

HEALTHY SMART BAKING RECIPES MOBILE
APPLICATION

TAN YI TIANG

BACHELOR OF COMPUTER SCIENCE
(COMPUTER SYSTEM & NETWORKING) WITH
HONOURS

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : TAN YI TIANG

Date of Birth :

Title : HEALTHY SMART BAKING RECIPES MOBILE
APPLICATION

Academic Session : 2022/2023

I declare that this thesis is classified as:

- CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*
- RESTRICTED (Contains restricted information as specified by the organization where research was done)*
- OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

(Supervisor's Signature)

Nur Shazwani Binti Kamarudin

New IC/Passport Number
Date: 19 January 2023

Name of Supervisor
Date: 19 January 2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

Author's Name

Thesis Title

Reasons (i)

(ii)

(iii)

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Thank you.

Yours faithfully,



(Supervisor's Signature)

Date: 19 January 2023

Stamp:

DR. NUR SHAZWANI KAMARUDIN
PENSYARAH KANAN
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG.
TEL : 09-424 4736

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.



SUPERVISOR'S DECLARATION

I hereby declare that I have checked this project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Science in Computer Systems and Networking.

A handwritten signature in black ink, appearing to be 'Nur Shazwani binti Kamarudin', written over a horizontal line.

(Supervisor's Signature)

Full Name : Nur Shazwani binti Kamarudin

Position : Senior Lecturer

Date : 19 January 2023

(Co-supervisor's Signature)

Full Name :

Position :

Date :



STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

A handwritten signature in black ink, appearing to read 'Tan Yi Tiang', is written above a horizontal line.

(Student's Signature)

Full Name : TAN YI TIANG

ID Number : CA19073

Date : 19 January 2023

HEALTHY SMART BAKING RECIPES MOBILE APPLICATION

TAN YI TIANG

Thesis submitted in fulfillment of the requirements

for the award of the degree of

Bachelor of Computer Science in Computer Systems and Networking

Faculty of Computer System and Software Engineering

UNIVERSITI MALAYSIA PAHANG

JANUARY 2023

ACKNOWLEDGEMENTS

First, I would like to thank my supervisor, Dr. Nur Shazwani Binti Kamrudin, who has been very responsible and providing me a lot of advice. She has been guiding me in the right path and has always been open to my ideas.

Next, I would like to thank to my family and friends who always give me support and share their knowledge with me in this period. Moreover, I would like to thanks to the end user who filled up the survey form and provide suggestions on my application development.

Last but not the least, I would like to express my gratitude to the final year project coordinator, Dr. Danakorn Nincarean in assisting and provide us necessary information to complete the project.

ABSTRAK

Membuat roti atau kek adalah cara yang baik untuk orang ramai untuk meluahkan kreativiti mereka. Ia bukan sekadar berfikir pada masa tersebut atau mengikuti butiran, ia juga tentang memikirkan resipi, hidangan, bagaimana menggunakannya, dengan siapa anda berhasrat untuk membaginya, dan juga waktunya. Oleh itu, ia adalah cara yang baik untuk mengekalkan keseimbangan ini antara masa dan gambaran yang lebih besar. Walau bagaimanapun, gaya hidup yang sibuk menjadikannya sukar bagi kebanyakan orang untuk belajar membuat roti atau kek dan menghargai prosesnya. Dalam projek ini, kami akan memberi tumpuan kepada membangunkan aplikasi mudah alih resipi baking yang terutamanya untuk pengguna yang tidak mempunyai pengalaman dalam baking untuk mempelajari resipi mudah. Tujuan projek ini adalah untuk mengkaji kesukaran dan batasan yang dihadapi oleh orang yang mengalami masalah kesihatan dan pengguna yang tidak berpengalaman ketika mempelajari resipi serta membangunkan aplikasi mudah alih yang mendigitalkan resipi dan membawa kemudahan kepada pengguna. Skop projek ini adalah untuk memberikan butiran resipi, senarai ramuan, fakta pemakanan dan fungsi komunikasi.

ABSTRACT

Baking is a great way for people to express their creativity. It isn't just about thinking in the moment and following details, it's also about thinking about the recipe, the dish, how you intend to use it, with whom you intend to share it, and at what time. Hence, it is a great way to maintain this balance between the moment and the bigger picture. However, a hectic lifestyle makes it difficult for most people to study baking and appreciate its process. In this paper, we will focus on developing a baking recipes mobile application that is mainly for the user who has no experiences in baking to learn simple recipes. The aim of this project is to study the difficulties and limitation faced by people with suffer health problems and non-experienced users when learning new baking recipes and to develop a mobile application which digitalized baking recipes and bring convenience to users. The scope of this project is to provide recipe details, ingredient list, nutritional facts and communication function.

TABLE OF CONTENT

DECLARATION	
TITLE PAGE	
ACKNOWLEDGEMENTS	ii
ABSTRAK	iii
ABSTRACT	iv
TABLE OF CONTENT	v
LIST OF TABLES	ix
LIST OF FIGURES	xi
CHAPTER 1 INTRODUCTION	1
1.1 Introduction	1
1.2 Problem Statement	3
1.3 Objective	4
1.4 Scope	4
1.5 Thesis Organization	5
CHAPTER 2 LITERATURE REVIEW	6
2.1 Introduction	6
2.2 Three Related Work	7
2.2.1 ALL BAKING RECIPES	7
2.2.2 Baking Recipes	9

2.2.3	Tasty	11
2.3	Comparative Analysis	13
2.3.1	Specification of existing system	13
2.3.2	Comparison of three existing system	14
2.3.3	Advantages and Disadvantages of the three existing system	15
2.4	Chapter Summary	16
CHAPTER 3 METHODOLOGY		17
3.1	Introduction	17
3.2	Methodology	17
3.3	Project Requirement	20
3.4	User Requirement	22
3.5	Proposed Design	30
3.5.1	Flowchart	30
3.5.2	Context Diagram	37
3.5.3	Use Case Diagram and Description	38
3.5.4	Acticity Diagram	53
3.5.5	Storyboard	54
3.6	Data Design	69
3.7	Design Prototype	73
3.8	Testing Plan	115
3.9	Potential Use of Proposed Solution	117
3.10	Hardware and Software Specification	117
3.10.1	Hardware Specification	117
3.10.2	Software Specification	118

CHAPTER 4 RESULT AND DISCUSSION	119
4.1 Introduction	119
4.2 Development of Application	119
4.2.1 Development Tools	119
4.2.2 Create and Development Project	120
4.2.3 Identity Module	122
4.2.4 Dietitian Login Module	124
4.2.5 User Login Module	126
4.2.6 User and Dietitian Registration Module	128
4.2.7 Forgot Password Page	130
4.2.8 User Menu Page	131
4.2.9 Dietitian Menu Page	133
4.2.10 User Profile Page	135
4.2.11 Dietitian Profile Page	137
4.2.12 Manage User Profile Page	139
4.2.13 Manage Dietitian Profile Page	141
4.2.14 View Recipe Page	143
4.2.15 Search Recipe	145
4.2.16 Recipe Details Page	147
4.2.17 Calculate BMI Page	149
4.2.18 Dietitian Edit Recipe Page	151
4.2.19 Dietitian Upload Page	153
4.2.20 Dietitian Edit Recipe Details Page	155
4.2.21 Dietitian Update Page	157
4.2.22 Chat Name List Page	159
4.2.23 Chat Page	161

4.3	Testing	163
4.3.1	User Acceptance Test	163
4.3.2	Result and Discussion	166
4.4	Chapter Summary	170
CHAPTER 5 CONCLUSION		171
5.1	Objective Revisited	171
5.2	Limitation	171
5.3	Future Work	172
REFERENCES		173
APPENDIX A SAMPLE APPENDIX 1		175
APPENDIX B USER ACCEPTANCE TEST		177

LIST OF TABLES

Table 2.3.1	Specification of existing system	13
Table 2.3.2	Comparison of three existing system	14
Table 2.3.3	Advantages and Disadvantages of the three existing system	15
Table 3.5.1	Use Case Description of Account Registration	39
Table 3.5.2	Use Case Description of Account Login	41
Table 3.5.3	Use Case Description of Edit User Profile	43
Table 3.5.4	Use Case Description of Search Recipes	45
Table 3.5.5	Use Case Description of Select Recipes	46
Table 3.5.6	Use Case Description of View Recipes Details	47
Table 3.5.7	Use Case Description of Calculate BMI	48
Table 3.5.8	Use Case Description of Chat	49
Table 3.5.9	Use Case Description of Add Recipes	50
Table 3.5.10	Use Case Description of Edit Recipes	51
Table 3.5.11	Use Case Description of Delete Recipes	52
Table 3.6.1	User Data Dictionary	70
Table 3.6.2	Recipe Data Dictionary	70
Table 3.6.3	Dietitian Data Dictionary	71
Table 3.6.4	Chat Data Dictionary	72
Table 3.7.1	Identity Page Interface	73
Table 3.7.2	User Login Page Interface	75
Table 3.7.3	Dietitian Login Page Interface	77
Table 3.7.4	User Registration Page Interface	79
Table 3.7.5	Dietitian Registration Page Interface	81
Table 3.7.6	Forgot Password Page Interface	83

Table 3.7.7	User Menu Page Interface	85
Table 3.7.8	Dietitian Menu Page Interface	87
Table 3.7.9	User Profile Page Interface	89
Table 3.7.10	Dietitian Profile Page Interface	91
Table 3.7.11	Manage User Profile Page Interface	93
Table 3.7.12	Manage Dietitian Profile Page Interface	95
Table 3.7.13	View Recipe Page Interface	97
Table 3.7.14	Recipe Details Page Interface	99
Table 3.7.15	Calculate BMI Page Interface	101
Table 3.7.16	Dietitian Edit Recipe Page Interface	103
Table 3.7.17	Dietitian Upload Page Interface	105
Table 3.7.18	Dietitian Edit Recipe Details Interface	107
Table 3.7.19	Dietitian Update Page Interface	109
Table 3.7.20	Chat Name List Page Interface	111
Table 3.7.21	Chat Page Interface	113
Table 3.8	Testing Plan	115
Table 3.10.1	Hardware Specification	117
Table 3.10.2	Software Specification	118

LIST OF FIGURES

Figure 2.2.1	ALL BAKING RECIPES app interface	8
Figure 2.2.2	Baking Recipes app interface	10
Figure 2.2.3	Tasty app interface	12
Figure 3.1	AGILE development Model	18
Figure 3.4.1	User Requirement Question 1	22
Figure 3.4.2	User Requirement Question 2	23
Figure 3.4.3	User Requirement Question 3	24
Figure 3.4.4	User Requirement Question 4	25
Figure 3.4.5	User Requirement Question 5	26
Figure 3.4.6	User Requirement Question 6	27
Figure 3.4.7	User Requirement Question 7	28
Figure 3.4.8	User Requirement Question 8	29
Figure 3.5.1	Flowchart For Login	30
Figure 3.5.2	Flowchart For Registration	31
Figure 3.5.3	Flowchart of User Activity	32
Figure 3.5.4	Flowchart of Dietitian Activity	34
Figure 3.5.5	Whole Flowchart	36
Figure 3.5.6	Context Diagram	37
Figure 3.5.7	Use Case Diagram	38
Figure 3.5.8	Activity Diagram	53
Figure 3.5.9	Choose Identity Interface	54
Figure 3.5.10	Login Interface	55
Figure 3.5.11	User and Dietitian Registration Interface	56
Figure 3.5.12	Forgot Password Interface	57

Figure 3.5.13	User Menu Page Interface	58
Figure 3.5.14	Dietitian Menu Page Interface	59
Figure 3.5.15	Manage User Profile Interface	60
Figure 3.5.16	Manage Dietitian Profile Interface	61
Figure 3.5.17	View Recipe Page Interface	62
Figure 3.5.18	Search Recipes Interface	63
Figure 3.5.19	Recipe Details Page Interface	64
Figure 3.5.20	Calculate BMI Page Interface	65
Figure 3.5.21	Add, Edit and Delete Recipes Interface	66
Figure 3.5.22	Dietitian Upload and Update Recipe Interface	67
Figure 3.5.23	Chat Page Interface	68
Figure 3.6.1	ERD	69
Figure 3.7.1	Identity Page Interface	73
Figure 3.7.2	User Login Page Interface	75
Figure 3.7.3	Dietitian Login Page Interface	77
Figure 3.7.4	User Registration Page Interface	79
Figure 3.7.5	Dietitian Registration Page Interface	81
Figure 3.7.6	Forgot Password Page Interface	83
Figure 3.7.7	User Menu Page Interface	85
Figure 3.7.8	Dietitian Menu Page Interface	87
Figure 3.7.9	User Profile Page Interface	89
Figure 3.7.10	Dietitian Profile Page Interface	91
Figure 3.7.11	Manage User Profile Page Interface	93
Figure 3.7.12	Manage Dietitian Profile Page Interface	95
Figure 3.7.13	View Recipe Page Interface	97
Figure 3.7.14	Recipe Details Page Interface	99

Figure 3.7.15 Calculate BMI Page Interface	101
Figure 3.7.16 Dietitian Edit Recipe Page Interface	103
Figure 3.7.17 Dietitian Upload Page Interface	105
Figure 3.7.18 Dietitian Edit Recipe Details Page Interface	107
Figure 3.7.19 Dietitian Update Page Interface	109
Figure 3.7.20 Chat Name List Page Interface	111
Figure 3.7.21 Chat Page Interface	113
Figure 4.2.1 Android Studio 2021.3.1 Dolphin	120
Figure 4.2.2 Create Project	121
Figure 4.2.3 Interface of Identity Module	122
Figure 4.2.4 Code of Identity Module	123
Figure 4.2.5 Dietitian Login Module Interface	124
Figure 4.2.6 Code of Dietitian Login Module	125
Figure 4.2.7 User Login Module Interface	126
Figure 4.2.8 Code of User Login Module	127
Figure 4.2.9 Registration Module Interface	128
Figure 4.2.10 Code for Registration Module	129
Figure 4.2.11 Forgot Password Page	130
Figure 4.2.12 Code for Forgot Password Page	130
Figure 4.2.13 User Menu Page	131
Figure 4.2.14 Code for User Menu Page	132
Figure 4.2.15 Dietitian Menu Page	133
Figure 4.2.16 Code for Dietitian Menu Page	134
Figure 4.2.17 User Profile Page	135
Figure 4.2.18 Code of User Profile Page	136
Figure 4.2.19 Dietitian Profile Page	137

Figure 4.2.20 Code for Dietitian Profile Page	138
Figure 4.2.21 Manage User Profile Page	139
Figure 4.2.22 Code for Manage User Profile Page	140
Figure 4.2.23 Manage Dietitian Profile Page	141
Figure 4.2.24 Code for Manage Dietitian Profile Page	142
Figure 4.2.25 View Recipe Page	143
Figure 4.2.26 Sample code of View Recipe page	144
Figure 4.2.27 Search Module	145
Figure 4.2.28 Code for Search Module	146
Figure 4.2.29 Recipe Details Page	147
Figure 4.2.30 Example Code of Recipe Details Page	148
Figure 4.2.31 Calculate BMI Page	149
Figure 4.2.32 Code for Calculate BMI Page	150
Figure 4.2.33 Dietitian Edit Recipe Page Interface	151
Figure 4.2.34 Code of Add Floating button	152
Figure 4.2.35 Dietitian Upload Page Interface	153
Figure 4.2.36 Sample Code of Dietitian Upload Page	154
Figure 4.2.37 Dietitian Edit Recipe Details Page	155
Figure 4.2.38 Code of Dietitian Edit Recipe Details Page	156
Figure 4.2.39 Dietitian Update Page	157
Figure 4.2.40 Code of Dietitian Update Page	158
Figure 4.2.41 Chat Name List Page	159
Figure 4.2.42 Code for Chat Name List Page	160
Figure 4.2.43 Chat Page	161
Figure 4.2.44 Code for Chat Page	162
Figure 4.3.1 User Acceptance Test Question 1	166

Figure 4.3.2 User Acceptance Test Question 2	167
Figure 4.3.3 User Acceptance Test Question 3	167
Figure 4.3.4 User Acceptance Test Question 4	168
Figure 4.3.5 User Acceptance Test Question 5	168
Figure 4.3.6 User Acceptance Test Question 6	169

CHAPTER 1

INTRODUCTION

1.1 Introduction

Cooking is a meditative activity especially baking. Baking is a method of cooking with dry heat, usually in an oven and it is probably the oldest way of cooking. A various type of baked goods derived from flour or meal, such as bread, rolls, cookie, pies, muffins, and pastries (R.Paul Singh & Samuel A. Matz, 2021). It makes people feel calming and creates space in the mind while weighing the ingredients, whisking the eggs, and shifting the flour. Feeling the flour in hands, hearing the blender, and smelling the delicious result stimulate the senses, thereby increasing endorphins (*5 Reasons Baking Is Good for Mental Health - Goodnet*, 2013).

However, most people do not have time to study and appreciate this baking procedure due to their hectic lifestyles. They do not have time to seek for a recipe book, and do not know when will be able to bake or how long does baking take. Therefore, making baking recipes available through a mobile application provides convenience to people, especially to those who work and need to cook at the same time. Nowadays, people spend more time on their mobile phone. Based on the statistic in Statista, nearly half of respondents to a survey from February 2021 stated that they spend five to six hours per day on their smartphones, excluding work-related smartphone use (L.Ceci, 2022). Instead of looking for information in a book, they are more inclined to utilize their phone and mobile apps become important in daily life. Communication, shopping, cooking, social media, education, business, and banking are all examples of where mobile apps are currently used (Harnil Oza, 2017) . Since recipes book is less friendly and convenience to user, this project intends a baking recipes application which aims to

provide digitalized recipes with step-by-step instructions, nutritional values, and list of ingredients for users to assist them in groceries shopping.

To begin baking, people must have all the necessary supplies on hand as well as a proper recipe. Those with no prior baking knowledge acquire step-by-step instructions during their baking process. Besides, for people who suffering obesity, diabetes, and other health problems, they necessitate to determine nutritional values and creating treats that are acceptable for them. In this application, recipes and nutritional value are available through the app. The recipes are provided by dietitians, users can consult to the dietitian if they have any problems.

1.2 Problem Statement

Today, people are more likely to use their phones to search information rather than read books. Many baking recipes books sold in the market contain a large number of recipes, but they take up a lot of shelf space. In addition, buying a baking recipes book from a bookstore is expensive.

Based on the statistic from World Health Organization, the majority of the world's people live in countries where overweight and obesity kill more people than underweight. In 2020, about 39 million children under the age of 5 were overweight and obese. A diet that high in fat and sugar is the primary contributor to obesity and overweight (Obesity and Overweight, 2021.). The recipes in most cookbooks only give the ingredients and steps, but not nutritional values. Because recipes do not provide nutritional values and calories per serving, people with health problems cannot enjoy their treats. They are also difficult to consult to dietitian about the recipes.

Apart from that, people today are occupied with work and family. There are many reasons why it is not possible to learn how to bake at home, such as insufficient time or forgetfulness of the necessary ingredients. To avoid missing ingredients, they need simple recipes that is easy to bring along. Nevertheless, they cannot bring recipes book to the supermarket when they want to pick up the ingredients they need.

1.3 Objective

The objectives of this projects are:

- i. To design a system for people who suffer obesity and for non-experienced users when learning new baking recipes.
- ii. To develop a mobile application that digitalizes baking recipes and provide communication function between user and dietitian.
- iii. To test the functionality of the mobile application designed to find out if the user interfaces is convenient and user-friendly.

1.4 Scope

The following scopes will be defined to determine how the problems stated could be solved:

- The system is mobile application based and focused on the users who has no experience in baking.
- The system is mobile application based and focused on the users who suffer from diabetes and obesity.
- The system allows user to login to their own account and get recipes they need from the application.
- The system allows user to edit profile and upload profile image.
- The system allows user to view list of the recipes posted by dietitian.
- The system provides step-by-step directions and the estimated prepared time to user.
- The system allows user to view nutritional information and calories per serving based on the recipes they choose.
- The system allows user to calculate their BMI.
- The system allows dietitian to upload, edit and delete recipes.
- The system allows user to consult to the dietitian about the recipes through chat.

1.5 Thesis Organization

There are five chapters in this thesis. The first chapter discusses the introduction of the Baking Recipe Mobile Application. As well, it covers the background of the problem, the objective to achieve in the project and the scope of the project. Thesis organization is also provided in this chapter.

In chapter 2, literature review of the project will be discussed. This part will include a list of the existing system, research and compare among them regarding the solution, advantages and disadvantages of the system.

Chapter 3 will discuss the methodology used in the process of developing the Baking Recipe Mobile Application. It includes the methodology used in the software, use case, context diagram, flowchart, and system requirements.

Chapter 4 will discuss the results of the project, and system testing after the actual system is completed.

Chapter 5 discusses the conclusion of the project. This chapter ensure that the main objectives of the project are achieved and defines the constraint for future work of the system.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Literature reviews, more accurately called scholarly surveys, involve scholarly research about a specific topic or question, in order to discuss the published information (*The Importance of Literature Review in Research Writing | Elsevier Blog*, n.d.). Research gaps, relevant theories, and methods are highlighted in this overview of current knowledge. Literature reviews can help us to understand the problems faced by users when using the current system, as well as to analyse the advantages and disadvantages the current system has. By identifying the gaps in existing research, we can propose a better solution.

In this chapter, we have done research and analyse on three existed baking recipes mobile application which are ALL BAKING RECIPES, baking recipes, and Tasty to find out their differences, advantages, and disadvantages of their features. With these existing systems, we have proposed a new baking recipes application which can solved the problems faced by users when using current applications.

