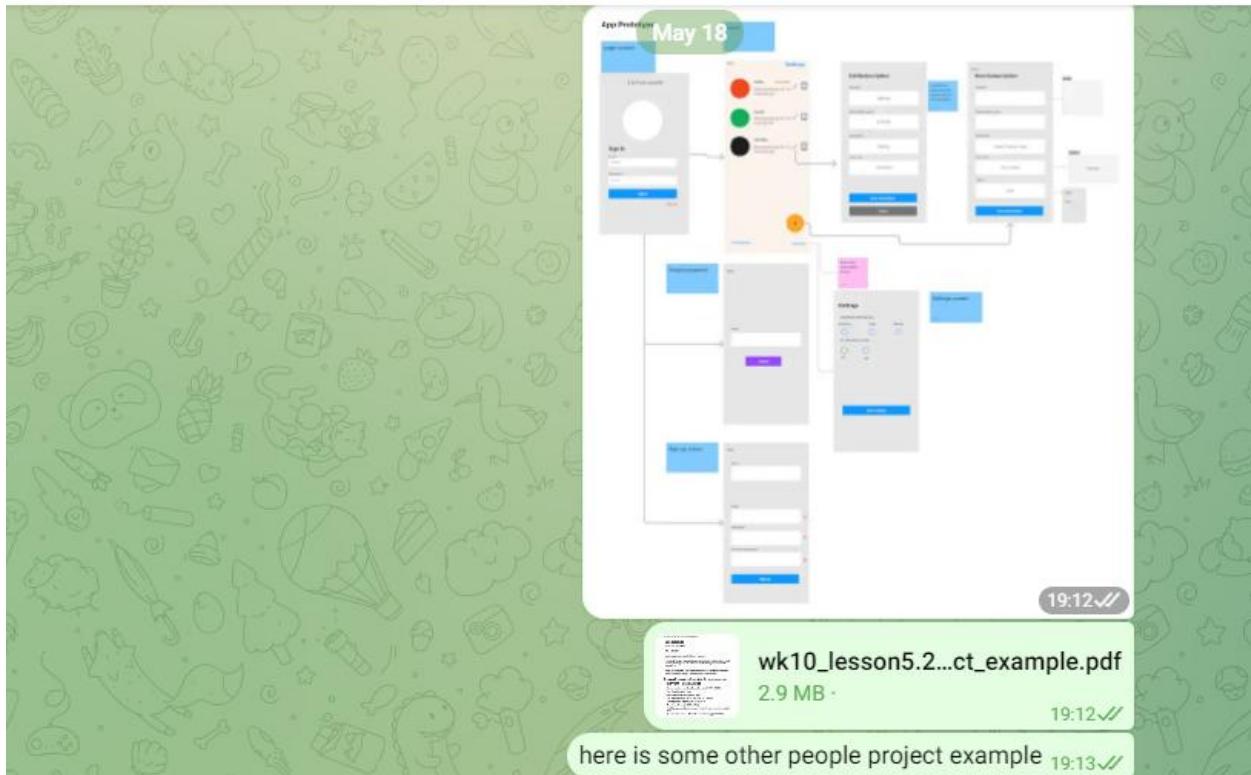
Date	Commit Message
28/8/2024	Updated Home.js; make Footer static, revised a bit css. Article.js: add test cases
28/8/2024	Added error handling for login and signup
28/8/2024	Merge branch 'bryan-updatedComments-home-branch'
28/8/2024	update home page screen if the added comments
28/8/2024	Update README.md
28/8/2024	Merge branch 'main' of github.com:BryanLooo/ASP_NutriTrack_Team89
28/8/2024	Updated Home page UI and calorie tracking features
28/8/2024	fixed bugs in social
26/8/2024	Merge branch 'main' of https://github.com/BryanLooo/ASP_NutriTrack_Team89
26/8/2024	Added comments and small adjustment of the codes
25/8/2024	added theme to social and feedback
25/8/2024	Include a new module, react-native-async-storage to store data. Update Food.js to include store the total data. Update Home.js to display the food nutrition store in search page and display on home.js page
24/8/2024	made minor changes to social
24/8/2024	Update on Food.js page with footer
23/8/2024	Update on Food.js page(search food) with new api. add welcoming message. fix minor errors
20/8/2024	changed logo
18/8/2024	firebaseConfig for feedback
16/8/2024	Update README.md
16/8/2024	Fixed some small errors and updated README instructions
16/8/2024	Integrated Google Sign and User Authentication with Firebase; Refer to readme
16/8/2024	Updated with main files
15/8/2024	add Footer.js component; Articles.js : remove original footer and import from Footer.js module; add ThemeContext.js make toggle theme global; App.js: return ThemeContext; Home.js changed it theme state.. replaced by global state
14/8/2024	reverted to v1 of social
13/8/2024	Articles.js: add like button, display like count. add dark mode for footer
13/8/2024	Articles.js: add comments. add loading effect
12/8/2024	made changes to social page
12/8/2024	Articles.js: add themes button. set diff themes. more articles
11/8/2024	made small changes to feedback pg
11/8/2024	made small changes to the feedback pg
11/8/2024	Update Articles.js
11/8/2024	Updated Article.js, ArticleWebView.js, App.js (add ArticleWebView Screen). Updated dependencies(webview version)
10/8/2024	added some styling and formatting
9/8/2024	added changes to food page
9/8/2024	Merge branch 'main' of github.com:BryanLooo/ASP_NutriTrack_Team89
9/8/2024	added navigation and edited the npm modules
9/8/2024	added login page and full app navigation panel
9/8/2024	Merge branch 'yx-branch'
9/8/2024	Update food.js page
9/8/2024	theme button added and navigation panele updated
8/8/2024	new changes
8/8/2024	remove asp folder
8/8/2024	added feedback and social page
8/8/2024	added project folder
5/8/2024	Add asp/with feedback and social page
25/7/2024	Initial commit

Team Communication History

We have established a group on Telegram to facilitate communication, delegate tasks, and schedule meetings, streamlining our coordination and workflow efficiently.

Conversation Example:

Sharing Prototype Example:



First Meeting:

<https://meet.google.com/ver-zimn-peg> May 21

Google Real-time meetings by Google. Using your browser, share your video, desktop, and presentations with teammates and customers.

18:21✓

@Boovanuhh 18:23✓

Hi we just need come some question for google form (eg, how can we improve the matching, what feature you like the most, do u think the ui easy to use, how old are you (for choosing target audience)) 18:49✓

@its_bryann @chanyuxiang 18:49✓

just finish by tmr night 18:50✓

Agile_project specification.pdf
103 KB ·

Grading 18:58✓

Chan Yu Xiang

jagger

Hi we just need come some question for google form (eg, how can we i...
What type of question? Is it for improving the software like u mention above or like how u wan the software to be builded. 19:43

Second meeting and finalized our idea:



First Survey:

Bryan Loo

Hey guys, I just created a google forms survey to gather feedback about the calorie tracking application. Please share this link to as many people as possible so that we have a good pool of people to work with for our report and analysis. Here is the google forms link:

<https://forms.gle/G8zHHNg94pxW4KvW7>

Google Docs

Calorie Tracking Survey

This calories tracking survey is created to gather feedback from users to help improve on current calories and health applications to encourage indivi...

Calorie Tracking Survey

This calorie tracking survey is created to gather feedback from users to help improve on current calories and health applications to encourage individuals to have healthy lifestyle. This is done by observing calories and the nutritional benefits of the ingredients that we are consuming on a daily basis.

* Indicates required question

1. Do you find the features of your current calorie tracking application useful? *

Yes, the features are useful

No

2. Do you think there are features missing from your current calorie tracking application? *

22:17

no problem 22:17✓

thanks man 22:18✓

Second Survey:

July 3
https://docs.google.com/forms/d/e/1FAIpQLSdAFPOkWLrZYW1XQKTnquaMkLQsI_7xTs-JDVZjHRH5ySxYFA/viewform?usp=pp_url

Google Docs

NutriTrack - Calorie Tracking Prototype Survey

NutriTrack - Calorie Tracking Prototype Survey

Thank you for participating in our survey! Your feedback is crucial in helping us improve the NutriTrac...

NutriTrack - Calorie Tracking Prototype Survey

NutriTrack - Calorie Tracking Prototype Survey

Thank you for participating in our survey! Your feedback is crucial in helping us improve the NutriTrack app. This survey focuses on the user interface and overall user experience of our calorie tracking prototype. Please answer the questions based on your experience with the prototype.

* indicates required question

1. On a scale of 1 to 5, how would you rate the user interface of this prototype?

■ NutriTrack
(0 items)

22:45✓/ will add more questions when social and article prototype is done

will add more questions when social and article prototype is done

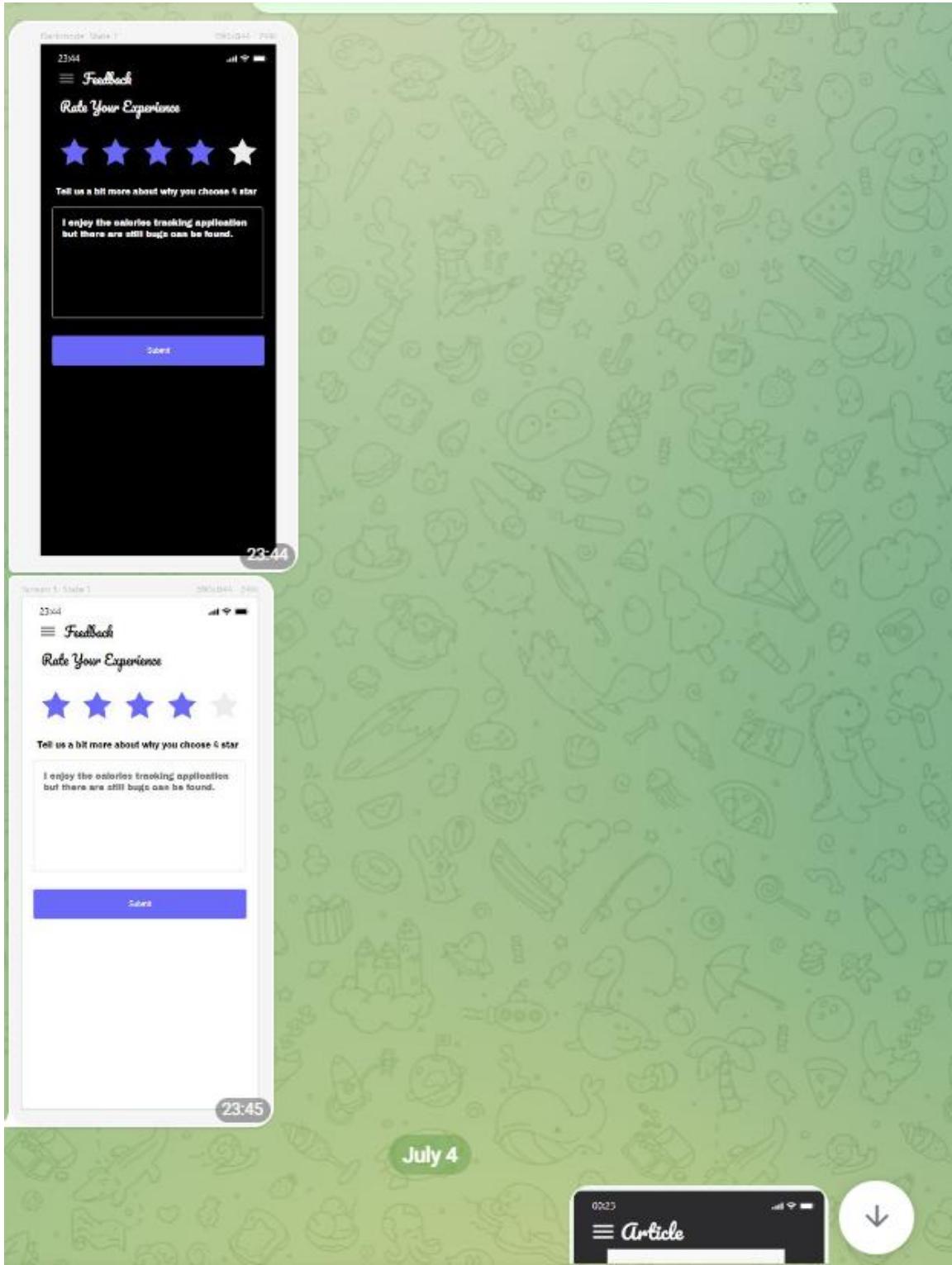
Bryan Loo
jagger

looks good bro

Creating poster for both surveys and prototype 22:47✓/ SCAN ME
SCAN ME
NUTRI TRACK
Please help us improve our app by filling out this survey!

23:22✓/ NutriTrack update

Sharing our prototype:



Git Instruction:

Bryan Loo

Step 1 (Clone the repository)

July 22

Command: git clone

<git@github.com:BryanLoooo/sampleproject.git>

Step 2 (redirect to the folder that is the same as the git repo file name)

Command: e.g cd sampleproject

Step 3 (Create a new branch)

Command: git checkout -b first-branch

or if the branch already exists:

Command: git checkout first-branch

Step 4 (Stage the changes)

Command: git add .

Step 5 (after staging commit the changes to the first-branch)

Command: git commit -m "added index html page"

Step 6 (push the first-branch to GitHub, this does not include the changes to the branch)

Command: git push origin first-branch

Step 7 (redirect back to the main branch)

Command: git checkout main

Step 8 (update the main branch onto your local system to the latest version)

Command: git pull origin main

Step 9 (you are now still on the main branch, merge the first-branch with the main)

Command: git merge first-branch

Step 10 (after merging first-branch and main, now there is only the main branch to worry about)

Command: git push origin main

Created a notion workspace for us to collaborate and track each other's progress

July 24

Bryan Loo
<https://www.notion.so/team/da12f0e4-ce95-40e1-8cf5-e0987b0711ba/join>

Notion
Notion – The all-in-one workspace for your notes, tasks, wikis, and databases.
A new tool that blends your everyday work apps into one. It's the all-in-one workspace for you and your team



Notion
The all-in-one workspace.
Notes, tasks, wikis, & databases.



00:21

Hey guys I created a notion workspace for us to collaborate and track each other's progress on the pages we are assigned 00:21

lmk if there are any issues with joining this workspace 00:21



as a requirement for the final report, we need evidence of collaboration and teamwork 00:22

July 25

Bryan Loo
Hey guys, I just created the GitHub repo for version control.
Here is the link:

https://github.com/BryanLoooo/ASP_NutriTrack_Team89.git

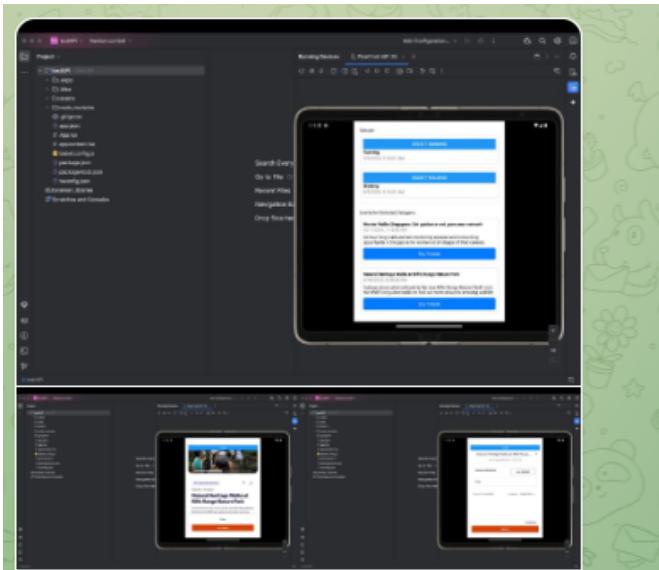
GitHub
GitHub - BryanLoooo/ASP_NutriTrack_Team89: NutriTrack provides daily calorie tra...
NutriTrack provides daily calorie tracking and suggests recommended exercises for various kinds of users - GitHub - BryanLoooo/ASP_NutriTrack_Team89:...

BryanLoooo/
ASP_NutriTrack_Team89



NutriTrack provides daily calorie tracking and suggests recommended exercises for various kinds

Faced technical challenges and discussed:



hi guys, just wanted to update on an issue I ran with social group page. I read and tried a few apis but due to permissions or paid api key problems, I went with the Eventbrite api, which seemed to be the most suitable and accessible

but right the problem with this api is that the reference to display every event related to a specific category is deprecated, so for the time being, only way around it is to hard code maybe like 3-5 events per category if available, using specific event IDs instead, I tried that and right now, it kinda looks like this, sooo lmk if we can proceed with this method, though not ideal, but kinda stuck because there aren't any other suitable api that I could find, or if there is any other way, lmk also!

01:25

Bryan Loo



Boovana

Album, hi guys, just wanted to update on an issue I ran with soci...

No worries boovana, I think for now hard coded content is fine since we are working on version 1 as of now



B

10:43

I feel that working with deprecated api can be a bit risky 10:44

Especially when the prof is marking our code 10:44



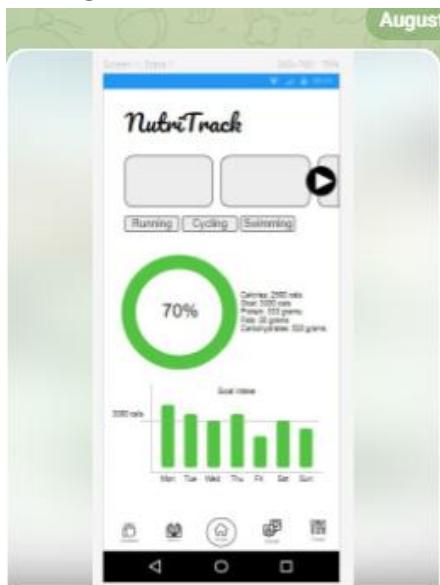
Boovana

Album, hi guys, just wanted to update on an issue I ran with soci...

