LIFESTYLE TRACKER MOBILE APPLICATION

CHUA QING WEN

UNIVERSITI TUNKU ABDUL RAHMAN

LIFESTYLE TRACKER MOBILE APPLICATION

CHUA QING WEN

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

May 2022

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature :

Name : Chua Qing Wen

ID No. : 18UEB03834

Date : 06/05/2022

APPROVAL FOR SUBMISSION

I certify that this project report entitled **“LIFESTYLE TRACKER MOBILE APPLICATION”** was prepared by **CHUA QING WEN** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : ______________________________

Supervisor : Beh Hooi Ching

Date : 7th May 2022

Signature : ______________________________

Co-Supervisor : ______________________________

Date : ______________________________

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Ms Beh Hooi Ching and my moderator, Dr Lai Yen Lung for their advice and guidance throughout the development of the project. Their supports and experiences are invaluable to me to complete this project.

In addition, I would like to express my gratitude to my loving family members who had given me support and encouragement all the time during my degree studies.

ABSTRACT

In recent years, obesity and overweight have become the common issues among Malaysians, especially the university students and working adults. This is due to the sedentary lifestyle and unhealthy dietary habits adopted by the people. The situation is worsen during the COVID-19 pandemic as people are required to work and stay at home. The stress problem has also arisen during the pandemic, disrupting the mental health of people. Hence, this Android mobile application development project was implemented to help people to keep track of their lifestyle. The features and functionalities of the implemented system include adding, viewing, updating and deleting the meal and exercise records, uploading food photo for meal record, browse and save the desired recipes and articles, share the desired articles and update the personal profile information to recalculate the daily net calories and macronutrients limit. This project solution was implemented by using React Native framework and connected to the Flask REST API to retrieve and send the necessary information to the PostgreSQL database. The Flask REST API was deployed on Heroku to provide access for the implemented system through Internet. The selected development methodology for this project was incremental model, which iterated the design, development and testing phases module by module. Usability testing and user acceptance testing were carried out to assess the usability and user satisfaction with the implemented system. The overall SUS score gained by the implemented system was 94.5, which was higher than the SUS benchmark score set during the planning phase (75 out of 100). Hence, it could be concluded that the implemented system consists of a simple and attractive UI design and could function as intended. Although there are many improvements that can be made on the implemented system, this project is still regarded as a success because it met the project's objectives and scope.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

APA	American Psychological Association
API	application programming interface
BMI	body mass index
CGS	curved grading scale
ERD	entity relationship diagram
FAB	floating action button
IDE	integrated development environment
NCD	non-communicable chronic disease
SDLC	software development lifecycle
SUS	System Usability Scale
UAT	user acceptance testing
VCS	version control system
WBS	Work Breakdown Structure
XP	Extreme Programming

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

According to the Oxford Advanced Learner's Dictionary of Current English (1995), lifestyle is the manner in which an individual or a group of individuals lives and works. Having a healthy lifestyle should be the ultimate goal of everyone in the world as it can ensure a healthy body and happy life. However, obesity and overweight are the common issues faced by Malaysians due to the sedentary lifestyle adopted by people, especially university students and working adults. Not only that, unhealthy dietary habits and the stress problem are also the usual issues that can be found among the people, and these issues became worse during the COVID-19 pandemic. Therefore, this project was aimed to develop a mobile application to help users to track their lifestyle and to adopt a healthy lifestyle. The target users of this project are the university students and working adults.

The scope of this project includes a meal tracker, exercise tracker, recipe module to give a guide of the preparation of healthy food, article module to provide dietary, fitness and mental health care information and personal profile to update the personal information such as weight, height and the current activity level and the body mass index (BMI) will be recalculated. The implemented solution for this project is an Android platform mobile application which was developed using React Native framework with Javascript as the programming language. The mobile application uses the REST application programming interface (API) to access a PostgreSQL database. The PostgreSQL database is used to store the user's data. Moreover, the food nutritions and recipe information are retrieved from the related external APIs that consist of a large amount of data related to the nutritions and recipes of various foods available in the world. With the usage of APIs, the mobile application works properly with its implemented features.

This project is done according to the completion date which is April 2022 with the Lifestyle Tracker Mobile Application as the final and functional deliverable.

1.2 Background of Problem

Overweight and obesity are one of the most common health issues in Malaysia. According to the National Health and Morbidity Survey in 2019, there are 50.1 percent of the Malaysian adults are either overweight or obese. Obesity increases the risk of cardiovascular disease by 81 percent and stroke by 64 percent (Moy, Hairi and Wan, 2021). This is a serious issue that Malaysians should be aware of.

Most of the population in Malaysia are having a sedentary lifestyle. They often engage in physically inactive activities such as sitting for a long time to perform their daily tasks such as studying and working by using the computer. This results in lesser energy expenditure and increases the risk of health issues such as overweight, obesity and also musculoskeletal disorders such as back pain and neck pain (Daneshmandi et al., 2017). The sedentary behaviour is getting much more common, especially due to the COVID-19 pandemic. These issues are not only bringing the negative health impacts to the normal healthy people, but also those people with non-communicable chronic diseases (NCDs) (Marçal et al., 2020).

The eating habits of individuals have changed too. Homemade meals are getting rare as the preparation work is time consuming compared to cooking the processed food such as canned food and takeaway from the restaurant (Hassan, Sade and Subramaniam, 2020). Online food ordering service is the new trend discovered, especially during the COVID-19 pandemic. It provides convenience to the individual to order food by using a smartphone and the food will reach the doorstep in a short time (Ramli et al., 2020). This causes the individual to rely more on the ready made food, and they cannot control their calories and nutrition intake for their meals as every food purchased is fixed in terms of portion and ingredients.

Apart from that, according to the survey conducted by the American Psychological Association (APA) in the year 2020, there are huge impacts brought by the COVID-19 pandemic to the Americans and it affects in many aspects such as education, work, economy and the relationship with one another. A big percentage of adults that took part in the survey stated that the pandemic has become the largest source of stress to their lives, and the stress level is higher than the time before the pandemic (American Psychological Association, 2020).

1.3 Problem Statements

1.3.1 Increasing sedentary behaviour in daily life

University students and working adults tend to practise a sedentary lifestyle such as sitting for a long time in front of the computer to complete their daily task. This is even worse especially during the COVID-19 pandemic as everyone is studying and working from home and hence they are less involved in the active activities such as jogging, running and playing basketball. They will more adapt to inactive activities such as browsing the Internet and watching television and they are more likely to eat unhealthy snacks while they are watching the television show. This problem statement was supported by the systematic review done by Stockwell et al. (2021), which shows that 25 out of 45 healthy adults participated in the studies had reported an overall declines of physical activity level in term of time spent compared to pre-COVID-19 pandemic. In addition, there was 26 studies conducted to look into the changes of sedentary behaviour level of the adults, and all of the studies recorded an increasing level of sedentary behaviour among the participants of the studies during the COVID-19 pandemic and lockdown. Not only that, this problem statement was supported by the survey conducted as attached in Appendix C, which involved a total of 42 respondents with a mixture of university students and working adults. The questionnaire was created using Google Form and distributed to social media platform, RED (Xiaohongshu) and messaging platforms such as WeChat and WhatsApp. Referring to the results of the survey conducted as attached in Appendix C as Figure C-11, which have shown that about 81% of the respondents were having a low physical activity level or were physically inactive. Not only that, referred to Figure C-5 in Appendix C, among 18 respondents that were not satisfied with their current lifestyle, 10 of them were having the reasons that their lifestyle was not healthy and they were not so physically active nowadays.

1.3.2 Unhealthy eating habits

Since most of the people are staying at home due to the COVID-19 pandemic, there is a change in eating habits. They spend all the time staying at home to do their work and studies, and there is no sufficient time for them to prepare a healthy meal. This is especially true for the university students and working adults who stay alone at outstation. They will rather settle their everyday meal by ordering food from the online food delivery platform, and they cannot control the portion and the ingredients used of the meal ordered. This problem statement was supported by the studies conducted by Ramli et al. (2020) with a total of 362 respondents who are the university students from Kuala Terengganu, which shows that the majority of the respondents agreed with the convenience brought by online food delivery service, which allows the users to order food at home during the lockdown period, and they can control over their time by making order whenever they wish when they dealing with their busy study schedule.

1.3.3 Increasing stress level that affect the mental health

Other than the physical health issues, there are psychological health issues that arise such as the sense of fear, anxiety and stress during the COVID-19 pandemic. This is due to the weak social connection with one another and people tend to overthink when staying at home for a long period of time. The stress level of adults is increasing due to the drastic changes that occur in their daily life that affect in many aspects as mentioned in Section 1.2. This problem statement was supported by the research conducted by Marzo et al. (2021), which indicates that among 1544 Malaysian participants of the study, 43.5% of them had mild to extreme depression problem and 70% of them had moderate to extreme depression problem during the third wave of COVID-19 pandemic in Malaysia. This shows an increasing trend of depression problem as compared to the first wave of COVID-19 pandemic, which was only around 14.6% to 48.3% of the participants reported they had depression problem of mild or moderate to extreme level. This problem statement was also supported by the results of the survey conducted as attached in Appendix C as Figure C-12, which have shown that 25 out of 42 (59.5%) respondents were facing the symptoms of mental disorder nowadays, and referred to Figure C-13 in Appendix C, among the respondents who were facing the symptoms, 68% of them were encountering the stress problem, and there were some of them encountering sleeping problem (32%), depression (20%) and anxiety (12%).

1.4 Project Goal and Objectives

1.4.1 Project Goal

To implement a mobile application that helps users to adopt a healthy lifestyle by April 2022.

1.4.2 Project Objectives

1. To analyze project requirements and review existing similar lifestyle tracker applications by 22 August 2021.

2. To develop a Lifestyle Tracker mobile application on the Android platform for the purposes of tracking and improving the lifestyle of users by April 2022.

3. To validate the application's usability by achieving 75 out of 100 on System Usability Scale (SUS) by April 2022.

1.5 Project Solution

The system architecture of the implemented system is depicted in Figure 1.1. The goal of this project is to build an Android-based mobile application. React Native framework was used to develop the mobile application as it uses Javascript as the programming language to develop the mobile application which could be the alternative of Java and Kotlin that used to develop a native Android application. The mobile application developed consists of several modules which allow to build up a complete Lifestyle Tracker mobile application, that are meal tracker, exercise tracker, recipes, articles and personal profile. The user's data such as name, password, weight, height and BMI are saved in the PostgreSQL database with the help of REST API. The REST API was written using Flask microframework. The REST API helps the mobile application to access the PostgreSQL database by receiving the HTTP/HTTPS requests from the mobile application and sending back the data in JSON format. The other external APIs such as Spoonacular Nutrition, Recipe, and Food, Article Data Extraction And Text Mining and Fitness Calculator APIs are used to retrieve the food nutritions, recipe information, article contents and the daily macronutrients of the user. The information retrieved will display to the user when the user navigates to the respective screens in the mobile application or when the user creates an account or updates his/her personal information such as his/her goal of using the lifestyle tracker mobile application as well as his/her current activity level. With the integration of APIs, the implemented system is able to perform the necessary functionalities.

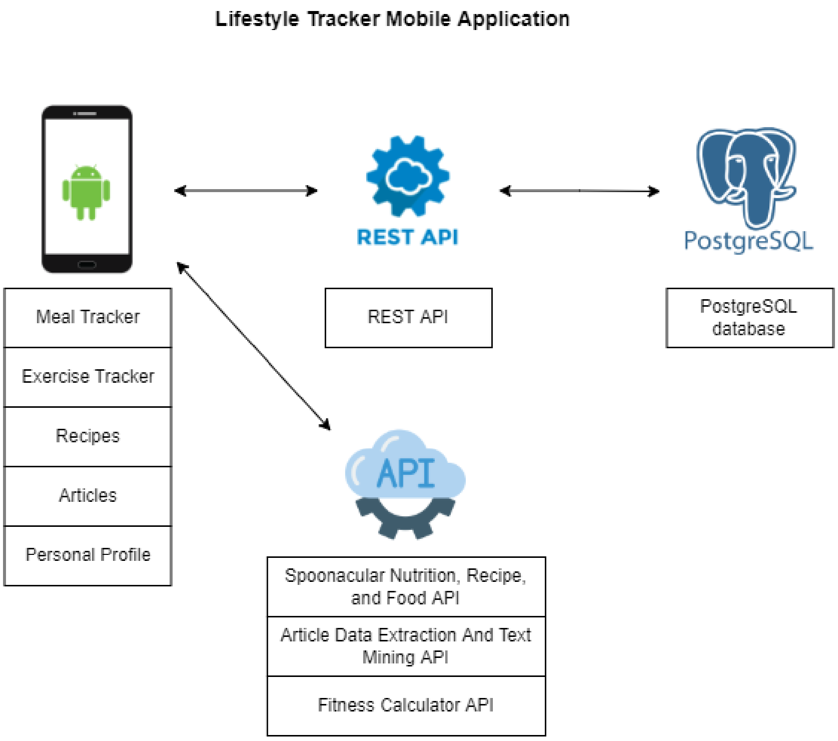


Figure 1.1: The system architecture of the implemented system.

1.6 Project Approach

1.6.1 Research Approach

The quantitative research approach was applied in this project. The data gathered and collected by using the quantitative research approach was about the numerical data and statistics which can easily convert into graphical representation such as graph and chart in order to have better insights of the research (Gounder, 2012). The samples collected were in large amounts and this enhanced the analysis work of the project such as to understand more clearly about the possible needs of a population. Although quantitative research approaches are often reliable due to the large quantity of data obtained, there are different responses gained because everyone has their own unique life experience and they are not having the same point of view to the respective research area. There are some people who are not competent in expressing their opinions and thoughts and hence lead to possibly misinterpretation of data from the team who are working on the project.

Questionnaires were used in this project to collect sufficient data and opinions from the target users regarding their lifestyle, problems they faced that were related to physical and mental health and also their expectations on the lifestyle tracker mobile application. From the responses of the questionnaires, charts were generated and all sorts of suggestions can be viewed at a glance. It was important to get the user's input to further support the problem statements and scope of the project, so that the application developed are able to meet the user requirements and also solve the current problems faced by the target users, which are the university students and working adults.

1.6.2 Development Approach

The development approach used in this project was the incremental model. The incremental model of software development lifecycle (SDLC) is the methodology that breaks the requirements into multiple parts which are the modules of the project. The main module was addressed in the first iteration, and the other modules were developed in the next iterations. By adopting the incremental model as shown in Figure 3.1 during the SDLC processes, the first version of software with the initial core function was finished in a short time, and the additional features were added into the software incrementally.

The incremental model is more flexible to handle the changing requirements as it performs the design, development and testing in every iteration to make improvements and add other features. The activities in each iteration were done simultaneously and if anything went wrong, the developer can still go back to the previous activities to make changes according to the testing result and the client's feedback. Another advantage of the incremental model is that the error in the code can be easily detected because the software was developed part by part with multiple iterations and error was fixed in the testing phase of each iteration. The iterations continued until a final software product was built (Sommerville et al., 2011).

1.7 Scope of the Project

1.7.1 Target Users

The target users would be ranging from university students to working adults, as most of them are trying to achieve the best outcome from either academic career or working career and do not put a lot of concern in maintaining and achieving a healthy lifestyle. They are quite busy in their daily studying or working schedule, and they spend most of their time sitting in front of a computer and this increases the risk of getting obesity and musculoskeletal disorders. Besides, they have no concern on what they eat and they have no much idea on how to control their daily meal intake. Since they are always immersed in a high stress environment in order to complete their task, they will most probably be facing psychological health issues such as anxiety, fear and insomnia. By using the lifestyle tracker mobile application, they can have a better way of monitoring what they eat, getting access to the food recipe and blog that shares about the tips of maintaining a healthy lifestyle including the tips to take care of their dietary, fitness and mental health.

1.7.2 Modules

1.7.2.1 Meal Tracker module

The user can log his/her meals in this module. The user can search for the meal in the search bar and the matching meals will be displayed. The respective nutrition information of the selected meal such as calories, carbohydrate, protein and fat will be displayed. After the user added the meal into his/her meal log, he/she can either capture the meal's photo or choose the meal's photo from the library and add it into the meal record as reference. The net calories will be recalculated by adding the calories intake into the net calories.

1.7.2.2 Exercise Tracker module

After analyzing the survey results collected from 42 respondents as attached in Appendix C, exercise tracker module is eventually become one of the modules in the implemented system. The user can log his/her exercises in this module. The user can search for the exercise and a list of matching exercises will be displayed. The user can choose the specific exercise and select the duration of exercise activity they have performed. The respective calories burnt will show to the user. After the user added the exercise record into his/her exercise log, he/she can edit the exercise duration and perform save action. The net calories will be recalculated by subtracting the calories burnt from the net calories.

1.7.2.3 Food Recipes module

The user can browse and search for the food recipes in this module. The main page of the food recipes module consists of two categories of recipes, which are the diet preferences and the cuisine types. The diet preferences include vegan, vegetarian, pescatarian, ovo-vegetarian and lecto-vegetarian, while the cuisine types involve Chinese, American, Thai and many more. The user can search for the food recipe using the search bar and a list of matching food recipes will be displayed. The user can click on the specific food recipe to view the recipe details and he or she can choose to save it as his/her favourite recipe and refer to it whenever needed.

1.7.2.4 Articles module

The user can obtain health-related articles such as dietary, fitness and also mental health from this module. The articles can be saved by the user as his/her favourite articles and he/she can choose to share the articles to social media platforms.

1.7.2.5 Personal Profile module

The user can see his/her personal information such as his/her name, weight, height, goal, activity level and BMI in this module. User can update his/her personal details and the BMI, daily net calories and macronutrients limit will be updated accordingly. The lists of saved recipes and articles can be found in this module.

1.7.3 Scope exception

This project was covered only the lifestyle related to dietary and fitness. Other lifestyles that are related to working style, spending behaviors and culture were not covered. Besides, this project developed the mobile application only on the Android platform.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Malaysia is recorded as having the highest overweight and obesity rate among the adult population compared to other South-East Asia countries, which hold 15.6 percent out of 100. This is not surprising because the overweight and obesity issues have been common among Malaysians for decades. According to the findings of the National Health and Morbidity Survey in year 2019, it is shown that almost half of the adults and 30 percent of the children in Malaysia are either overweight or obese. The risk of getting NCDs by overweight and obesity people is higher than those with normal weight. This is an alarming problem as the NCDs patients are having the higher possibility of getting premature death. Apart from this, the NCDs patients are those who are more vulnerable to the COVID-19 virus infection, and might having serious symptoms and require hospitalization compared to those with normal weight (Anon., 2020). Therefore, it is important for everyone to adopt a healthy lifestyle in order to prevent getting overweight or obesed.

In this chapter, research was done on the importance of a healthy lifestyle by reviewing the advantages that can be acquired from adopting a healthy lifestyle and also the consequences of having a poor lifestyle. Besides, the development methodologies and usability testing methods were analyzed and the most suitable methodology and method were selected as the way to implement a mobile application to help people to adopt a healthy lifestyle. Subsequently, the existing similar application review was conducted to find out the common and unique features that are available in the existing systems.

2.2 Importance of Healthy Lifestyle

The lifestyle is defined as the way of the individual or group of individuals living. It includes many aspects such as habits, thoughts, culture, working style and spending behaviours. A healthy lifestyle means that the individual or group of individuals are having a lifestyle that will bring positive impacts to them. By having a healthy lifestyle, an individual will possess a healthy body with good physical and mental health, and this is considered as the ultimate goal for everyone in order to live longer.

People’s lifestyle has experienced a dramatic change due to the COVID-19 pandemic. The daily schedules of people have been disrupted to adhere with the home confinement policy, leading to the unhealthy and sedentary lifestyle as everyone is practicing work from home and studying from home (Zheng et al., 2020). In addition to this, the research conducted by Koci et al. (2021) found that this pandemic has affected everyone that generates significant social and psychological difficulties due to the fast rate of virus spreading. According to several Chinese studies, feelings such as anxiety and stress are having substantial growth in the group of students and medical professionals and also general people. People have increased negative feelings and thoughts as a result of the huge changes happening in their lifestyle. Those people who have suffered from psychological disorders have experienced deterioration of their current health status as they feel more stress during COVID-19 pandemic. As well as that, based on the studies by Zheng et al. (2020), 37% of the participants mentioned that they are having a higher stress level and causing poorer sleeping quality because they feel worried of the pandemic situation. Apart from the mental health condition, there is a statistics showing that one-third of the adults across the world are physically inactive and 45 percent of them are having daily sitting time of more than 4 hours before the COVID-19 pandemic. During this pandemic, there is a significant decline in physical activity levels among Australia adults, and a study conducted on the Canada adults reported that 40.5 percent of them who are always physically inactive have become more inactive during the COVID-19 pandemic. Undoubtedly, the way of people living nowadays is completely different compared to the pre-pandemic period, and people require time to adjust both their physical and mental conditions in order to live healthily.

Looking from another perspective, Pell et al. (2016) reported that the increasing rate of overweight and obesity issues among Malaysians is due to several reasons such as the background of individuals, eating habits, physical activity level, and also the evolution of Malaysia towards urbanization and industrialization. Malaysia is among the countries in Asia that have the most people having a high consumption of fat and oil. This may be due to the rapid development of western fast food chains in Malaysia and fast food has become popular among young people despite it being expensive. In fact, the prevalence of obesity and overweight has a relationship with the transition from adolescence to young adulthood. The cumulative eating habits and physical activity level from a young age may be the reason for overweight and obesity in adult life. Despite the obesity issues being quite common among Malaysians due to the poor eating habits and unhealthy lifestyle, the studies by Al Mamun, Hayat and Zainol (2020) showed that the Malaysian young adults have the health awareness slowly building up. This is due to their positive perspective towards a healthy diet that leads them to have the intention of having healthy eating habits. More and more young adults are adopting healthy eating habits because they are having more knowledge about health and they would like to prevent health diseases such as obesity and cardiovascular disease.

There are negative impacts from practising an unhealthy lifestyle. Unhealthy lifestyle includes physically inactive and poor eating habits for a long period of time. The individual who is having a sedentary lifestyle will result in cardiovascular disease, diabetes, cancers, weak bone and muscles, and also spine problems. Long hours of inactive status by sitting in front of TV and computer may cause sleeping and headache problems as well because the body is always in a tense condition. Moreover, because of spending most of the time doing something by sitting for a long period of time, one might have developed bad eating habits by intaking unhealthy food with high fat and salt intake which might increase the risk of overweight and obesity due to lack of physical activities performed (Kumar, 2017).

Consequently, a healthy lifestyle needs to be practiced by people as it brings a lot of benefits to people's physical and mental health, and saves medical costs. According to Kumar (2017), a healthy lifestyle is a choice for people to adopt various kinds of health-related behaviours so that people can prevent getting disease, reduce stress levels, build up a healthy and positive self image and so on. By having a fit and healthy body, people can live longer and have a high quality of future life.

In conclusion, a healthy lifestyle needs to be adopted by everyone to have a healthy body and better quality of life. Sedentary behaviours need to be reduced and people need to try their ways to change their lifestyle gradually by having some light exercises in the beginning, and increase the intensity of exercise once people are having exercises as their regular activities to be performed daily or weekly (Kumar, 2017). Furthermore, healthy eating habits need to be practised to reduce the risk of getting NCDs and other diseases which might cause high possibility of premature death. Lastly, people should take care of their mental health conditions during the COVID-19 pandemic as poor mental health conditions will affect their physical health. Koci et al. (2021) reported that the people who suffer from the mental health disorders will lead to poor physical health which may cause the one to easily have illness. Some protective actions that help the psychological disorder from getting serious include the habits of sharing concerns with the loved ones, having regular exercise and the satisfaction towards the employer in facing the COVID-19 pandemic. Not only that, everyone should have the correct and positive mindset to face the pandemic, and hope for the best.

2.3 Development Methodologies

Before implementing a system, the planning stage is required. The planning stage of SDLC is not only collecting the requirements from the users, but also deciding on which development methodology to be applied in the project. There are numerous development methodologies for a software project, and choosing the most suitable one will aid the success rate of a software project. Different development methodologies have different characteristics and bring divergent advantages and disadvantages depending on the nature of the project. In this section, various development methodologies are categorised into 2 big categories, which are the traditional methodologies and agile methodologies.

2.3.1 Traditional Development Methodologies

The first traditional development methodology to be reviewed is the waterfall model. Waterfall model is the sequential top-down model. It means that the next phase of SDLC will only start once the current phase has been completed. The biggest advantage of the waterfall model is that the documentation produced is very comprehensive. However, the final software product will only be produced at the end of all the processes in the waterfall model.

Waterfall model is suitable to be used for the project with minimal uncertainties as it is a systematic model. In each phase, there will be some specific deliverables, but due to each phase in the process not interleaving, the changing requirements at the later stage are difficult to be adapted into the project (Saranya et al., 2017). Figure 2.1 depicts a waterfall model diagram and it indicates the phases are not parallel in the whole SDLC processes.

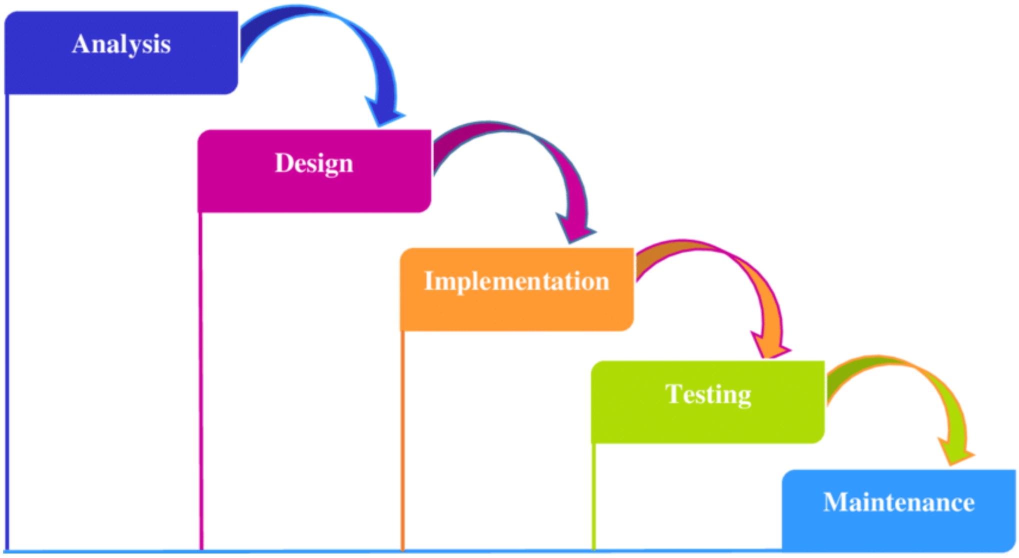


Figure 2.1: Waterfall model diagram (Senarath, 2021).

The second traditional development methodology to be reviewed is the incremental model. The incremental model will break the clearly defined requirements into smaller parts, and each increment will work on the specific module which the core module will be implemented first. By using an incremental model, it is easier for the developers to test and debug the software in each iteration, but the project which adopts the incremental model must have gone through the good initial planning, analysis and design phases so that the focus of the iterations can be determined.

The incremental model is suitable to be used when there is a need to show a workable product to the stakeholder after each iteration. The incremental model allows the development team to revise the module in each iteration and build up the software stage by stage. It allows early software product delivery as each iteration will produce a workable software product. Still, by using the incremental model, the initial planning, analysis and design stage must be well defined and stated to ensure the smooth transition of the software development process (Saranya et al., 2017). Figure 2.2 shows the diagram of the incremental model.

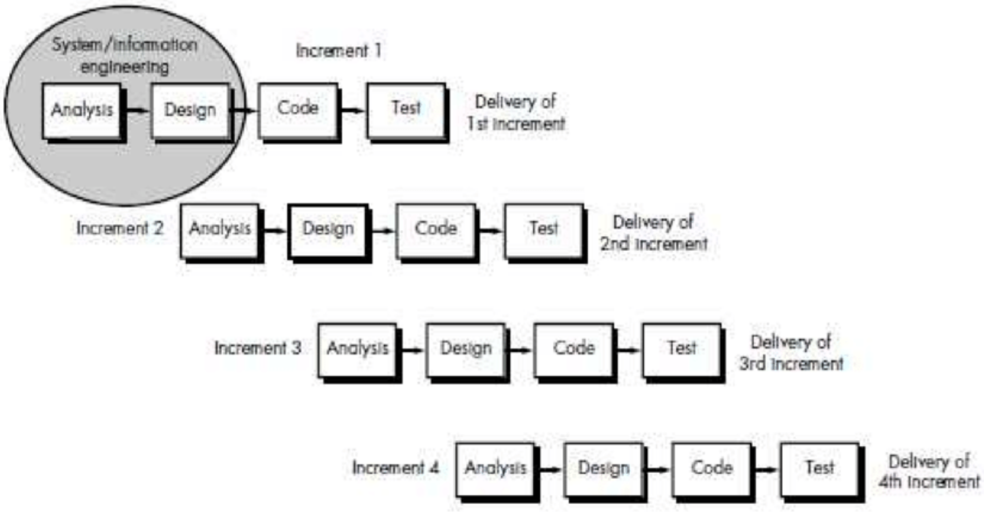


Figure 2.2: Incremental model diagram (Olajide, 2016).

The third traditional model to be reviewed is the spiral model. The spiral model is an iterative model which will repeat the planning, risk analysis, development and evaluation phases until getting a software product that meets the user requirements. It has a risk analysis phase which allows the risk evaluation process to be conducted in the early stage of the project. If the project is considered as high risk, then the clients can know about it earlier and decide whether to continue or abort the project. In fact, the spiral model is the combination of the waterfall model and the unified model as it applied the concept of stages from the unified model and also the concept of sequential processes from the waterfall model. The next iteration will only start after getting feedback from the stakeholders and the next iteration will make modification and improvement according to the feedback obtained (Shaikh and Abro, 2019).

The spiral model is suitable to be used for the high risk project as it will evaluate the risk of the project earlier and compare the alternative project in terms of risk level. The downside of the spiral model would be the extra cost incurred by conducting the risk analysis process. Figure 2.3 shows the diagram of the spiral model.

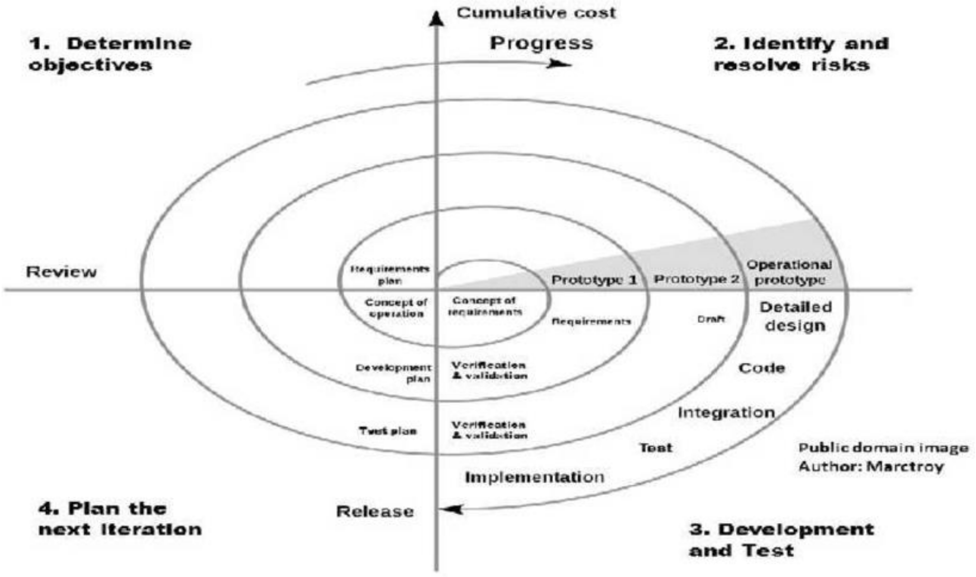


Figure 2.3: Spiral model diagram (Olajide, 2016).

2.3.2 Agile Development Methodologies

After reviewing the traditional development methodologies, a few of agile development methodologies were discussed in this subsection. One of the famous agile methodology would be the Extreme Programming (XP) model. The XP model is an iterative model, but with some best practices to be adhered to by the development team. This model emphasizes on the frequent small release in every iteration and also effective communication between the development team and also the stakeholders. After the software product has been implemented in every iteration, automated tests are performed to check for the quality of the software product. This helps to minimize the risk of having the errors or bugs at the end of the software development processes.

The concept of pair programming is being applied in the XP model. There will be two developers collaborating together to complete a certain part of the software product. One is responsible for writing the code, while another developer will need to review the code and spot out the bugs. Moreover, by adopting the XP model in the software project, there will need to have a time agreement by all the developers on the duration of coding per week so that the software product can be done within the time frame given. In order to produce a clean and effective code, a certain coding standard is set and to be followed by the developers to aid in enhancing the readability and shorten the time required for other developers to pick up the software project in another iteration of software development (Yasvi, Yadav and Sahendrasingh. S., 2019).

The XP model is suitable to be used for the small scale software project. This is due to the longer time needed to have two-way communication between developers and the stakeholders in order to get feedback and make enhancement on the software product. With the condition that the time allocation is not sufficient, then a more systematic model should be embraced such as the waterfall model, instead of the XP model. Figure 2.4 shows the diagram of the XP model.

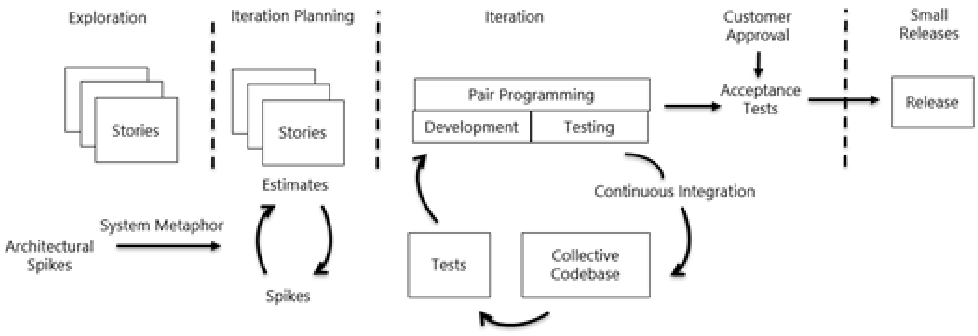


Figure 2.4: XP model diagram (Meier, 2019).

Next, the scrum model is another agile development methodologies that has started to gain popularity in recent years. The scrum model focuses on the user requirements and will break them as sprints. Each sprint consists of some features and is assigned to the scrum team. The scrum team will need to complete each sprint within 1 to 3 weeks. Not only that, they will need to have a daily scrum to report their development progress. After completing the sprint, the workable software product will be reviewed by the product owner and the review will lead to either bug fixing or the next sprint to continue developing the next part of the software product (Srivastava, Bhardwaj and Saraswat, 2017).

The scrum model is suitable to be used for the project that requires high productivity rate and flexibility, and still to maintain a structure of the SDLC processes. The scrum model can help to deliver a high quality software product with high customers’ satisfaction due to the flexibility given to plan the sprints workload. The daily meeting during the sprint assists in figuring out the problems faced by every member in a scrum team, and as a result, the scrum team can solve the potential problems as soon as possible. By clearing out the bugs in each sprint, the project cost can be reduced because it is always cheaper to fix the error before the software product is delivered. Figure 2.5 shows the diagram of the scrum model.

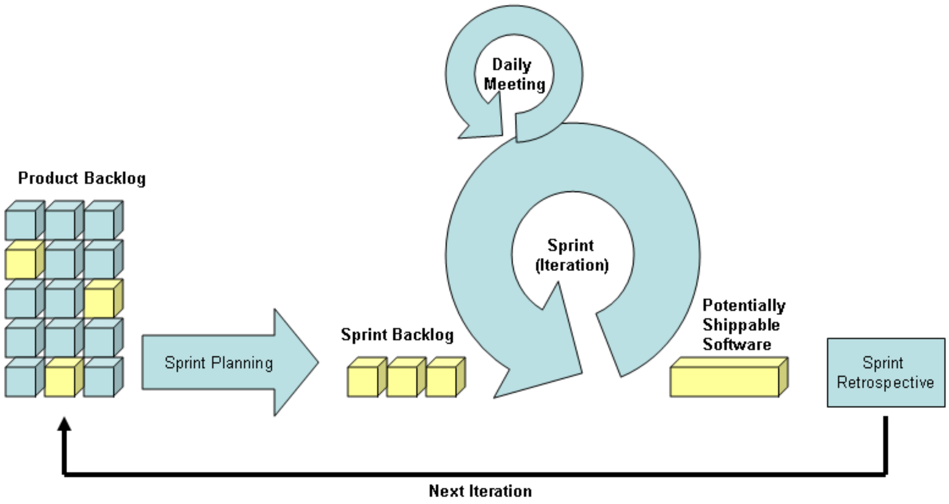


Figure 2.5: Scrum model diagram (Correa, 2020).

2.3.3 Comparison of Traditional and Agile Development Methodologies

There are strengths and shortcomings for both traditional and agile development methodologies. Traditional methodology is said to be more suitable for a large scale project due to its systematic structure and less communication time required as compared to the agile methodology. However, according to the findings from Shaikh and Abro (2019), the software development teams that adopt the traditional methodology are still facing some challenges in managing the complexity of the large project. Hence, it is important to keep the documentation and code simple and clear to ensure the project success and reduce the maintenance cost in the future.

Next, for the project that adapts the traditional methodology, it is more focused on the process and everyone is assigned to a role with the expectation of them having their work done perfectly without understanding their interests and capabilities. The agile methodology, on the other hand, requires the team members to be talented and skillful to handle the development process as to tackle the problem in the traditional model which needs a comprehensive documentation to guide the development process (Shaikh and Abro, 2019).

Despite the fact that the agile methodology may help to drive the project to success in a rapid manner, it is not suitable for the projects under government and some businesses which will run in a long term and require better maintenance service. This is because the agile methodology does not emphasize the completeness and details of the documentation.

To conclude, the choice of the software development methodology depends on the type of the project and also some other factors such as budget and planned completion time. This statement is agreed by Saranya et al. (2017) and Shaikh and Abro (2019) as there is no universal methodology that can be used for all the software projects.

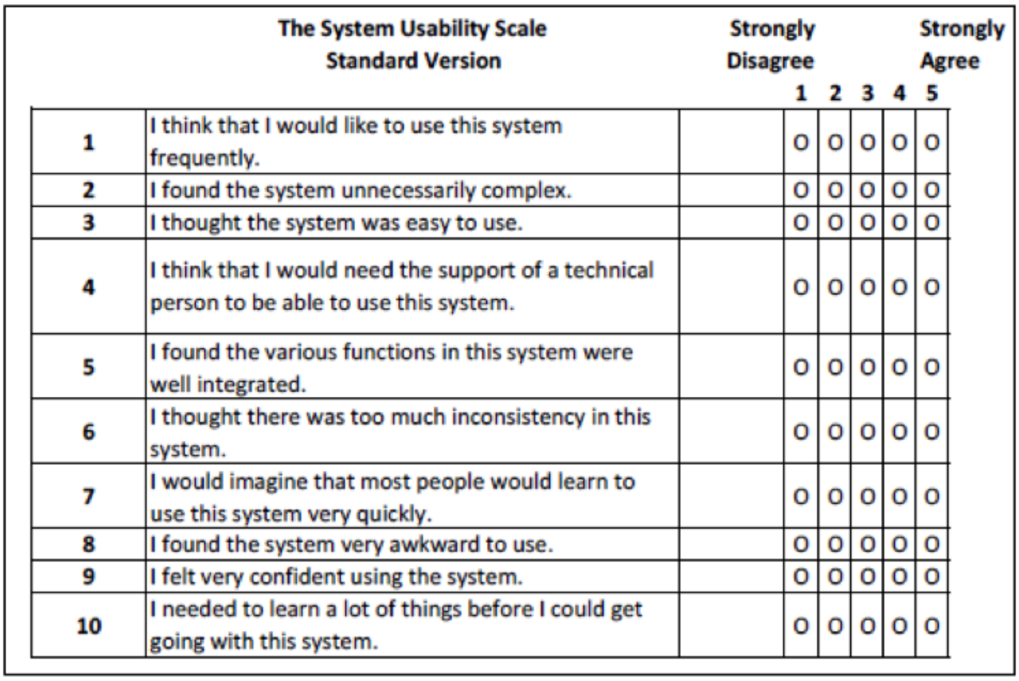
2.3.4 Conclusion

Following a review of all of the software development methodologies listed above, the incremental model under traditional methodologies was chosen to apply for this project for a variety of reasons. Firstly, this project has only 1 person that handled all the SDLC processes, which are planning, analysis, design, development, testing and deployment. It was best to have the increments concept to aid in implementing the project solution so that the final mobile application was slowly built up after several increments. The modified incremental model adopted in this project is shown in the development approach in Chapter 1. The planning, analysis and initial design phases needed to complete before entering the first increment to revise the design phase for the specific module. After revising the design, development work was started and testing was conducted to evaluate the work of the increment. Once the testing phase was done and all the bugs were fixed, the next increment started to repeat the design, development and testing phases again to build up the specified module. Moreover, another reason to adopt the incremental model in this project was because of the time constraint to complete this project. The incremental model can help to build the project solution more faster and in an organized manner. It does not require frequent feedback from the target users during the SDLC processes as compared to the agile methodologies.

2.4 Usability Testing Method

A successful software project not only requires to have a most suitable software development methodology being adopted, but also needs to pass all the testing phases such as unit testing, integration testing, system testing and also acceptance testing. Apart from these testings, the usability testing will be performed to review the usability of the system developed. A software product with a good usability means that it is easy to use and learn in terms of the system design. There is a famous usability testing method used for many software projects, which is SUS. In this section, the SUS method will be discussed to understand how it works in evaluating the usability score of a software.

The SUS method is used to assess the perceived usability of the software by having a pre-defined questionnaire which consists of 10 items as shown in Figure 2.6 (Lewis and Sauro, 2018). The SUS score is a composite usability measure computed from having all the score of the items, and when there is an empty SUS rating for the item, it will fill in with the average score of 3. There is a specific formula used to calculate the SUS score, and it is quite complicated to calculate the SUS score manually (Lewis, 2018).



	The System Usability Scale Standard Version	Strongly Disagree					Strongly Agree
			1	2	3	4	5
1	I think that I would like to use this system frequently.		○	○	○	○	○
2	I found the system unnecessarily complex.		○	○	○	○	○
3	I thought the system was easy to use.		○	○	○	○	○
4	I think that I would need the support of a technical person to be able to use this system.		○	○	○	○	○
5	I found the various functions in this system were well integrated.		○	○	○	○	○
6	I thought there was too much inconsistency in this system.		○	○	○	○	○
7	I would imagine that most people would learn to use this system very quickly.		○	○	○	○	○
8	I found the system very awkward to use.		○	○	○	○	○
9	I felt very confident using the system.		○	○	○	○	○
10	I needed to learn a lot of things before I could get going with this system.		○	○	○	○	○

Figure 2.6: The standard SUS questionnaire consists of 10 items (Lewis and Sauro, 2018).

The evolution of SUS starting from the formation of 50 statements by Brooke in the year 1996 and lastly the specific 10 items of the mixture of positive and negative tone has been fixed. The purpose of having the mixture of tones in the SUS questionnaire is to eliminate the bias that happens when the users filling the questionnaires. However, most of the unexpected SUS scoring is not caused by the bias, but the human error when filling the questionnaire or to calculate the SUS score by using the complicated formula. According to the conference of the Usability Professionals Association in the year 2004, when comparing various conventional usability surveys, SUS was the first to reach a definite determination about the selected system. SUS was proved to be efficient in measuring the usability (Lewis, 2018).

In addition, Lewis (2018) emphasized the categorization of SUS score by using a curved grading scale (CGS) by Sauro and Lewis, which is to normalize the distribution of SUS scoring. The average score of CGS obtained from the data set from 446 studies and over 5000 single SUS responses was 68, which is a grade C. A system with an above average user experience in the industry will need to achieve a score of at least 80, which is a grade A-. Nevertheless, the mean SUS varies depending on the type of the system, as some of the systems are developed for daily usage and it will have a higher SUS mean score due to the simplicity of the system. While for those systems which require more features and are designed for the usage of professionals, then it will have a lower SUS mean score as most of the people will have difficulty in using the system without having proper training. In contrast, Sasmito, Zulfiqar and Nishom (2019) categorized the SUS score by using the SUS adjective rating. In order to use SUS in measuring the usability of a system, the practitioners will need to deploy the questionnaire, calculate the SUS score, and finally classify the SUS score into an adjective rating. The SUS score which is higher than 80.2 is considered as an excellent product, while for a good product, it needs to achieve a minimum score of 68. As a matter of fact, a good software product will need to achieve the same score in both CGS and SUS adjacent ratings, which is 68.

After understanding the concept of SUS and the average SUS score for a good software project, it is important to know how to utilize the SUS questionnaire to measure the usability of a software system. By using the standard SUS questionnaire, the practitioners should adhere to the standard procedures in order to get the outcome with no error, unless there are human mistakes of either the user has checked the wrong scale or the practitioner has miscalculated the SUS scoring. Adding to this, the practitioners can also modify the SUS questionnaire by either removing one of the items or switching the item tone to become all positive. If the practitioners alter the SUS questionnaire, it is important to make adjustments to the formula used to calculate the SUS scoring. In truth, the SUS questionnaire is quite flexible in that the practitioners can make the items become the extreme statement. The extreme items will cause the mean SUS score to be either very high or very low depending on the extreme tone which is either positive or negative (Lewis, 2018).

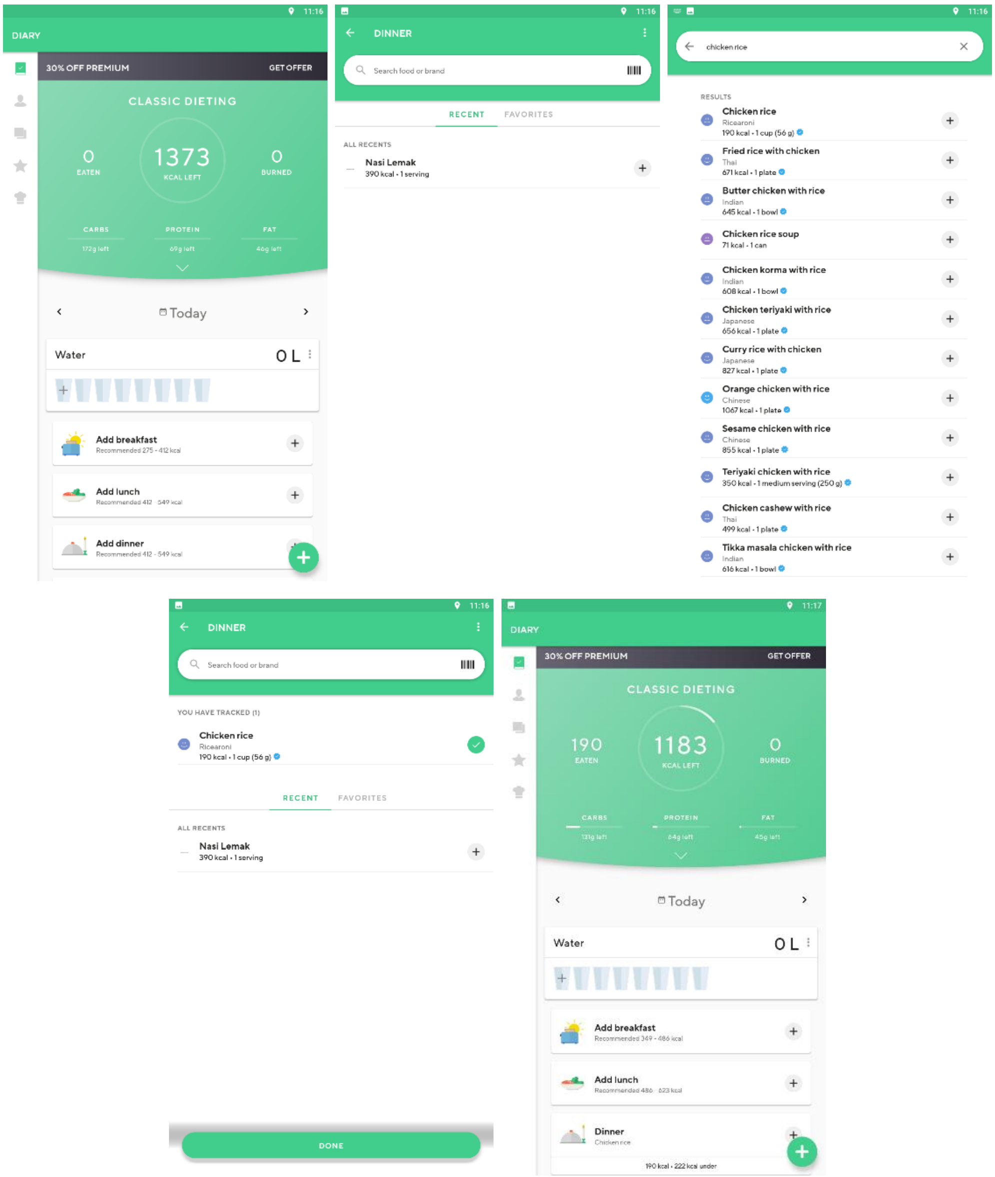
As a conclusion, by highlighting the usability of the system, the system developed will be more easy to satisfy the user’s needs, and the usability testing would get better results that help the developers team to gain confidence about the system built. The SUS method is a powerful usability measure technique that now requires people to make translations to many other languages with the use of accurate terms to represent the current items in the SUS questionnaires. The SUS needs to expand its usage to not only measure the usability of the system, but also to measure other criteria such as the visual representation, usefulness, trust and loyalty level on the outcome of the system. This is to make sure that the usability testing becomes more comprehensive and can guide to the correct path of system’s usability improvement (Lewis, 2018).

2.5 Review of Existing Similar Systems

A few existing similar systems were selected to be reviewed and analyzed to study the common features, differences and the unique functionalities that are available in these lifestyle mobile applications. These mobile applications were selected from the health and fitness category based on the highest ratings received from the users of Google Play Store. An application review matrix had been created latest by 27 July 2021 to record down the information of these existing similar mobile applications such as the basic information of the application, the description, features available in both free and premium (paid) versions of the application and also some useful user reviews after using the application. This application review matrix has been attached as Appendix A.

2.5.1 Comparison of the Existing Similar Systems

The selected existing similar systems were Carb Manager, Lifesum, MyPlate Calorie Tracker, and ekilu. The reasons of selecting these four existing systems to do comparison were the higher downloading rate, positive users' reviews and the most important reason was they are having some common features that the lifestyle mobile application would have, such as meal tracker, exercise tracker, food recipes and meal planner. The details of the selected existing similar systems were tabulated in Appendix A such as the downloading rate, useful reviews and the features provided to the users latest by 27 July 2021. However, there is limited access for the food recipes features for these 4 mobile applications and Lifesum and MyPlate Calorie Tracker require the users to upgrade the mobile application to premium version, then only can access to the food recipes. Same goes to the meal planner features, all of the reviewed existing similar systems have made the meal planner as one of the features in the premium version. Figure 2.7 shows the sample screens of the meal tracker, while Figure 2.8 shows the sample screens of the exercise tracker. These figures were attached to give a clearer picture of how these tracker features work and how are the screens flow.



RESULTS

Chicken rice
Ricearoni
190 kcal • 1 cup (56 g)

Fried rice with chicken
Thai
671 kcal • 1 plate

Butter chicken with rice
Indian
645 kcal • 1 bowl

Chicken rice soup
71 kcal • 1 can

Chicken korma with rice
Indian
608 kcal • 1 bowl

Chicken teriyaki with rice
Japanese
656 kcal • 1 plate

Curry rice with chicken
Japanese
827 kcal • 1 plate

Orange chicken with rice
Chinese
1067 kcal • 1 plate

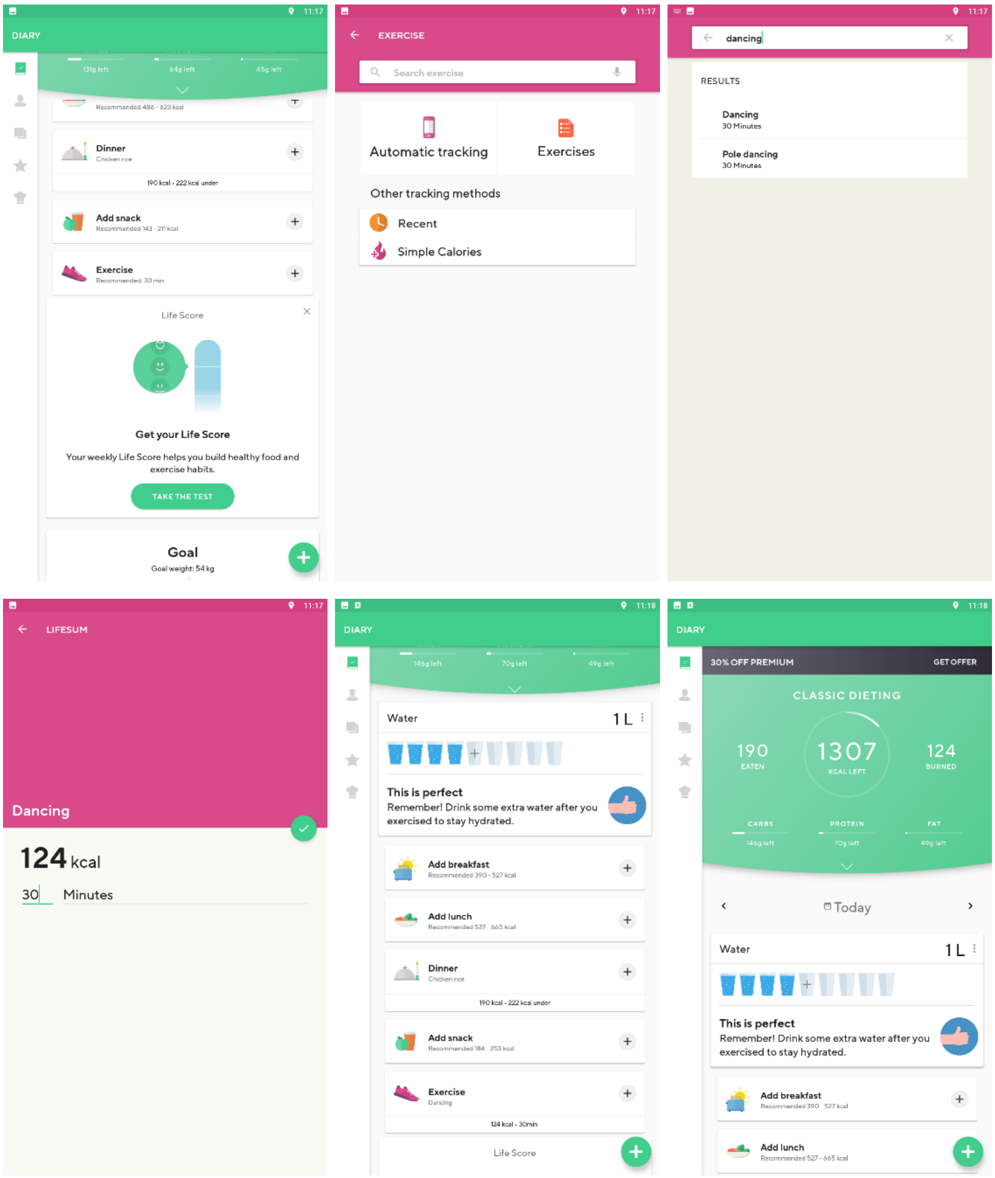
Sesame chicken with rice
Chinese
855 kcal • 1 plate

Teriyaki chicken with rice
350 kcal • 1 medium serving (250 g)

Chicken cashew with rice
Thai
499 kcal • 1 plate

Tikka masala chicken with rice
Indian
616 kcal • 1 bowl

Figure 2.7: Sample screens of meal tracker.



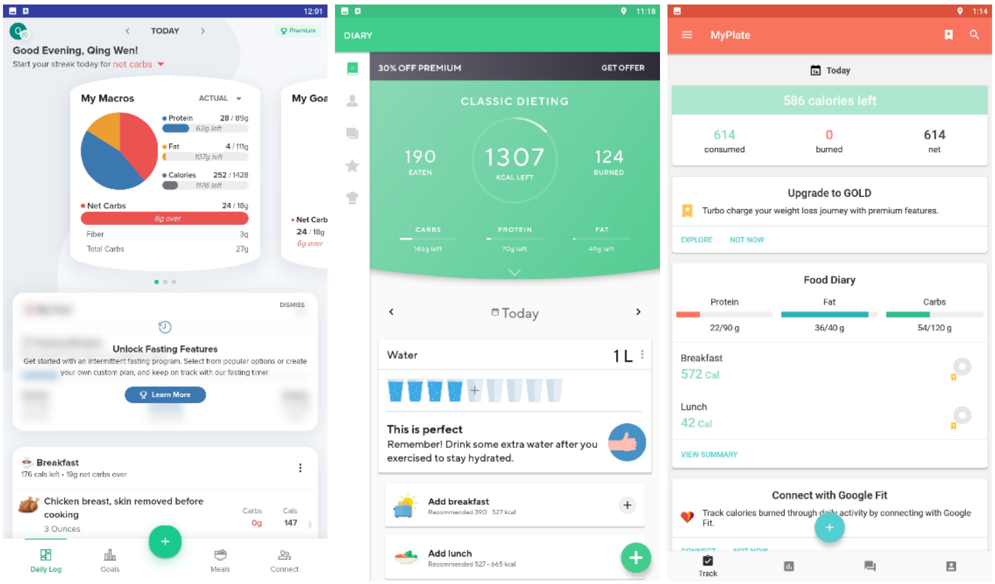
Your weekly Life Score helps you build healthy food and exercise habits.

This is perfect
Remember! Drink some extra water after you exercised to stay hydrated.