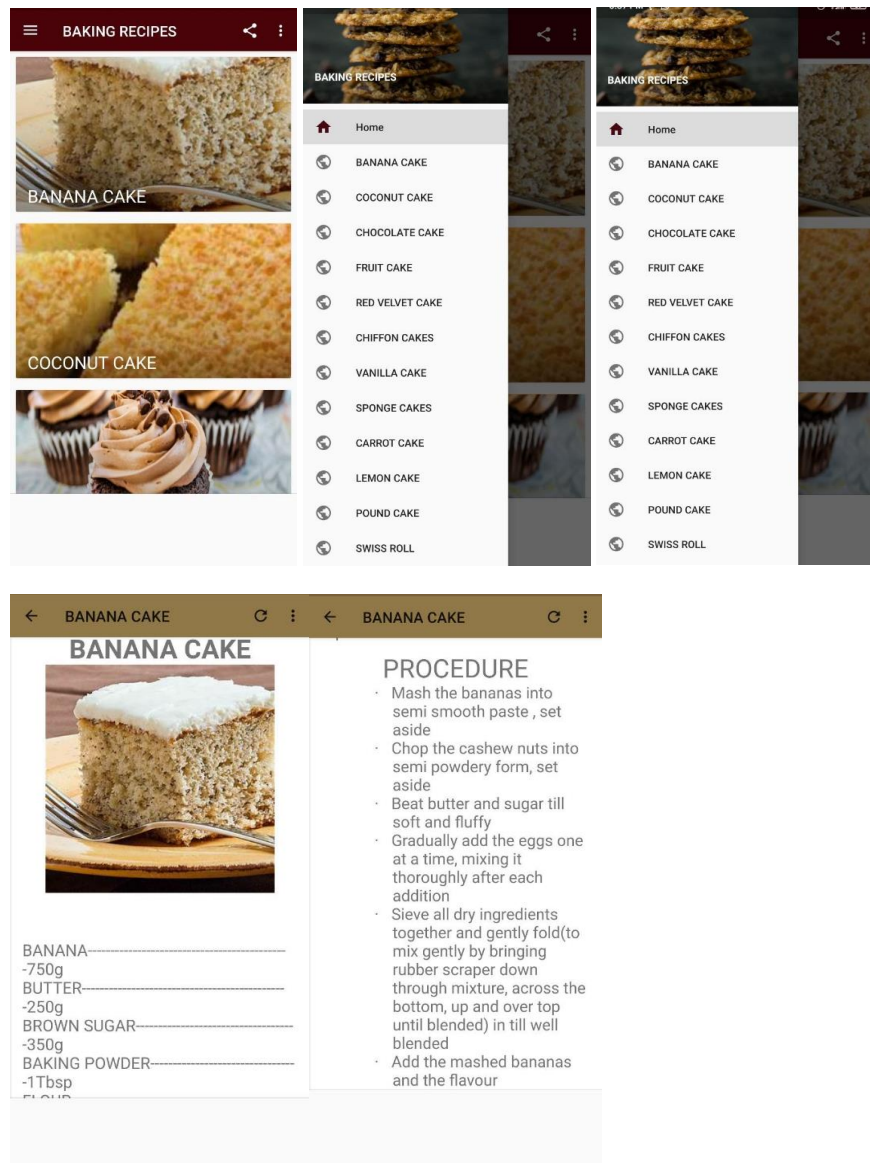
2.2 Three Related Work

2.2.1 ALL BAKING RECIPES

The ALL BAKING RECIPES is an application that can be used to find cake recipes. It provides step-by-step guidance on how to prepare cakes. There is a list of baking recipes, and the user can choose one by pressing the recipe's name in the list. Afterward, the recipes appear at the screen along with the preparation steps. The user can also leave a review of the recipes after baking. The app can be downloaded from Google Play store for Android user.

Among the strengths of this app is the user interface, which allows new users to operate it easily. Moreover, the picture provided is relevant to the result outcome, thereby giving the user a sense of what to expect.

This mobile app has a weakness in that the site doesn't provide nutritional information and serving sizes for the recipes, only the recipes. People with medical problems cannot determine whether the recipes are suitable for them to consume. Additionally, it does not provide an estimated time for preparing and it is not helpful to those who are busy.



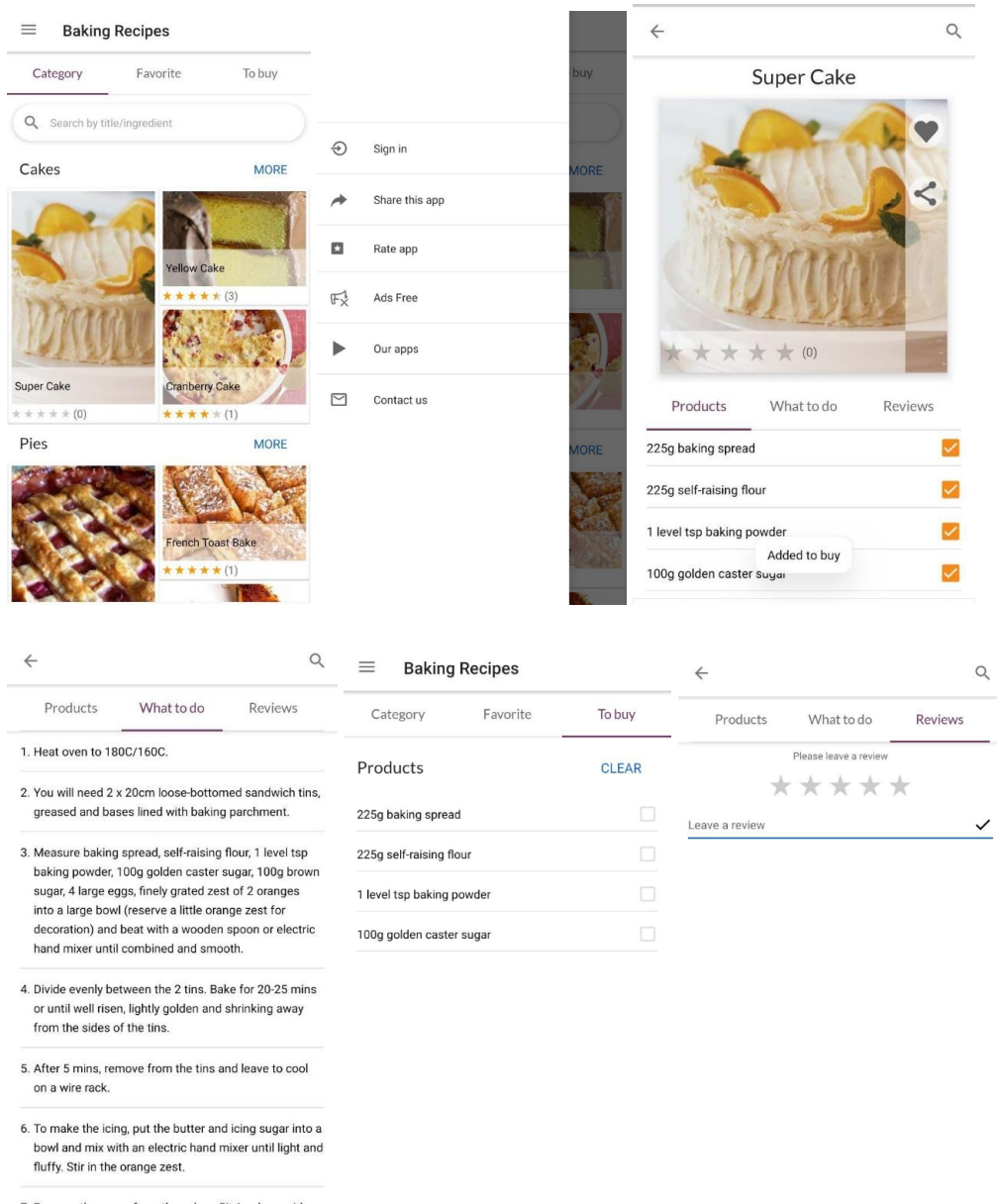
Figures 2.2.1: ALL BAKING RECIPES app interface

2.2.2 Baking Recipe

The Baking Recipes mobile application allows users to search and save the recipes. Recipes are divided into a few categories, such as cakes, pies, cookies, muffins, etc. Additionally, it provides a shopping list for users to mark out the things they need to buy. Initially, the user must log in or register for a new account. Select the recipe and click on the picture displayed. It will display a recipe with an ingredient list and the user can mark off the ingredients that they want to add to their To Buy list. As part of the application, the user can also leave a review of the recipes after baking. The app can be downloaded from Google Play store for Android user.

This application is very user-friendly; users can create an account and save their favourite recipes for future use. Besides, this application provides a list of ingredients that users can mark out when going shopping. This will make the process much easier for them. In addition, the picture provided is relevant to the result outcome, thereby giving the user a sense of what to expect.

This mobile app has a weakness in that it does not provide nutritional information and serving sizes for the recipes, only the recipes. People with medical problems cannot determine whether the recipes are suitable for them to consume. Additionally, it does not provide an estimated time for preparing and it is not helpful to those who are busy.



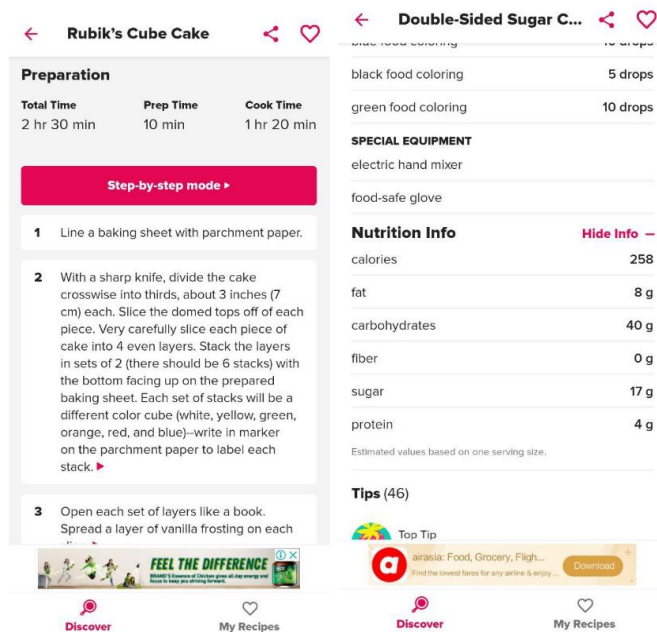
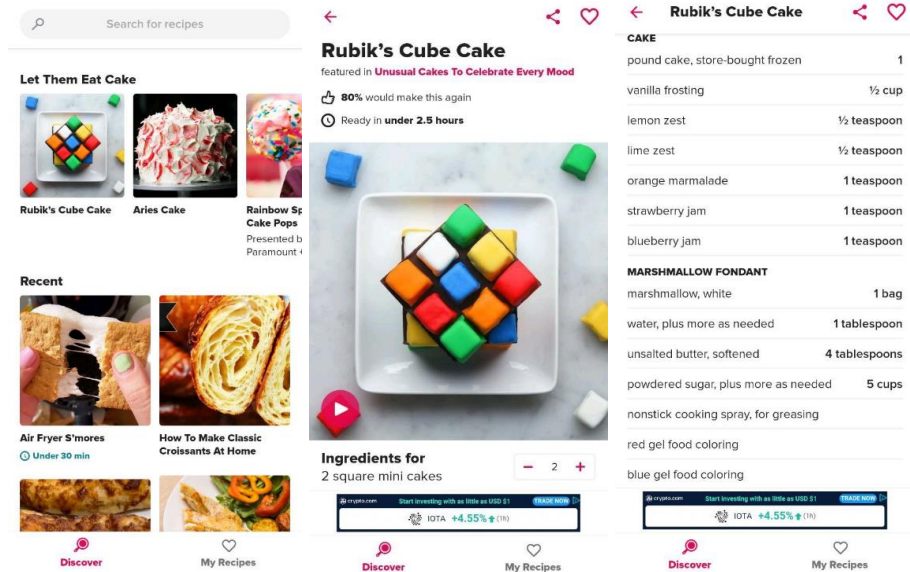
Figures 2.2.2: Baking Recipes app interface

2.2.3 Tasty

The Tasty mobile application is a recipes application developed by US. Recipes for meals and desserts are provided in this application. For users to be able to save recipes, they need to log in via Facebook or Gmail. Select the recipe and click on the picture displayed. It will display a recipe with an ingredient list, preparation time, special equipment needed, and nutritional info of the recipes. By adjusting the serving size, the user can adjust the portion of ingredients. As part of the application, the user can also leave a review of the recipes after baking. The app can be downloaded from Google Play store for Android user.

Among the strengths of this app is the user interface, which allows new users to operate it easily. This application provides nutritional info and preparation time for user to refer. People who suffer from medical conditions can check the nutritional info to determine whether the recipes are appropriate for them. People who are busy, especially those who work and cook at the same time, can refer to the preparation time for a rough estimate of baking time. In addition, the picture provided is relevant to the result outcome, thereby giving the user a sense of what to expect.

This mobile app has a weakness in that it does not provide a shopping list for users to mark and record. Forgetful users might overlook essential ingredients.



Figures 2.2.3: Tasty app interface

2.3 Comparative Analysis

2.3.1 Specification of existing system

Existing System	Specification/ Features
ALL BAKING RECIPES	<ol style="list-style-type: none">1. Able to see preparation steps2. Able to leave reviews
Baking Recipes	<ol style="list-style-type: none">1. Able to see preparation steps2. Able to search recipes3. Able to save favorite recipes4. Able to add ingredients to shopping list5. Able to leave reviews
Tasty	<ol style="list-style-type: none">1. Able to see preparation steps2. Able to search recipes3. Able to save favorite recipes4. Able to see preparation time5. Able to see nutritional values6. Able to leave reviews

Table 2.3.1 Specification of existing system

2.3.2 Comparison of three existing system

Apps/Features	ALL BAKING RECIPES	Baking Recipes	Tasty
Application Type	Mobile	Mobile	Mobile
Internet	Yes	Yes	Yes
Languages	English	English	English
Covered Recipes	Cakes only	Desserts	Meals and Desserts
Recipes Procedures	Yes	Yes	Yes
Preparation Time	No	No	Yes
Shopping List	No	Yes	No
Nutritional Value	No	No	Yes
Reviews	Yes	Yes	Yes

Table 2.3.2 Comparison of three existing systems

2.3.3 Advantages and Disadvantages of three existing system

Application	Advantages	Disadvantages
ALL BAKING RECIPES	<ol style="list-style-type: none"> 1. Consists of the picture of result outcome, giving the user a sense of what to expect. 2. Easy to operate, user can get recipes by clicking the picture displayed. 	<ol style="list-style-type: none"> 1. It does not provide nutritional information and serving sizes for the recipes. 2. It does not provide an estimated time for preparation.
Baking Recipes	<ol style="list-style-type: none"> 1. Consists of the picture of result outcome, giving the user a sense of what to expect. 2. Easy to operate, user can get recipes by clicking the picture displayed. 3. Users can create an account and save their favorite recipes for future use. 4. Has a list of ingredients for users to mark out when going shopping. 	<ol style="list-style-type: none"> 1. It does not provide nutritional information and serving sizes for the recipes. 2. It does not provide an estimated time for preparing and it is not helpful to those who are busy.
Tasty	<ol style="list-style-type: none"> 1. Consists of the picture of result outcome, giving the user a sense of what to expect. 2. Easy to operate, user can get recipes by clicking the picture displayed. 3. Users can create an account and save their favorite recipes for future use. 	<ol style="list-style-type: none"> 1. It does not provide a shopping list for users to mark and record. Forgetful users might overlook essential ingredients.

	4. Provides nutritional info and preparation time for user to refer.	
--	--	--

Table 2.3.3 Advantages and Disadvantages of the three existing system

2.4 Chapter Summary

Based on the research and comparison above, it can be concluded that each system of baking recipes has its advantages and disadvantages. In this chapter, three different mobile apps have been reviewed. As such, it is crucial that the developer conduct research and compare existing systems to find the best ideas and design the most useful application that helps solve the user's problems.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter will mainly discuss about the methodology used in this project. An implementation methodology, also known as the software development lifecycle, is the set of principles or rules that guide the project development group. Methodology selection depends on the project requirements and timeframe. Based upon the development methodology chosen, projects are divided into phases (Priya Pedamkar, n.d.). In this chapter, the development of baking recipes mobile application will be discussed in detail.

Thus, the methodology chosen for this proposed system is Agile development. Agile development has been chosen because the project will be less affected by changes on the project if there are any changes in the project requirements compared to the waterfall model. Unlike the waterfall model, an agile model involves writing and updating a test plan on a regular basis. Test plans in agile usually include the following: Testing Scope, New Functionalities that will be tested, Performance and Load Testing, and Infrastructure considerations (Thomas Hamilton, 2022). Besides, in this method, the development process is divided into iterations, which are fast and flexible.

3.2 Methodology

Agile methodology refers to an iterative approach to the software development with continuous testing throughout a project's lifetime. It consists of 6 phases which are plan, design, develop, test, release, and feedback. One of the most unique features of this model is its ability to take the developer back to the previous stage. Besides, rather than taking months to develop a product, it is often delivered much faster. In addition, requirements are modified even if they are late and regular adjustments are made to

accommodate changes. Therefore, this methodology is suitable for this projects that have short time to be developed.

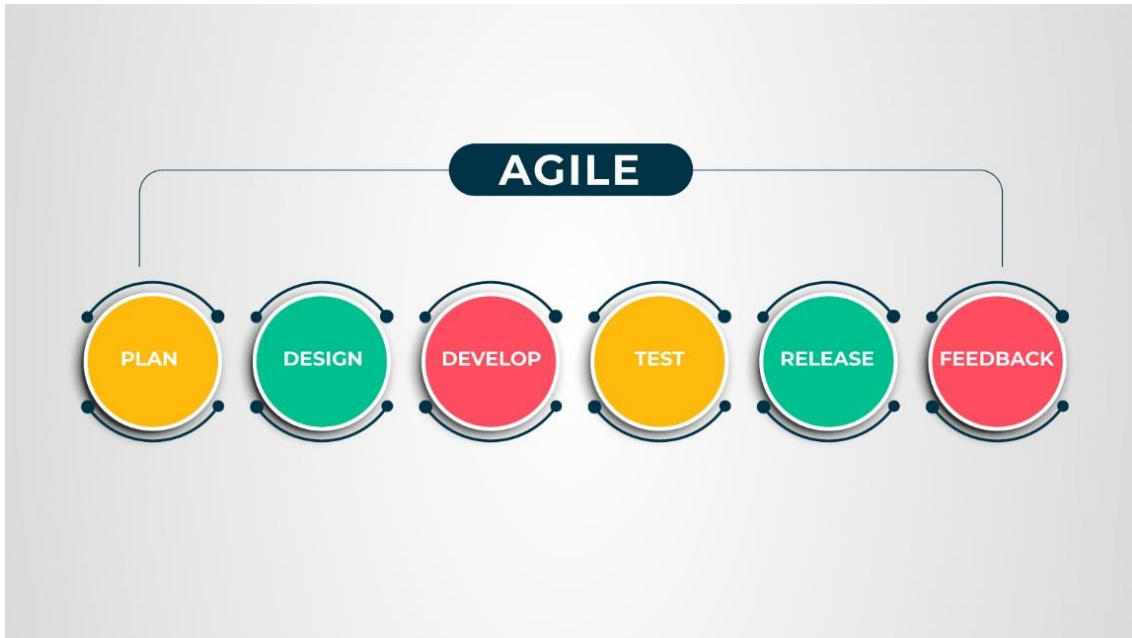


Figure 3.1: AGILE development Model

1. Planning Phase

For the development of the whole system, the requirement planning phase is an essential first step. Because at this point the requirements for the Baking Recipes Mobile Application could be gathered around by the stack holders. The requirements can be separate into two categories which are functional and non-functional requirements. For example, the functional requirement in this baking recipes application is the system should allow users to search the recipes needed, while the non-functional requirement is the data shown to user must be correct and available.

2. Design Phase

The design phase included two different phases which are interface design and software design. User flow diagram or the high-level UML diagram can be used to show

the work of features and how it will be applied in the existing system(*Agile Model (Software Engineering) - Javatpoint, n.d.*). As part of interface design, a demo of interface design of Baking Recipes Mobile Application will be conducted to provide a general impression of how the system looks. In software design, the developed requirements proposed during the previous stage is introduced and reviewed. Next, objectives and tools that required to obtain the outcome will be explored.

3. Develop Phase

During the development phase of the software development process, software code is produced, and technical documents are translated into software. As part of this phase, the developer team works with the UX designer to develop code based on all the requirements(*The Phases Of Agile Software Development Life Cycle & Workflow And Project Management, 2022*). Baking Recipes Mobile Application will use android studio to carry out the coding phase and this stage will take a much longer period since it will continue to be done until all the requirements are satisfied, and the system runs smoothly.

4. Test Phase

During this stage, the software is tested for compatibility with everything else that have been built previously. This is to make sure the system is working and is bug-free, flaws-free. The testing phases in this SDLC stage develop more comprehensively, and they include not only functional tests but also systems integration, interoperability, and user acceptance testing, among others. In Baking Recipes Mobile Application, the testing part will mainly focus on debugging to make sure the system is error-free.

5. Release Phase

In release stage, the software is now fully deployed and ongoing support will be provided to keep the system running smoothly and any new bugs will be fixed. The software is updated subsequently by adding new features and fixing issues.

6. Feedback Phase

Feedback phase or review phase is the last phase in the Agile Development cycle. At this phase, the progress that has been made toward completion of the requirement will once again be evaluated. Any possible ideas for solving the problem found in the previous phase will be considered and implemented in the next iteration.

3.3 Project Requirement

Functional Requirement

- The system should allow users to login to their account email and password.
- The system should allow users and dietitians to edit profiles and add profile pictures.
- The system should allow users to search for the recipes needed.
- The system should allow users and dietitians to view recipes details and nutritional value of the baked goods they made.
- The system should allow dietitians to add, edit and delete recipes.
- The system allows user to consult to the dietitian about the recipes through chat or email.
- The system should have a user-friendly interface and easy to understand by user and dietitians.

Non-functional Requirement

1. Security
 - The account and password of user must be protected properly.
 - The user data must be kept and save properly.
2. Reliability
 - The data shown to user must be correct and available.

3. Performance

- The system must be accessible in anytime.

4. Usability

- The system interface must be user-friendly and easy to understand.

Constraints and Limitation

- The application is only available for Android devices.

3.4 User Requirement

Question 1 : Are you a male or a female?