But the user interface looks pretty good

10:45

Getting feedback from each others:



This is the home page template that I just created 00:46

let me know what yall think abt the navigation and the overall feel 00:46

then I can send the measurements and the icons so we can copy this to maintain to the rest of the pages 00:47

looks good 00:48✓

Bryan Loo
okay I will send the icons to you guys 00:48

and the font style of "nutritrack" 00:50✓

Bryan Loo
hmmmm that one I not sure eh 00:50

what font do you think is good 00:50

or are you okay with this current font style? 00:50

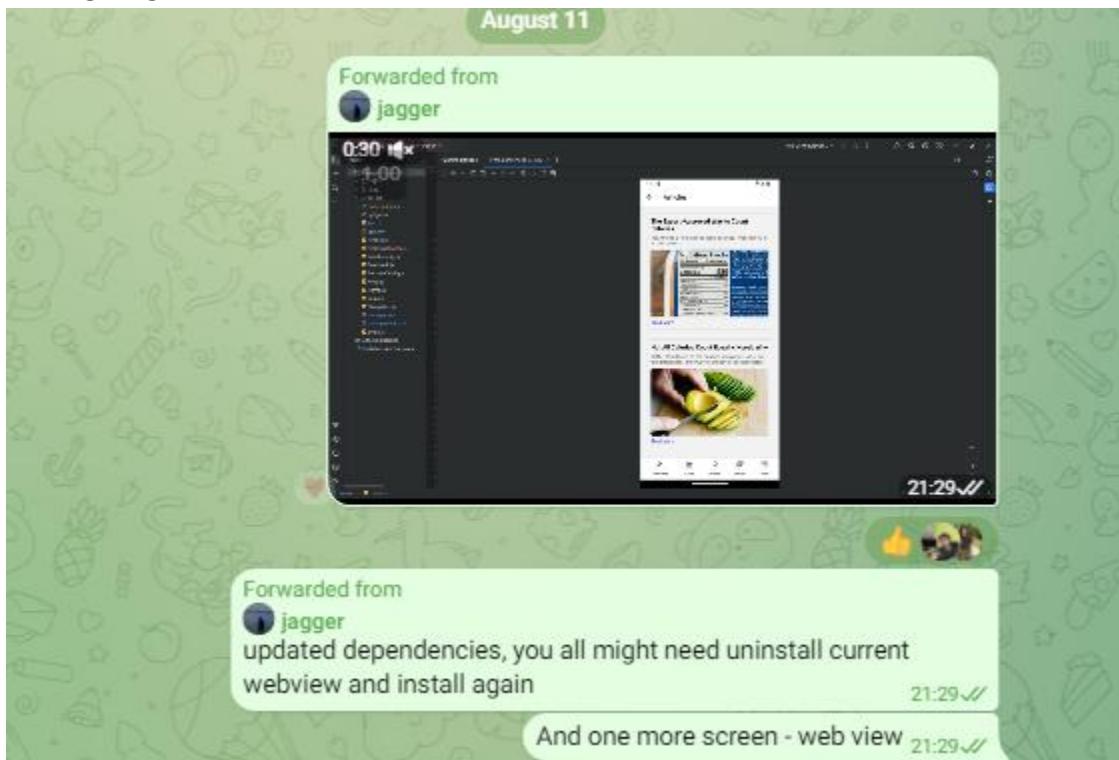
Current one 00:51✓

Or we can make it our logo, you ss it send us png or jpeg 00:51✓

Bryan Loo

Navigation Icons.zip
zip 15 KB 00:52

Sharing Progress:



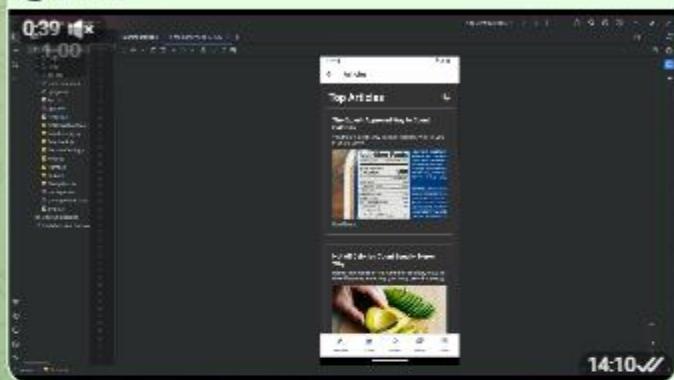
August 12

update:
themes button, more articles.

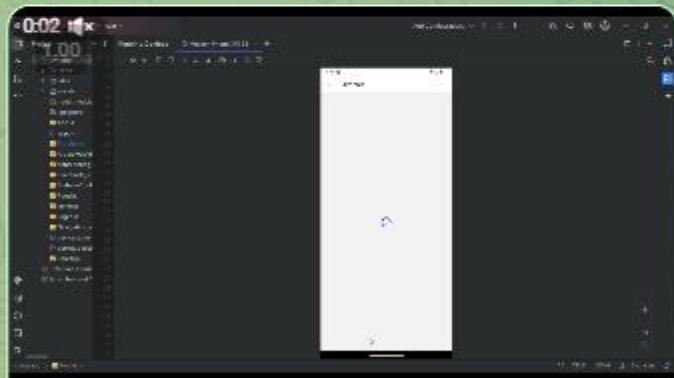
note:
set themes toggle is not global 14:10✓

Forwarded from

jagger



August 13



update:
loading effect

hey guys we can take this current version of the application for user testing for our report



01:05

Delegating Tasks and sharing progress:

Bryan Loo

Things to do by tomorrow (14/08):

- 1) Think of some of the improvements for version 2 of NutriTrack
- 2) ERD structure and necessary table columns for your pages
- 3) Some useful user interactions for users

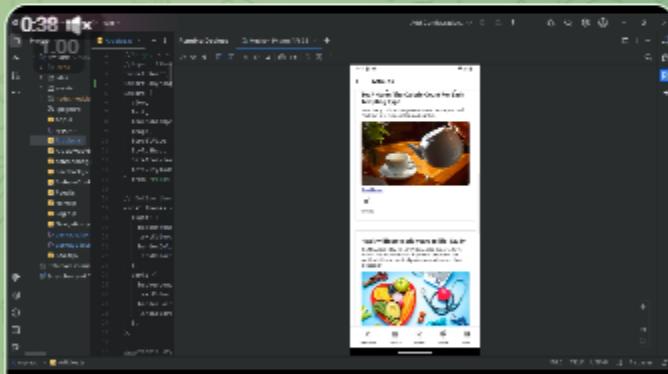
Things to do by this Friday (16/08)

- 1) Finish up on your page for version 1
- 2) After finishing screenshot and update the final report google docs

React to this message after reading.



11:22



update:

add like button (once user registration is done, will integrate with it: set one user can only press once)

like count store in user machine currently, might store in db in future, see how it goes first.

version 2:

update all footer icon, display two modes: light & dark

Sharing Information:

August 19

Hey guys,

As you may recall, during our last meeting, I expressed concerns about managing dependencies in our project. Initially, it seemed that having each member focus on their own page would minimize dependency conflicts. However, we've continued to see issues arise.

To address this, I'd like to introduce the use of package-lock.json in our workflow. This file ensures that we all use the exact same versions of npm packages, thereby preventing discrepancies that can lead to unexpected behavior or bugs in our application.

Please let me know if you have any questions or need assistance integrating this into your development setup. I believe this will help us maintain a more stable and consistent environment across our project.



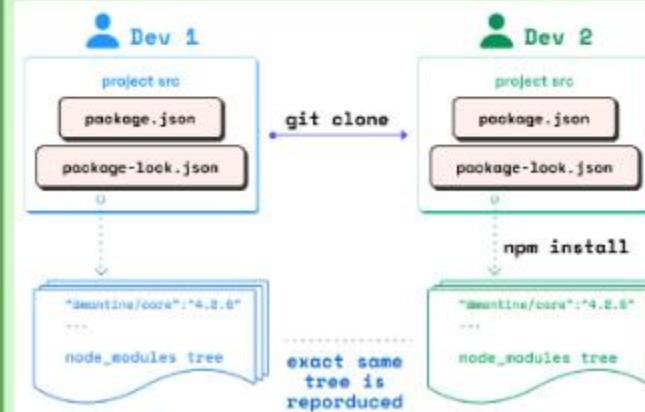
13:10✓

<https://medium.com/pavesoft/package-lock-json-the-complete-guide-2ae40175ebdd>

Medium

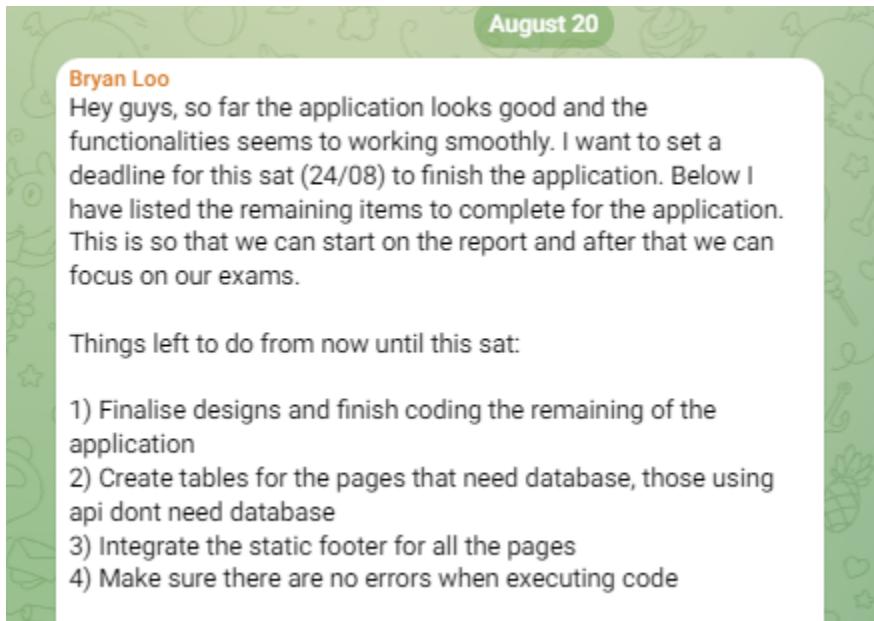
package-lock.json: The Complete Guide

What is package-lock.json? And why should you care?

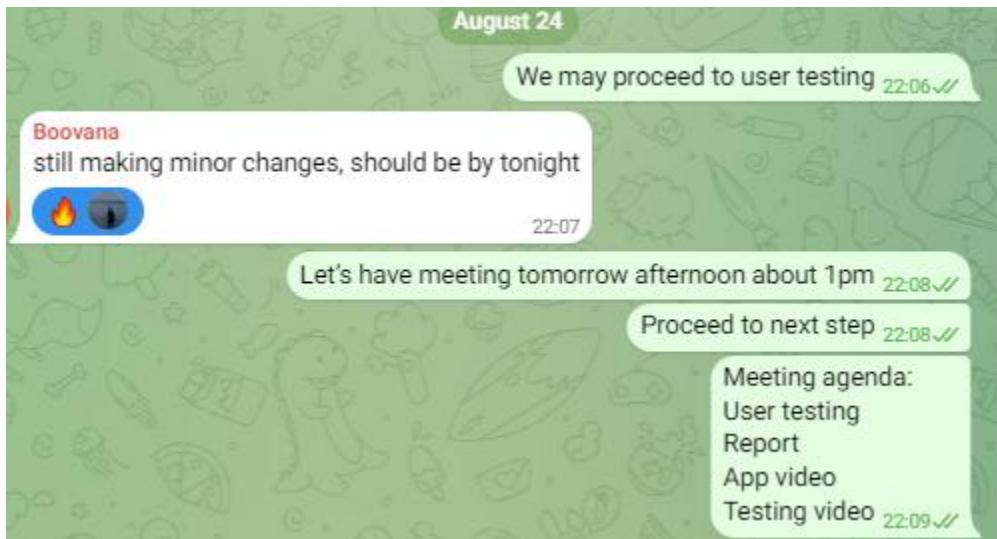


13:10✓

Delegating Tasks:



Scheduling Meeting:



Meeting Summary:

August 25

<https://reloadux.com/blog/ui-ux-design-evaluation-methods-for-usability-testing/>

ReloadUX
UI UX Design evaluation methods for usability testing – The Ultimate Guide – 2023 - reloadux
Master UI UX design evaluation methods for usability testing. Follow our ultimate guide to ensure your designs meet user needs and expectations.



13:54 //

meeting summary:

- 1.add comment to code
- 2.add review to each commit (github comment)
- 3.add error handling (return http status, due Wednesday)
- 4.testing (I will do survey, please send each version screenshot, preferably video to notion, due tomorrow)
- 5.test own page (console log, due Wednesday)

15:04 //

Delegating Tasks:

September 3

@chanyuxiang :
Survey & response - Analysis
I. UI evaluation
II. User Evaluation stakeholders summative & formative
III. Overall design Justification

@Boovanuhh
Iterative design for every page (eg; flow chart)

@jagger
Evidence:
Change request form for every changes
Github commit
Discord & telegram chat history

@its_bryann
Update gantt chart with explanation
delegate role;(eg project manager with explanation)

@BCW_Loo24
Discussion of literature (reference and cite)
Problem analysis & argumentation (reference and cite)
User guide(video and readme)
Evaluation of work in relation to original plan
conclusion and discussion of future work

@everyone
Technical challenge(what have you faced, how you solved),
explain code for their own page(in code it self, add as comment,
send screenshot)

  00:03✓/

finish by Saturday morning, following with a final meeting to review 00:04✓/

Testing

Login Page:

Test Case ID	Test Scenario	Steps
TC001	Successful login	1. Enter registered email address and password 2. Click login button 3. Redirected to application's home page
TC002	Successful login (Google)	1. Click "Sign in with Google" button 2. Give permissions 3. Redirected to application's home page
TC003	Unsuccessful login	1. Enter wrong email address or password 2. Click login button 3. Error message displayed in the application and console
TC004	Successful registration	1. Enter details in the right format 2. Click sign up button 3. Redirected to application's home page
TC005	Unsuccessful registration (wrong email format)	1. Enter email in wrong format for registering 2. Click sign up button 3. Error message displayed in the application and console
TC006	Unsuccessful registration (weak password)	1. Enter password that is shorter than 6 characters 2. Click sign up button 3. Error message displayed in the application and console
TC007	Unsuccessful registration (same email used)	1. Enter email that has already been used 2. Click sign up button 3. Error message displayed in the application and console

Test Case ID	Expected Result	Actual Result	Status
TC001	Successful login	Matched the expected result	Pass
TC002	Successful login	Matched the expected result	Pass
TC003	Unsuccessful login	Matched the expected result	Pass
TC004	Successful registration	Matched the expected result	Pass
TC005	Unsuccessful registration	Matched the expected result	Pass
TC006	Unsuccessful registration	Matched the expected result	Pass
TC007	Unsuccessful registration	Matched the expected result	Pass

Test Case 1: Successful Login

The screenshot shows a Windows desktop environment with several open windows:

- VS Code (Left):** The code editor displays the `src/App.js` file for a React Native application named "NUTRITRACK". The code handles sign-up logic, including error handling for email and password.
- Android Emulator (Right):** An emulator window titled "Android Emulator - Resizable_Experimental_API_345534" shows the "Home Page" of the app. It features a header with a fruit icon and the text "NUTRITRACK". Below the header are two cards: "How to eat healthy" (with an apple icon) and "Regular exercise helps" (with a person running icon). A green button at the bottom right says "+ Add post".
- Terminal (Bottom Left):** The terminal window shows logs from the Expo development client, indicating successful bundle compilation and a warning about calorie consumption being outside the recommended range.
- Status Bar (Bottom):** The status bar includes icons for battery, signal, and network, along with the text "Activate Windows Go to Settings to activate Windows."

Test Case 2: Successful Login (Google)

Test Case 3: Unsuccessful Login

← LOGIN



Hey,
Welcome
Back

Forgot Password?

Login

Invalid email or password

or continue with

Sign in with Google

Don't have an account? [Sign up](#)

[REDACTED]

ERROR Login Error: [Error: [auth/invalid-email] The email address is badly formatted.]
ERROR Login Error: [Error: [auth/invalid-credential] The supplied auth credential is incorrect, malformed or has expired.]