This is perfect
Remember! Drink some extra water after you exercised to stay hydrated.

Figure 2.8: Sample screens of exercise tracker.

All of these reviewed existing similar systems are having the macro calculator features except ekilu. Macro calculator is the function that will calculate the daily limit of the macronutrients and calories required by the user based on the user’s weight, height, activity level and goal. Macronutrients consist of carbohydrates, protein and fat which is important for human body growth. Macro calculator is a very useful feature in lifestyle tracker mobile application, as it helps the users to track their daily progress of their healthy lifestyle, which is to have a balanced diet and fitness activities. Figure 2.9 shows the sample screens of macro calculators from 3 different reviewed existing similar systems.



Get started with an intermittent fasting program. Select from popular options or create your own custom plan, and keep on track with our fasting timer.

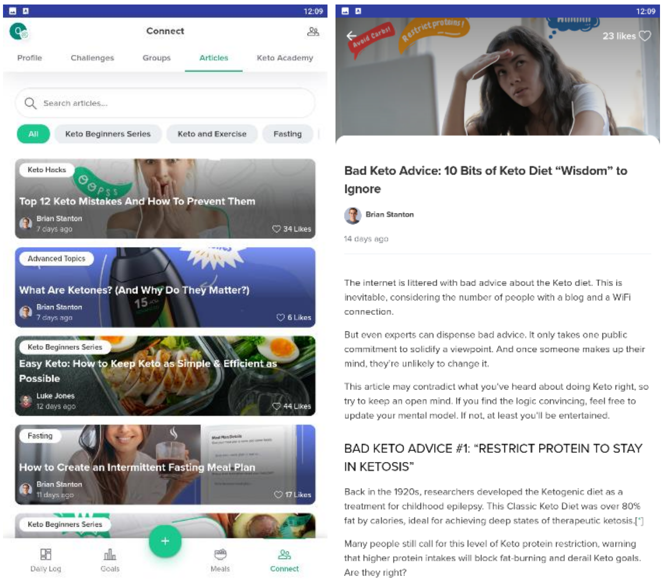
Remember! Drink some extra water after you exercised to stay hydrated.

Turbo charge your weight loss journey with premium features.

Track calories burned through daily activity by connecting with Google Fit.

Figure 2.9: Sample screens of macro calculators from 3 different reviewed existing similar systems.

There are some unique features possessed by each of the reviewed existing similar systems, such as articles related to healthy lifestyle, community and also mindfulness tracker. The articles module is found in Carb Manager and ekilu. This module provides a lot of articles related to healthy lifestyle such as nutrition, fitness, health condition, weight loss tips and a lot more. It provides users with some useful tips to tackle the problems they would face when trying to achieve a healthy lifestyle. Figure 2.10 shows the sample screens of articles module in Carb Manager.



Bad Keto Advice: 10 Bits of Keto Diet “Wisdom” to Ignore

Brian Stanton

14 days ago

The internet is littered with bad advice about the Keto diet. This is inevitable, considering the number of people with a blog and a WiFi connection.

But even experts can dispense bad advice. It only takes one public commitment to solidify a viewpoint. And once someone makes up their mind, they're unlikely to change it.

This article may contradict what you've heard about doing Keto right, so try to keep an open mind. If you find the logic convincing, feel free to update your mental model. If not, at least you'll be entertained.

BAD KETO ADVICE #1: “RESTRICT PROTEIN TO STAY IN KETOSIS”

Back in the 1920s, researchers developed the Ketogenic diet as a treatment for childhood epilepsy. This Classic Keto Diet was over 80% fat by calories, ideal for achieving deep states of therapeutic ketosis.[*]

Many people still call for this level of Keto protein restriction, warning that higher protein intakes will block fat-burning and derail Keto goals. Are they right?

Figure 2.10: Sample screens of articles module in Carb Manager.

The community module can be found in both Carb Manager and MyPlate Calorie Tracker. The community module allows the users to post anything they would like to share and have a discussion with the other users. The other users would provide comments to the post and this enables two-way communication between the users.

For the mindfulness tracker feature, it can be found in ekilu. It allows the user to record down the daily mindfulness period, so that the users can know the duration of the mindfulness per day and hence help the users move towards a balanced healthy life.

Below is the summarized version of the comparison of features among the 4 reviewed existing similar systems by 27 July 2021. The complete application review matrix has been attached as Appendix A. The common features listed were taken into consideration as the scope and implementation of this project.

Table 2.1: Comparison of features among the 4 reviewed existing similar systems.

	Lifestyle Application			
Applications / Features	Carb Manager: Keto Diet Tracker & Fasting App	Lifesum - Diet Plan, Macro Calculator & Food Diary	MyPlate Calorie Tracker	ekilu - eat well, exercise & mindfulness
Meal Tracker	✓	✓	✓	✓
Macro Calculator	✓	✓	✓ (Limited)	✗
Exercise Tracker	✓	✓	✓	✓ (Record steps)
Water Tracker	✓	✓	✓	✗
Weight Tracker	✓	✓	✗	✗

Food Recipes	✓ (Limited)	[Premium]	[Premium]	✓ (Limited)
Articles related to healthy lifestyle	✓	✗	✗	✓
Meal Planner	[Premium]	[Premium]	[Premium]	[Premium]
Community	✓	✗	✓	✗
Mindfulness Tracker	✗	✗	✗	✓

2.5.2 Discussion

It can be said that Lifesum is having the better and smoother user interface design and application flow among these 4 existing similar systems according to the review of existing similar systems, followed by Carb Manager. The navigator used by Lifesum is simple as it only uses the side navigator so that the users will not be confused when they need to go to another module. Nevertheless, Carb Manager is having the most comprehensive and sophisticated features provided to the users to better manage and track their diets and exercises. It does provide a community platform and diet groups for their users to communicate, share thoughts and experiences to one another. On the other hand, ekilu is having the most aesthetic and simplest user interface design and application flow. The color matching of the user interface is very soothing and it does provide a mindfulness tracker feature to help in improving the users' mental health. Moreover, it implements the recipe suggestions feature in which the users are required to key in what ingredients they have in their fridge and a list of recipe suggestions will be displayed. This is considered as a good feature as the users do not need to purchase extra ingredients in order to prepare the meals instructed by the recipes. Lastly, MyPlate Calorie Tracker does have a strong design of native Android application. It is simple to use, easy to navigate and record the meals and exercises. It does provide the community platform to the users, but with only simple post sharing features.

2.5.3 Conclusion

After reviewing these 4 existing similar systems, the common features of a free version lifestyle tracker mobile applications were identified, which are the meal tracker, exercise tracker, food recipes and macro calculator. The unique features in each of the existing similar systems were also reviewed. These features were referred to help in deciding and developing the functionalities of the project solution.

The common features available in all these 4 reviewed existing systems are the must-have features in the implemented system as most of the users will expect them to be available in every lifestyle tracker mobile application, and some new and modified features were identified to add uniqueness to the implemented system. Table 2.2 shows the comparison of the features between one of the existing similar system, which is Carb Manager and the implemented system in this project by 27 July 2021. Carb Manager was chosen to do the comparison with the implemented system due to the highest similarity rate with the implemented system in term of the features provided. Carb Manager acts as a benchmark for this project, and some unique features were developed in order to attract users from using the implemented system.

Table 2.2: Comparison of features between Carb Manager and the implemented system in this project.

	Lifestyle Application	
Applications Features	Carb Manager: Keto Diet Tracker & Fasting App	Implemented system in this project
Meal Tracker	✓	✓
Macro Calculator	✓	✓
Exercise Tracker	✓	✓
Food Recipes	✓ (Limited)	✓
Articles related	✓	✓

to healthy lifestyle		
Share article	✗	✓
View BMI	✗	✓
Favourite list of recipes and articles	✓ [Favourite list of recipes only]	✓
Meal check-in	✗	✓
Group discussion forum (Community)	✓	✗
Meal planner	[Premium]	✗
Track weight	✓	✗

After all, the implemented system includes the features such as:

1. Meal tracker
2. Exercise tracker
3. Macro calculator
4. Food recipes
5. Articles module to provide health care information
6. Share article to social media
7. Personal profile module to view the BMI
8. Favorite list of food recipes and articles
9. Check-in feature to capture the food photo or choose the food photo from library and attach it in the meal log

2.6 Summary

In short, this chapter has reviewed the importance of healthy lifestyle by understanding the statistics regarding people’s eating habits, activity level and mental health condition either before the COVID-19 pandemic or during the pandemic. It is shown that people are having different lifestyles and there are some people that are trying to improve their lifestyle to become healthier by changing their eating habits, doing more exercises and adjusting their attitude to face the COVID-19 situation with a positive mind.

Next, several software development methodologies have been studied and categorized into the traditional and agile methodologies. Every development methodology has its own benefits and drawbacks, and the choice of the development methodology to be adapted into a project depends on various factors such as the nature of project, the skill level of the developers, the time and budget constraints and a lot more.

Moreover, the usability testing method was identified, which was the SUS method. The SUS method provides a standard questionnaire that can be used to get the user's feedback regarding the perceived usability of the system. After getting the filled questionnaire, the SUS score was calculated using a complicated formula. The average SUS score for a good software product is 68 according to both CGS and SUS adjacent ratings.

Finally, a review of existing similar systems was used to determine the fundamental elements of the implemented system. Each of the reviewed current similar systems has certain distinctive qualities that were referred during the design phase of the implemented system.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

In this chapter, the selected development methodology was described in full, and all phases of the methodology were elaborated on the basis of this project. Not only that, the planning, design and development tools used in this project were also defined. At the end of this chapter, the work breakdown structure and Gantt chart were generated and included to show the project scope and schedule planning.

3.2 Incremental Development Methodology

The incremental model was the development methodology chosen for this project. The four primary phases of this incremental model include the planning, analysis and design, increments of design, development and testing, and lastly closing phase. When the previous phase was finished, the next phase began. After the analysis and initial design was done, the first increment started by having the design phase again, followed by the development and testing phases. The order of the features to be implemented depends on the priority level of the features. The features with the highest priority were implemented first in the first increment, and the following features were implemented in the next few increments. When each of the increments was done, the partial workable product was produced. The software system was slowly built up from the increments until reaching the final complete software system. Finally, the closing phase was conducted to do the necessary testing activities and generate a system documentation for future reference. Figure 3.1 shows the incremental model applied in this project.

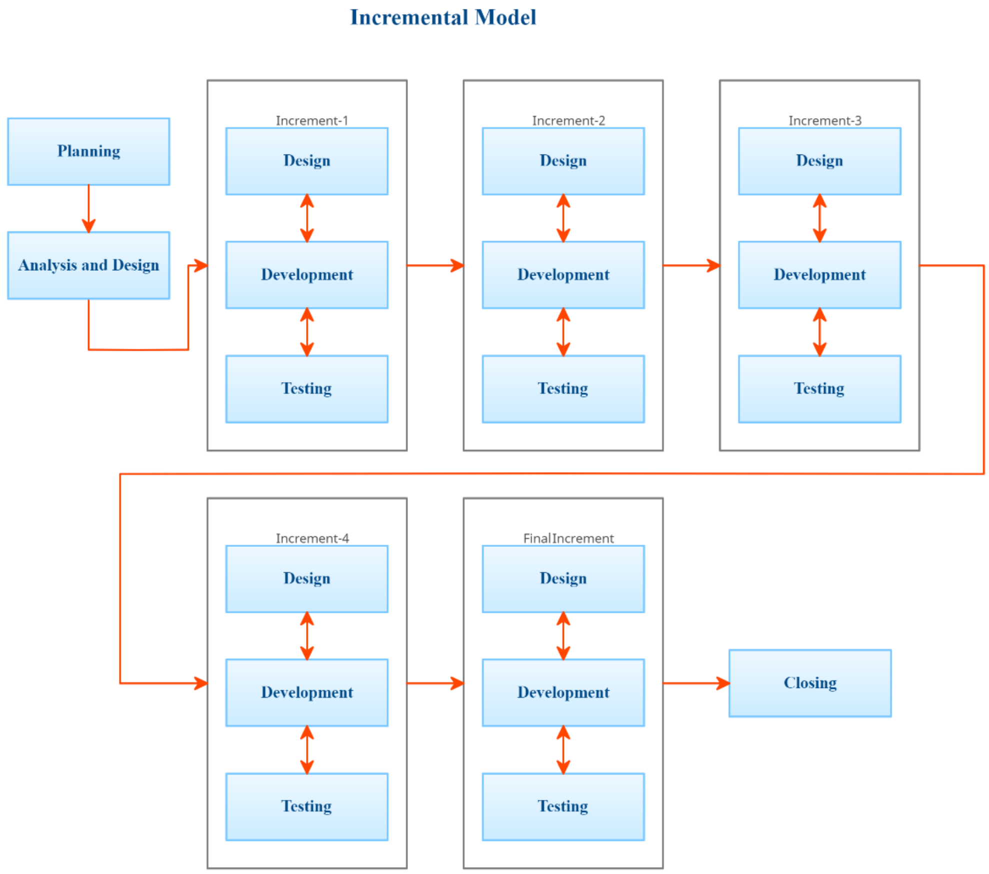


Figure 3.1: Incremental model applied in this project.

3.2.1 Planning

Every SDLC development process begins with the planning phase. A well-defined plan in the early stage of the software development processes will guide the project towards the right track. During the planning phase, the project objectives, scope and solution were identified. The development methodology was also defined in this phase. Furthermore, the requirement gathering and elicitation were conducted to generate a list of functional and non-functional requirements. At the end of this phase, the project schedule was created in order to estimate the project duration.

3.2.1.1 Project Proposal

Before establishing this project, a project proposal was compiled. First of all, the project title was studied by understanding the problems faced by the target users, which are university students and working adults in their current lifestyle. Several journal articles were studied and analyzed to find out the challenges encountered by most of the university students as well as the working adults in their daily life. The problem statements were defined after figuring out the background of the problems. The project objectives were constructed to list out what is to be achieved when the project completes. Moreover, the project solution was defined by researching the suitable system architecture. The research and development methodologies were identified to guide the project when doing the fact finding activity and managing the SDLC processes. Finally, the project scope was defined to narrow down the project scale and only to deliver what is needed by the specific target users within the time frame.

3.2.1.2 Requirement Gathering and Elicitation

After having the project proposal, the requirement gathering activities were conducted by first distributing the survey forms to collect more information from the target users such as their current lifestyle and their expectations towards a lifestyle tracker mobile application. The distribution of the survey form took about 2 weeks. After getting the sufficient amount of responses, the collected responses were analyzed and summarized to extract some useful statistics and suggestions regarding the expected features from a lifestyle tracker mobile application. The review of similar existing systems were also conducted in parallel with the analysis work on the survey responses. Several similar existing systems were chosen to review and study on their common and special features. Not only that, the literature review activity was also started to study the importance of healthy lifestyle, the development methodologies and also the usability testing methods. After finishing all the requirement gathering activities, the requirement elicitation has begun to decide on the main features of the project solution and generate a list of functional and non-functional requirements of the project solution.

3.2.1.3 Project Scheduling

The final action in the planning phase was project scheduling. A work breakdown structure was developed after all the details of the project have been determined in the preceding planning processes, and all the tasks and activities were deconstructed into smaller bits of tasks during the production of the work breakdown structure. Subsequently, a Gantt chart was generated according to the tasks in the work breakdown structure. ProjectLibre, a project management software tool was used to generate a Gantt chart that shows all the tasks timeline with a start date, the estimated duration for a task to be done and a finish date. The Gantt chart is the visualization of work breakdown structure and it helped to estimate the project duration and guided the project progress.

3.2.2 Analysis and Design

When the planning phase was done, the analysis and design phase started. The analysis and design phase did the analysis work on the objectives, scopes and the list of requirements of the implemented system. The use case diagram was produced during this phase to show how the target users will interact with the implemented system. Besides, the use case descriptions were generated to provide more details and elaboration on how the implemented system will respond to the various interactions done by the user.

After that, the interface flow diagram was produced to show the possible flows of the modules of the implemented system. The high fidelity user interface prototype design was done using Figma tool to draft out the screens layout of the implemented system. By having the initial user interface prototype design done before the development phase, it shows an overview idea of the project solution in a graphical manner. Not only that, it guided the iterations of the design, development and testing so that there was no need to spend much time on figuring out the overall layout of the module screens during each iteration.

3.2.3 Design, Development and Testing

After the analysis and design phase was done, the project continued with the iterations of design, development and testing. Since this project was adapting the incremental model, there were some iterations to be done before the final software system was produced. The user requirements were broken down into smaller pieces and the most important piece was addressed in the first iteration of the design, development and testing, followed by the other pieces of the requirements.

3.2.3.1 First Iteration

In the first iteration, the Sign Up, Login and Meal Tracker modules were implemented. Before starting the first iteration, the preliminary activities were done such as to create a GitHub repository, database and the web service program for the implemented system. A python script was written to create the PostgreSQL database and tables. Another python script was written to create a web service program which is also a Flask REST API in order to help the system to interact with the PostgreSQL database. After that, the connection between the implemented system and the web service program was set up and tested to ensure that there was no failure in the connection. Moreover, the files structure of the implemented system was built to separate out the different modules with different folders and files.

When the preliminary activities of the design, development and testing phase were completed, the Sign Up and Login modules were addressed first. In the design phase, the user interface design of Sign Up and Login modules was refined. The development phase was started by creating the user interface of the Sign Up and Login modules. The functionalities were slowly added on into the Sign Up and Login modules.

When the development process was done, the testing activity was conducted to check on the functionalities of the Sign Up and Login modules. If there were no errors on the Sign Up and Login modules, then the first iteration was proceeded to create the Meal Tracker module.

Firstly, the user interface design of the Meal Tracker module was refined according to the needs. The development phase was started by creating the user interface of the Meal Tracker module. Later, the functionalities of the Meal Tracker module were added.

When the development process was done, the testing activity was conducted to test the functionalities of the Meal Tracker module. If there were any errors found, the first iteration was returned back to the development phase or even the design phase of the Meal Tracker module to detect and fix the errors. The next iteration was started only when the testing phase of the first iteration had been completed.

3.2.3.2 Second Iteration

In the second iteration, the Exercise Tracker and Macro Calculator modules were implemented. The Exercise Tracker module was implemented first, followed by the Macro Calculator module as the Macro Calculator module depends on the Exercise Tracker as well as the Meal Tracker module that was done in the first iteration.

In the design phase, the user interface design of the Exercise Tracker module was refined. Then, the development phase was started by creating the user interface of the Exercise Tracker module. The functionalities of the Exercise Tracker such as search, add, display and edit exercise activity were implemented. After completing the development process, the testing phase was started to test on the functionalities of the Exercise Tracker to ensure the Exercise Tracker module was implemented properly.

After having the Exercise Tracker module finished, the design phase of the Macro Calculator was started by refining the user interface design. The development phase was started by creating the user interface of the Macro Calculator module. The functionalities of the Macro Calculator were developed, which include performing the calculation of the daily macronutrients required and the daily calories limit according to the data provided by the users. The testing phase was conducted to test the functionalities of the Macro Calculator to make sure that no logic error occurred when calculating the macronutrients and the calories limit. The next iteration was started only when the testing phase of this iteration had been completed.

3.2.3.3 Third Iteration

In the third iteration, the Food Recipes module was implemented. Before going into the development phase, the design phase was started by refining the user interface design of the Food Recipes module.

During the development phase, the user interface of the Food Recipes module was created. After that, the functionalities of the Food Recipes such as displaying, searching and saving the recipes were implemented.

The testing phase was conducted to ensure that the functionalities implemented are able to work properly without having errors.

3.2.3.4 Fourth Iteration

In the fourth iteration, the Articles module was implemented. During the design phase, the user interface design of the Articles module was refined. Then, the development phase was started by creating the user interface of the Articles module. The functionalities of the Articles module such as display, save and share articles were implemented.

After completing the development process, the testing phase was started to test on the functionalities of the Articles module to ensure the Articles module was implemented properly.

3.2.3.5 Final Iteration

In the final iteration, the Personal Profile module was implemented. During the design phase, the user interface design of the Personal Profile module was refined. Then, the development phase was started by creating the user interface of the Personal Profile module. The functionalities of the Personal Profile module such as display and edit the profile details and also the saved recipes and articles lists were implemented.

After completing the development process, the testing phase was started to test on the functionalities of the Personal Profile module to ensure the functionalities of the Personal Profile module were implemented correctly without errors. Not only that, because this was the final iteration of the design, development and testing, the system integration testing was conducted to test on the overall system that was built up by combining all the modules developed in different iterations. This was to make sure that the final software system could work properly with all the modules combined together.

3.2.4 Closing

After the phase of the design, development and testing iterations was completed, the SDLC processes had reached the closing phase. A good final software system produced is not only about having all the functionalities working properly, but also needs to meet the user requirements and expectations so that the software system produced can satisfy the specific target users. Hence, the SUS method in the usability testing was conducted to understand the perceived usability of the software system produced. Then, the user acceptance test was performed to understand the acceptance rate of the final software system.

After getting the results from the usability testing and user acceptance test, the system documentation was generated to document all of the activities that were done during the project and the SDLC so that it is easier for the developers and readers to refer to whenever it is needed. Lastly, the project documentation was finalized and the project was considered as completed.

3.3 Tools Used in This Project

3.3.1 ProjectLibre

ProjectLibre is an open source project management software that helps the project team to plan and estimate the project scope, budget and duration. It was used in this project to generate the work breakdown structure and Gantt chart. It was quite useful to visualize the project timeline and the relationship between the tasks that were required to be done in the whole project lifecycle.

3.3.2 Figma

Figma is a user interface design tool that is available online and can be downloaded as a desktop software. Figma was used in this project to aid in designing the user interface prototype in order to illustrate the concept of the project solution. It provides plug-ins that can help to speed up the design process such as the icon library. It could also link the user interface screen together and mock the actual mobile application interactions by providing the frames presentation feature.

3.3.3 Git and GitHub

Git is the open source version control system (VCS) that helps the developers to perform the collaboration work in developing a software system, while GitHub is the web server for Git. They were used in this project to manage the versions of the project solution. Git and GitHub allow the developer to create different branches for different features development processes. This is a great feature provided because by working on the specific branch, there is no need to worry about messing up the workable parts of the software which is saved in the master branch. This feature could help in developing the project solution as this project was adapting the incremental model which was developed the project solution part by part. For each increment or feature, a branch was created to perform the implementation of the project solution. Once the development and testing processes were done, the branch could be committed and merged with the master branch. Apart from that, if the developer accidentally messes up the source code, the revert function can be performed to revert the source code to the previous version. In short, Git and GitHub could help to manage the different increments of the project solution more easily without needing to do the manual back up on the various versions of the project solution.

3.3.4 React Native

React Native is the cross-platform UI framework that used to develop a mobile application in both Android and iOS platforms. React Native framework was adopted in this project to implement the project solution. Although this project was mainly focused on Android mobile application, but React Native was still be selected to be the framework adopted due to its usage of Javascript language as the programming language. Moreover, by using React Native framework, it could still provide the native platform capabilities and help to produce a native-like Android mobile application. The most interesting part of React Native framework is the great variety of third-party and external UI and functional libraries that could be used in this project to speed up the development process.

3.3.5 Visual Studio Code and Android Emulator

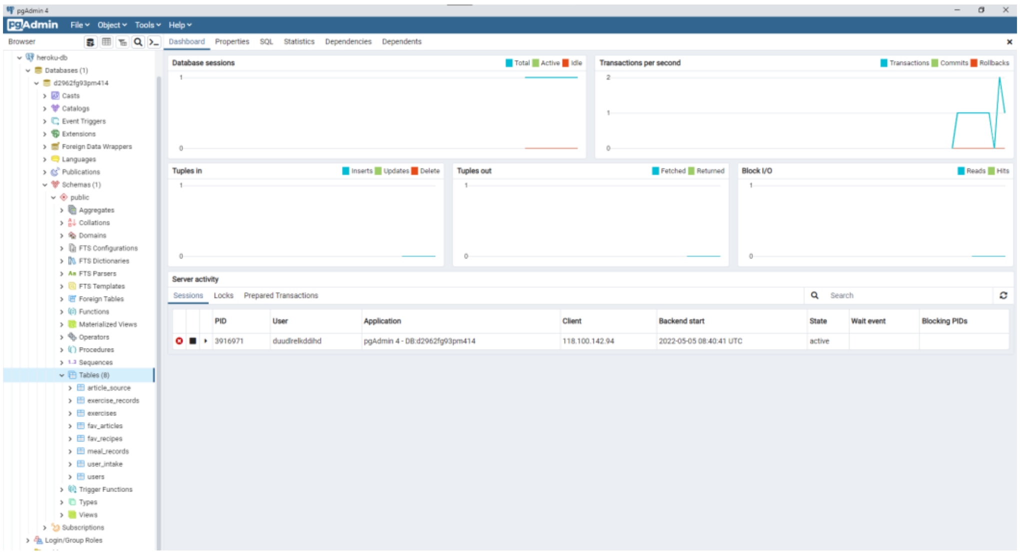
Visual Studio Code is one of the popular open source integrated development environments (IDEs) that can help the developers to work in various programming languages such as HTML, CSS, Javascript, Python and a lot more. It was used in this project as the main IDE to implement the project solution in JavaScript programming language. The proposed Android mobile application was implemented using the React Native framework, which uses the JavaScript programming language, which is simple to learn, understand, and adapt to. It provides great interface components which look similar to the native Android mobile application. Furthermore, Android Emulator was used in this project to display the output of the React Native mobile application when developing the proposed Android mobile application.

3.3.6 Notepad++

Notepad++ is the powerful text and source code editor which supports many programming languages. It allows the developer to perform editing on multiple files within a single window. It was used in this project as the IDE to implement the web service program which is Flask REST API to connect the project solution and database. It was also used to write a Python script to create the PostgreSQL database and tables.

3.3.7 pgAdmin 4

pgAdmin 4 is an open source PostgreSQL database management and development platform. It was used in this project to manage the schema and data in the PostgreSQL database tables. pgAdmin 4 provides an interactive graphical user interface to aid the process of viewing the structure of database tables. It also has the front-end terminal for PostgreSQL, which is psql, and psql can be used to perform create, select, insert, delete and update commands on the PostgreSQL's tables. By using pgAdmin 4 in this project, it made the development process more convenient as the database details could be visualized and monitored easily.



			PID	User	Application	Client	Backend start	State	Wait event	Blocking PIDs
⊗	■	▸	3916971	duudlrelkddihd	pgAdmin 4 - DB:d2962fg93pm414	118.100.142.94	2022-05-05 08:40:41 UTC	active		

Figure 3.2: Interface of pgAdmin 4.

3.3.8 Heroku

Heroku is a cloud Platform as a Service (PaaS) that runs on containers. It allows developers to deploy, manage and expand software applications with the usage of its services. Heroku is free to use unless the developers wish to upgrade their accounts to deploy the business usage application with more features provided. Heroku was used in this project to aid in deploying the Flask REST API and PostgreSQL database so that the implemented mobile application can interact with the REST API and database in the live environment.

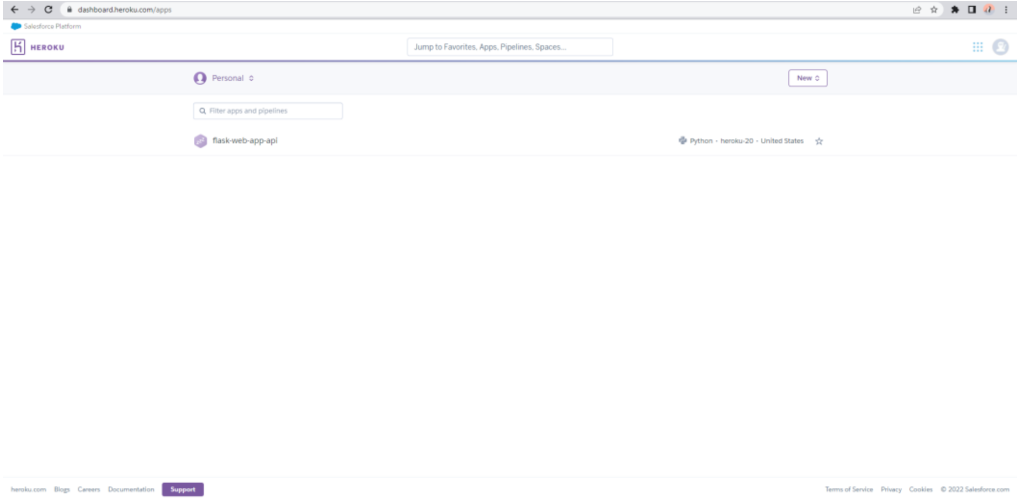


Figure 3.3: Heroku application created to host the Flask REST API.

3.4 Work Breakdown Structure (WBS)

0.0 Lifestyle Tracker Mobile Application

1.0 Planning

- 1.1 Analyze the project title
- 1.2 Study background of the problem
- 1.3 Define problem statements
- 1.4 Define project objectives
- 1.5 Define project solution
  - 1.5.1 Study the similar system architecture
  - 1.5.2 Design project's system architecture
- 1.6 Define project approach
  - 1.6.1 Define research approach
  - 1.6.2 Define development approach
- 1.7 Define project scope
  - 1.7.1 Identify target users
  - 1.7.2 Identify the covered modules
  - 1.7.3 Identify the scope exception
- 1.8 Requirement gathering
  - 1.8.1 Conduct survey
    - 1.8.1.1 Generate questions for questionnaire
    - 1.8.1.2 Distribute the questionnaire
    - 1.8.1.3 Analyze and summarize the responses
  - 1.8.2 Review similar existing systems
    - 1.8.2.1 Review Carb Manager
    - 1.8.2.2 Review Lifesum
    - 1.8.2.3 Review MyPlate Calorie Tracker
    - 1.8.2.4 Review ekilu
    - 1.8.2.5 Identify the common features
    - 1.8.2.6 Generate an application review matrix
- 1.9 Literature review
  - 1.9.1 Research on the importance of healthy lifestyle
  - 1.9.2 Research on the development methodologies
  - 1.9.3 Research on the usability testing methods
- 1.10 Requirement elicitation

1.10.1 Choose the recommended features

1.10.2 Define the functional and non-functional requirements

1.10.3 Refine the functional and non-functional requirements

1.10.4 Finalize the functional and non-functional requirements

1.11 Project scheduling

1.11.1 Create Work Breakdown Structure

1.11.1.1 Identify the main activities

1.11.1.2 Decompose the activities into smaller tasks

1.11.2 Create Gantt chart

1.11.2.1 Determine task dependency

1.11.2.2 Estmate task duration

1.11.2.3 Draft Gantt chart

1.11.2.4 Refine Gantt chart

1.11.2.5 Finalize Gantt chart

2.0 Analysis and design

2.1 Design use case diagram

2.2 Create use case description

2.3 Design interface flow diagram

2.4 Design prototype

3.0 Review feedback from Project I submission

4.0 Refine Project I documentation

5.0 Study the implementation framework and tools

6.0 Design, development and testing Iteration 1

6.1 Set up connection

6.1.1 Create repository

6.1.2 Set up database

6.1.2.1 Create a database

6.1.2.2 Create REST API

6.1.3 Connect the application to the REST API

6.2 Test connection

6.3 Create application structure

6.4 Create Sign Up and Login features

6.4.1 Design

6.4.1.1 Refine the UI of the Sign Up and Login features

6.4.2 Development

6.4.2.1 Create Sign Up UI

6.4.2.2 Create Login UI

6.4.2.3 Implement Sign Up algorithm

6.4.2.4 Implement Login algorithm

6.4.3 Testing

6.4.3.1 Test Sign Up algorithm

6.4.3.2 Test Login algorithm

6.5 Create Meal Tracker feature

6.5.1 Design

6.5.1.1 Refine the UI of the Meal Tracker feature

6.5.2 Development

6.5.2.1 Create Meal Tracker UI

6.5.2.2 Implement search meal algorithm

6.5.2.3 Implement add meal algorithm

6.5.2.4 Implement display meals algorithm

6.5.2.5 Implement edit meal algorithm

6.5.3 Testing

6.5.3.1 Test search meal algorithm

6.5.3.2 Test add meal algorithm

6.5.3.3 Test display meals algorithm

6.5.3.4 Test edit meal algorithm

7.0 Design, development and testing Iteration 2

7.1 Create Exercise Tracker feature

7.1.1 Design

7.1.1.1 Refine the UI of the Exercise Tracker feature

7.1.2 Development

7.1.2.1 Create Exercise Tracker UI

7.1.2.2 Implement search exercise algorithm

7.1.2.3 Implement add exercise algorithm

7.1.2.4 Implement display exercises algorithm

7.1.2.5 Implement edit exercise algorithm

7.1.3 Testing

7.1.3.1 Test search exercise algorithm

7.1.3.2 Test add exercise algorithm

7.1.3.3 Test display exercises algorithm

7.1.3.4 Test edit exercise algorithm

7.2 Create Macro Calculator feature

7.2.1 Design

7.2.1.1 Refine the UI of the Macro Calculator feature

7.2.2 Development

7.2.2.1 Create Macro Calculator UI

7.2.2.2 Implement Macro Calculator algorithm

7.2.3 Testing

7.2.3.1 Test Macro Calculator algorithm

8.0 Design, development and testing Iteration 3

8.1 Create Food Recipes feature

8.1.1 Design

8.1.1.1 Refine the UI of the Food Recipes feature

8.1.2 Development

8.1.2.1 Create Food Recipes UI

8.1.2.2 Implement display recipes algorithm

8.1.2.3 Implement search recipe algorithm

8.1.2.4 Implement save recipe algorithm

8.1.3 Testing

8.1.3.1 Test display recipes algorithm

8.1.3.2 Test search recipe algorithm

8.1.3.3 Test save recipe algorithm

9.0 Design, development and testing Iteration 4

9.1 Create Articles feature

9.1.1 Design

9.1.1.1 Refine the UI of Articles feature

9.1.2 Development

9.1.2.1 Create Articles UI

9.1.2.2 Implement display articles algorithm

9.1.2.3 Implement save article algorithm

- 9.1.2.4 Implement share article algorithm
- 9.1.3 Testing
  - 9.1.3.1 Test display articles algorithm
  - 9.1.3.2 Test save article algorithm
  - 9.1.3.3 Test share article algorithm

10.0 Design, development and testing Final Iteration

- 10.1 Create Personal Profile feature
  - 10.1.1 Design
    - 10.1.1.1 Refine the UI of Personal Profile feature
  - 10.1.2 Development
    - 10.1.2.1 Create Personal Profile UI
    - 10.1.2.2 Implement display profile details algorithm
    - 10.1.2.3 Implement edit profile details algorithm
    - 10.1.2.4 Implement saved recipes list
    - 10.1.2.5 Implement saved articles list
  - 10.1.3 Testing
    - 10.1.3.1 Test display profile details algorithm
    - 10.1.3.2 Test edit profile details algorithm
    - 10.1.3.3 Test saved recipes list
    - 10.1.3.4 Test saved articles list
    - 10.1.3.5 Conduct system integration testing

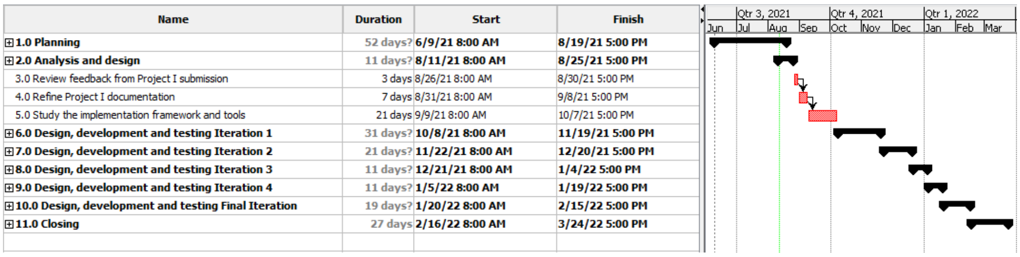
11.0 Closing

- 11.1 Conduct usability testing
- 11.2 Conduct user acceptance test
- 11.3 Create system documentation
- 11.4 Finalize the project documentation

3.5 Gantt Chart

The adoption of the incremental model in this project was the right choice. This is because the implemented system consists of several modules which are interrelated with one another and by implementing and testing module by module, the bugs could be detected and fixed easily. Nevertheless, the planned project timeline as shown in Figure 3.2 was not followed exactly, especially the iterations of design, development and testing. The design, development and testing period required for each iteration strongly depended on the complexity of the features included in the specific module. The estimated period was not sufficient for the iterations of design, development and testing for the meal tracker, exercise tracker and macro calculator modules. Moreover, an unexpected situation happened in the middle of the design, development and testing phases. The endpoints of Spoonacular Nutrition, Recipe, and Food API had once failed to respond to every request made due to the bugs from the API side. Fortunately, the API provider fixed this issue after a few days they received feedback from users. Although the planned project timeline was not followed exactly, the SDLC of the implemented system and the project documentation writing were still completed within the overall planned timeline.

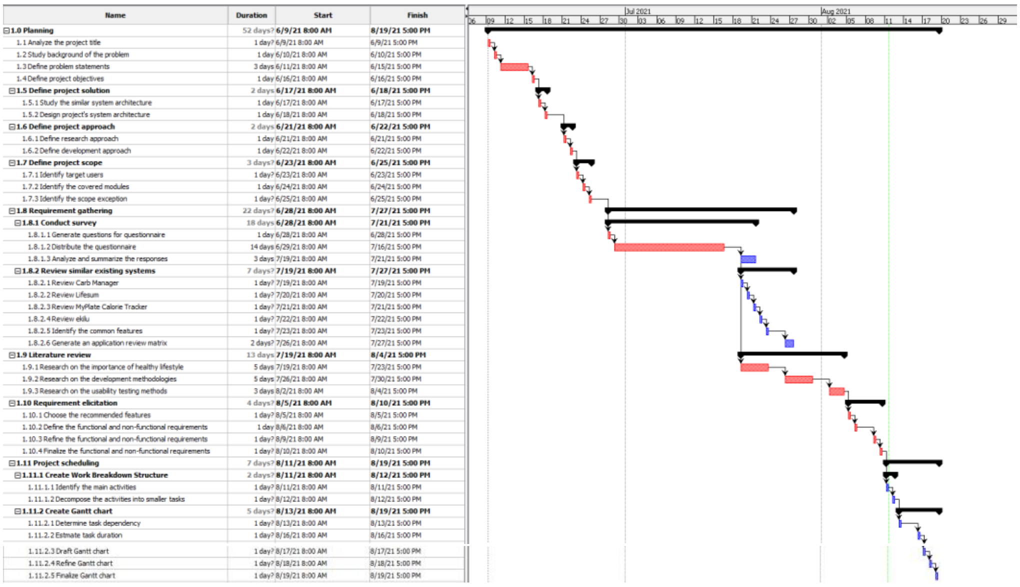
3.5.1 Overview of the Project Timeline



Name	Duration	Start	Finish
⊞ **1.0 Planning**	52 days?	**6/9/21 8:00 AM**	**8/19/21 5:00 PM**
⊞ **2.0 Analysis and design**	11 days?	**8/11/21 8:00 AM**	**8/25/21 5:00 PM**
3.0 Review feedback from Project I submission	3 days	8/26/21 8:00 AM	8/30/21 5:00 PM
4.0 Refine Project I documentation	7 days	8/31/21 8:00 AM	9/8/21 5:00 PM
5.0 Study the implementation framework and tools	21 days	9/9/21 8:00 AM	10/7/21 5:00 PM
⊞ **6.0 Design, development and testing Iteration 1**	31 days?	**10/8/21 8:00 AM**	**11/19/21 5:00 PM**
⊞ **7.0 Design, development and testing Iteration 2**	21 days?	**11/22/21 8:00 AM**	**12/20/21 5:00 PM**
⊞ **8.0 Design, development and testing Iteration 3**	11 days?	**12/21/21 8:00 AM**	**1/4/22 5:00 PM**
⊞ **9.0 Design, development and testing Iteration 4**	11 days?	**1/5/22 8:00 AM**	**1/19/22 5:00 PM**
⊞ **10.0 Design, development and testing Final Iteration**	19 days?	**1/20/22 8:00 AM**	**2/15/22 5:00 PM**
⊞ **11.0 Closing**	27 days	**2/16/22 8:00 AM**	**3/24/22 5:00 PM**

Figure 3.4: Overview of the Project Timeline.

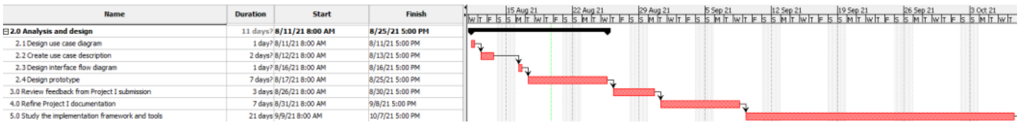
3.5.2 Planning Phase Timeline



Name	Duration	Start	Finish
⊟ **1.0 Planning**	52 days?	**6/9/21 8:00 AM**	**8/19/21 5:00 PM**
1.1 Analyze the project title	1 day?	6/9/21 8:00 AM	6/9/21 5:00 PM
1.2 Study background of the problem	1 day	6/10/21 8:00 AM	6/10/21 5:00 PM
1.3 Define problem statements	3 days	6/11/21 8:00 AM	6/15/21 5:00 PM
1.4 Define project objectives	1 day	6/16/21 8:00 AM	6/16/21 5:00 PM
⊟ **1.5 Define project solution**	2 days	**6/17/21 8:00 AM**	**6/18/21 5:00 PM**
1.5.1 Study the similar system architecture	1 day	6/17/21 8:00 AM	6/17/21 5:00 PM
1.5.2 Design project's system architecture	1 day	6/18/21 8:00 AM	6/18/21 5:00 PM
⊟ **1.6 Define project approach**	2 days	**6/21/21 8:00 AM**	**6/22/21 5:00 PM**
1.6.1 Define research approach	1 day	6/21/21 8:00 AM	6/21/21 5:00 PM
1.6.2 Define development approach	1 day	6/22/21 8:00 AM	6/22/21 5:00 PM
⊟ **1.7 Define project scope**	3 days?	**6/23/21 8:00 AM**	**6/25/21 5:00 PM**
1.7.1 Identify target users	1 day?	6/23/21 8:00 AM	6/23/21 5:00 PM
1.7.2 Identify the covered modules	1 day	6/24/21 8:00 AM	6/24/21 5:00 PM
1.7.3 Identify the scope exception	1 day?	6/25/21 8:00 AM	6/25/21 5:00 PM
⊟ **1.8 Requirement gathering**	22 days?	**6/28/21 8:00 AM**	**7/27/21 5:00 PM**
⊟ **1.8.1 Conduct survey**	18 days	**6/28/21 8:00 AM**	**7/21/21 5:00 PM**
1.8.1.1 Generate questions for questionnaire	1 day	6/28/21 8:00 AM	6/28/21 5:00 PM
1.8.1.2 Distribute the questionnaire	14 days	6/29/21 8:00 AM	7/16/21 5:00 PM
1.8.1.3 Analyze and summarize the responses	3 days	7/19/21 8:00 AM	7/21/21 5:00 PM
⊟ **1.8.2 Review similar existing systems**	7 days?	**7/19/21 8:00 AM**	**7/27/21 5:00 PM**
1.8.2.1 Review Carb Manager	1 day?	7/19/21 8:00 AM	7/19/21 5:00 PM
1.8.2.2 Review Lifesum	1 day?	7/20/21 8:00 AM	7/20/21 5:00 PM
1.8.2.3 Review MyPlate Calorie Tracker	1 day?	7/21/21 8:00 AM	7/21/21 5:00 PM
1.8.2.4 Review ekilu	1 day?	7/22/21 8:00 AM	7/22/21 5:00 PM
1.8.2.5 Identify the common features	1 day?	7/23/21 8:00 AM	7/23/21 5:00 PM
1.8.2.6 Generate an application review matrix	2 days?	7/26/21 8:00 AM	7/27/21 5:00 PM
⊟ **1.9 Literature review**	13 days	**7/19/21 8:00 AM**	**8/4/21 5:00 PM**
1.9.1 Research on the importance of healthy lifestyle	5 days	7/19/21 8:00 AM	7/23/21 5:00 PM
1.9.2 Research on the development methodologies	5 days	7/26/21 8:00 AM	7/30/21 5:00 PM
1.9.3 Research on the usability testing methods	3 days	8/2/21 8:00 AM	8/4/21 5:00 PM
⊟ **1.10 Requirement elicitation**	4 days?	**8/5/21 8:00 AM**	**8/10/21 5:00 PM**
1.10.1 Choose the recommended features	1 day?	8/5/21 8:00 AM	8/5/21 5:00 PM
1.10.2 Define the functional and non-functional requirements	1 day	8/6/21 8:00 AM	8/6/21 5:00 PM
1.10.3 Refine the functional and non-functional requirements	1 day?	8/9/21 8:00 AM	8/9/21 5:00 PM
1.10.4 Finalize the functional and non-functional requirements	1 day?	8/10/21 8:00 AM	8/10/21 5:00 PM
⊟ **1.11 Project scheduling**	7 days?	**8/11/21 8:00 AM**	**8/19/21 5:00 PM**
⊟ **1.11.1 Create Work Breakdown Structure**	2 days?	**8/11/21 8:00 AM**	**8/12/21 5:00 PM**
1.11.1.1 Identify the main activities	1 day?	8/11/21 8:00 AM	8/11/21 5:00 PM
1.11.1.2 Decompose the activities into smaller tasks	1 day?	8/12/21 8:00 AM	8/12/21 5:00 PM
⊟ **1.11.2 Create Gantt chart**	5 days?	**8/13/21 8:00 AM**	**8/19/21 5:00 PM**
1.11.2.1 Determine task dependency	1 day?	8/13/21 8:00 AM	8/13/21 5:00 PM
1.11.2.2 Estmate task duration	1 day?	8/16/21 8:00 AM	8/16/21 5:00 PM
1.11.2.3 Draft Gantt chart	1 day?	8/17/21 8:00 AM	8/17/21 5:00 PM
1.11.2.4 Refine Gantt chart	1 day?	8/18/21 8:00 AM	8/18/21 5:00 PM
1.11.2.5 Finalize Gantt chart	1 day?	8/19/21 8:00 AM	8/19/21 5:00 PM

Figure 3.5: Planning Phase Timeline.

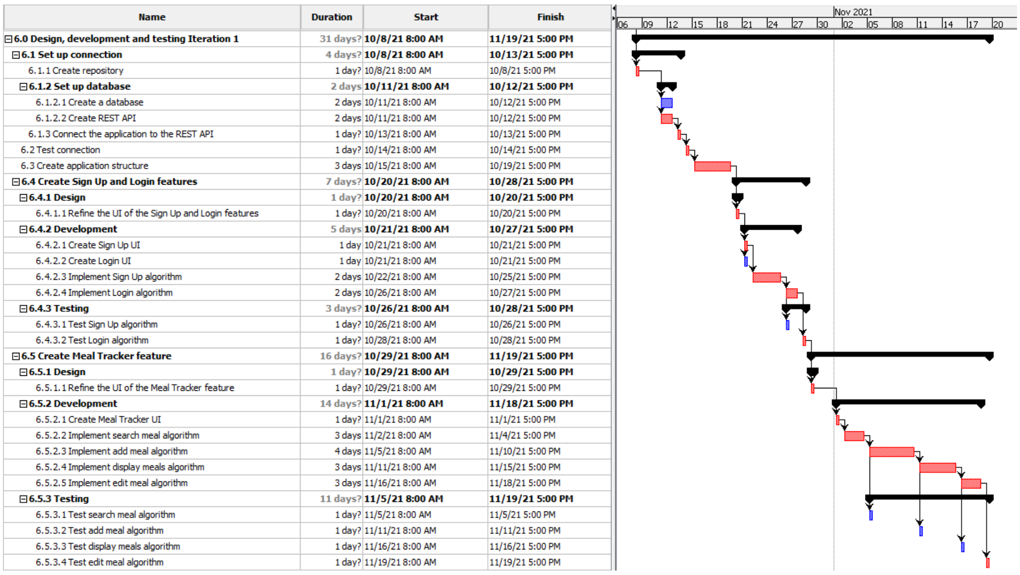
3.5.3 Analysis and Design Phase Timeline, and The Transition between Project I and II



Name	Duration	Start	Finish
⊟ **2.0 Analysis and design**	11 days?	**8/11/21 8:00 AM**	**8/25/21 5:00 PM**
2.1 Design use case diagram	1 day?	8/11/21 8:00 AM	8/11/21 5:00 PM
2.2 Create use case description	2 days?	8/12/21 8:00 AM	8/13/21 5:00 PM
2.3 Design interface flow diagram	1 day?	8/16/21 8:00 AM	8/16/21 5:00 PM
2.4 Design prototype	7 days?	8/17/21 8:00 AM	8/25/21 5:00 PM
3.0 Review feedback from Project I submission	3 days	8/26/21 8:00 AM	8/30/21 5:00 PM
4.0 Refine Project I documentation	7 days	8/31/21 8:00 AM	9/8/21 5:00 PM
5.0 Study the implementation framework and tools	21 days	9/9/21 8:00 AM	10/7/21 5:00 PM

Figure 3.6: Analysis and Design Phase Timeline including the Transition between Project I and II.

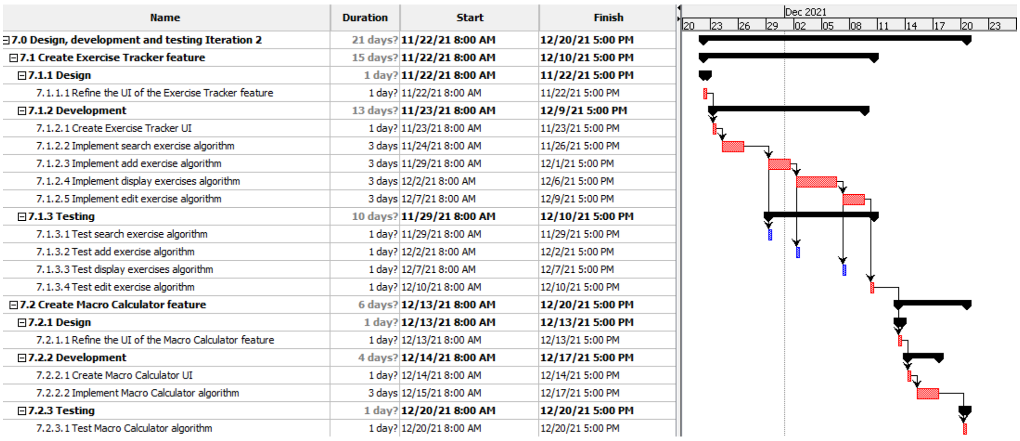
3.5.4 Design, Development and Testing Iteration 1 Timeline



Name	Duration	Start	Finish
⊟ **6.0 Design, development and testing Iteration 1**	31 days?	**10/8/21 8:00 AM**	**11/19/21 5:00 PM**
⊟ **6.1 Set up connection**	4 days?	**10/8/21 8:00 AM**	**10/13/21 5:00 PM**
6.1.1 Create repository	1 day?	10/8/21 8:00 AM	10/8/21 5:00 PM
⊟ **6.1.2 Set up database**	2 days	**10/11/21 8:00 AM**	**10/12/21 5:00 PM**
6.1.2.1 Create a database	2 days	10/11/21 8:00 AM	10/12/21 5:00 PM
6.1.2.2 Create REST API	2 days	10/11/21 8:00 AM	10/12/21 5:00 PM
6.1.3 Connect the application to the REST API	1 day?	10/13/21 8:00 AM	10/13/21 5:00 PM
6.2 Test connection	1 day?	10/14/21 8:00 AM	10/14/21 5:00 PM
6.3 Create application structure	3 days	10/15/21 8:00 AM	10/19/21 5:00 PM
⊟ **6.4 Create Sign Up and Login features**	7 days?	**10/20/21 8:00 AM**	**10/28/21 5:00 PM**
⊟ **6.4.1 Design**	1 day?	**10/20/21 8:00 AM**	**10/20/21 5:00 PM**
6.4.1.1 Refine the UI of the Sign Up and Login features	1 day?	10/20/21 8:00 AM	10/20/21 5:00 PM
⊟ **6.4.2 Development**	5 days	**10/21/21 8:00 AM**	**10/27/21 5:00 PM**
6.4.2.1 Create Sign Up UI	1 day	10/21/21 8:00 AM	10/21/21 5:00 PM
6.4.2.2 Create Login UI	1 day	10/21/21 8:00 AM	10/21/21 5:00 PM
6.4.2.3 Implement Sign Up algorithm	2 days	10/22/21 8:00 AM	10/25/21 5:00 PM
6.4.2.4 Implement Login algorithm	2 days	10/26/21 8:00 AM	10/27/21 5:00 PM
⊟ **6.4.3 Testing**	3 days?	**10/26/21 8:00 AM**	**10/28/21 5:00 PM**
6.4.3.1 Test Sign Up algorithm	1 day?	10/26/21 8:00 AM	10/26/21 5:00 PM
6.4.3.2 Test Login algorithm	1 day?	10/28/21 8:00 AM	10/28/21 5:00 PM
⊟ **6.5 Create Meal Tracker feature**	16 days?	**10/29/21 8:00 AM**	**11/19/21 5:00 PM**
⊟ **6.5.1 Design**	1 day?	**10/29/21 8:00 AM**	**10/29/21 5:00 PM**
6.5.1.1 Refine the UI of the Meal Tracker feature	1 day?	10/29/21 8:00 AM	10/29/21 5:00 PM
⊟ **6.5.2 Development**	14 days?	**11/1/21 8:00 AM**	**11/18/21 5:00 PM**
6.5.2.1 Create Meal Tracker UI	1 day?	11/1/21 8:00 AM	11/1/21 5:00 PM
6.5.2.2 Implement search meal algorithm	3 days	11/2/21 8:00 AM	11/4/21 5:00 PM
6.5.2.3 Implement add meal algorithm	4 days	11/5/21 8:00 AM	11/10/21 5:00 PM
6.5.2.4 Implement display meals algorithm	3 days	11/11/21 8:00 AM	11/15/21 5:00 PM
6.5.2.5 Implement edit meal algorithm	3 days	11/16/21 8:00 AM	11/18/21 5:00 PM
⊟ **6.5.3 Testing**	11 days?	**11/5/21 8:00 AM**	**11/19/21 5:00 PM**
6.5.3.1 Test search meal algorithm	1 day?	11/5/21 8:00 AM	11/5/21 5:00 PM
6.5.3.2 Test add meal algorithm	1 day?	11/11/21 8:00 AM	11/11/21 5:00 PM
6.5.3.3 Test display meals algorithm	1 day?	11/16/21 8:00 AM	11/16/21 5:00 PM
6.5.3.4 Test edit meal algorithm	1 day?	11/19/21 8:00 AM	11/19/21 5:00 PM

Figure 3.7: Design, Development and Testing Iteration 1 Timeline.

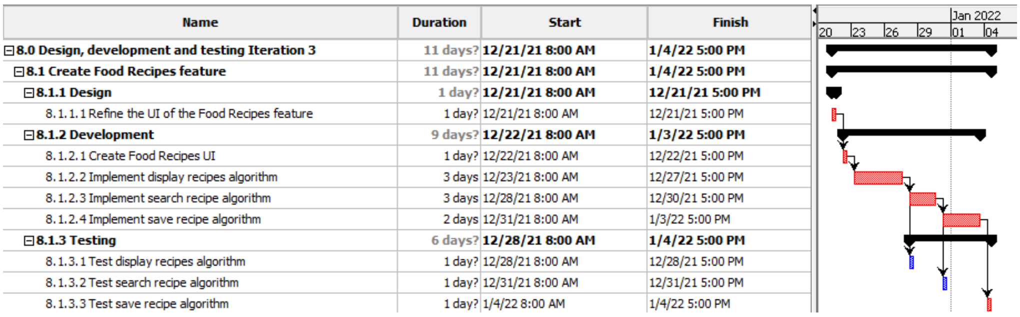
3.5.5 Design, Development and Testing Iteration 2 Timeline



Name	Duration	Start	Finish
⊟7.0 Design, development and testing Iteration 2	21 days?	11/22/21 8:00 AM	12/20/21 5:00 PM
⊟7.1 Create Exercise Tracker feature	15 days?	11/22/21 8:00 AM	12/10/21 5:00 PM
⊟7.1.1 Design	1 day?	11/22/21 8:00 AM	11/22/21 5:00 PM
7.1.1.1 Refine the UI of the Exercise Tracker feature	1 day?	11/22/21 8:00 AM	11/22/21 5:00 PM
⊟7.1.2 Development	13 days?	11/23/21 8:00 AM	12/9/21 5:00 PM
7.1.2.1 Create Exercise Tracker UI	1 day?	11/23/21 8:00 AM	11/23/21 5:00 PM
7.1.2.2 Implement search exercise algorithm	3 days	11/24/21 8:00 AM	11/26/21 5:00 PM
7.1.2.3 Implement add exercise algorithm	3 days	11/29/21 8:00 AM	12/1/21 5:00 PM
7.1.2.4 Implement display exercises algorithm	3 days	12/2/21 8:00 AM	12/6/21 5:00 PM
7.1.2.5 Implement edit exercise algorithm	3 days	12/7/21 8:00 AM	12/9/21 5:00 PM
⊟7.1.3 Testing	10 days?	11/29/21 8:00 AM	12/10/21 5:00 PM
7.1.3.1 Test search exercise algorithm	1 day?	11/29/21 8:00 AM	11/29/21 5:00 PM
7.1.3.2 Test add exercise algorithm	1 day?	12/2/21 8:00 AM	12/2/21 5:00 PM
7.1.3.3 Test display exercises algorithm	1 day?	12/7/21 8:00 AM	12/7/21 5:00 PM
7.1.3.4 Test edit exercise algorithm	1 day?	12/10/21 8:00 AM	12/10/21 5:00 PM
⊟7.2 Create Macro Calculator feature	6 days?	12/13/21 8:00 AM	12/20/21 5:00 PM
⊟7.2.1 Design	1 day?	12/13/21 8:00 AM	12/13/21 5:00 PM
7.2.1.1 Refine the UI of the Macro Calculator feature	1 day?	12/13/21 8:00 AM	12/13/21 5:00 PM
⊟7.2.2 Development	4 days?	12/14/21 8:00 AM	12/17/21 5:00 PM
7.2.2.1 Create Macro Calculator UI	1 day?	12/14/21 8:00 AM	12/14/21 5:00 PM
7.2.2.2 Implement Macro Calculator algorithm	3 days	12/15/21 8:00 AM	12/17/21 5:00 PM
⊟7.2.3 Testing	1 day?	12/20/21 8:00 AM	12/20/21 5:00 PM
7.2.3.1 Test Macro Calculator algorithm	1 day?	12/20/21 8:00 AM	12/20/21 5:00 PM

Figure 3.8: Design, Development and Testing Iteration 2 Timeline.