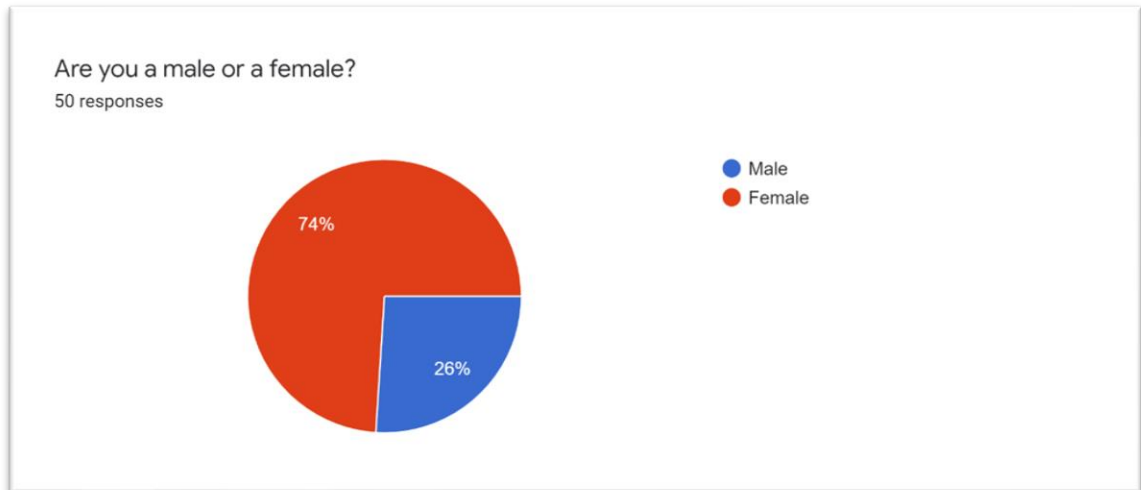


Figure 3.4.1 User Requirement Question 1

Based on the survey, there are 37 female (74%) and 13 males (26%) responded to the survey.

Question 2 : Do you have any experiences in baking?

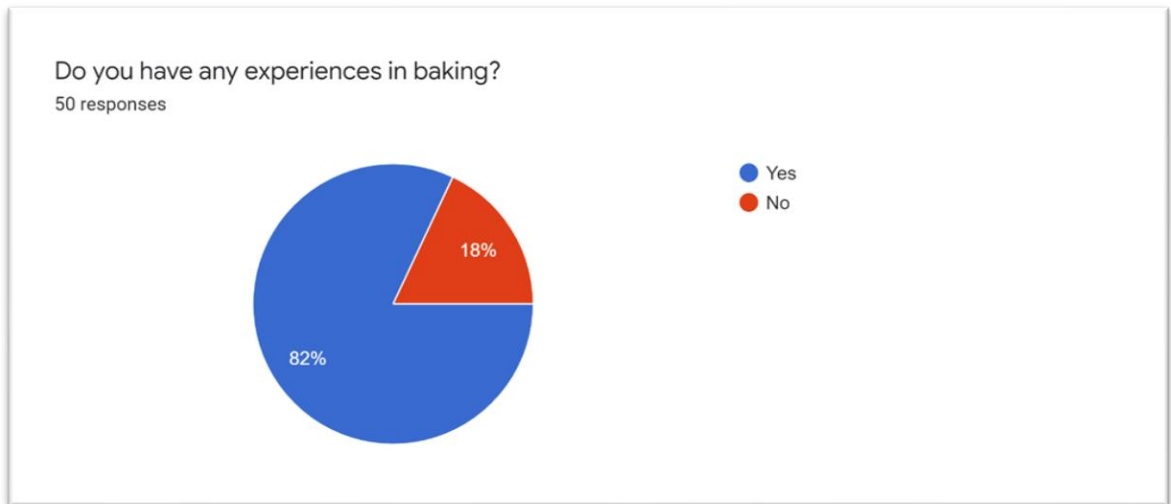


Figure 3.4.2 User Requirement Question 2

Based on the survey, there are 41 respondents (82%) have experiences in baking while 9 respondents (18%) have no experiences in baking.

Question 3: How often do you bake in a month?

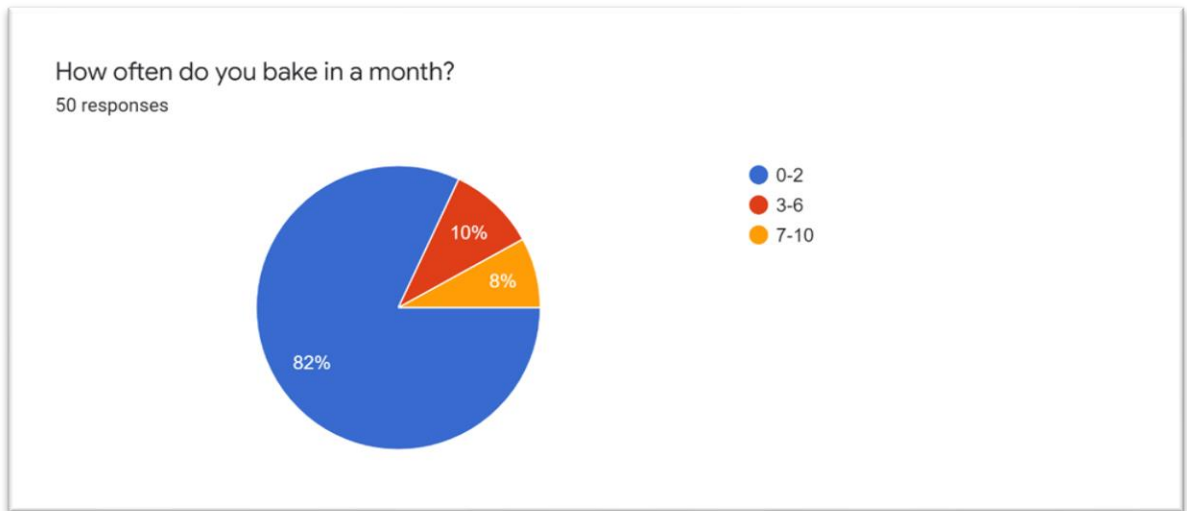


Figure 3.4.3 User Requirement Question 3

Based on the survey, 41 respondents (82%) bake 0-2 times in a month, 5 respondents (10%) bake 3-6 times a month and 4 respondents (8%) bake 7-10 times a month.

Question 4 :Where do you find inspiration for baking projects?

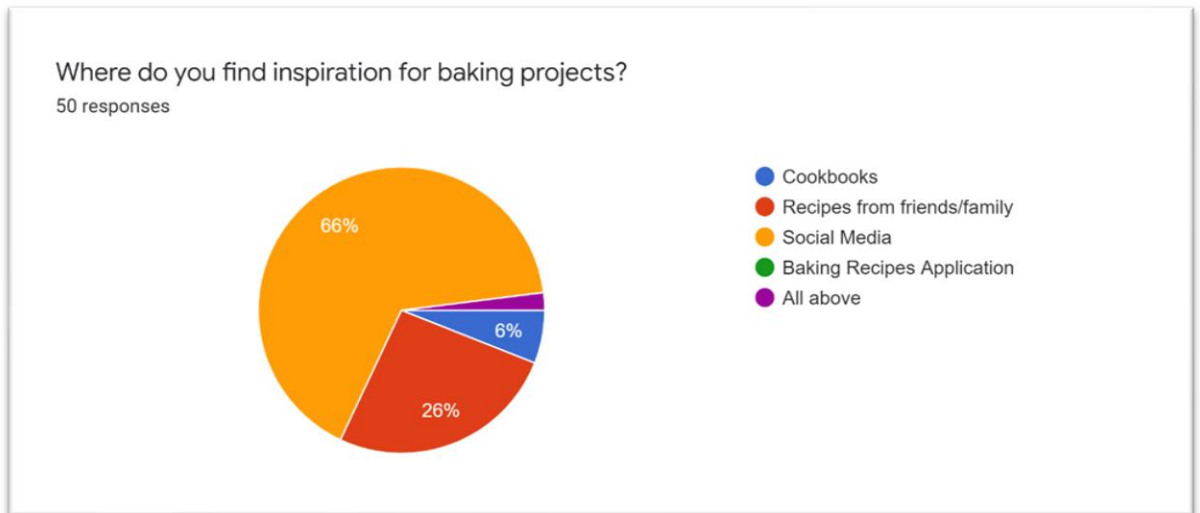


Figure 3.4.4 User Requirement Question 4

Based on the survey, there are 33 respondents (66%) find inspiration for baking projects from social media, 13 respondents (26%) from friends and family, 3 respondents (6%) from cookbooks and 1 respondents (2%) from all above.

Question 5: Have you ever use any kinds of baking recipes apps before?

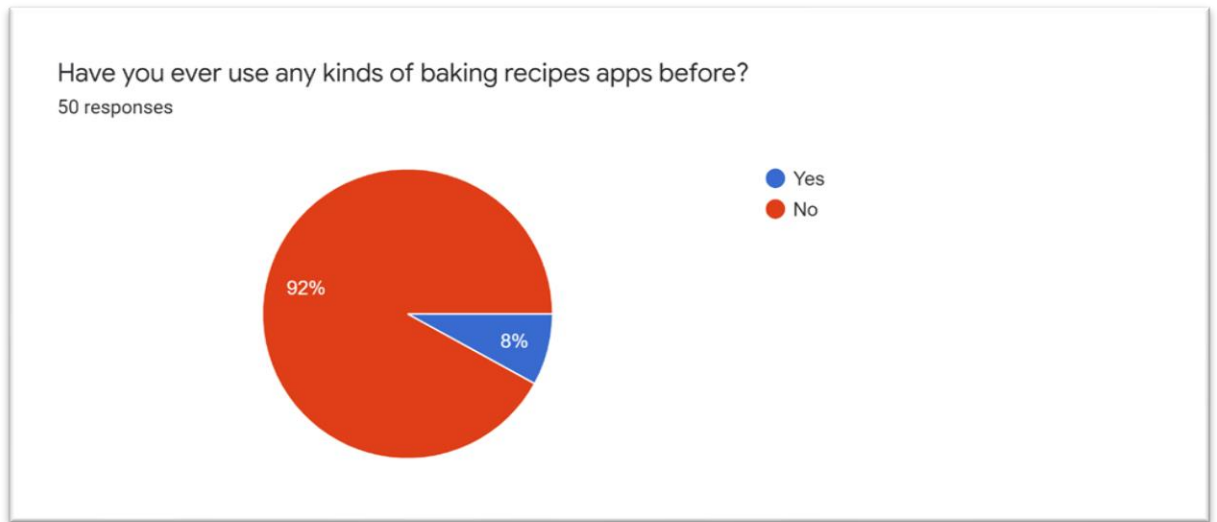


Figure 3.4.5 User Requirement Question 5

Based on the survey, there are 46 respondents (92%) ever use any kinds of baking recipes apps before while 4 respondents (8%) never use any kinds of baking recipes apps before.

Question 6 :In terms of preparing baking ingredients, is the ingredient shopping list helpful?



Figure 3.4.6 User Requirement Question 6

Based on the survey, there are 48 respondents (96%) feel that ingredient shopping list helpful while 2 respondents (4%) feel that ingredient shopping list is not helpful.

Question 7 :Does the nutritional facts and allergen advice of the recipe is helpful to user?

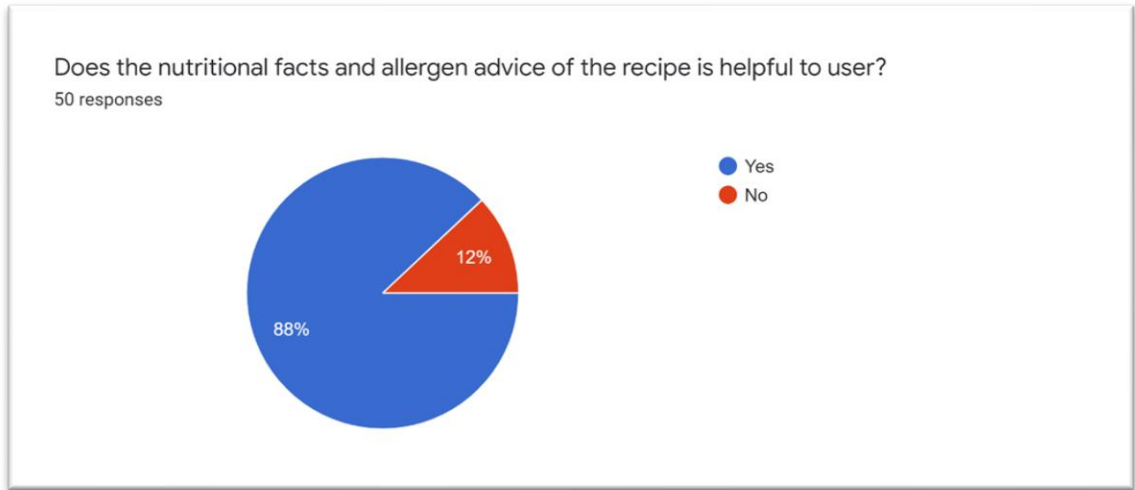


Figure 3.4.7 User Requirement Question 7

Based on the survey, there are 44 respondents (88%) think that the nutritional facts and allergen advice of the recipe is helpful to user while 6 respondents (12%) think that the nutritional facts and allergen advice of the recipe is not helpful to user.

Question 8 :What kinds of recipes do you expect to be included in baking recipes mobile application?

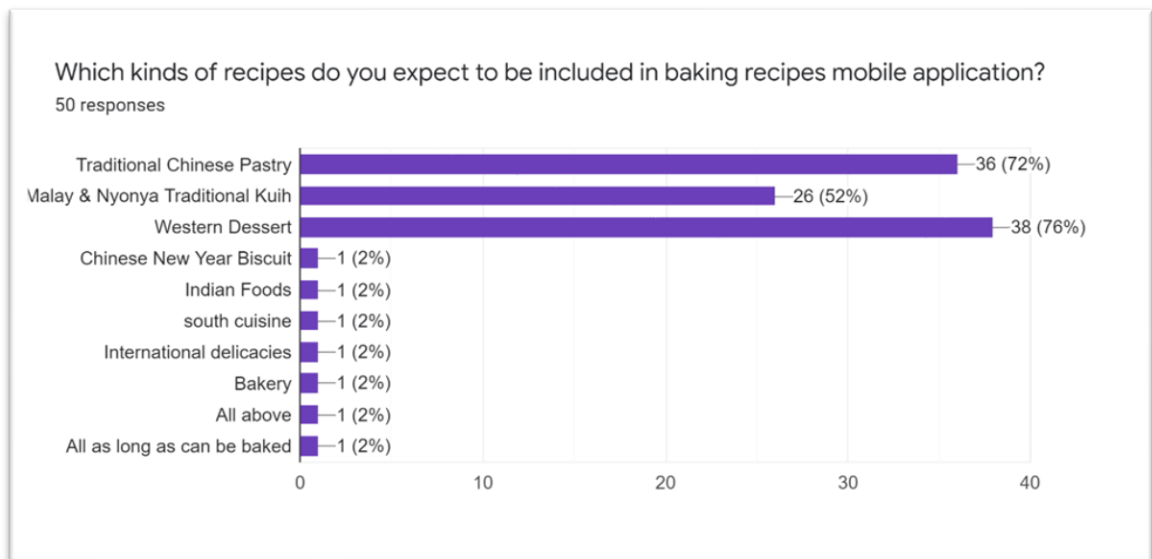


Figure 3.4.8 User Requirement Question 8

Based on the survey, there are 36 responses (72%) in traditional Chinese Pastry, 26 responses (52%) in Malay & Nyonya Traditional Kueh, 38 responses (76%) in Western Dessert, and 1 response (2%) in Chinese New Year Biscuit, Indian Foods, and etc. respectively.

3.5 Proposed Design

3.5.1 Flowchart

Flowchart For Login

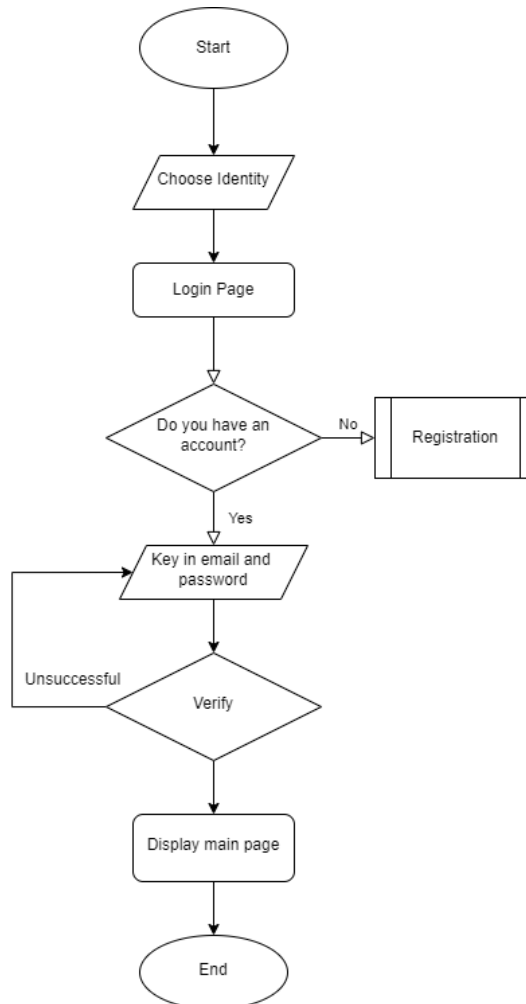


Figure 3.5.1 Flowchart for Login

The system starts with the user login interface. First, users are required to choose their identity whether they are users or dietitians. For the first time user, they need to register for a new account while for the existed user, they can log into their account via email and password. After click on the login button, the information will be verified. If the email and password is true, user can enter to the main page, else user are required to re-enter their email and password.

Flowchart For Registration

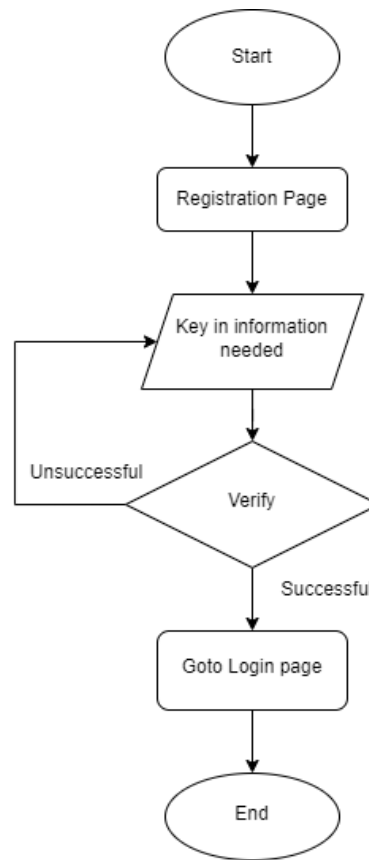


Figure 3.5.2 Flowchart for Registration

The system starts with the registration interface for those who do not have an account in the application. Users need to insert their name, age, email address, password and confirm password to sign up for a new account. If the user email does not register before and the password is valid, the system will show successful register messages and user will be direct to the login page. User can login to their account using the registered email and password.

Flowchart of User Activity

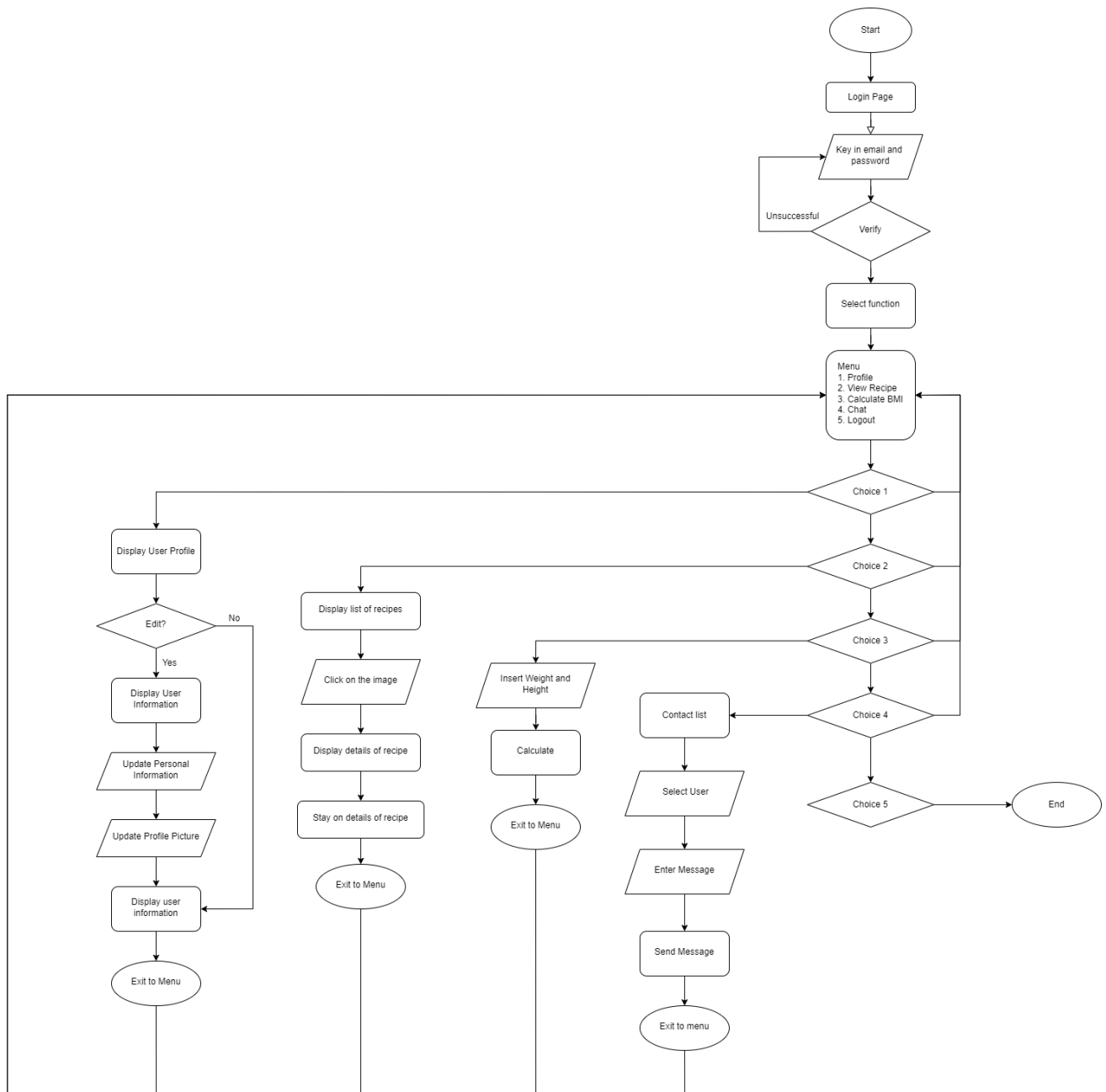


Figure 3.5.3 Flowchart of User Activity

The system starts user menu page. There are five choices in the user menu page which are profile, recipes, calculate BMI, chat, and logout. In the profile interface, user can choose to update their information by retype them. They can update their name, age, and phone number. Besides, they can change and upload profile picture in this interface.