Test Case 4: Successful Registration

The screenshot shows the 'Home Page' of the Nutritrack app. The page features a header with the app's logo and navigation icons. Below the header, there are two cards: one for 'How to eat healthy' (apple icon) and one for 'Regular exercise helps' (person running icon). A green button labeled '+ Add post' is located below the cards. Under the cards, a section titled 'Today's progress:' displays nutritional goals and intake for three days: Monday, Tuesday, and Wednesday. The nutritional data includes Calorie intake (0), Goal (3000 cal), Protein (0g), Fat (0g), and Carbohydrates (0g). The footer bar contains various icons for account management and settings.

```

project > my-app > SignupScreens > SignupScreen.jsx
const handleSignup = () => {
  const handleSignup = async () => {
    // Close the error
    setError("");
    // Navigate to the main app screen after successful signup
    navigation.navigate("Login");
    if (error) {
      console.error(`Signup Error: ${error}`);
      if (error.code === "auth/email-already-in-use") {
        setError("Email already in use");
      } else if (error.code === "auth/invalid-email") {
        setError("Invalid email");
      } else if (error.code === "auth/weak-password") {
        setError("Password should be at least 6 characters");
      } else {
        setError(error.message);
      }
    }
  };
  const handleGoBack = () => {
    navigation.goBack();
  };
  const handleLogin = () => {
    navigation.navigate("Login");
  };
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

- LOG Signup Successful
- LOG Home screen page has been executed successfully
- LOG Calories for the day are in the safe range. User can continue consuming calories.
- LOG Current nutrients information back to 0
- LOG Retrieving total nutrients information
- LOG Fetching total nutrients information using focus
- LOG Retrieving total nutrients information
- LOG Successfully reset nutrients information
- LOG Current nutrients information is all set to 0. Please add food items to see updated information
- LOG Current nutrients information is all set to 0. Please add food items to see updated information

Activate Windows
Go to Settings to activate Windows.

Ln 89, Col 8 Spaces: 2 UTT-B CRLF () JavaScript JSX ⚡ Go Live ⌂ Current tab ⌂ Prettier ⌂

Test Case 5: Unsuccessful Registration (wrong email format)

← SIGNUP



Let's get
started



test4



123456



12345678

Sign up

Invalid email

or continue with

Sign up with Google

Already have an account! [Login](#)

ERROR Signup Error: [Error: [auth/email-already-in-use] The email address is already in use by another account.]
ERROR Signup Error: [Error: [auth/weak-password] The given password is invalid. [Password should be at least 6 characters]]
ERROR Signup Error: [Error: [auth/invalid-email] The email address is badly formatted.]

Test Case 6: Unsuccessful Registration (weak password)

← SIGNUP



Let's get
started

test4@gmail.com

12345



12345678

Sign up

Password should be at least 6 characters

or continue with

Sign up with Google

Already have an account! Login



```
[ERROR] Signup Error: [Error: [auth/email-already-in-use] The email address is already in use by another account.]
[ERROR] Signup Error: [Error: [auth/weak-password] The given password is invalid. [ Password should be at least 6 characters ]]
[ERROR] Signup Error: [Error: [auth/invalid-email] The email address is badly formatted.]
```

Test Case 7: Unsuccessful Registration (email in use)

The screenshot shows a mobile application's sign-up screen. At the top, there is a back arrow labeled "SIGNUP". Below it, the text "Let's get started" is displayed. There are three input fields: the first contains "test@gmail.com", the second contains a password starting with ".....", and the third contains "12345678". A large "Sign up" button is centered below the inputs. A red error message "Email already in use" is displayed above the button. Below the button, there is a link "or continue with" followed by a "Sign up with Google" button. At the bottom, there is a link "Already have an account! Login". A black bar at the very bottom displays three error logs:

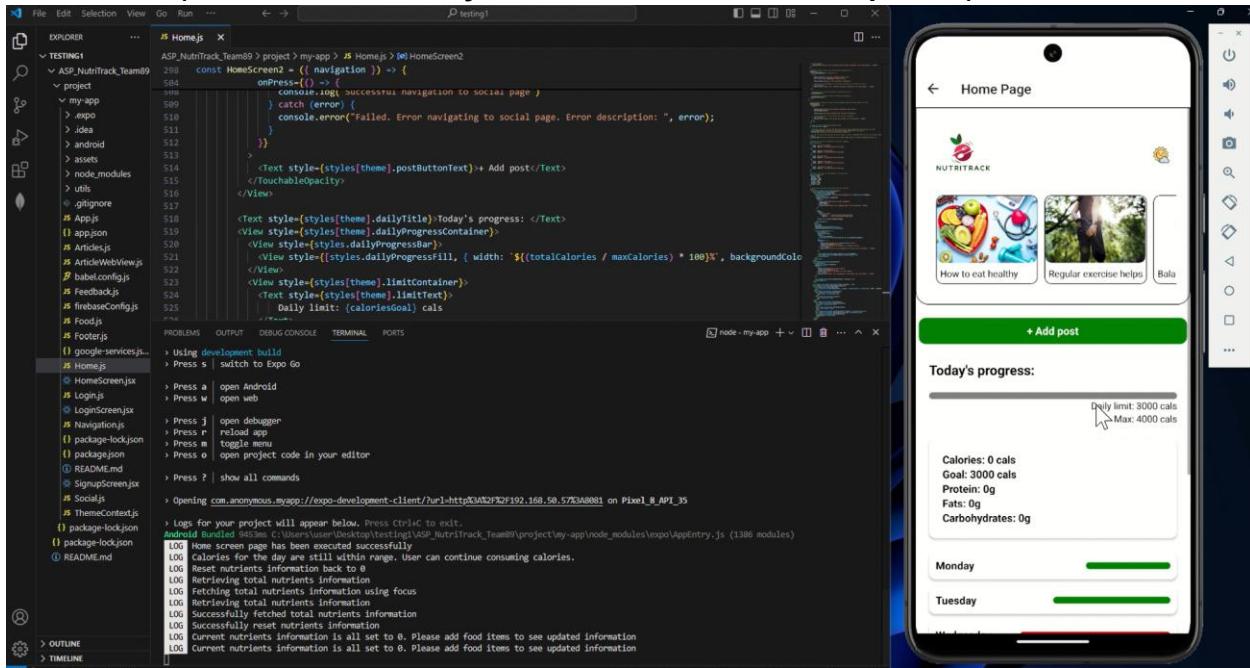
```
ERROR Signup Error: [Error: [auth/email-already-in-use] The email address is already in use by another account.]
ERROR Signup Error: [Error: [auth/weak-password] The given password is invalid. [ Password should be at least 6 characters ]]
ERROR Signup Error: [Error: [auth/invalid-email] The email address is badly formatted.]
```

Home screen page:

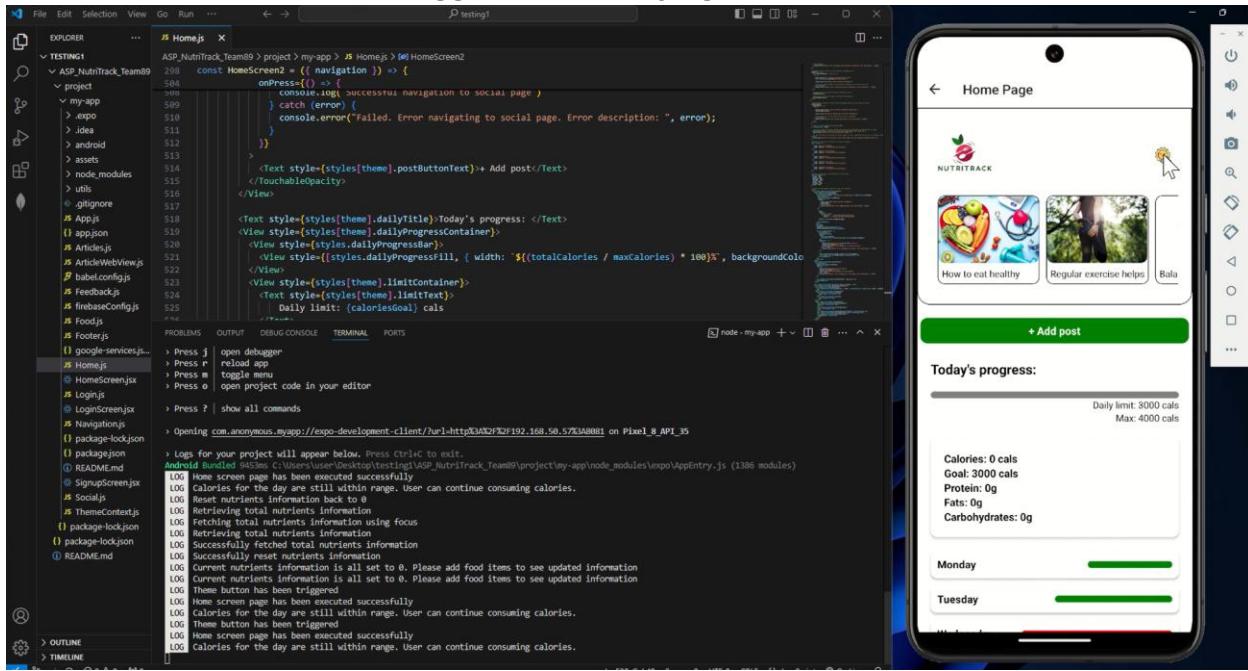
Test Case ID	Test Scenario	Steps
TC001	Succesfully renders HomeScreen2 component	Step 1. Click login button Step 2. Insert email and password Step 3. Click login button to insert Step 4. Display the home screen
TC002	Theme toggle functionality - Light Theme	Step 1. Locate the theme icon button Step 2. Click on the theme button Step 3. Notice app change to light theme Step 4. Icon image change to sun
TC003	Theme toggle functionality - Dark Theme	Step 1. Locate the theme icon button Step 2. Click on the theme button Step 3. Notice app change to dark theme Step 4. Icon image change to moon
TC004	Navigate to Articles screen	Step 1. Locate the list of articles Step 2. Browse the list of articles Step 3. Click on the selected article Step 4. Redirect to article screen
TC005	Navigate to Social screen	Step 1. Locate the add post button Step 2. Click on the add post button Step 3. Redirect to social screen
TC006	Daily calorie goal reached alert	Step 1. Locate the food navigation icon Step 2. Redirect to search food screen Step 3. Search for food item Step 4. Click on save food button Step 5. Navigate back to home and notice calorie reached alert
TC007	Daily calorie goal not reached	Step 1. Locate the food navigation icon Step 2. Redirect to search food screen Step 3. Search for food item Step 4. Click on save food button Step 5. Notice the daily calorie progress bar get updated with the new calories
TC008	Fetch total nutrients information - Success	Step 1. Locate the food navigation icon Step 2. Redirect to search food screen Step 3. Search for food item Step 4. Click on save food button Step 5. Notice the daily calorie progress bar get updated with the new calories Step 6. Notice the macronutrients data is updated with the new food item
TC009	Fetch total nutrients information - No data	Step 1. Navigate to any other page in the app Step 2. Navigate back to home screen Step 3. Notice that the macronutrients and daily calorie count is still set at 0
TC010	Reset nutrients information	Step 1. Click login button Step 2. Insert email and password Step 3. Click login button to insert Step 4. Display the home screen Step 5. Notice that the macronutrients and calorie count is reset to 0

Test Case ID	Expected Result	Actual Result	Status
TC001	Successfully renders HomeScreen2 component	Matched the expected result	Pass
TC002	Theme toggle functionality - Light Theme	Matched the expected result	Pass
TC003	Theme toggle functionality - Dark Theme	Matched the expected result	Pass
TC004	Navigate to Articles screen	Matched the expected result	Pass
TC005	Navigate to Social screen	Matched the expected result	Pass
TC006	Daily calorie goal reached alert	Matched the expected result	Pass
TC007	Daily calorie goal not reached	Matched the expected result	Pass
TC008	Fetch total nutrients information - Success	Matched the expected result	Pass
TC009	Fetch total nutrients information - No data	Matched the expected result	Pass
TC010	Reset nutrients information	Matched the expected result	Pass

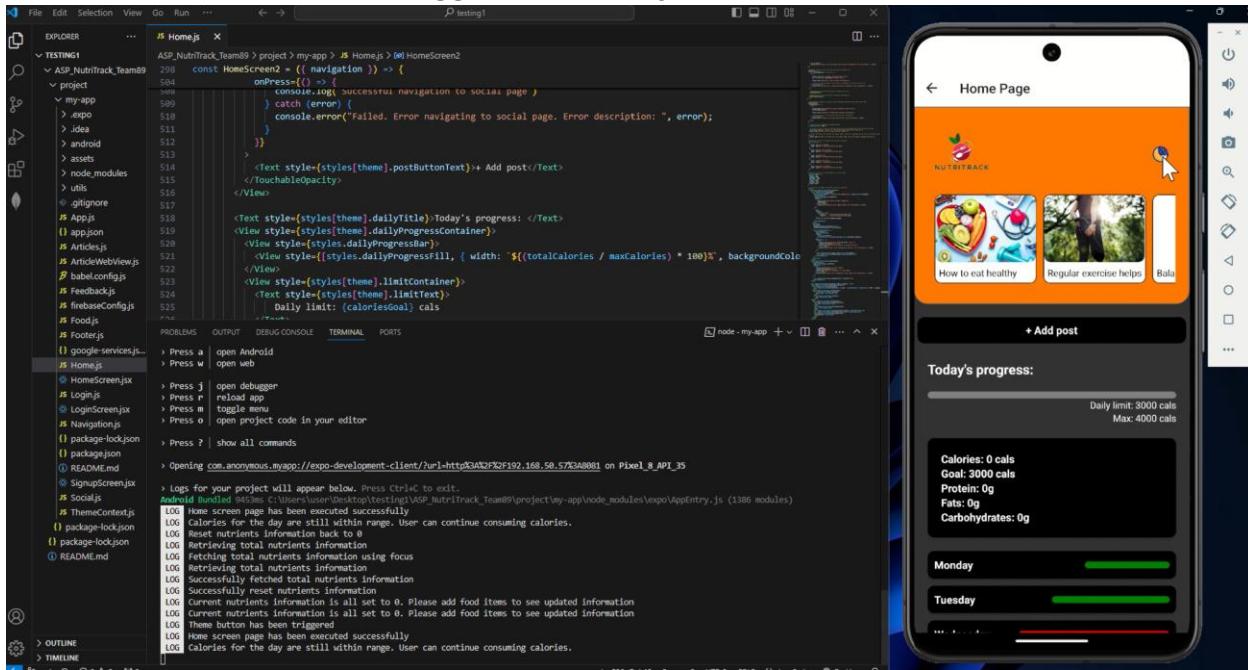
Test Case 1 (TC001 - Successfully renders HomeScreen2 component)



Test Case 2 (TC002 - Theme toggle functionality light theme)



Test Case 3 (TC003 - Theme toggle functionality dark theme)



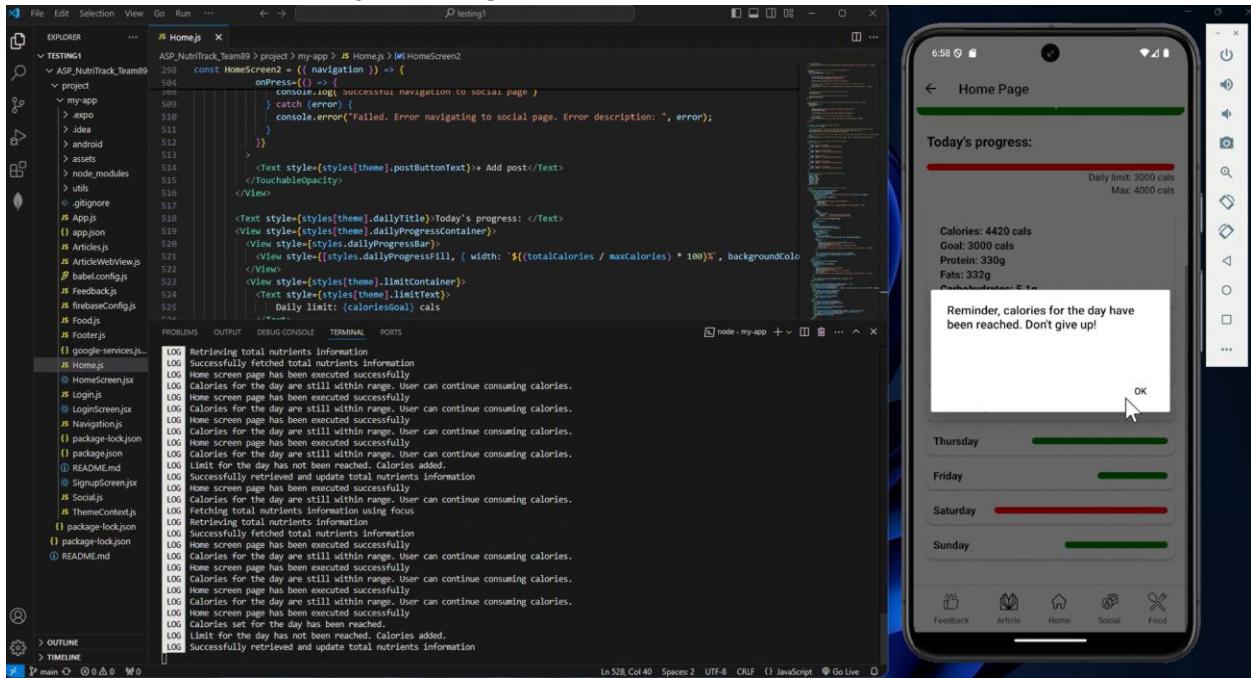
Test Case 4 (TC004 - Navigate to Articles screen)

The screenshot shows a mobile application interface. On the right is a smartphone displaying the 'Articles' screen. The screen has a header 'Articles' with a back arrow. Below it is a section titled 'Top Articles' with a single card. The card features the title 'The Expert-Approved Way to Count Calories', a 'Nutrition Facts' box with details like 'Calories 580', and a 'Read More' button. At the bottom of the card are social sharing icons. On the left, a laptop displays the expo testing interface with the code for the 'Home.js' file. The code includes logic for navigating to the social page if successful or handling errors. The terminal tab shows logs indicating a successful navigation to the selected article.

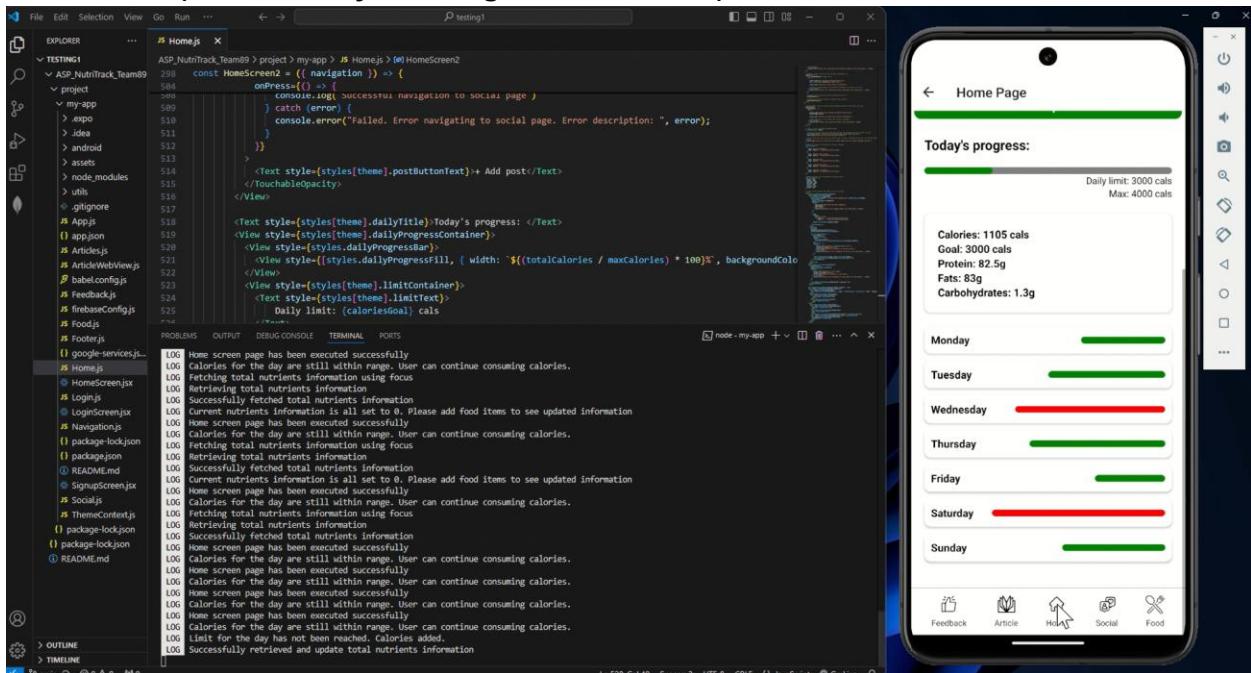
Test Case 5 (TC005 - Navigate to Social screen)

The screenshot shows a mobile application interface. On the right is a smartphone displaying the 'Social' screen. The screen has a header 'Social' with a back arrow. Below it is a welcome message: 'Welcome to NutriTrack's social page, a space where you can share and explore the health journeys of fellow community members!'. At the bottom is a blue 'Add Post' button. On the left, a laptop displays the expo testing interface with the code for the 'Home.js' file. The code includes logic for navigating to the social page if successful or handling errors. The terminal tab shows logs indicating a successful navigation to the social page.

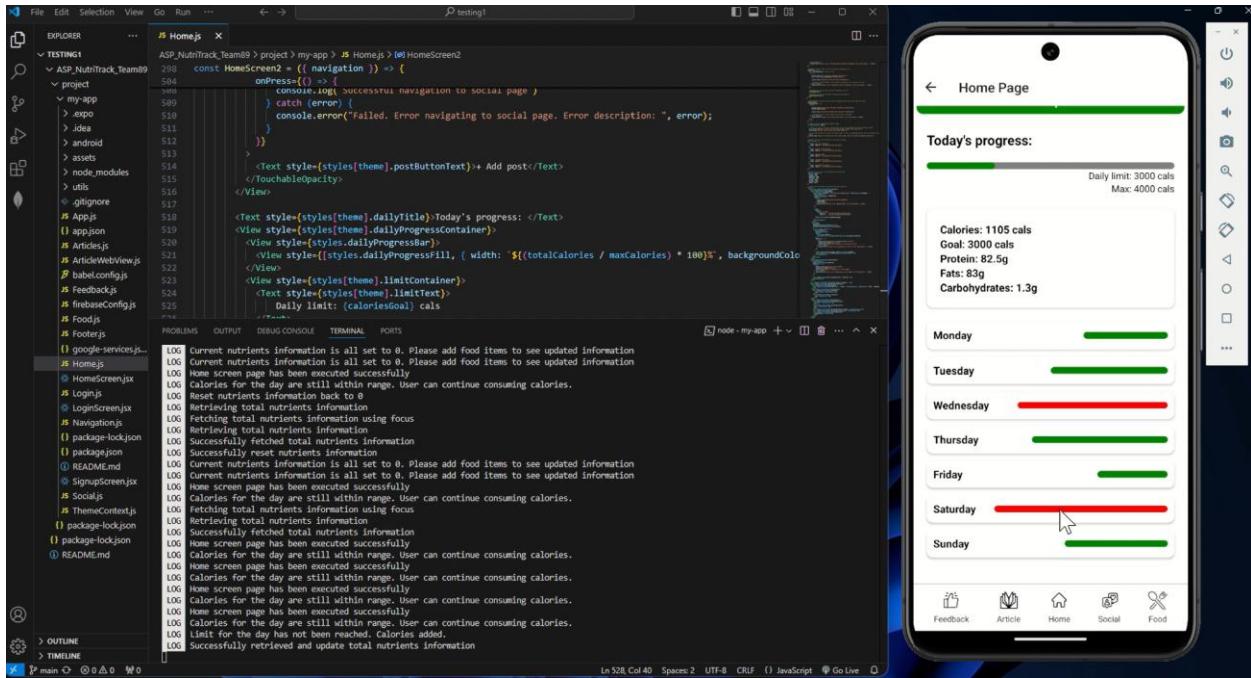
Test Case 6 (TC006 - Daily calorie goal reached alert)



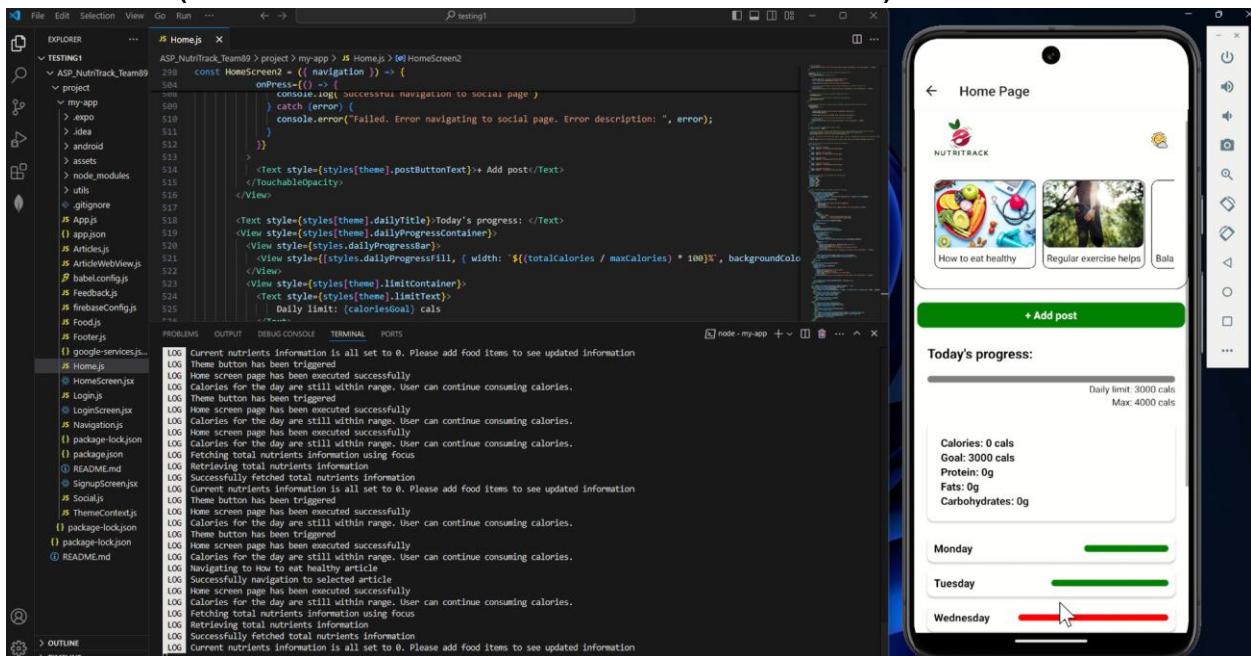
Test Case 7 (TC007 - Daily calorie goal not reached)



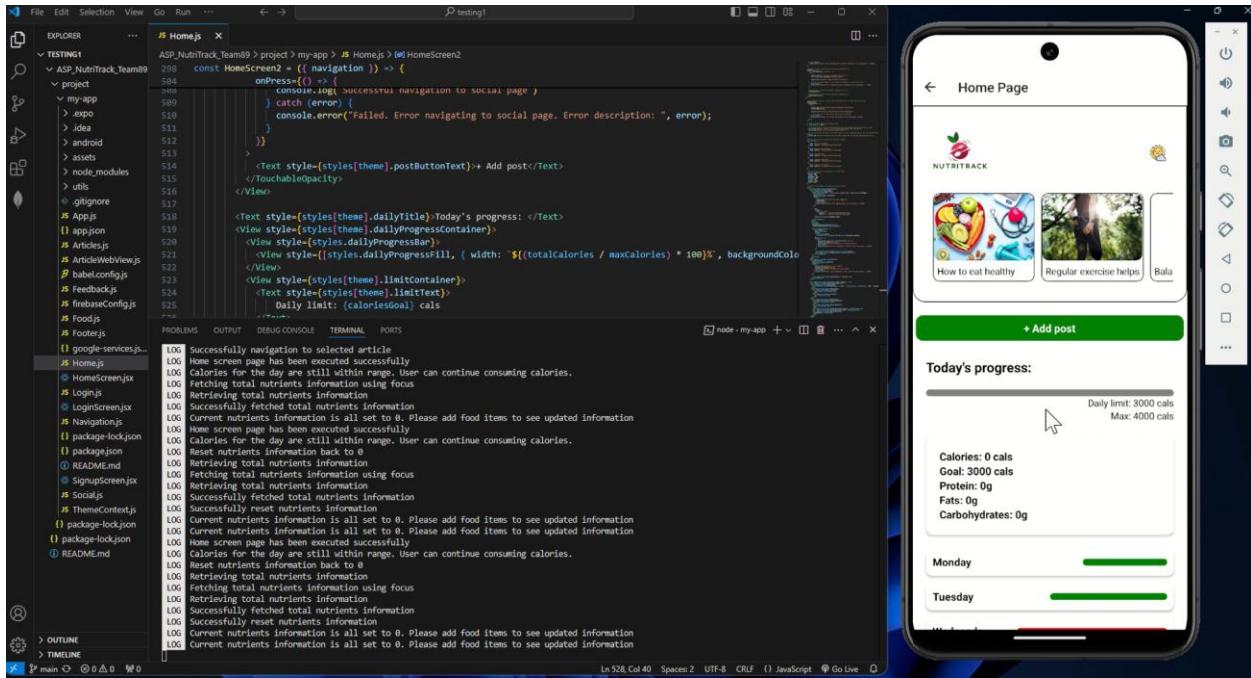
Test Case 8 (TC008 - Fetch total nutrients information - Success)



Test Case 9 (TC009 - Fetch total nutrients information - no data)



Test Case 10 (TC010 - Reset nutrients information)



Feedback Page

Test Case ID	Expected Result	Actual Result	Status
TC001	The page should load with subtitle, feedback input and star rating system submit button	Matched the expected result	Pass
TC002	Theme toggle functionality - Light & dark theme	Matched the expected result	Pass
TC003	Page should be able to submit feedback and rating	Matched the expected result	Pass
TC004	Thank you message displayed when submit button is pressed	Matched the expected result	Pass
TC005	Error message should be displayed if submit button is pressed with only the feedback and without the rating	Matched the expected result	Pass
TC006	Error message should be displayed if submit button is pressed with only the rating and without the feedback	Matched the expected result	Pass

Test Case ID	Test Scenario	Steps
TC001	The page should load with subtitle, feedback input and star rating system submit button	<ol style="list-style-type: none"> 1. Login using credentials 2. Navigate to feedback page 3. Verify if all the components are visible
TC002	Theme toggle functionality - Light & dark theme	<ol style="list-style-type: none"> 1. Locate the theme icon located on right corner of the page. 2. Click on the sun icon 3. Verify if dark mode is applied to the screen while the icon changes to a moon icon 4. Click on the moon icon 5. Verify if light mode is applied, while moon icon switched to sun
TC003	Page should be able to submit feedback and rating	<ol style="list-style-type: none"> 1. Enter feedback and choose number of stars to rate, eg. Good app, 4 stars 2. Click on submit button 3. Page navigates to display a thank you message
TC004	Thank you message displayed when submit button is pressed	<ol style="list-style-type: none"> 1. Enter feedback and choose number of stars to rate, eg. Good app, 4 stars 2. Click on submit button 3. Page navigates to display a thank you message
TC005	Error message should be displayed if submit button is pressed with only the feedback and without the rating	<ol style="list-style-type: none"> 1. Enter feedback in the input field 2. Click on submit button. 3. Verify if error message is displayed in red
TC006	Error message should be displayed if submit button is pressed with only the rating and without the feedback	<ol style="list-style-type: none"> 1. Click on preferred number of stars only 2. Click on submit button. 3. Verify if error message is displayed in red

Test case TC001 - The page should load with subtitle, feedback input and star rating system submit button

The screenshot shows a developer's environment with an open project in the Explorer sidebar. The current file is `Feedback.js`, which contains logic for handling user feedback submission. To the right, an Android emulator displays the `Feedback` screen. The screen has a subtitle "Please rate your experience and drop your honest review of the app to let us know how we can improve!", a text input field labeled "Enter your feedback", a star rating system with five stars, and a blue "SUBMIT FEEDBACK" button.

Test case TC002 - Theme toggle functionality - Dark theme

The screenshot shows the same developer environment and Android emulator setup as the previous test case, but with a dark theme applied. The code in `Feedback.js` remains identical, but the Android emulator now displays the `Feedback` screen in a dark mode theme. The background is dark, and the text and UI elements are white or light-colored to match the theme.

Test case TC002 - Theme toggle functionality - Light mode

The screenshot shows a developer's workspace with a code editor and a mobile application running on an emulator. The code editor displays `Feedback.js` with logic for theme switching and feedback submission. The mobile application on the right shows a feedback form with a light blue header and a light gray background, indicating it is in light mode.