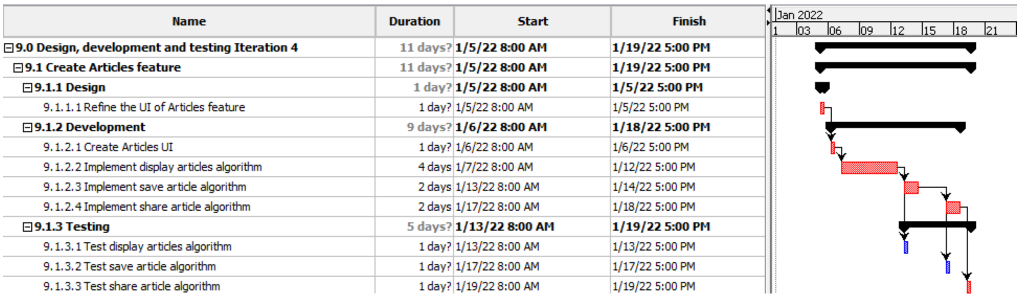
3.5.6 Design, Development and Testing Iteration 3 Timeline



Name	Duration	Start	Finish
⊟8.0 Design, development and testing Iteration 3	11 days?	12/21/21 8:00 AM	1/4/22 5:00 PM
⊟8.1 Create Food Recipes feature	11 days?	12/21/21 8:00 AM	1/4/22 5:00 PM
⊟8.1.1 Design	1 day?	12/21/21 8:00 AM	12/21/21 5:00 PM
8.1.1.1 Refine the UI of the Food Recipes feature	1 day?	12/21/21 8:00 AM	12/21/21 5:00 PM
⊟8.1.2 Development	9 days?	12/22/21 8:00 AM	1/3/22 5:00 PM
8.1.2.1 Create Food Recipes UI	1 day?	12/22/21 8:00 AM	12/22/21 5:00 PM
8.1.2.2 Implement display recipes algorithm	3 days	12/23/21 8:00 AM	12/27/21 5:00 PM
8.1.2.3 Implement search recipe algorithm	3 days	12/28/21 8:00 AM	12/30/21 5:00 PM
8.1.2.4 Implement save recipe algorithm	2 days	12/31/21 8:00 AM	1/3/22 5:00 PM
⊟8.1.3 Testing	6 days?	12/28/21 8:00 AM	1/4/22 5:00 PM
8.1.3.1 Test display recipes algorithm	1 day?	12/28/21 8:00 AM	12/28/21 5:00 PM
8.1.3.2 Test search recipe algorithm	1 day?	12/31/21 8:00 AM	12/31/21 5:00 PM
8.1.3.3 Test save recipe algorithm	1 day?	1/4/22 8:00 AM	1/4/22 5:00 PM

Figure 3.9: Design, Development and Testing Iteration 3 Timeline.

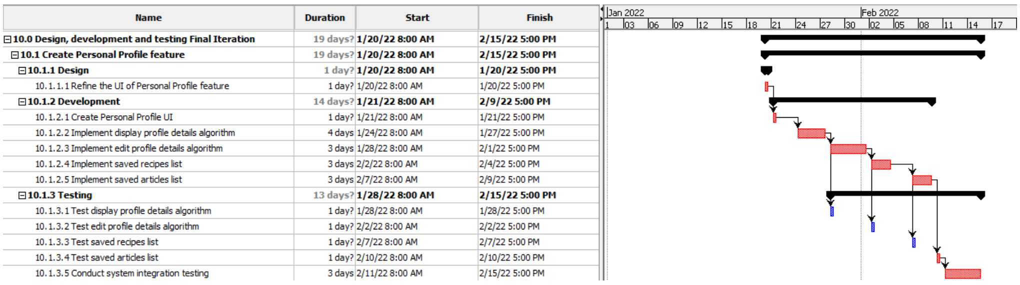
3.5.7 Design, Development and Testing Iteration 4 Timeline



Name	Duration	Start	Finish
⊟9.0 Design, development and testing Iteration 4	11 days?	1/5/22 8:00 AM	1/19/22 5:00 PM
⊟9.1 Create Articles feature	11 days?	1/5/22 8:00 AM	1/19/22 5:00 PM
⊟9.1.1 Design	1 day?	1/5/22 8:00 AM	1/5/22 5:00 PM
9.1.1.1 Refine the UI of Articles feature	1 day?	1/5/22 8:00 AM	1/5/22 5:00 PM
⊟9.1.2 Development	9 days?	1/6/22 8:00 AM	1/18/22 5:00 PM
9.1.2.1 Create Articles UI	1 day?	1/6/22 8:00 AM	1/6/22 5:00 PM
9.1.2.2 Implement display articles algorithm	4 days	1/7/22 8:00 AM	1/12/22 5:00 PM
9.1.2.3 Implement save article algorithm	2 days	1/13/22 8:00 AM	1/14/22 5:00 PM
9.1.2.4 Implement share article algorithm	2 days	1/17/22 8:00 AM	1/18/22 5:00 PM
⊟9.1.3 Testing	5 days?	1/13/22 8:00 AM	1/19/22 5:00 PM
9.1.3.1 Test display articles algorithm	1 day?	1/13/22 8:00 AM	1/13/22 5:00 PM
9.1.3.2 Test save article algorithm	1 day?	1/17/22 8:00 AM	1/17/22 5:00 PM
9.1.3.3 Test share article algorithm	1 day?	1/19/22 8:00 AM	1/19/22 5:00 PM

Figure 3.10: Design, Development and Testing Iteration 4 Timeline.

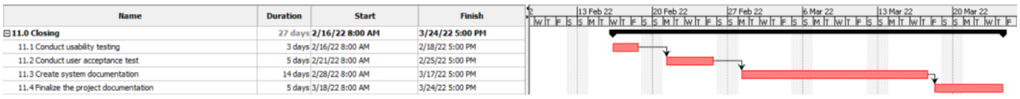
3.5.8 Design, Development and Testing Final Iteration Timeline



Name	Duration	Start	Finish
⊟ **10.0 Design, development and testing Final Iteration**	19 days?	**1/20/22 8:00 AM**	**2/15/22 5:00 PM**
⊟ **10.1 Create Personal Profile feature**	19 days?	**1/20/22 8:00 AM**	**2/15/22 5:00 PM**
⊟ **10.1.1 Design**	1 day?	**1/20/22 8:00 AM**	**1/20/22 5:00 PM**
10.1.1.1 Refine the UI of Personal Profile feature	1 day?	1/20/22 8:00 AM	1/20/22 5:00 PM
⊟ **10.1.2 Development**	14 days?	**1/21/22 8:00 AM**	**2/9/22 5:00 PM**
10.1.2.1 Create Personal Profile UI	1 day?	1/21/22 8:00 AM	1/21/22 5:00 PM
10.1.2.2 Implement display profile details algorithm	4 days	1/24/22 8:00 AM	1/27/22 5:00 PM
10.1.2.3 Implement edit profile details algorithm	3 days	1/28/22 8:00 AM	2/1/22 5:00 PM
10.1.2.4 Implement saved recipes list	3 days	2/2/22 8:00 AM	2/4/22 5:00 PM
10.1.2.5 Implement saved articles list	3 days	2/7/22 8:00 AM	2/9/22 5:00 PM
⊟ **10.1.3 Testing**	13 days?	**1/28/22 8:00 AM**	**2/15/22 5:00 PM**
10.1.3.1 Test display profile details algorithm	1 day?	1/28/22 8:00 AM	1/28/22 5:00 PM
10.1.3.2 Test edit profile details algorithm	1 day?	2/2/22 8:00 AM	2/2/22 5:00 PM
10.1.3.3 Test saved recipes list	1 day?	2/7/22 8:00 AM	2/7/22 5:00 PM
10.1.3.4 Test saved articles list	1 day?	2/10/22 8:00 AM	2/10/22 5:00 PM
10.1.3.5 Conduct system integration testing	3 days	2/11/22 8:00 AM	2/15/22 5:00 PM

Figure 3.11: Design, Development and Testing Final Iteration Timeline.

3.5.9 Closing Phase Timeline



Name	Duration	Start	Finish
⊟ **11.0 Closing**	27 days	**2/16/22 8:00 AM**	**3/24/22 5:00 PM**
11.1 Conduct usability testing	3 days	2/16/22 8:00 AM	2/18/22 5:00 PM
11.2 Conduct user acceptance test	5 days	2/21/22 8:00 AM	2/25/22 5:00 PM
11.3 Create system documentation	14 days	2/28/22 8:00 AM	3/17/22 5:00 PM
11.4 Finalize the project documentation	5 days	3/18/22 8:00 AM	3/24/22 5:00 PM

Figure 3.12: Closing Phase Timeline.

3.6 Summary

The development method chosen for this project is an incremental model, as only 1 person handles the entire project. It is important to implement the project solution part by part to ensure that the new module started developing when the previous module had been completed and tested with no error found.

There were certain deliverables required for each of the phases in the incremental model. For the planning phase, a planning proposal that consists of the problem statements, objectives, research and development methodology, project solution and scope was produced. Apart from that, there were some documents generated in the planning phase such as the project initial requirement specification and the project schedule that consists of a work breakdown structure and a Gantt chart.

When reaching the analysis and design phase, the deliverables were the use case diagram with descriptions, interface flow diagram and the high fidelity user interface prototype design. By having these deliverables, the project could proceed to the phase of design, development and testing that involves five iterations to build up a complete project solution.

During the design, development and testing phase, a part of the workable software system was produced after each iteration. After completing all the iterations in this phase, a complete software system would be the final deliverable of the design, development and testing phase.

Lastly, when reaching the closing phase of this project, a complete project documentation that includes all the research findings, deliverables from each project phase and other related information was produced.

CHAPTER 4

PROJECT SPECIFICATION

4.1 Introduction

In this chapter, the project specification was identified through the facts finding method by conducting a survey using Google Form. The questions and results of the survey are attached in Appendix B and Appendix C. The responses collected from 42 respondents, who are the target users of the implemented system had been studied and analyzed to identify the user requirements and generate a list of functional and non-functional requirements of the implemented system. Furthermore, the use case diagram and use case descriptions were created based on the functional requirements listed. Not only that, the interface flow diagram was created to show the overall ideas of the implemented system and how the implemented system reacts to the user input.

4.2 Requirements Specification

The requirement specification was divided into two categories: functional requirements and non-functional requirements. The functional requirements are a set of features that the implemented system should include in order to meet the user's needs. On the other hand, the non-functional requirements of this implemented system consists of the performance, availability, compatibility, security and usability. The implemented system has to achieve the above-stated categories of non-functional requirements in order to get the final software system with high quality and performance.

4.2.1 Functional Requirement

1. The system shall be able to allow the users to add their daily meals into their food log.
2. The system shall be able to calculate the macronutrients (carbohydrates, protein and fat) needed for the users depending on the age, weight, height, goal and the current activity level of the users and to view by the users.

3. The system shall be able to allow the users to add their exercise activities into their exercise log.
4. The system shall be able to calculate the daily net calories of the users based on their food log and exercise log and to view by the users.
5. The system shall be able to provide the users a list of recipes that can be filtered based on the eating preferences and cuisine types.
6. The system shall be able to provide the users a list of articles that related to dietary, fitness and mental health care tips.
7. The system shall be able to allow the users to save their favorite recipes and articles for future review.
8. The system shall be able to allow the users to select and add their favorite recipes as one of the daily meals in their food log.
9. The system shall be able to allow the user to update their personal information such as date of birth, weight, height, goal and the current activity level.
10. The system shall be able to allow the user to capture a food photo and upload it into their food log.
11. The system shall be able to allow the user to share the article to their social media.

4.2.2 Non-Functional Requirement

1.0 Performance

1.1 The system shall be able to respond to the user input in less than 5 seconds.

1.2 The system shall be able to effectively process all user inputs without crashing.

2.0 Availability

2.1 The system shall be available all the time without having a major downtime such as server and storage failures that will cause the system to become unavailable to the user.

3.0 Compatibility

3.1 The system shall be able to run smoothly on Android version 9.0 and above.

3.2 The system shall be able to work smoothly on Android devices of various resolutions.

4.0 Security

4.1 The system shall verify that the user has a valid account.

4.2 The system shall not allow the user to access the account without the correct matching of user credentials.

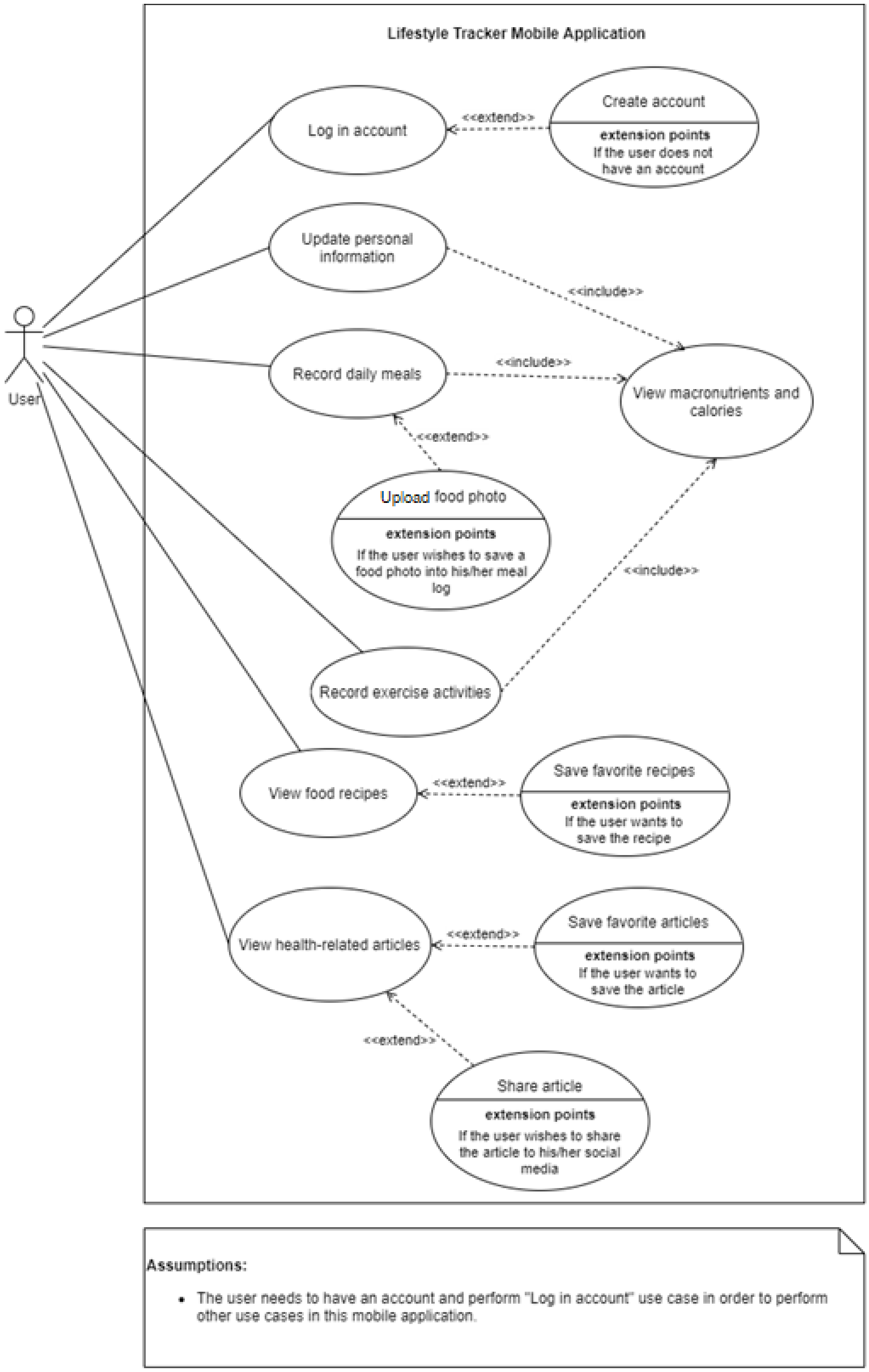
5.0 Usability

5.1 The system shall be easy to use, navigate and learn by the user without the guidance of a technical person.

5.2 The system shall be able to display the prompt message clearly so that the user understands what he should do next.

4.3 System Use Case

4.3.1 Use Case Diagram


Lifestyle Tracker Mobile Application
User
Log in account
<<extend>>
Create account
extension points
If the user does not have an account
Update personal information
<<include>>
Record daily meals
<<include>>
View macronutrients and calories
<<extend>>
Upload food photo
extension points
If the user wishes to save a food photo into his/her meal log
<<include>>
Record exercise activities
View food recipes
<<extend>>
Save favorite recipes
extension points
If the user wants to save the recipe
View health-related articles
<<extend>>
Save favorite articles
extension points
If the user wants to save the article
<<extend>>
Share article
extension points
If the user wishes to share the article to his/her social media

Assumptions:

- The user needs to have an account and perform "Log in account" use case in order to perform other use cases in this mobile application.

Figure 4.1: Use case diagram of the proposed system.

4.3.2 Use Case Description

Table 4.1: Use Case of Log In Account.

Use Case Name: Log in account	ID: 1	Importance Level: High
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to log in into his/her respective account to access the mobile application.		
Brief Description: This use case describes how the user can log in to his/her account in order to access the mobile application.		
Trigger: User wants to log in to his/her account to access the mobile application.		
Relationships: Association: User Include: - Extend: Create account Generalization: -		
Normal Flow of Events: 1. The user launches the lifestyle tracker mobile application. 2. The user enters his/her email address and password in the Sign In screen. 3. The system validates the email address and password. 4. The user is successfully logged in to the mobile application to perform the functions of the lifestyle tracker mobile application such as meal tracker, exercise tracker, view food recipes and health-related articles, and also update the personal information in the profile screen.		
Subflows: -		
Alternate/Exceptional Flows: 2a: If the user does not have an account, “Create account” use case (ID: 2) is performed. 3a: If the user enters the invalid email address or password, the system will prompt an error message and the user will need to re-enter the email address and password.		

Table 4.2: Use Case of Create Account.

Use Case Name: Create account	ID: 2	Importance Level: High
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to create an account to be used to log in and access the mobile application.		
Brief Description: This use case describes how the user can create an account to be used to log in and access the mobile application.		
Trigger: User wants to create an account to log in and access the mobile application.		
Relationships: Association: Log in account (ID: 1) Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The user clicks on the "Get Started" button in the Onboarding screen. 2. The user selects the primary goal of using this mobile application. 3. The user selects his/her gender and date of birth. 4. The user enters his/her weight and height. 5. The user selects his/her activity level. 6. The user enters his/her nickname, email address and password in the Create Account screen. 7. The system validates the email address and password. 8. The user will redirect to the Home screen, which is the Meal Tracker screen.		
Subflows: -		
Alternate/Exceptional Flows: 7a: If the user enters the invalid email address, the system will prompt an error message and the user will need to re-enter the details to create an account.		

Table 4.3: Use Case of Update Personal Information.

Use Case Name: Update personal information	ID: 3	Importance Level: Moderate
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to update his/her personal information to get the latest macronutrients needed.		
Brief Description: This use case describes how the user can update his/her personal information to get the latest macronutrients needed.		
Trigger: User wants to update his/her personal information to get the latest macronutrients needed.		
Relationships: Association: User Include: View macronutrients and calories (ID: 7) Extend: - Generalization: -		
Normal Flow of Events: 1. The user clicks on the Edit icon button beside the particular information that he/she wishes to make change in the Personal Profile screen. 2. The user enters the new information. 3. The user clicks the Save button to save the change made. 4. The user will redirect to the Personal Profile screen. 5. "View macronutrients and calories" use case (ID: 7) flow number 1 and 2 are performed.		
Subflows: -		
Alternate/Exceptional Flows: 3a: If the user does not make any changes, he/she can return to the Personal Profile screen by clicking on the Back button instead of the Save button. 5a: If the user does not wish to view the updated macronutrients, "View macronutrients and calories" use case (ID: 7) flow number 2 will not be performed.		

Table 4.4: Use Case of Record Daily Meals.

<table>
<tr><td>Use Case Name: Record daily meals</td><td>ID: 4</td><td>Importance Level: High</td></tr>
<tr><td>Primary Actor: User</td><td colspan="2">Use Case Type: Detail and Real</td></tr>
<tr><td colspan="3">Stakeholders and Interests:
User – wants to record his/her daily meals to keep track of the daily calories intake.</td></tr>
<tr><td colspan="3">Brief Description: This use case describes how the user can record his/her daily meals to keep track of the daily calories intake.</td></tr>
<tr><td colspan="3">Trigger: User wants to record his/her daily meals to keep track of the daily calories intake.</td></tr>
<tr><td colspan="3">Relationships:
Association: User
Include: View macronutrients and calories (ID: 7)
Extend: Capture food photo (ID: 6)
Generalization: -</td></tr>
<tr><td colspan="3">Normal Flow of Events:
1. The user wishes to add the meal into the meal log.
2. The user clicks on the add symbol (+) floating button and chooses the Meal option from the floating button in the Meal Tracker screen.
3. The user enters the meal name in the search bar.
4. The user selects the respective meal from the list.
5. The user views the calories and macronutrients of the meal.
6. The user clicks on the Add Food button to add the meal into his/her food log.
7. The user will redirect to the Meal Tracker screen.
8. “View macronutrients and calories” use case (ID: 7) flow number 1 and 3 are performed.</td></tr>
<tr><td colspan="3">Subflows: -</td></tr>
<tr><td colspan="3">Alternate/Exceptional Flows:
3a: If the meal cannot be found, a prompt message will display and the user will need to input another meal name to search again.</td></tr>
</table>

3b: If the user wants to choose meal from his/her list of favorite recipes, the user can click on “Favorite Recipes” option on the top bar of the screen instead of searching a meal using the search bar. 6a. If the user wishes to include a meal photo to the meal log, “Upload food photo” use case (ID: 6) is performed.

Table 4.5: Use Case of Record Exercise Activities.

Use Case Name: Record exercise activities	ID: 5	Importance Level: High
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to record his/her exercise activities to keep track of the daily calories burnt.		
Brief Description: This use case describes how the user can record his/her exercise activities to keep track of the daily calories burnt.		
Trigger: User wants to record his/her exercise activities to keep track of the daily calories burnt.		
Relationships: Association: User Include: View macronutrients and calories (ID: 7) Extend: - Generalization: -		
Normal Flow of Events: 1. The user wishes to add the exercise activity into the exercise log. 2. The user clicks on the add symbol (+) floating button and choose the Exercise option from the floating button in the Exercise Tracker screen. 3. The user enters the exercise name in the search bar. 4. The user selects the respective exercise activity in the list. 5. The user chooses the duration of exercise activity performed. 6. The user clicks on the Add Exercise button to add the exercise activity into his/her exercise log. 7. The user will redirect to the Exercise Tracker screen.		

8. “View macronutrients and calories” use case (ID: 7) flow number 1 and 4 are performed.
Subflows: -
Alternate/Exceptional Flows: 3a: If the exercise activity cannot be found, a prompt message will display and the user will need to input another exercise activity name to search again.

Table 4.6: Use Case of Upload Food Photo.

Use Case Name: Upload food photo	ID: 6	Importance Level: Moderate
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to upload the food photo and attach it to the meal log record.		
Brief Description: This use case describes how the user can upload the food photo and attach it to the meal log record.		
Trigger: User wants to upload the food photo and attach it to the meal log record.		
Relationships: Association: Record daily meals (ID: 4) Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The user wishes to upload the food photo into the meal log record. 2. The user selects the particular meal record in the meal log which he/she would like to upload the food photo. 3. The user clicks on either “take photo” or “choose from library” option in the Edit Food Log Record screen to open the device camera application or the photo gallary. 4. The user captures the food photo or choose a food photo. 5. The user clicks the Save button to save the food photo into the meal log record. 6. The user will redirect to the Meal Tracker screen.		

Subflows: -
Alternate/Exceptional Flows: 4a: If the user is not satisfied with the food photo captured or chosen, he/she can discard the photo and re-capture or re-choose again.

Table 4.7: Use Case of View Macronutrients and Calories.

Use Case Name: View macronutrients and calories	ID: 7	Importance Level: High
Primary Actor: -	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to view the updated required macronutrients, the macronutrients intake and net calories after updating the personal information, meal and tracker log.		
Brief Description: This use case describes how the user can view the updated required macronutrients, the macronutrients intake and net calories after updating the personal information, meal and tracker log.		
Trigger: User wants to view the updated required macronutrients, the macronutrients intake and net calories after updating the personal information, meal and tracker log.		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The system performs the calculation of the daily net calories and the macronutrients required by the user. 2. The user views the updated macronutrients required by him/her in the Meal Tracker screen. (ID: 3) 3. The user views the updated macronutrients intake and the daily net calories in the Meal Tracker screen. (ID: 4) 4. The user views the daily net calories in the Exercise Tracker screen. (ID: 5)		

Subflows: -
Alternate/Exceptional Flows: -

Table 4.8: Use Case of View Food Recipes.

Use Case Name: View food recipes	ID: 8	Importance Level: High
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to view food recipes to learn how to prepare the meal with the ingredients and instructions given.		
Brief Description: This use case describes how the user can view food recipes to learn how to prepare the meal with the ingredients and instructions given.		
Trigger: User wants to view food recipes to learn how to prepare the meal with the ingredients and instructions given.		
Relationships: Association: User Include: - Extend: Save favorite recipes (ID: 9) Generalization: -		
Normal Flow of Events: 1. The user navigates to the Recipes module. 2. The user views few categories of recipes in the Recipes home screen. S-1: View a list of food recipes. 3. The user clicks on a recipe to view the recipe information such as the recipe name, cooking minutes required, serving size, ingredients, instructions to prepare the meal and also the nutritional data.		
Subflows: S-1: View a list of food recipes 1. The user can view a list of food recipes by clicking on one of the recipe categories in the Recipes home screen. 2. The user can view a list of food recipes by using the search bar. - The user clicks on the Search bar		

- The user enters the food recipe name in the search bar and press Enter key to search the recipes - A list of food recipes that match with the search term will be displayed.
Alternate/Exceptional Flows: 3a: If the user wishes to save the recipe, "Save favorite recipes" use case (ID: 9) is performed.

Table 4.9: Use Case of Save Favorite Recipes.

Use Case Name: Save favorite recipes	ID: 9	Importance Level: Moderate
Primary Actor: -	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to save favorite recipes for future review.		
Brief Description: This use case describes how the user can save favorite recipes for future review.		
Trigger: User wants to save favorite recipes for future review.		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The user clicks on the bookmark icon at the top right corner of the Recipe Details screen. 2. The system will notify the user that he/she has successfully saved a recipe. 3. The user can manage his/her list of saved recipes. S-1: Manage list of saved recipes.		
Subflows: S-1: Manage list of saved recipes		

1. The user can view his/her saved recipes by clicking on the My Favorite Recipes tab in the Personal Profile screen. 2. The user can view the particular saved recipe details by clicking on that saved recipe. 3. The user can remove the saved recipe from his/her list of saved recipes by clicking on the bookmark icon in either the Favorite Recipes screen or the Recipe Details screen.
Alternate/Exceptional Flows: -

Table 4.10: Use Case of View Health-Related Articles.

Use Case Name: View health-related articles	ID: 10	Importance Level: High
Primary Actor: User	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to view health-related articles such as dietary, fitness and mental health care tips.		
Brief Description: This use case describes how the user can view health-related articles such as dietary, fitness and mental health care tips.		
Trigger: User wants to view health-related articles such as dietary, fitness and mental health care tips.		
Relationships: Association: User Include: - Extend: Save favorite articles (ID: 11), Share article (ID: 12) Generalization: -		
Normal Flow of Events: 1. The user navigates to the Articles module. 2. The user views a list of health-related articles. S-1: Filter the list of health-related articles. 3. The user clicks on an article to view the contents of the article.		
Subflows: S-1: Filter the list of health-related articles		

1. The user can filter the list of articles by clicking on the desired category tag such as dietary, fitness or mental health care tips.
Alternate/Exceptional Flows: 3a: If the user wishes to save the article, "Save favorite articles" use case (ID: 11) is performed. 3b: If the user wishes to share the article to social media, "Share article" use case (ID: 12) is performed.

Table 4.11: Use Case of Save Favorite Articles.

Use Case Name: Save favorite articles	ID: 11	Importance Level: Moderate
Primary Actor: -	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to save favorite articles for future review.		
Brief Description: This use case describes how the user can save favorite articles for future review.		
Trigger: User wants to save favorite articles for future review.		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The user clicks on the bookmark icon at the top right corner of the Article Contents screen. 2. The system will notify the user that he/she has successfully saved an article. 3. The user can manage his/her list of saved articles. S-1: Manage list of saved articles.		
Subflows: S-1: Manage list of saved articles 1. The user can view his/her saved articles by clicking on the My Favorite Articles tab in the Personal Profile screen.		

2. The user can view the particular saved article contents by clicking on that saved article. 3. The user can remove the saved article from his/her list of saved articles by clicking on the bookmark icon in either the Favorite Articles screen or the Article Contents screen.
Alternate/Exceptional Flows: -

Table 4.12: Use Case of Share Article.

Use Case Name: Share article	ID: 12	Importance Level: Moderate
Primary Actor: -	Use Case Type: Detail and Real	
Stakeholders and Interests: User – wants to share an article to social media.		
Brief Description: This use case describes how the user can share an article to social media.		
Trigger: User wants to share an article to social media.		
Relationships: Association: - Include: - Extend: - Generalization: -		
Normal Flow of Events: 1. The user clicks on the share icon at the top right corner of the Article Contents screen. 2. The user selects the social media he/she would like to share the article to. 3. The system will redirect the user to the respective social media’s post status or message screen. 4. The article summary and photo will attach in the status or message box. 5. The user clicks on the Post or Send button in the respective social media’s post status or message screen. 6. The user can view the shared content in his/her social media account.		
Subflows: -		

Alternate/Exceptional Flows: -

4.4 Interface Flow Diagram

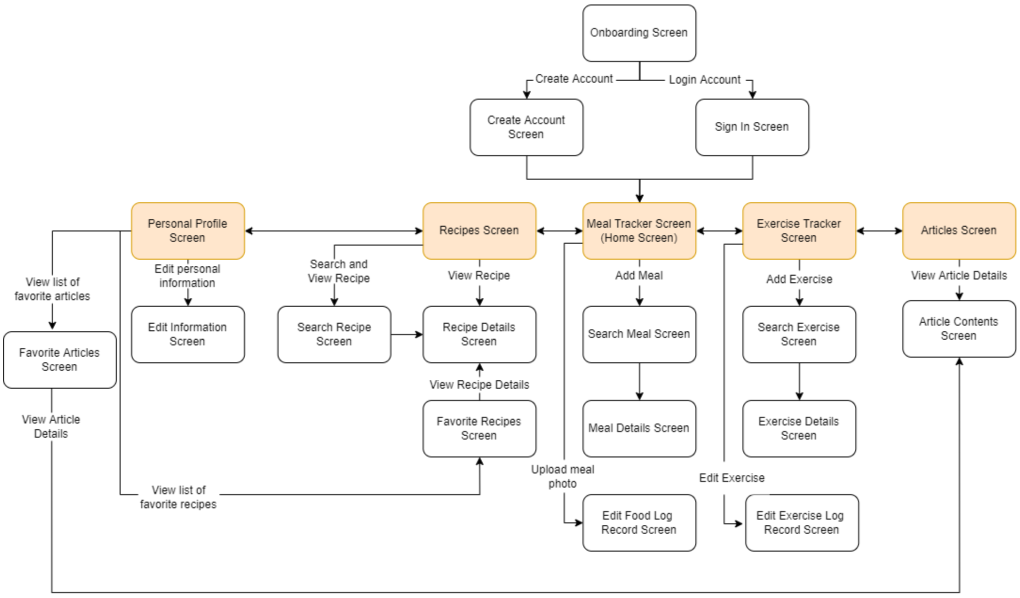


Figure 4.2: Interface flow diagram of the proposed system.

4.5 Summary

In short, this chapter displays a list of functional requirements and non-functional requirements that were generated based on the review of existing similar systems and the survey results as attached in Appendix C. A use case diagram with descriptions was produced to show the possible users' interactions with the implemented system. Lastly, an interface flow diagram was provided to display the idea of the implemented system and how the implemented system flows through the screens.

After this chapter, the design, development and testing phases were discussed to lay out the details of the implemented system in each phase of SDLC.

CHAPTER 5

DESIGN

5.1 Introduction

The system architecture design, modelling diagrams, and prototype designs of the implemented system are presented in this chapter. The system architecture design was shown to demonstrate the connection and communication between the implemented system, backend server, database, and external APIs. The modelling diagrams such as logical and physical entity relationship diagrams (ERD), data dictionary and activity diagrams were created to show the database structure and the flow of the activities and logic of the implemented system. Moreover, the high fidelity prototype design was produced to guide the implementation of system’s UI and the actual prototype design was documented to compare and contrast with the high fidelity prototype design.

5.2 System Architecture Design

Figure 5.1 depicts the implemented system's system architecture design. It shows the structure and behaviour of the implemented system.

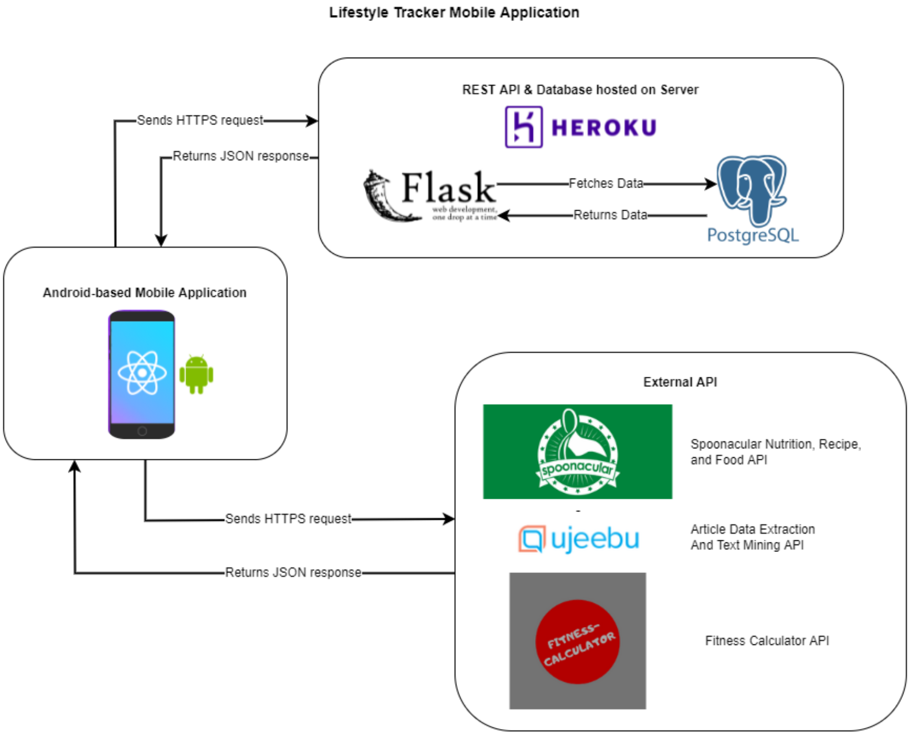

Lifestyle Tracker Mobile Application
REST API & Database hosted on Server
HEROKU
Flask
web development, one drop at a time
Fetches Data
Returns Data
PostgreSQL
Sends HTTPS request
Returns JSON response
Android-based Mobile Application
External API
spoonacular
Spoonacular Nutrition, Recipe, and Food API
ujeebu
Article Data Extraction And Text Mining API
FITNESS-CALCULATOR
Fitness Calculator API
Sends HTTPS request
Returns JSON response

Figure 5.1: System Architecture Design.

The implemented system is an Android-based mobile application developed by using React Native framework. The implemented system interacts with the backend server hosted on Heroku, which is the cloud-based hosting service platform. The backend Flask API server was developed by using Flask micro framework, and it is used by the implemented system to communicate with the PostgreSQL database indirectly. The implemented system sends the HTTPS request to the backend Flask API server and the backend Flask API server will help to retrieve and receive the data from the PostgreSQL database. After receiving data, the backend Flask API server will return the JSON response to the implemented system, which is the retrieved data from database in JSON format.

Not only that, the implemented system has integrated with the external APIs such as Spoonacular Nutrition, Recipe, and Food API, Article Data Extraction and Text Mining API and Fitness Calculator API to retrieve the necessary information for the specific features and functionalities provided by the implemented system. Spoonacular Nutrition, Recipe, and Food API is utilized by the implemented system to retrieve the meals and recipes information. Besides, Article Data Extraction and Text Mining API is utilized by the implemented system to retrieve the article contents. Fitness Calculator API is utilized to retrieve the daily net calories and macronutrients limit of the user when the user performs sign up or update his/her personal information.

5.3 Modelling Diagram

The modelling diagrams generated include the data models, data dictionary and the activity diagrams. The logical ERD was the first data model generated, indicating how the functional requirements aided in determining the relationship between entities. Later on, the logical ERD has converted to become physical ERD to determine the actual database structure, relationship between entities, primary and foreign keys and the attribute types and sizes. The data dictionary was created to document the collection of data saved in the tables of the database. Not only that, the activity diagrams were created to show the flow of the possible activities and use cases in the implemented system.

5.3.1 Data Model

Figure 5.2 illustrates the logical ERD of the implemented system, while Figure 5.3 illustrates the physical ERD of the implemented system. Both diagrams show how the functional requirements have transformed to the schema level design of the database that was used to create the database tables during the implementation phase of SDLC.

5.3.1.1 Logical ERD

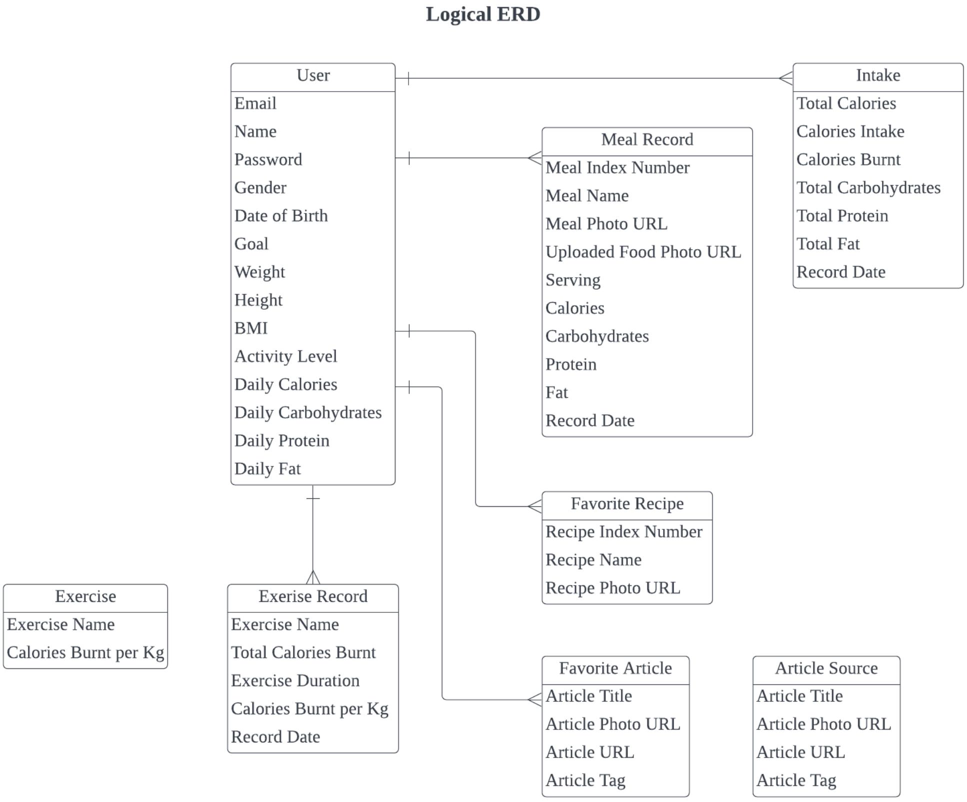
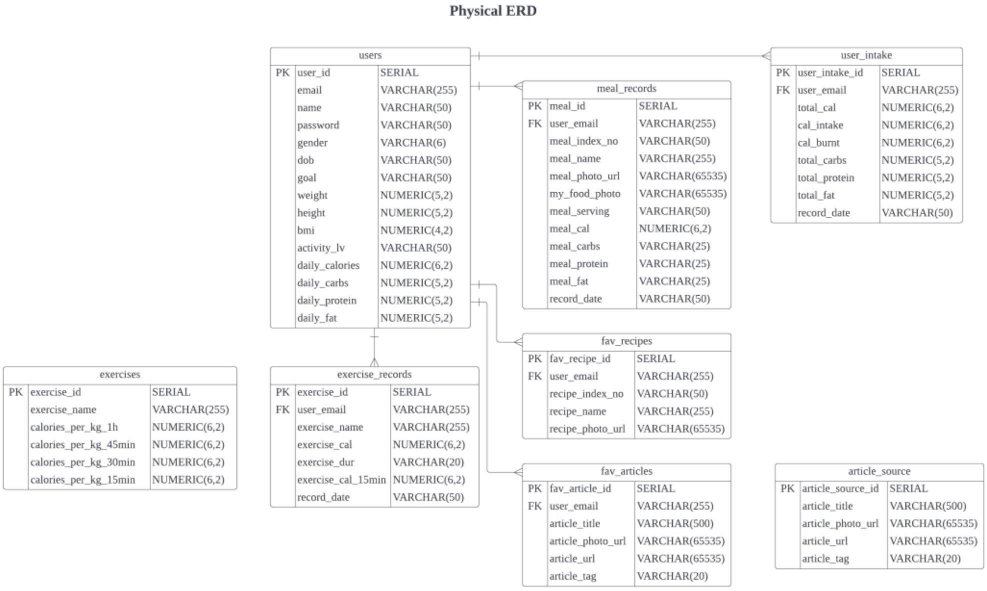


Figure 5.2: Logical ERD.

5.3.1.2 Physical ERD



users

PK	user_id	SERIAL
	email	VARCHAR(255)
	name	VARCHAR(50)
	password	VARCHAR(50)
	gender	VARCHAR(6)
	dob	VARCHAR(50)
	goal	VARCHAR(50)
	weight	NUMERIC(5,2)
	height	NUMERIC(5,2)
	bmi	NUMERIC(4,2)
	activity_lv	VARCHAR(50)
	daily_calories	NUMERIC(6,2)
	daily_carbs	NUMERIC(5,2)
	daily_protein	NUMERIC(5,2)
	daily_fat	NUMERIC(5,2)

meal_records

PK	meal_id	SERIAL
FK	user_email	VARCHAR(255)
	meal_index_no	VARCHAR(50)
	meal_name	VARCHAR(255)
	meal_photo_url	VARCHAR(65535)
	my_food_photo	VARCHAR(65535)
	meal_serving	VARCHAR(50)
	meal_cal	NUMERIC(6,2)
	meal_carbs	VARCHAR(25)
	meal_protein	VARCHAR(25)
	meal_fat	VARCHAR(25)
	record_date	VARCHAR(50)

user_intake

PK	user_intake_id	SERIAL
FK	user_email	VARCHAR(255)
	total_cal	NUMERIC(6,2)
	cal_intake	NUMERIC(6,2)
	cal_burnt	NUMERIC(6,2)
	total_carbs	NUMERIC(5,2)
	total_protein	NUMERIC(5,2)
	total_fat	NUMERIC(5,2)
	record_date	VARCHAR(50)

exercises

PK	exercise_id	SERIAL
	exercise_name	VARCHAR(255)
	calories_per_kg_1h	NUMERIC(6,2)
	calories_per_kg_45min	NUMERIC(6,2)
	calories_per_kg_30min	NUMERIC(6,2)
	calories_per_kg_15min	NUMERIC(6,2)

exercise_records

PK	exercise_id	SERIAL
FK	user_email	VARCHAR(255)
	exercise_name	VARCHAR(255)
	exercise_cal	NUMERIC(6,2)
	exercise_dur	VARCHAR(20)
	exercise_cal_15min	NUMERIC(6,2)
	record_date	VARCHAR(50)

fav_recipes

PK	fav_recipe_id	SERIAL
FK	user_email	VARCHAR(255)
	recipe_index_no	VARCHAR(50)
	recipe_name	VARCHAR(255)
	recipe_photo_url	VARCHAR(65535)

fav_articles

PK	fav_article_id	SERIAL
FK	user_email	VARCHAR(255)
	article_title	VARCHAR(500)
	article_photo_url	VARCHAR(65535)
	article_url	VARCHAR(65535)
	article_tag	VARCHAR(20)

article_source

PK	article_source_id	SERIAL
	article_title	VARCHAR(500)
	article_photo_url	VARCHAR(65535)
	article_url	VARCHAR(65535)
	article_tag	VARCHAR(20)

Figure 5.3: Physical ERD.

5.3.2 Data Dictionary

Table Name: users

Table 5.1: Data dictionary for users table.

Column Name	Description	Data Type	Size	Primary Key?	Foreign Key?	FK reference table
user_id	Unique identification of user	SERIAL	-	Yes	-	-
email	User’s email address	VARCHAR	255	-	-	-
name	User’s name	VARCHAR	50	-	-	-
password	User’s password	VARCHAR	50	-	-	-
gender	User’s gender	VARCHAR	6	-	-	-
dob	User’s date of birth	VARCHAR	50	-	-	-
goal	User’s goal of using this mobile application	VARCHAR	50	-	-	-
weight	User’s weight	NUMERIC	5,2	-	-	-
height	User’s hight	NUMERIC	5,2	-	-	-
bmi	User’s BMI value	NUMERIC	4,2	-	-	-
activity_lv	User’s current activity level	VARCHAR	50	-	-	-
daily_calories	User’s daily calories limit	NUMERIC	6,2	-	-	-
daily_carbs	User’s daily carbohydrates limit	NUMERIC	5,2	-	-	-

daily_protein	User’s daily protein limit	NUMERIC	5,2	-	-	-
daily_fat	User’s daily fat limit	NUMERIC	5,2	-	-	-

Table Name: user_intake

Table 5.2: Data dictionary for user_intake table.

Column Name	Description	Data Type	Size	Primary Key?	Foreign Key?	FK reference table
user_intake_id	Unique identification of user’s intake of the day	SERIAL	-	Yes	-	-
user_email	User’s email address	VARCHAR	255	-	Yes	users
total_cal	User’s net calories of the day	NUMERIC	6,2	-	-	-
cal_intake	User’s calories intake of the day	NUMERIC	6,2	-	-	-
cal_burnt	User’s calories burnt of the day	NUMERIC	6,2	-	-	-
total_carbs	User’s total carbohydrates intake of the day	NUMERIC	5,2	-	-	-
total_protein	User’s total protein intake of the day	NUMERIC	5,2	-	-	-
total_fat	User’s total fat intake of the day	NUMERIC	5,2	-	-	-
record_date	Record date of the user’s intake	VARCHAR	50	-	-	-

Table Name: meal_records

Table 5.3: Data dictionary for meal_records table.

Column Name	Description	Data Type	Size	Primary Key?	Foreign Key?	FK reference table
meal_id	Unique identification of meal record	SERIAL	-	Yes	-	-
user_email	User’s email address	VARCHAR	255	-	Yes	users
meal_index_no	Meal’s index number	VARCHAR	50	-	-	-
meal_name	Meal’s name	VARCHAR	255	-	-	-
meal_photo_url	URL of the meal photo	VARCHAR	65535	-	-	-
my_food_photo	URL of the uploaded food photo	VARCHAR	65535	-	-	-
meal_serving	Meal’s serving size	VARCHAR	50	-	-	-
meal_cal	Meal’s calories	NUMERIC	6,2	-	-	-
meal_carbs	Meal’s carbohydrates	VARCHAR	25	-	-	-
meal_protein	Meal’s protein	VARCHAR	25	-	-	-
meal_fat	Meal’s fat	VARCHAR	25	-	-	-
record_date	Record date of the meal record	VARCHAR	50	-	-	-

Table Name: exercises

Table 5.4: Data dictionary for exercises table.

Column Name	**Description**	**Data Type**	**Size**	**Primary Key?**	**Foreign Key?**	**FK reference table**
exercise_id	Unique identification of exercise	SERIAL	-	Yes	-	-
exercise_name	Exercise's name	VARCHAR	255	-	-	-
calories_per_kg_1h	Calories burnt per kg in one hour	NUMERIC	6,2	-	-	-
calories_per_kg_45min	Calories burnt per kg in 45 minutes	NUMERIC	6,2	-	-	-
calories_per_kg_30min	Calories burnt per kg in 30 minutes	NUMERIC	6,2	-	-	-
calories_per_kg_15min	Calories burnt per kg in 15 minutes	NUMERIC	6,2	-	-	-

Table Name: exercise_records

Table 5.5: Data dictionary for exercise_records table.

Column Name	Description	Data Type	Size	Primary Key?	Foreign Key?	FK reference table
exercise_id	Unique identification of exercise record	SERIAL	-	Yes	-	-
user_email	User’s email address	VARCHAR	255	-	Yes	users
exercise_name	Exercise’s name	VARCHAR	255	-	-	-
exercise_cal	Total calories burnt of the exercise	NUMERIC	6,2	-	-	-
exercise_dur	Duration of exercise performed	VARCHAR	20	-	-	-
exercise_cal_15min	Calories burnt per kg in 15 minutes	NUMERIC	6,2	-	-	-
record_date	Record date of the exercise record	VARCHAR	50	-	-	-

Table Name: fav_recipes

Table 5.6: Data dictionary for fav_recipes table.

Column Name	**Description**	**Data Type**	**Size**	**Primary Key?**	**Foreign Key?**	**FK reference table**
fav_recipe_id	Unique identification of the favorite recipe	SERIAL	-	Yes	-	-
user_email	User’s email address	VARCHAR	255	-	Yes	users
recipe_index_no	Recipe’s index number	VARCHAR	50	-	-	-
recipe_name	Recipe’s name or title	VARCHAR	255	-	-	-
recipe_photo_url	URL of the recipe photo	VARCHAR	65535	-	-	-

Table Name: article_source

Table 5.7: Data dictionary for article_source table.

Column Name	**Description**	**Data Type**	**Size**	**Primary Key?**	**Foreign Key?**	**FK reference table**
article_source_id	Unique identification of the article source	SERIAL	-	Yes	-	-
article_title	Article’s title	VARCHAR	500	-	-	-
article_photo_url	URL of the article photo	VARCHAR	65535	-	-	-
article_url	URL of the article	VARCHAR	65535	-	-	-
article_tag	Article’s tag or category	VARCHAR	20	-	-	-

Table Name: fav_articles

Table 5.8: Data dictionary for fav_articles table.

Column Name	**Description**	**Data Type**	**Size**	**Primary Key?**	**Foreign Key?**	**FK reference table**
fav_article_id	Unique identification of the favorite article	SERIAL	-	Yes	-	-
user_email	User’s email address	VARCHAR	255	-	Yes	users
article_title	Article’s title	VARCHAR	500	-	-	-
article_photo_url	URL of the article photo	VARCHAR	65535	-	-	-
article_url	URL of the article	VARCHAR	65535	-	-	-
article_tag	Article’s tag or category	VARCHAR	20	-	-	-

5.3.3 Activity Diagram

The activity diagrams were generated to describe the workflows of possible activities of the implemented system. The graphical activity diagrams allow the reader to better understand the overall flow of application logic and how the user and system interact with each other during the process.

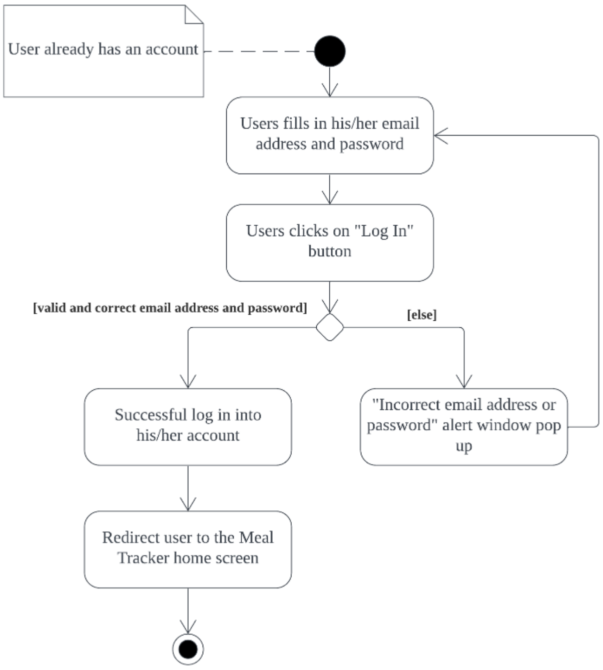

User already has an account
Users fills in his/her email address and password
Users clicks on "Log In" button
[valid and correct email address and password]
[else]
Successful log in into his/her account
"Incorrect email address or password" alert window pop up
Redirect user to the Meal Tracker home screen

Figure 5.4: Activity Diagram for Login module.

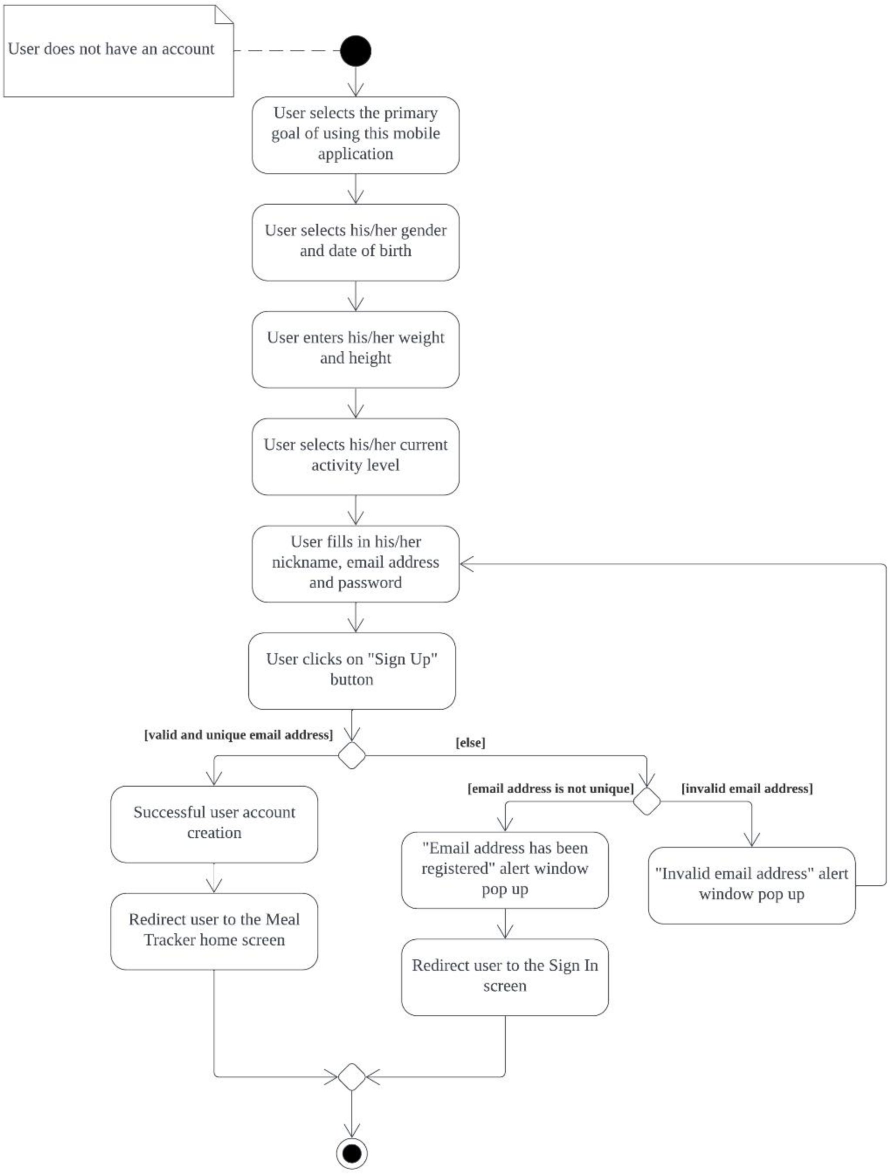

User does not have an account
User selects the primary goal of using this mobile application
User selects his/her gender and date of birth
User enters his/her weight and height
User selects his/her current activity level
User fills in his/her nickname, email address and password
User clicks on "Sign Up" button
[valid and unique email address]
[else]
[email address is not unique]
[invalid email address]
Successful user account creation
"Email address has been registered" alert window pop up
"Invalid email address" alert window pop up
Redirect user to the Meal Tracker home screen
Redirect user to the Sign In screen

Figure 5.5: Activity Diagram for Sign Up module.