After finished update their information, click save button to save the changes. If the change is valid, the system will display the profile interface with the updated information. In recipe page, system will show the image and the name of the recipes. User can click on the image to select the recipe they need, and they will be directed to the recipe details page. In the recipe details, user can view ingredients, prepared time, size serving, level of difficult and instructions of preparing the baked goods. In calculate BMI page, user can enter their weight and height to calculate their BMI value. In chat page, user can choose to chat with the dietitian by clicking to their chat box. The logout button redirect user to the identity page.

Flowchart of Dietitian Activity

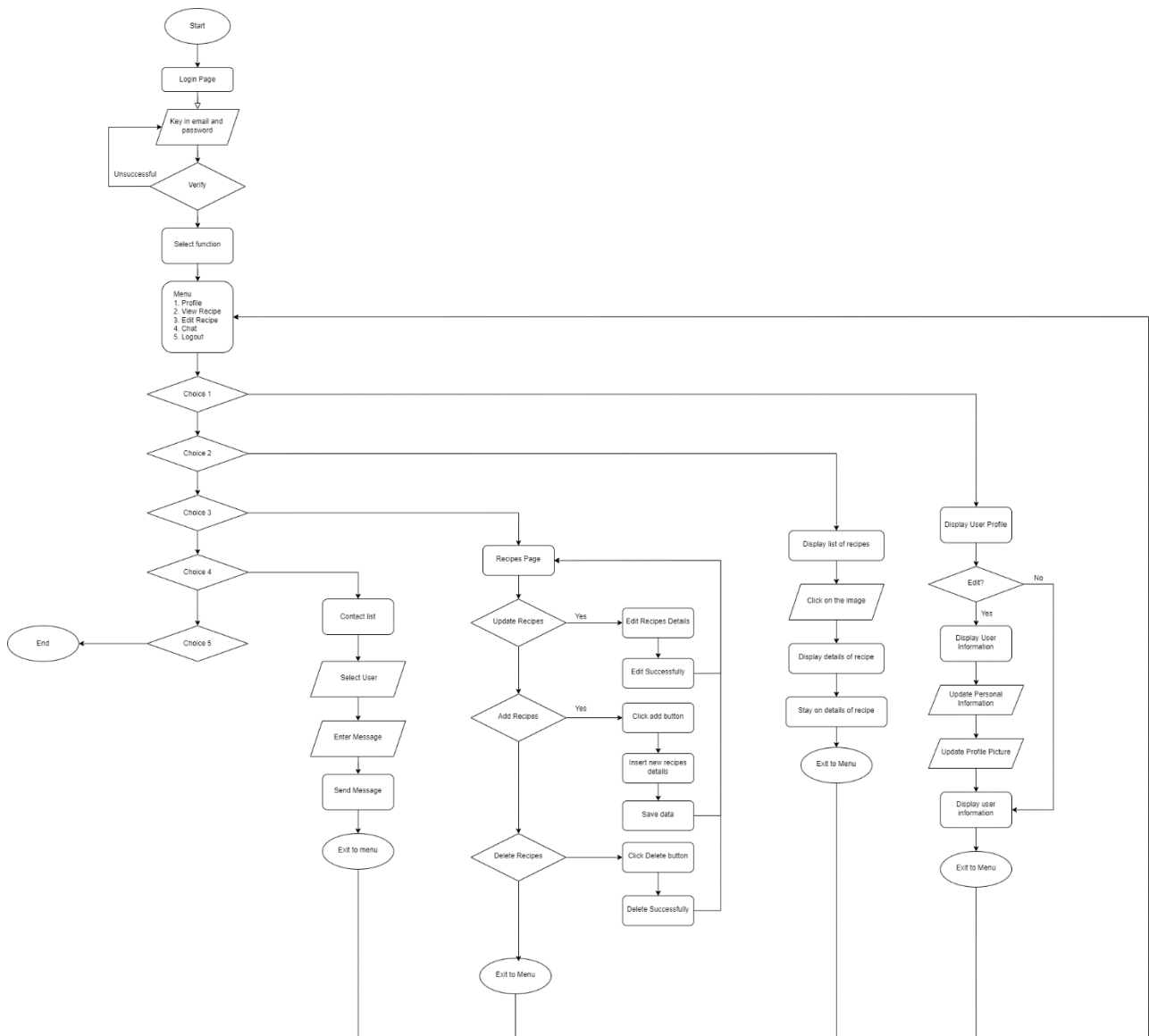


Figure 3.5.4 Flowchart of Dietitian Activity

The system starts dietitian menu page. There are five choices in the dietitian menu page which are profile, recipes, edit recipe, chat, and logout. In the profile interface, dietitian can choose to update their information by retype them. They can update their name, age, and phone number. Besides, they can change and upload profile picture in this interface. Dietitians are also required to upload their professional certificate to proof their identity. After finished update their information, click save button to save the changes. If the change is valid, the system will display the profile interface with the updated

information. In recipe page, system will show the image and the name of the recipes. Dietitians can click on the image to select the recipe they need, and they will be directed to the recipe details page. In the recipe details, dietitians can view ingredients, prepared time, size serving, level of difficult and instructions of preparing the baked goods. In edit recipe page, dietitian can add new recipes, edit recipes, and delete recipes that they upload before. In chat page, dietitian can reply chat from user by clicking to their chat box. The logout button redirect dietitian to the identity page.

3.5.2 Context Diagram

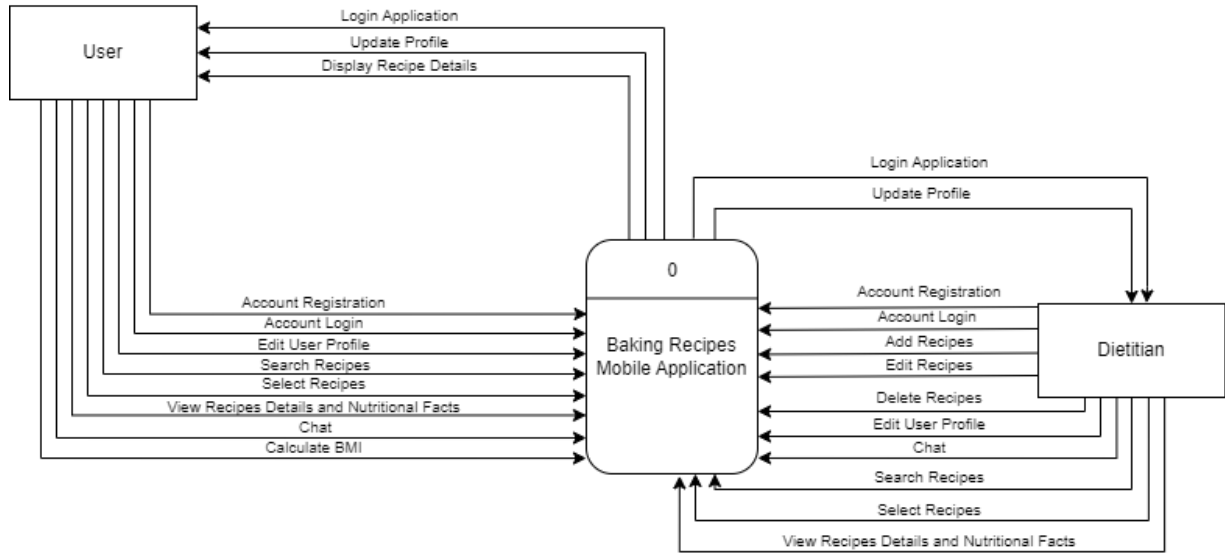


Figure 3.5.6 Context Diagram

3.5.3 Use Case Diagram & Description

Baking Recipes Mobile Application

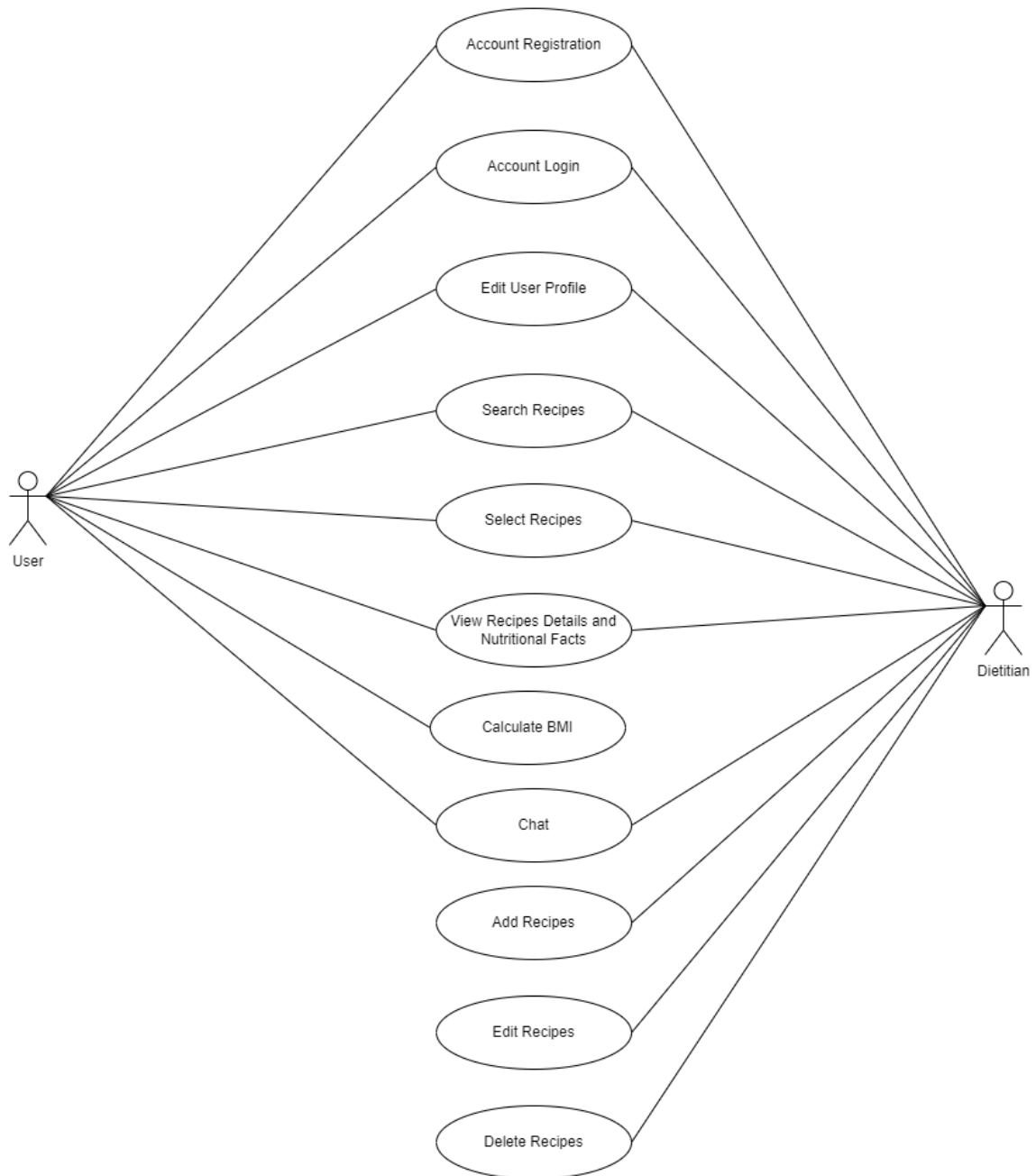


Figure 3.5.7 Use Case Diagram

01- Account Registration

Use Case Name	Account Registration
Brief Description	This use case is to register a new account for first time user.
Actor	User/Dietitian
Precondition	<ul style="list-style-type: none"> - The user is first-time user without an account. - The system must have an email.
Basic Flow	<p>User/ Dietitian</p> <ol style="list-style-type: none"> 1. The use case starts when the user clicks “Don’t have an Account” button. 2. The system will display a registration page. 3. User/Dietitian enters username. 4. User/Dietitian enters valid email address and personal details. 5. User/Dietitian enters password. 6. User/Dietitian enters confirm password. 7. User/Dietitian clicks “Sign Up” button. [A1] Cancel [E1] Invalid Registration Information [E2] Wrong Confirm Password 8. User/Dietitian registers success. 9. The use case end.
Alternative Flow	<p>[A1- Cancel]</p> <ol style="list-style-type: none"> 1. User clicks the “CANCEL” button. 2. Use cases ended.

Exception Flow	<p>[E1] Invalid Registration Information</p> <ol style="list-style-type: none"> 1. User/Dietitian retypes new username, password, and other personal details. 2. The use case continues to step number 3 in basic flow. <p>[E2] Wrong Confirm Password</p> <ol style="list-style-type: none"> 1. User/Dietitian enter different confirm password. 2. System display “Please enter correct password” message. 3. The use case continues to step 6 in basic flow.
Post-Conditions	User/Dietitian successfully register an account.
Constraints	-

Table 3.5.1 Use Case Description of Account Registration

02- Account Login

Use Case Name	Account Login
Brief Description	This use case is to login existing account for user who has register before.
Actor	User /Dietitian
Precondition	<ul style="list-style-type: none"> - The User/ Dietitian must have account database in system. - The login system is ready for user /dietitian to login.
Basic Flow	<p>User/Dietitian</p> <ol style="list-style-type: none"> 1. The use case starts when user/admin/dietitian enters Login Page. 2. User/Dietitian key in valid email and passwords. 3. User/Dietitian click “Login” button. [E1 – Invalid Email or Password] 4. User/Dietitian click “Forgot Password” button. [A1- Forgot Password] 5. The user/dietitian login to the system. 6. The use case end.
Alternative Flow	<p>[A1- Forgot Password]</p> <ol style="list-style-type: none"> 1. User/Dietitian clicks on “Forgot Password” button to request changing a new password. 2. User/Dietitian receives verification in email and change password.

	<ol style="list-style-type: none"> 3. The use case continues to step number 2 in basic flow.
Exception Flow	<p>[E1 – Invalid Email or Password]</p> <ol style="list-style-type: none"> 1. User/Dietitian enter wrong username or password. 2. The system checks the availability of the account. 3. System display “Please enter correct email or password” message. 4. The use case continues to step number 2 in basic flow.
Post-Conditions	User/Dietitian login to their account.
Constraints	-

Table 3.5.2 Use Case Description of Account Login

03- Edit User Profile

Use Case Name	Edit User Profile
Brief Description	This use case is for user to update profile information.
Actor	User/Dietitian
Precondition	- The user/dietitian must login to system.
Basic Flow	<p>User/Dietitian</p> <ol style="list-style-type: none"> 1. The use case starts when user clicks the profile at the dashboard. 2. The system will display user profile page. 3. User/Dietitian updates information in the profile. [A1- Update profile picture] / [A2- Dietitian professional certificate] 4. User/Dietitian clicks “Save Changes” button. [A3- Cancel] 5. User/Dietitian information will be updated in database. 6. The use case end.
Alternative Flow	<p>[A1- Update profile picture]</p> <ol style="list-style-type: none"> 1. User/Dietitian clicks upload button. 2. User/Dietitian chooses picture from gallery. 3. User/Dietitian uploads profile picture. 4. Use cases ended. <p>[A2- Dietitian professional certificate]</p>

	<ol style="list-style-type: none"> 1. Dietitian clicks upload button. 2. Dietitian chooses picture from gallery. 3. Dietitian uploads picture. 4. Use cases ended. <p>[A3- Cancel]</p> <ol style="list-style-type: none"> 1. User/Dietitian clicks the “CANCEL” button. 2. Use cases ended.
Exception Flow	-
Post-Conditions	User/Dietitian profile will be updated.
Constraints	-

Table 3.5.3 Use Case Description of Edit User Profile

04- Search Recipes

Use Case Name	Search Recipes
Brief Description	The use case allows user to search recipes in search bar.
Actor	User/Dietitian
Precondition	- The user must login the system.
Basic Flow	User/Dietitian <ol style="list-style-type: none">1. The use case starts when user clicks the search recipes at the main page.2. User enters the keywords of ingredients they needed.3. The system display recipes.4. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	User found recipes needed.
Constraints	-

Table 3.5.4 Use Case Description of Search Recipes

05- Select Recipes

Use Case Name	Select Recipes
Brief Description	The use case allows user to select recipes contents.
Actor	User/ Dietitian
Precondition	- The user must login the system.
Basic Flow	User/Dietitian <ol style="list-style-type: none">1. The use case starts when user login to account.2. The system displays the list of recipes at the main interface.3. The user clicks on the image of the selected recipes to view the recipes contents.4. The recipes details shown.
Alternative Flow	-
Exception Flow	-
Post-Conditions	User will direct back to main page.
Constraints	-

Table 3.5.5 Use Case Description of Select Recipes

06- View Recipes Details

Use Case Name	View Recipes Details
Brief Description	This use case describes the process of viewing recipes content and nutritional facts of recipes.
Actor	User/Dietitian
Precondition	<ul style="list-style-type: none"> - The user/dietitian must login the system. - The user/dietitian must select the recipes they needed
Basic Flow	<p>User/Dietitian</p> <ol style="list-style-type: none"> 1. This use case starts when user/dietitian click into the recipe details. 2. System displays recipe details and nutritional value. 3. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	User/dietitian will direct back to recipe details page.
Constraints	-

Table 3.5.6 Use Case Description of View Recipes Details

07- Calculate BMI

Use Case Name	Calculate BMI
Brief Description	This use case is to allow user to calculate BMI.
Actor	User
Precondition	- The user login to the account.
Basic Flow	User <ol style="list-style-type: none">1. This use case starts when user click Calculate BMI button.2. User inserts weight and height.3. User clicks on calculate button.4. System displays result of calculation to user.5. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	User gets the results of BMI.
Constraints	-

Table 3.5.7 Use Case Description of Calculate BMI

08- Chat

Use Case Name	Chat
Brief Description	This use case is to allow user and dietitian to communicate.
Actor	User/Dietitian
Precondition	- The user and dietitian login to the account.
Basic Flow	User/ Dietitian <ol style="list-style-type: none">1. This use case starts when user click chat button.2. User sends message to the dietitian to ask question.3. Dietitian can reply user's chat by selecting user's chat box.4. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	User/Dietitian communicate through chat.
Constraints	-

Table 3.5.8 Use Case Description of Chat

09- Add Recipes

Use Case Name	Add Recipes
Brief Description	This use case is to allow dietitian to add recipes.
Actor	Dietitian
Precondition	- The dietitian login to the account.
Basic Flow	<p>Dietitian</p> <ol style="list-style-type: none"> 1. The dietitian view list of the recipes. 2. The dietitian clicks the add recipes button. 3. The dietitian adds the recipes by type in the details in textbox. 4. The dietitian clicks “upload” button to add in database. 5. The recipes upload successfully. 6. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	Dietitians add new recipes in the application.
Constraints	-

Table 3.5.9 Use Case Description of Add Recipes

10- Edit Recipes

Use Case Name	Edit Recipes
Brief Description	This use case is to allow dietitian to edit recipes.
Actor	Dietitian
Precondition	- The dietitian login to the account.
Basic Flow	<p>Dietitian</p> <ol style="list-style-type: none"> 1. The dietitian view list of the recipes uploaded by themselves. 2. The dietitian clicks the image of the selected recipes. 3. The dietitian edits the recipes by type in the details in textbox. 4. The dietitian clicks “Update” button to update data in firebase. 5. The recipes updated successfully. 6. The use case end.
Alternative Flow	
Exception Flow	-
Post-Conditions	Dietitians update new recipes details.
Constraints	-

Table 3.5.10 Use Case Description of Edit Recipes

11- Delete Recipes

Use Case Name	Delete Recipes
Brief Description	This use case is to allow dietitian to delete recipes.
Actor	Dietitian
Precondition	- The dietitian login to the account.
Basic Flow	<p>Dietitian</p> <ol style="list-style-type: none"> 1. The dietitian view list of the recipes. 2. The dietitian clicks the image of the selected recipes. 3. The dietitian clicks on the delete recipes button. 4. The use case end.
Alternative Flow	-
Exception Flow	-
Post-Conditions	Dietitian deletes the recipes they uploaded.
Constraints	-

Table 3.5.11 Use Case Description of Delete Recipes

3.5.4 Activity Diagram

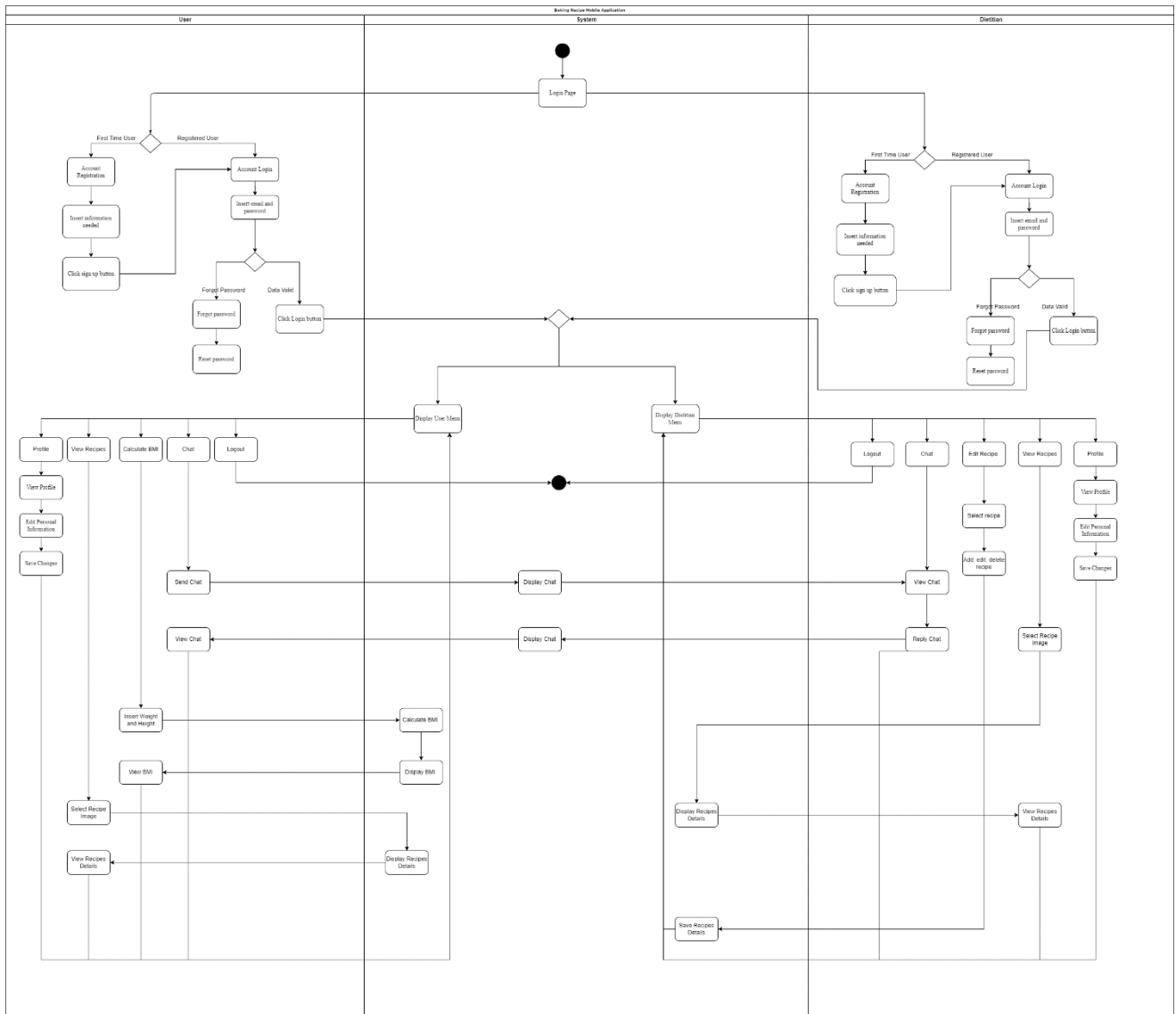


Figure 3.5.8: Activity Diagram

3.5.5 Storyboard

1. Identity

When entering the application, user can choose their identity whether user or dietitian.