```

Feedback.js - ASP_NutriTrack_Team89
project > my-app > JS Feedback.js > [o] FeedbackForm
  94   bordercolor: "#0088";
  95   linkColor: "lightblue",
  96 },
  97 };
  98
  99 const FeedbackForm = ({ navigation }) => {
100   const { theme, toggleTheme } = useTheme();
101   const styles = getStyles(theme);
102
103   const [feedback, setFeedback] = useState('');
104   const [submitted, setSubmitted] = useState(false);
105   const [error, setError] = useState('');
106   const [rating, setRating] = useState(0);
107
108   //ensure feedback page is rendered successfully
109   console.log('Feedback page executed successfully')
110
111   // handle submission of feedback
112   const handleSubmit = async () => {
113     // if feedback is not empty and there is a rating from the user
114     if (feedback.trim() && rating > 0) {
115       // error handling
116       try {
117         await addFeedback({ feedback, rating });
118         // test case 2: prints the submitted feedback and rating
119         console.log('Submitted feedback:', feedback, 'Rating:', rating);
120         // after submitted the feedback, it should reset the feedback input field, rating and an
121         setFeedback('');
122         setRating(0);
123       } catch (err) {
124         setError(err.message);
125       }
126     }
127   }
128
129   // handle theme switch
130   const toggleTheme = () => {
131     toggleTheme();
132     navigation.navigate('Home');
133   }
134
135   return (
136     <View style={styles.container}>
137       <Text>Please rate your experience and drop your honest<br/>review of the app to let us know how we can<br/>improve!</Text>
138       <Text>Enter your feedback</Text>
139       <Text>Star Rating</Text>
140       <Rating value={rating} onRatingChange={setRating} />
141       <Text>SUBMIT FEEDBACK</Text>
142     </View>
143   );
144 }
  
```

Test case TC003 & TC004 - Page should be able to submit feedback and rating and thank you message displayed

The screenshot shows a developer's workspace with a code editor and a mobile application running on an emulator. The code editor displays `Feedback.js` with logic for theme switching and feedback submission. The mobile application on the right shows a feedback form with a light blue header and a light gray background, and a "Thank you for your feedback!" message displayed below the form, indicating successful submission.