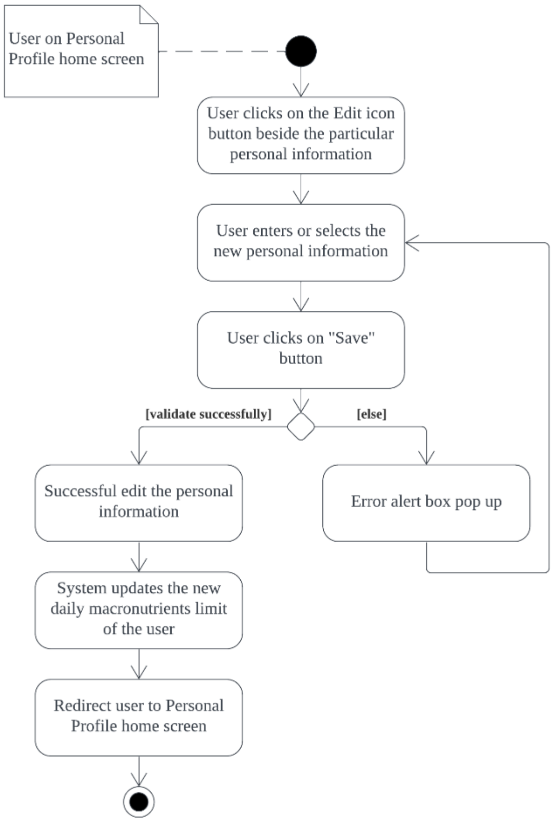

User on Personal Profile home screen
User clicks on the Edit icon button beside the particular personal information
User enters or selects the new personal information
User clicks on "Save" button
[validate successfully]
[else]
Successful edit the personal information
Error alert box pop up
System updates the new daily macronutrients limit of the user
Redirect user to Personal Profile home screen

Figure 5.6: Activity Diagram for Update Personal Information process.

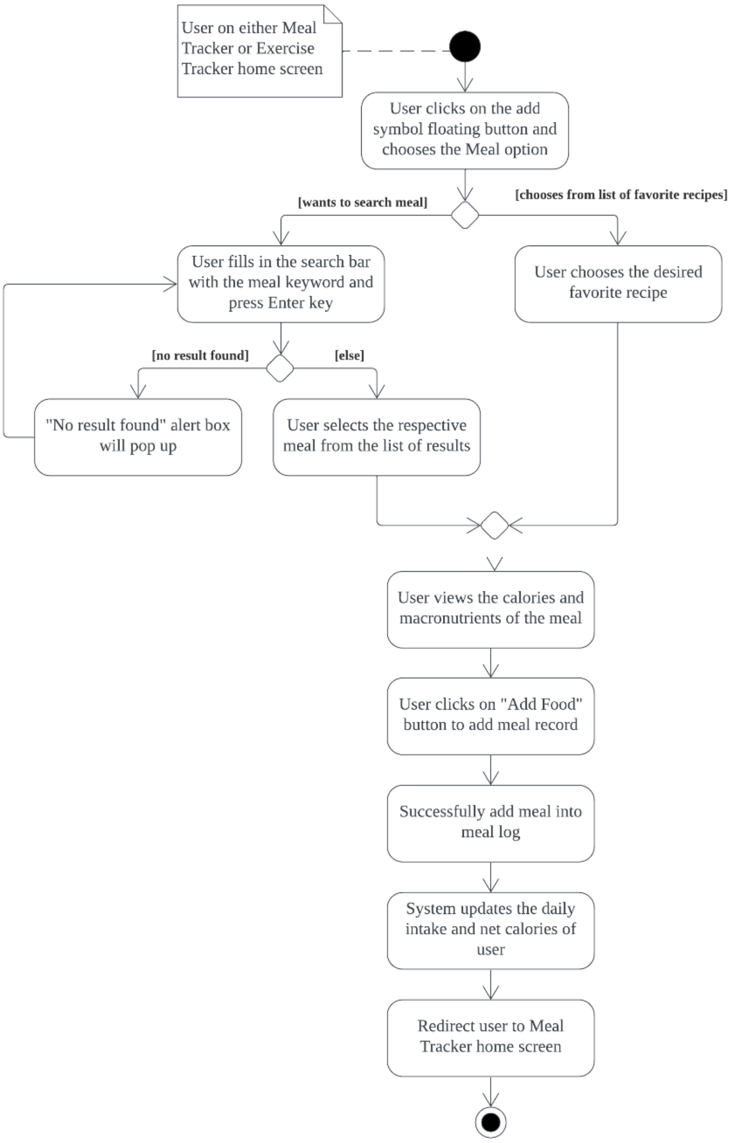

User on either Meal Tracker or Exercise Tracker home screen
User clicks on the add symbol floating button and chooses the Meal option
[wants to search meal]
[chooses from list of favorite recipes]
User fills in the search bar with the meal keyword and press Enter key
User chooses the desired favorite recipe
[no result found]
[else]
"No result found" alert box will pop up
User selects the respective meal from the list of results
User views the calories and macronutrients of the meal
User clicks on "Add Food" button to add meal record
Successfully add meal into meal log
System updates the daily intake and net calories of user
Redirect user to Meal Tracker home screen

Figure 5.7: Activity Diagram for Record Daily Meals process.

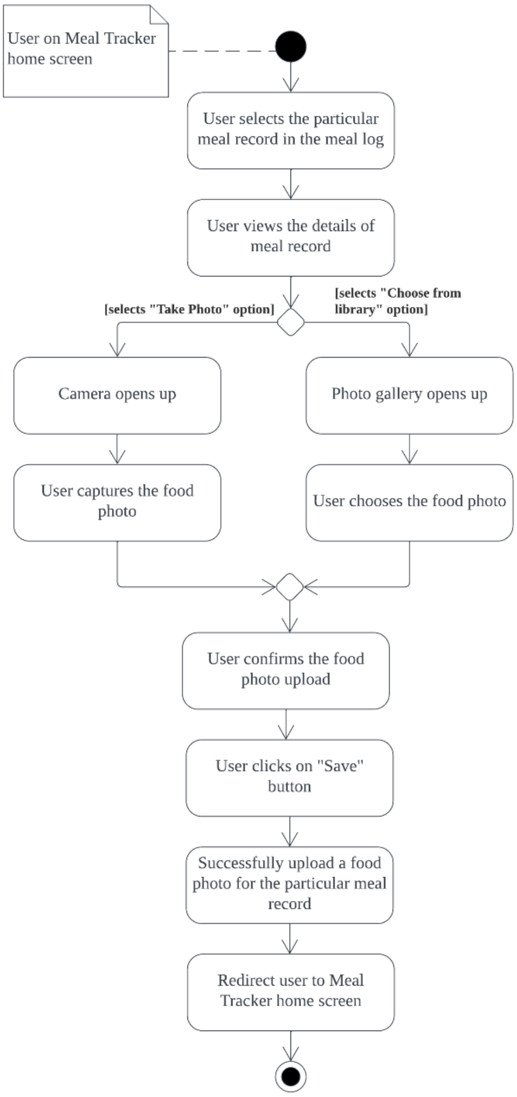

User on Meal Tracker home screen
User selects the particular meal record in the meal log
User views the details of meal record
[selects "Take Photo" option]
[selects "Choose from library" option]
Camera opens up
Photo gallery opens up
User captures the food photo
User chooses the food photo
User confirms the food photo upload
User clicks on "Save" button
Successfully upload a food photo for the particular meal record
Redirect user to Meal Tracker home screen

Figure 5.8: Activity Diagram for Upload Food Photo process.

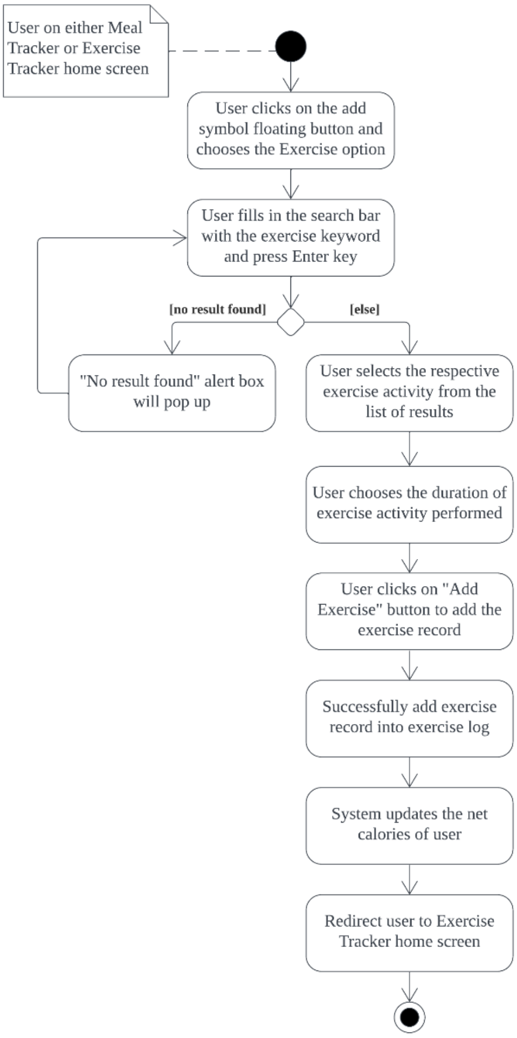

User on either Meal Tracker or Exercise Tracker home screen
User clicks on the add symbol floating button and chooses the Exercise option
User fills in the search bar with the exercise keyword and press Enter key
[no result found]
[else]
"No result found" alert box will pop up
User selects the respective exercise activity from the list of results
User chooses the duration of exercise activity performed
User clicks on "Add Exercise" button to add the exercise record
Successfully add exercise record into exercise log
System updates the net calories of user
Redirect user to Exercise Tracker home screen

Figure 5.9: Activity Diagram for Record Exercise Activities process.

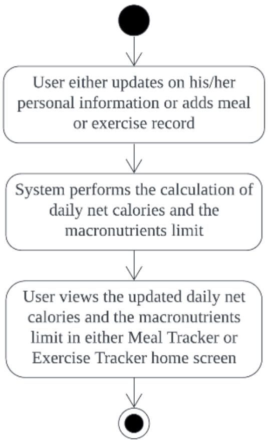

User either updates on his/her personal information or adds meal or exercise record
System performs the calculation of daily net calories and the macronutrients limit
User views the updated daily net calories and the macronutrients limit in either Meal Tracker or Exercise Tracker home screen

Figure 5.10: Activity Diagram for View Macronutrients and Calories process.

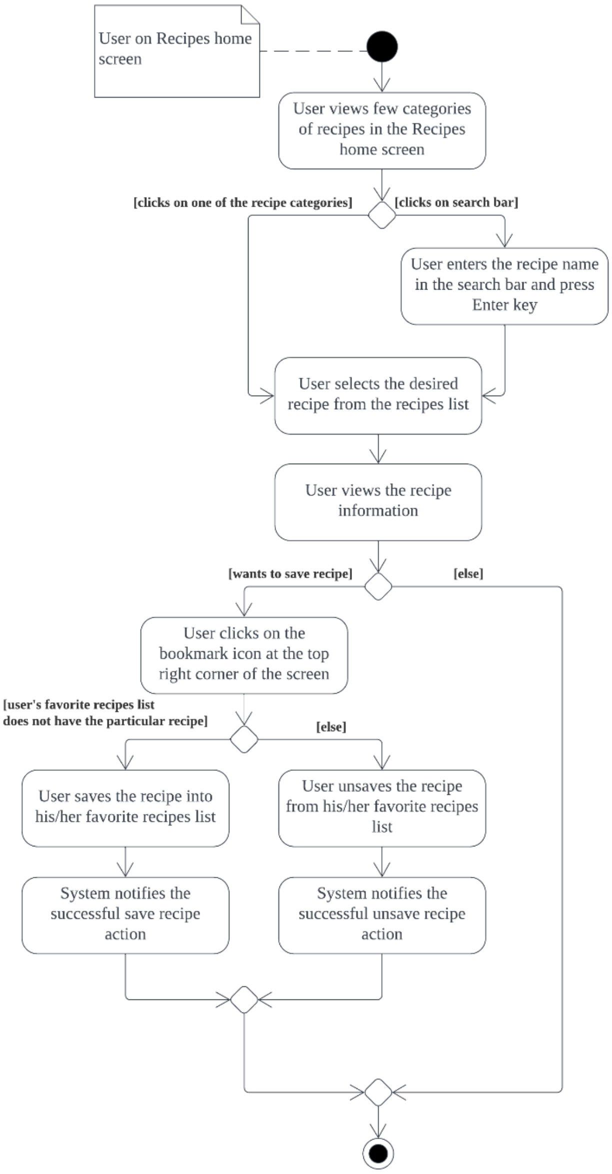

User on Recipes home screen
User views few categories of recipes in the Recipes home screen
[clicks on one of the recipe categories]
[clicks on search bar]
User enters the recipe name in the search bar and press Enter key
User selects the desired recipe from the recipes list
User views the recipe information
[wants to save recipe]
[else]
User clicks on the bookmark icon at the top right corner of the screen
[user's favorite recipes list does not have the particular recipe]
[else]
User saves the recipe into his/her favorite recipes list
User unsaves the recipe from his/her favorite recipes list
System notifies the successful save recipe action
System notifies the successful unsave recipe action

Figure 5.11: Activity Diagram for View Food Recipes process.

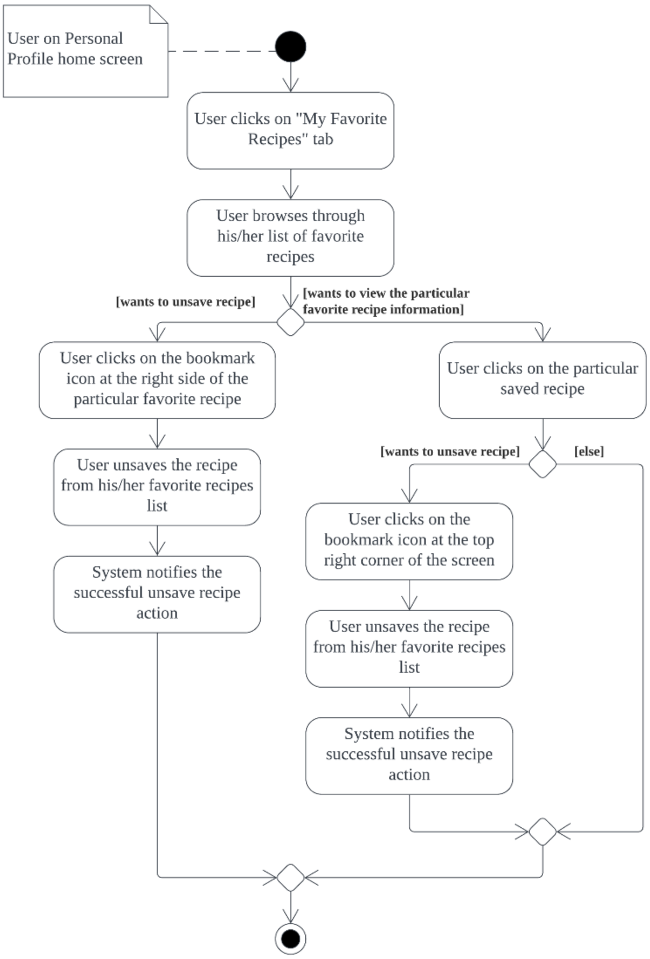

User on Personal Profile home screen
User clicks on "My Favorite Recipes" tab
User browses through his/her list of favorite recipes
[wants to unsave recipe]
[wants to view the particular favorite recipe information]
User clicks on the bookmark icon at the right side of the particular favorite recipe
User unsaves the recipe from his/her favorite recipes list
System notifies the successful unsave recipe action
User clicks on the particular saved recipe
[wants to unsave recipe]
[else]
User clicks on the bookmark icon at the top right corner of the screen
User unsaves the recipe from his/her favorite recipes list
System notifies the successful unsave recipe action

Figure 5.12: Activity Diagram for View Favorite Recipes List process.

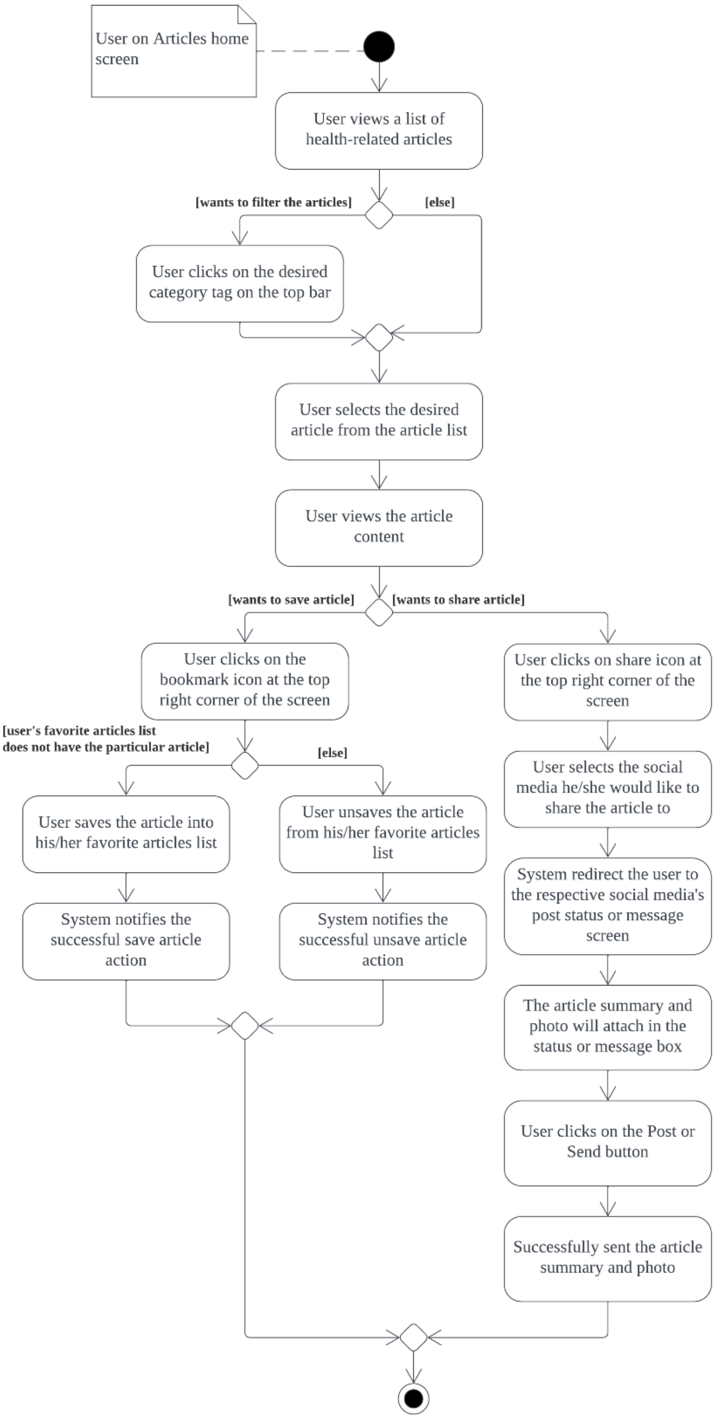

User on Articles home screen
User views a list of health-related articles
[wants to filter the articles]
[else]
User clicks on the desired category tag on the top bar
User selects the desired article from the article list
User views the article content
[wants to save article]
[wants to share article]
User clicks on the bookmark icon at the top right corner of the screen
[user's favorite articles list does not have the particular article]
[else]
User saves the article into his/her favorite articles list
User unsaves the article from his/her favorite articles list
System notifies the successful save article action
System notifies the successful unsave article action
User clicks on share icon at the top right corner of the screen
User selects the social media he/she would like to share the article to
System redirect the user to the respective social media's post status or message screen
The article summary and photo will attach in the status or message box
User clicks on the Post or Send button
Successfully sent the article summary and photo

Figure 5.13: Activity Diagram for View Health-related Articles process.

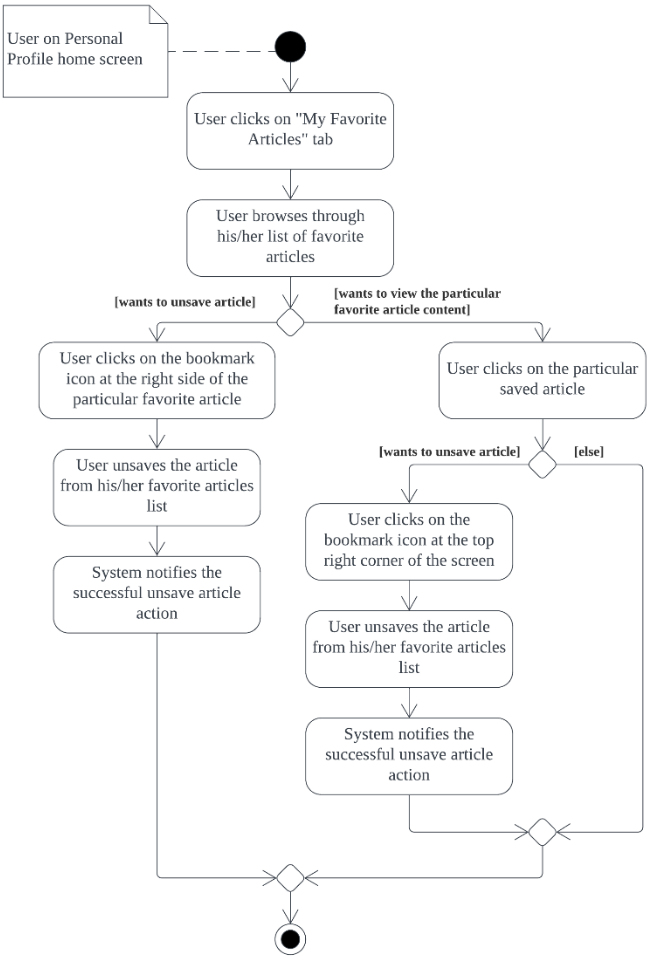

User on Personal Profile home screen
User clicks on "My Favorite Articles" tab
User browses through his/her list of favorite articles
[wants to unsave article]
[wants to view the particular favorite article content]
User clicks on the bookmark icon at the right side of the particular favorite article
User clicks on the particular saved article
User unsaves the article from his/her favorite articles list
[wants to unsave article]
[else]
System notifies the successful unsave article action
User clicks on the bookmark icon at the top right corner of the screen
User unsaves the article from his/her favorite articles list
System notifies the successful unsave article action

Figure 5.14: Activity Diagram for View Favorite Articles List process.

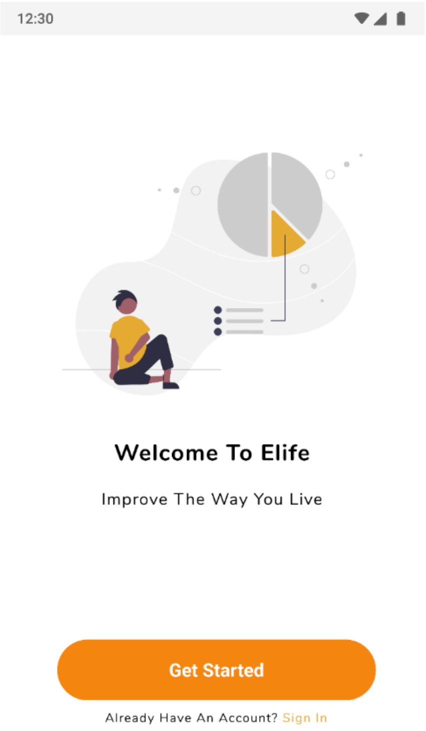
5.4 User Interface Design

A user interface (UI) prototype design was done to visualize the idea of this project solution. The high fidelity prototype design shown below covers the functional requirements and the scope of this project solution. It provided guidance to the development phase of the SDLC processes as the layout of the project solution and also the system logic. The actual prototype design was documented after the development and implementation of the project solution to do comparison with the high fidelity prototype design that had been done during the design phase of SDLC.

5.4.1 High Fidelity Prototype Design

Figure 5.1 and 5.2 shows the high fidelity prototype design of the onboarding screen and register screen that asking for the goal of user using the mobile application. The rest of the high fidelity prototype design's screens are available in Appendix D.

Onboarding



Welcome To Elife

Improve The Way You Live

Figure 5.15: Onboarding screen.

The process of registering an account

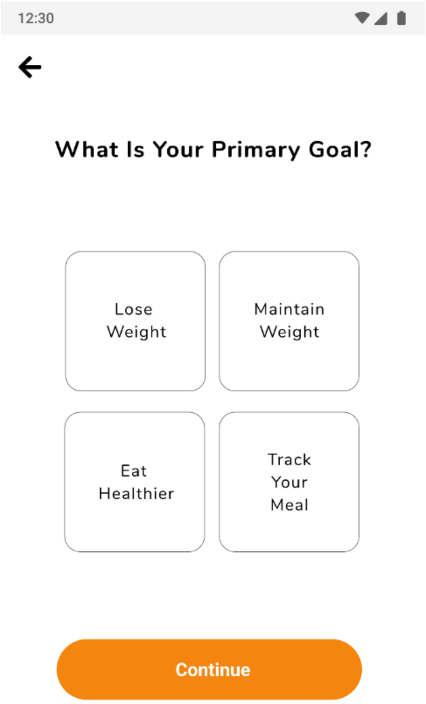
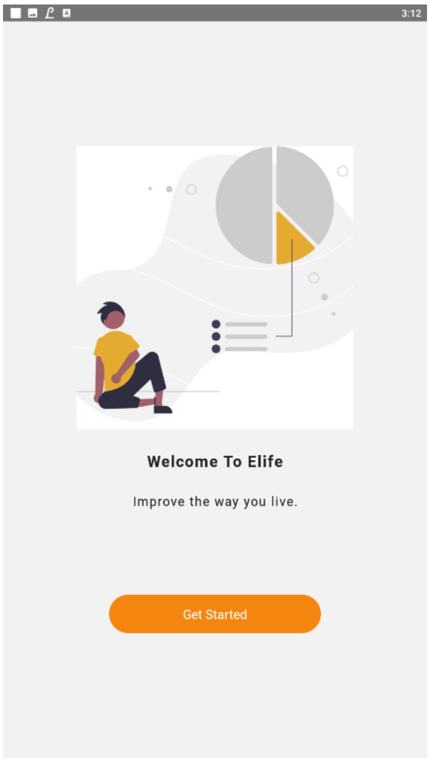


Figure 5.16: Register screen that asking for the goal.

5.4.2 Actual Prototype Design

The actual prototype design is almost the same with the high fidelity prototype design as shown in Appendix D with slightly modifications to provide a better user experience.

1. Onboarding

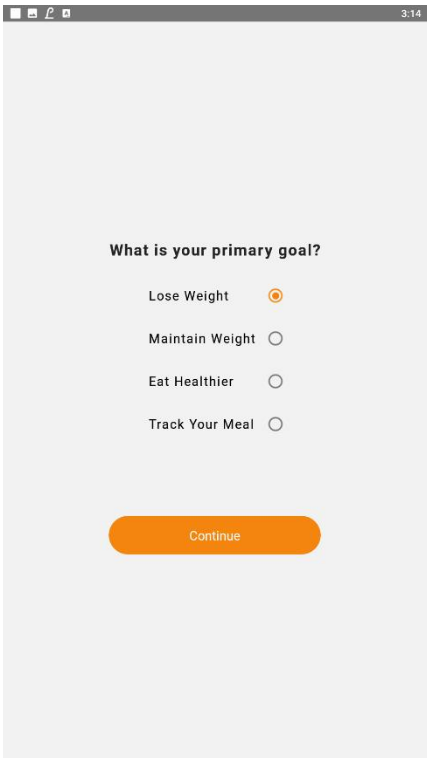


Welcome To Elife

Improve the way you live.

Figure 5.17: Actual Onboarding screen.

2. The process of registering an account



What is your primary goal?

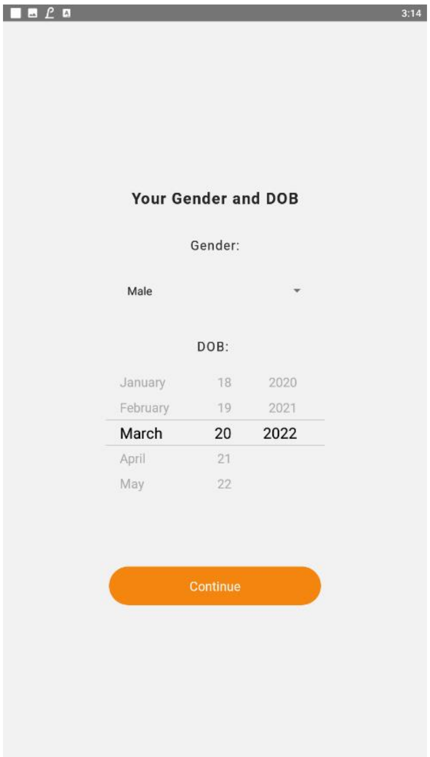
Lose Weight ●

Maintain Weight ○

Eat Healthier ○

Track Your Meal ○

Figure 5.18: Actual Register screen that asking for the goal.



Your Gender and DOB

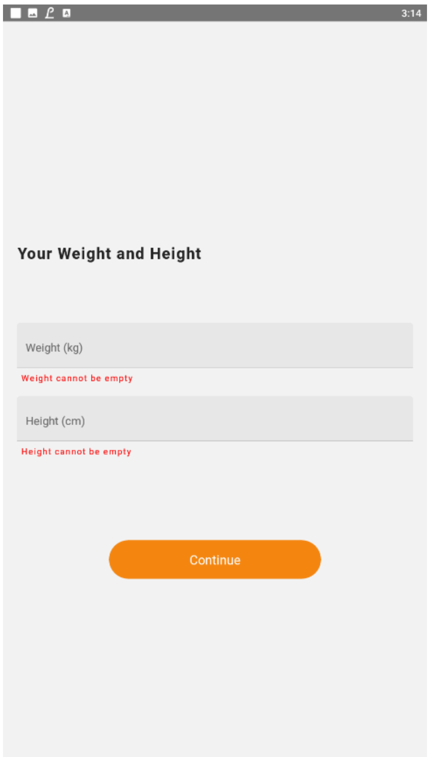
Gender:

Male

DOB:

January 18 2020
February 19 2021
March 20 2022
April 21
May 22

Figure 5.19: Actual Register screen that asking for the gender and date of birth/age.



Your Weight and Height

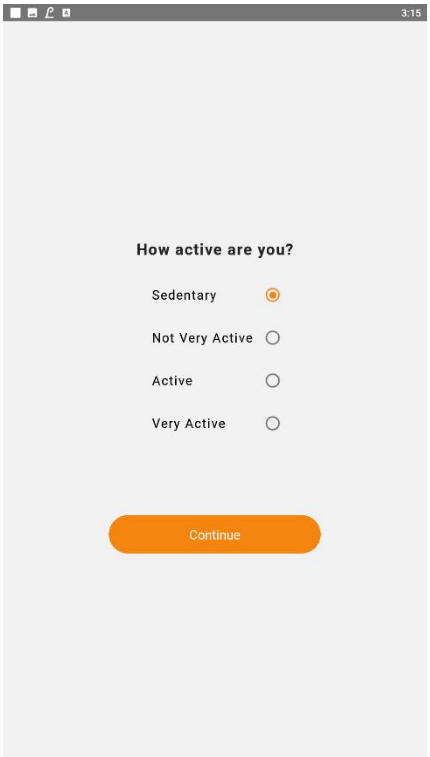
Weight (kg)

Weight cannot be empty

Height (cm)

Height cannot be empty

Figure 5.20: Actual Register screen that asking for the weight and height.



How active are you?

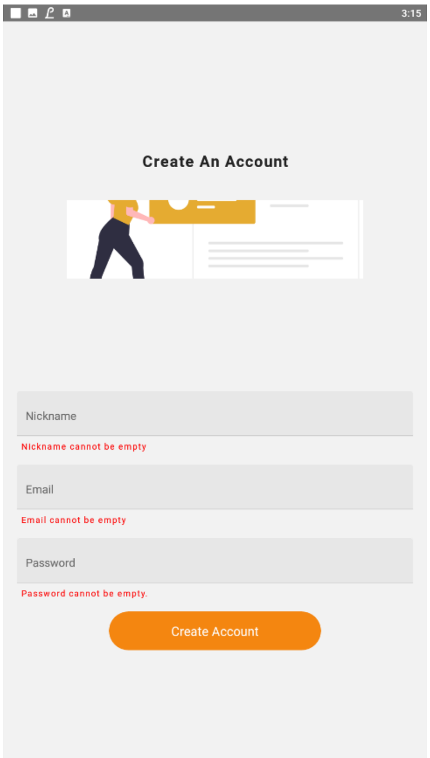
Sedentary ●

Not Very Active ○

Active ○

Very Active ○

Figure 5.21: Actual Register screen that asking for the activity level.



Create An Account

Nickname

Nickname cannot be empty

Email

Email cannot be empty

Password

Password cannot be empty.

Figure 5.22: Actual Create Account screen.

3. Sign In Account

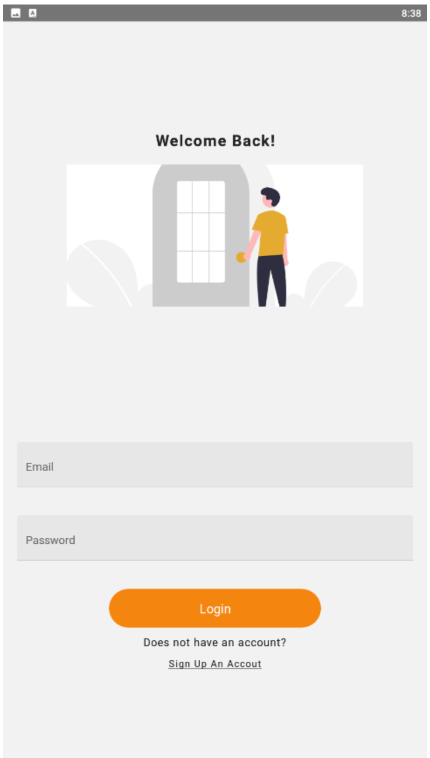
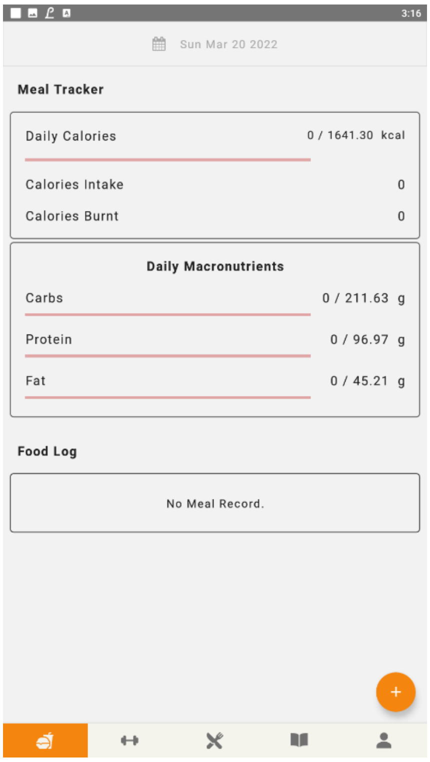


Figure 5.23: Actual Sign In screen.

4. Meal Tracker



Meal Tracker

Daily Calories 0 / 1641.30 kcal

Calories Intake 0

Calories Burnt 0

Daily Macronutrients

Carbs 0 / 211.63 g

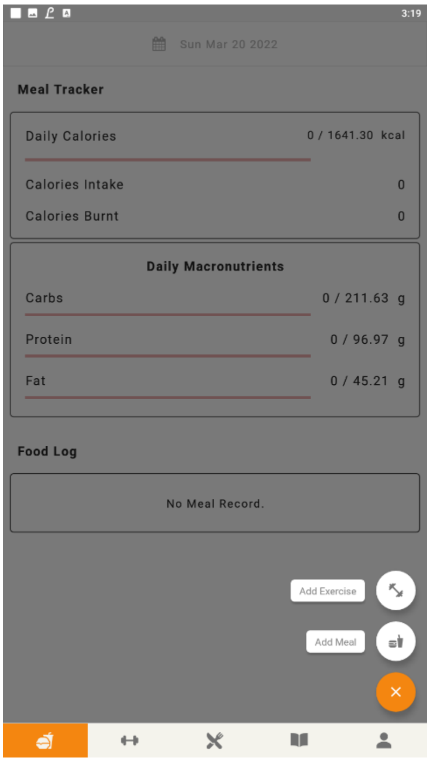
Protein 0 / 96.97 g

Fat 0 / 45.21 g

Food Log

No Meal Record.

Figure 5.24: Actual Meal Tracker home screen.



3:19

Sun Mar 20 2022

Meal Tracker

Daily Calories 0 / 1641.30 kcal

Calories Intake 0

Calories Burnt 0

Daily Macronutrients

Carbs 0 / 211.63 g

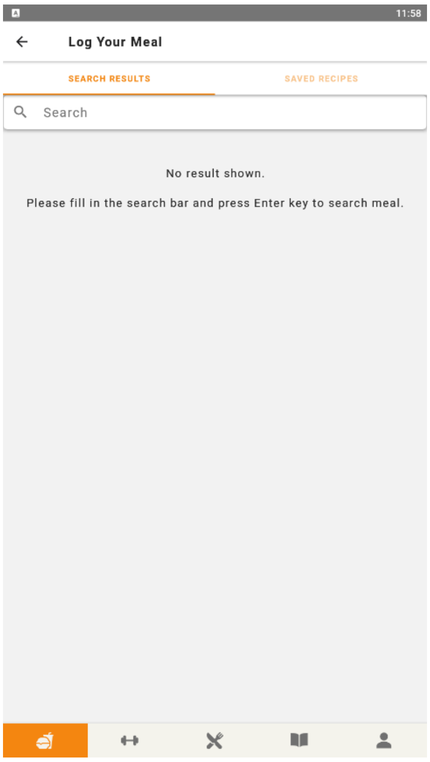
Protein 0 / 96.97 g

Fat 0 / 45.21 g

Food Log

No Meal Record.

Figure 5.25: Actual Options from the Add symbol (+) floating button.



11:58

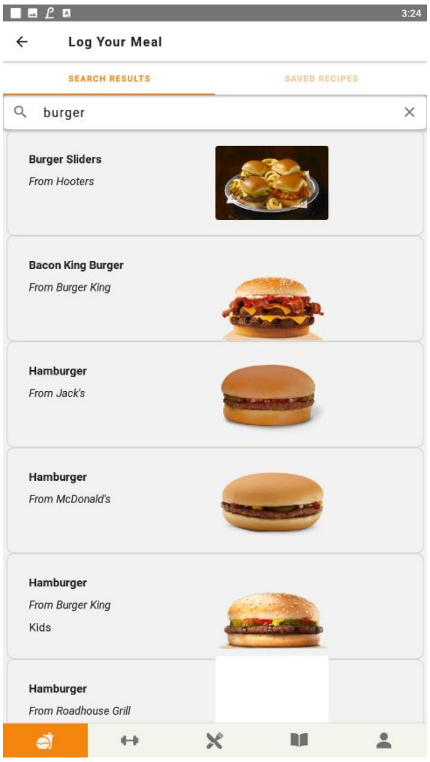
Log Your Meal

Search

No result shown.

Please fill in the search bar and press Enter key to search meal.

Figure 5.26: Actual Search Meal screen.



burger

Burger Sliders
From Hooters

Bacon King Burger
From Burger King

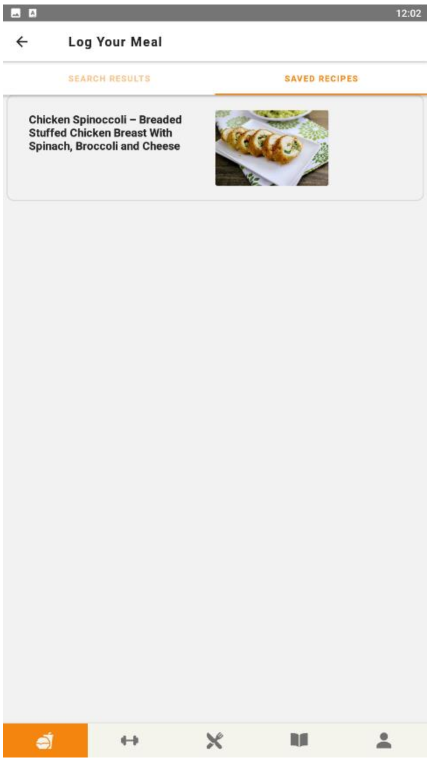
Hamburger
From Jack's

Hamburger
From McDonald's

Hamburger
From Burger King
Kids

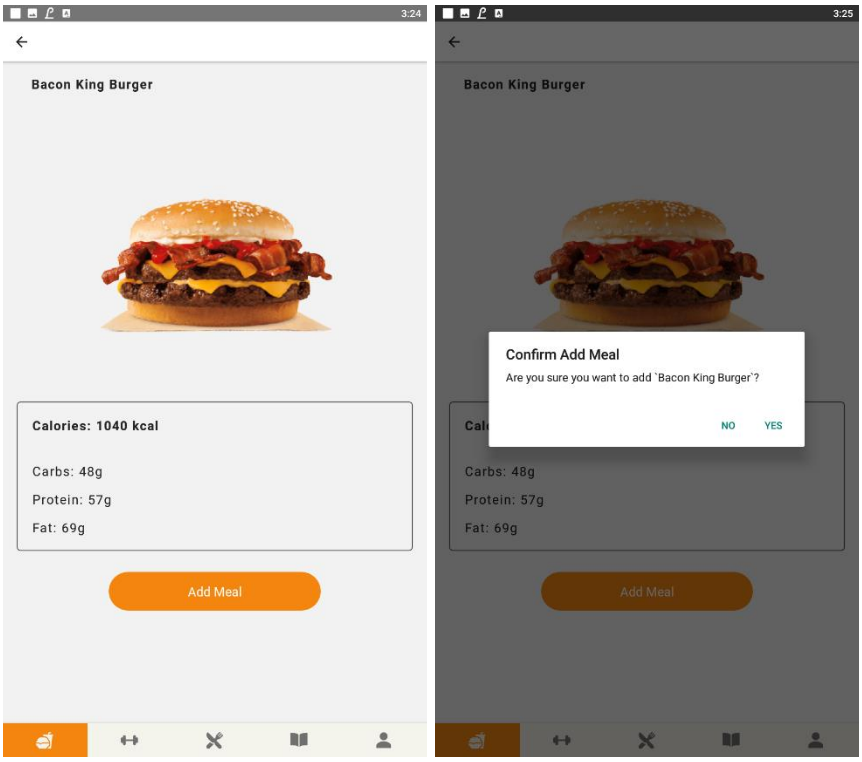
Hamburger
From Roadhouse Grill

Figure 5.27: Actual Search Meal screen with results.



Chicken Spinoccoli – Breaded Stuffed Chicken Breast With Spinach, Broccoli and Cheese

Figure 5.28: Actual Search Meal screen from Saved Recipes.



Bacon King Burger

Calories: 1040 kcal

Carbs: 48g

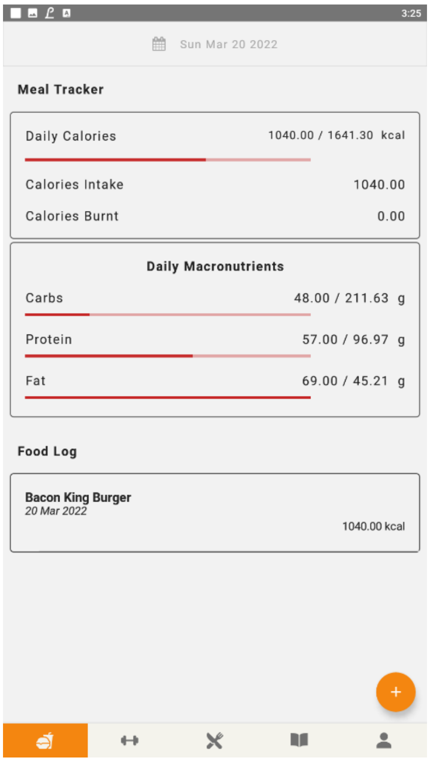
Protein: 57g

Fat: 69g

Confirm Add Meal

Are you sure you want to add `Bacon King Burger`?

Figure 5.29: Actual Meal Details screen.



Sun Mar 20 2022

Meal Tracker

Daily Calories	1040.00 / 1641.30 kcal
Calories Intake	1040.00
Calories Burnt	0.00

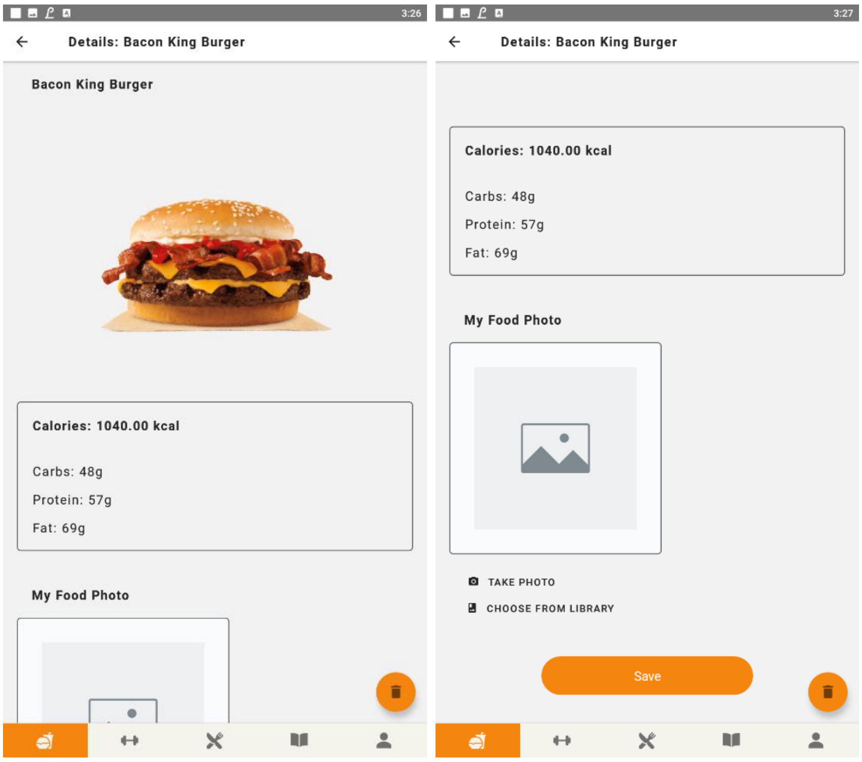
Daily Macronutrients

Carbs	48.00 / 211.63 g
Protein	57.00 / 96.97 g
Fat	69.00 / 45.21 g

Food Log

Bacon King Burger
20 Mar 2022
1040.00 kcal

Figure 5.30: Actual Meal Tracker home screen after adding a meal record.



3:26

Details: Bacon King Burger

Bacon King Burger

Calories: 1040.00 kcal

Carbs: 48g

Protein: 57g

Fat: 69g

My Food Photo

3:27

Details: Bacon King Burger

Calories: 1040.00 kcal

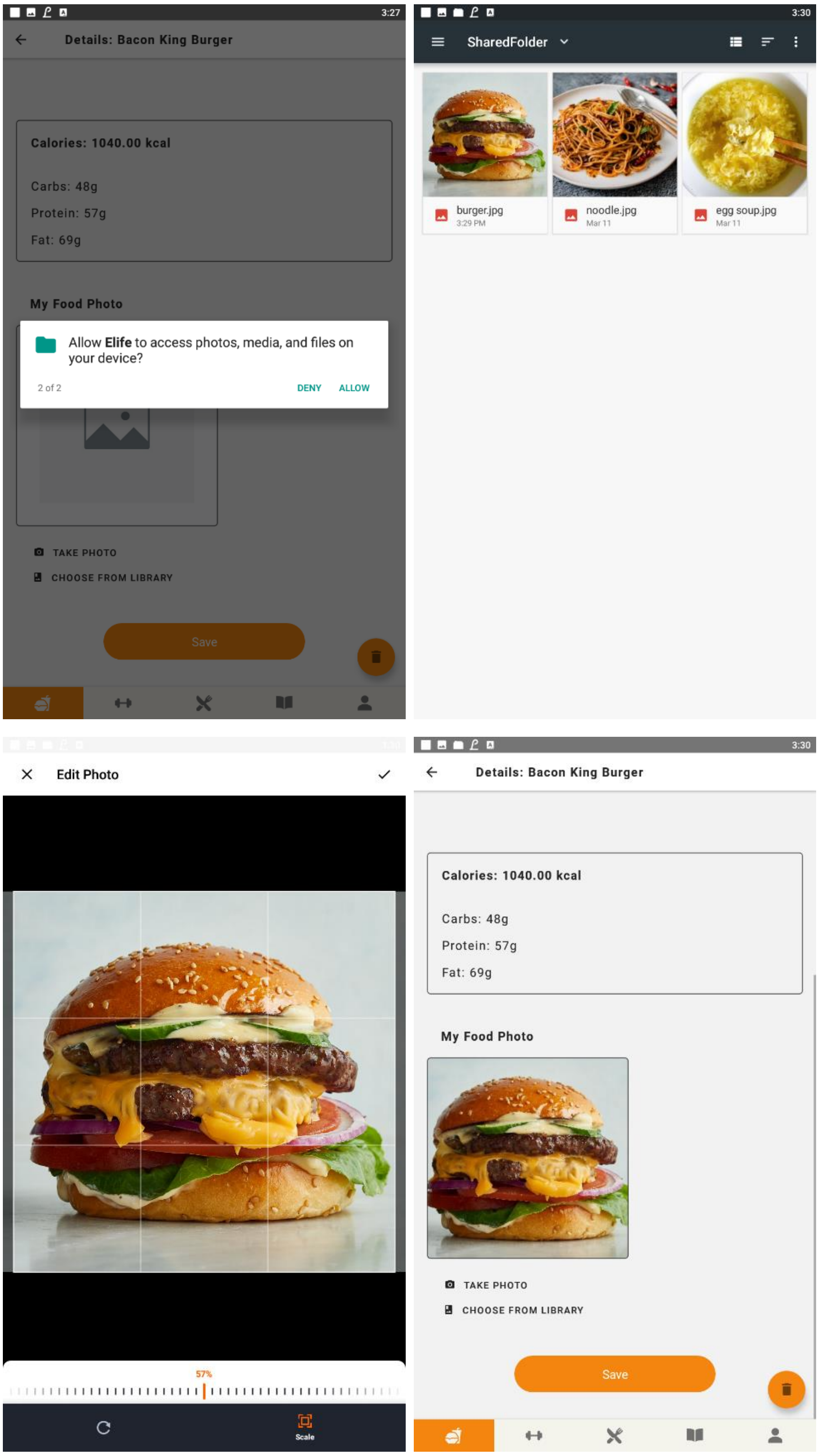
Carbs: 48g

Protein: 57g

Fat: 69g

My Food Photo

Figure 5.31: Actual Edit Food Log Record screen.



Details: Bacon King Burger

Calories: 1040.00 kcal

Carbs: 48g

Protein: 57g

Fat: 69g

My Food Photo

Allow **Elife** to access photos, media, and files on your device?

2 of 2

burger.jpg 3:29 PM

noodle.jpg Mar 11

egg soup.jpg Mar 11

Edit Photo

Details: Bacon King Burger

Calories: 1040.00 kcal

Carbs: 48g

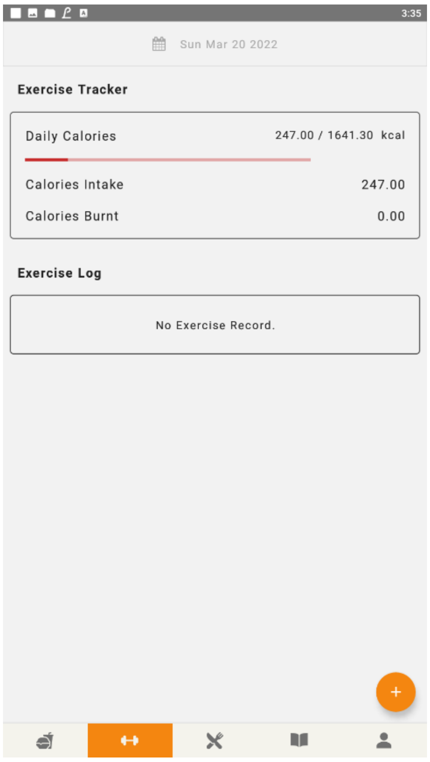
Protein: 57g

Fat: 69g

My Food Photo

Figure 5.32: Actual process of uploading a food photo for the meal record.

5. Exercise Tracker



3:35

Sun Mar 20 2022

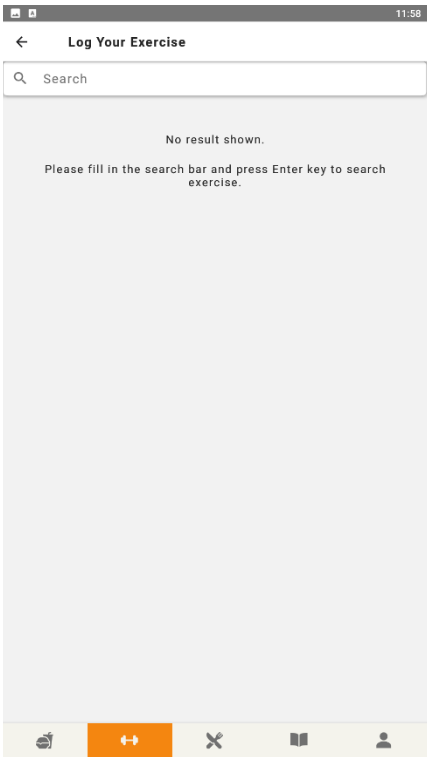
Exercise Tracker

Daily Calories	247.00 / 1641.30 kcal
Calories Intake	247.00
Calories Burnt	0.00

Exercise Log

No Exercise Record.

Figure 5.33: Actual Exercise Tracker home screen.



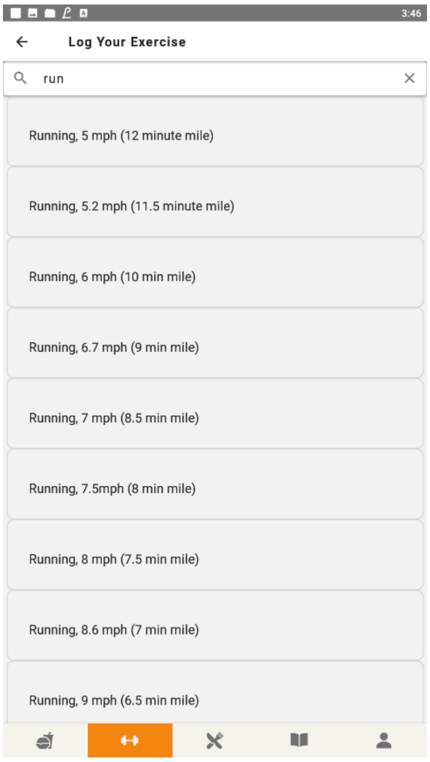
11:58

Log Your Exercise

No result shown.

Please fill in the search bar and press Enter key to search exercise.

Figure 5.34: Actual Search Exercise screen.



3:46

Log Your Exercise

run

Running, 5 mph (12 minute mile)

Running, 5.2 mph (11.5 minute mile)

Running, 6 mph (10 min mile)

Running, 6.7 mph (9 min mile)

Running, 7 mph (8.5 min mile)

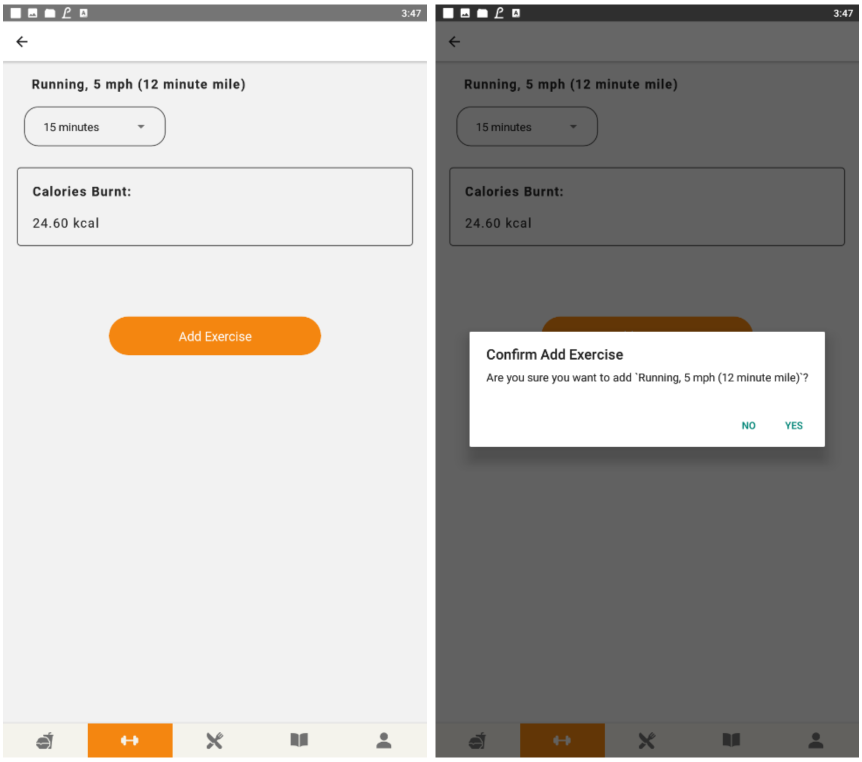
Running, 7.5mph (8 min mile)

Running, 8 mph (7.5 min mile)

Running, 8.6 mph (7 min mile)

Running, 9 mph (6.5 min mile)

Figure 5.35: Actual Search Exercise screen with results.