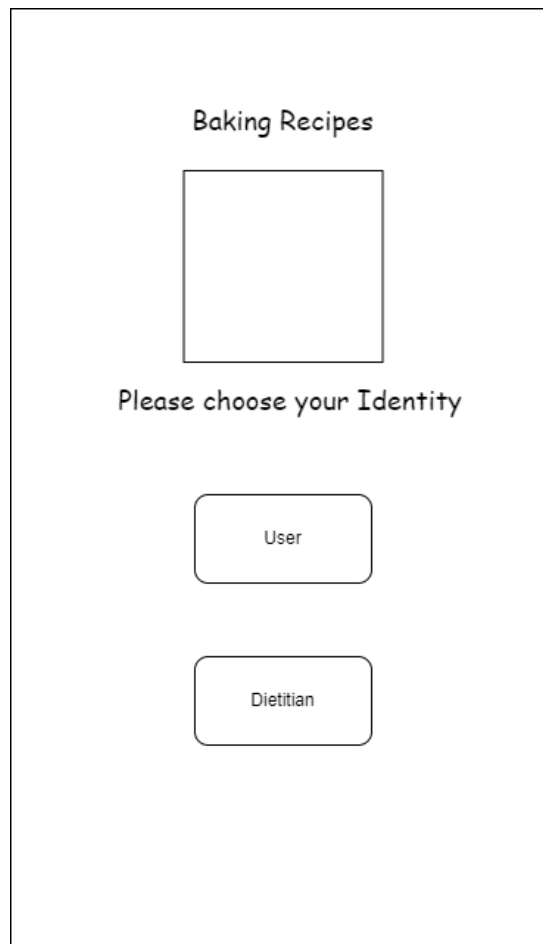
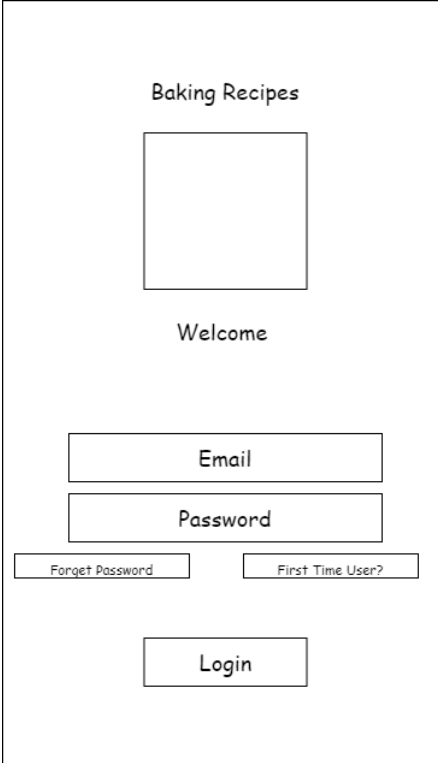


Figure 3.5.9: Choose Identity Interface

2. Login

After choosing their identity, system starts with the login interface where users/dietitians can enter email and password to login to their account. If they do not have an account, they can choose to register a new account. If they forgot their password, they could reset password through forgot password. After login, users/dietitians will enter the main page.



The login interface is a vertical rectangular box. At the top, it says "Baking Recipes". Below that is a square placeholder for a logo or image. Underneath is the word "Welcome". There are two input fields: "Email" and "Password". Below these are two smaller buttons: "Forgot Password" and "First Time User?". At the bottom is a large "Login" button.

Figure 3.5.10: Login Interface

3. Registration

To register an account, users/dietitians can click “First Time User?” button. If users or dietitian do not have an account, they can register a new account in this page by enter the information needed.

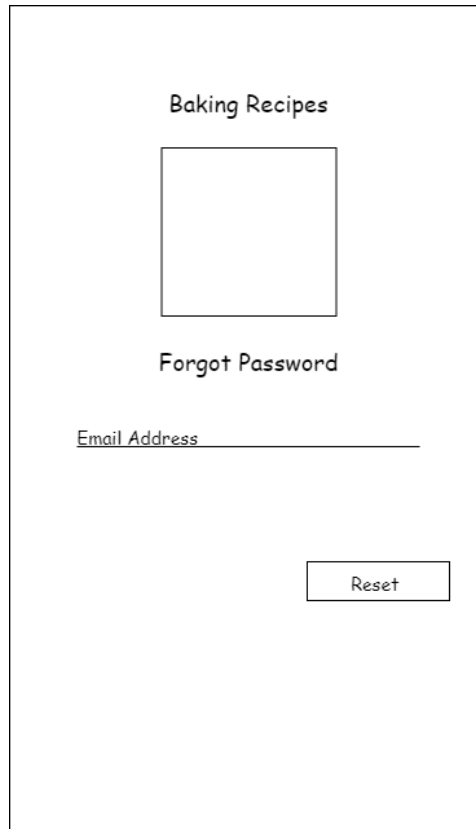


The registration interface is enclosed in a rectangular border. At the top center, the text "Baking Recipes" is displayed. Below this text is a large, empty square box. Underneath the square box, the text "Sign Up" is centered. Below "Sign Up" are six horizontal input fields, each with a label to its left: "Name", "Gender", "Date of Birth", "Email Address", "Password", and "Confirm password". At the bottom center of the form is a rectangular button labeled "Register".

Figure 3.5.11: User and Dietitian Registration Interface

4. Forgot Password

If users forgot their password, they could click Forgot Password button. The users require to key in their email and submit to get a new password reset chance.



The image shows a user interface for a 'Forgot Password' feature. At the top, the text 'Baking Recipes' is centered. Below it is a large, empty square box, likely a placeholder for a logo or image. Underneath the box, the text 'Forgot Password' is centered. Below that is a text input field with the label 'Email Address' on the left and a horizontal line for the input. At the bottom right of the form is a rectangular button labeled 'Reset'.

Figure3.5.12: Forgot Password Interface

5. User Menu Page

After successfully login, user will be directed to the user menu page. The user menu page contains profile, view recipes, calculate BMI, chat, and logout button. When user click logout button, they can exit from the application.

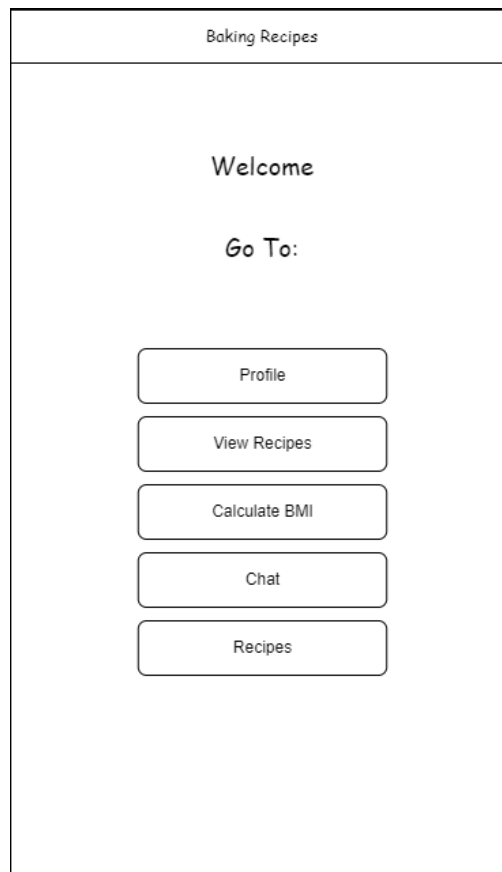


Figure3.5.13: User Menu Page Interface

6. Dietitian Menu Page

After successfully login, dietitian will be directed to the user menu page. The dietitian menu page contains profile, view recipes, edit recipe, chat, and logout button. When dietitian click logout button, they can exit from the application.

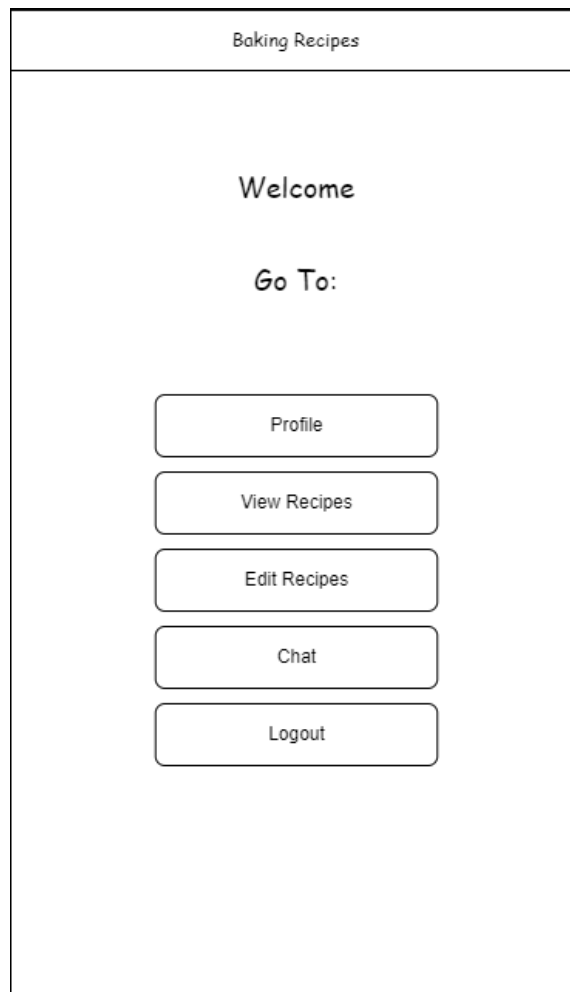
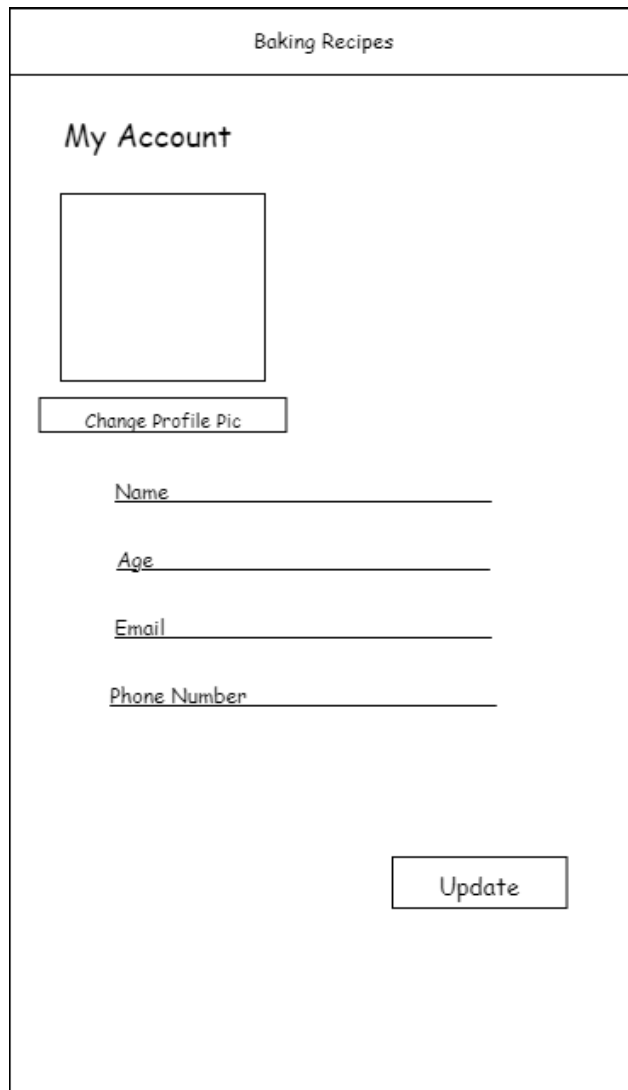


Figure3.5.14: Dietitian Menu Page Interface

7. User Profile Page

When users click profile on their menu, they will be directed to the profile page. They can click on the profile picture to update profile picture. The user can reset all information in this profile interface and save in database.

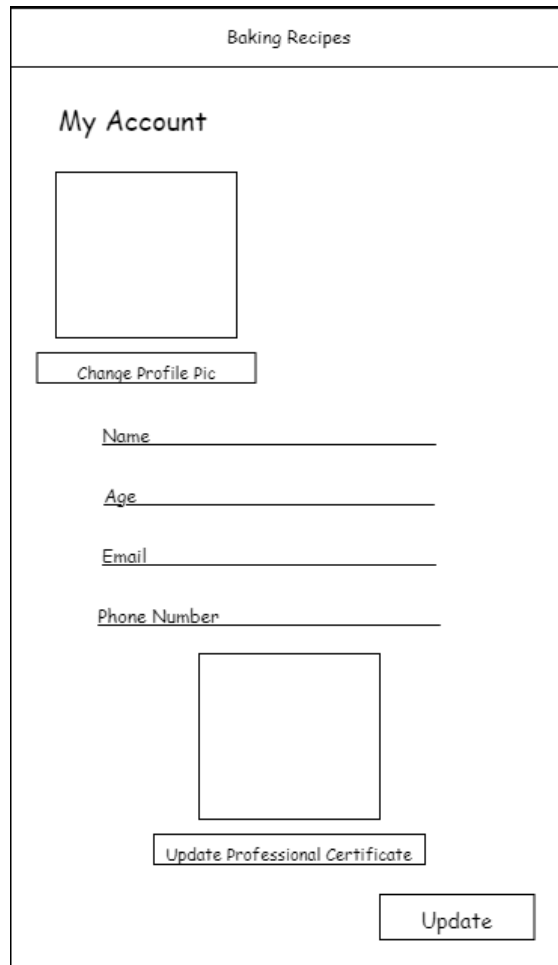


The image shows a wireframe of a user profile management interface. At the top, there is a header bar labeled "Baking Recipes". Below this, the main content area is titled "My Account". It features a square placeholder for a profile picture, with a "Change Profile Pic" button positioned directly below it. Underneath the picture area are four text input fields, each with a label: "Name", "Age", "Email", and "Phone Number". At the bottom right of the form area, there is an "Update" button.

Figure3.5.15: Manage User Profile Interface

8. Dietitian Profile Page

When dietitians click profile on their menu, they will be directed to the profile page. They can click on the profile picture to update profile picture. The dietitians can reset all information in this profile interface and save in database. Besides, they are required to update their professional certificate.



The image shows a web interface for managing a dietitian's profile. At the top, there is a header labeled "Baking Recipes". Below this, the main content area is titled "My Account". It features a square placeholder for a profile picture, followed by a button labeled "Change Profile Pic". Below the picture area are four text input fields labeled "Name", "Age", "Email", and "Phone Number". At the bottom of the form, there is a square placeholder for a professional certificate, followed by a button labeled "Update Professional Certificate". A final "Update" button is located at the bottom right of the interface.

Figure3.5.16: Manage Dietitian Profile Interface

9. View Recipes Page

When user/ dietitian clicks into view recipe button, they will be directed to the recipes list page. They can choose to view the recipes detail by selecting the image.

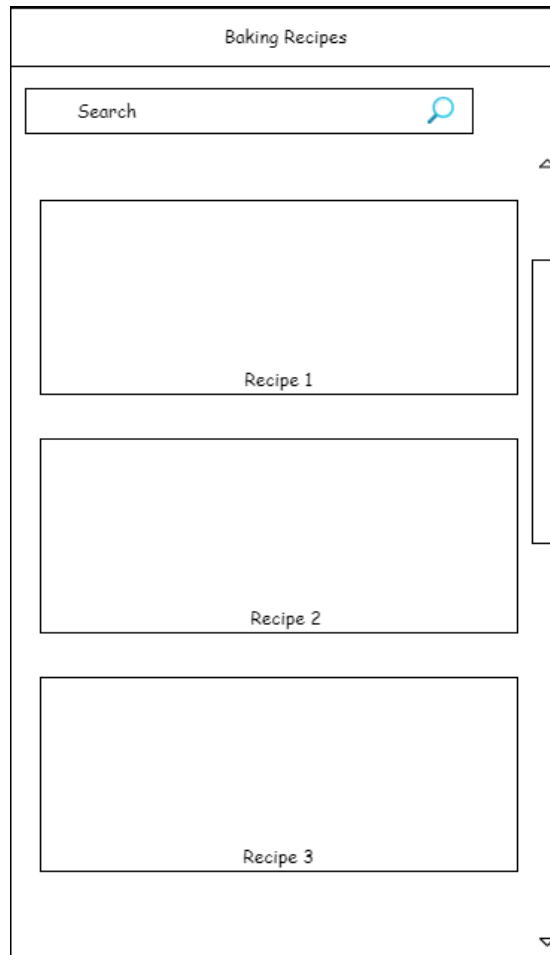
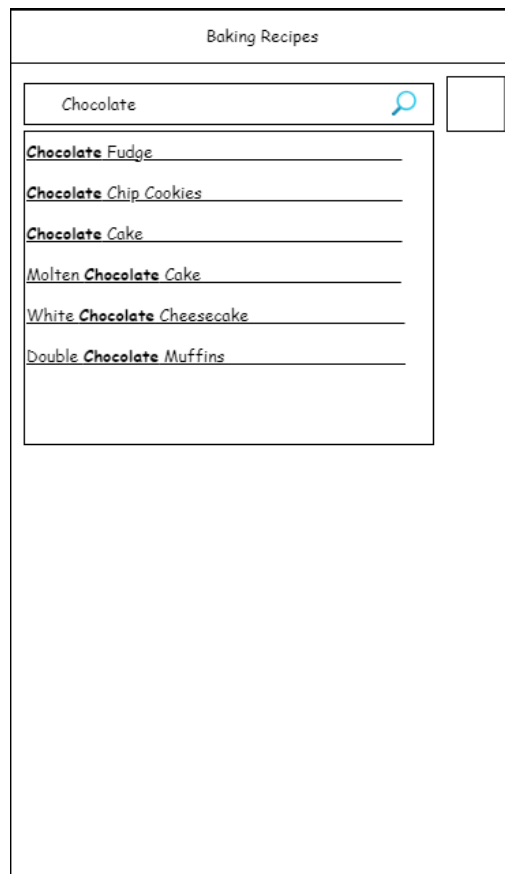


Figure3.5.17: View Recipes Page Interface

10. Search Recipes

They can search recipes in the search bar. The search results will display related recipes.



The image shows a search interface for 'Baking Recipes'. At the top, the title 'Baking Recipes' is centered. Below it is a search bar containing the text 'Chocolate' and a magnifying glass icon. To the right of the search bar is a small, empty square box. Below the search bar, a list of search results is displayed, each on a new line and underlined. The results are: 'Chocolate Fudge', 'Chocolate Chip Cookies', 'Chocolate Cake', 'Molten Chocolate Cake', 'White Chocolate Cheesecake', and 'Double Chocolate Muffins'. There is an additional empty line below the last result.

Figure 3.5.18: Search Interface

11. Recipes Details Page

Users can select and view recipes at the main page. When they click into the recipes, they can view recipes details including steps, preparation time, ingredients needed and nutritional facts. User can also chat to the dietitian to ask about the recipes.

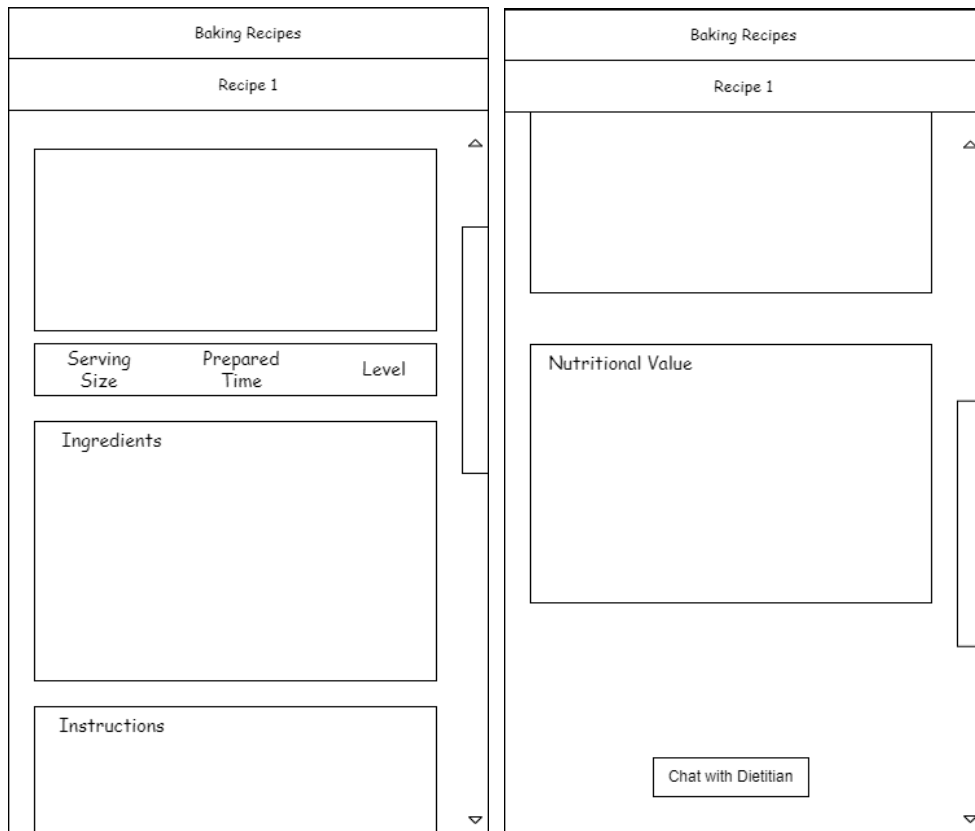
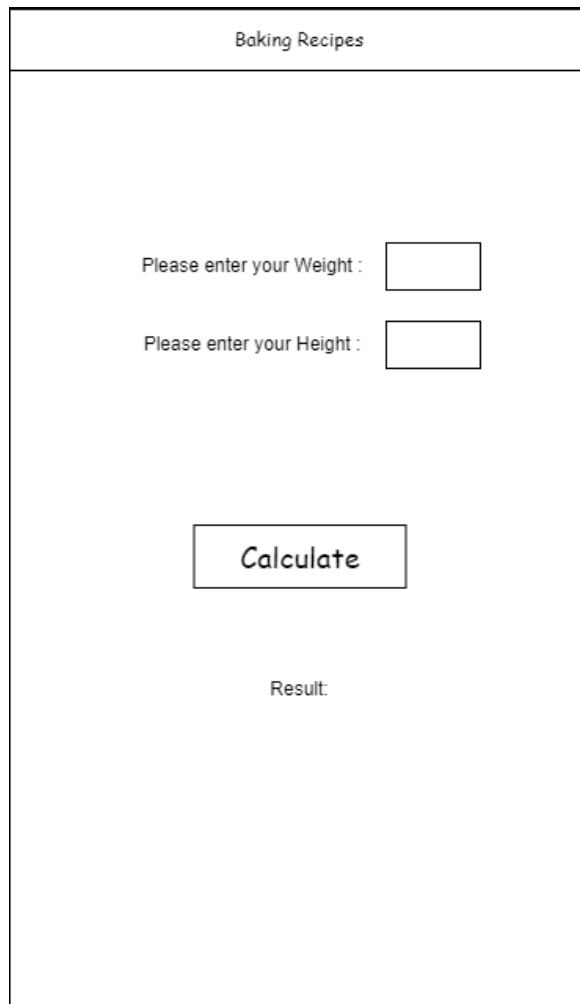


Figure 3.5.19: Recipe Details Page Interface

12. Calculate BMI Page

User can calculate their BMI in calculate BMI page. They need to insert their height and weight before clicking calculate button.