```

Feedback.js - ASP_NutriTrack_Team89
project > my-app > JS Feedback.js > [o] FeedbackForm
  94   bordercolor: "#0088";
  95   linkColor: "lightblue",
  96 },
  97 };
  98
  99 const FeedbackForm = ({ navigation }) => {
100   const { theme, toggleTheme } = useTheme();
101   const styles = getStyles(theme);
102
103   const [feedback, setFeedback] = useState('');
104   const [submitted, setSubmitted] = useState(false);
105   const [error, setError] = useState('');
106   const [rating, setRating] = useState(0);
107
108   // handle submission of feedback
109   const handleSubmit = async () => {
110     // if feedback is not empty and there is a rating from the user
111     if (feedback.trim() && rating > 0) {
112       // error handling
113       try {
114         await addFeedback({ feedback, rating });
115         // test case 2: prints the submitted feedback and rating
116         console.log('Submitted feedback:', feedback, 'Rating:', rating);
117         // after submitted the feedback, it should reset the feedback input field, rating and an
118         setFeedback('');
119         setRating(0);
120         setError('');
121         setSubmitted(true);
122       } catch (err) {
123         setError(err.message);
124       }
125     }
126   }
127
128   // handle theme switch
129   const toggleTheme = () => {
130     toggleTheme();
131     navigation.navigate('Home');
132   }
133
134   return (
135     <View style={styles.container}>
136       <Text>Thank you for your feedback!</Text>
137     </View>
138   );
139 }
  
```

Test case TC005 - Error message should be displayed if submit button is pressed with only the feedback and without the rating

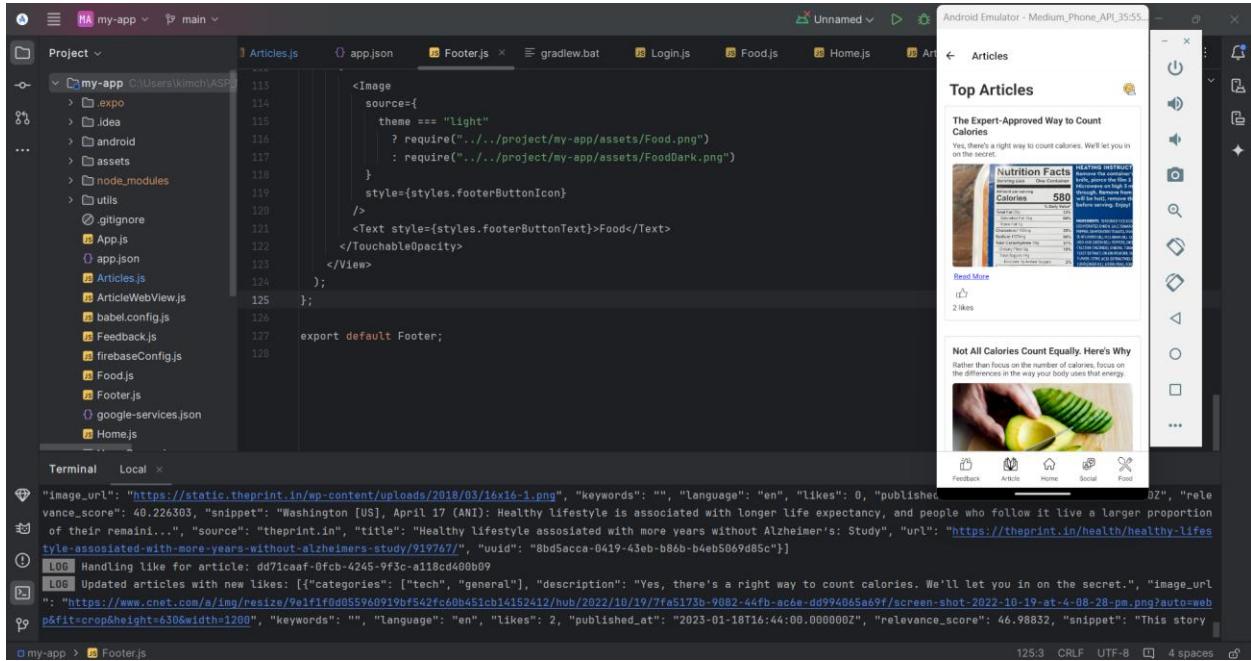
Test case TC006 - Error message should be displayed if submit button is pressed with only the feedback and without the rating

Articles Page

Test Case ID	Expected Result	Actual Result	Status
TC001	The like count of the article with the specified UUID should increase by 1.	Matched the expected result	Pass
TC002	Verify that the articles are updated in the local storage without any errors.	Matched the expected result	Pass
TC003	Check that the component initializes and starts the article fetching process correctly.	Matched the expected result	Pass
TC004	Verify that articles load correctly from local storage.	Matched the expected result	Pass
TC005	Verify the behavior when no articles are in local storage, ensuring API fetch occurs.	Matched the expected result	Pass
TC006	Ensure that data fetching from the API executes correctly and that results are parsed and stored without errors.	Matched the expected result	Pass
TC007	Verify that new articles fetched from the API are correctly added to local storage.	Matched the expected result	Pass

Test Case ID	Test Scenario	Steps
TC001	A user likes an article by clicking the like button.	<ol style="list-style-type: none"> Trigger the handleLike function with a specific article UUID. Verify that the like count for the specified article increases by 1. Check if the updated articles array reflects the increased like count.
TC002	The application updates the local storage after a user likes an article.	<ol style="list-style-type: none"> Modify the articles array by updating an article or increasing a like count. Save the updated articles array to local storage. Retrieve the articles back from local storage. Verify no errors were thrown during the saving process.
TC003	The ArticlesScreen component initializes and fetches articles on launch.	<ol style="list-style-type: none"> Launch the ArticlesScreen component. Verify that useEffect is triggered on component mount. Check console logs for initial fetch initiation. Ensure setLoading is called to manage the loading state appropriately.
TC004	The application loads articles from local storage if available.	<ol style="list-style-type: none"> Check for existing articles in local storage upon component mount. Load articles from local storage if present. Update the articles state with the loaded articles. Verify articles are displayed correctly in the component.
TC005	The app fetches articles from the API when local storage is empty.	<ol style="list-style-type: none"> Ensure local storage is empty or simulate empty storage. Trigger component mount and start the article fetching process. Fetch data from API due to absence in local storage. Confirm articles fetched from API are displayed.
TC006	The application correctly fetches and handles data from external APIs.	<ol style="list-style-type: none"> Simulate absence of articles in local storage or clear it. Fetch articles from API. Parse and store results from API without errors. Update articles state with new data from API.
TC007	Newly fetched articles from the API are stored in local storage.	<ol style="list-style-type: none"> Fetch new articles from API as described in TC006. Save the fetched articles to local storage. Retrieve the articles to verify they are correctly stored. Ensure the app displays the new articles from local storage on next launch.

Test case TC001- The like count of the article with the specified UUID should increase by 1.



The screenshot shows a development environment with a code editor and a running emulator. The code editor displays `Articles.js` with some logic for handling likes. The emulator shows a news feed with two articles. The first article, titled "The Expert-Approved Way to Count Calories", has a thumbnail image of a nutrition facts label and a like count of 2. The second article, titled "Not All Calories Count Equally. Here's Why", has a thumbnail image of a person peeling a lime and a like count of 0. The bottom of the screen shows navigation icons for Feedback, Article, Home, Social, and Food.

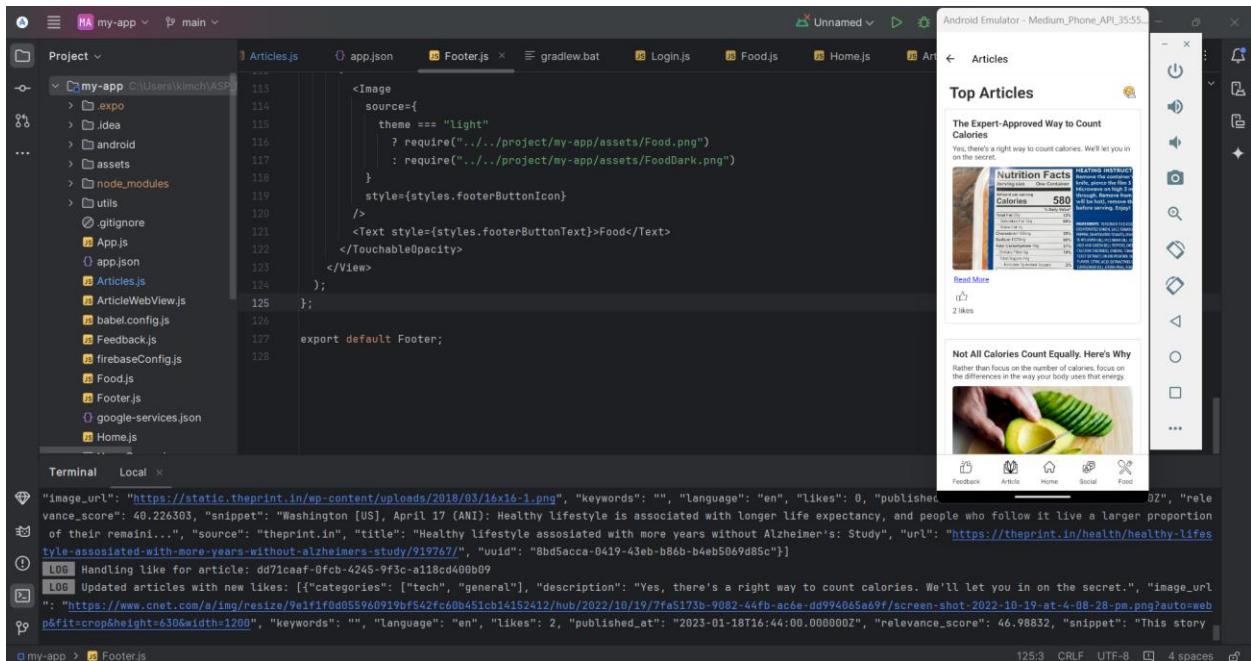
```

    <Image
      source={
        theme === "Light"
          ? require("../project/my-app/assets/Food.png")
          : require("../project/my-app/assets/FoodDark.png")
      }
      style={styles.footerButtonIcon}
    />
    <Text style={styles.footerButtonText}>Food</Text>
  </TouchableOpacity>
</View>
);
};

export default Footer;

```

Test case TC002 - Verify that the articles are updated in the local storage without any errors.



This screenshot is identical to the one above, showing the same code in `Articles.js` and the same two articles in the emulator. The first article still has 2 likes and the second has 0 likes. The bottom navigation bar is visible.

```

    <Image
      source={
        theme === "Light"
          ? require("../project/my-app/assets/Food.png")
          : require("../project/my-app/assets/FoodDark.png")
      }
      style={styles.footerButtonIcon}
    />
    <Text style={styles.footerButtonText}>Food</Text>
  </TouchableOpacity>
</View>
);
};

export default Footer;

```

Test case TC003 - Check that the component initializes and starts the article fetching process correctly.

The screenshot shows the Android Studio interface with the code editor open to `Articles.js`. The code handles the rendering of a footer component based on the theme. It includes imports for `Image`, `Text`, and `View` from `react-native`, along with `styles` and `Footer.js`. The `Footer.js` file is shown in the background. The terminal window at the bottom displays log messages indicating the retrieval of nutrients information and the loading of articles from local storage. To the right, the Android emulator shows the application running on a medium phone. The main screen displays a list titled "Top Articles" with two items: "The Expert-Approved Way to Count Calories" and "Not All Calories Count Equally. Here's Why". Each article card includes a thumbnail image, a title, a snippet of text, a "Read More" button, and a "2 likes" counter. The bottom navigation bar of the app includes "Feedback", "Article", "Home", "Social", and "Food" tabs.

Test case TC004 - Verify that articles load correctly from local storage.

This screenshot is identical to the one above, showing the same code in `Articles.js` and the same application state in the emulator. The logs in the terminal show the same sequence of events: retrieving nutrients information, successfully fetching total nutrients, and loading articles from local storage. The "Top Articles" section of the app shows the same two articles: "The Expert-Approved Way to Count Calories" and "Not All Calories Count Equally. Here's Why". The navigation bar at the bottom remains the same, with tabs for Feedback, Article, Home, Social, and Food.

Test case TC005 - Verify the behavior when no articles are in local storage, ensuring API fetch occurs.

The screenshot shows the 'Articles' screen of a mobile application. At the top, there is a header with the title 'Top Articles'. Below the header, there is a card for an article titled 'The Expert-Approved Way to Count Calories'. The card features a nutrition facts table and a snippet of text. Below this card, there is another article card with a thumbnail of a person cutting an apple. The bottom of the screen has a navigation bar with icons for 'Feedback', 'Article', 'Home', 'Social', and 'Food'.

```

my-app C:\Users\kimch\AS... Articles.js app.json Footer.js gradlew.bat Login.js Food.js Home.js Ar
113     <Image
114       source={
115         theme === "light"
116           ? require("../project/my-app/assets/Food.png")
117           : require("../project/my-app/assets/FoodDark.png")
118       }
119       style={styles.footerButtonIcon}
120     />
121     <Text style={styles.footerButtonText}>Food</Text>
122   </TouchableOpacity>
123 </View>
124 );
125 };
126
127 export default Footer;
  
```

Terminal Local x +

```

L06 Component mounted, starting to fetch articles...
L06 Loaded articles from local storage.
L06 Component mounted, starting to fetch articles...
L06 No articles in local storage, fetching from API.
L06 Fetching data from API...
L06 Fetched and stored new articles: [{"categories": ["tech", "general"], "description": "Yes, there's a right way to count calories. We'll let you in on the secret.", "image_url": "https://www.cnet.com/e/img/resize/9e1ff0d055960919bf542fc60b451cb14152412/hub/2022/10/19/7fa5173b-9082-44fb-ac6e-dd994065a69f/screen-shot-2022-10-19-at-4-08-28-pm.png?auto=web&fit=crop&height=630&width=1200", "keywords": "", "language": "en", "likes": 0, "published_at": "2023-01-18T16:44:00.000000Z", "relevance_score": 46.988304, "snippet": "This story"}]
  
```

Test case TC006 - Ensure that data fetching from the API executes correctly and that results are parsed and stored without errors.

The screenshot shows the 'Articles' screen of a mobile application. At the top, there is a header with the title 'Top Articles'. Below the header, there is a card for an article titled 'The Expert-Approved Way to Count Calories'. The card features a nutrition facts table and a snippet of text. Below this card, there is another article card with a thumbnail of a person cutting an apple. The bottom of the screen has a navigation bar with icons for 'Feedback', 'Article', 'Home', 'Social', and 'Food'.

```

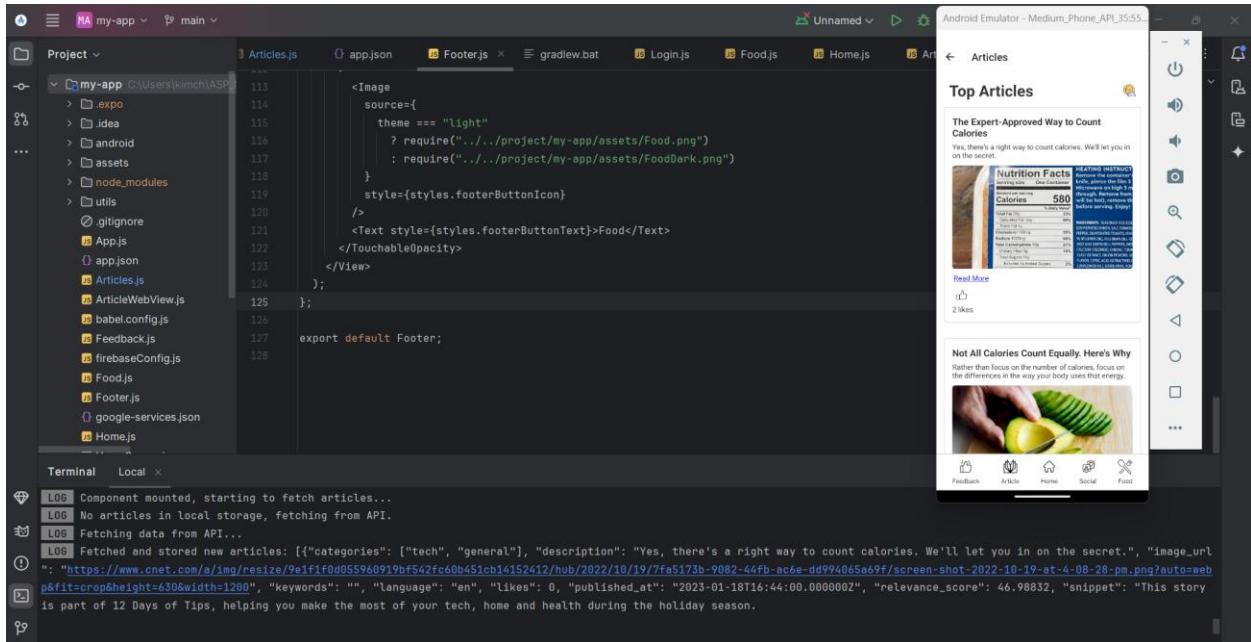
my-app C:\Users\kimch\AS... Articles.js app.json Footer.js gradlew.bat Login.js Food.js Home.js Ar
113     <Image
114       source={
115         theme === "light"
116           ? require("../project/my-app/assets/Food.png")
117           : require("../project/my-app/assets/FoodDark.png")
118       }
119       style={styles.footerButtonIcon}
120     />
121     <Text style={styles.footerButtonText}>Food</Text>
122   </TouchableOpacity>
123 </View>
124 );
125 };
126
127 export default Footer;
  
```

Terminal Local x +