3:47

Running, 5 mph (12 minute mile)

15 minutes

Calories Burnt:

24.60 kcal

Add Exercise

3:47

Running, 5 mph (12 minute mile)

15 minutes

Calories Burnt:

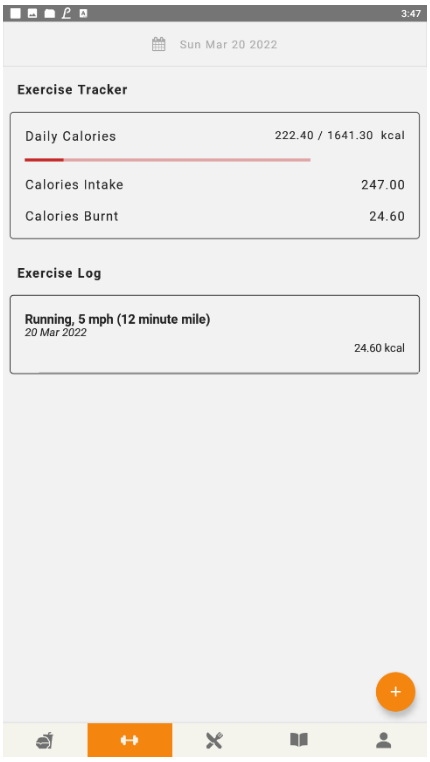
24.60 kcal

Confirm Add Exercise

Are you sure you want to add `Running, 5 mph (12 minute mile)`?

NO YES

Figure 5.36: Actual Exercise Details screen.



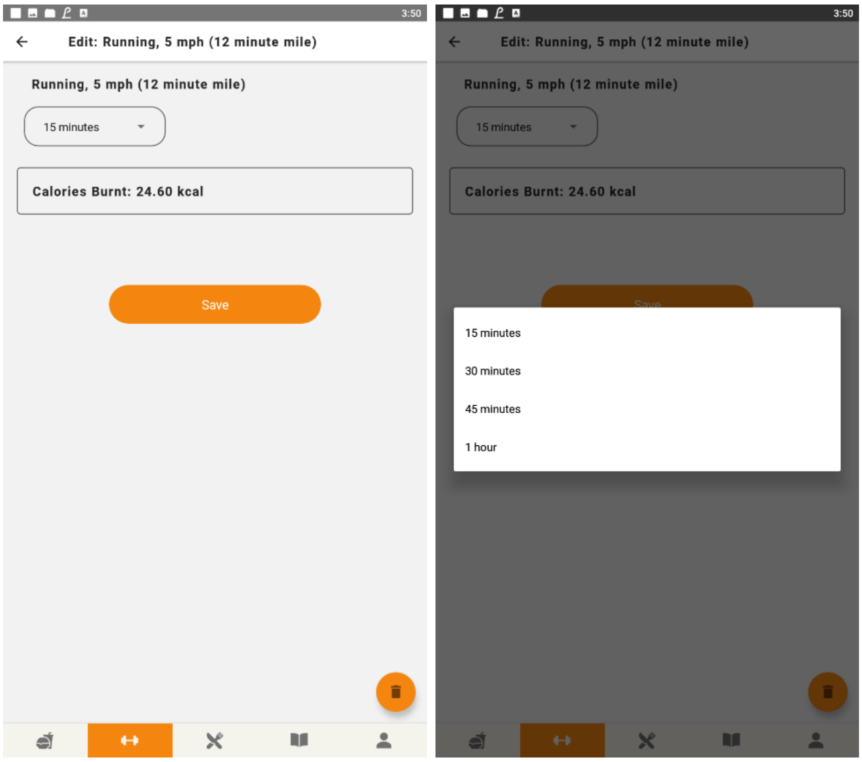
Exercise Tracker

Daily Calories	222.40 / 1641.30 kcal
Calories Intake	247.00
Calories Burnt	24.60

Exercise Log

Running, 5 mph (12 minute mile)
20 Mar 2022
24.60 kcal

Figure 5.37: Actual Exercise Tracker home screen after adding an exercise record.



Edit: Running, 5 mph (12 minute mile)

Running, 5 mph (12 minute mile)

15 minutes

Calories Burnt: 24.60 kcal

15 minutes
30 minutes
45 minutes
1 hour

Figure 5.38: Actual Edit Exercise Log Record screen.

6. Recipes

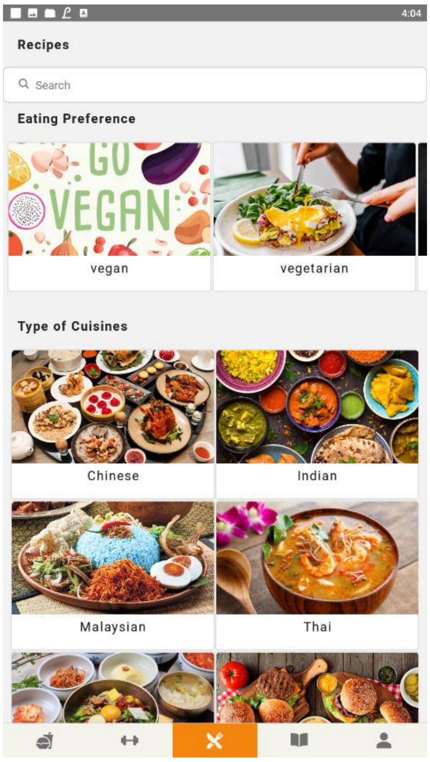


Figure 5.39: Actual Recipes screen.

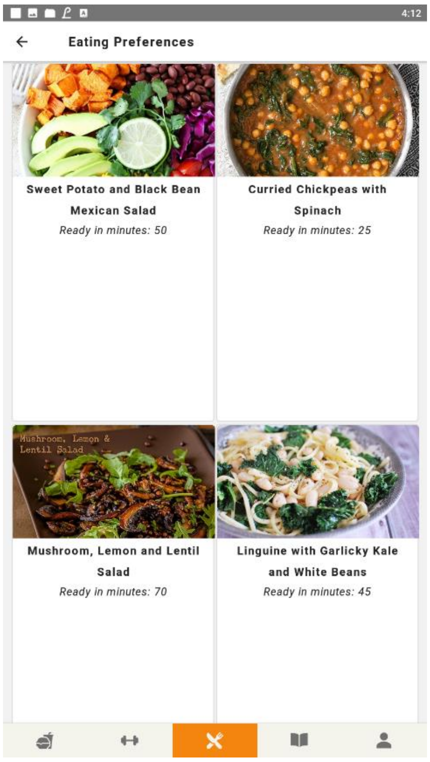
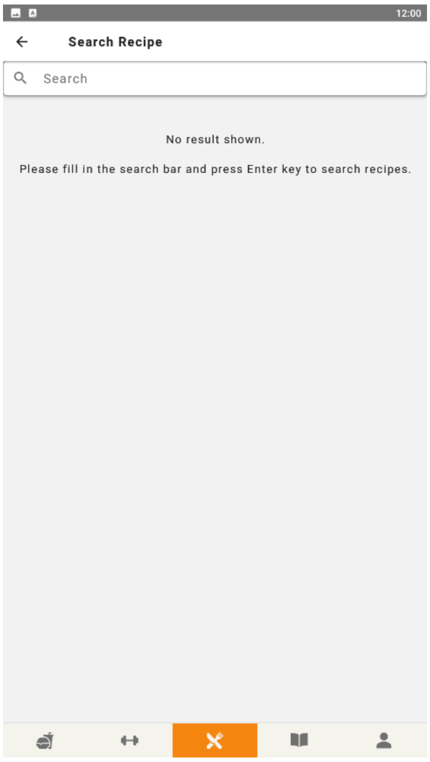

4:12
Eating Preferences
Sweet Potato and Black Bean Mexican Salad
Ready in minutes: 50
Curried Chickpeas with Spinach
Ready in minutes: 25
Mushroom, Lemon and Lentil Salad
Ready in minutes: 70
Linguine with Garlicky Kale and White Beans
Ready in minutes: 45

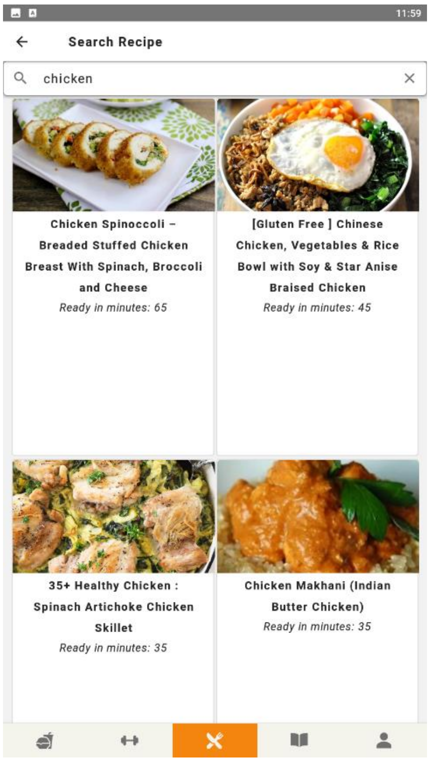
Figure 5.40: Actual Recipes List screen.



No result shown.

Please fill in the search bar and press Enter key to search recipes.

Figure 5.41: Actual Search Recipe screen.



Chicken Spinoccoli – Breaded Stuffed Chicken Breast With Spinach, Broccoli and Cheese

Ready in minutes: 65

[Gluten Free] Chinese Chicken, Vegetables & Rice Bowl with Soy & Star Anise Braised Chicken

Ready in minutes: 45

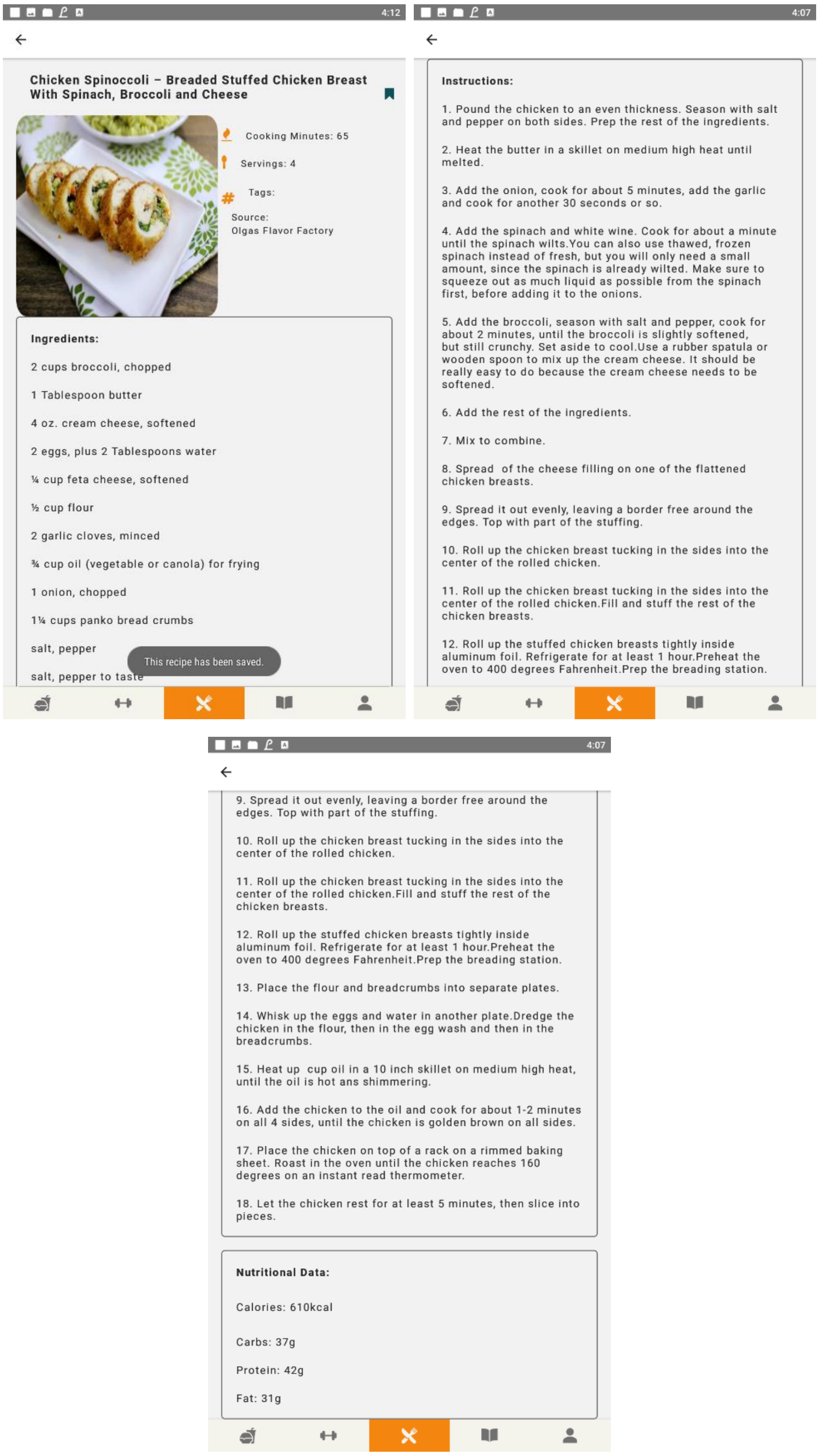
35+ Healthy Chicken : Spinach Artichoke Chicken Skillet

Ready in minutes: 35

Chicken Makhani (Indian Butter Chicken)

Ready in minutes: 35

Figure 5.42: Actual Search Recipe screen with results.



Chicken Spinoccoli – Breaded Stuffed Chicken Breast With Spinach, Broccoli and Cheese

Cooking Minutes: 65

Servings: 4

Tags:

Source:
Olgas Flavor Factory

Ingredients:

2 cups broccoli, chopped

1 Tablespoon butter

4 oz. cream cheese, softened

2 eggs, plus 2 Tablespoons water

¼ cup feta cheese, softened

½ cup flour

2 garlic cloves, minced

¾ cup oil (vegetable or canola) for frying

1 onion, chopped

1¼ cups panko bread crumbs

salt, pepper

salt, pepper to taste

This recipe has been saved.

Instructions:

1. Pound the chicken to an even thickness. Season with salt and pepper on both sides. Prep the rest of the ingredients.

2. Heat the butter in a skillet on medium high heat until melted.

3. Add the onion, cook for about 5 minutes, add the garlic and cook for another 30 seconds or so.

4. Add the spinach and white wine. Cook for about a minute until the spinach wilts.You can also use thawed, frozen spinach instead of fresh, but you will only need a small amount, since the spinach is already wilted. Make sure to squeeze out as much liquid as possible from the spinach first, before adding it to the onions.

5. Add the broccoli, season with salt and pepper, cook for about 2 minutes, until the broccoli is slightly softened, but still crunchy. Set aside to cool.Use a rubber spatula or wooden spoon to mix up the cream cheese. It should be really easy to do because the cream cheese needs to be softened.

6. Add the rest of the ingredients.

7. Mix to combine.

8. Spread of the cheese filling on one of the flattened chicken breasts.

9. Spread it out evenly, leaving a border free around the edges. Top with part of the stuffing.

10. Roll up the chicken breast tucking in the sides into the center of the rolled chicken.

11. Roll up the chicken breast tucking in the sides into the center of the rolled chicken.Fill and stuff the rest of the chicken breasts.

12. Roll up the stuffed chicken breasts tightly inside aluminum foil. Refrigerate for at least 1 hour.Preheat the oven to 400 degrees Fahrenheit.Prep the breading station.

9. Spread it out evenly, leaving a border free around the edges. Top with part of the stuffing.

10. Roll up the chicken breast tucking in the sides into the center of the rolled chicken.

11. Roll up the chicken breast tucking in the sides into the center of the rolled chicken.Fill and stuff the rest of the chicken breasts.

12. Roll up the stuffed chicken breasts tightly inside aluminum foil. Refrigerate for at least 1 hour.Preheat the oven to 400 degrees Fahrenheit.Prep the breading station.

13. Place the flour and breadcrumbs into separate plates.

14. Whisk up the eggs and water in another plate.Dredge the chicken in the flour, then in the egg wash and then in the breadcrumbs.

15. Heat up cup oil in a 10 inch skillet on medium high heat, until the oil is hot ans shimmering.

16. Add the chicken to the oil and cook for about 1-2 minutes on all 4 sides, until the chicken is golden brown on all sides.

17. Place the chicken on top of a rack on a rimmed baking sheet. Roast in the oven until the chicken reaches 160 degrees on an instant read thermometer.

18. Let the chicken rest for at least 5 minutes, then slice into pieces.

Nutritional Data:

Calories: 610kcal

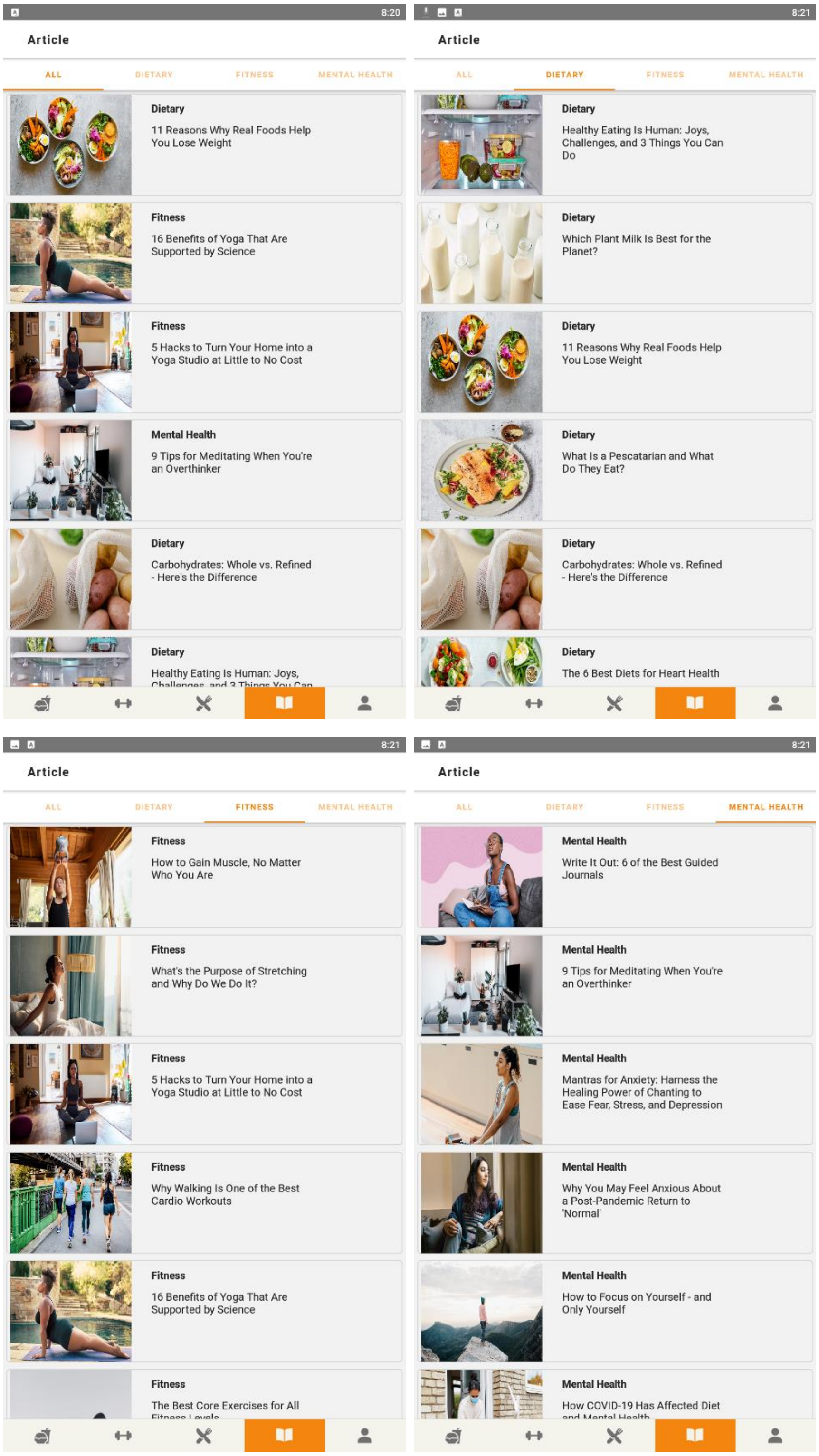
Carbs: 37g

Protein: 42g

Fat: 31g

Figure 5.43: Actual Recipe Details screen.

7. Articles



Dietary
11 Reasons Why Real Foods Help You Lose Weight

Fitness
16 Benefits of Yoga That Are Supported by Science

Fitness
5 Hacks to Turn Your Home into a Yoga Studio at Little to No Cost

Mental Health
9 Tips for Meditating When You're an Overthinker

Dietary
Carbohydrates: Whole vs. Refined - Here's the Difference

Dietary
Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can

Dietary
Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do

Dietary
Which Plant Milk Is Best for the Planet?

Dietary
11 Reasons Why Real Foods Help You Lose Weight

Dietary
What Is a Pescatarian and What Do They Eat?

Dietary
Carbohydrates: Whole vs. Refined - Here's the Difference

Dietary
The 6 Best Diets for Heart Health

Fitness
How to Gain Muscle, No Matter Who You Are

Fitness
What's the Purpose of Stretching and Why Do We Do It?

Fitness
5 Hacks to Turn Your Home into a Yoga Studio at Little to No Cost

Fitness
Why Walking Is One of the Best Cardio Workouts

Fitness
16 Benefits of Yoga That Are Supported by Science

Fitness
The Best Core Exercises for All Fitness Levels

Mental Health
Write It Out: 6 of the Best Guided Journals

Mental Health
9 Tips for Meditating When You're an Overthinker

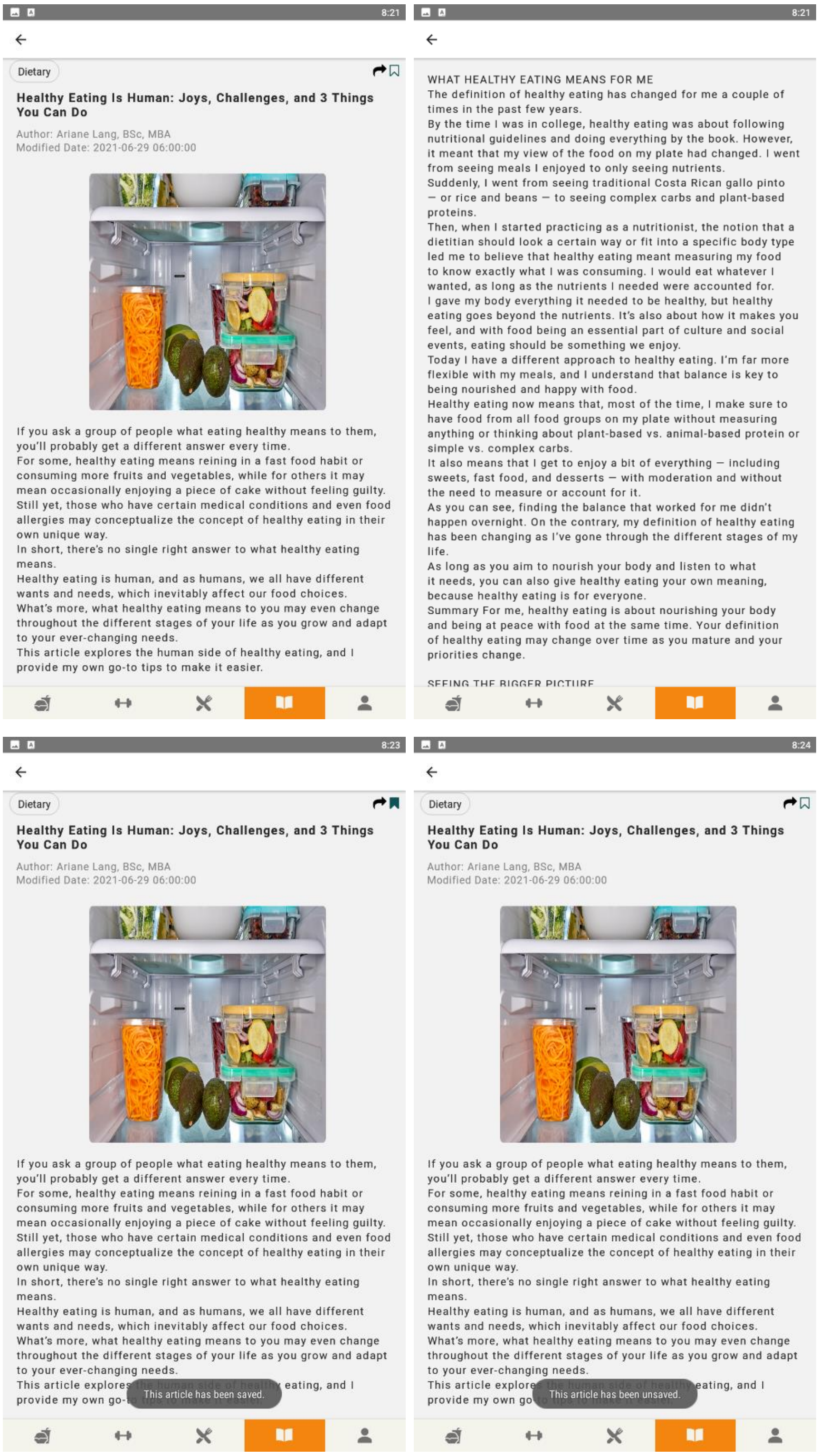
Mental Health
Mantras for Anxiety: Harness the Healing Power of Chanting to Ease Fear, Stress, and Depression

Mental Health
Why You May Feel Anxious About a Post-Pandemic Return to 'Normal'

Mental Health
How to Focus on Yourself - and Only Yourself

Mental Health
How COVID-19 Has Affected Diet and Mental Health

Figure 5.44: Actual Articles home screen.



Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do

Author: Ariane Lang, BSc, MBA
Modified Date: 2021-06-29 06:00:00

If you ask a group of people what eating healthy means to them, you'll probably get a different answer every time.
For some, healthy eating means reining in a fast food habit or consuming more fruits and vegetables, while for others it may mean occasionally enjoying a piece of cake without feeling guilty.
Still yet, those who have certain medical conditions and even food allergies may conceptualize the concept of healthy eating in their own unique way.
In short, there's no single right answer to what healthy eating means.
Healthy eating is human, and as humans, we all have different wants and needs, which inevitably affect our food choices.
What's more, what healthy eating means to you may even change throughout the different stages of your life as you grow and adapt to your ever-changing needs.
This article explores the human side of healthy eating, and I provide my own go-to tips to make it easier.

WHAT HEALTHY EATING MEANS FOR ME
The definition of healthy eating has changed for me a couple of times in the past few years.
By the time I was in college, healthy eating was about following nutritional guidelines and doing everything by the book. However, it meant that my view of the food on my plate had changed. I went from seeing meals I enjoyed to only seeing nutrients.
Suddenly, I went from seeing traditional Costa Rican gallo pinto — or rice and beans — to seeing complex carbs and plant-based proteins.
Then, when I started practicing as a nutritionist, the notion that a dietitian should look a certain way or fit into a specific body type led me to believe that healthy eating meant measuring my food to know exactly what I was consuming. I would eat whatever I wanted, as long as the nutrients I needed were accounted for.
I gave my body everything it needed to be healthy, but healthy eating goes beyond the nutrients. It's also about how it makes you feel, and with food being an essential part of culture and social events, eating should be something we enjoy.
Today I have a different approach to healthy eating. I'm far more flexible with my meals, and I understand that balance is key to being nourished and happy with food.
Healthy eating now means that, most of the time, I make sure to have food from all food groups on my plate without measuring anything or thinking about plant-based vs. animal-based protein or simple vs. complex carbs.
It also means that I get to enjoy a bit of everything — including sweets, fast food, and desserts — with moderation and without the need to measure or account for it.
As you can see, finding the balance that worked for me didn't happen overnight. On the contrary, my definition of healthy eating has been changing as I've gone through the different stages of my life.
As long as you aim to nourish your body and listen to what it needs, you can also give healthy eating your own meaning, because healthy eating is for everyone.
Summary For me, healthy eating is about nourishing your body and being at peace with food at the same time. Your definition of healthy eating may change over time as you mature and your priorities change.

SEEING THE BIGGER PICTURE

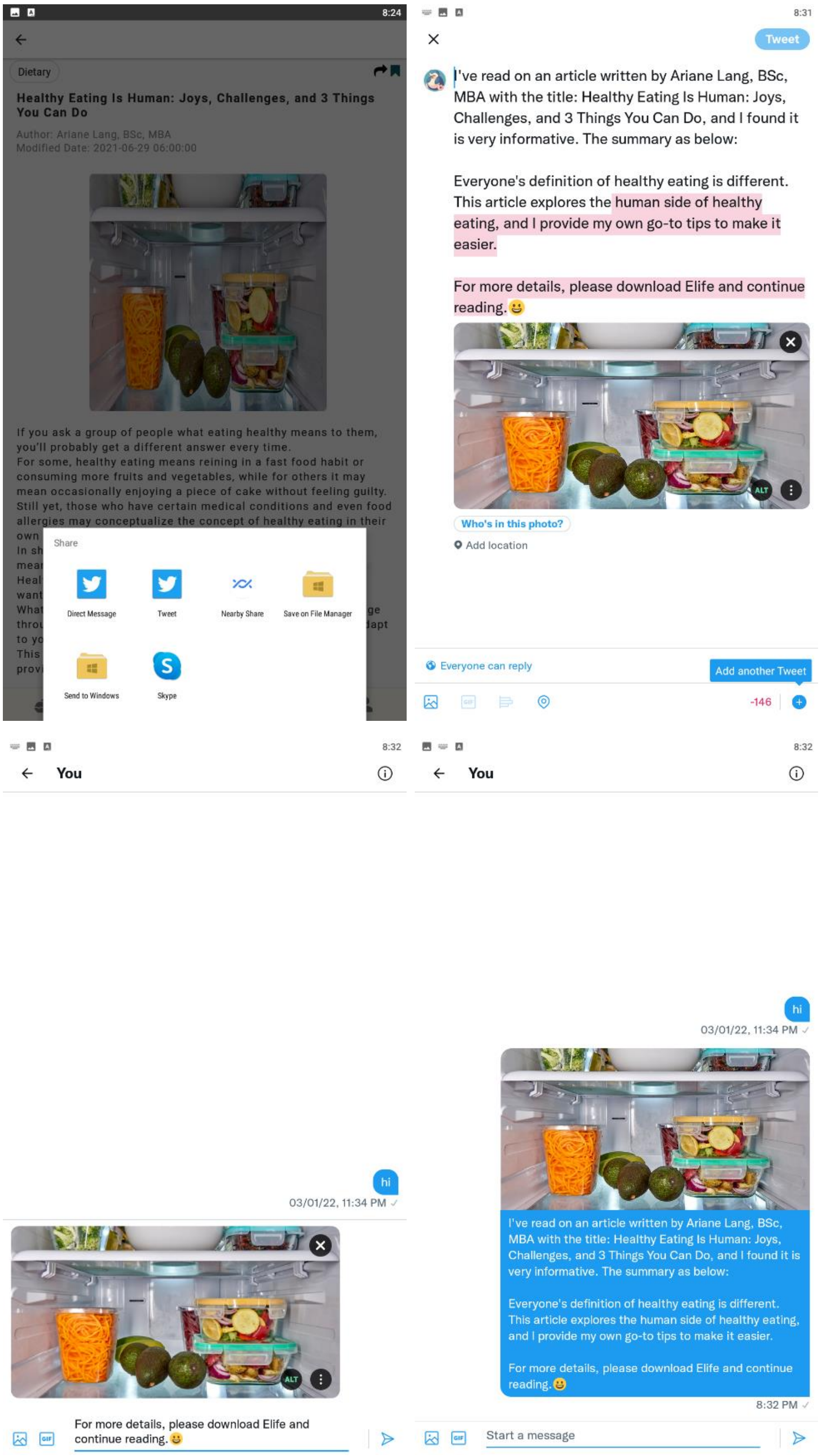
Dietary
Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do
Author: Ariane Lang, BSc, MBA
Modified Date: 2021-06-29 06:00:00
If you ask a group of people what eating healthy means to them, you'll probably get a different answer every time.
For some, healthy eating means reining in a fast food habit or consuming more fruits and vegetables, while for others it may mean occasionally enjoying a piece of cake without feeling guilty.
Still yet, those who have certain medical conditions and even food allergies may conceptualize the concept of healthy eating in their own unique way.
In short, there's no single right answer to what healthy eating means.
Healthy eating is human, and as humans, we all have different wants and needs, which inevitably affect our food choices.
What's more, what healthy eating means to you may even change throughout the different stages of your life as you grow and adapt to your ever-changing needs.
This article explores the human side of healthy eating, and I provide my own go-to tips to make it easier.

This article has been saved.

Dietary
Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do
Author: Ariane Lang, BSc, MBA
Modified Date: 2021-06-29 06:00:00
If you ask a group of people what eating healthy means to them, you'll probably get a different answer every time.
For some, healthy eating means reining in a fast food habit or consuming more fruits and vegetables, while for others it may mean occasionally enjoying a piece of cake without feeling guilty.
Still yet, those who have certain medical conditions and even food allergies may conceptualize the concept of healthy eating in their own unique way.
In short, there's no single right answer to what healthy eating means.
Healthy eating is human, and as humans, we all have different wants and needs, which inevitably affect our food choices.
What's more, what healthy eating means to you may even change throughout the different stages of your life as you grow and adapt to your ever-changing needs.
This article explores the human side of healthy eating, and I provide my own go-to tips to make it easier.

This article has been unsaved.

Figure 5.45: Actual Article Contents screen.



Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do

Author: Ariane Lang, BSc, MBA
Modified Date: 2021-06-29 06:00:00

If you ask a group of people what eating healthy means to them, you'll probably get a different answer every time.
For some, healthy eating means reining in a fast food habit or consuming more fruits and vegetables, while for others it may mean occasionally enjoying a piece of cake without feeling guilty.
Still yet, those who have certain medical conditions and even food allergies may conceptualize the concept of healthy eating in their own
In sh
mear
Heal
want
What ge
throu dapt
to yo
This
provi

I've read on an article written by Ariane Lang, BSc, MBA with the title: Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do, and I found it is very informative. The summary as below:

Everyone's definition of healthy eating is different. This article explores the human side of healthy eating, and I provide my own go-to tips to make it easier.

For more details, please download Elife and continue reading.😃

hi
03/01/22, 11:34 PM ✓

For more details, please download Elife and continue reading.😃

hi
03/01/22, 11:34 PM ✓

I've read on an article written by Ariane Lang, BSc, MBA with the title: Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do, and I found it is very informative. The summary as below:

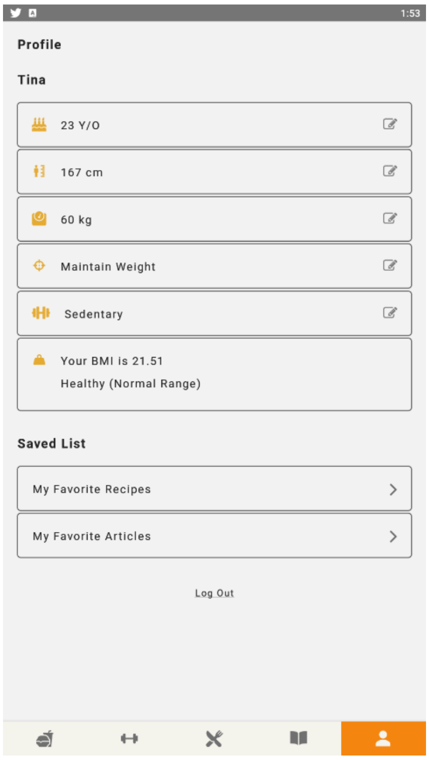
Everyone's definition of healthy eating is different. This article explores the human side of healthy eating, and I provide my own go-to tips to make it easier.

For more details, please download Elife and continue reading.😃

8:32 PM ✓

Figure 5.46: Actual process of sharing the article summary to social media (Twitter's Tweet / Direct Message).

8. Personal Profile



1:53

Profile

Tina

23 Y/O

167 cm

60 kg

Maintain Weight

Sedentary

Your BMI is 21.51
Healthy (Normal Range)

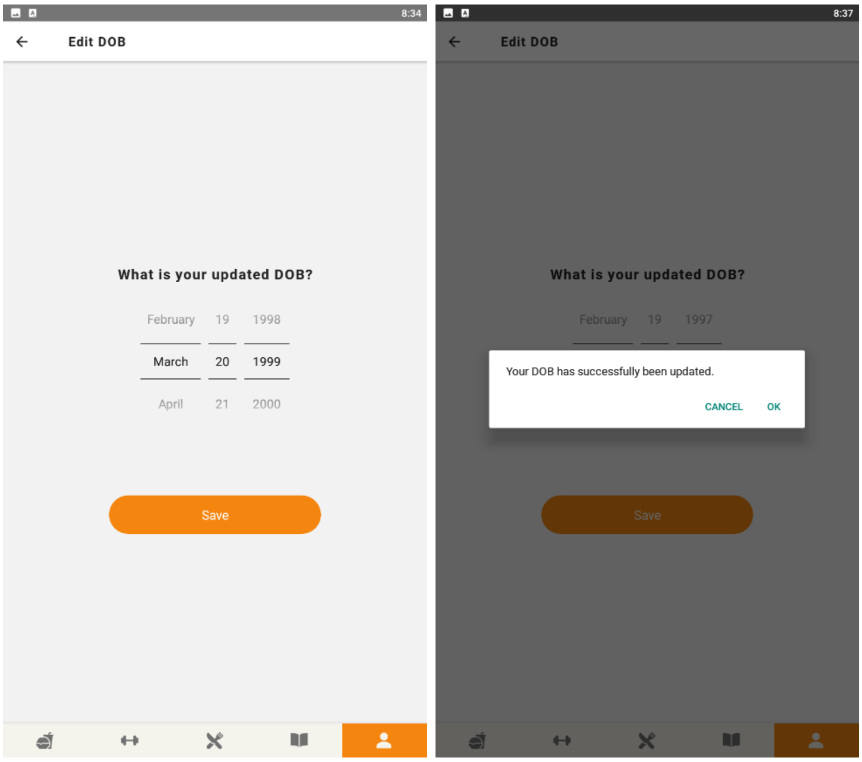
Saved List

My Favorite Recipes

My Favorite Articles

Log Out

Figure 5.47: Actual Personal Profile screen.



8:34

Edit DOB

What is your updated DOB?

February	19	1998
March	20	1999
April	21	2000

Save

8:37

Edit DOB

What is your updated DOB?

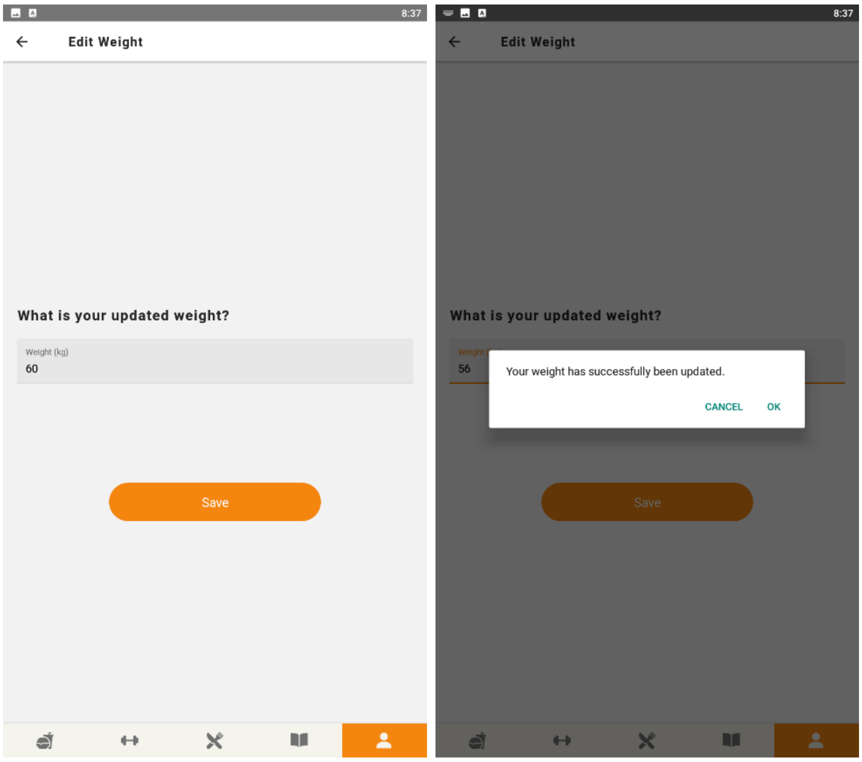
February 19 1997

Your DOB has successfully been updated.

CANCEL OK

Save

Figure 5.48: Actual Edit Date of Birth screen.



Edit Weight

What is your updated weight?

Weight (kg)
60

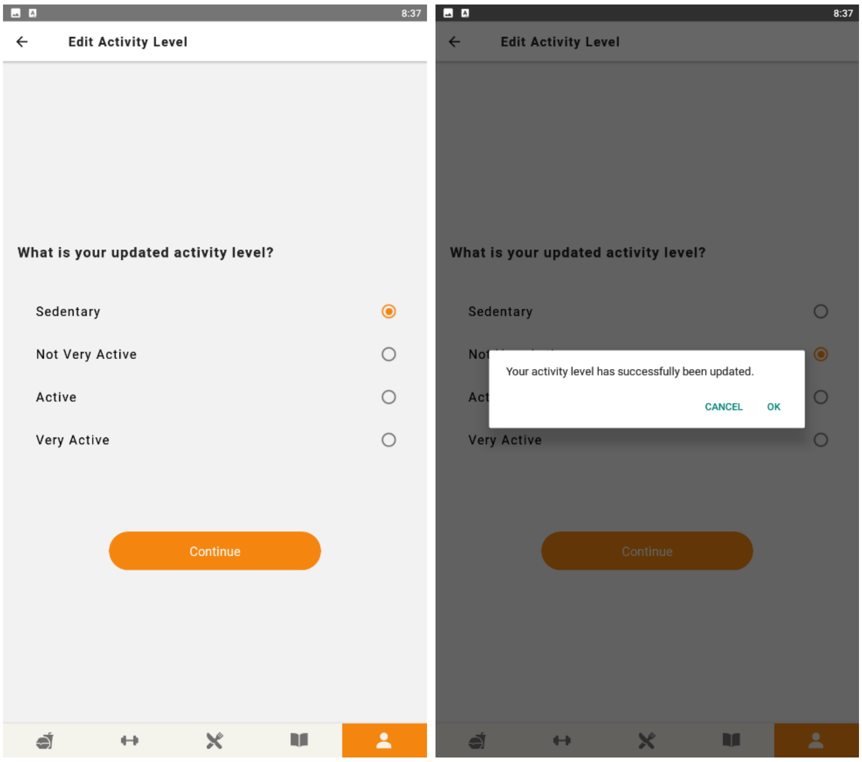
Edit Weight

What is your updated weight?

Weight
56

Your weight has successfully been updated.

Figure 5.49: Actual Edit Weight screen.



Edit Activity Level

What is your updated activity level?

Sedentary ●
Not Very Active ○
Active ○
Very Active ○

Edit Activity Level

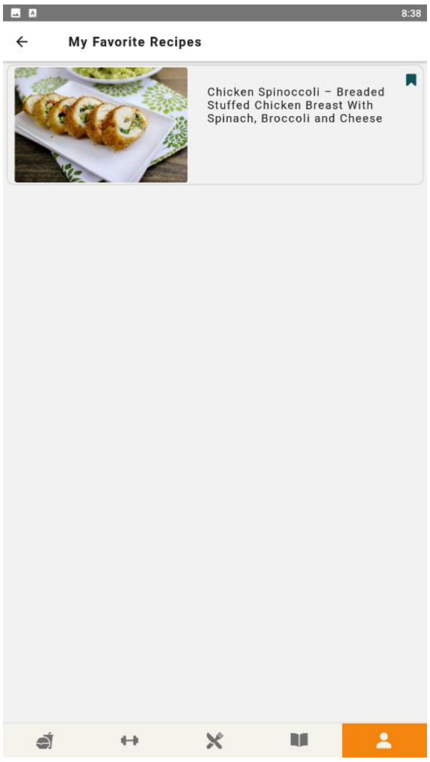
What is your updated activity level?

Sedentary ○
Not ●
Act ○
Very Active ○

Your activity level has successfully been updated.

Figure 5.50: Actual Edit Activity Level screen.

9. Favorite Recipes List



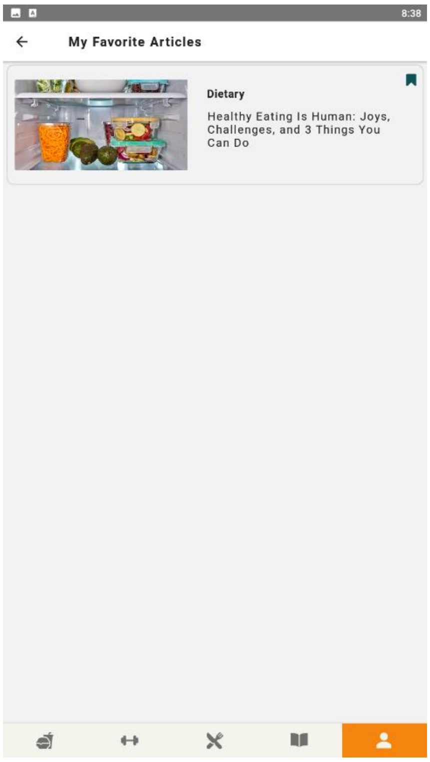
8:38

My Favorite Recipes

Chicken Spinoccoli – Breaded Stuffed Chicken Breast With Spinach, Broccoli and Cheese

Figure 5.51: Actual Favorite Recipes screen.

10. Favorite Articles List



8:38

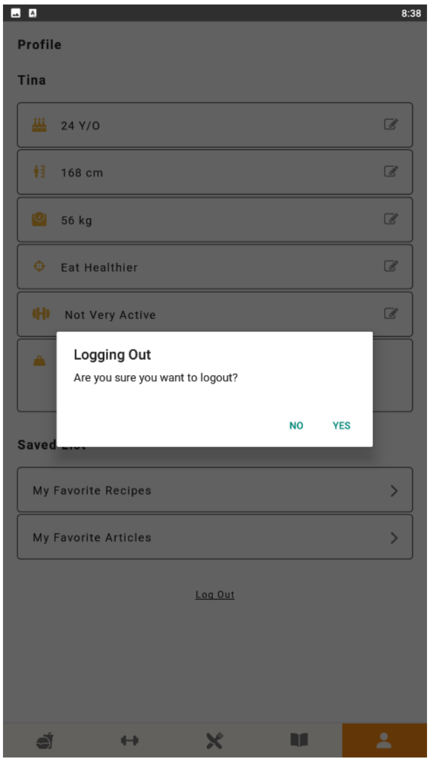
My Favorite Articles

Dietary

Healthy Eating Is Human: Joys, Challenges, and 3 Things You Can Do

Figure 5.52: Actual Favorite Articles screen.

11. Log Out



Profile

Tina

24 Y/O

168 cm

56 kg

Eat Healthier

Not Very Active

Logging Out

Are you sure you want to logout?

Figure 5.53: Actual Log Out alert box.

5.5 Summary

In short, this chapter displays a series of diagrams to act as a guidelines during the system implementation phase of SDLC. The detailed system architecture design illustrates the architecture of the implemented system and how the implemented system communicates with the database with the help of Flask REST API server. Not only that, the implemented system has also interacted with the external APIs to get the required information to fulfil the modules and functionalities listed in the project scope from Chapter 1 and functional requirements from Chapter 4.

Moreover, the data models were produced to show the design of database structure according to the functional requirements and the relationships between entities. Data dictionary was created to list out the attributes details for each table. A collection of activity diagrams was produced to demonstrate the application logic of the implemented system, and how the user and system perform during the specific activity occurred.

Lastly, the high fidelity prototype design was drafted out to guide the system UI implementation and the actual prototype design was later been documented as an evidence of comformance to the high fidelity prototype design done in the design phase of SDLC.

CHAPTER 6

IMPLEMENTATION

6.1 Introduction

This chapter documents the implementation details of the project solution. The third party libraries were integrated into the implemented system such as React Navigation and React Native Paper. Moreover, there is a list of API endpoints used by the implemented system to communicate with the system database as well as the external API's database. After the development and web service testing, the implemented system was deployed by using Heroku. This was to make the implemented system going live and could be used by the target users for usability testing, UAT and in the future.

6.2 React Navigation

React Navigation is the third party standalone library that can be used to apply navigation functionalities on React Native mobile applications. There are various kinds of navigators provided by React Navigation such as stack navigator, tab navigator and drawer navigator. In this project, stack navigators and top and bottom tab navigators were utilized to build the project solution that is able to navigate through the screens with a proper navigation logic.

Stack navigation is the most common navigation option used for almost all mobile applications regardless of the application category it falls in. This is because the stack navigation helps to group the screens under the same module into a stack, making the navigation through the screens of one module become easier and without causing much overhead. Hence, the stack navigation was applied in the implemented system as the main navigator and inside the main navigator, there were 5 sub stack navigators representing the 5 different modules that were included in the implemented system.

Furthermore, the bottom tab navigation was applied as the sub navigator in the implemented system but it shows in the implemented system as the main visible navigator. Figure 6.1 shows the bottom tab navigation bar of the implemented system. The bottom tab navigation bar allows the users to navigate through different modules of the implemented system.

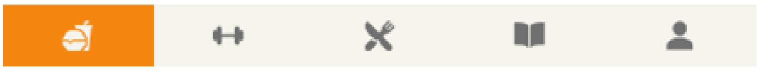


Figure 6.1: Bottom tab navigation bar of the implemented system.

Moreover, the top tab navigation bar was utilized to help filter the sources of meal records in the search meal screen and also the categories of articles in the articles home screen. Figure 6.2 and 6.3 display the top tab navigation bars of the implemented system.

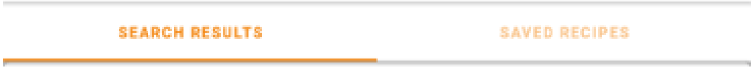


Figure 6.2: Bottom tab navigation bar of the implemented system.

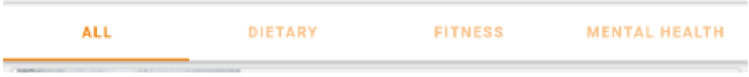


Figure 6.3: Bottom tab navigation bar of the implemented system.

The hierarchy of the navigators used by the implemented system is illustrated in Figure 6.4. The concept of nested navigations was applied in order to build the project solution to function well and navigate according to the users' intention.

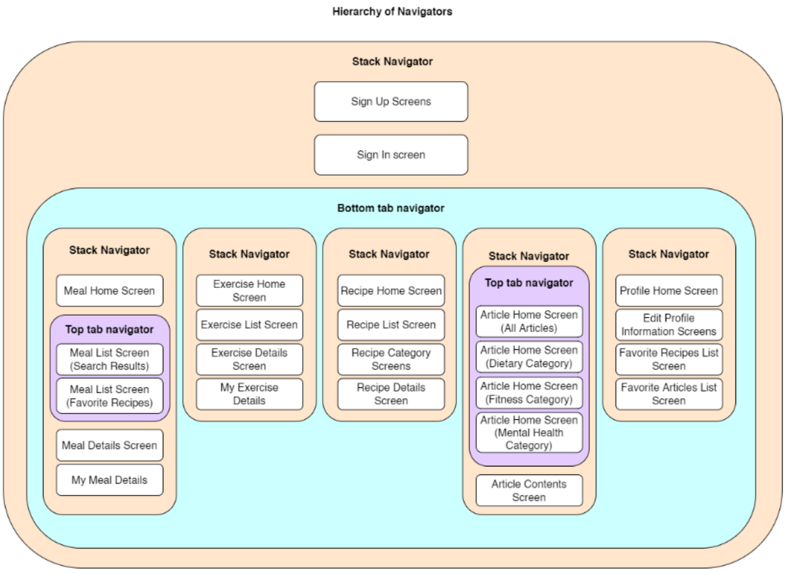
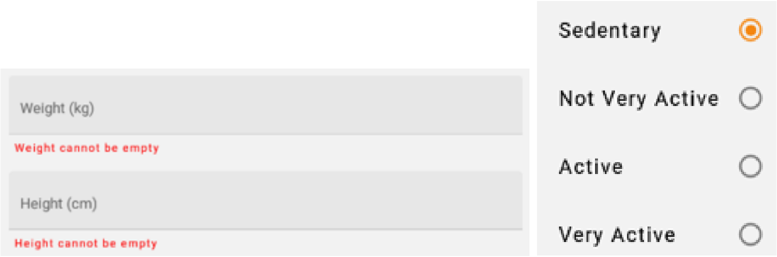


Figure 6.4: Hierarchy of navigators used by the implemented system.

6.3 React Native Paper

React Native Paper is one of the popular third party UI libraries. It provides a collection of UI elements for the React Native developers to develop a mobile application with aesthetic UI design that can attract the users from consistently using the mobile application in their daily life. In this project, the UI elements such as text input, radio button, progress bar, floating action button (FAB), chip, search bar and card are used. The attributes of the UI elements can be set based on preference. The text inputs and radio buttons are used in the sign up, sign in and update personal profile information screen. Figure 6.5 shows a part of the usage of text inputs and radio buttons in the implemented system.

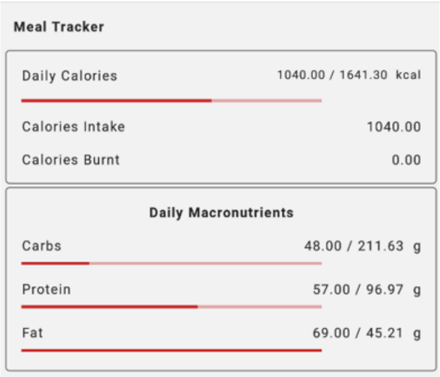


Weight (kg)
Weight cannot be empty
Height (cm)
Height cannot be empty

Sedentary ●
Not Very Active ○
Active ○
Very Active ○

Figure 6.5: Usage of text inputs and radio buttons UI elements.

The progress bars and FABs are used in the meal tracker and exercise tracker home screen. The progress bars illustrate the daily net calories and macronutrients intake of the user as shown in Figure 6.6, whereas the FABs allow the user to perform add meal or exercise record as shown in Figure 6.7.



Meal Tracker

Daily Calories 1040.00 / 1641.30 kcal
Calories Intake 1040.00
Calories Burnt 0.00

Daily Macronutrients
Carbs 48.00 / 211.63 g
Protein 57.00 / 96.97 g
Fat 69.00 / 45.21 g

Figure 6.6: Usage of progress bars UI elements.

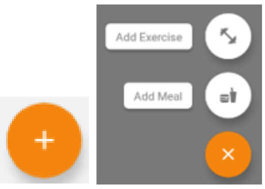


Figure 6.7: Usage of FABs UI elements.

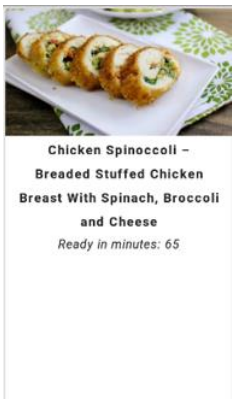
Moreover, chip is used in the implemented system to show the article category as a chip on the article contents screen. Figure 6.8 shows the usage of chip UI element.

Figure 6.8: Usage of chip UI element.

The search bar UI element is used in the meal, exercise and recipe list screen to allow the user to perform search action to find for the desired meal, exercise and recipe results. Figure 6.9 shows the usage of search bar UI element.

Figure 6.9: Usage of search bar UI element.

Furthermore, the card UI element is used in the recipe module of the implemented system to display the recipe name and photo before viewing the details of the recipe. The usage of card UI element is as shown in Figure 6.10.



Chicken Spinoccoli – Breaded Stuffed Chicken Breast With Spinach, Broccoli and Cheese

Ready in minutes: 65

Figure 6.10: Usage of card UI element.

6.4 API List

The API endpoints are used intensively in the implemented system to assist in the communication between the implemented system and the database. The implemented system calls the respective API endpoint to retrieve or update the specific information. There are internal and external API endpoints being used by the implementation system and these API endpoints with the respective description and the required parameters were documented as shown in Table 6.1.