The image shows a user interface for calculating BMI. It is titled "Baking Recipes" at the top. Below the title, there are two input fields: "Please enter your Weight :" followed by a text box, and "Please enter your Height :" followed by another text box. Below these fields is a large button labeled "Calculate". At the bottom of the interface, there is a label "Result:".

Figure 3.5.20: Calculate BMI Page Interface

12. Add, Edit and Delete Recipe

Dietitian can add, edit, and delete recipes that they uploaded before in the application when they click on edit recipes button.

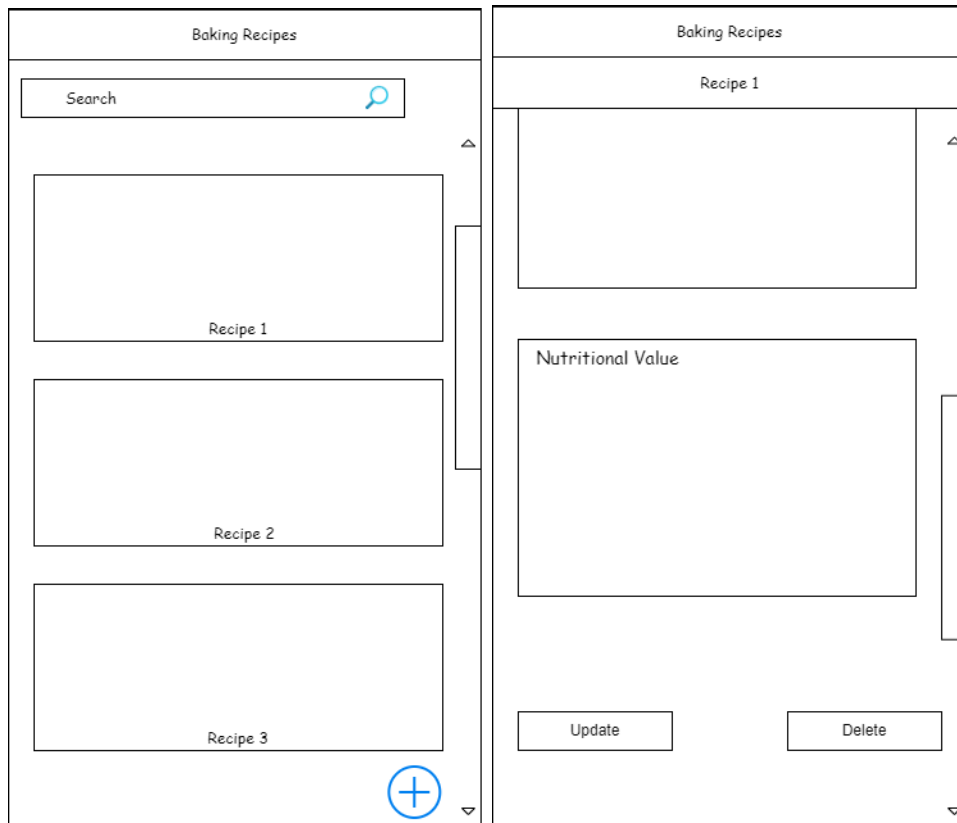


Figure 3.5.21: Add, Edit and Delete Recipes Interface

13. Dietitian Upload and Update Recipe

When dietitian add or edit recipes, they need to type all the details of the recipe in the text box and upload picture of the recipe. Next, click upload button to save the data.

Baking Recipe

Recipe Name _____

Time Needed _____

Recipe Steps _____

Recipe Ingredients _____

Recipe Nutritional Value _____

Recipe Serving Size _____

Select Image

Upload Recipe

Figure 3.5.22: Dietitian Upload and Update Recipe Interface

14. Chat Page

When user or dietitians click on the chat button, they can view the user/dietitian name list. When they click on the name, they can chat to each other.

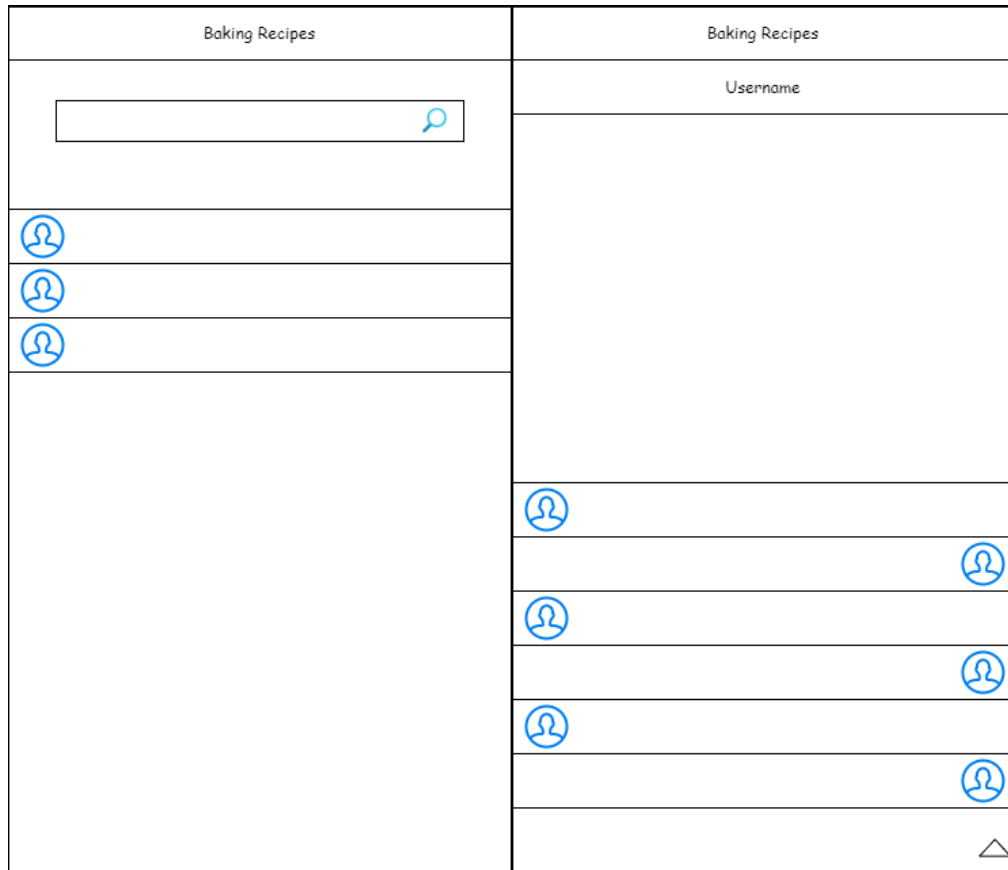


Figure 3.5.23: Chat Page Interface

3.6 Data Design

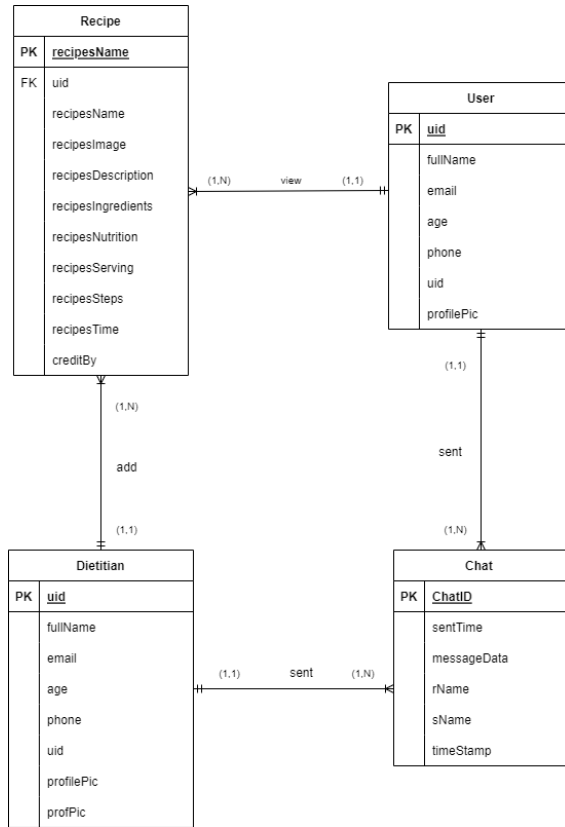


Figure 3.6.1: ERD

User

Field Name	Description	Data Type	Constraint
uid	User ID	VARCHAR(255)	PK
fullName	User's Name	VARCHAR(255)	
email	User Email	VARCHAR(255)	
phone	User Contact Number	INT(30)	
profilePic	User Profile Picture	VARCHAR(255)	

Table 3.6.1: User Data Dictionary

Recipe

Field Name	Description	Data Type	Constraint
recipesName	Recipe Name	VARCHAR(255)	PK
uid	Dietitian ID	VARCHAR(255)	FK
recipesImage	Recipe Image	VARCHAR(255)	
recipesDescription	Recipe Description	VARCHAR(255)	

recipesIngredient	Recipe Ingredient	VARCHAR(255)	
recipesNutrition	Recipe Nutritional Value	VARCHAR(255)	
recipesServing	Recipe Serving Size	VARCHAR(255)	
recipesSteps	Recipe Steps	LONGTEXT	
recipesTime	Recipe Prepared Time	VARCHAR(255)	
creditBy	Recipe credited by	VARCHAR(255)	

Table 3.6.2: Recipe Data Dictionary

Dietitian

Field Name	Description	Data Type	Constraint
uid	Dietitian ID	VARCHAR(255)	PK
fullName	Dietitian Name	VARCHAR(255)	
email	Dietitian Email	VARCHAR(255)	
phone	Dietitian Contact Number	VARCHAR(255)	

profilePic	Dietitian Profile Picture	VARCHAR(255)	
profPic	Dietitian Professional Certificate	VARCAHR(255)	

Table 3.6.3: Dietitian Data Dictionary

Chat

Field Name	Description	Data Type	Constraint
ChatID	Chat ID	VARCHAR(255)	PK
sentTime	Message Sent Time	VARCHAR(255)	
messageData	Message Context	VARCHAR(255)	
rName	Receiver Name	VARCHAR(255)	
sName	Sender Name	VARCHAR(255)	
timeStamp	Timestamp	VARCHAR(255)	

Table 3.6.4: Chat Data Dictionary

3.7 Design Prototype

1. Identity Page

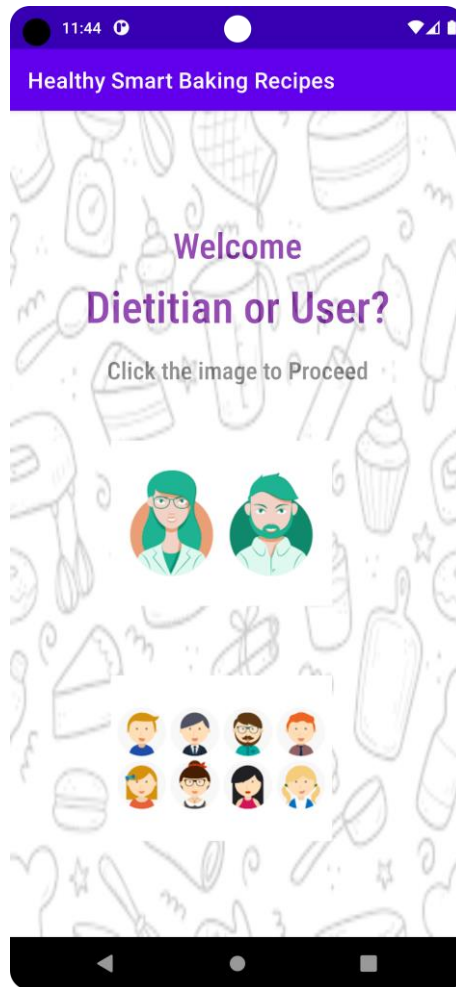


Figure 3.7.1: Identity Page Interface

Interface	Identity Page
Button	Identity choice button
Text Input	-

Previous Scene	-
Next Scene	User/Dietitian Login Page
Remarks	1. Users click on the button to choose their identity.

Table 3.7.1 Identity Page Interface

2. User Login Page

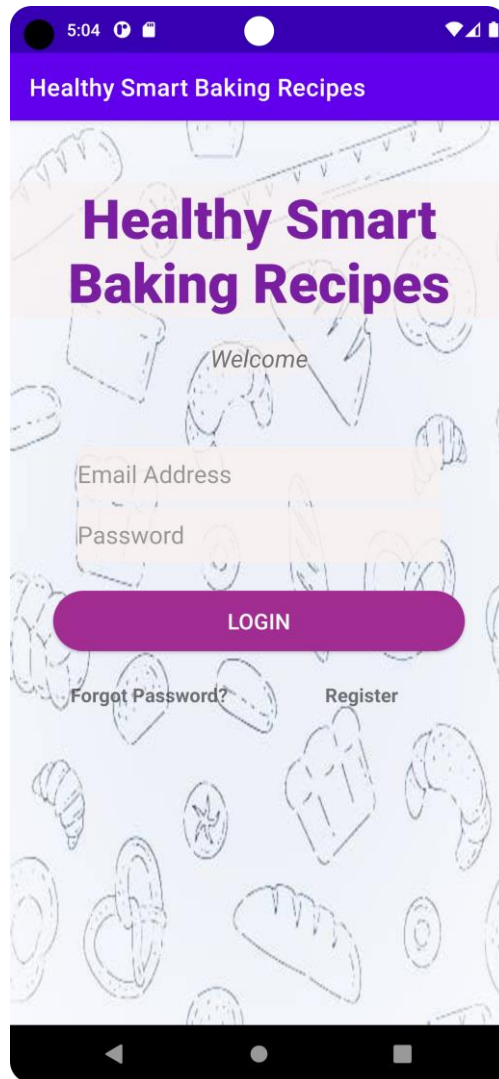


Figure 3.7.2: User Login Page Interface

Interface	User Login Page
Button	Login button, Forgot Password button, Register button
Text Input	Email, Password

Previous Scene	Identity Page
Next Scene	User Menu Page, User Registration Page, Forgot Password Page
Remarks	<ol style="list-style-type: none"> 1. User can login to their account from this page by enter correct email and password. 2. If the login is valid, they can proceed to User Menu page, else they need to register an account or change correct password.

Table 3.7.2 User Login Page Interface

3. Dietitian Login Page



Figure 3.7.3: Dietitian Login Page Interface

Interface	Dietitian Login Page
Button	Login button, Forgot Password button, Register button
Text Input	Email, Password

Previous Scene	Identity Page
Next Scene	Dietitian Menu Page, Dietitian Registration Page, Forgot Password Page
Remarks	<ol style="list-style-type: none"> 1. Dietitian can login to their account from this page by enter correct email and password. 2. If the login is valid, they can proceed to Dietitian Menu page, else they need to register an account or change correct password.

Table 3.7.3 Dietitian Login Page Interface

4. User Registration Page

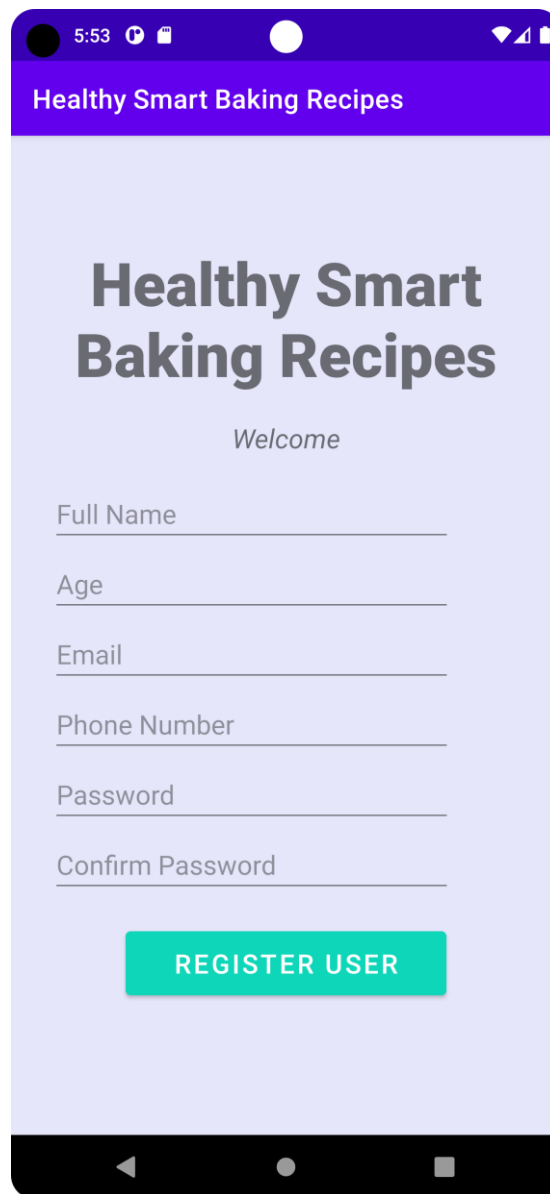


Figure 3.7.4: User Registration Page Interface

Interface	User Registration Page
Button	Register button
Text Input	Name, Age, Phone Number, Email, Password

Previous Scene	User Login Page
Next Scene	User Login Interface
Remarks	<ol style="list-style-type: none"> 1. For the user who do not have an account, they can sign up a new account in this registration page. 2. They need to insert their personal details such as name, phone number, email, password and click sign up button to finish the process. 3. After register, they need to re-login in the login page.

Table 3.7.4 User Registration Page Interface

5. Dietitian Registration Page

5:52

Healthy Smart Baking Recipes

Healthy Smart Baking Recipes

Welcome

Full Name

Age

Email

Phone Number

Password

Confirm Password

REGISTER

Figure 3.7.5: Dietitian Registration Page Interface

Interface	Dietitian Registration Page
Button	Register button
Text Input	Name, Age, Phone Number, Email, Password

Previous Scene	Dietitian Login Page
Next Scene	Dietitian Login Page
Remarks	<ol style="list-style-type: none"> 1. For the dietitian who do not have an account, they can sign up a new account in this registration page. 2. They need to insert their personal details such as name, phone number, email, password and click sign up button to finish the process. 3. After register, they need to login in the login page.

Table 3.7.5 Dietitian Registration Page Interface

6. Forgot Password Page

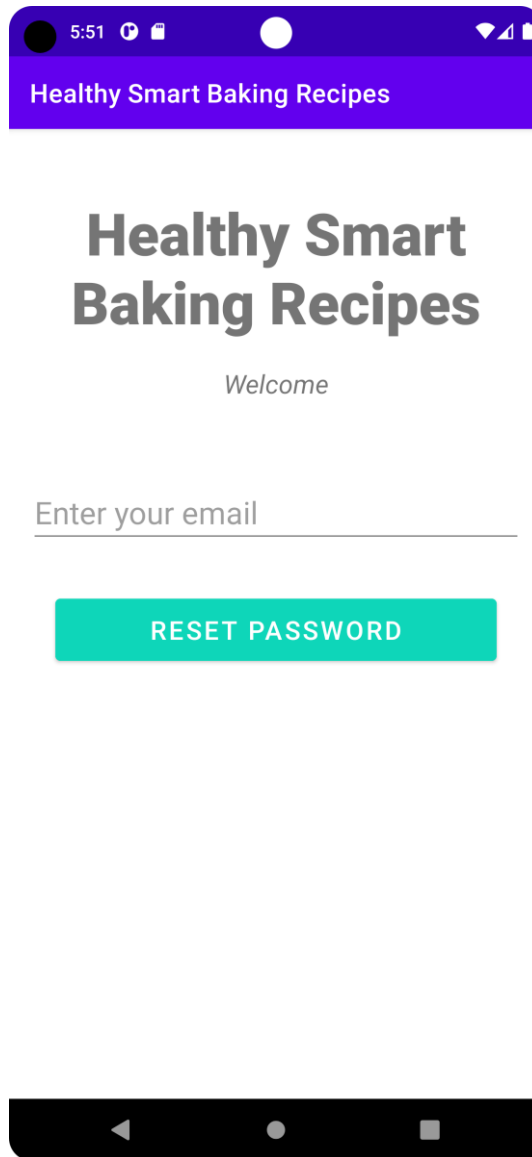


Figure 3.7.6: Forgot Password Page Interface

Interface	Forgot Password Page
Button	Reset button
Text Input	Email

Previous Scene	User Login Page, Dietitian Login Page
Next Scene	User Login Page, Dietitian Login Page
Remarks	<ol style="list-style-type: none"> 1. If user/dietitian forgot their password, by clicking forgot password button, they will be directed to this forgot password page. 2. User/Dietitian can reset their password by email verification. 3. After reset password, they can login to their account in login page.