```

L06 Component mounted, starting to fetch articles...
L06 Loaded articles from local storage.
L06 Component mounted, starting to fetch articles...
L06 No articles in local storage, fetching from API.
L06 Fetching data from API...
L06 Fetched and stored new articles: [{"categories": ["tech", "general"], "description": "Yes, there's a right way to count calories. We'll let you in on the secret.", "image_url": "https://www.cnet.com/e/img/resize/9e1ff0d055960919bf542fc60b451cb14152412/hub/2022/10/19/7fa5173b-9082-44fb-ac6e-dd994065a69f/screen-shot-2022-10-19-at-4-08-28-pm.png?auto=web&fit=crop&height=630&width=1200", "keywords": "", "language": "en", "likes": 0, "published_at": "2023-01-18T16:44:00.000000Z", "relevance_score": 46.988304, "snippet": "This story"}]
  
```

Test case TC007 - Verify that new articles fetched from the API are correctly added to local storage.



The screenshot shows a development environment with the following details:

- Project Structure:** my-app (C:\Users\kimch\AS...), containing files like Articles.js, app.json, Footer.js, gradlew.bat, Login.js, Food.js, Home.js, and Articles.
- Code Editor:** Shows the content of Footer.js, which includes logic for rendering a footer button icon based on theme (light or dark).
- Terminal:** Displays log messages related to article fetching:
 - L06 Component mounted, starting to fetch articles...
 - L06 No articles in local storage, fetching from API...
 - L06 Fetching data from API...
 - L06 Fetched and stored new articles: [{"categories": ["tech", "general"], "description": "Yes, there's a right way to count calories. We'll let you in on the secret.", "image_url": "https://www.cnet.com/a/img/resize/9ef1f0d055960919bf542fe60b451cb14152412/hub/2022/10/19/7fe5173e-9082-44fb-ac6a-dd974065a69f/screen-shot-2022-10-19-at-4-08-28-pm.png?auto=web&fit=crop&height=630&width=1200", "keywords": "", "language": "en", "Likes": 0, "published_at": "2023-01-18T16:44:00.000000Z", "relevance_score": 46.98832, "snippet": "This story is part of 12 Days of Tips, helping you make the most of your tech, home and health during the holiday season."}]
- Preview Window:** Shows two articles:
 - Top Articles:** "The Expert-Approved Way to Count Calories".

Yes, there's a right way to count calories. We'll let you in on the secret.



[Read More](#)

2 likes
 - Not All Calories Count Equally. Here's Why:**

Rather than focus on the number of calories, focus on the differences in the way your body uses that energy.

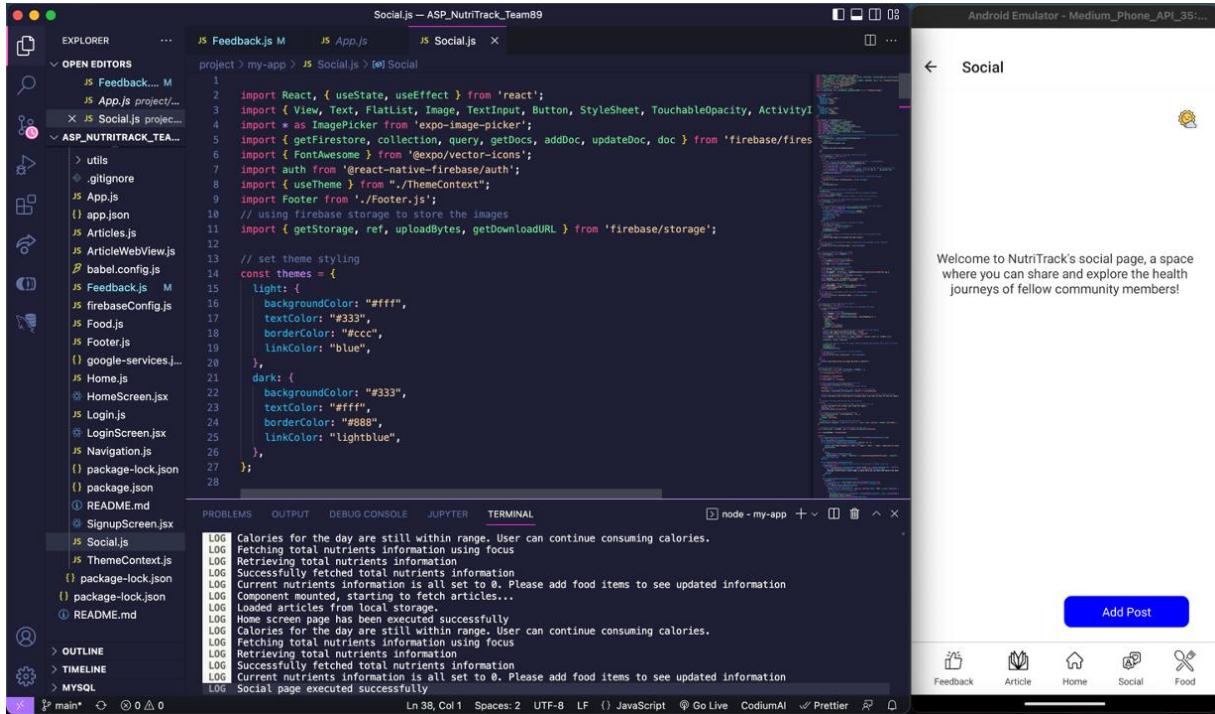


Social page

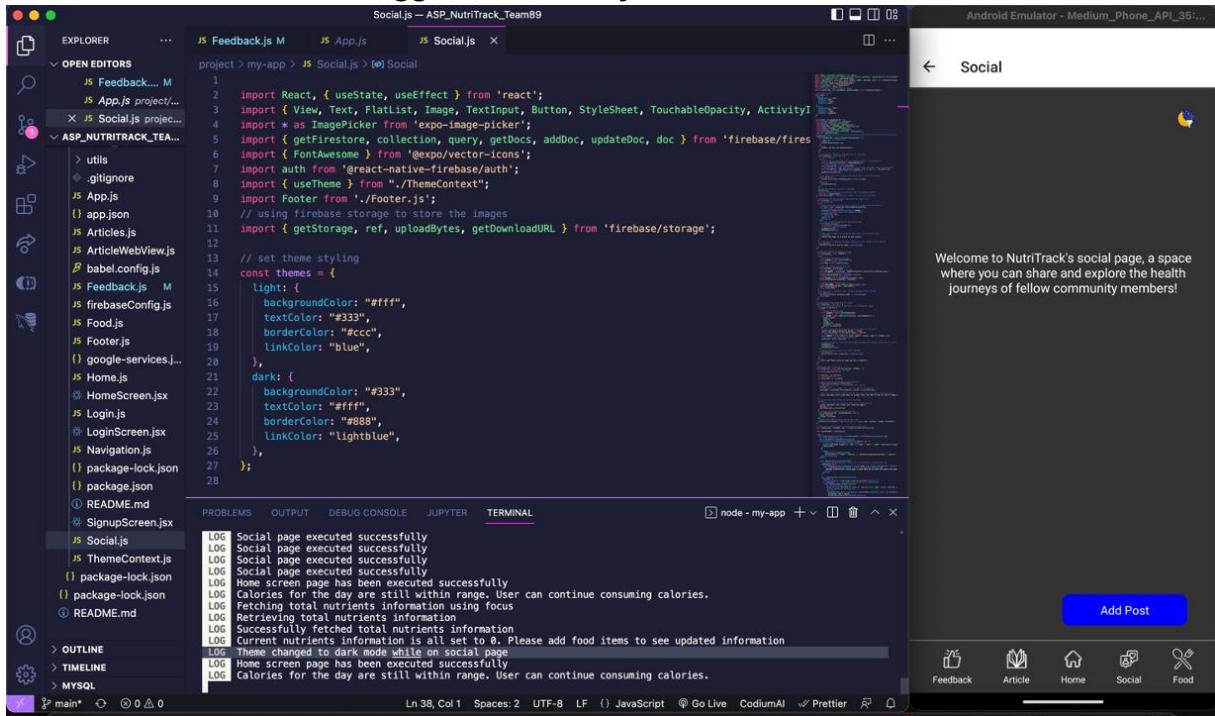
Test Case ID	Expected Result	Actual Result	Status
TC001	Page Load Verification	Matched the expected result	Pass
TC002	Theme toggle functionality - Light & dark theme	Matched the expected result	Pass
TC003	Add Post button opens up device gallery	Matched the expected result	Pass
TC004	Image can be chosen from the gallery	Matched the expected result	Pass
TC005	Caption can be written for the chosen image	Matched the expected result	Pass
TC006	Image has been posted to the page	Matched the expected result	Pass
TC007	Cancel the process of posting an image	Matched the expected result	Pass
TC008	Like an image	Matched the expected result	Pass
TC009	dislike an image	Matched the expected result	Pass

Test Case ID	Test Scenario	Steps
TC001	Page Load Verification	1. Navigate to the Social Page. 2. Verify if all components are visible.
TC002	Theme toggle functionality - Light & dark theme	1. Locate the theme icon located on right corner of the page. 2. Click on the sun icon 3. Verify if dark mode is applied to the screen while the icon changes to a moon icon 4. Click on the moon icon 5. Verify if light mode is applied, while moon icon switched to sun
TC003	Add Post button opens up device gallery	1.Click on the add post button 2.Verify if the gallery opens up
TC004	Image can be chosen from the gallery	1. Click on the add post button 2. Choose an image from the gallery 3. Verify if the chosen image is seen on the caption page with a post and cancel button
TC005	Caption can be written for the chosen image	1. Click on the add post button 2. Choose an image from the gallery 3.Verify visually if caption can be written for the image chosen
TC006	Image has been posted to the page	1. Click on the add post button 2. Choose an image from the gallery 3. Write a caption for the image 4. Click on the 'Post' button after writing the caption 5. Posted image should be shown on the page with th caption and like button
TC007	Cancel the process of posting an image	1. Click on the add post button 2. Choose an image from the gallery 3. Write a caption for the image 4. Click on the 'Cancel' button after writing the caption 5. Page should return to the social page
TC008	Like an image	1. Click on the heart icon for an image on the feed 2. Heart icon should turn red and like should increase by 1
TC009	dislike an image	1. Click on the heart icon for an image that's already liked 2. Heart icon should turn white and like should decrease by 1

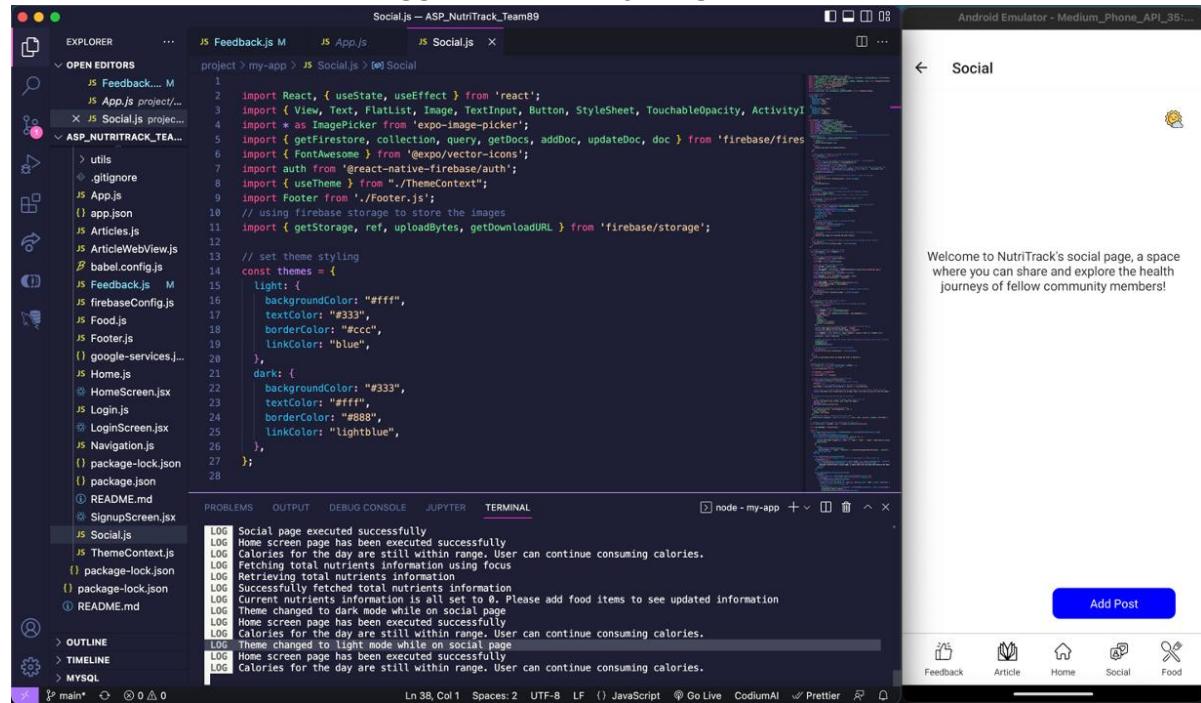
Test case TC001 - Page Load Verification



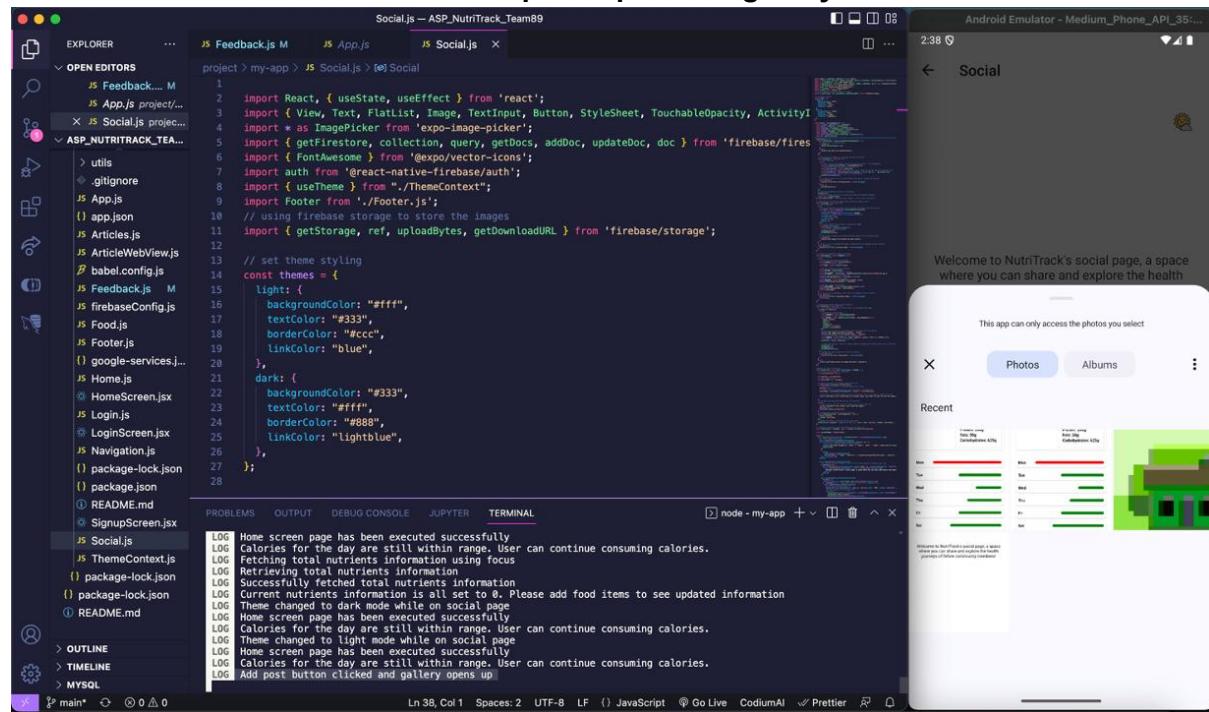
Test case TC002 - Theme toggle functionality - Dark theme



Test case TC002 - Theme toggle functionality - Light theme



Test case TC003 - Add Post button opens up device gallery



Test case TC004 - Image can be chosen from the gallery

Social.js – ASP_NutriTrack_Team89

EXPLORER

OPEN EDITORS

- JS Feedback.js M
- JS App.js
- JS Social.js X

```

project > my-app > JS Social.js > Social
  const pickImage = async () => {
    // error handling
    try {
      // using imagePicker to allow users to choose the images from their device
      let result = await ImagePicker.launchImageLibraryAsync({
        // only images are allowed
        mediaTypes: ImagePicker.MediaTypeOptions.Images,
        // allow the chosen image to be cropped and edited
        allowsEditing: true,
        // image aspect ratio
        aspect: [4, 3],
        quality: 1,
      });
      // if user cancels the process of posting the image
      if (!result.canceled) {
        const source = result.assets[0].uri;
        // store the image URI locally
        setImage(source);
        setAddingPost(true);
      }
      // test case 3: Image to be posted has been chosen, at the preview stage
      if (result) {
        console.log('Image to be posted has been chosen')
      }
      // if any error in choosing the images, display the error message to test function
      catch (error) {
        console.error('Error picking image:', error.message);
      }
    }
  }
  // log messages
  LOG Calories for the day are still within range. User can continue consuming calories.
  LOG Fetching total nutrients information using focus
  LOG Retrieving total nutrients information
  LOG Successfully fetched total nutrients information
  LOG Current nutrients information is all set to 0. Please add food items to see updated information
  LOG Theme changed to dark mode while on social page
  LOG Home screen page has been executed successfully
  LOG Calories for the day are still within range. User can continue consuming calories.
  LOG Theme changed to light mode while on social page
  LOG Home screen page has been executed successfully
  LOG Calories for the day are still within range. User can continue consuming calories.
  LOG Add post button clicked and gallery opens up
  LOG Image to be posted has been chosen

```

PROBLEMS **OUTPUT** **DEBUG CONSOLE** **JUPITER** **TERMINAL**

Ln 38, Col 1 Spaces: 2 UTF-8 LF {} JavaScript Go Live CodiumAI Prettier

Android Emulator - Medium_Phone_API_35...

Social

Tue

Wed

Thu

Fri

Sat

Sun

Add a caption...

CANCEL POST

Feedback **Article** **Home** **Social** **Food**

Test case TC005 - Caption can be written for the chosen image

Social.js – ASP_NutriTrack_Team89

EXPLORER

OPEN EDITORS

- JS Feedback.js M
- JS App.js
- JS Social.js X