Table 6.1: List of API endpoints

Endpoint	Description	Parameters required
https://fitness-calculator.p.rapidapi.com/macrocalculator?age={age}&gender={gender}&height={height}&weight={weight}&activitylevel={activityLvNum}&goal={goalScale}	To calculate the daily net calories and macronutrients limit of the user based on the user's information	Age, gender, height, weight, activity level number, goal scale
/api/users/{email}	To retrieve the specific user's information	User's email address
/api/users	To insert a new user record	User's information
https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/food/menuItems/search?query={searchQuery}&offset=0&number=50&minCalories=0&maxCalories=5000&minProtein=0&minFat=0&minCarbs=0	To retrieve a list of meals based on the search query	Meal keyword (searchQuery)
/api/meals	To insert a new meal record	Meal information

/api/meals/{meal_id}	To retrieve, update or delete the specific meal record	Meal ID
/api/intake	To insert a new intake record	Intake information
/api/intake/{email}/{record_date}	To retrieve, update or delete the intake information of the specific user and the specific record date	User’s email address and the intake’s record date
/api/load_exercises/{matchText}	Search the exercise activities based on the exercise keyword entered	Exercise keword (matchText)
/api/exercises	To insert new exercise activity record	Exercise record information
/api/exercises/{exercise_id}	To retrieve, update or delete the specific exercise activity record	Exercise ID
https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/search?query=food&instructionsRequired=true&diet={searchQuery}&number=10&offset=0&type=main%20course	To retrieve a list of matched recipes based on the search query	Recipe keyword (searchQuery)
https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/informationBulk?ids={recipeID}	To retrieve the specific recipe details	Recipe ID
https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/{recipeID}/nutritionWidget.json	To retrieve the nutritional data of the specific recipe	Recipe ID

/api/fav_recipes	To insert a new favorite recipe record	Recipe information
/api/fav_recipes/{user_email}/{recipe_index_no}	To retrieve or delete the specific favorite recipe of the user	User's email address and recipe index number
/api/article_source	To retrieve a list of article sources	-
/api/article_source/{article_tag}	To retrieve a list of article sources based on the category	Article category (article_tag)
https://lexper.p.rapidapi.com/v1.1/extract?url={article_url}&media=1	To retrieve the article contents based on the article URL	Article URL
/api/fav_articles	To insert a new favorite article record	Article information
/api/fav_articles/{user_email}/{article_title}	To retrieve or delete the specific favorite article of the user	User's email address and article title
/api/user_age/{email}	To update the user's age	User's email address
/api/user_daily_nutrition/{email}	To update the user's daily net calories and macronutrients limit	User's email address
/api/user_weight/{email}	To update the user's weight	User's email address
/api/user_height/{email}	To update the user's height	User's email address
/api/user_goal/{email}	To update the user's goal	User's email address
/api/user_activity_lv/{email}	To update the user's activity level	User's email address

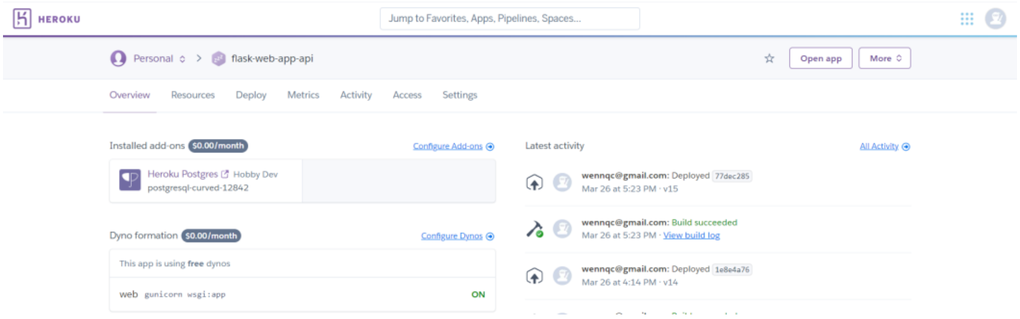
/api/fav_recipes/{email}	To retrieve the favorite recipes of the specific user	User’s email address
/api/fav_articles/{email}	To retrieve the favorite articles of the specific user	User’s email address

6.5 Heroku Deployment

Heroku is used to host the Flask REST API and the PostgreSQL database that are being used by the implemented system. Heroku is a cloud platform that provides the developers a space to deploy their backend server and APIs either for free or with a monthly fee depending on the scale of the server and APIs. In this project, the free version of Heroku is sufficient enough to support the current deployment of the Flask REST API and PostgreSQL database.

In this project, before deploying the Flask REST API and PostgreSQL database to Heroku, the necessary tools such as Git, Python and Heroku CLI were required to install on the local machine. After that, a Python virtual environment was created to separate the Python environment of the Flask REST API with other Python’s project that are available in the local machine. The necessary libraries and dependencies for the Flask REST API were installed into the virtual environment.

Furthermore, Procfile was created to declare the commands executed by the application’s container on Heroku and it was placed in the root folder of the Flask REST API’s project. After that, a new Heroku app was set up to prepare for the deployment of Flask REST API and PostgreSQL database. A remote access of the Flask REST API was created by creating and pushing in the Flask REST API to GitHub. Then, Heroku Postgres add-on is used as the resource of the Heroku app to help setting up the PostgreSQL database and creating the database credentials that is used by Flask REST API to access the PostgreSQL database. Finally, the Flask REST API is live and can be accessed using the specific URL generated by Heroku.



Heroku Postgres Hobby Dev
postgresql-curved-12842

This app is using **free** dynos

web `gunicorn wsgi:app` ON

wennqc@gmail.com: Deployed 77dec285
Mar 26 at 5:23 PM · v15

wennqc@gmail.com: Build succeeded
Mar 26 at 5:23 PM · View build log

wennqc@gmail.com: Deployed 1e8e4a76
Mar 26 at 4:14 PM · v14

Figure 6.11: Heroku app and Heroku Postgres add-on

6.6 Conclusion

This chapter describes the implementation phase of the project. React Navigation and React Native Paper are the 2 third-party libraries utilized by the implemented system to make use of the navigation functionalities and the UI elements. Besides, the API endpoints used by the implemented system were compiled as a list. At the end of the development phase, the Flask REST API and PostgreSQL database were deployed on Heroku to make the REST API and PostgreSQL database accessible by the implemented system.

CHAPTER 7

TESTING AND EVALUATION

7.1 Introduction

This chapter summarizes the testing activities of the implemented system and the evaluation of the testing results. This project's testing activities included web service testing, usability testing, and user acceptance testing. The traceability matrix between use cases, functional requirements and test cases were produced to monitor of the conformance of the test cases with the use cases and functional requirements written in Chapter 4.

7.2 Web Service Testing

Web service testing was conducted in this project to test out the functionality and reliability of the web service APIs used by the implemented system. It acted as unit testing and integration testing to ensure that there was no major errors or failures occurred when both the internal and external APIs were requested for the responses by the implemented system. Web service APIs are used intensively in the implemented system to supply the important functionalities. Table 7.1 indicates the summary of test cases that were used to test out the internal and external APIs' endpoints with the status of test cases.

Table 7.1: Summary of test cases of web service testing and the status

Test Case ID	Test Case Title	Status
TC001	Test Case of User Account Creation	Pass
TC002	Test Case of Successful User Login	Pass
TC003	Test Case of Failed User Login	Pass
TC004	Test Case of Searching Meal	Pass
TC005	Test Case of Adding First Meal Record of the Day	Pass
TC006	Test Case of Adding Another Meal Record of the Day	Pass
TC007	Test Case of Uploading Meal Photo	Pass

TC008	Test Case of Deleting Meal Record	Pass
TC009	Test Case of Searching Exercise	Pass
TC010	Test Case of Adding First Exercise Record of the Day	Pass
TC011	Test Case of Adding Another Exercise Record of the Day	Pass
TC012	Test Case of Editing Exercise Record	Pass
TC013	Test Case of Deleting Exercise Record	Pass
TC014	Test Case of Searching Recipe	Pass
TC015	Test Case of Viewing Recipe Details	Pass
TC016	Test Case of Adding Favorite Recipe	Pass
TC017	Test Case of Deleting Favorite Recipe	Pass
TC018	Test Case of Displaying A List of Articles	Pass
TC019	Test Case of Viewing Article Contents	Pass
TC020	Test Case of Adding Favorite Article	Pass
TC021	Test Case of Deleting Favorite Article	Pass
TC022	Test Case of Editing Date of Birth	Pass
TC023	Test Case of Editing Weight	Pass
TC024	Test Case of Editing Height	Pass
TC025	Test Case of Editing Goal	Pass
TC026	Test Case of Editing Activity Level	Pass
TC027	Test Case of Viewing Favorite Recipes	Pass
TC028	Test Case of Viewing Favorite Recipes from Empty List	Pass
TC029	Test Case of Viewing Favorite Articles	Pass
TC030	Test Case of Viewing Favorite Articles from Empty List	Pass

Table 7.2: Test Case of User Account Creation

Test Case ID	TC001	Status	Pass
Test Case Title	Test Case of User Account Creation		
Test Case Description	To verify the sign up functionality with unique email address and daily macronutrients limit based on the user’s information		
Endpoints Involved	“https://fitness-calculator.p.rapidapi.com/macrocalculator? age={age}&gender={gender}&height={height}&weight={weight} &activitylevel={activityLvNum}&goal={goalScale}”, “/api/users/{email}”, “/api/users”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/ macrocalculator?age={age}&gender={gender} &height={height}&weight={weight} &activitylevel={activityLvNum} &goal={goalScale}”	1. User’s age 2. User’s gender 3. User’s height 4. User’s weight 5. User’s activity level 6. User’s goal	1. A JSON object with the daily net calories and macronutrients limit	The user account is created successfully with the daily net calories and macronutrients limit retrieved based on the user’s information. The user account’s information is

2. The user retrieves the account information from the endpoint "/api/users/{email}" to check whether the email address was stored in the database already.	1. Email address	1. A null object with status code 200	stored in the database table with no duplication of email address.
3. The user sends the account information to the endpoint "/api/users"	1. Account information	1. A JSON object with user's email address and the successful value 1 with status code 201	

Table 7.3: Test Case of Successful User Login

Test Case ID	TC002	Status	Pass
Test Case Title	Test Case of Successful User Login		
Test Case Description	To verify the successful user login status if the user account information exists in the database table		
Endpoints Involved	“/api/users/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the account information from the endpoint “/api/users/{email}” to check the availability of user account information in the database table	1. Email address that has been used to create a user account before	1. A user object in JSON format with status code 200	The user account information is retrieved successfully.

Table 7.4: Test Case of Failed User Login

Test Case ID	TC003	Status	Pass
Test Case Title	Test Case of Failed User Login		
Test Case Description	To verify the failed user login status if the user account information does not exist in the database table		
Endpoints Involved	“/api/users/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the account information from the endpoint “/api/users/{email}” to check the availability of user account information in the database table	1. Email address that is not used to create a user account before	1. A null object with status code 200	There is no matched user account information being retrieved.

Table 7.5: Test Case of Searching Meal

Test Case ID	TC004	Status	Pass
Test Case Title	Test Case of Searching Meal		
Test Case Description	To verify the search meal functionality		
Endpoints Involved	“https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/food/menuItems/search?query={searchQuery}&offset=0&number=50&minCalories=0&maxCalories=5000&minProtein=0&minFat=0&minCarbs=0”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves a list of meals from the endpoint “https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/food/menuItems/search?query={searchQuery}&offset=0&number=50&minCalories=0&maxCalories=5000&minProtein=0&minFat=0&minCarbs=0” to search a list of matched meals according to the keyword entered	1. Meal keyword	1. An array of meal objects and other information of the response in JSON format	A list of meal objects are retrieved if there are matched meals information found

Table 7.6: Test Case of Adding First Meal Record of the Day

Test Case ID	TC005	Status	Pass
Test Case Title	Test Case of Adding First Meal Record of the Day		
Test Case Description	To verify the add meal record functionality. The first meal record creation of the day will create a meal record and a record of daily intake information.		
Endpoints Involved	“/api/meals”, “/api/intake/{email}/{record_date}”, “/api/intake”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the meal record information to the endpoint “/api/meals” to add the meal record.	1. Meal record information	1. A successful value 1 with status code 201	The meal record is successfully added into the database table and a record of daily intake information has been created
2. The user retrieves the daily intake information from the endpoint “/api/intake/{email}/{record_date}”	1. User’s email address and the record date	1. A null object with status code 200	
3. The user sends the meal intake information to the endpoint “/api/intake” to create a record of daily intake information	1. Meal intake information	1. A JSON object with user’s email address and the successful value 1 with status code 201	

Table 7.7: Test Case of Adding Another Meal Record of the Day

Test Case ID	TC006	Status	Pass
Test Case Title	Test Case of Adding Another Meal Record of the Day		
Test Case Description	To verify the add meal record functionality. The meal record creation will create a meal record and update the record of daily intake information.		
Endpoints Involved	“/api/meals”, “/api/intake/{email}/{record_date}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the meal record information to the endpoint “/api/meals” to add the meal record.	1. Meal record information	1. A successful value 1 with status code 201	The meal record is successfully added into the database table and the daily intake information has been updated accordingly
2. The user retrieves the daily intake information from the endpoint “/api/intake/{email}/{record_date}”	1. User’s email address and the record date	1. A daily intake object in JSON format with status code 200	
3. The user sends the meal intake information to the endpoint “/api/intake/{email}/{record_date}” to update the record of daily intake information	1. User’s email address and the record date 2. Meal intake information	1. A JSON object with user’s email address, record date and the successful value 1 with status code 200	

Table 7.8: Test Case of Uploading Meal Photo

Test Case ID	TC007	Status	Pass
Test Case Title	Test Case of Uploading Meal Photo		
Test Case Description	To verify the upload meal photo functionality. The meal record will be updated with the meal photo saved in the meal record		
Endpoints Involved	“/api/meals/{meal_id}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific meal record from the endpoint “/api/meals/{meal_id}”	1. Meal ID	1. A meal record object in JSON format with status code 200	The meal record is retrieved from the database table and updated with the meal photo URL
2. The user sends the meal photo URL to the endpoint “/api/meals/{meal_id}” to update the meal record	1. Meal ID 2. Meal photo URL	1. A JSON object with meal ID and the successful value 1 with status code 200	

Table 7.9: Test Case of Deleting Meal Record

Test Case ID	TC008	Status	Pass
Test Case Title	Test Case of Deleting Meal Record		
Test Case Description	To verify the delete meal record functionality		
Endpoints Involved	“/api/meals/{meal_id}”, “/api/intake/{email}/{record_date}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the meal ID to the endpoint “/api/meals/{meal_id}” to delete the meal record	1. Meal ID	1. A JSON object with meal ID and the successful value 1 with status code 200	The meal record is deleted from the database table and the daily intake information has been updated accordingly
2. The user retrieves the daily intake information from the endpoint “/api/intake/{email}/{record_date}”	1. User’s email address and the record date	1. A daily intake object in JSON format with status code 200	
3. The user sends the meal intake information to the endpoint “/api/intake/{email}/{record_date}” to update the record of daily intake information	1. User’s email address and the record date 2. Meal intake information	1. A JSON object with user’s email address, record date and the successful value 1 with status code 200	

Table 7.10: Test Case of Searching Exercise

Test Case ID	TC009	Status	Pass
Test Case Title	Test Case of Searching Exercise		
Test Case Description	To verify the search exercise functionality. A list of matched exercises will be retrieved.		
Endpoints Involved	“/api/load_exercises/{matchText}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends a keyword to the endpoint “/api/load_exercises/{matchText}” to search for the exercises	1. Exercise keyword	1. An array of exercise objects in JSON format with status code 200	A list of matched exercises will be retrieved from the database table

Table 7.11: Test Case of Adding First Exercise Record of the Day

Test Case ID	TC010	Status	Pass
Test Case Title	Test Case of Adding First Exercise Record of the Day		
Test Case Description	To verify the add exercise record functionality. The first exercise record creation of the day will create an exercise record and a record of daily net calories information.		
Endpoints Involved	"/api/exercises", "/api/intake/{email}/{record_date}", "/api/intake"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the exercise record information to the endpoint "/api/exercises" to add the exercise record.	1. Exercise record information	1. A successful value 1 with status code 201	The exercise record is successfully added into the database table and a record of daily net calories information has been created
2. The user retrieves the daily net calories information from the endpoint "/api/intake/{email}/{record_date}"	1. User's email address and the record date	1. An empty JSON object with status code 200	
3. The user sends the calories information to the endpoint "/api/intake" to create a record of daily net calories information	1. Calories information	1. A JSON object with user's email address and the successful value 1 with status code 201	

Table 7.12: Test Case of Adding Another Exercise Record of the Day

Test Case ID	TC011	Status	Pass
Test Case Title	Test Case of Adding Another Exercise Record of the Day		
Test Case Description	To verify the add exercise record functionality. The exercise record creation will create an exercise record and update the record of daily net calories information.		
Endpoints Involved	“/api/exercises”, “/api/intake/{email}/{record_date}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the exercise record information to the endpoint “/api/exercises” to add the exercise record.	1. Exercise record information	1. A successful value 1 with status code 201	The exercise record is successfully added into the database table and the daily net calories information has been updated accordingly
2. The user retrieves the daily net calories information from the endpoint “/api/intake/{email}/{record_date}”	1. User’s email address and the record date	1. A daily intake object in JSON format with status code 200	
3. The user sends the calories information to the endpoint “/api/intake/{email}/{record_date}” to update the record of daily net calories information	1. User’s email address and the record date 2. Calories information	1. A JSON object with user’s email address, record date and the successful value 1 with status code 200	

Table 7.13: Test Case of Editing Exercise Record

Test Case ID	TC012	Status	Pass
Test Case Title	Test Case of Editing Exercise Record		
Test Case Description	To verify the edit exercise record functionality		
Endpoints Involved	"/api/exercises/{exercise_id}", "/api/intake/{email}/{record_date}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific exercise record from the endpoint "/api/exercises/{exercise_id}"	1. Exercise ID	1. An exercise record object in JSON format with status code 200	The exercise record is retrieved from the database table and updated with the new exercise record information. The daily net calories information has been updated accordingly
2. The user sends the exercise record information to the endpoint "/api/exercises/{exercise_id}" to update the exercise record	1. Exercise ID 2. Exercie record information	1. A JSON object with exercise ID and the successful value 1 with status code 200	
3. The user retrieves the daily net calories information from the endpoint "/api/intake/{email}/{record_date}"	1. User's email address and the record date	1. A daily intake object in JSON format with status code 200	
4. The user sends the calories information to the endpoint	1. User's email address and the record date	1. A JSON object with user's email address, record date and	

"/api/intake/{email}/{record_date}" to update the record of daily net calories information	2. Calories information	the successful value 1 with status code 200	

Table 7.14: Test Case of Deleting Exercise Record

Test Case ID	TC013	Status	Pass
Test Case Title	Test Case of Deleting Exercise Record		
Test Case Description	To verify the delete exercise record functionality		
Endpoints Involved	"/api/exercises/{exercise_id}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the exercise ID to the endpoint "/api/exercises/{exercise_id}" to delete the exercise record	1. Exercise ID	1. A JSON object with exercise ID and the successful value 1 with status code 200	The exercise record is deleted from the database table and the daily net calories information has been updated accordingly
2. The user retrieves the daily net calories information from the endpoint "/api/intake/{email}/{record_date}"	1. User's email address and the record date	1. A daily intake object in JSON format with status code 200	
3. The user sends the calories information to the endpoint	1. User's email address and the record date 2. Calories information	1. A JSON object with user's email address, record date and	

“/api/intake/{email}/{record_date}” to update the record of daily net calories information		the successful value 1 with status code 200	

Table 7.15: Test Case of Searching Recipe

Test Case ID	TC014	Status	Pass
Test Case Title	Test Case of Searching Recipe		
Test Case Description	To verify the search recipe functionality. A list of matched recipes will be retrieved.		
Endpoints Involved	“https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/search?query=food&instructionsRequired=true&diet={searchQuery}&number=10&offset=0&type=main%20course”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves a list of recipes from the endpoint “https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/search?query=food&instructionsRequired=true&diet={searchQuery}&number=10&offset=0&type=main%20cou	1. Recipe keyword	1. An array of recipe objects and other information of the response in JSON format	A list of recipe objects are retrieved if there are matched recipes information found

rse” to search a list of matched recipes according to the keyword entered			

Table 7.16: Test Case of Viewing Recipe Details

Test Case ID	TC015	Status	Pass
Test Case Title	Test Case of Viewing Recipe Details		
Test Case Description	To verify the view recipe details functionality. The recipe details will be retrieved based on the recipe ID entered.		
Endpoints Involved	“https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/informationBulk?ids={recipeID}”, “https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/{recipeID}/nutritionWidget.json”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific recipe details from the endpoint “https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/informationBulk?ids={recipeID}”	1. Recipe ID	1. An array of recipe object in JSON format	The specific recipe details and nutrition information

2. The user retrieves the nutrition information of the specific recipe from the endpoint “https://spoonacular-recipe-food-nutrition-v1.p.rapidapi.com/recipes/{recipeID}/nutritionWidget.json”	1. Recipe ID	1. A JSON object with nutrition information	are retrieved based on the recipe ID entered

Table 7.17: Test Case of Adding Favorite Recipe

Test Case ID	TC016	Status	Pass
Test Case Title	Test Case of Adding Favorite Recipe		
Test Case Description	To verify the add favorite recipe functionality. The add favorite recipe can be performd only if the user’s favorite recipes list does not contain the specific recipe record.		
Endpoints Involved	“/api/fav_recipes/{user_email}/{recipe_index_no}”, “/api/fav_recipes”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific favorite recipe from the endpoint “/api/fav_recipes/{user_email}/{recipe_index_no}”	1. User’s email address and recipe index number	1. A null object with status code 200	The specific recipe is saved as the favorite recipe only if the user’s favorite recipes list does not contain the specific recipe record
2. The user sends the recipe information to the endpoint “/api/fav_recipes” to add the recipe into his/her favorite recipes list	1. Recipe information	1. A JSON object with the successful value 1 with status code 201	

Table 7.18: Test Case of Deleting Favorite Recipe

Test Case ID	TC017	Status	Pass
Test Case Title	Test Case of Deleting Favorite Recipe		
Test Case Description	To verify the delete favorite recipe functionality. The delete favorite recipe can be performd only if the user’s favorite recipes list contains the specific recipe record.		
Endpoints Involved	“/api/fav_recipes/{user_email}/{recipe_index_no}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific favorite recipe from the endpoint “/api/fav_recipes/{user_email}/{recipe_index_no}”	1. User’s email address and recipe index number	1. A favorite recipe object in JSON format with status code 200	The specific recipe is deleted from the favorite recipes list only if the user’s favorite recipes list contains the specific recipe record
2. The user sends the recipe index number to the endpoint “/api/fav_recipes/{user_email}/{recipe_index_no}” to delete the recipe record from his/her favorite recipes list	1. User’s email address and recipe index number	1. A JSON object with favorite recipe ID and the successful value 1 with status code 200	

Table 7.19: Test Case of Displaying A List of Articles

Test Case ID	TC018	Status	Pass
Test Case Title	Test Case of Displaying A List of Articles		
Test Case Description	To verify the display a list of articles functionality. A list of articles will be retrieved.		
Endpoints Involved	"/api/article_source", "/api/article_source/{article_tag}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves a complete list of articles from the endpoint "/api/article_source"	-	1. An array of article objects with status code 200	A complete list of article objects are retrieved if there is no keyword inputted, otherwise a specific category of article objects will be retrieved based to the keyword inputted
2. The user retrieves a list of articles of specific category from the endpoints "/api/article_source/{article_tag}"	1. Category keyword	1. An array of article objects with status code 200	

Table 7.20: Test Case of Viewing Article Contents

Test Case ID	TC019	Status	Pass
Test Case Title	Test Case of Viewing Article Contents		
Test Case Description	To verify the view article contents functionality. The article contents will be retrieved based on the recipe ID entered.		
Endpoints Involved	“https://lexper.p.rapidapi.com/v1.1/extract?url={article_url}&media=1”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific article contents from the endpoint “https://lexper.p.rapidapi.com/v1.1/extract?url={article_url}&media=1”	1. Article URL	1. An array of article object with the response information in JSON format	The specific article contents are retrieved based on the article URL entered

Table 7.21: Test Case of Adding Favorite Article

Test Case ID	TC020	Status	Pass
Test Case Title	Test Case of Adding Favorite Article		
Test Case Description	To verify the add favorite article functionality. The add favorite article can be performd only if the user's favorite articles list does not contain the specific article record.		
Endpoints Involved	"/api/fav_articles/{user_email}/{article_title}", "/api/fav_articles"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific favorite article from the endpoint "/api/fav_articles/{user_email}/{article_title}"	1. User's email address and article title	1. A null object with status code 200	The specific article is saved as the favorite article only if the user's favorite articles list does not contain the specific article record
2. The user sends the article information to the endpoint "/api/fav_articles" to add the article into his/her favorite articles list	1. Article information	1. A JSON object with the successful value 1 with status code 201	

Table 7.22: Test Case of Deleting Favorite Article

Test Case ID	TC021	Status	Pass
Test Case Title	Test Case of Deleting Favorite Article		
Test Case Description	To verify the delete favorite article functionality. The delete favorite article can be performd only if the user's favorite articles list contains the specific article record.		
Endpoints Involved	"/api/fav_articles/{user_email}/{article_title}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves the specific favorite article from the endpoint "/api/fav_articles/{user_email}/{article_title}"	1. User's email address and article title	1. A favorite article object in JSON format with status code 200	The specific article is deleted from the favorite articles list only if the user's favorite articles list contains the specific article record
2. The user sends the article title to the endpoint "/api/fav_articles/{user_email}/{article_title}" to delete the article record from his/her favorite articles list	1. User's email address and article title	1. A JSON object with favorite article ID and the successful value 1 with status code 200	

Table 7.23: Test Case of Editing Date of Birth

Test Case ID	TC022	Status	Pass
Test Case Title	Test Case of Editing Date of Birth		
Test Case Description	To verify the edit personal information (date of birth) functionality. The updated daily macronutrients limit will be retrieved and updated as well.		
Endpoints Involved	“/api/user_age/{email}”, “https://fitness-calculator.p.rapidapi.com/macrocalculator?age={updatedAge}&gender={gender}&height={height}&weight={weight}&activitylevel={activityLvNum}&goal={goalScale}”, “/api/user_daily_nutrition/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the updated date of birth to the endpoint “/api/user_age/{email}”	1. User’s email address 2. Updated date of birth	1. A JSON object with user’s email address and the successful value 1 with status code 200	The user’s date of birth and daily macronutrients limit are updated
2. The user retrieves the updated daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/	1. Updated user’s age 2. User’s gender 3. User’s height	1. A JSON object with the daily net calories and macronutrients limit	

macrocalculator?age={updatedAge}&gender={gender}&height={height}&weight={weight}&activitylevel={activityLvNum}&goal={goalScale}”	4. User’s weight 5. User’s activity level 6. User’s goal		
3. The user sends the updated daily net calories and macronutrients limit to the endpoint “/api/user_daily_nutrition/{email}” to update his/her daily macronutrients limit	1. User’s email address 2. Updated daily macronutrients limit	1. A JSON object with user’s email address and the successful value 1 with status code 200	

Table 7.24: Test Case of Editing Weight

Test Case ID	TC023	Status	Pass
Test Case Title	Test Case of Editing Weight		
Test Case Description	To verify the edit personal information (weight) functionality. The updated daily macronutrients limit will be retrieved and updated as well.		
Endpoints Involved	“/api/user_weight/{email}”, “https://fitness-calculator.p.rapidapi.com/macrocalculator?age={age}&gender={gender}&height={height}&weight={updatedWeight}&activitylevel={activityLvNum}&goal={goalScale}”,		

	“/api/user_daily_nutrition/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the updated weight and BMI to the endpoint “/api/user_weight/{email}”	1. User’s email address 2. Updated weight 3. Updated BMI	1. A JSON object with user’s email address and the successful value 1 with status code 200	The user’s weight, BMI and daily macronutrients limit are updated
2. The user retrieves the updated daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/ macrocalculator?age={age}&gender={gender} &height={height}&weight={updatedWeight} &activitylevel={activityLvNum} &goal={goalScale}”	1. User’s age 2. User’s gender 3. User’s height 4. Updated user’s weight 5. User’s activity level 6. User’s goal	1. A JSON object with the daily net calories and macronutrients limit	
3. The user sends the updated daily net calories and macronutrients limit to the endpoint “/api/user_daily_nutrition/{email}” to update his/her daily macronutrients limit	1. User’s email address 2. Updated daily macronutrients limit	1. A JSON object with user’s email address and the successful value 1 with status code 200	

Table 7.25: Test Case of Editing Height

Test Case ID	TC024	Status	Pass
Test Case Title	Test Case of Editing Height		
Test Case Description	To verify the edit personal information (height) functionality. The updated daily macronutrients limit will be retrieved and updated as well.		
Endpoints Involved	“/api/user_height/{email}”, “https://fitness-calculator.p.rapidapi.com/macrocalculator?age={age}&gender={gender}&height={updatedHeight}&weight={weight}&activitylevel={activityLvNum}&goal={goalScale}”, “/api/user_daily_nutrition/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the updated height and BMI to the endpoint “/api/user_height/{email}”	1. User’s email address 2. Updated height 3. Updated BMI	1. A JSON object with user’s email address and the successful value 1 with status code 200	The user’s height, BMI and daily macronutrients limit are updated
2. The user retrieves the updated daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/	1. User’s age 2. User’s gender 3. Updated user’s height	1. A JSON object with the daily net calories and macronutrients limit	

macrocalculator?age={age}&gender={gender}
&height={updatedHeight}&weight={weight}
&activitylevel={activityLvNum}
&goal={goalScale}”
	4. User’s weight
	5. User’s activity level
	6. User’s goal		

3. The user sends the updated daily net calories and macronutrients limit to the endpoint “/api/user_daily_nutrition/{email}” to update his/her daily macronutrients limit	1. User’s email address 2. Updated daily macronutrients limit	1. A JSON object with user’s email address and the successful value 1 with status code 200	

Table 7.26: Test Case of Editing Goal

Test Case ID	TC025	Status	Pass
Test Case Title	Test Case of Editing Goal		
Test Case Description	To verify the edit personal information (goal) functionality. The updated daily macronutrients limit will be retrieved and updated as well.		
Endpoints Involved	“/api/user_goal/{email}”, “https://fitness-calculator.p.rapidapi.com/macrocalculator? age={age}&gender={gender}&height={height}&weight={weight} &activitylevel={activityLvNum}&goal={updatedGoalScale}”,		

	“/api/user_daily_nutrition/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the updated goal to the endpoint “/api/user_goal/{email}”	1. User’s email address 2. Updated goal	1. A JSON object with user’s email address and the successful value 1 with status code 200	The user’s goal and daily macronutrients limit are updated
2. The user retrieves the updated daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/macrocalculator?age={age}&gender={gender}&height={height}&weight={weight}&activitylevel={activityLvNum}&goal={updatedGoalScale}”	1. User’s age 2. User’s gender 3. User’s height 4. User’s weight 5. User’s activity level 6. Updated user’s goal	1. A JSON object with the daily net calories and macronutrients limit	
3. The user sends the updated daily net calories and macronutrients limit to the endpoint “/api/user_daily_nutrition/{email}” to update his/her daily macronutrients limit	1. User’s email address 2. Updated daily macronutrients limit	1. A JSON object with user’s email address and the successful value 1 with status code 200	

Table 7.27: Test Case of Editing Activity Level

Test Case ID	TC026	Status	Pass
Test Case Title	Test Case of Editing Activity Level		
Test Case Description	To verify the edit personal information (activity level) functionality. The updated daily macronutrients limit will be retrieved and updated as well.		
Endpoints Involved	“/api/user_activity_lv/{email}”, “https://fitness-calculator.p.rapidapi.com/macrocalculator?age={age}&gender={gender}&height={height}&weight={weight}&activitylevel={updatedActivityLvNum}&goal={goalScale}”, “/api/user_daily_nutrition/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user sends the updated activity level to the endpoint “/api/user_activity_lv/{email}”	1. User’s email address 2. Updated activity level	1. A JSON object with user’s email address and the successful value 1 with status code 200	The user’s activity level and daily macronutrients limit are updated
2. The user retrieves the updated daily macronutrients limit from the endpoint “https://fitness-calculator.p.rapidapi.com/	1. User’s age 2. User’s gender 3. User’s height	1. A JSON object with the daily net calories and macronutrients limit	

macrocalculator?age={age}&gender={gender}
&height={height}&weight={weight}
&activitylevel={updatedActivityLvNum}
&goal={goalScale}”
	4. User’s weight
	5. Updated user’s activity level
	6. User’s goal		

3. The user sends the updated daily net calories and macronutrients limit to the endpoint “/api/user_daily_nutrition/{email}” to update his/her daily macronutrients limit	1. User’s email address 2. Updated daily macronutrients limit	1. A JSON object with user’s email address and the successful value 1 with status code 200	

Table 7.28: Test Case of Viewing Favorite Recipes

Test Case ID	TC027	Status	Pass
Test Case Title	Test Case of Viewing Favorite Recipes		
Test Case Description	To verify the view favorite recipes functionality		
Endpoints Involved	“/api/fav_recipes/{email}”		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves his/her favorite recipes from the endpoint “/api/fav_recipes/{email}”	1. User’s email address	1. An array of favorite recipe objects in JSON format with status code 200	All the user’s favorite recipes are retrieved

Table 7.29: Test Case of Viewing Favorite Recipes from Empty List

Test Case ID	TC028	Status	Pass
Test Case Title	Test Case of Viewing Favorite Recipes from Empty List		
Test Case Description	To verify the view favorite recipes functionality if the list of favorite recipes is empty		
Endpoints Involved	"/api/fav_recipes/{email}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves his/her favorite recipes from the endpoint "/api/fav_recipes/{email}"	1. User's email address	1. An empty array with status code 200	None of the user's favorite recipes are successfully retrieved as there is no recipe saved by the user

Table 7.30: Test Case of Viewing Favorite Articles

Test Case ID	TC029	Status	Pass
Test Case Title	Test Case of Viewing Favorite Articles		
Test Case Description	To verify the view favorite articles functionality		
Endpoints Involved	"/api/fav_articles/{email}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves his/her favorite articles from the endpoint "/api/fav_articles/{email}"	1. User's email address	1. An array of favorite article objects in JSON format with status code 200	All the user's favorite articles are retrieved

Table 7.31: Test Case of Viewing Favorite Articles from Empty List

Test Case ID	TC030	Status	Pass
Test Case Title	Test Case of Viewing Favorite Articles from Empty List		
Test Case Description	To verify the view favorite articles functionality if the list of favorite articles is empty		
Endpoints Involved	"/api/fav_articles/{email}"		
Test Steps	Test Data	Expected Result	Actual Result
1. The user retrieves his/her favorite articles from the endpoint "/api/fav_articles/{email}"	1. User's email address	1. An empty array with status code 200	None of the user's favorite articles are successfully retrieved as there is no article saved by the user

7.3 Usability Testing

Usability testing was conducted via online after the web service testing to evaluate how easy to use the implemented system is. It is important to have usability testing being done before releasing the software to public to ensure the target users of the implemented system are able to perform the basic operations using the implemented system without any major doubts. The final outcomes of usability testing were the users' feedbacks regarding the implemented system and also the calculated average SUS score.

There are few rules that need to follow when conducting usability testing. Before beginning the usability testing, participants are expected to read through the test scenarios. They are required to complete the test scenarios under the observation of the developer. If there are any doubt faced when completing the test scenarios, they can always ask for help from the developer. The developer will observe and monitor the whole usability testing process and answer the inquiries raised by the participants. After the usability testing, the participants need to fill in the user satisfaction survey form.

In this project, five target users, who are the university students were invited to participate in the online usability testing. They were required to perform the usability testing by following the prepared test scenarios as shown in Table 7.32. After performing the usability testing, five of them were asked to complete the user satisfaction survey form as attached in Appendix F and the survey's results as attached in Appendix G were analyzed and used to calculate the average SUS score of the implemented system.

7.3.1 Test Scenarios of Usability Testing

Table 7.32: Test Scenarios of Usability Testing

No	Test Scenario Title	Test Scenario Description
1	Register an account	Imagine you are the user who wants to track your lifestyle such as meal and exercise records and also wish to browse through a list of recipes and health-related articles by using a mobile application. Before you can use the mobile application, you need to register an account.

		After having an account, you can track your meal and exercise records and also browse through the recipes and articles. Hence, what would you do to register an account?
2	Add a meal record into meal log	Imagine you want to add your daily meal record into your meal log. After adding meal record, you get to view your daily macronutrients intake. What would you do to add the meal record? Where can you view the changes of your daily macronutrients intake?
3	Edit the meal record	Imagine you want to upload a food photo for the specific meal record. What would you do to upload and save the food photo? After that, you wish to delete the meal record from your meal log because you have recorded the wrong meal. What would you do to delete the meal record?
4	Add an exercise record into exercise log	Imagine you want to add your daily exercise record into your exercise log. After adding exercise record, you get to view your daily net calories information. What would you do to add the exercise record? Where can you view the changes of your daily net calories information?
5	Edit the exercise record	Imagine you wish to update the duration of exercise performed because you have recorded the wrong duration. What would you do to update and save the duration of exercise performed?

		After that, you wish to delete the exercise record because you have recorded the wrong exercise activity. What would you do to delete the exercise record?
6	Search and view the recipe	Imagine you want to search for the desired recipe and view the recipe details to learn the steps of preparing the food. What would you do to view the recipe details?
7	Save recipe and view the favorite recipes	Imagine when you viewing the recipe details, you want to save the recipe for future reference. What would you do to save the recipe? After that, you wish to check which recipes you have saved from the day you started to use this mobile application. What would you do to view your favorite recipes list?
8	View article	Imagine you want to view the specific article contents to learn about the health care tips. After that, you feel that the article is useful and informative, and you would like to share to your social media. What would you do to view and share the article?
9	Save article and view the favorite article	Imagine when you viewing the article contents, you want to save the article for future reference. What would you do to save the article? After that, you wish to check which articles you have saved from the day you started to use this mobile application. What would you do to view your favorite articles list?
10	Update personal profile information	Personal profile information is used to calculate the exact daily macronutrients limit for each user. Imagine you want to update your personal profile

		information such as date of birth, weight, height, goal of using this mobile application and current activity level in order to update your daily macronutrients limit. What would you do to update your personal profile information?

7.3.2 Results of Usability Testing

After conducting the usability testing, the user satisfaction survey forms were received from all the participants as shown in Appendix G. The results from each user satisfaction survey form were extracted and displayed in Table 7.33. The average SUS score obtained by this project were 94.5, which was higher than the target SUS score set in the project objectives section in Chapter 1 (75 out of 100). This indicates that the implemented system can provide a good user experience to the users. The users can use and learn the implemented system without having to spend much time to do exploration and getting help from a technical person. However, the average SUS score (68 out of 100) calculated from the studies done by Lewis (2018) and Sasmito, Zulfiqar and Nishom (2019) using both CGS and SUS adjacent ratings involved a huge amount of participants in the usability testing. The details of the SUS benchmark score can be found in Section 2.4 in Chapter 2. As for this project, only 5 participants were involved in the usability testing, which indicates the limitation of the SUS score obtained. Hence, more participants are needed to involve in the usability testing of this project in the future to obtain the more accurate and reliable SUS score.

Table 7.33: Summary of User Satisfaction Survey Results

Participant	**Score for each item**										**Total**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
Participant #1	5	1	5	1	5	1	5	1	5	1	100
Participant #2	4	1	5	1	5	1	5	1	5	1	97.5

Participant #3	5	1	5	1	5	1	5	1	5	1	100
Participant #4	4	2	4	2	4	1	5	1	4	1	85
Participant #5	4	2	5	1	5	2	4	1	5	1	90
Average SUS score											94.5

Moreover, the user satisfaction survey forms did collect the information regarding the users' feelings and thoughts towards the implemented system. The features and functionalities they like the most and least were gathered and tabulated in Table 7.34. It could be summarized that the participants were satisfied with most of the features provided by the implemented system. There were some parts they liked the least which were considered minor issues and those parts would be improved in the future.

Table 7.34: Listing of features and functionalities that were most liked and least liked by the participants

Most Liked Features and Functionalities	**Least Liked Features and Functionalities**
Calories intake is recorded for each meal record added	Appearance of Log Out button is not obvious in the Profile Home Screen
Upload food photo feature allows to save a food photo for the specific meal record	Recipes provided in the Recipes module are of less variety
Able to view a variety of recipes and articles and also the daily calories information	Loading time required for the article contents part is long, which is about 2 to 3 seconds
Save favorite articles feature to save the specific article for future reference	

Apart from that, the user satisfaction survey form did ask for the participants' sincere reviews and suggestions about the implemented system. The reviews and suggestions given by the participants were summarized as shown in Table 7.35. The suggestions received from the particapants would take into consideration as parts of the future improvement of the implemented system.

Table 7.35: Summary of reviews and suggestions given by the participants

Reviews	**Suggestions**
The implemented system is useful for people to track their diet. The simple user interface is suitable to use even for the elders.	Wish to have a more interactive UI design in order to motivate users to use the implemented system.
The implemented system covers a lot of features that the lifestyle tracker mobile application should have.	
The implemented system has a minimalist design. It is a simple and sufficient system for users to track the calories and BMI.	It would be better if the implemented system is able to suggest the articles and recipes based on the user's current BMI.
Huge variety of food recipes and articles provided by the implemented system.	

7.4 User Acceptance Testing

After conducting the web service and usability testing, user acceptance testing (UAT) was conducted via online to make sure the implemented system is accepted by the target users and to validate the operational flow of the implemented system. Hence, in this project, there were five target users being invited to perform the UAT. The participants were required to perform the UAT according to the prepared test form as attached in Appendix H. During the process of UAT, the developer acted as the observer of the testing and provided assistance to the participants when there was any question being raised.

Moreover, the participants had to mark down the starting and ending time of each test case conducted and logged the issue faced or comments if any.

According to the results of UAT attached in Appendix I, there were two failed test cases due to the different path of test steps chosen by the participants. There was also an error discovered by one of the participants when she explored the specific test case. The related test case information with the error occurred and action taken was recorded in Table 7.36.

Table 7.36: Summary of error occurred and the action taken

Test Case ID	Test Case Title	Status	Error Occurred	Action Taken
UAT003	Upload food photo for the meal record	Fail	The specific food photo selected from the library was not shown in the meal record screen	After testing the upload food photo feature by selecting some of the food photos to perform upload, it was found that only the food photos with JPG format (.jpg) were able to upload and show in the meal record.
UAT008	Search and view recipe	Fail	When clicking on one of the recipe in the “Malaysian cuisine” category, the application does not respond and does not show the recipe information	After inspecting the source code and testing the related API endpoint by sending the respective request, there was no issue on the application logic and the returned response from the API endpoints. Hence, the action taken was to remove the respective recipe that could not return the recipe information in details.
UAT011	Save article	Pass	Although the participant that found out this error had passed this test case according to the test steps given, but she discovered this error when she wished to unsave the article from the article contents screen when the previous screen was the favorite article list	The save article action does save the title with question mark into the database table correctly, but it could not retrieve the title with question mark properly. Hence, the action taken was to remove the specific article source record which its title contains a question mark.

According to Table 7.36, it has shown that some temporary solutions were provided to solve the issues faced by the implemented system. Future improvement would be performed to enhance the implemented system to accept the article source with any special characters in the title.

Overall, the UAT was considered successful with minimal actions needed to take in order to pass all the UAT test cases and to provide users a good system without major errors occurred.

7.5 Traceability between Use Cases, Functional Requirements and Test Cases

Software testing is undeniably an important phase in SDLC. Without testing, the developer team would not detect the potential and existing bugs and there will not have any preventive and corrective actions to avoid and solve the root cause of the bug. There are many types of software testing can be conducted, and the selection of testing activities depends on the nature of the software and which important software modules would the project team want to test. It is difficult and tedious for the project team to keep track of the conformance of test cases with use cases and also functional requirements if the number of test cases is too much. Hence, the traceability matrix was used in this project to show the purpose of the test cases generation as well as the relationship between the test cases, use cases and the functional requirements.

7.5.1 Use Cases Table

Table 7.37 displays the use cases table that includes the use case ID and use case name.

Table 7.37: Use Cases Table

Use Case ID	**Use Case Name**
UC001	Log in account
UC002	Create account
UC003	Update personal information
UC004	Record daily meals
UC005	Record exercise activities

UC006	Upload food photo
UC007	View macronutrients and calories
UC008	View food recipes
UC009	Save favorite recipes
UC010	View health-related articles
UC011	Save favorite articles
UC012	Share article

7.5.2 Functional Requirements Table

Table 7.38 displays the functional requirements table that includes the functional requirement ID and the functional requirement statement.

Table 7.38: Functional Requirements Table

Functional Requirement ID	**Functional Requirement Statement**
FR001	The system shall be able to allow the users to add their daily meals into their food log.
FR002	The system shall be able to calculate the macronutrients (carbohydrates, protein and fat) needed for the users depending on the age, weight, height, goal and the current activity level of the users and to view by the users.
FR003	The system shall be able to allow the users to add their exercise activities into their exercise log.
FR004	The system shall be able to calculate the daily net calories of the users based on their food log and exercise log and to view by the users.
FR005	The system shall be able to provide the users a list of recipes that can be filtered based on the eating preferences and cuisine types.
FR006	The system shall be able to provide the users a list of articles that related to dietary, fitness and mental health care tips.

FR007	The system shall be able to allow the users to save their favorite recipes and articles for future review.
FR008	The system shall be able to allow the users to select and add their favorite recipes as one of the daily meals in their food log.
FR009	The system shall be able to allow the user to update their personal information such as date of birth, weight, height, goal and the current activity level.
FR010	The system shall be able to allow the user to capture a food photo and upload it into their food log.
FR011	The system shall be able to allow the user to share the article to their social media.

7.5.3 Traceability Matrix

Table 7.39 shows the traceability matrix that includes the use case ID, functional requirement ID and test case ID. This traceability matrix was produced to show the relationship between the use cases, functional requirements and test cases from different types of testing performed. Refer to Table 7.37 and 7.38 for the details of use cases and functional requirements.

Table 7.39: Traceability Matrix

Use Case ID	**Functional Requirement ID**	**Web Service Test Case ID**	**UAT Test Case ID**
UC001	-	TC002, TC003	UAT014
UC002	FR002	TC001	UAT001
UC003	FR002, FR009	TC022, TC023, TC024, TC025, TC026	UAT013
UC004	FR001, FR004, FR008	TC004, TC005, TC006, TC008	UAT002, UAT004
UC005	FR003, FR004	TC009, TC010, TC011, TC012, TC013	UAT005, UAT006, UAT007

UC006	FR010	TC007	UAT003
UC007	FR002, FR004	TC005, TC006, TC010, TC011, TC022, TC023, TC024, TC025, TC026	UAT002, UAT005, UAT013
UC008	FR005	TC014, TC015	UAT008
UC009	FR007	TC016, TC017, TC027, TC028	UAT009
UC010	FR006	TC018, TC019	UAT010
UC011	FR007	TC020, TC021, TC029, TC030	UAT011
UC012	FR011	-	UAT012

7.6 Conclusion

This chapter discusses about the testing activities conducted, which were the web service testing, usability testing and UAT. It ensures that the implemented system complies to the functional requirements and non-functional requirements outlined in Chapter 4. By conducting the usability testing and analyzing the results of testing, the user satisfaction towards the implemented system could be defined with the positive comments received and the average SUS score calculated. Not only that, a few errors were found from conducting the UAT and actions were taken to solve the errors. In summary, the implemented system consists of a simple and attractive UI design and is able to function properly as a lifestyle tracker mobile application.

CHAPTER 8

CONCLUSION AND RECOMMENDATION

8.1 Introduction

This project took approximately 10 months to complete, starting from the planning phase to the closing phase. This project is to implement a lifestyle tracker mobile application that helps users to adopt a healthy lifestyle. In the beginning of the project, planning was done to plan for the project solution, approach, scope, gather the requirements through survey and review the similar existing system, generate the requirements and lastly estimate the project schedule.The analysis and design phase was started after the planning phase, which was to perform the analysis of system requirements and necessary design activities. The interface flow diagram and high fidelity prototype design were created after that to illustrate the overall flow and design of the implemented system.

After that, the iterations of design, development and testing were started to implement the project solution module by module. Before starting to develop the lifestyle tracker mobile application, the GitHub repository and database were to set up first. Then, the Sign Up and Login modules were implemented, followed by the Meal Tracker, Exercise Tracker, Macro Calculator, Recipes, Articles and Personal Profile modules. During each iteration, the testing was conducted to ensure the particular module had no major bugs before moving to the next iteration. Finally, before moving to the closing phase, the web service testing was conducted to act as the system integration testing to make sure that all the modules in the implemented system could function as a whole complete system.

Moving to the closing phase of the project, the usability testing and UAT were conducted to evaluate the usability of the implemented system and the user satisfaction and acceptance rate towards the implemented system. The overall SUS score gained for the implemented system was 94.5, which shows that the implemented system can function well and provide the users a better user experience. However, there were few errors found by the participants during UAT, which caused few of the UAT test cases to fail. The errors were fixed after UAT, and future improvements will be made according to the feedback received from both usability testing and UAT to enhance the overall performance and user experience of the implemented system. After that, the system documentation was created to record down all the project activities for future reference.

In the following sections of this chapter, the objectives achievement, project limitations and recommendations for future work are demonstrated.

8.2 Objectives Achievement

The project objectives stated in Chapter 1 were achieved as described below:

The first project objective was to analyze project requirements and review existing similar lifestyle tracker applications. This objective was achieved during the planning phase of the project. The project solution, scope, functional and non-functional requirements were finalized after conducting the survey and reviewing the existing similar lifestyle tracker applications. The survey results helped a lot in determining what are the target users’ expectations towards a lifestyle tracker mobile application, while the review of existing similar applications helped to discover the must-have features of a lifestyle tracker mobile application.

The second project objective was to develop a Lifestyle Tracker mobile application on the Android platform for the purposes of tracking and improving the lifestyle of users. This objective was achieved as well as the Android based lifestyle tracker mobile application was one of the project’s deliverables, followed by the system documentation. The Android based lifestyle tracker mobile application was successfully implemented with the proposed modules stated in Chapter 2, which are the meal tracker, exercise tracker, macro calculator, recipes, articles and personal profile modules and with few unique features such as to save a food photo into the specific meal record, save the favorite recipes and articles and share the articles to social media.

The third project objective was to validate the application’s usability by achieving 75 out of 100 on SUS. This objective was achieved because the implemented system of this project had obtained an average SUS score of 94.5, which was higher than the SUS benchmark score set in the project objective. This indicates that the implemented system could provide the users the useful features that can help the users to keep track of their daily lifestyle and browse through a collection of recipes and health-related articles to learn more about how to prepare a meal and learn the tips related to either dietary, fitness or mental health.

8.3 Project Limitations

Although all the project objectives had been successfully achieved, there are some project limitations discovered during the usability testing and UAT.

The first project limitation is the long loading time required for the article contents, which is about 2 to 3 seconds. This is due to the API endpoint used needs some time to extract, process and return the article contents to the implemented system.

The second project limitation is the implemented system could not accept the article source with any special characters in the article title. The implemented system can save the article with special characters in the article title into the favorite articles database table, but it failed to retrieve the correct article title to do checking about the favorite article.

The third project limitation is the fewer categories of recipes provided to the target users. One of the participants of usability testing is hoping to have more variety of recipes shown in the Recipes home screen, instead of searching the desired recipes using the search bar every time.

The fourth project limitation is the implemented system does not suggest the recipes and articles based on the user’s current BMI. One of the participants from usability testing has suggested this feature to be implemented in the lifestyle tracker mobile application to have a better guide to keep track of his or her lifestyle.

8.4 Recommendations for Future Work

Continuous enhancements and improvements are necessary in order to make the implemented system more robust and complete to provide better user experience and increase the user retention of using it. Therefore, in this section, there are some recommendations for future work being proposed and the future work is not limited to the list as shown in Table 8.1.

Table 8.1: Listing of recommendations for future work

Recommendations	**Description**
Enhance the favorite article retrieval algorithm so that the article title with special characters could be retrieved without error	The implemented system should be able to save and retrieve the favorite articles with any special characters in the title. By having this error fixed, the article source database table can be expanded to accept every article without restricting by the format of article title and the favorite article can successfully retrieve from the favorite article database table.
Add in more categories of recipes	The implemented system should have a variety of recipe categories provided on the Recipes home screen. The current implemented system has only 2 categories, which are the eating preferences and the cuisine types. By adding in more categories of recipes, the users can browse through different categories easily without needing to search from the search bar frequently.
Implement the suggestions feature to suggest the recipes and articles based on the user's current BMI	The implemented system should have a suggestions feature to suggest the recipes and articles based on the user's current BMI. This feature is extremely useful to the users as the users can know which recipes to be prepared without over intake. Not only that, a collection of appropriate health-related articles suggested to the users can help them to tackle the possible health-related issues faced.
Subscribe to the premium version of APIs	The implemented system should subscribe to the premium version of APIs. The current implemented system is subscribing to the

	free version of APIs with the limited quota. When the implemented system scales up, it is better to subscribe to the premium version of APIs to prevent extra fees incurred when a lot of requests are sent to the API endpoints.
Implement the user authentication	The implemented system should implement the user authentication to confirm the user's identity and for better security.

REFERENCES

Al Mamun, A., Hayat, N. and Zainol, N.R.B., 2020. Healthy eating determinants: a study among malaysian young adults. *Foods*, 9(8).

American Psychological Association, 2020. Stress in America^{TM} 2020: A National Mental Health Crisis. [online] Available at: <https://www.apa.org/news/press/releases/stress/2020/sia-mental-health-crisis.pdf>.

Anon 2020. MCO diets drive up Malaysia's obesity rates | The Star. *The Star*. [online] 22 Jun. Available at: <https://www.thestar.com.my/news/nation/2020/06/22/mco-diets-drive-up-malaysias-obesity-rates> [Accessed 7 Aug. 2021].

Correa, H., 2020. *Introduction to Scrum*. [online] Available at: <https://www.codemag.com/article/0805051/Introduction-to-Scrum> [Accessed 10 August 2021].

Daneshmandi, H., Choobineh, A., Ghaem, H. and Karimi, M., 2017. Adverse Effects of Prolonged Sitting Behavior on the General Health of Office Workers. *Journal of Lifestyle Medicine*, [online] 7(2), pp.69–75. Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5618737/>.

Gounder, S., 2012. Chapter 3 - Research methodology and research questions. *Research Methodology and Research Method*, (March 2012), pp.84–193.

Hassan, H., Sade, A.B. and Subramaniam, L.S., 2020. Purchasing functional foods to stay fit. *Journal of Humanities and Applied Social Sciences*, [online] 2(1), pp.3–18. Available at: <https://www.emerald.com/insight/content/doi/10.1108/JHASS-11-2019-0073/full/html>.

Koci, D., Morina, F., Hoxhaj, F. and Ukëhaxhaj, A., 2021. The impact of COVID-19 on mental health. *International Journal of Biology and Biomedical Engineering*, 15(1), pp.202–211.

Kumar, K., 2017. Importance of Healthy Life Style in Healthy living. *Juniper Online Journal of Public Health*, 2(5), pp.9–11.

Lewis, J.R., 2018. The System Usability Scale: Past, Present, and Future. *International Journal of Human-Computer Interaction*, 34(7), pp.577–590.

Lewis, J.R. and Sauro, J., 2018. Item Benchmarks for the System Usability Scale. *Journal of Usability Studies*, 13(3), pp.158–167.

Marçal, I.R., Fernandes, B., Viana, A.A. and Ciolac, E.G., 2020. The Urgent Need for Recommending Physical Activity for the Management of Diabetes

During and Beyond COVID-19 Outbreak. *Frontiers in Endocrinology*, [online] 11(October). Available at: <https://www.frontiersin.org/articles/10.3389/fendo.2020.584642/full>.

Marzo, R.R., Vinay, V., Bahari, R., Chauhan, S., Ming, D.A.F., Nelson Fernandez, S.F.A., Johnson, C.C.P., Thivakaran, A.Q.A., Rahman, M.M. and Goel, S., 2021. Depression and anxiety in Malaysian population during third wave of the COVID-19 pandemic. *Clinical Epidemiology and Global Health*, 12. https://doi.org/10.1016/j.cegh.2021.100868.

Meier, J., 2019. *Extreme Programming at a Glance - JD Meier*. [online] Available at: <https://jdmeier.com/extreme-programming-at-a-glance/> [Accessed 10 August 2021].

Moy, F.M., Hairi, N.N. and Wan, K.S., 2021. LETTER | Obesity in Malaysia is a ticking time bomb. *Malaysiakini*. [online] Available at: <https://www.malaysiakini.com/letters/565417> [Accessed 2 Jul. 2021].

Olajide, A.O., 2016. *Software Engineering Lecture Notes on SDLC Models*. https://doi.org/10.13140/RG.2.2.30487.78243.

Pell, C., Allotey, P., Evans, N., Hardon, A., Imelda, J.D., Soyiri, I. and Reidpath, D.D., 2016. Coming of age, becoming obese: A cross-sectional analysis of obesity among adolescents and young adults in Malaysia. *BMC Public Health*, 16(1), pp.1–11.

Ramli, N., Ghani, F.A., Nawawi, W.N.W. and Majid, H.A.M.A., 2020. Intention to Use Online Food Ordering Services Among Universities Students During COVID-19 Pandemic. *International Journal of Academic Research in Business and Social Sciences*, [online] 10(17), pp.1–12. Available at: <https://hrmars.com/papers_submitted/8556/intention-to-use-online-food-ordering-services-among-universities-students-during-covid-19-pandemic.pdf>.

Saranya, P., Monica, V., Priyadharshini, J. and N, A.P.D., 2017. Comparative Study of Software Development Methodologies. pp.172–179.

Sasmito, G.W., Zulfiqar, L.O.M. and Nishom, M., 2019. Usability Testing based on System Usability Scale and Net Promoter Score. *2019 2nd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2019*, pp.540–545.

Senarath, U.S., 2021. Waterfall Methodology , Prototyping and Agile Development By. (July).

Shaikh, S. and Abro, S., 2019. Comparison of Traditional and Agile Software Development Methodology: a Short Survey. *International Journal of Software Engineering and Computer Systems*, 5(2), pp.1–14.

Sommerville, I., Columbus, B., New, I., San, Y., Upper, F., River, S., Cape, A., Dubai, T., Madrid, L., Munich, M., Montreal, P., Delhi, T., São, M.C., Sydney, P., Kong, H., Singapore, S. and Tokyo, T., 2011. *SOFTWARE ENGINEERING*. 9th ed. [online] Available at: <https://engineering.futureuniversity.com/BOOKS FOR IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf> [Accessed 14 Jul. 2021].

Srivastava, A., Bhardwaj, S. and Saraswat, S., 2017. SCRUM model for agile methodology. *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, 2017-Janua, pp.864–869.

Stockwell, S., Trott, M., Tully, M., Shin, J., Barnett, Y., Butler, L., McDermott, D., Schuch, F. and Smith, L., 2021. *Changes in physical activity and sedentary behaviours from before to during the COVID-19 pandemic lockdown: A systematic review*. *BMJ Open Sport and Exercise Medicine*, https://doi.org/10.1136/bmjsem-2020-000960.

Yasvi, M., Yadav, K. and Sahendrasingh. S., 2019. Review On Extreme Programming-XP. *International Conference on Robotics, Smart Technology and Electronics Engineering, At Delhi*, [online] (April), pp.1–8. Available at: <https://www.researchgate.net/publication/332465869%0D>.

Zheng, C., Huang, W.Y., Sheridan, S., Sit, C.H.P., Chen, X.K. and Wong, S.H.S., 2020. Covid-19 pandemic brings a sedentary lifestyle in young adults: A cross-sectional and longitudinal study. *International Journal of Environmental Research and Public Health*, 17(17), pp.1–11.