Table 3.7.6 Forgot Password Page Interface

7. User Menu Page

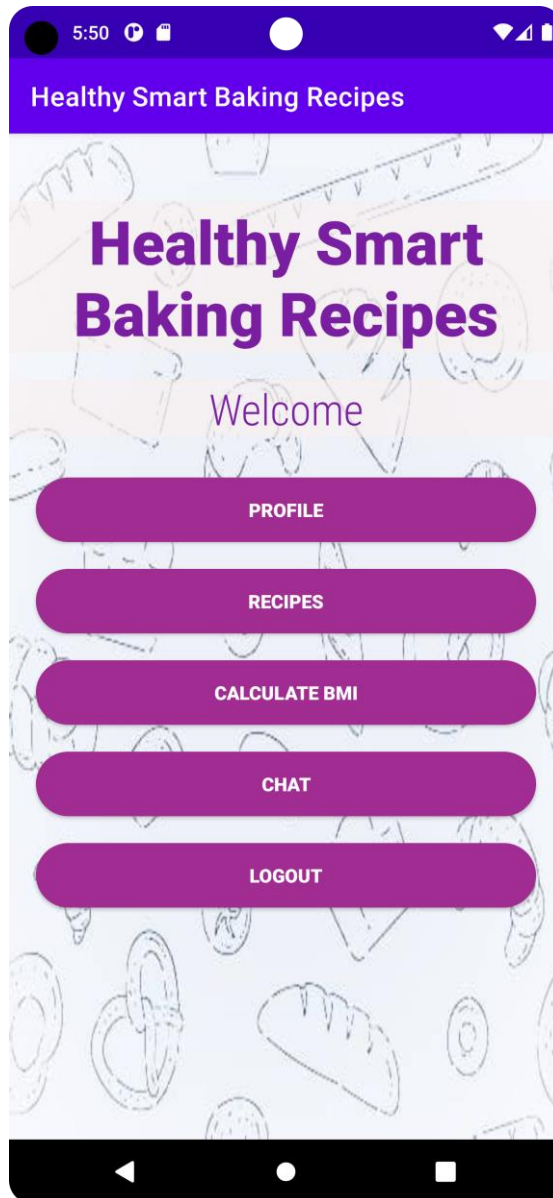


Figure 3.7.7: User Menu Page Interface

Interface	User Menu Page
Button	Profile button, Recipe button, Calculate BMI button, Chat button, Logout button

Text Input	-
Previous Scene	User Login page
Next Scene	Profile Page, View Recipe Page, Calculate BMI Page, Chat Page
Remarks	<ol style="list-style-type: none"> 1. User can access this page only when they successfully login their account. User can choose their activity in this menu page.

Table 3.7.7 User Menu Page Interface

8. Dietitian Menu Page



Figure 3.7.8: Dietitian Menu Page Interface

Interface	Dietitian Menu Page
Button	Profile button, Recipe button, Edit Recipe button, Chat button, Logout button
Text Input	-

Previous Scene	Dietitian Login page
Next Scene	Profile Page, View Recipe Page, Edit Recipe Page, Chat Page
Remarks	<ol style="list-style-type: none"> 1. Dietitian can access this page only when they successfully login their account. Dietitian can choose their activity in this menu page.

Table 3.7.8 Dietitian Menu Page Interface

9. User Profile Page

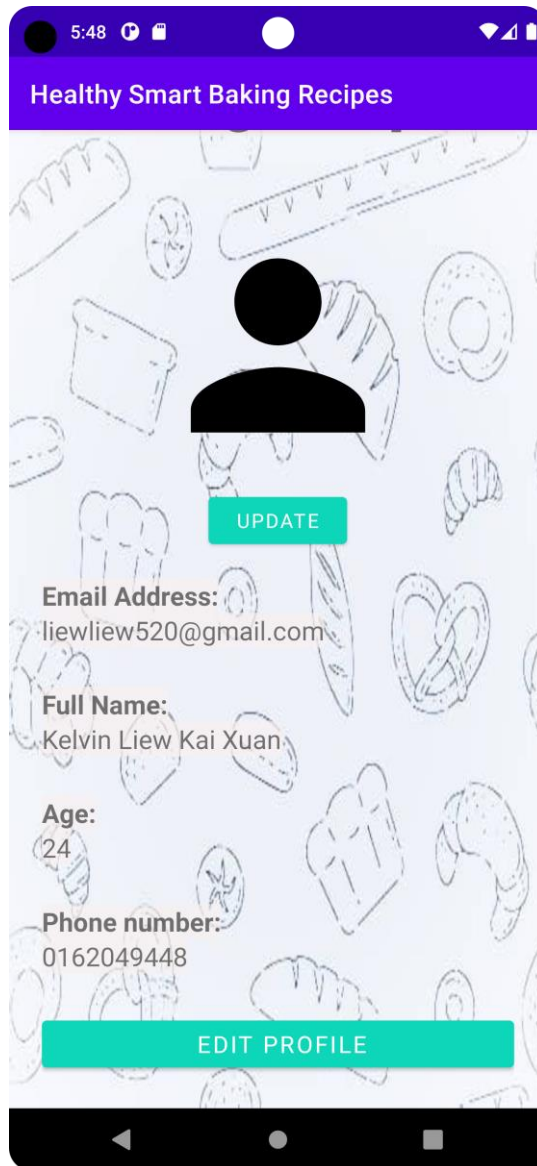


Figure 3.7.9: User Profile Page Interface

Interface	User Profile Page
Button	Upload Profile Image Button, Edit Profile Button
Text Input	-

Previous Scene	User Menu Page
Next Scene	Manage User Profile Page
Remarks	1. User can upload their profile picture by clicking upload button.

Table 3.7.9 User Profile Page Interface

10. Dietitian Profile Page

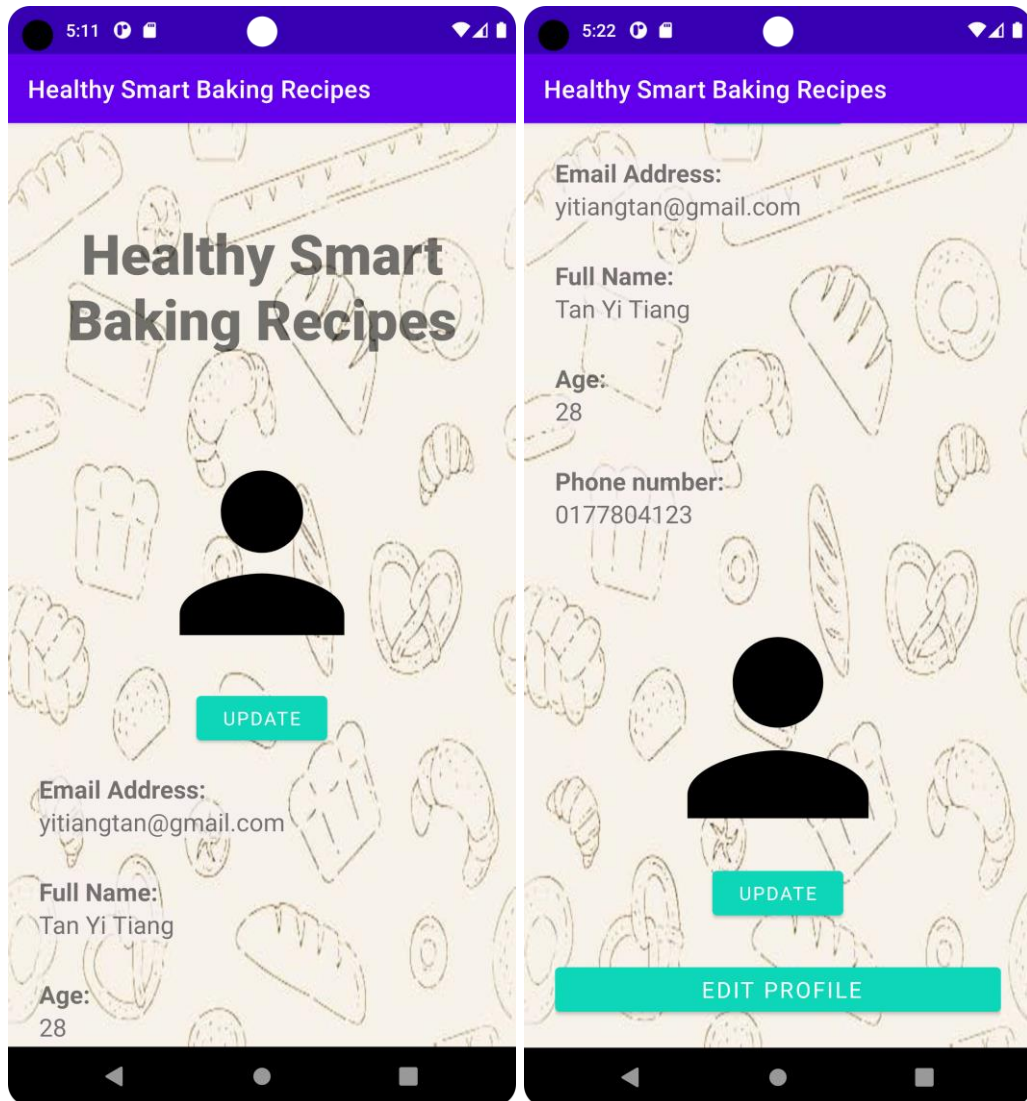


Figure 3.7.10: Dietitian Profile Page Interface

Interface	Dietitian Profile Page
Button	Upload Profile Image Button, Update Professional Certificate Button, Edit Profile Button
Text Input	-

Previous Scene	Dietitian Menu Page
Next Scene	Manage Dietitian Profile Page
Remarks	<ol style="list-style-type: none"> 1. Dietitian can change their profile picture by clicking upload button. 2. Dietitian are required to upload their professional certificate in the profile page.

Table 3.7.10 Dietitian Profile Page Interface

11. Manage User Profile Page

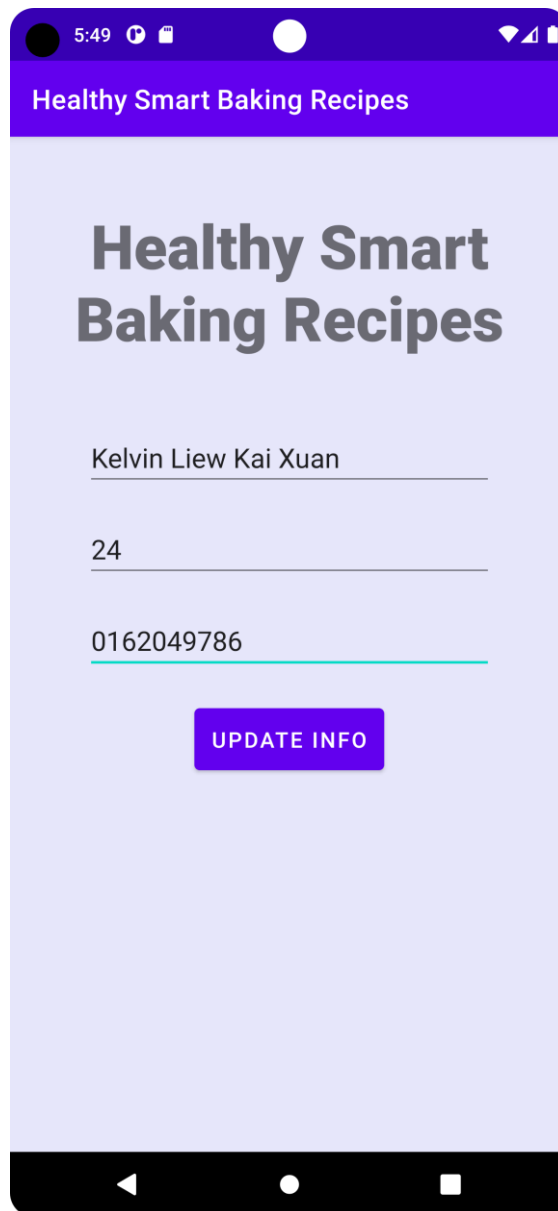


Figure 3.7.11: Manage User Profile Page Interface

Interface	Manage User Profile Page
Button	Update Info button
Text Input	Name, Email, Phone Number

Previous Scene	User Profile Page
Next Scene	-
Remarks	<ol style="list-style-type: none"> 1. User can change their personal information through this manage account page. 2. After inserting new data, they can click save button to save them into database.

Table 3.7.11 Manage User Profile Page Interface

12. Manage Dietitian Profile Page

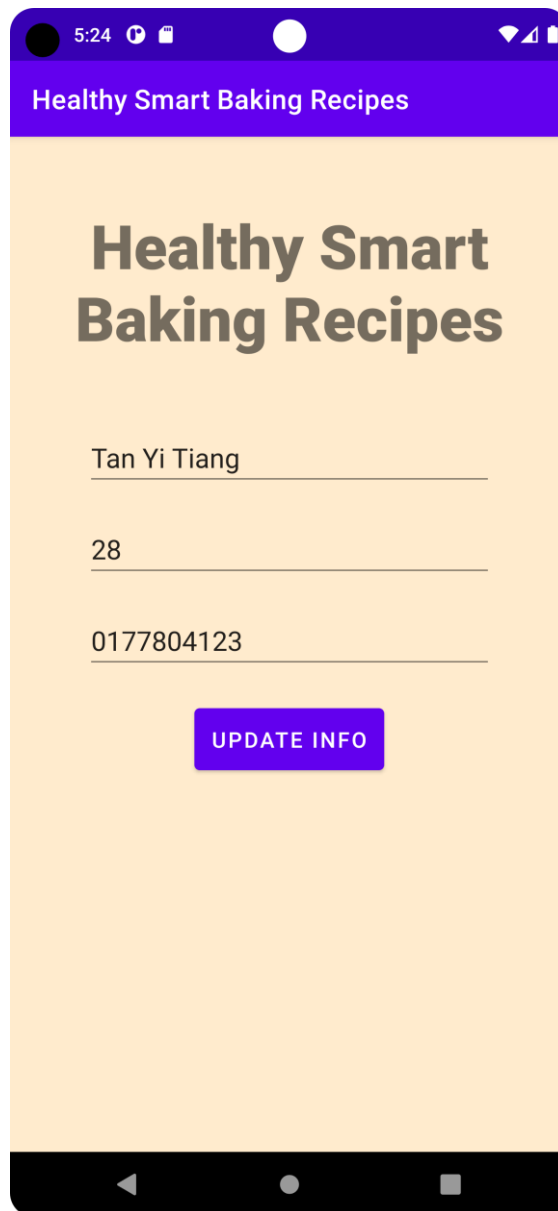


Figure 3.7.12: Manage Dietitian Profile Page Interface

Interface	Manage Dietitian Profile Page
Button	Update Info button
Text Input	Name, Email, Phone Number

Previous Scene	Dietitian Profile Page
Next Scene	-
Remarks	<ol style="list-style-type: none"> 1. Dietitian can change their personal information through this manage account page. 2. After inserting new data, they can click save button to save them into database.

Table 3.7.12 Manage Dietitian Profile Page Interface

13. View Recipe Page

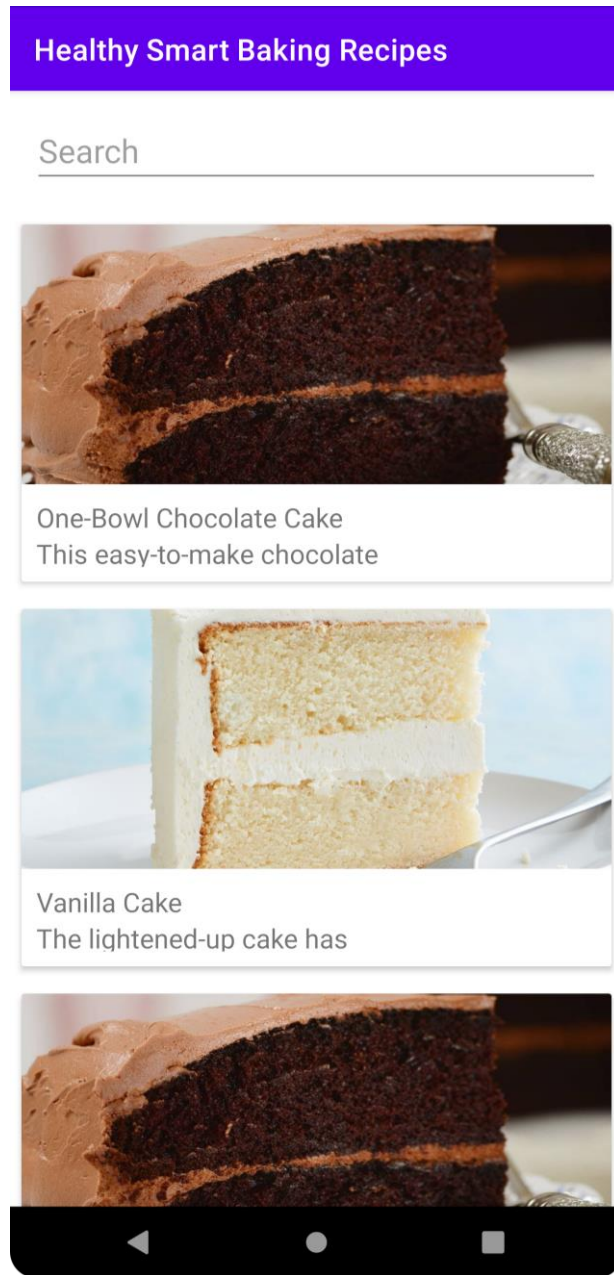


Figure 3.7.13: View Recipe Page Interface

Interface	View Recipe Page
Button	Image button

Text Input	Search bar
Previous Scene	User Menu page, Dietitian Menu Page
Next Scene	View Recipe Details Page
Remarks	<ol style="list-style-type: none"> 1. User/Dietitian can access this page only when they successfully login their account. User can choose their recipes in this page to view recipe details. 2. User/Dietitian can use the search bar to find the recipe they need by insert the keyword/ingredient name.

Table 3.7.13 View Recipe Page Interface

14. Recipe Details Page

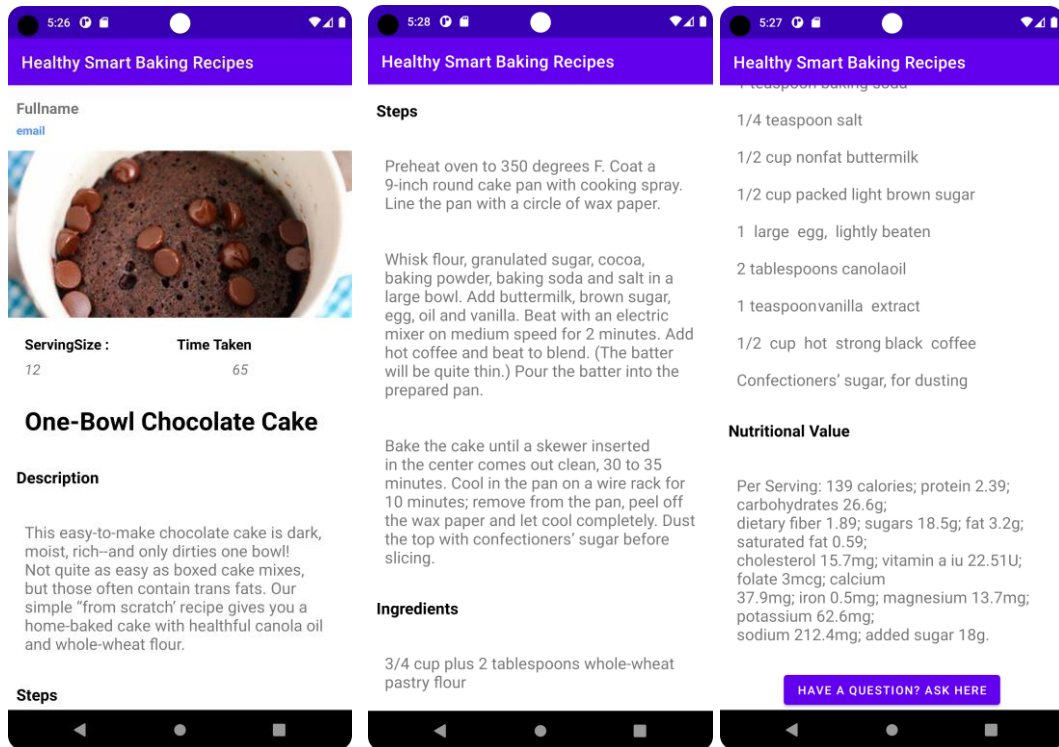


Figure 3.7.14: Recipe Details Page Interface

Interface	Recipe Details Page
Button	Chat a Dietitian button
Text Input	-
Previous Scene	View Recipe Page
Next Scene	Chat Name List Page
Remarks	1. User can view recipe details in this page in this interface.

	<ol style="list-style-type: none"><li data-bbox="555 197 1326 338">2. This interface contains serving size of recipes, prepared time, difficult level, ingredients, and steps of baking process.<li data-bbox="555 360 1369 450">3. User can also communicate to dietitian via this interface by clicking chat a dietitian button.
--	---

Table 3.7.14 Recipe Details Page Interface

15. Calculate BMI Page

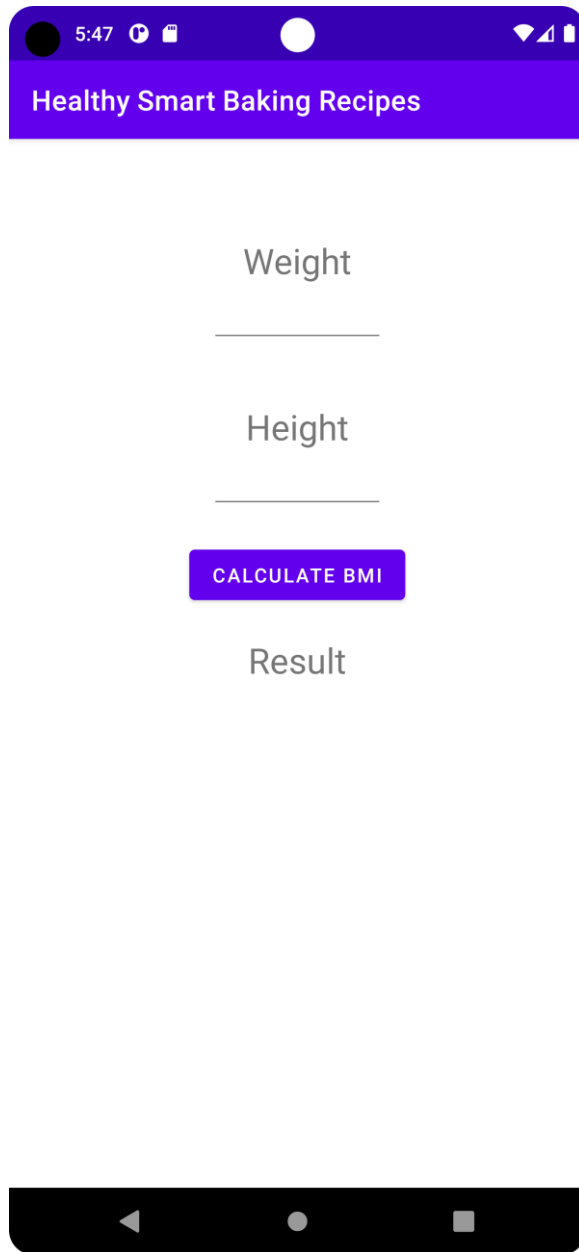


Figure 3.7.15: Calculate BMI Page Interface

Interface	Calculate BMI Page
Button	Calculate BMI button

Text Input	Weight, Height
Previous Scene	User Menu Page
Next Scene	-
Remarks	1. User can calculate their BMI by inserting their weight and height and click on calculate button.