```

project > my-app > JS Social.js > Social
  // posting the chosen image with the caption
  const handlePost = async () => {
    // if the image and caption is both there, try uploading the image
    if (image && caption) {
      try {
        // Upload the edited image
        const imageUri = await uploadImage(image);
        // reference to the firestore collection
        const docRef = await addDoc(collection(db, 'socialPagePosts'), {
          image: imageUri,
          caption,
          likes: 0,
          likedBy: [],
          createdAt: new Date(),
          userId: currentUserId,
        });
        // test case 4: able to successfully post the image with the caption
        console.log('image successfully posted: ', image)
        console.log('caption for the chosen image: ', caption)
        // create a new post object with the uploaded data
        const newPost = { id: docRef.id, image: imageUri, caption, likes: 0, likedBy: [] };
        // include the new post in the posts state
        setPosts([...posts, newPost]);
        // reset the caption, clear the chosen image and adding post becomes false since its posted
        setCaption('');
        setImage(null);
        setAddingPost(false);
      }
    }
    // log messages
    LOG Calories for the day are still within range. User can continue consuming calories.
    LOG Fetching total nutrients information using focus
    LOG Retrieving total nutrients information
    LOG Successfully fetched total nutrients information
    LOG Current nutrients information is all set to 0. Please add food items to see updated information
    LOG Theme changed to dark mode while on social page
    LOG Home screen page has been executed successfully
    LOG Calories for the day are still within range. User can continue consuming calories.
    LOG Theme changed to light mode while on social page
    LOG Home screen page has been executed successfully
    LOG Calories for the day are still within range. User can continue consuming calories.
    LOG Add post button clicked and gallery opens up
    LOG Image to be posted has been chosen

```

PROBLEMS **OUTPUT** **DEBUG CONSOLE** **JUPITER** **TERMINAL**

Ln 38, Col 1 Spaces: 2 UTF-8 LF {} JavaScript Go Live CodiumAI Prettier

Android Emulator - Medium_Phone_API_35...

Social

Tue

Wed

Thu

Fri

Sat

Sun

testest

CANCEL POST

Feedback **Article** **Home** **Social** **Food**

Test case TC006 - Image has been posted to the page

The screenshot shows a developer's environment with a code editor and a mobile application running in an emulator.

Code Editor:

- Project: my-app > JS Social.js > Social
- File: Social.js
- Content: JavaScript code for handling image posting. It includes logic for handling both image and caption, creating a new post object with the uploaded data, and updating the posts state.

Android Emulator:

- Title: Social
- Content: A social media feed with a post from "testest". The post has 0 likes.
- Buttons: Add Post, Feedback, Article, Home, Social, Food.

Test case TC007 - Cancel the process of posting an image

The screenshot shows a developer's environment with a code editor and a mobile application running in an emulator.

Code Editor:

- Project: my-app > JS Social.js > Social
- File: Social.js
- Content: JavaScript code for handling image posting. It includes logic for handling both image and caption, creating a new post object with the uploaded data, and updating the posts state. The code also includes a section for canceling the process.

Android Emulator:

- Title: Social
- Content: A social media feed with a post from "testest". The post has 0 likes.
- Buttons: Add Post, Feedback, Article, Home, Social, Food.

Test case TC008 - Like an image

The screenshot shows a developer's environment with multiple windows open. On the left, the VS Code interface displays several files under the 'ASPNUTRITRACK_TEAM...' folder, including Feedback.js, App.js, and Social.js. The Social.js file is currently active, showing code related to posting images and captions. Below the code editor are sections for PROBLEMS, OUTPUT, DEBUG CONSOLE, JUPITER, TERMINAL, and a node - my-app terminal tab. The terminal shows log output for a file named 5e2f8.appspot.com, which includes a URL for a feedback form and a message about a successful image post. On the right, a mobile application window titled 'Social' is displayed, showing a welcome message and a timeline of posts. One post by 'testtest' is shown with a red heart icon indicating it has 1 like. At the bottom right of the mobile screen is a blue 'Add Post' button.

Test case TC009 - Undo a like

The screenshot shows a developer's environment with two main windows. On the left is a Visual Studio Code (VS Code) instance with the title "Social.js - ASP_NutriTrack_Team89". The Explorer sidebar lists several files under "OPEN EDITORS" and "ASP_NUTRITRACK_TEA...". The "Social.js" file is open in the editor, showing code related to posting images and captions to a Firestore collection. The bottom of the VS Code window shows tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, JUPYTER, and TERMINAL.

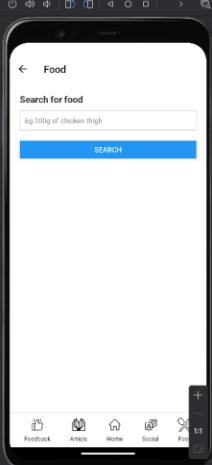
The right side of the screenshot shows an "Android Emulator - Medium_Phone_API_35..." window titled "Social". It displays a social feed with a post from "testest" (ID 167). The post includes a placeholder image and the caption "image successfully posted". Below the post are four progress bars for the days of the week: Wed, Thu, Fri, and Sat, all of which are fully green. At the bottom of the feed is a blue button labeled "Add Post".

Food Search Page:

Test Case ID	Test Scenario	Steps
TC001	Page Load Verification	1. Navigate to the Food Page. 2. Verify if all components are visible.
TC002	Search Functionality	1. Enter a food item name (e.g., "100g of chicken breast") in the search bar. 2. Click on the "Search" button.
TC003	Food item Selection	1. Perform a search. 2. Select a food item. 3. Check if nutrient details appear.
TC004	Save Food Item	1. Search for a food item. 2. Select a food item. 3. Click the "Save Food" button.
TC005	Dark Mode Toggle	1. Go back to homepage, Toggle dark mode. 2. Verify if the Food Page adapts to dark mode.
TC006	Error Handling in Search	1. Search for a wrong format food item 2. Perform a search.

Test Case ID	Expected Result	Actual Result	Status	
TC001	The page should load with search bar, food list, and buttons visible.	Matched the expected result	Pass	
TC002	The food item should appear in the list with its data fetched.	Matched the expected result	Pass	
TC003	The selected food's detail should be displayed correctly	Matched the expected result	Pass	
TC004	Food data should be saved, and a success message should be shown.	Matched the expected result	Pass	
TC005	The Food Page should switch to dark mode with appropriate background and text color changes.	Matched the expected result	Pass	
TC006	No nutrition value should be shown.	Matched the expected result	Pass	

Test Case 1: Page Load Verification



The screenshot shows a mobile application interface. At the top, there is a navigation bar with icons for Feedback, Article, Home, and Social. Below it is a search bar with the placeholder 'Search for food'. A blue button labeled 'SEARCH' is visible. The main content area displays a list of food items, with the first item being '100g of chicken thigh'.

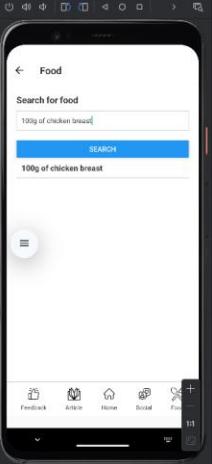
```

my-app C:\Users\user\Desktop\nutri
|__ expo
|__ idea
|__ android
|__ assets
|__ node_modules
|__ utils
|__ .gitignore
|__ App.js
|__ app.json
|__ Articles.js
|__ ArticleWebView.js
|__ babel.config.js
|__ Feedback.js
|__ firebaseConfig.js
|__ Food.js <-- Selected file
|__ Footer.js
|__ google-services.json
|__ Home.js
|__ HomeScreen.jsx
|__ Login.js
|__ LoginScreen.jsx
|__ Navigation.js
|__ package.json
|__ package-lock.json
|__ README.md
|__ SignupScreen.jsx
|__ Social.js
|__ ThemeContext.js

Terminal Local
LOG Retrieving total nutrients information
LOG Successfully fetched total nutrients information
LOG Successfully reset nutrients information
LOG Current nutrients information is all set to 0. Please add food items to see updated information
LOG Current nutrients information is all set to 0. Please add food items to see updated information
LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: Current theme is: light

```

Test Case 2 Search Functionality:



The screenshot shows a mobile application interface. At the top, there is a navigation bar with icons for Feedback, Article, Home, and Social. Below it is a search bar with the placeholder 'Search for food'. A blue button labeled 'SEARCH' is visible. The main content area displays a list of food items, with the first item being '100g of chicken breast'.

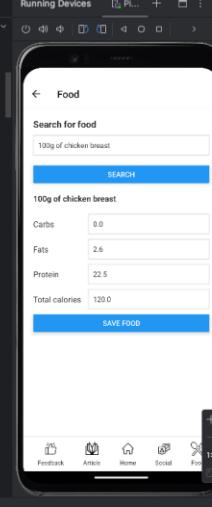
```

my-app C:\Users\user\Desktop\nutri
|__ expo
|__ idea
|__ android
|__ assets
|__ node_modules
|__ utils
|__ .gitignore
|__ App.js
|__ app.json
|__ Articles.js
|__ ArticleWebView.js
|__ babel.config.js
|__ Feedback.js
|__ firebaseConfig.js
|__ Food.js <-- Selected file
|__ Footer.js
|__ google-services.json
|__ Home.js
|__ HomeScreen.jsx
|__ Login.js
|__ LoginScreen.jsx
|__ Navigation.js
|__ package.json
|__ package-lock.json
|__ README.md
|__ SignupScreen.jsx
|__ Social.js
|__ ThemeContext.js

Terminal Local
LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: Current theme is: light
LOG Test Case 3: Search button clicked with query: 100g of chicken breast
LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: Current theme is: light
LOG Test Case 4: Food data fetched successfully!!
LOG Test Case 6: Food item selected: 100g of chicken breast

```

Test Case 3 Food Item Selection:



Project

```

my-app C:\Users\User\Desktop\Nutri...
  > expo
  > idea
  > android
  > assets
  > node_modules
  > utils
    .gitignore
    App.js
    app.json
    Articles.js
    ArticleWebView.js
    babel.config.js
    Feedback.js
    firebaseConfig.js
    Food.js
    Footer.js
    google-services.json
    Home.js
    HomeScreen.jsx
    Login.js
    LoginScreen.jsx
    Navigation.js
    package.json
    package-lock.json
    README.md
    SignupScreen.jsx
    Social.js
    ThemeContext.js

```

Food.js

```

backgroundColor: "#333",
textColor: "#fff",
borderColor: "#888",
imColor: "lightblue",
placeholderColor: "#aaa", // Placeholder color for dark theme
},
};

// Dynamically generate styles based on the selected theme
const getStyles = (theme) =>
StyleSheet.create({
  container: {
    flex: 1,
    padding: 20,
    paddingBottom: 110,
    backgroundColor: themes[theme].backgroundColor,
  },
  label: {
    fontSize: 18,
    fontWeight: "bold",
    marginBottom: 10,
    color: themes[theme].textColor,
  },
  input: {
    height: 40,
    borderColor: themes[theme].borderColor,
    borderWidth: 1,
    paddingHorizontal: 10,
    marginBottom: 20,
    color: themes[theme].textColor,
  },
},
&ndash;: f

```

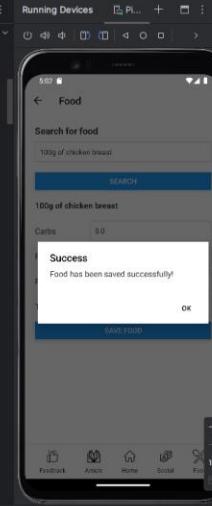
Terminal Local

```

LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: Current theme is: light
LOG Test Case 4: Food data fetched successfully!
LOG Test Case 6: Food item selected: 100g of chicken breast
LOG Test Case 7: Nutrients calculated - Carbs: 0 Fats: 2.62 Protein: 22.5 Calories: 120
LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: current theme is: light

```

Test Case 4 Save Food Item:



Project

```

my-app C:\Users\User\Desktop\Nutri...
  > expo
  > idea
  > android
  > assets
  > node_modules
  > utils
    .gitignore
    App.js
    app.json
    Articles.js
    ArticleWebView.js
    babel.config.js
    Feedback.js
    firebaseConfig.js
    Food.js
    Footer.js
    google-services.json
    Home.js
    HomeScreen.jsx
    Login.js
    LoginScreen.jsx
    Navigation.js
    package.json
    package-lock.json
    README.md
    SignupScreen.jsx
    Social.js
    ThemeContext.js

```

Food.js

```

backgroundColor: "#333",
textColor: "#fff",
borderColor: "#888",
imColor: "lightblue",
placeholderColor: "#aaa", // Placeholder color for dark theme
},
};

// Dynamically generate styles based on the selected theme
const getStyles = (theme) =>
StyleSheet.create({
  container: {
    flex: 1,
    padding: 20,
    paddingBottom: 110,
    backgroundColor: themes[theme].backgroundColor,
  },
  label: {
    fontSize: 18,
    fontWeight: "bold",
    marginBottom: 10,
    color: themes[theme].textColor,
  },
  input: {
    height: 40,
    borderColor: themes[theme].borderColor,
    borderWidth: 1,
    paddingHorizontal: 10,
    marginBottom: 20,
    color: themes[theme].textColor,
  },
},
&ndash;: f

```

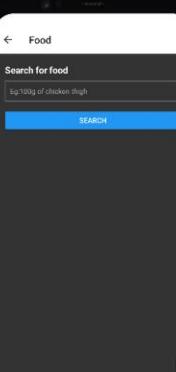
Terminal Local

```

LOG Test Case 4: Food data fetched successfully!
LOG Test Case 6: Food item selected: 100g of chicken breast
LOG Test Case 7: Nutrients calculated - Carbs: 0 Fats: 2.62 Protein: 22.5 Calories: 120
LOG Test Case 1: Food.js page loaded successfully.
LOG Test Case 2: Current theme is: light
LOG Test Case 8: Current nutrient values before saving: {"calories": 120, "carbs": 0, "fats": 2.6, "protein": 22.5}
LOG Test Case 9: Food data saved successfully: {"calories": 120, "carbs": 0, "fats": 2.6, "protein": 22.5}

```

Test Case 5 Dark Mode Toggle:



Project

my-app

expo

idea

android

assets

node_modules

utils

.gitignore

App.js

app.json

Articles.js

Article.js

ArticleWebView.js

babel.config.js

Feedback.js

firebaseConfig.js

Food.js

Footer.js

google-services.json

Home.js

HomeScreen.jsx

Login.js

LoginScreen.jsx

Navigation.js

package.json

package-lock.json

README.md

SignupScreen.jsx

Social.js

ThemeContext.js

Food.js

```
 37   backgroundColor: "#3333",
 38   textColor: "ffff",
 39   borderColor: "#8888",
 40   linkColor: "lightblue",
 41   placeholderColor: "#aaa", // Placeholder color for dark theme
 42 },
 43 };
 44
 45 // Dynamically generate styles based on the selected theme
 46 const getStyles = (theme) =>
 47   StyleSheet.create({
 48     container: {
 49       flex: 1,
 50       padding: 20,
 51       paddingBottom: 110,
 52       backgroundColor: themes[theme].backgroundColor,
 53     },
 54     label: {
 55       fontSize: 18,
 56       fontWeight: "bold",
 57       marginBottom: 10,
 58       color: themes[theme].textColor,
 59     },
 60     input: {
 61       height: 40,
 62       borderColor: themes[theme].borderColor,
 63       borderWidth: 1,
 64       paddingHorizontal: 10,
 65       marginBottom: 20,
 66       color: themes[theme].textColor,
 67     },
 68   });
 69 }
```

Terminal Local

LOG Limit for the day has not been reached. Calories added.

LOG Successfully retrieved and update total nutrients information

LOG Theme button has been triggered

LOG Home screen page has been executed successfully

LOG Calories for the day are still within range. User can continue consuming calories.

LOG Test Case 1: Food.js page loaded successfully.

LOG Test Case 2: Current theme is: dark

Test Case 6 Error Handling in Search:

Technical Challenge faced during development phase:

Login Page:

Handling multiple asynchronous operations

```
286 // Handling Asynchronous Operations: The LoginScreen.jsx file contains several asynchronous operations, such as signing in  
287 // with Google and handling login. These operations can lead to race conditions if not handled properly.  
288 // For example, if a user quickly taps the sign-in button multiple times, it could trigger multiple sign-in requests.  
289 // I mitigated this by disabling the button while the sign-in operation is in progress.
```

Home Page:

Theme toggle

```
659 // Theme toggling issues  
660 // The toggle functionality was not elaborate enough in the first iteration. Thus causing only a small change in the theme colour  
661 // Due to this issue I added a custom theme variable where when the user clicks onto the theme button, the icon image will change,  
662 // and the state of the variable will change accordingly. The state will then request for the custom light or dark mode UI themes.
```

Asyncstorage errors

```
664 // AsyncStorage errors  
665 // The home screen page uses an AsyncStorage to store and retrieve data, but errors might occur during this process. This might  
666 // cause the app to crash or to not retrieve and display accurate data. By integrating additional error checks and verifications,  
667 // I have improved the execution of the retrieval and accuracy of the macronutrients and calorie information being displayed.  
668 // With more checks and detailed error messages, it allows developers to be able to find the source of the issue quickly, and  
669 // address it immediately.
```

Navigation issues

```
671 // Navigation issues  
672 // The app has a navigational function to navigate to different screens, but navigation might not work as expected. This might  
673 // cause the app to crash and not display the correct screens as expected. I solved this by adding additional checks and print  
674 // messages to ensure that the targeted page has been successfully reached as well as error messages to indicate a failure to  
675 // reach the targeted page
```

Data calculation and display

```
677 // Data calculation and display  
678 // In the app it calculates and displays various data, such as daily progress and nutrient information, but errors might occur  
679 // during these calculations. This might cause the app to display incorrect information. To resolve this issue I ensured that  
680 // the formula that I used to calculate the daily calorie count was accurate, and I also included a style for each progress  
681 // bar to be limited to a set length so as to stop the bar from exceeding the length of the set screen.
```

User interface experience

```
683 // User interface experience  
684 // In the first iteration of the application, the user interface was functional but not visually pleasing and modern. To resolve  
685 // this issue I researched on modern home screen dashboards and came up with a new clean and seamless design that users can enjoy  
686 // using on a daily basis. I took out the search bar and included on the components included in the final iteration. Having a  
687 // clean and efficient design promotes higher level of learnability and user experience as well as maintains the feel of the app.
```

Feedback Page:

Firebase

```
/* The page is of low complexity since it contains only 2 elements - feedback input and star rating system. The feedback is stored using  
firebase so it can be updated and viewed. Creating the firebase was the only issue in the beginning because of the configuration while creating  
the database. As it was my first time utilizing firebase, applying the correct settings such as adding the application, creating the collection  
was new but nevertheless it was easily manageable using youtube videos and once that was done, creating the page was simple */
```

Article Page:

Asynchronous Data Handling

```
// Asynchronous Data Handling

// Challenge: Managing asynchronous operations such as fetching data from an API or
// local storage can lead to race conditions or inconsistencies in state.