APPENDICES

Appendix A: Existing Similar Application Review

Table A-1: Existing Similar Application Review Matrix

Application	Descriptions	Features	Useful User Reviews
Carb Manager: Keto Diet Tracker & Fasting App (released in year 2010) Author: Wombat Apps LLC Rating: 4.6 (192K reviews) No. of downloads: 5M+	This application allows the user to track their daily meals, guide the users by providing a macro calculator, low carb food recipes, articles, forums, meal planner and establish weekly challenges in losing weight. It is an application which acts like a small community that provides almost everything needed for an individual who wishes to lose weight and take care of their health.	Free version provides the features such as: - Track daily meals - Track exercise - Track weight - Limited food recipes - Articles related to keto, weight loss, and health conditions There is a paid version which provides additional features such as: - Exclusive keto recipes - Meal planning - Track intermittent fasting	Author: Angela Valdez Date: 4 July 2021 Rating: 5 Review: It helps to keep track of carb, protein and calories in each meal. It also recommends the user how much is the calories limit for each meal. It provides a lot of healthy food recipes. Author: Salvatore Leone Date: 8 July 2021 Rating: 3 Review: This application helps to track macro,

		- Keto video courses - Full nutrition tracking - Sync fitness wearable devices - Generate shopping list - Generate various kind of report such as macro and meal analysis - Community support and weekly challenges - etc.	and it has a huge food database that provides the nutrient information for the meal. However, it does not calculate the calories intake and burnt. Author: Donefe Don Date: 23 July 2021 Rating: 1 Review: This application is not user friendly for vegetarian users. The filter function in the recipe section does not help a lot in filtering out the vegan friendly recipes.
Lifesum - Diet Plan, Macro Calculator & Food Diary (released in year 2013)	This application allows the users to track their meals, nutrition, calories and fitness. It helps	Free version provides the features such as: - Track daily meals - Track exercise	Author: KathrynOwn Date: 22 July 2021 Rating: 5

Author: Lifesum AB Rating: 4.5 (282K reviews) No. of downloads: 10M+	the users to monitor the macro. There are some premium features such as a barcode scanner to log the meal, meal planner and premium keto and weight loss recipes.	- Track weight - Macro calculator There is a paid version which provides additional features such as: - Macro tracking - Diet plans and recipes - Save meals - Sync with health application - Life Score nutrition insights	Review: This application has a good layout that makes sense to me. It is simple to use. However, it can sometimes be hard to add custom food into the food log, but this is not a big issue as I can just work around it and make adjustments on the measurement of food. Author: KDi Date: 11 July 2021 Rating: 4 Review: The free version of this application provides the basic features to the users. However, it should allow the user to set the timer for

			notifications to pop out. Author: Emily Hartigan Date: 3 July 2021 Rating: 5 Review: This application is pleasant to use. The new features that I wish it could have include photo capture for recording the meal portion and the time of the meal intake.
MyPlate Calorie Tracker Author: LIVESTRONG Rating: 4.6 (42K reviews) No. of downloads: 1M+	This application allows the users to track their daily meal, exercise and water consumption. It will calculate the net calories of the user by deducting the calories burnt from the calories	Free version provides the features such as: - Track daily meals - Track exercise - Macro calculator - Meal, exercise and water reminders - Tracking of activity by	Author: Peter Sheaf Date: 11 June 2021 Rating: 4 Review: This application can combine the diet and exercise data for the ease of tracking the progress. However, the

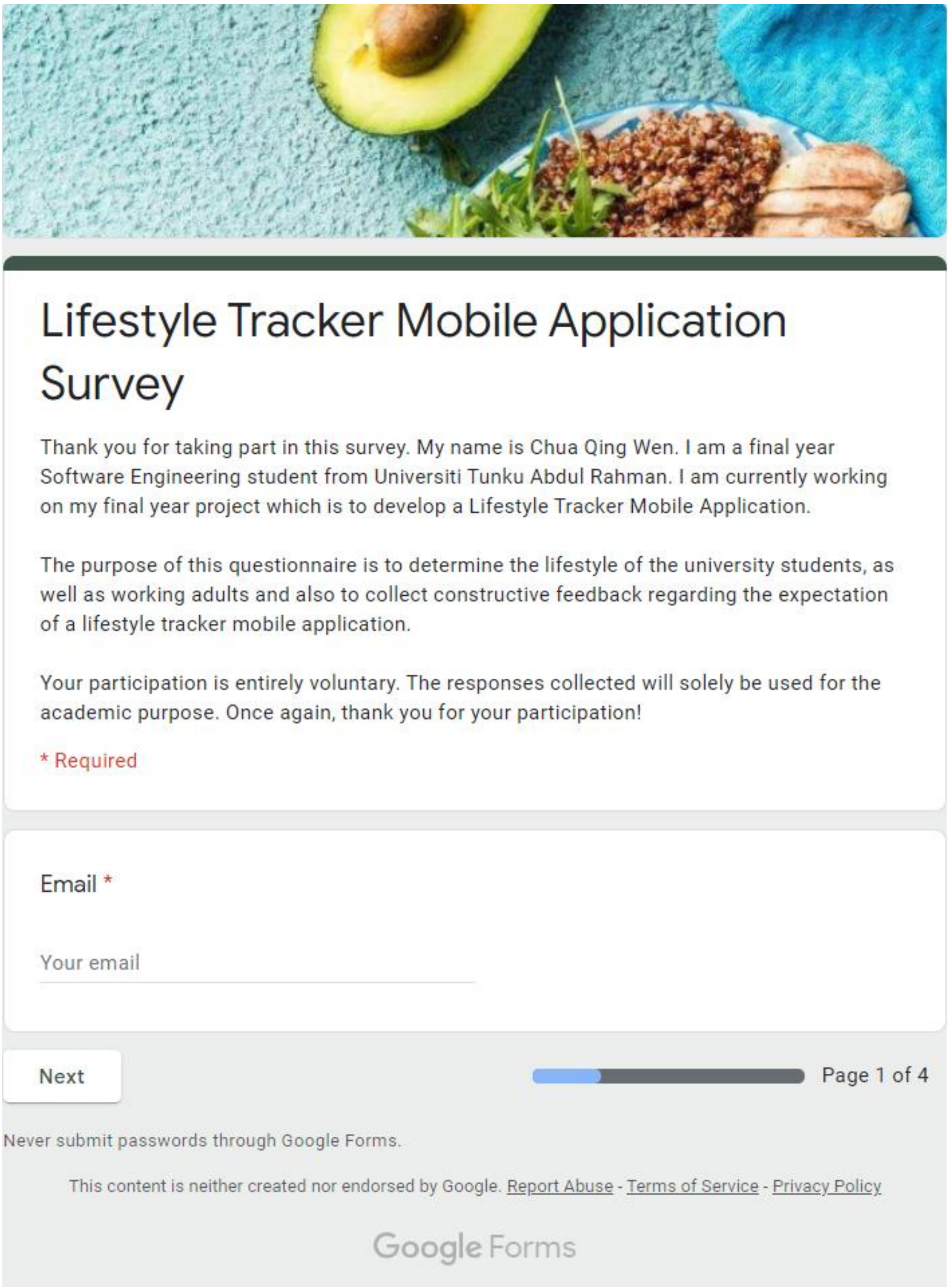
	consumed. It also allows the users to set the meal, exercise and water reminders. The users can post their status and progress in the community and interact with other users.	integrating with Google Fit app - Community There is a paid version which provides additional features such as: - Advanced statistics and meal macro's breakdown - Private community board and priority customer support - Diet plans and recipes	units used are the US-centric measures and the conversion from the unit cups to grams is tedious. Author: Kathy Den Date: 27 May 2021 Rating: 5 Review: This application consists of an exhaustive list of food that allows the users to add into the food log. But the calories count of some foods listed are wrong. The user can just modify it manually, so it is not a big issue. Author: Blong Vang Date: 11 July 2021 Rating: 5

			Review: This application has a simple interface. It allows the users to set up their own foods in the food log.
ekilu - eat well, exercise & mindfulness Author: Noodle App Rating: 4.0 (5K reviews) No. of downloads: 1M+	This application promotes a balanced lifestyle in terms of nutrition, movement and mindfulness. It provides the users with a lot of recipes with various categories based on the users' eating preference and restrictions. It also tracks the users' movement by recording the steps the user takes per day. Not only that, it allows the user to record down the mindfulness time too. There are tips provided	Free version provides the features such as: - Track daily meals - Track daily steps - Track daily mindfulness time - Limited access of food recipes There is a paid version which provides additional features such as: - Meal planner - Exclusive food recipes - Nutrition breakdown for every recipe - Track meal balance	Author: XxMokiii_xX Date: 17 July 2021 Rating: 5 Review: This application provides ideas of meal preparation that can be done within 10 minutes. It is helpful to me as I do not always have time and money to make fancy dishes everyday. Author: SSDOGLOVER Date: 11 April 2021 Rating: 5 Review: This application is the

	to let the users learn more on how to achieve a balanced life.	- Meal recommendations - Automated grocery list	best cooking application. It allows the user to input any of the ingredients to search for relevant recipes. It also will ask for your eating preference and only bring up those recipes the users probably will like. Author: merna mohamed Date: 20 August 2020 Rating: 5 Review: This application is great. I wish there is a calories section for each meal so that I can keep track of the carbs, protein and fat.

Appendix B: Complete Survey Form Questions

Figure B-1: Introduction Page Of The Survey Form



Lifestyle Tracker Mobile Application Survey

Thank you for taking part in this survey. My name is Chua Qing Wen. I am a final year Software Engineering student from Universiti Tunku Abdul Rahman. I am currently working on my final year project which is to develop a Lifestyle Tracker Mobile Application.

The purpose of this questionnaire is to determine the lifestyle of the university students, as well as working adults and also to collect constructive feedback regarding the expectation of a lifestyle tracker mobile application.

Your participation is entirely voluntary. The responses collected will solely be used for the academic purpose. Once again, thank you for your participation!

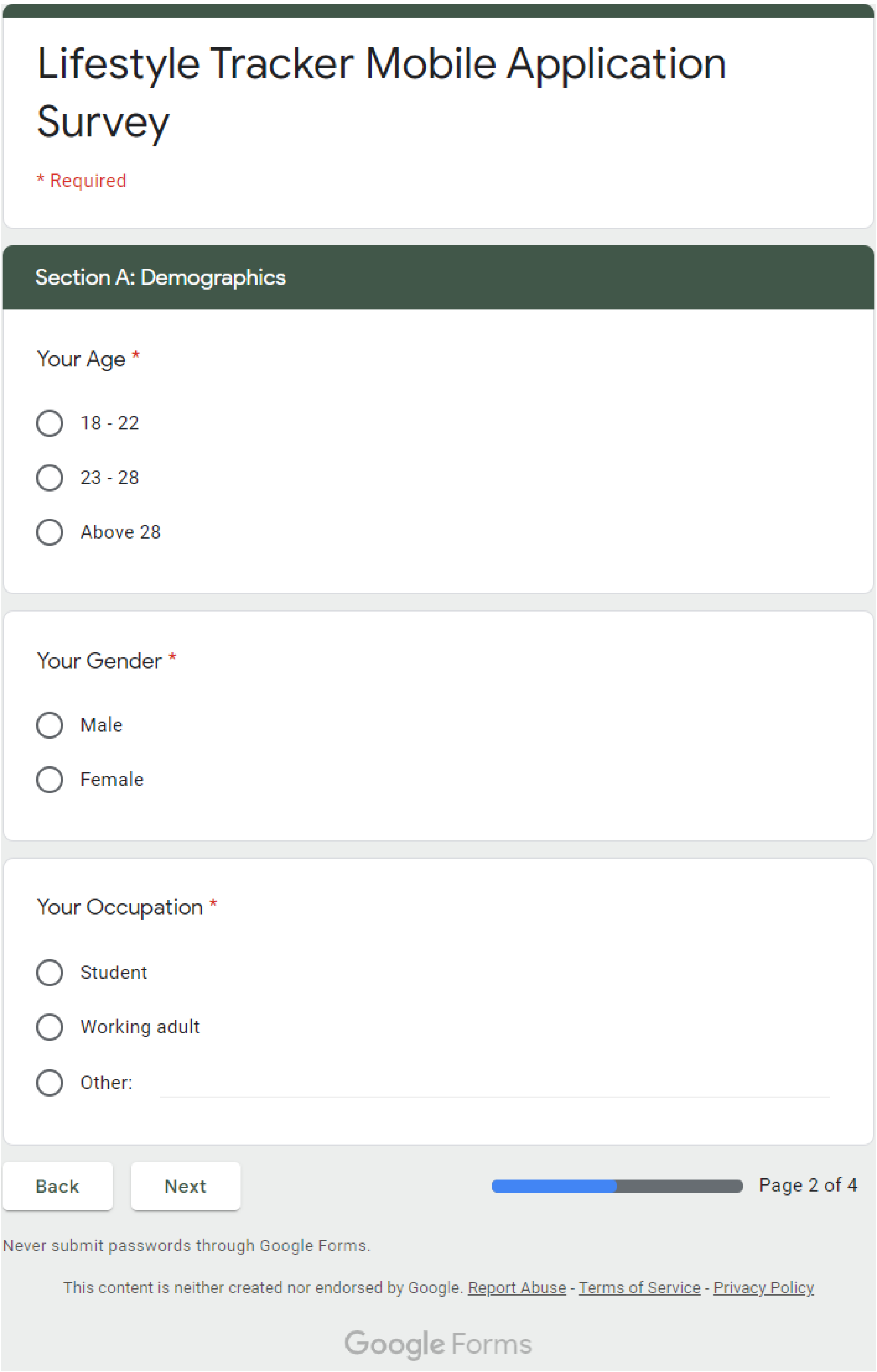
* Required

Email *

Your email

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Figure B-2: Section A Of The Survey Form



Lifestyle Tracker Mobile Application Survey

* Required

Section A: Demographics

Your Age *

○ 18 - 22

○ 23 - 28

○ Above 28

Your Gender *

○ Male

○ Female

Your Occupation *

○ Student

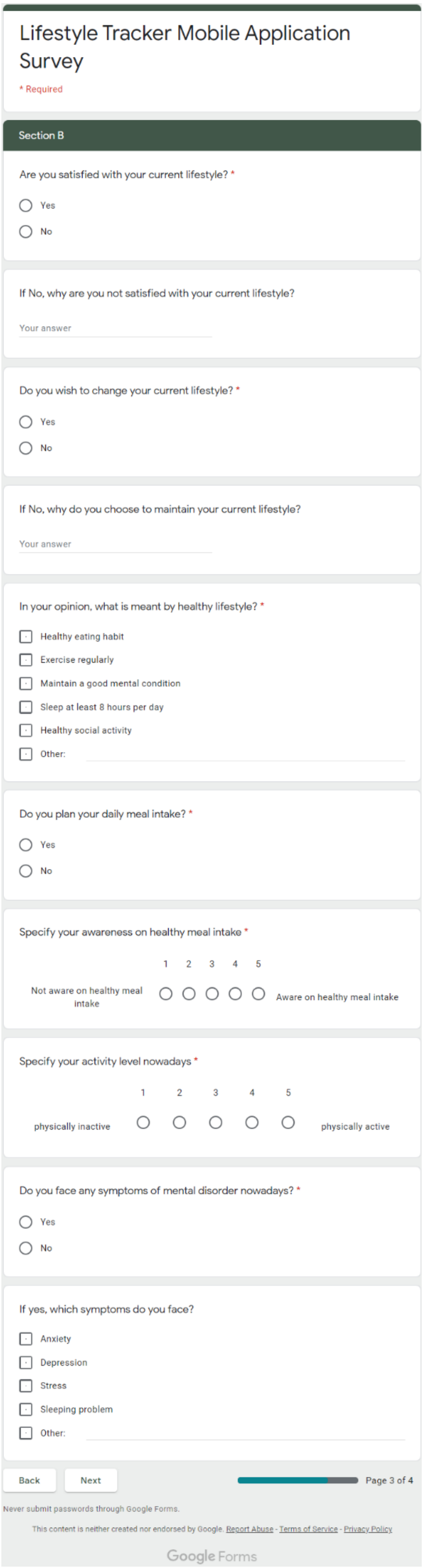
○ Working adult

○ Other: ____________________

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Figure B-3: Section B Of The Survey Form



Lifestyle Tracker Mobile Application Survey

* Required

Section B

Are you satisfied with your current lifestyle? *

○ Yes

○ No

If No, why are you not satisfied with your current lifestyle?

Your answer

Do you wish to change your current lifestyle? *

○ Yes

○ No

If No, why do you choose to maintain your current lifestyle?

Your answer

In your opinion, what is meant by healthy lifestyle? *

□ Healthy eating habit

□ Exercise regularly

□ Maintain a good mental condition

□ Sleep at least 8 hours per day

□ Healthy social activity

□ Other:

Do you plan your daily meal intake? *

○ Yes

○ No

Specify your awareness on healthy meal intake *

	1	2	3	4	5	
Not aware on healthy meal intake	○	○	○	○	○	Aware on healthy meal intake

Specify your activity level nowadays *

	1	2	3	4	5	
physically inactive	○	○	○	○	○	physically active

Do you face any symptoms of mental disorder nowadays? *

○ Yes

○ No

If yes, which symptoms do you face?

□ Anxiety

□ Depression

□ Stress

□ Sleeping problem

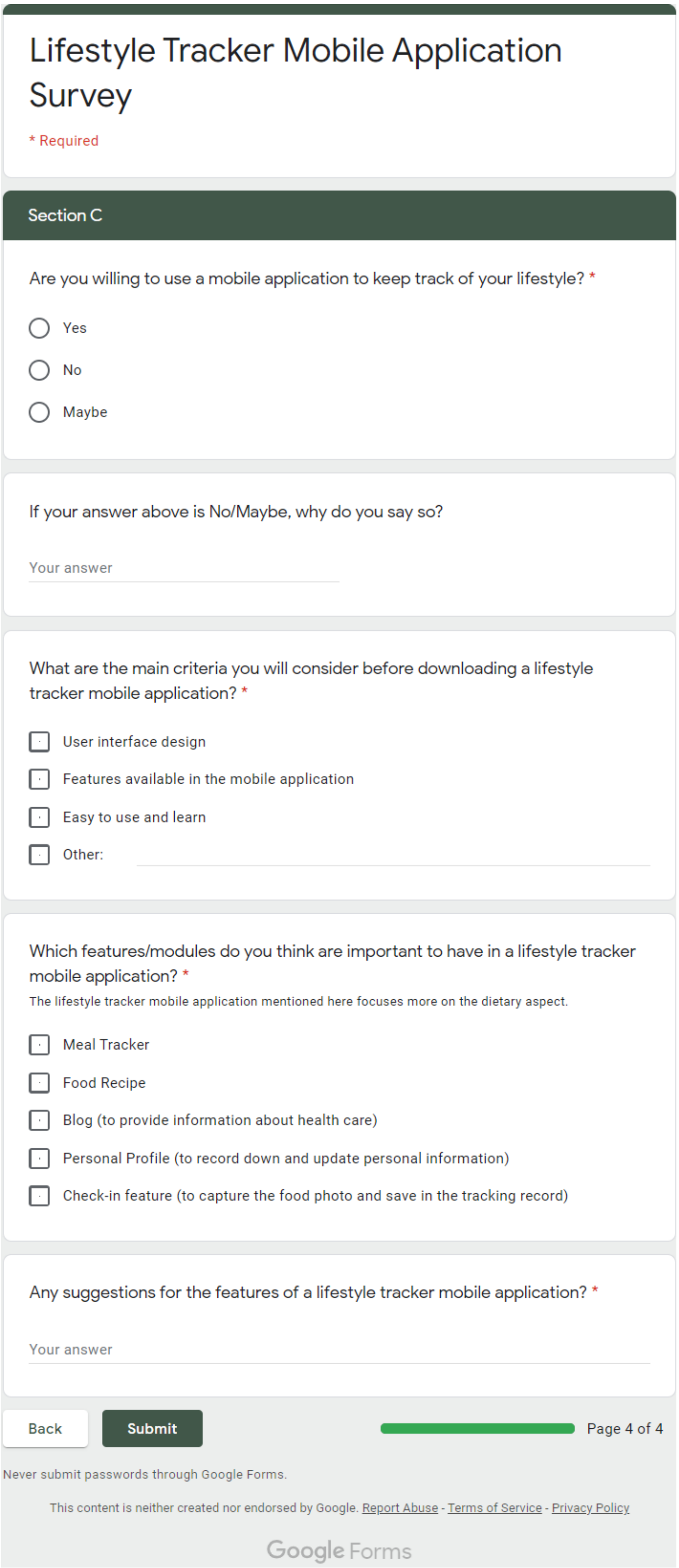
□ Other:

Never submit passwords through Google Forms.

This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy

Google Forms

Figure B-4: Section C Of The Survey Form



Lifestyle Tracker Mobile Application Survey

* Required

Section C

Are you willing to use a mobile application to keep track of your lifestyle? *

○ Yes

○ No

○ Maybe

If your answer above is No/Maybe, why do you say so?

Your answer

What are the main criteria you will consider before downloading a lifestyle tracker mobile application? *

□ User interface design

□ Features available in the mobile application

□ Easy to use and learn

□ Other:

Which features/modules do you think are important to have in a lifestyle tracker mobile application? *

The lifestyle tracker mobile application mentioned here focuses more on the dietary aspect.

□ Meal Tracker

□ Food Recipe

□ Blog (to provide information about health care)

□ Personal Profile (to record down and update personal information)

□ Check-in feature (to capture the food photo and save in the tracking record)

Any suggestions for the features of a lifestyle tracker mobile application? *

Your answer

Back
Submit
Page 4 of 4
Never submit passwords through Google Forms.
This content is neither created nor endorsed by Google. Report Abuse - Terms of Service - Privacy Policy
Google Forms

Appendix C: Complete Survey Form Responses

Figure C-1: Responses from Question 1 In Section A

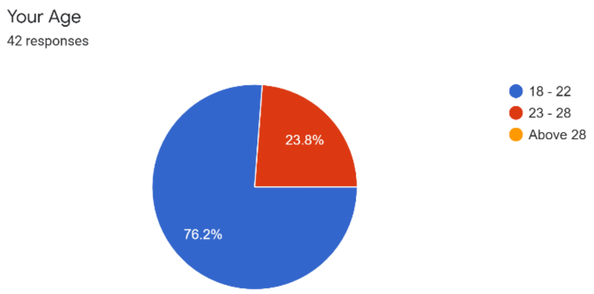


Figure C-2: Responses from Question 2 In Section A

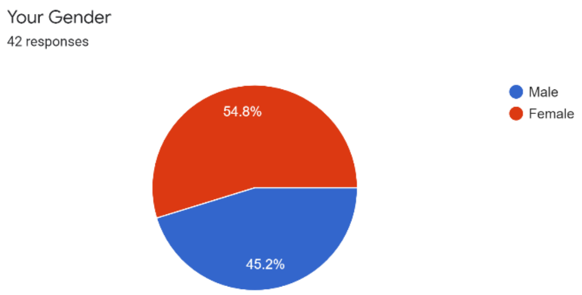


Figure C-3: Responses from Question 3 In Section A

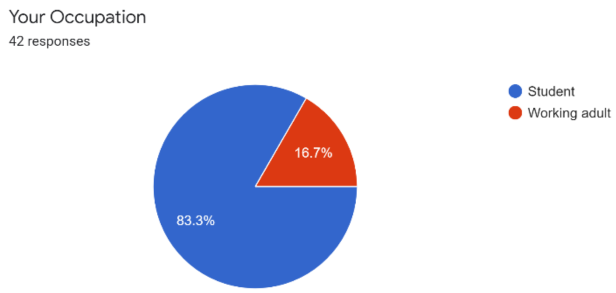


Figure C-4: Responses from Question 1 In Section B

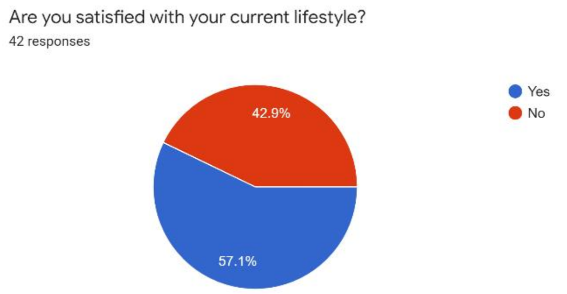
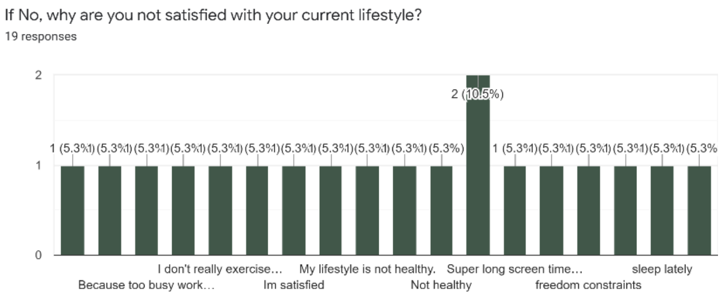


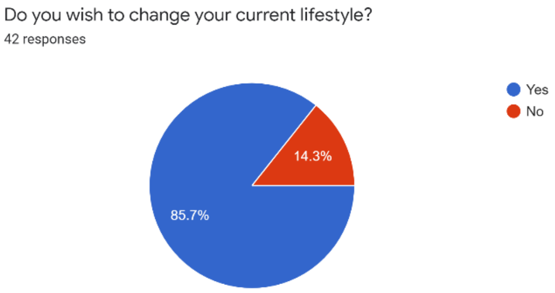
Figure C-5: Responses from Question 2 In Section B



If No, why are you not satisfied with your current lifestyle?

19 responses

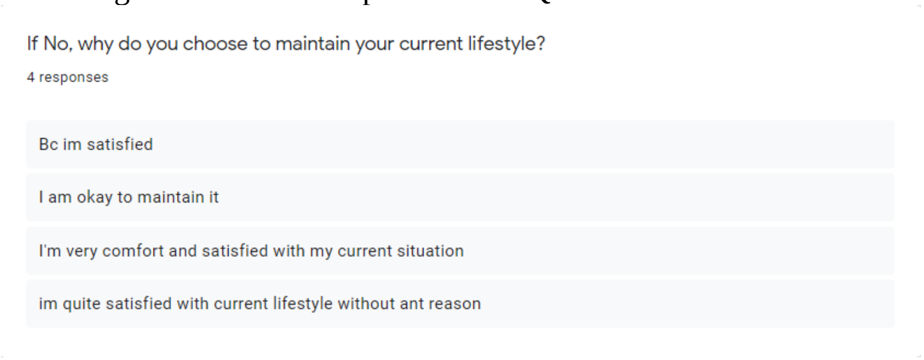
Figure C-6: Responses from Question 3 In Section B



Do you wish to change your current lifestyle?

42 responses

Figure C-7: Responses from Question 4 In Section B



If No, why do you choose to maintain your current lifestyle?

4 responses

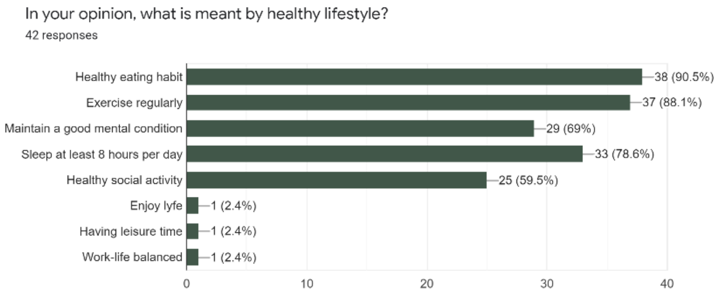
Bc im satisfied

I am okay to maintain it

I'm very comfort and satisfied with my current situation

im quite satisfied with current lifestyle without ant reason

Figure C-8: Responses from Question 5 In Section B



In your opinion, what is meant by healthy lifestyle?

42 responses

Figure C-9: Responses from Question 6 In Section B

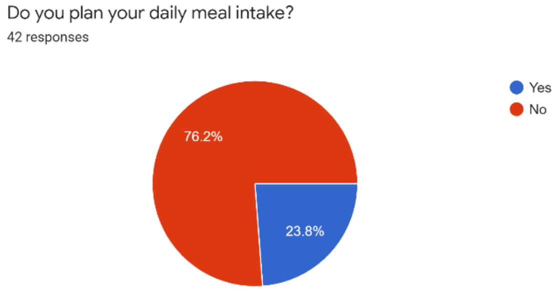


Figure C-10: Responses from Question 7 In Section B

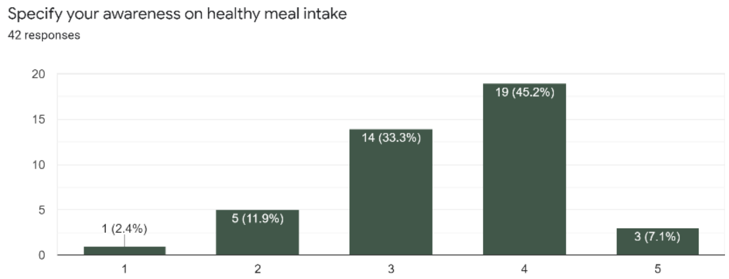


Figure C-11: Responses from Question 8 In Section B

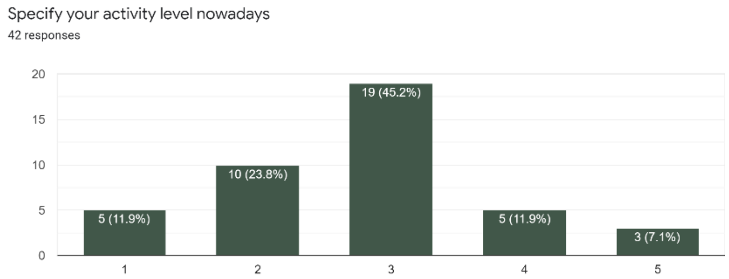


Figure C-12: Responses from Question 9 In Section B

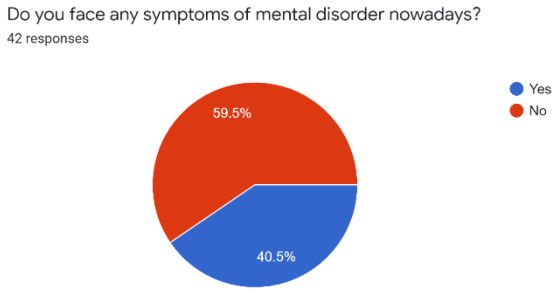
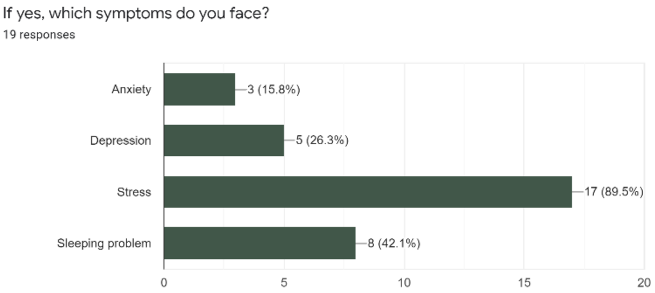
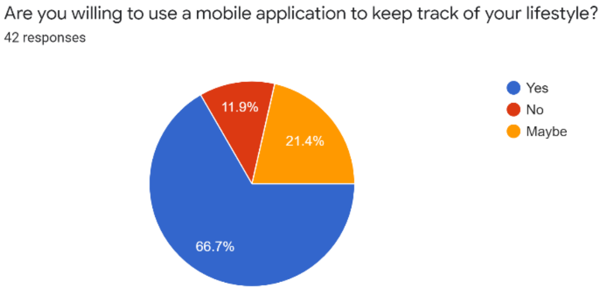


Figure C-13: Responses from Question 10 In Section B



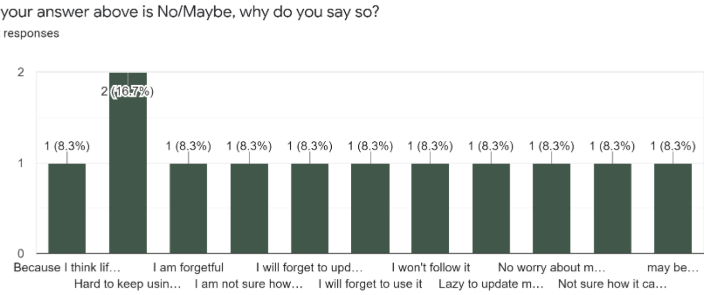
If yes, which symptoms do you face?
19 responses

Figure C-14: Responses from Question 1 In Section C



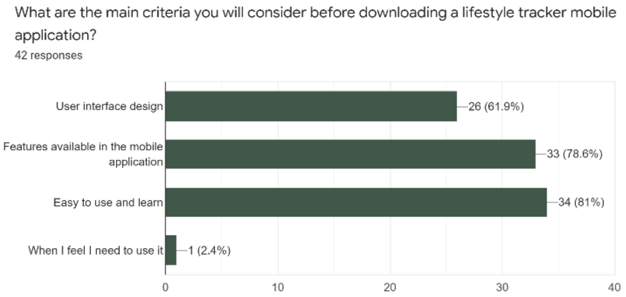
Are you willing to use a mobile application to keep track of your lifestyle?
42 responses

Figure C-15: Responses from Question 2 In Section C



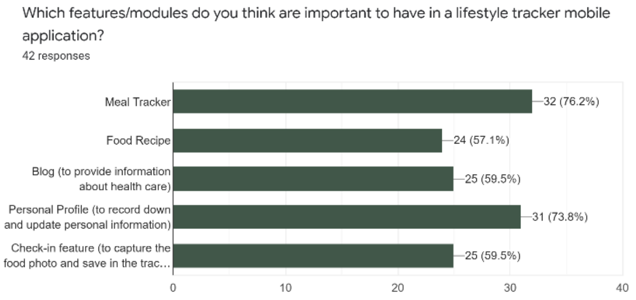
If your answer above is No/Maybe, why do you say so?
12 responses

Figure C-16: Responses from Question 3 In Section C



What are the main criteria you will consider before downloading a lifestyle tracker mobile application?
42 responses

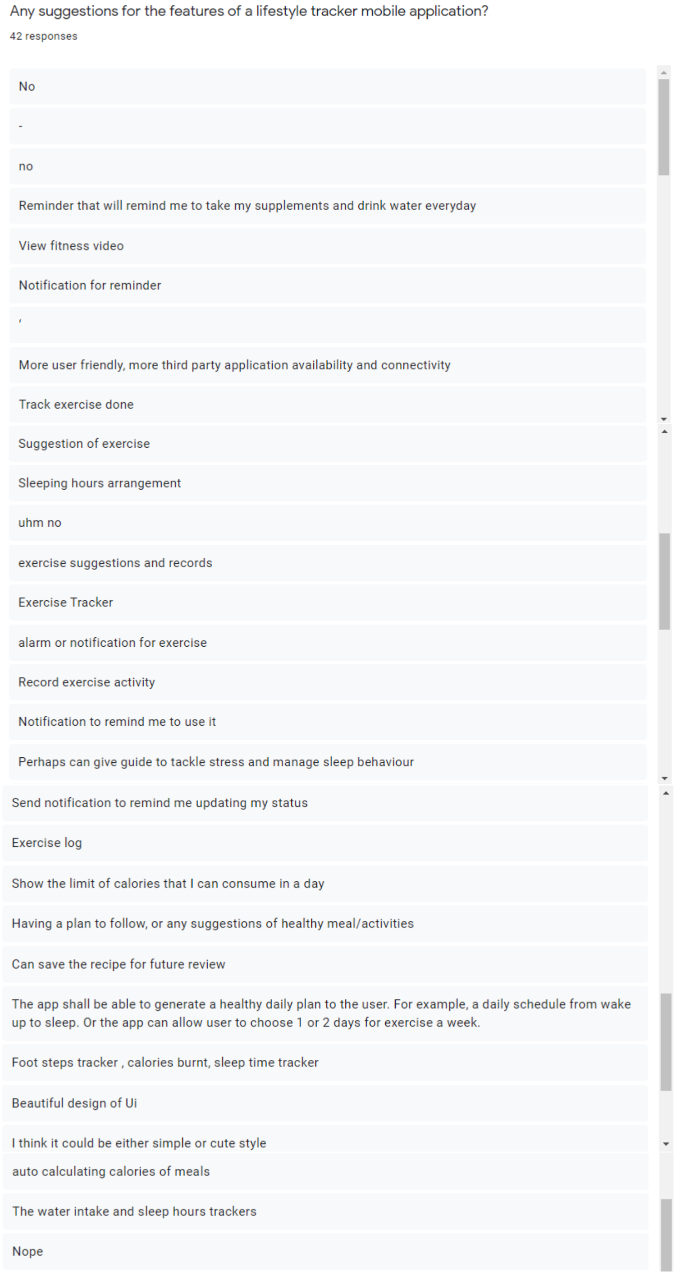
Figure C-17: Responses from Question 4 In Section C



Which features/modules do you think are important to have in a lifestyle tracker mobile application?

42 responses

Figure C-18: Responses from Question 5 In Section C



Any suggestions for the features of a lifestyle tracker mobile application?

42 responses

No

-

no

Reminder that will remind me to take my supplements and drink water everyday

View fitness video

Notification for reminder

,

More user friendly, more third party application availability and connectivity

Track exercise done

Suggestion of exercise

Sleeping hours arrangement

uhm no

exercise suggestions and records

Exercise Tracker

alarm or notification for exercise

Record exercise activity

Notification to remind me to use it

Perhaps can give guide to tackle stress and manage sleep behaviour

Send notification to remind me updating my status

Exercise log

Show the limit of calories that I can consume in a day

Having a plan to follow, or any suggestions of healthy meal/activities

Can save the recipe for future review

The app shall be able to generate a healthy daily plan to the user. For example, a daily schedule from wake up to sleep. Or the app can allow user to choose 1 or 2 days for exercise a week.

Foot steps tracker , calories burnt, sleep time tracker

Beautiful design of Ui

I think it could be either simple or cute style

auto calculating calories of meals

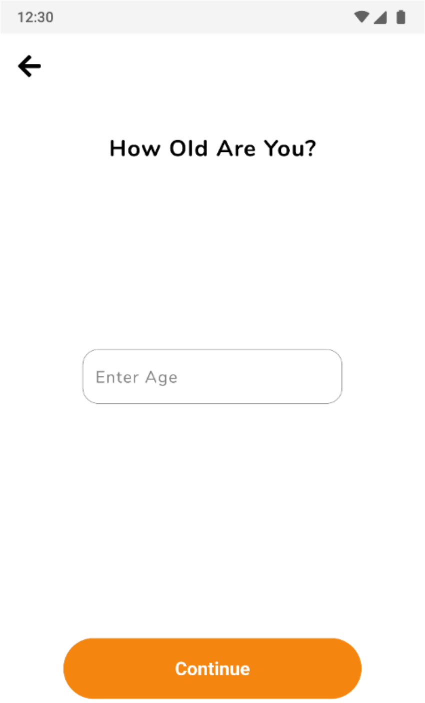
The water intake and sleep hours trackers

Nope

Appendix D: High Fidelity Prototype Design of the Implemented System

The process of registering an account

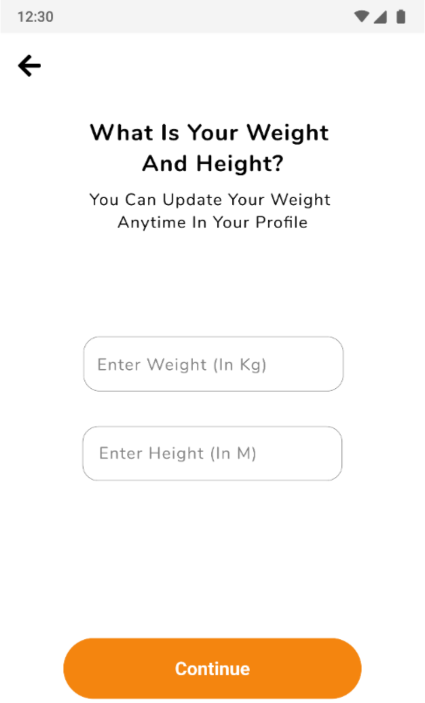
Figure D-1: Register screen that asking for the age



12:30

How Old Are You?

Figure D-2: Register screen that asking for the weight and height

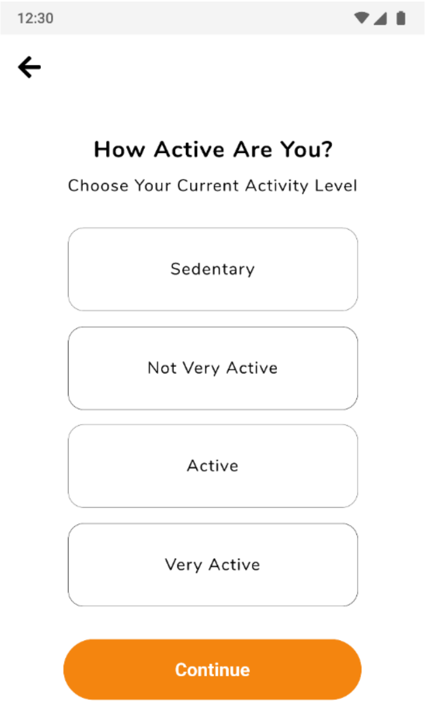


12:30

What Is Your Weight And Height?

You Can Update Your Weight Anytime In Your Profile

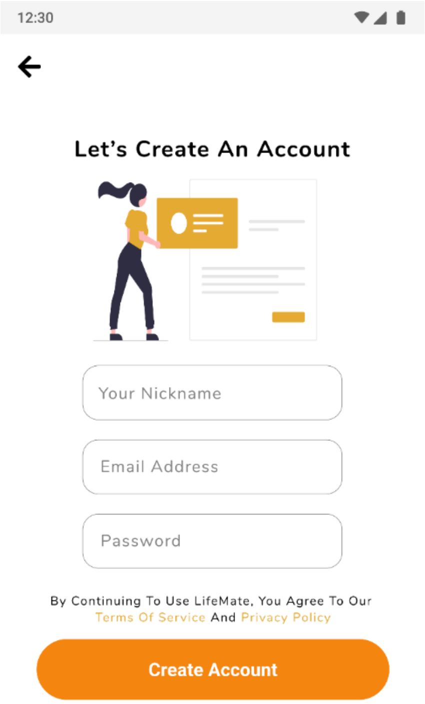
Figure D-3: Register screen that asking for the activity level



How Active Are You?

Choose Your Current Activity Level

Figure D-4: Create Account screen

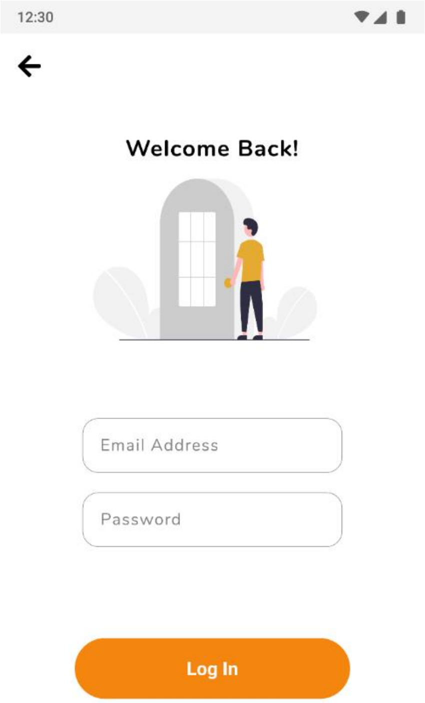


Let's Create An Account

By Continuing To Use LifeMate, You Agree To Our Terms Of Service And Privacy Policy

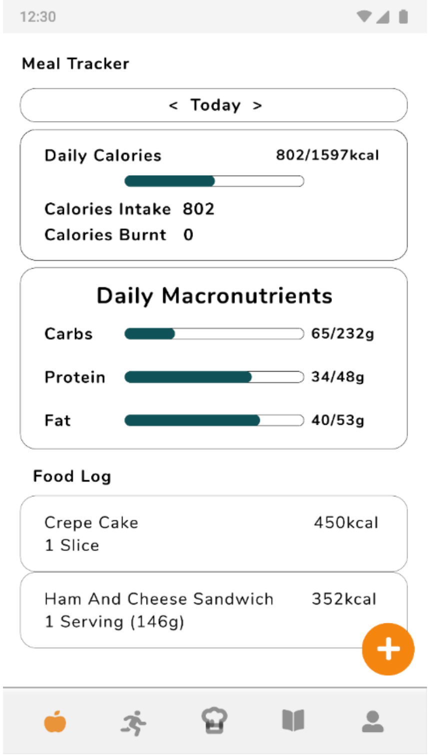
Sign In Account

Figure D-5: Sign In screen



Meal Tracker

Figure D-6: Meal Tracker home screen



Meal Tracker

< Today >

Daily Calories 802/1597kcal

Calories Intake 802
Calories Burnt 0

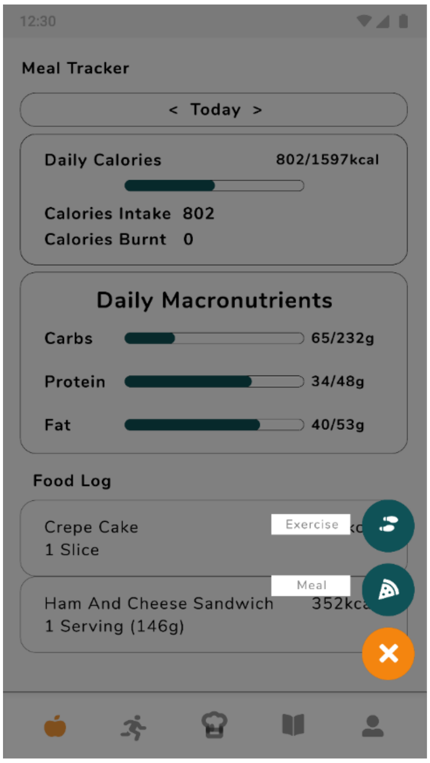
Daily Macronutrients

Carbs	65/232g
Protein	34/48g
Fat	40/53g

Food Log

Crepe Cake 1 Slice	450kcal
Ham And Cheese Sandwich 1 Serving (146g)	352kcal

Figure D-7: Options from the Add symbol (+) floating button



12:30

Meal Tracker

< Today >

Daily Calories 802/1597kcal

Calories Intake 802
Calories Burnt 0

Daily Macronutrients

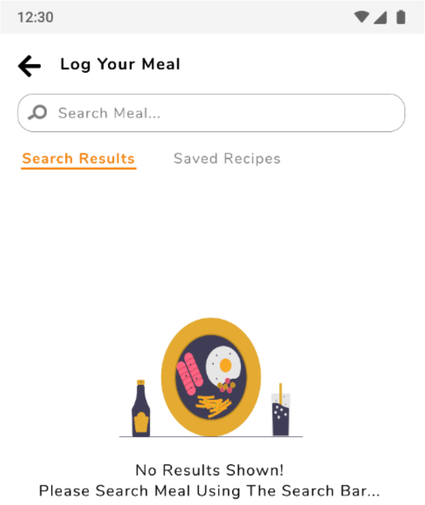
Carbs	65/232g
Protein	34/48g
Fat	40/53g

Food Log

Crepe Cake
1 Slice

Ham And Cheese Sandwich 352kca
1 Serving (146g)

Figure D-8: Search Meal screen



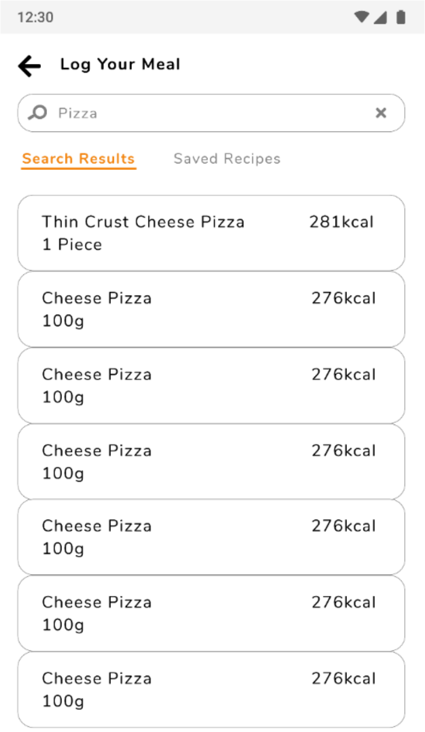
12:30

Log Your Meal

Search Meal...

No Results Shown!
Please Search Meal Using The Search Bar...

Figure D-9: Search Meal screen with results



12:30

Log Your Meal

Pizza

Search Results

Saved Recipes

Thin Crust Cheese Pizza 281kcal
1 Piece

Cheese Pizza 276kcal
100g

Cheese Pizza 276kcal
100g

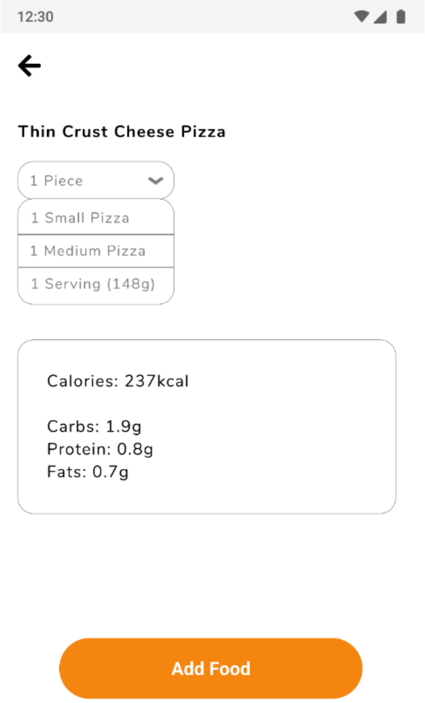
Cheese Pizza 276kcal
100g

Cheese Pizza 276kcal
100g

Cheese Pizza 276kcal
100g

Cheese Pizza 276kcal
100g

Figure D-10: Meal Details screen



12:30

Thin Crust Cheese Pizza

1 Piece

1 Small Pizza

1 Medium Pizza

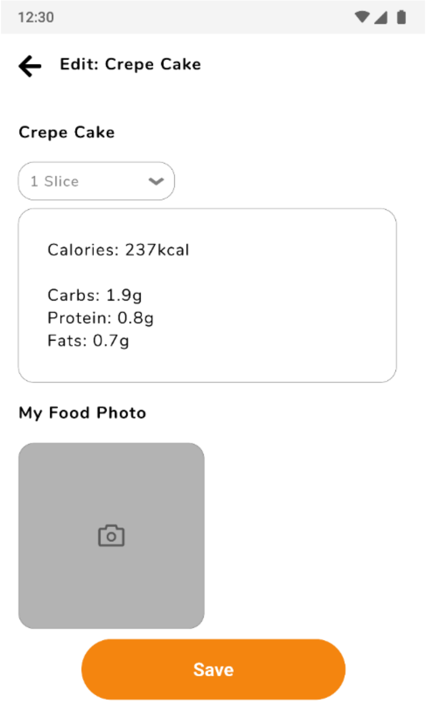
1 Serving (148g)

Calories: 237kcal

Carbs: 1.9g
Protein: 0.8g
Fats: 0.7g

Add Food

Figure D-11: Edit Food Log Record screen



12:30

Edit: Crepe Cake

Crepe Cake

1 Slice

Calories: 237kcal

Carbs: 1.9g
Protein: 0.8g
Fats: 0.7g

My Food Photo

Figure D-12: User’s device camera

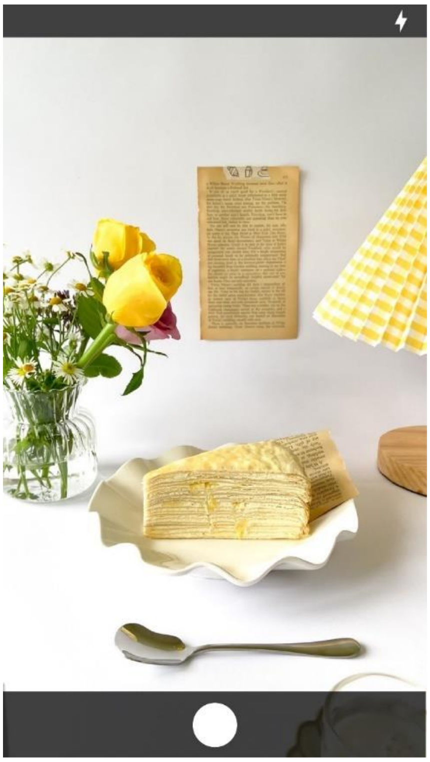
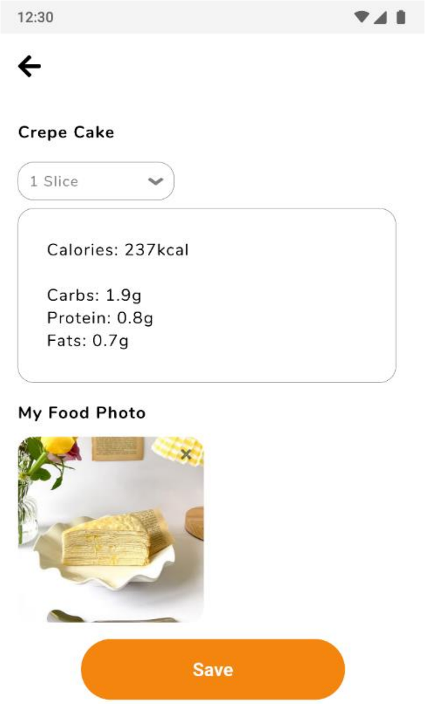


Figure D-13: Edit Food Log Record screen with food photo



Crepe Cake

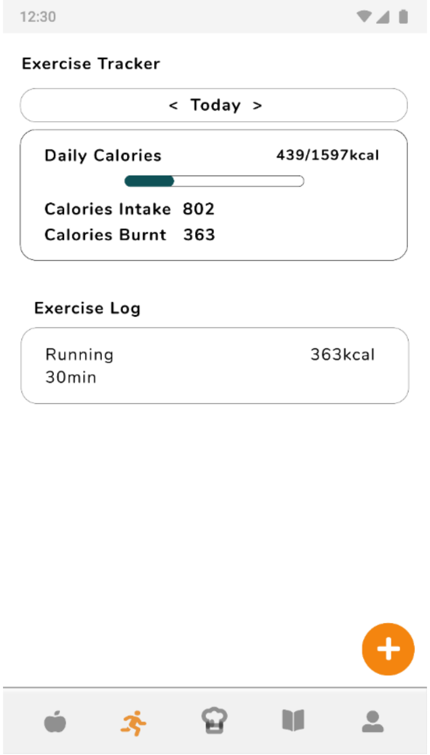
Calories: 237kcal

Carbs: 1.9g
Protein: 0.8g
Fats: 0.7g

My Food Photo

Exercise Tracker

Figure D-14: Exercise Tracker screen



Exercise Tracker

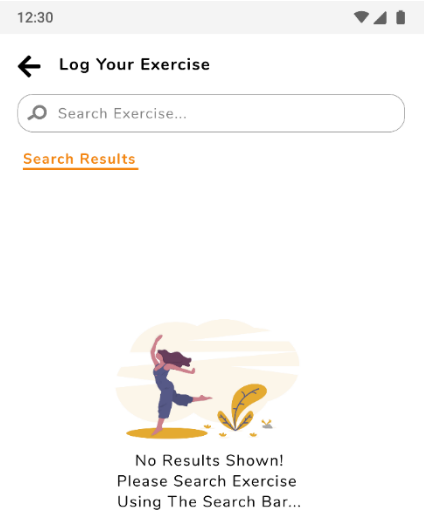
Daily Calories 439/1597kcal

Calories Intake 802
Calories Burnt 363

Exercise Log

Running 363kcal
30min

Figure D-15: Search Exercise screen



12:30

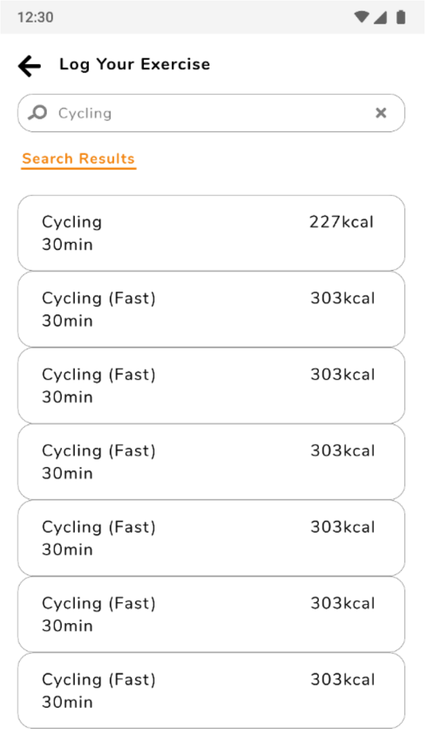
Log Your Exercise

Search Exercise...

Search Results

No Results Shown!
Please Search Exercise
Using The Search Bar...