// Solution: Use async/await consistently and handle all promises properly to ensure that
// operations complete in the expected order. Utilizing try/catch blocks within async functions
// helps manage errors effectively. The use of useEffect with an empty dependency array ensures
// operations are performed right after the component mounts, reducing timing issues.
```

Dynamic Theme Handling

```
// Dynamic Theme Handling

// Challenge: Ensuring data like article likes and fetched articles persist across app restarts.

// Solution: Using a context (useTheme) to handle theme changes ensures that the theme data flows through the component
// tree without unnecessary re-renders. The context provider at the app level can manage theme states and propagate
// changes efficiently.
```

Data Persistence

```
// Data Persistence

// Challenge: Ensuring data like article likes and fetched articles persist across app restarts.

// Solution: Utilizing AsyncStorage to save and retrieve articles ensures that user data like article
// likes persists even when the app is closed and reopened. Handling JSON data serialization and parsing
// carefully prevents data corruption.
```

Managing Navigation and User Interactions

```
// Managing Navigation and User Interactions

// Challenge: Handling navigation and interactions like liking an article or reading more about it
// without introducing bugs in the navigation flow or state management.

// Solution: Using React Navigation effectively, ensuring that navigation props are passed correctly
// and that routes are defined clearly. Additionally, updating the articles state immediately when a like button
// is pressed ensures the UI is responsive and reflects changes instantly.
```

Social Page: API integration

```
/* For the first version of social page, I used Eventbrite's API but there were issues like the references for searching events, retrieving all events under a category, etc were deprecated. This meant that the events had to specifically be retrieved using their unique event ID, venue ID, organisation ID, etc. Using this API was risky and tedious to move forward with as it meant that the current references might also be deprecated. The event addition into the categories would also require a lot of time and effort to manually find the event ID, ensure event is no longer displayed after the date has passed, event falls under the health and fitness sector, and more. Hence why there were only limited options for the users to choose from. While it will meet the aim of the page, it will not be as efficient and effective as it would be with a proper API, where there are references to insert all the events under a specific category and remove them once the data has passed.
```

Storage

```
Thus why we decided to change the objective of the page, so it can be more functional and effective. So for the second version of the page, the social page is a simple social feed where users can add images from their device, add a description and like posts. For this version, the only issue was creating the firebase storage because with a local storage, there were issues viewing the posted images from another device, hence opted for firebase storage. The problem was integrating the storage into the code, but with the aid of youtube videos, it could be properly set up and used to store the images.*/
```

Food Search Page:

API Integration:

```
//API Integration:  
//One of the significant challenges during the development phase of the Food Search Page was to identify the correct and reliable API.  
//Initially, I tried Nutritionix API and USDA FoodData Central API.  
//However, these 2 APIs I experienced slow responding when I request from their API.  
//As a result, I need to find another API that is more suitable to deliver the necessary nutritional details for my requirements.  
//As a result I switch around with different APIs and testing different ones for performance  
//and Edamam API provides a good balance between the depth of nutritional data and the speed of responses.
```

Displaying Nutritional Information:

```
//Displaying Nutritional Information:  
//One of the challenges that I faced is after retrieving the food data,  
//parsing the JSON response to display the nutritional information is a challenge  
//as some of the food does not have certain nutritional data which could lead to error or blank field in the UI.  
//As a result, I need to ensure data validation for missing or incomplete nutritional values before putting them into key.
```

UI Responsiveness in Dark Mode.

```
//UI Responsiveness in Dark Mode.  
//One of the challenges that I faced is to change the page to dark mode when the button is toggled on home page.  
//Implementing dynamic theme switcher while ensuring the Food Search Page is fully responsive across all devices.  
//
```

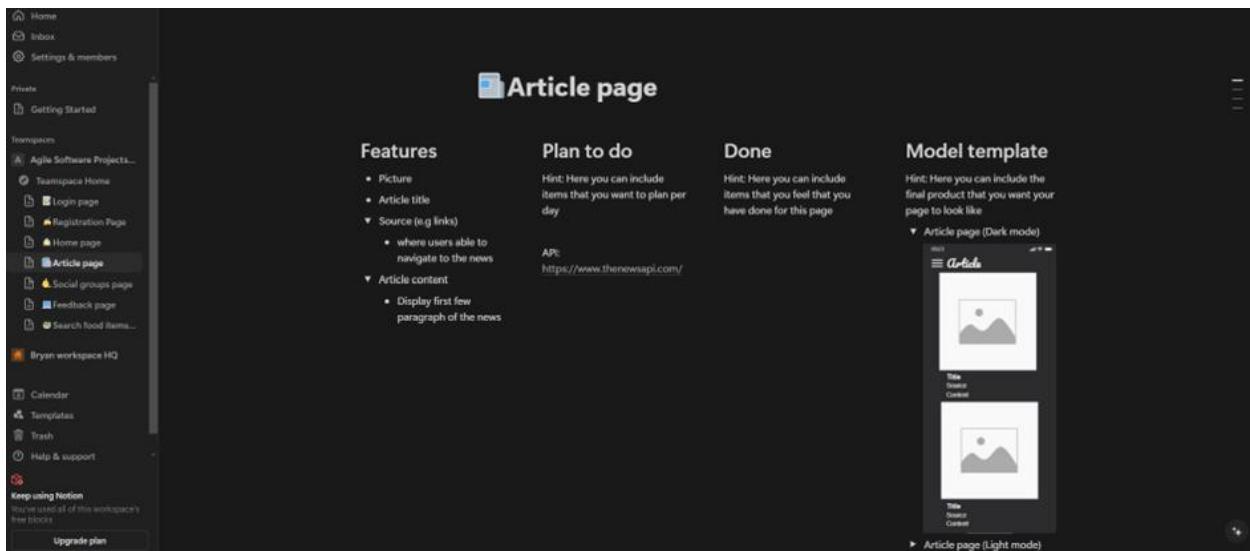
Methodologies Used for Project Development

Impact Mapping

Impact mapping is a strategic planning technique that aligns the team's work with business goals. It helps visualize the connection between deliverables and their impact on stakeholders or the organization. This approach ensures that development efforts directly contribute to desired outcomes and helps prevent scope creep.

In our project, we were able to understand the needs of the Singaporean audience and determine how our idea could improve the lifestyle of busy Singaporeans. By analyzing statistics and trends, we identified key areas for improvement in current calorie tracking applications, particularly in terms of data reliability. Our goal is to deliver a reliable and easy-to-use user experience, helping Singaporeans reach their health and fitness goals while excelling in their daily tasks.

Kanban



The screenshot shows a Notion workspace with a dark theme. On the left, there is a sidebar with various sections like Home, Inbox, Settings & members, Private, Getting Started, Teamspaces, and Bryan workspace HQ. The main area is titled "Article page". It features a "Features" section with a list of items: Picture, Article title, Source (e.g. links), and Article content. Below this is a "Plan to do" section with a hint about including items to plan per day, an API field containing "https://www.thenewsapi.com/", and a "Done" section with a hint about including items felt done. To the right is a "Model template" section showing two versions of the article page: "Article page (Dark mode)" and "Article page (Light mode)". Both templates show a layout with Title, Source, and Content.

Kanban is a method for managing workflows and improving efficiency in Agile teams. Originating from lean manufacturing, Kanban uses a visual board to track the progress of tasks through various stages of development. It emphasizes limiting work in progress (WIP) to maintain focus, reduce bottlenecks, and deliver work incrementally. Teams pull tasks when ready, ensuring a smooth flow of work without overloading anyone.

In the provided diagram, the distribution of work and task delegation is evident. By utilizing Kanban methodologies, we aim to enhance work efficiency and ensure fair task allocation among team members.

White Box and Black Box Testing

Effective testing ensures the app is reliable, secure, and functional under real-world conditions.

White Box Testing

Involves testing the internal code structure of the app. White box testing helps catch issues like incorrect calorie calculations or inefficiencies in data retrieval.

We did this by inserting specific try catch blocks in the source code. With the relevant error and success print statements to verify the status of the function and tasks being tested. This can be seen in the clearly labeled comments in the source code.

Black Box Testing

Focuses on testing the app without knowledge of the internal code. Testers would verify the functionality from the user's perspective. For example, checking if users can successfully log meals, view calorie summaries, or set fitness goals. Black box testing helps identify issues with UI/UX flow or missed functional requirements.

We did this by testing the user interface and documenting the process in test case tables. The tables have a clear set of steps to take to test each function of the pages. With blackbox testing it ensures that the target user is able to use the function and also provides the developer more innovative ideas to improve the page to better cater to the needs of the user.

Black Box testing documentation (Home screen)

Test Case ID	Test Scenario	Steps
TC001	Succesfully renders HomeScreen2 component	Step 1. Click login button Step 2. Insert email and password Step 3. Click login button to insert Step 4. Display the home screen
TC002	Theme toggle functionality - Light Theme	Step 1. Locate the theme icon button Step 2. Click on the theme button Step 3. Notice app change to light theme Step 4. Icon Image change to sun
TC003	Theme toggle functionality - Dark Theme	Step 1. Locate the theme icon button Step 2. Click on the theme button Step 3. Notice app change to dark theme Step 4. Icon Image change to moon
TC004	Navigate to Articles screen	Step 1. Locate the list of articles Step 2. Browse the list of articles Step 3. Click on the selected article Step 4. Redirect to article screen
TC005	Navigate to Social screen	Step 1. Locate the add post button Step 2. Click on the add post button Step 3. Redirect to social screen
TC006	Daily calorie goal reached alert	Step 1. Locate the food navigation icon Step 2. Redirect to search food screen Step 3. Search for food item Step 4. Click on save food button Step 5. Navigate back to home and notice calorie reached alert
TC007	Daily calorie goal not reached	Step 1. Locate the food navigation icon Step 2. Redirect to search food screen Step 3. Search for food item Step 4. Click on save food button Step 5. Notice the daily calorie progress bar get updated with the new calories

User Centered Design

Stage	Activities	Purpose
Understanding the User	Surveys, interviews, usability tests	In our report we were able understand the needs of the user by observing trends as well as relying on studies done by trustable sources and organisations such as the National Library of Medicine. We carried out our own surveys as well together insights into user needs, behaviors, and preferences. We did this by creating google forms and concluded the feedback from the questionnaire and used that information to formulate the main areas of our application for the target audience.
Personas	Create detailed user profiles	Wei Ling: A 28-year-old working professional who wants to track her daily calorie intake but has very limited time. She prefers a minimalist interface and fast logging features. By representing different user segments and their specific requirements. By visualising user examples we are able to better cater to the needs of busy lifestyle by providing a simple onboarding process for our application.
Prototyping and Testing	Develop low-fidelity wireframes, gather feedback	Through continuous research and iteration, we improve by utilising the feedback and opinions of users of the app. We created google forms to gather feedback about designs and some user preferences that users might feel were lacking from the app as well.
Continuous Feedback	Implement feedback loops	Continuously improve the app by addressing user issues and introducing new features. We carried out this segment by creating a custom feedback page that users are able to input their feedback and rating of the current system. This provides the user with a route to voice their opinions as well as encourage the developers of the system.

STEEPLE analysis:

Factor	Impact on App	Recommendations
Social	Singaporeans are health-conscious but busy.	Focus on ease of use, quick food logging, personalized goals, and community features.
Technological	High smartphone penetration and strong internet infrastructure.	Leverage efficient data storage, technological integration, and prioritize a responsive user interface.
Economic	High cost of living.	Consider a free business model
Environmental	Promote sustainable eating habits.	Incorporate features for tracking the environmental impact of food choices.
Political	Government places importance on public health.	Align with government campaigns and food policies in the context of Singapore
Legal	Need to comply with data privacy laws.	Ensure user verification, secure data storage, and transparency in data usage.
Ethical	Avoid reinforcing unhealthy dieting behaviors.	Promote a balanced approach to calorie tracking and maintain transparency in data usage.

SWOT analysis:

Factor	Strengths	Weaknesses	Opportunities	Threats
Internal	<p>1) Wide range of nutritional information</p> <p>2) User friendly interface with a simple onboarding process.</p> <p>Tailored to busy individuals who want to quickly log meals</p>	<p>1) Limited resources for marketing, which could make it hard to reach a broad audience</p> <p>2) Initial lack of advanced features, such as a personal nutrition recommendations</p>	<p>1) Rising health awareness in Singapore due to government initiatives offers opportunities to promote healthier eating</p> <p>2) Growing interest in plant based diets and sustainable living provides an opportunity for features to track sustainable</p>	<p>1) High competition from international apps like myfitnesspal, which already have large user bases and advanced features</p> <p>2) Privacy concerns related to tracking personal health data may deter some users from adopting the app</p>

Discussion of Future Work

In conclusion, with these 5 areas of focus we plan to improve the fitness and healthy eating habits of busy Singaporeans. By providing a reliable and efficient app that they can use on a daily basis without much effort or frustrations. These 5 areas of focus include a comprehensive food database, community features, insightful articles for browsing and daily calorie tracker. Health is a big part of not only a higher life expectancy but also better mental well being.

Identify potential areas for future development and improvement:

Expanded Features:

- Recipe suggestions based on personal dietary preferences and restrictions
- Integration with popular grocery delivery services
- Meal planning for specific dietary needs such as for vegetarian, vegan, diabetic users

Enhanced Data Analysis

- Trends in food consumption and activity levels
- Personalized nutrition reports
- Comparison to national dietary guidelines

Community Building:

- Creating more interactive forums and challenges
- Facilitating peer support and motivation
- Integrating with social media platforms to expand reach

Integration with Other Health Apps:

- Explore opportunities to integrate with other health and fitness apps to provide a more comprehensive solution for users.

Research and Development:

- Continue to conduct research and gather user feedback to identify new opportunities for improvement and innovation.

Conclusion

By discussing future work, you demonstrate your commitment to ongoing development and your vision for the app's growth. This can help attract investors, partners, and users who are interested in a long-term solution for their health and fitness goals

References and citations

National Library of Medicine, nutrition profile of Singaporeans

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10282443/#:~:text=Conclusions,nutrient%20profiles%20and%20abdominal%20obesity.>

National Library of Medicine, calorie dense food and consumption from different ethnic groups

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8304763/>

Channel News Asia, hard to lower calorie consumption

<https://www.channelnewsasia.com/singapore/healthy-eating-food-paradise-singapore-salt-sugar-oil-cholesterol-high-blood-pressure-obesity-calories-wellness-weight-loss-lifestyle-nutrition-diet-3869756>

Straits Times Article, increase in sports activities among Singaporeans

<https://www.straitstimes.com/sport/sport-participation-at-an-all-time-high-in-singapore-gradual-increase-since-pandemic>

National Library of Medicine, Singaporeans work long hours and have little time to include exercise

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10291095/>

Food Industry Asia, increase in obesity in Singapore over the years

<https://www.evolveasia.org/the-state-of-obesity-in-singapore/>

Global nutrition report, Malnourished Singaporeans

<https://globalnutritionreport.org/resources/nutrition-profiles/asia/south-eastern-asia/singapore/>