Figure D-16: Search Exercise screen with results



12:30

Log Your Exercise

Cycling

Search Results

Cycling 227kcal
30min

Cycling (Fast) 303kcal
30min

Cycling (Fast) 303kcal
30min

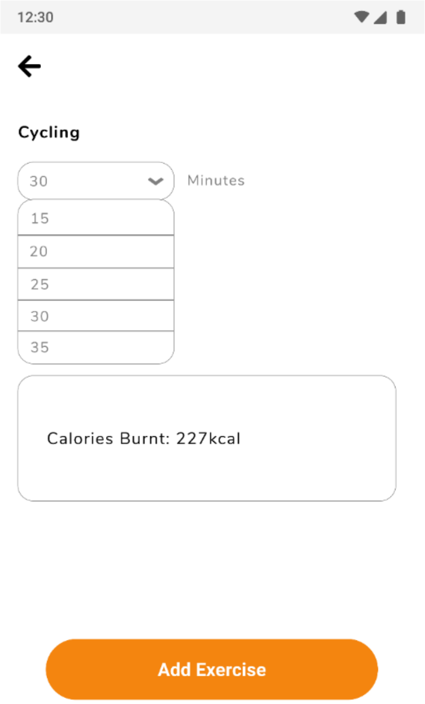
Cycling (Fast) 303kcal
30min

Cycling (Fast) 303kcal
30min

Cycling (Fast) 303kcal
30min

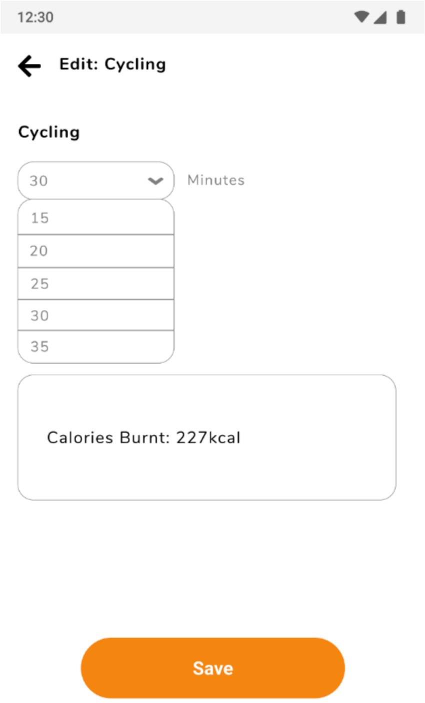
Cycling (Fast) 303kcal
30min

Figure D-17: Exercise Details screen



Calories Burnt: 227kcal

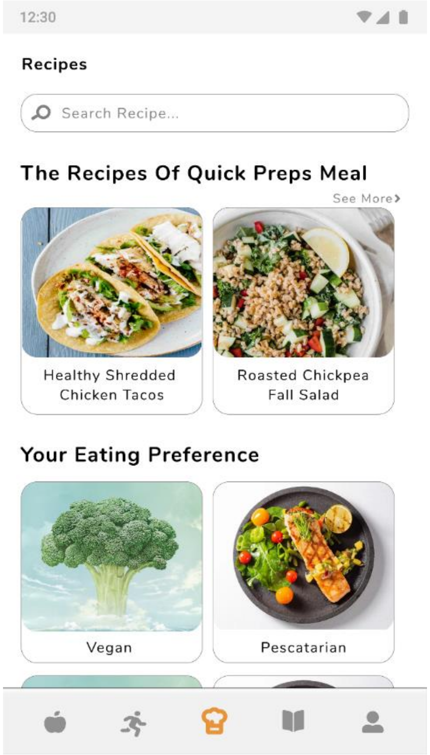
Figure D-18: Edit Exercise Log Record screen



Calories Burnt: 227kcal

Recipes

Figure D-19: Recipes screen



12:30

Recipes

Search Recipe...

The Recipes Of Quick Preps Meal

See More

Healthy Shredded Chicken Tacos

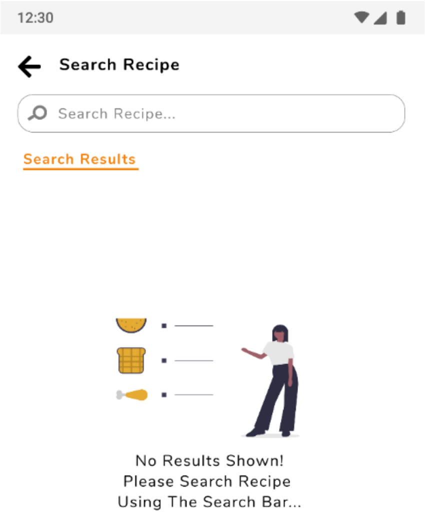
Roasted Chickpea Fall Salad

Your Eating Preference

Vegan

Pescatarian

Figure D-20: Search Recipe screen



12:30

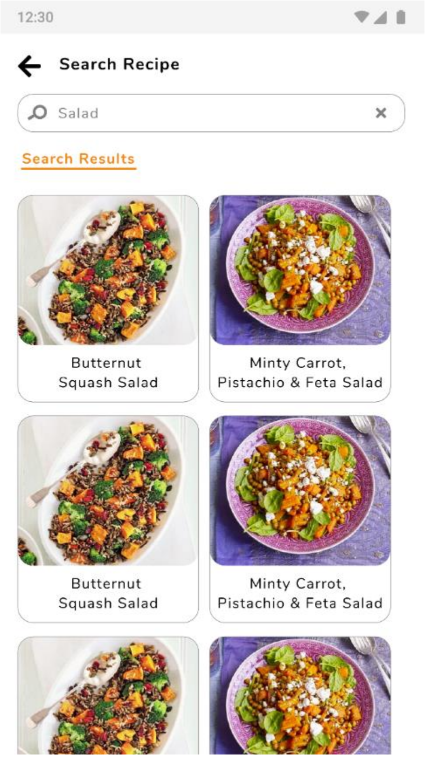
Search Recipe

Search Recipe...

Search Results

No Results Shown!
Please Search Recipe
Using The Search Bar...

Figure D-21: Search Recipe screen with results



12:30

Search Recipe

Salad

Search Results

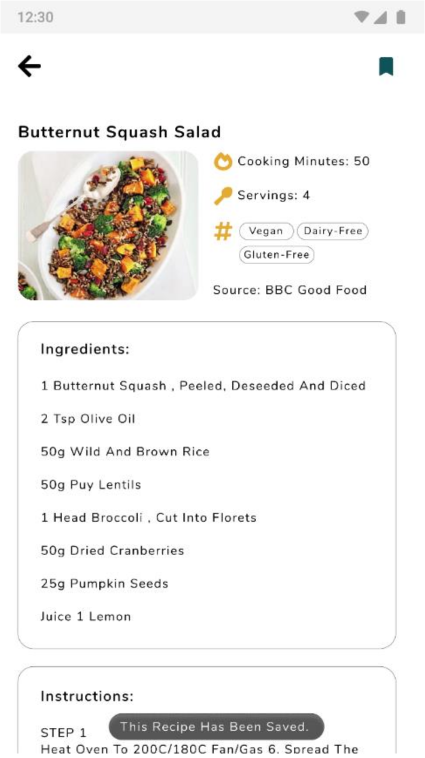
Butternut Squash Salad

Minty Carrot, Pistachio & Feta Salad

Butternut Squash Salad

Minty Carrot, Pistachio & Feta Salad

Figure D-22: First Half of the Recipe Details screen



12:30

Butternut Squash Salad

Cooking Minutes: 50

Servings: 4

Vegan Dairy-Free Gluten-Free

Source: BBC Good Food

Ingredients:

1 Butternut Squash , Peeled, Deseeded And Diced

2 Tsp Olive Oil

50g Wild And Brown Rice

50g Puy Lentils

1 Head Broccoli , Cut Into Florets

50g Dried Cranberries

25g Pumpkin Seeds

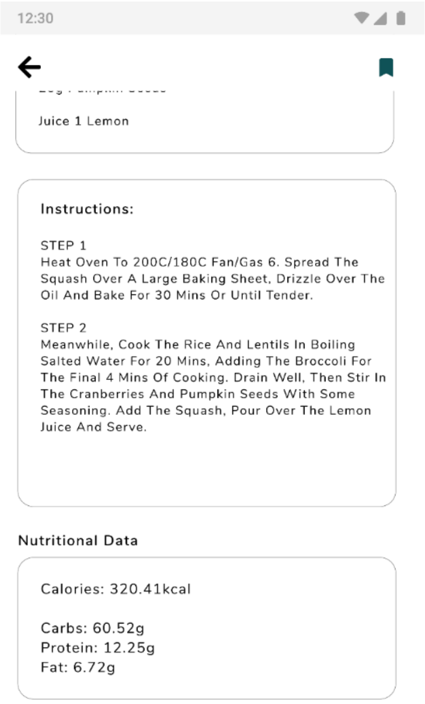
Juice 1 Lemon

Instructions:

STEP 1 This Recipe Has Been Saved.

Heat Oven To 200C/180C Fan/Gas 6. Spread The

Figure D-23: Second Half of the Recipe Details screen



Juice 1 Lemon

Instructions:

STEP 1
Heat Oven To 200C/180C Fan/Gas 6. Spread The Squash Over A Large Baking Sheet, Drizzle Over The Oil And Bake For 30 Mins Or Until Tender.

STEP 2
Meanwhile, Cook The Rice And Lentils In Boiling Salted Water For 20 Mins, Adding The Broccoli For The Final 4 Mins Of Cooking. Drain Well, Then Stir In The Cranberries And Pumpkin Seeds With Some Seasoning. Add The Squash, Pour Over The Lemon Juice And Serve.

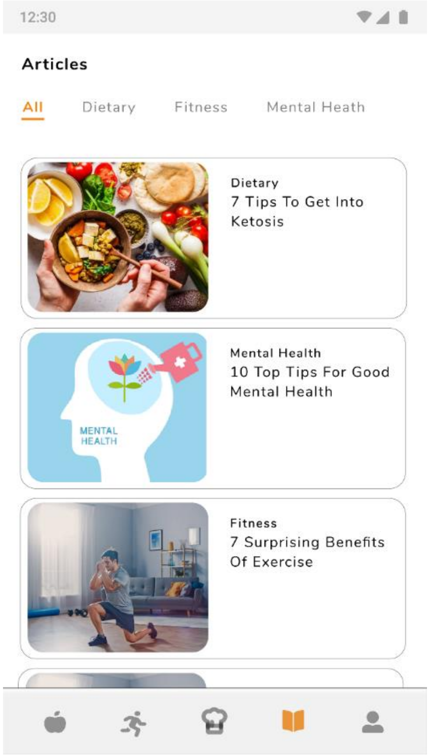
Nutritional Data

Calories: 320.41kcal

Carbs: 60.52g
Protein: 12.25g
Fat: 6.72g

Articles

Figure D-24: Articles screen

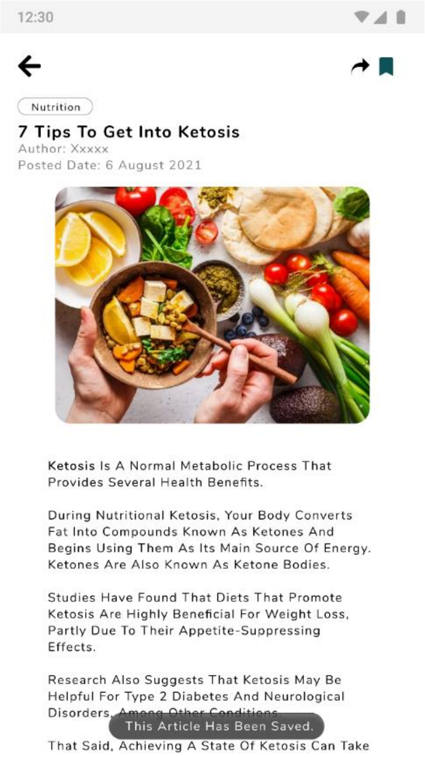


Dietary
7 Tips To Get Into Ketosis

Mental Health
10 Top Tips For Good Mental Health

Fitness
7 Surprising Benefits Of Exercise

Figure D-25: Article Contents screen



7 Tips To Get Into Ketosis

Author: Xxxxx

Posted Date: 6 August 2021

Ketosis Is A Normal Metabolic Process That Provides Several Health Benefits.

During Nutritional Ketosis, Your Body Converts Fat Into Compounds Known As Ketones And Begins Using Them As Its Main Source Of Energy. Ketones Are Also Known As Ketone Bodies.

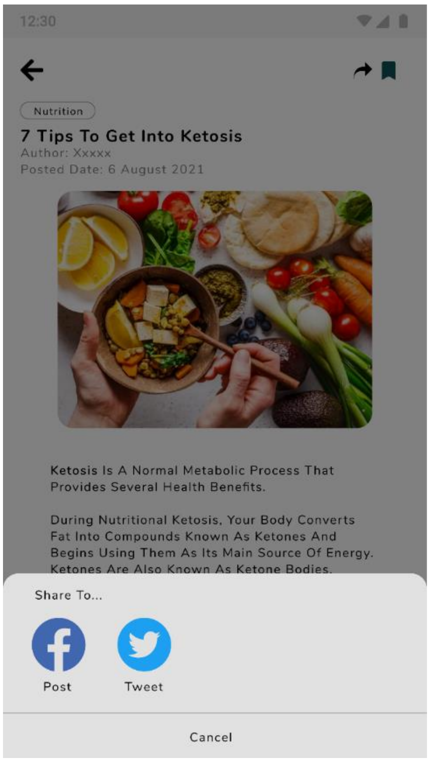
Studies Have Found That Diets That Promote Ketosis Are Highly Beneficial For Weight Loss, Partly Due To Their Appetite-Suppressing Effects.

Research Also Suggests That Ketosis May Be Helpful For Type 2 Diabetes And Neurological Disorders, Among Other Conditions.

This Article Has Been Saved.

That Said, Achieving A State Of Ketosis Can Take

Figure D-26: Share Article Pop Out Box



7 Tips To Get Into Ketosis

Author: Xxxxx

Posted Date: 6 August 2021

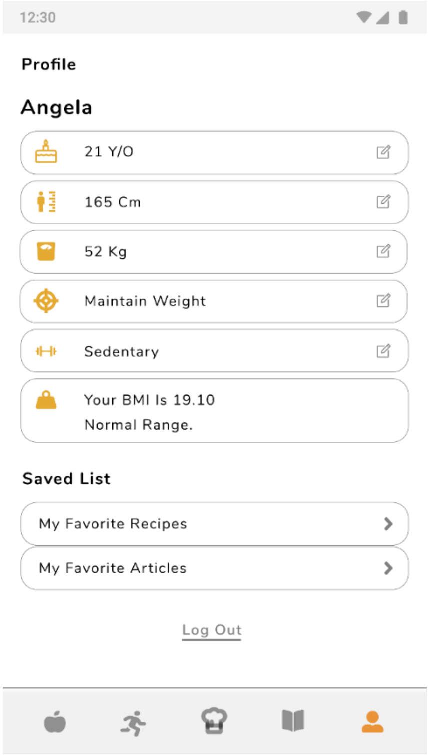
Ketosis Is A Normal Metabolic Process That Provides Several Health Benefits.

During Nutritional Ketosis, Your Body Converts Fat Into Compounds Known As Ketones And Begins Using Them As Its Main Source Of Energy. Ketones Are Also Known As Ketone Bodies.

Share To...

Personal Profile

Figure D-27: Personal Profile screen



12:30

Profile

Angela

21 Y/O

165 Cm

52 Kg

Maintain Weight

Sedentary

Your BMI Is 19.10
Normal Range.

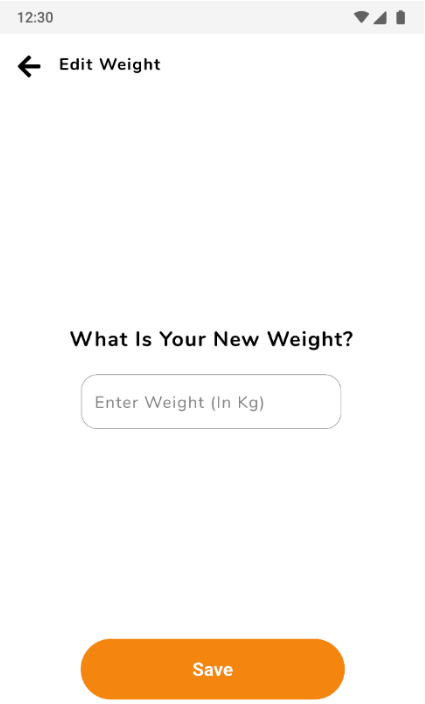
Saved List

My Favorite Recipes

My Favorite Articles

Log Out

Figure D-28: Edit Weight screen



12:30

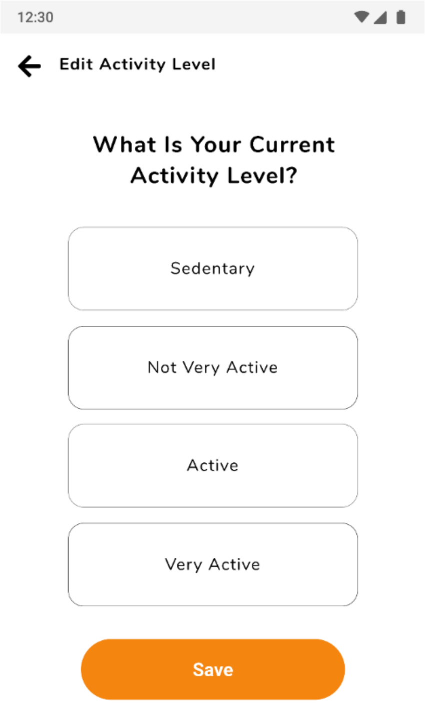
Edit Weight

What Is Your New Weight?

Enter Weight (In Kg)

Save

Figure D-29: Edit Activity Level screen



12:30

Edit Activity Level

What Is Your Current Activity Level?

Sedentary

Not Very Active

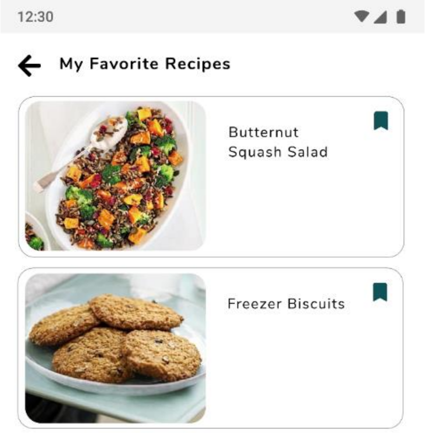
Active

Very Active

Save

Favorite Recipes List

Figure D-30: Favorite Recipes screen



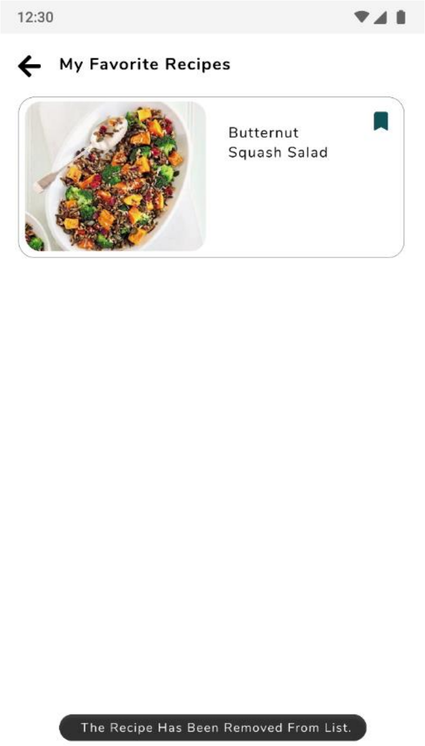
12:30

My Favorite Recipes

Butternut Squash Salad

Freezer Biscuits

Figure D-31: Favorite Recipes screen after removing recipe



12:30

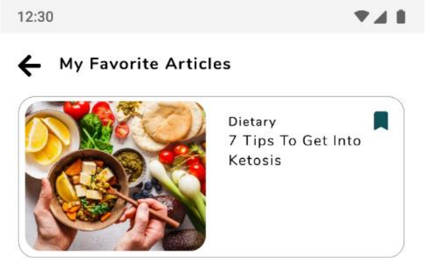
My Favorite Recipes

Butternut Squash Salad

The Recipe Has Been Removed From List.

Favorite Articles List

Figure D-32: Favorite Articles screen



12:30

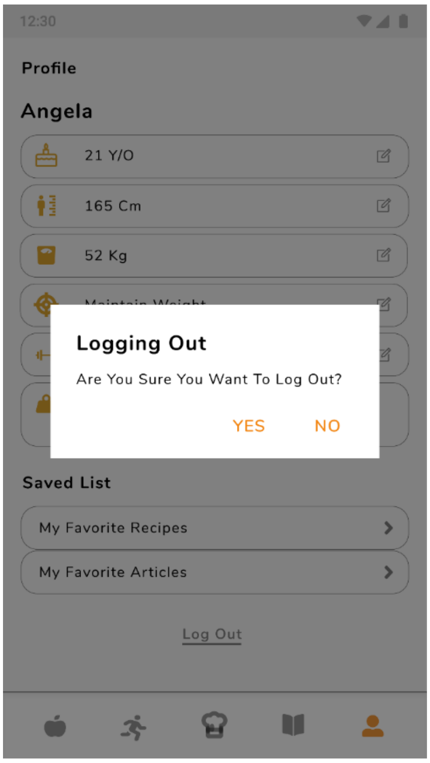
My Favorite Articles

Dietary

7 Tips To Get Into Ketosis

Log Out

Figure D-33: Log Out alert box



12:30

Profile

Angela

21 Y/O

165 Cm

52 Kg

Maintain Weight

Logging Out

Are You Sure You Want To Log Out?

YES NO

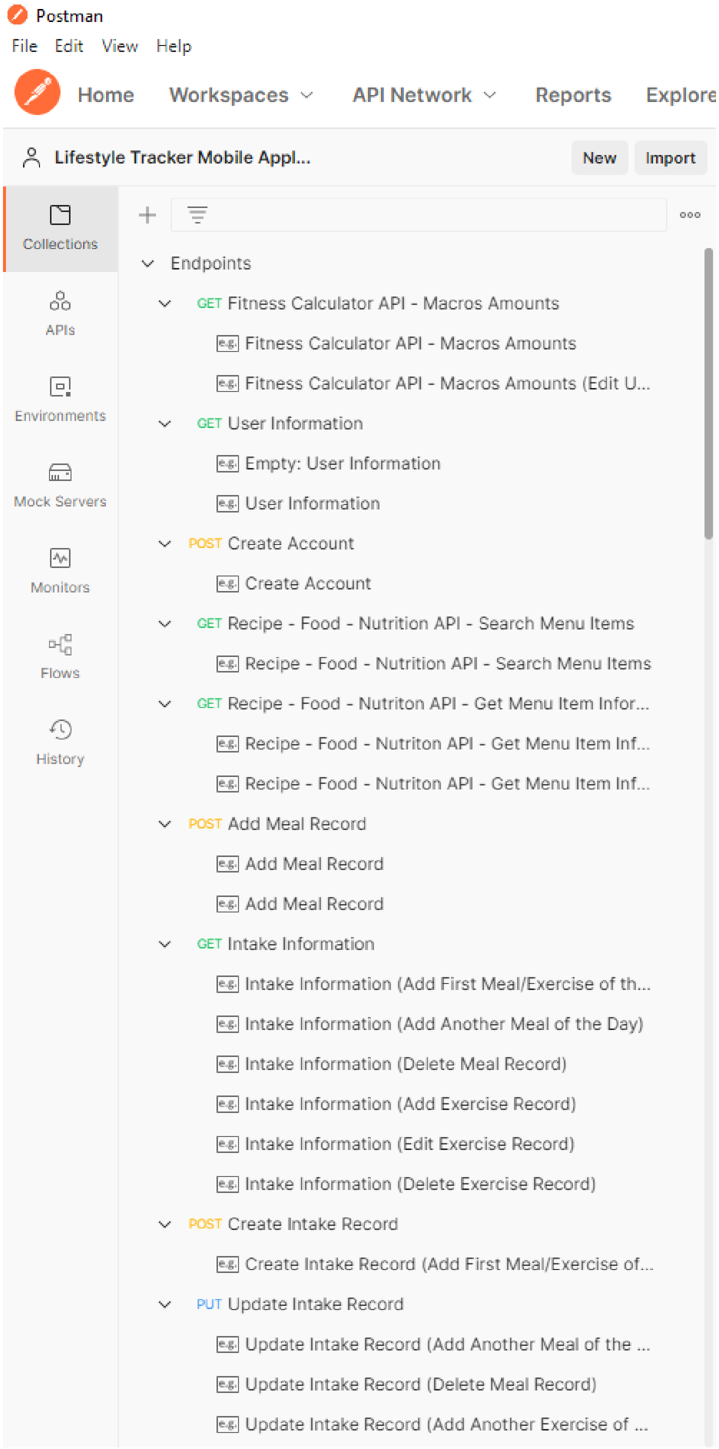
Saved List

My Favorite Recipes

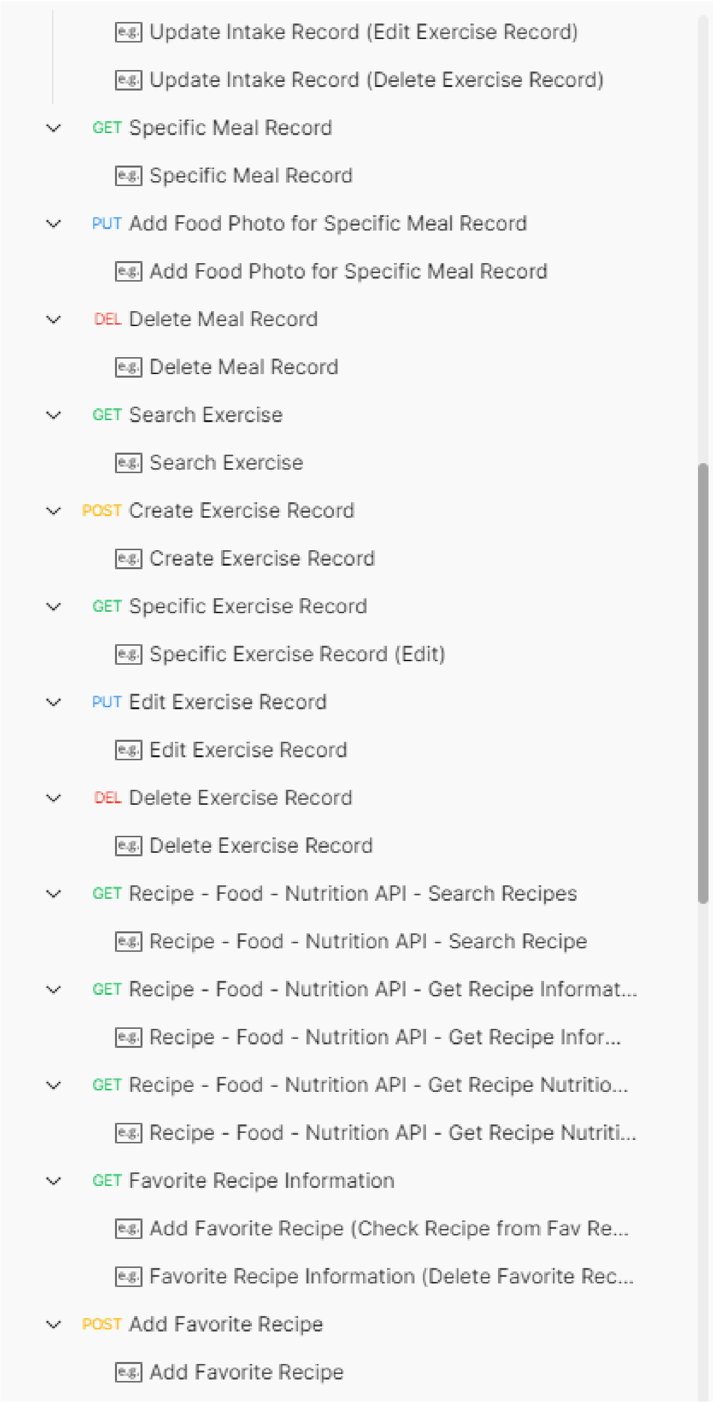
My Favorite Articles

Log Out

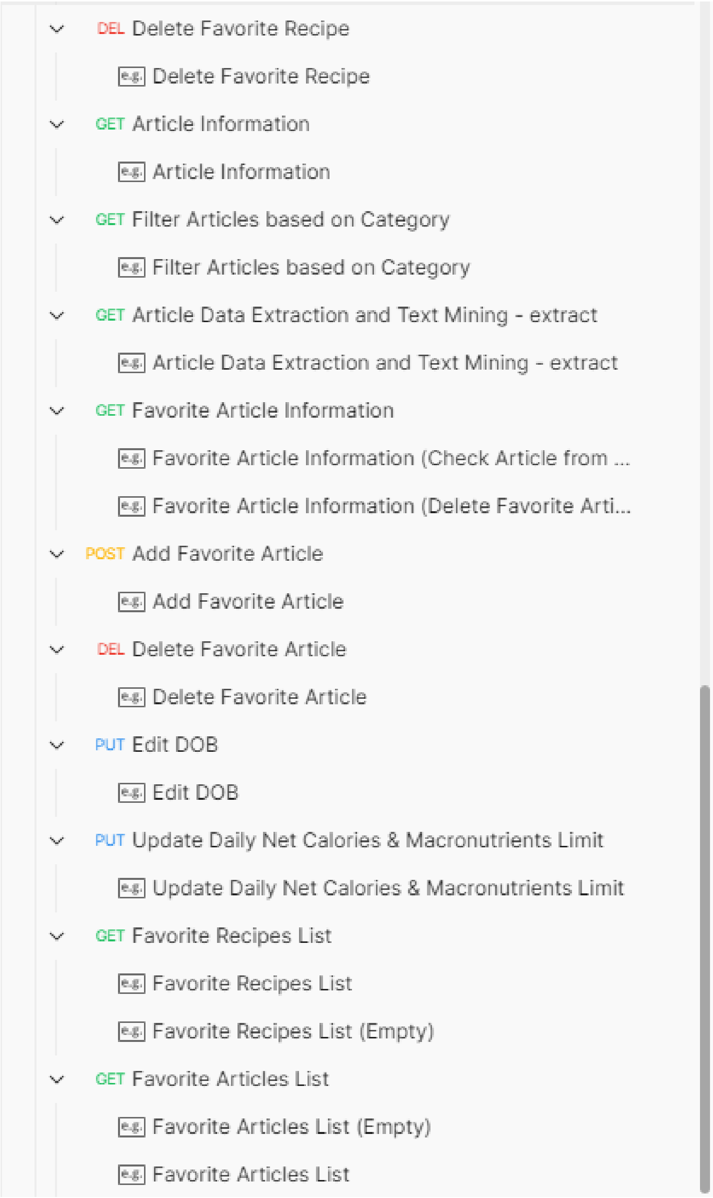
Appendix E: Summary of Web Service Testing in Postman



- Endpoints
  - GET Fitness Calculator API - Macros Amounts
    - e.g. Fitness Calculator API - Macros Amounts
    - e.g. Fitness Calculator API - Macros Amounts (Edit U...
  - GET User Information
    - e.g. Empty: User Information
    - e.g. User Information
  - POST Create Account
    - e.g. Create Account
  - GET Recipe - Food - Nutrition API - Search Menu Items
    - e.g. Recipe - Food - Nutrition API - Search Menu Items
  - GET Recipe - Food - Nutriton API - Get Menu Item Infor...
    - e.g. Recipe - Food - Nutriton API - Get Menu Item Inf...
    - e.g. Recipe - Food - Nutriton API - Get Menu Item Inf...
  - POST Add Meal Record
    - e.g. Add Meal Record
    - e.g. Add Meal Record
  - GET Intake Information
    - e.g. Intake Information (Add First Meal/Exercise of th...
    - e.g. Intake Information (Add Another Meal of the Day)
    - e.g. Intake Information (Delete Meal Record)
    - e.g. Intake Information (Add Exercise Record)
    - e.g. Intake Information (Edit Exercise Record)
    - e.g. Intake Information (Delete Exercise Record)
  - POST Create Intake Record
    - e.g. Create Intake Record (Add First Meal/Exercise of...
  - PUT Update Intake Record
    - e.g. Update Intake Record (Add Another Meal of the ...
    - e.g. Update Intake Record (Delete Meal Record)
    - e.g. Update Intake Record (Add Another Exercise of ...



- Update Intake Record (Edit Exercise Record)
- Update Intake Record (Delete Exercise Record)
- GET Specific Meal Record
  - Specific Meal Record
- PUT Add Food Photo for Specific Meal Record
  - Add Food Photo for Specific Meal Record
- DEL Delete Meal Record
  - Delete Meal Record
- GET Search Exercise
  - Search Exercise
- POST Create Exercise Record
  - Create Exercise Record
- GET Specific Exercise Record
  - Specific Exercise Record (Edit)
- PUT Edit Exercise Record
  - Edit Exercise Record
- DEL Delete Exercise Record
  - Delete Exercise Record
- GET Recipe - Food - Nutrition API - Search Recipes
  - Recipe - Food - Nutrition API - Search Recipe
- GET Recipe - Food - Nutrition API - Get Recipe Informat...
  - Recipe - Food - Nutrition API - Get Recipe Infor...
- GET Recipe - Food - Nutrition API - Get Recipe Nutritio...
  - Recipe - Food - Nutrition API - Get Recipe Nutriti...
- GET Favorite Recipe Information
  - Add Favorite Recipe (Check Recipe from Fav Re...
  - Favorite Recipe Information (Delete Favorite Rec...
- POST Add Favorite Recipe
  - Add Favorite Recipe



- DEL Delete Favorite Recipe
  - Delete Favorite Recipe
- GET Article Information
  - Article Information
- GET Filter Articles based on Category
  - Filter Articles based on Category
- GET Article Data Extraction and Text Mining - extract
  - Article Data Extraction and Text Mining - extract
- GET Favorite Article Information
  - Favorite Article Information (Check Article from ...
  - Favorite Article Information (Delete Favorite Arti...
- POST Add Favorite Article
  - Add Favorite Article
- DEL Delete Favorite Article
  - Delete Favorite Article
- PUT Edit DOB
  - Edit DOB
- PUT Update Daily Net Calories & Macronutrients Limit
  - Update Daily Net Calories & Macronutrients Limit
- GET Favorite Recipes List
  - Favorite Recipes List
  - Favorite Recipes List (Empty)
- GET Favorite Articles List
  - Favorite Articles List (Empty)
  - Favorite Articles List

Appendix F: User Satisfaction Survey Form

Participant’s name:

User Satisfaction Survey

	Strongly Disagree 1	**2**	**3**	**4**	**Strongly Agree 5**
1. I think that I would like to use this system for lifestyle tracking related matter.					
2. I found the system unnecessarily complex.					
3. I thought the system was easy to use					
4. I think that I would need the support of a technical person to be able to use this system.					
5. I found this system was easily moved through without a lot of backtracking or data re-entry.					
6. I thought there was too much inconsistency in this system.					

7. I would imagine that most people would learn to use this system very quickly.					
8. I found the system very awkward to use.					
9. I felt very confident using the system.					
10. I needed to learn a lot of things before I could get going with this system.					

What did you like best about the application?

What did you like least about the application?

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

Do you have any other final comments or questions?

Appendix G: User Satisfaction Survey Results

Participant’s name: Low Chen Wan

User Satisfaction Survey

	Strongly Disagree **1**	**2**	**3**	**4**	**Strongly Agree** **5**
1. I think that I would like to use this system for lifestyle tracking related matter.					✓
2. I found the system unnecessarily complex.	✓				
3. I thought the system was easy to use					✓
4. I think that I would need the support of a technical person to be able to use this system.	✓				
5. I found this system was easily moved through without a lot of backtracking or data re-entry.					✓
6. I thought there was too much inconsistency in this system.	✓				
7. I would imagine that most people would learn to use this system very quickly.					✓

8. I found the system very awkward to use.	✓				
9. I felt very confident using the system.					✓
10. I needed to learn a lot of things before I could get going with this system.	✓				

What did you like best about the application?

The calories input is recorded for the food I take in which helps the people to keep track of their calorie’s intake level per day.

What did you like least about the application?

the logout button is no easily to be find out at the profile screen.

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

Health is important, track your calories intake and calories burnt with this app helps you understand more on your body status.

Do you have any other final comments or questions?

Overall the app had provided the necessary functions/ features but I hope to see more interactive design of UI as it helps to motivate user to use this app more frequently and keep the user’s excitement on using it.

Participant’s name: Thian Qi Wee

User Satisfaction Survey

	Strongly Disagree 1	**2**	**3**	**4**	**Strongly Agree 5**
1. I think that I would like to use this system for lifestyle tracking related matter.				✓	
2. I found the system unnecessarily complex.	✓				
3. I thought the system was easy to use					✓
4. I think that I would need the support of a technical person to be able to use this system.	✓				
5. I found this system was easily moved through without a lot of backtracking or data re-entry.					✓
6. I thought there was too much inconsistency in this system.	✓				
7. I would imagine that most people would learn to use this system very quickly.					✓

8. I found the system very awkward to use.	✓				
9. I felt very confident using the system.					✓
10. I needed to learn a lot of things before I could get going with this system.	✓				

What did you like best about the application?

I like the photo uploading for the meal record which I can watch back what I have ate from the food log.

What did you like least about the application?

All good.

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

This is a very useful app for people who want to keep track of their diet. It is designed with a simple interface and easy to use even for the elderly.

Do you have any other final comments or questions?

No.

Participant’s name: Lee Yan

User Satisfaction Survey

	Strongly Disagree 1	**2**	**3**	**4**	**Strongly Agree 5**
1. I think that I would like to use this system for lifestyle tracking related matter.					✓
2. I found the system unnecessarily complex.	✓				
3. I thought the system was easy to use					✓
4. I think that I would need the support of a technical person to be able to use this system.	✓				
5. I found this system was easily moved through without a lot of backtracking or data re-entry.					✓
6. I thought there was too much inconsistency in this system.	✓				
7. I would imagine that most people would learn to use this system very quickly.					✓

8. I found the system very awkward to use.	✓				
9. I felt very confident using the system.					✓
10. I needed to learn a lot of things before I could get going with this system.	✓				

What did you like best about the application?

I can view a variety of recipes, articles and calories.

What did you like least about the application?

Nothing.

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

This is the best Lifestyle Tracker Mobile Application that covers a lot of functions.

Do you have any other final comments or questions?

No.

Participant’s name: Koo Xin Tong

User Satisfaction Survey

	Strongly Disagree 1	**2**	**3**	**4**	**Strongly Agree 5**
1. I think that I would like to use this system for lifestyle tracking related matter.				✓	
2. I found the system unnecessarily complex.		✓			
3. I thought the system was easy to use				✓	
4. I think that I would need the support of a technical person to be able to use this system.		✓			
5. I found this system was easily moved through without a lot of backtracking or data re-entry.				✓	
6. I thought there was too much inconsistency in this system.	✓				
7. I would imagine that most people would learn to use this system very quickly.					✓

8. I found the system very awkward to use.	✓				
9. I felt very confident using the system.				✓	
10. I needed to learn a lot of things before I could get going with this system.	✓				

What did you like best about the application?

The function of allowing user to save favourite articles for own future reference.

What did you like least about the application?

The recipe provides were not variety enough.

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

This is a simple but sufficient app for you to track your Calories and BMI.

Do you have any other final comments or questions?

I like the minimalist design of this app very much.

Participant’s name: Lim Teik Quan, Ian

User Satisfaction Survey

	Strongly Disagree 1	**2**	**3**	**4**	**Strongly Agree 5**
1. I think that I would like to use this system for lifestyle tracking related matter.				✓	
2. I found the system unnecessarily complex.		✓			
3. I thought the system was easy to use					✓
4. I think that I would need the support of a technical person to be able to use this system.	✓				
5. I found this system was easily moved through without a lot of backtracking or data re-entry.					✓
6. I thought there was too much inconsistency in this system.		✓			
7. I would imagine that most people would learn to use this system very quickly.				✓	

8. I found the system very awkward to use.	✓				
9. I felt very confident using the system.					✓
10. I needed to learn a lot of things before I could get going with this system.	✓				

What did you like best about the application?

The application is very user-friendly and easy to use, it also provides many features that many people would find useful in their day-to-day life.

What did you like least about the application?

The loading time for the article part is quite long, about 2 – 3 seconds.

If you were to describe this application to a friend/colleague in a sentence or two, what would you say?

This application helps you keep track of your daily food intake and also daily activity. Other than that, it also has a huge variety of food recipes and articles on how to achieve a healthy lifestyle.

Do you have any other final comments or questions?

It would be better if the application is able to suggest me articles and recipes based on my current BMI that is recorded in the system. Other than that, the application is already very feature-rich, so no extra comments.

Appendix H: User Acceptance Testing Form

User Acceptance Testing (UAT)						
Tester’s Name				Testing Start Date/Time		
				Testing End Date/Time		
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on “Get Started” button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level	User will successfully create an account and redirect to the meal tracker home screen		

			6. User fill in his/her nickname, email address and password 7. User clicks on “Create Account” button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on “Add Meal” button 6. User confirms the meal record creation	User will successfully add a meal record from either a list of meals or favorite recipes into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen		

			OR if the user already saved a recipe as favorite recipe: 1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User clicks on “Favorite Recipes” option from the top bar 4. User chooses the desired favorite recipe from the list to add it as meal record 5. User clicks on “Add Meal” button 6. User confirms the meal record creation			

UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either "Take photo" or "Choose from library" option to capture a food photo or select a food photo from photo gallery 3. User clicks on "Save" button to save the uploaded food photo	The food photo will be saved in the specific meal record for reference.		
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.		

UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Exercise” option 3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on “Add Exercise” button 7. User confirms the exercise record creation	User will successfully add an exercise record into his/her exercise log and the daily calories information will be updated and showed in the meal tracker and exercise tracker home screen		

UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by choosing the duration from the dropdown button 3. User clicks on “Save” button to save the updated duration of exercise activity performed	User will successfully edit the duration of the specific exercise activity performed. The daily calories information will be updated.		
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.		

UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the meal's calories and macronutrients		

			4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details			
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the recipe 2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on "My Favorite Recipes" options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon	User will successfully save the recipe. The recipe will be saved in his/her list of favorite recipes.		

UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles 3. User browses through a list of articles appeared 4. User chooses the desired article from the list of articles appeared 5. User views the article contents	User will successfully view the article contents		
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module	User will successfully save the article. The article will be saved in his/her list of favorite articles.		

			4. User clicks on "My Favorite Articles" options to view his/her list of favorite articles 5. User can unsave the article by clicking on the colored bookmark icon			
UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary		
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age),	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients		

			height, weight, goal of using this mobile application and his/her current activity level by clicking on the edit symbol button at the right side of the respective personal information 3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information	limit and BMI value will be updated according to the updated personal information		
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the		

			4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button 6. User relaunches the mobile application again	user has not logged in to his/her account yet. User will redirect to meal tracker module after logging in. User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		

Appendix I: Results of User Acceptance Testing

User Acceptance Testing (UAT)						
Tester's Name	Low Chen Wan			Testing Start Date/Time	24/3/2022 5:33p.m.	
				Testing End Date/Time	24/3/2022 5:58p.m.	
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on "Get Started" button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level	User will successfully create an account and redirect to the meal tracker home screen	Pass	Better to have some indication to tell user that they had done registered.

			6. User fill in his/her nickname, email address and password 7. User clicks on “Create Account” button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on “Add Meal” button 6. User confirms the meal record creation	User will successfully add a meal record into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-

UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either “Take photo” or “Choose from library” option to capture a food photo or select a food photo from photo gallery 3. User clicks on “Save” button to save the uploaded food photo	The food photo will be saved in the specific meal record for reference.	Pass	-
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.	Pass	-

UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Exercise” option 3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on “Add Exercise” button 7. User confirms the exercise record creation	User will successfully add an exercise record into his/her exercise log and the daily calories information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-

UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by choosing the duration from the dropdown button 3. User clicks on "Save" button to save the updated duration of exercise activity performed	User will successfully edit the duration of the specific exercise activity performed. The daily calories information will be updated.	Pass	-
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.	Pass	-

UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the meal's calories and macronutrients	Fail	When clicking on one of the recipe in the "Malaysian cuisine" category, the application does not respond and does not show the recipe information.

			4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details			
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the recipe 2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on “My Favorite Recipes” options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon	User will successfully save the recipe. The recipe will be saved in his/her list of favorite recipes.	Pass	-

UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles 3. User browses through a list of articles appeared 4. User chooses the desired article from the list of articles appeared 5. User views the article contents	User will successfully view the article contents	Pass	-
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module	User will successfully save the article. The article will be saved in his/her list of favorite articles.	Pass	-

			4. User clicks on “My Favorite Articles” options to view his/her list of favorite articles 5. User can unsave the article by clicking on the colored bookmark icon			
UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary	Pass	-
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age),	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients	Pass	-

			height, weight, goal of using this mobile application and his/her current activity level by clicking on the edit symbol button at the right side of the respective personal information 3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information	limit and BMI value will be updated according to the updated personal information		
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the	Pass	-

			4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button 6. User relaunches the mobile application again	user has not logged in to his/her account yet. User will redirect to meal tracker module after logging in. User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		

User Acceptance Testing (UAT)						
Tester's Name	Thian Qi Wee			Testing Start Date/Time	24/3/2022 6:08p.m.	
				Testing End Date/Time	24/3/2022 6:21p.m.	
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on "Get Started" button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level 6. User fill in his/her nickname, email address and password	User will successfully create an account and redirect to the meal tracker home screen	Pass	-

			7. User clicks on “Create Account” button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on “Add Meal” button 6. User confirms the meal record creation	User will successfully add a meal record into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-
UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either “Take photo” or “Choose from library”	The food photo will be saved in the specific meal record for reference.	Pass	-

			option to capture a food photo or select a food photo from photo gallery 3. User clicks on “Save” button to save the uploaded food photo			
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.	Pass	-
UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen	User will successfully add an exercise record into his/her exercise log and the daily calories information will be	Pass	-

			2. User chooses “Add Exercise” option 3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on “Add Exercise” button 7. User confirms the exercise record creation	updated and showed in the meal tracker and exercise tracker home screen		
UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by	User will successfully edit the duration of the specific exercise activity performed. The daily	Pass	-

			choosing the duration from the dropdown button 3. User clicks on "Save" button to save the updated duration of exercise activity performed	calories information will be updated.		
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.	Pass	-
UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the	Pass	-

			4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details	meal's calories and macronutrients		
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right	User will successfully save the recipe. The recipe will be saved in	Pass	-

			corner of the screen to save the recipe 2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on "My Favorite Recipes" options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon	his/her list of favorite recipes.		
UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles	User will successfully view the article contents	Pass	-

			3. User browses through a list of articles appeared 4. User chooses the desired article from the list of articles appeared 5. User views the article contents			
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module 4. User clicks on “My Favorite Articles” options to view his/her list of favorite articles	User will successfully save the article. The article will be saved in his/her list of favorite articles.	Pass	-

			5. User can unsave the article by clicking on the colored bookmark icon			
UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary	Pass	-
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age), height, weight, goal of using this mobile application and his/her current activity level by clicking on	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients limit and BMI value will be updated according to	Pass	-

			the edit symbol button at the right side of the respective personal information 3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information	the updated personal information		
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application 4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the user has not logged in to his/her account yet.	Pass	-

			6. User relaunches the mobile application again	User will redirect to meal tracker module after logging in. User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		

User Acceptance Testing (UAT)						
Tester's Name	Lee Yan			Testing Start Date/Time	25/3/2022 10:51a.m.	
				Testing End Date/Time	25/3/2022 10:59a.m.	
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on "Get Started" button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level 6. User fill in his/her nickname, email address and password	User will successfully create an account and redirect to the meal tracker home screen	Pass	-

			7. User clicks on “Create Account” button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on “Add Meal” button 6. User confirms the meal record creation OR if the user already saved a recipe as favorite recipe:	User will successfully add a meal record from either a list of meals or favorite recipes into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-

			1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Meal” option 3. User clicks on “Favorite Recipes” option from the top bar 4. User chooses the desired favorite recipe from the list to add it as meal record 5. User clicks on “Add Meal” button 6. User confirms the meal record creation			
UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either “Take photo” or “Choose from library” option to capture a food photo or	The food photo will be saved in the specific meal record for reference.	Pass	-

			select a food photo from photo gallery 3. User clicks on “Save” button to save the uploaded food photo			
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.	Pass	-
UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Exercise” option	User will successfully add an exercise record into his/her exercise log and the daily calories information will be updated and showed in	Pass	-

			3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on "Add Exercise" button 7. User confirms the exercise record creation	the meal tracker and exercise tracker home screen		
UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by choosing the duration from the dropdown button	User will successfully edit the duration of the specific exercise activity performed. The daily calories information will be updated.	Pass	-

			3. User clicks on “Save” button to save the updated duration of exercise activity performed			
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.	Pass	-
UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key 4. User chooses the desired recipe from the list of recipes appeared	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the meal’s calories and macronutrients	Pass	-

			5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details			
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the recipe	User will successfully save the recipe. The recipe will be saved in his/her list of favorite recipes.	Pass	-

			2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on “My Favorite Recipes” options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon			
UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles 3. User browses through a list of articles appeared	User will successfully view the article contents	Pass	-

			4. User chooses the desired article from the list of articles appeared 5. User views the article contents			
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module 4. User clicks on “My Favorite Articles” options to view his/her list of favorite articles 5. User can unsave the article by clicking on the colored bookmark icon	User will successfully save the article. The article will be saved in his/her list of favorite articles.	Pass	-

UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary	Pass	-
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age), height, weight, goal of using this mobile application and his/her current activity level by clicking on the edit symbol button at the right side of the respective personal information	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients limit and BMI value will be updated according to the updated personal information	Pass	-

			3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information			
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application 4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button 6. User relaunches the mobile application again	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the user has not logged in to his/her account yet. User will redirect to meal tracker module after logging in.	Pass	-

				User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		

User Acceptance Testing (UAT)						
Tester's Name	Koo Xin Tong			Testing Start Date/Time	26/3/2022 11:41a.m.	
				Testing End Date/Time	26/3/2022 12:04p.m.	
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on "Get Started" button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level 6. User fill in his/her nickname, email address and password	User will successfully create an account and redirect to the meal tracker home screen	Pass	For, No. 5, I don't quite get the activity level at first

			7. User clicks on "Create Account" button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses "Add Meal" option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on "Add Meal" button 6. User confirms the meal record creation OR if the user already saved a recipe as favorite recipe:	User will successfully add a meal record from either a list of meals or favorite recipes into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-

			1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses "Add Meal" option 3. User clicks on "Favorite Recipes" option from the top bar 4. User chooses the desired favorite recipe from the list to add it as meal record 5. User clicks on "Add Meal" button 6. User confirms the meal record creation			
UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either "Take photo" or "Choose from library" option to capture a food photo or	The food photo will be saved in the specific meal record for reference.	Pass	Perhaps can specify where the meal log is in my opinion, I was lost to the No. 1 instruction at first.

			select a food photo from photo gallery 3. User clicks on “Save” button to save the uploaded food photo			
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.	Pass	-
UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses “Add Exercise” option	User will successfully add an exercise record into his/her exercise log and the daily calories information will be updated and showed in	Pass	-

			3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on “Add Exercise” button 7. User confirms the exercise record creation	the meal tracker and exercise tracker home screen		
UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by choosing the duration from the dropdown button	User will successfully edit the duration of the specific exercise activity performed. The daily calories information will be updated.	Pass	-

			3. User clicks on “Save” button to save the updated duration of exercise activity performed			
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.	Pass	-
UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key 4. User chooses the desired recipe from the list of recipes appeared	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the meal’s calories and macronutrients	Pass	-

			5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details			
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the recipe	User will successfully save the recipe. The recipe will be saved in his/her list of favorite recipes.	Pass	-

			2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on “My Favorite Recipes” options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon			
UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles 3. User browses through a list of articles appeared	User will successfully view the article contents	Pass	-

			4. User chooses the desired article from the list of articles appeared 5. User views the article contents			
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module 4. User clicks on "My Favorite Articles" options to view his/her list of favorite articles 5. User can unsave the article by clicking on the colored bookmark icon	User will successfully save the article. The article will be saved in his/her list of favorite articles.	Pass	Although I had passed this test case according to the test steps given, but I did discover an error when I wished to unsave the article from the article contents screen when the previous screen was the favorite article list.

UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary	Pass	-
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age), height, weight, goal of using this mobile application and his/her current activity level by clicking on the edit symbol button at the right side of the respective personal information	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients limit and BMI value will be updated according to the updated personal information	Pass	-

			3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information			
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application 4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button 6. User relaunches the mobile application again	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the user has not logged in to his/her account yet. User will redirect to meal tracker module after logging in.	Pass	-

				User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		

User Acceptance Testing (UAT)						
Tester’s Name	Lim Teik Quan, Ian			Testing Start Date/Time	29/3/2022 5:59p.m.	
				Testing End Date/Time	29/3/2022 6:08p.m.	
Test Case ID	Module	Test Case Title	Test Steps	Expected Results	Status (Pass/Fail)	Comments
UAT001	Sign Up	Register an account	1. User clicks on “Get Started” button to proceed to the sign up process 2. User chooses his/her primary goal of using this mobile application 3. User fills in his/her gender and date of birth 4. User fills in his/her weight and height 5. User chooses his/her current activity level 6. User fill in his/her nickname, email address and password	User will successfully create an account and redirect to the meal tracker home screen	Pass	-

			7. User clicks on "Create Account" button			
UAT002	Meal Tracker	Add meal record into meal log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen 2. User chooses "Add Meal" option 3. User inputs the meal keyword in the search bar and press Enter key 4. User chooses the meal from the list of meals appeared 5. User clicks on "Add Meal" button 6. User confirms the meal record creation	User will successfully add a meal record into his/her meal log and the daily calories and macronutrients information will be updated and showed in the meal tracker and exercise tracker home screen	Pass	-
UAT003	Meal Tracker	Upload food photo for the meal record	1. User chooses the specific meal record from his/her meal log 2. User clicks on either "Take photo" or "Choose from library"	The food photo will be saved in the specific meal record for reference.	Fail	The specific food photo selected from the library was not

			option to capture a food photo or select a food photo from photo gallery 3. User clicks on “Save” button to save the uploaded food photo			shown in the meal record screen.
UAT004	Meal Tracker	Delete meal record from meal log	1. User chooses the specific meal record from his/her meal log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the meal record 3. User confirms the meal record deletion	User will redirect to the meal tracker home screen after deleting the specific meal record. The daily macronutrients information will be updated.	Pass	-
UAT005	Exercise Tracker	Add exercise record into exercise log	1. User clicks on the add symbol floating button at the bottom right corner of the meal tracker or exercise tracker home screen	User will successfully add an exercise record into his/her exercise log and the daily calories information will be	Pass	-

			2. User chooses "Add Exercise" option 3. User inputs the exercise keyword in the search bar and press Enter key 4. User chooses the exercise from the list of exercise activities appeared 5. User chooses the duration of exercise activity performed 6. User clicks on "Add Exercise" button 7. User confirms the exercise record creation	updated and showed in the meal tracker and exercise tracker home screen		
UAT006	Exercise Tracker	Edit the exercise record	1. User chooses the specific exercise record from his/her exercise log 2. User updates the duration of exercise activity performed by	User will successfully edit the duration of the specific exercise activity performed. The daily	Pass	-

			choosing the duration from the dropdown button 3. User clicks on “Save” button to save the updated duration of exercise activity performed	calories information will be updated.		
UAT007	Exercise Tracker	Delete exercise record from exercise log	1. User chooses the specific exercise record from his/her exercise log 2. User clicks on the dustbin symbol floating button at the bottom right corner of the screen to delete the exercise record 3. User confirms the exercise record deletion	User will redirect to the exercise tracker home screen after deleting the specific exercise record. The daily calories information will be updated.	Pass	-
UAT008	Recipes	Search and view recipe	1. User navigates to the recipes module 2. User clicks on the Search bar 3. User inputs the recipe keyword in the search bar and press Enter key	User will successfully view the recipe details such as the steps to prepare the meal, ingredients and the	Pass	-

			4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details OR 1. User navigates to the recipes module 2. User chooses the desired category of recipes 3. User browses through a list of recipes appeared 4. User chooses the desired recipe from the list of recipes appeared 5. User views the recipe details	meal’s calories and macronutrients		
UAT009	Recipes	Save recipe	1. User clicks on the outlined bookmark icon at the top right	User will successfully save the recipe. The recipe will be saved in	Pass	-

			corner of the screen to save the recipe 2. A toast message shows up to notify the user that he/she has successfully saved the recipe 3. User navigates to the personal profile module 4. User clicks on “My Favorite Recipes” options to view his/her list of favorite recipes 5. User can unsave the recipe by clicking on the colored bookmark icon	his/her list of favorite recipes.		
UAT010	Articles	View article	1. User navigates to the articles module 2. User chooses the desired category of articles	User will successfully view the article contents	Pass	-

			3. User browses through a list of articles appeared 4. User chooses the desired article from the list of articles appeared 5. User views the article contents			
UAT011	Articles	Save article	1. User clicks on the outlined bookmark icon at the top right corner of the screen to save the article 2. A toast message shows up to notify the user that he/she has successfully saved the article 3. User navigates to the personal profile module 4. User clicks on “My Favorite Articles” options to view his/her list of favorite articles	User will successfully save the article. The article will be saved in his/her list of favorite articles.	Pass	-

			5. User can unsave the article by clicking on the colored bookmark icon			
UAT012	Articles	Share article	1. User views the article contents 2. User clicks on the share symbol button at the top right corner of the screen to share the article to his/her social media 3. User redirects to his/her social media to share the article summary	User will successfully be brought to his/her social media and share the article summary	Pass	-
UAT013	Personal Profile	Update personal information	1. User navigates to the personal profile module 2. User chooses the personal information that he/she would like to update such as date of birth (age), height, weight, goal of using this mobile application and his/her current activity level by clicking on	User will successfully update his/her specific personal information. The respective details such as the daily macronutrients limit and BMI value will be updated according to	Pass	-

			the edit symbol button at the right side of the respective personal information 3. User edits the specific personal information 4. User clicks on “Save” button to save the updated personal information	the updated personal information		
UAT014	Sign In	Log in and log out account	1. User clicks on “Log Out” button at the bottom of the personal profile home screen 2. User confirms the log out action 3. User relaunches the mobile application 4. User fills in his/her email address and password at the sign in screen 5. User clicks on “Login” button	User will be brought to sign in screen after logging out. User will remain in the sign in screen after relaunching the application because the user has not logged in to his/her account yet.	Pass	-

			6. User relaunches the mobile application again	User will redirect to meal tracker module after logging in. User will remain in the meal tracker home screen after relaunching the application because the user has logged in to his/her account already.		