Table 3.7.15 Calculate BMI Page Interface

16. Dietitian Edit Recipe Page

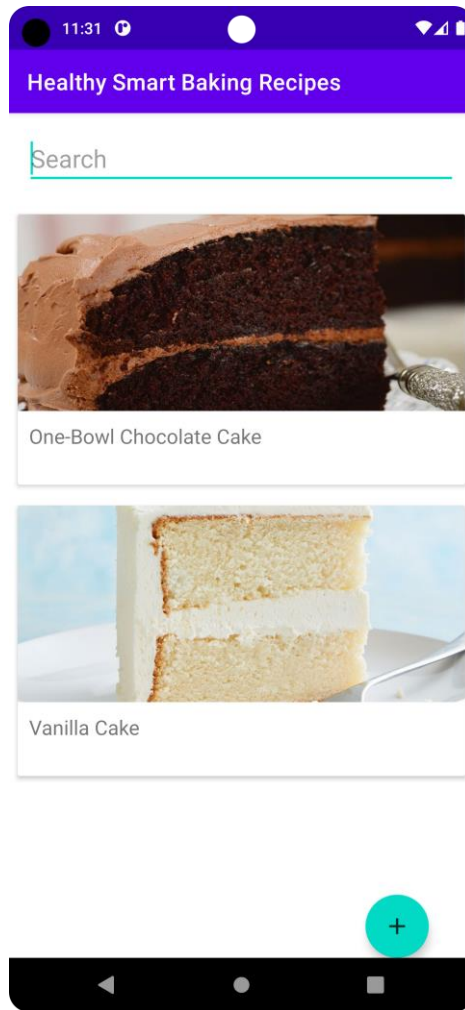


Figure 3.7.16: Dietitian Edit Recipe Page Interface

Interface	Dietitian Edit Recipe Page
Button	Add button
Text Input	-
Previous Scene	Dietitian Menu Page

Next Scene	Dietitian Upload Page
Remarks	<ol style="list-style-type: none">1. Dietitian can click on add button to insert new recipes.2. They will be directed to an upload page.

Table 3.7.16 Dietitian Edit Recipe Page Interface

17. Dietitian Upload Page

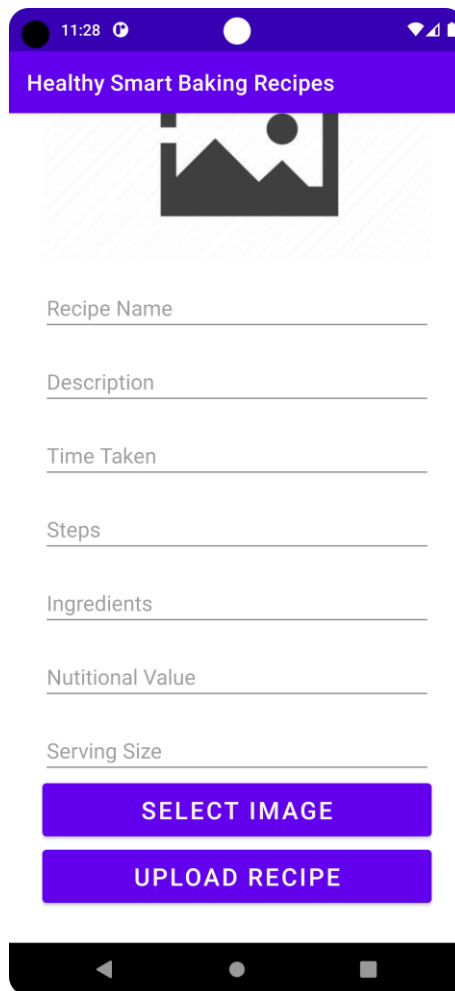


Figure 3.7.17: Dietitian Upload Page Interface

Interface	Dietitian Upload Page
Button	Select Image button, Upload Button.
Text Input	Recipes name, Recipes Ingredients, Recipe Time Taken, Recipes Serving Size, Recipes Steps, Recipes Nutritional Value, Recipes Credit By

Previous Scene	Dietitian Add Recipe Page
Next Scene	-
Remarks	<ol style="list-style-type: none"> 1. Dietitian inserts all the recipes details in the text box given. 2. Dietitian can select to upload picture of the recipes. 3. Dietitian click save button to upload recipes.

Table 3.7.17 Dietitian Upload Page Interface

18. Dietitian Edit Recipe Details Page

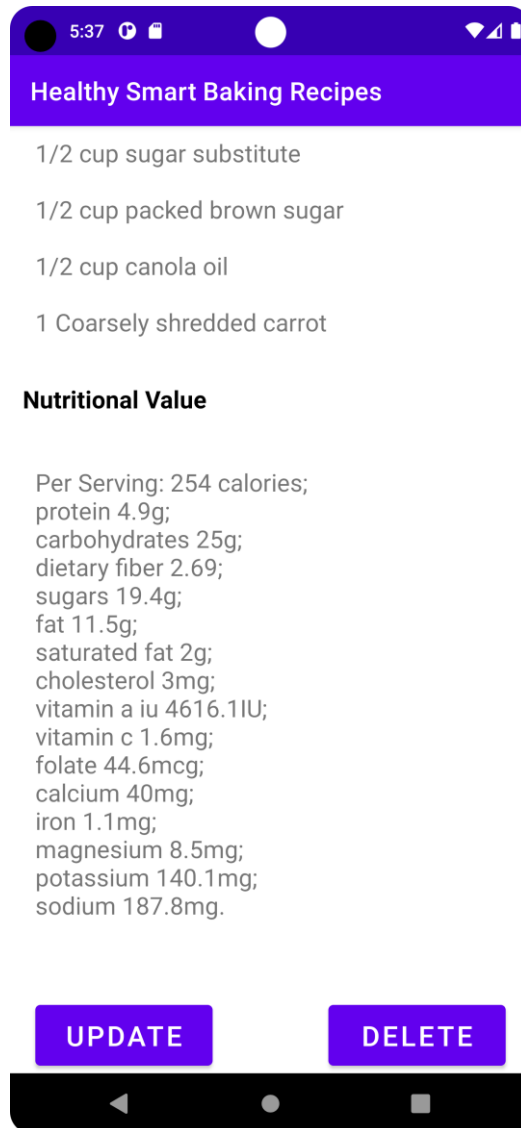


Figure 3.7.18: Dietitian Edit Recipe Details Page Interface

Interface	Dietitian Edit Recipe Details Page
Button	Update Button, Delete Button.
Text Input	-

Previous Scene	Dietitian Edit Recipe Page
Next Scene	Dietitian Update Page
Remarks	<ol style="list-style-type: none"> 1. Dietitian can click Update button to update recipes. 2. Dietitian can click Delete button to delete the recipes.

Table 3.7.18 Dietitian Edit Recipe Details Page Interface

19. Dietitian Update Page

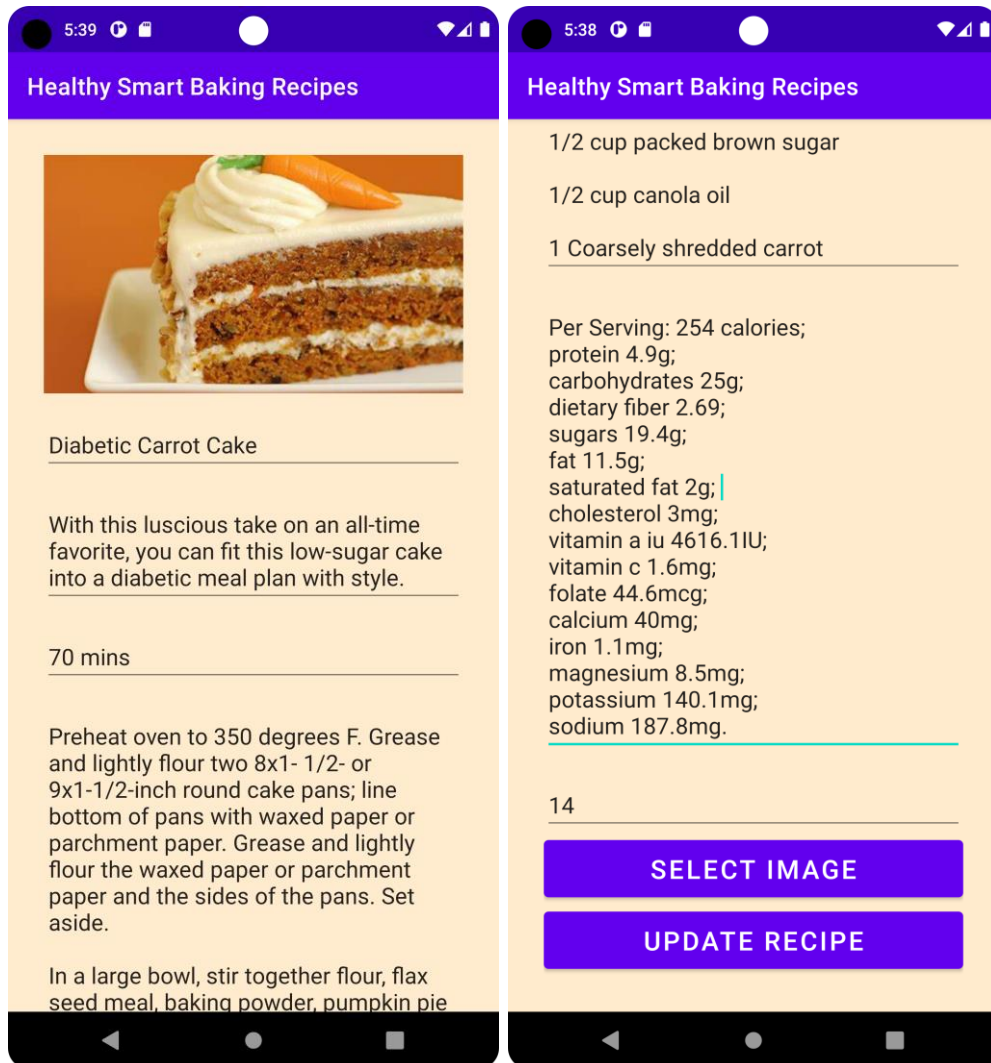


Figure 3.7.19: Dietitian Update Page Interface

Interface	Dietitian Update Page
Button	Select Image button, Update Button.
Text Input	Recipes name, Recipes Ingredients, Recipe Time Taken, Recipes Serving Size, Recipes Steps, Recipes Nutritional Value, Recipes Credit By

Previous Scene	Dietitian Recipe Edit Page
Next Scene	-
Remarks	<ol style="list-style-type: none"> 1. Dietitian inserts all the recipes details in the text box given. 2. Dietitian can select to upload picture of the recipes. 3. Dietitian click save button to upload recipes.

Table 3.7.19 Dietitian Update Page Interface

20. Chat Name List Page

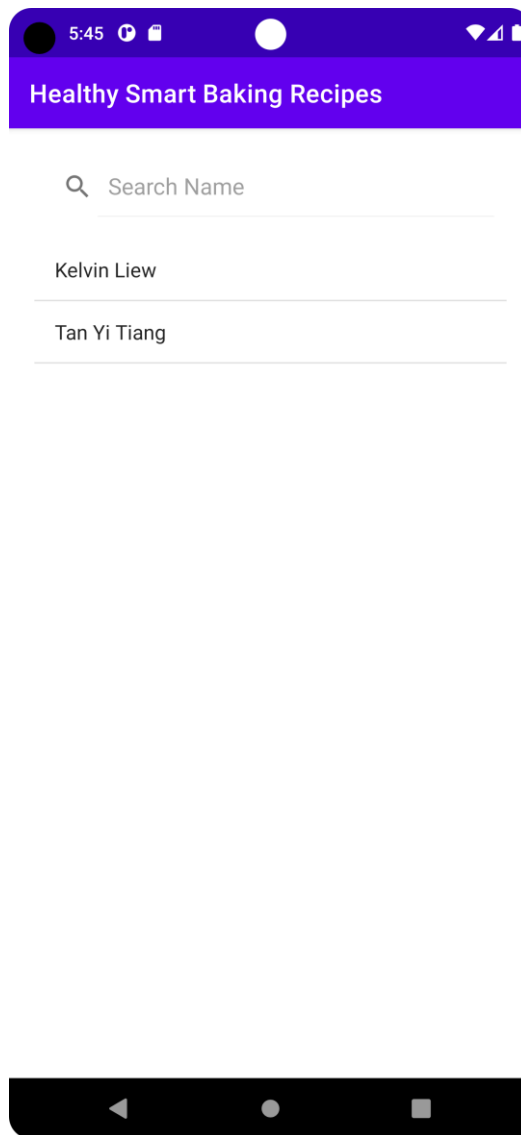


Figure 3.7.20: Chat Name List Page Interface

Interface	Chat Name List Page
Button	Name Button
Text Input	-

Previous Scene	Recipe Details Page, User Menu Page, Dietitian Menu Page
Next Scene	Chat Page
Remarks	<ol style="list-style-type: none"> 1. User can select the Dietitian name to chat with them. 2. Dietitian can select username to reply to their chat.

Table 3.7.20 Chat Name List Page Interface

21. Chat Page

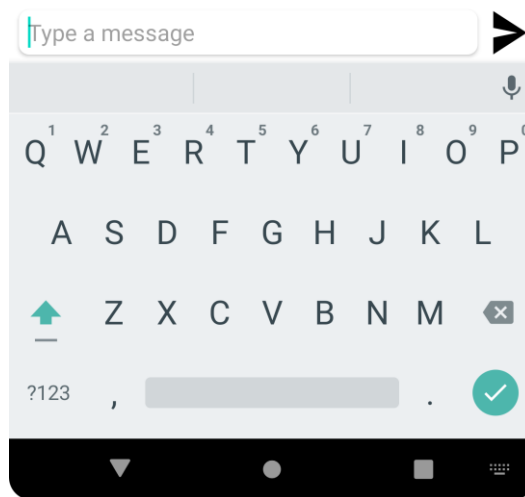
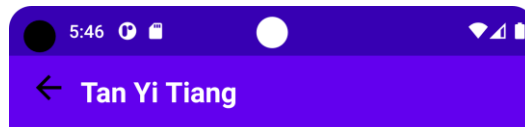


Figure 3.7.21: Chat Page Interface

Interface	Chat Page
Button	Sent Button.
Text Input	Messages

Previous Scene	Chat Name List Page
Next Scene	-
Remarks	1. User/Dietitian can type message in the text box and click sent button to send the message.

Table 3.7.21 Chat Page Interface

3.8 Testing Plan

User Acceptance Test (UAT) form

No	Module	Activities	Status			Comments
1.	Registration	User Sign Up	Yes		No	
2.	Login	User Login	Yes		No	
3.		User Forgot Password	Yes		No	
4.		User Logout	Yes		No	
5.	Manage User and Dietitian Profile	Change Profile Picture	Yes		No	
6.		Update Name	Yes		No	
7.		Update Age	Yes		No	
8.		Update Phone Number	Yes		No	
9.		(Dietitian) Upload Professional Certificate	Yes		No	
10.	Search Recipe	Search Bar	Yes		No	

11.	Select and View Recipes	View Recipe Details	Yes		No		
12.		Chat a Dietitian	Yes		No		
13.	Add, Edit, Delete Recipes	Add Recipes	Yes		No		
14.		Edit Recipes	Yes		No		
15.		Delete Recipes	Yes		No		
16.	Chat	Select name to Chat	Yes		No		
17.		Sent Messages	Yes		No		

Table 3.8 Testing Plan

This test has been performed by:

Name:

Signature:

Date:

3.9 Potential Use of Proposed Solution

The use of this mobile application is to help people who are new in baking projects. It provides easy interfaces for user to find recipes. People can use the application to find recipes, get ideas in their baking projects and make delicious baked goods. In the long run, if the application is working well, it might be uploaded in Google Store for user worldwide.

3.10 Hardware and Software Specification

3.10.1 Hardware Specification

Hardware	Version	Description	Purpose of use
LAPTOP-GRB0MR2U	Window 10	A device which used to install software and develop code.	<ul style="list-style-type: none">- To generate documents, develop Baking Recipes Mobile Application.- To display and perform the work done.- To run and test the application built.
Mi 9T Pro	Android 10	A device which used to install the application and run it.	<ul style="list-style-type: none">- To install the mobile application and use it as a real simulator to run the application.

Table 3.10.1 Hardware Specification

3.10.2 Software Specification

Software	Version	Description	Purpose of use
Microsoft Word	Microsoft Office 365 Pro	A software to record and type the report.	<ul style="list-style-type: none">- To do documentation of the project.
Android Studio	3.1	A software to build the android system application.	<ul style="list-style-type: none">- To develop and built the application for the project.- Design code for the mobile application.
Draw.io	18.0.6.0	A software to create and sketch diagrams and user interface.	<ul style="list-style-type: none">- To design and sketch the diagrams and flow of the projects.- To design interface of the project.
Firebase	Firebase	A database tool to save all the data used in the application.	<ul style="list-style-type: none">- To save the data insert in the application.

Table 3.10.2 Software Specification

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this chapter, we will discuss the phases of implementation, testing, and integration. A mobile application based on Healthy Smart Baking Recipes was created with Android Studio. Several features are available for users of the Healthy Smart Baking Recipe Mobile Application. Users can search for recipes, select recipes, and view recipe details. Additionally, the recipes allow them to add shopping lists and chat with dietitians. A dietitian can update, edit, and delete recipes that have been uploaded. In addition, dietitians can also view recipes as users.

During this session, we will cover the development and testing of the developed system.

4.2 Development of Application

The following section will describe the development process, including testing tools and results.

4.2.1 Development Tools

In software development, a development tool is a computer program used to create, debug, maintain, or otherwise support additional programs. The chosen

development tool in this project is Android Studio. Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. This tool provides an Android-friendly Gradle-based build system, a feature-rich emulator, and a unified environment to help developers develop a wide variety of Android devices. The complete Healthy Smart Baking Recipe Mobile Application is developed using Android Studio Dolphin // 2021.3.1 version.



Figure 4.2.1 Android Studio 2021.3.1 Dolphin

4.2.2 Create and Development Project

To develop Healthy Smart Baking Recipe Mobile Application, an Android Studio project is being created.

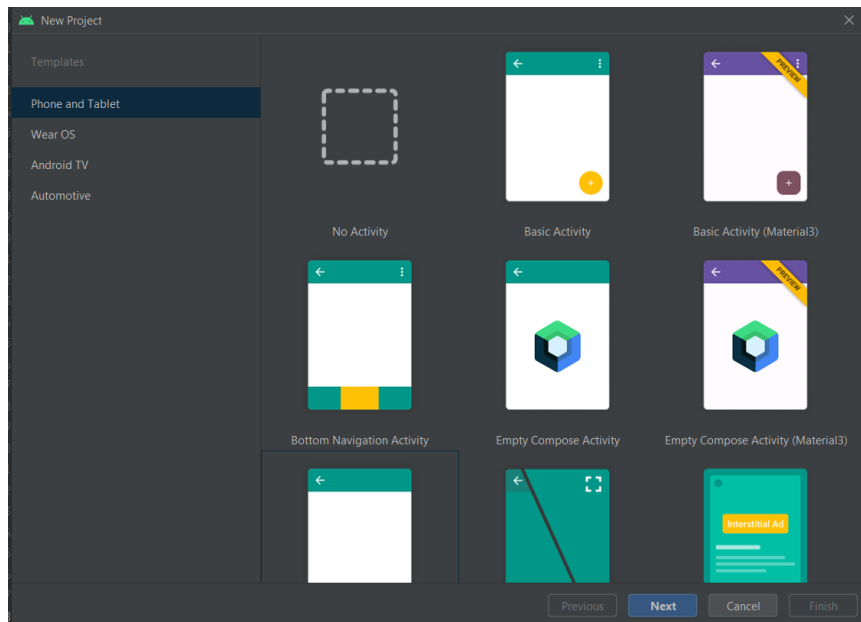


Figure 4.2.2 Create Project

4.2.3 Identity Module

In Identity Module, the user can choose their own identity as dietitian or user. Dietitians are allowed to update, edit, delete recipes and chat to the users while users are allowed to view the recipes provided by dietitian and chat to dietitians. Besides uploading recipes, dietitians can view other recipes as users. When the user chooses their identity, they will be directed to the Login Module.

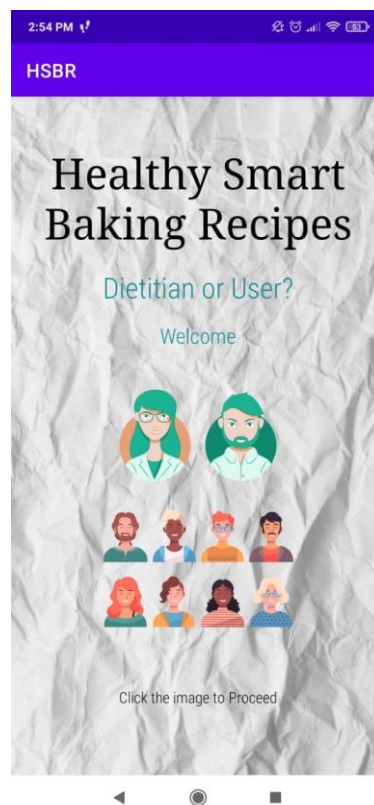


Figure 4.2.3: Interface of Identity Module


```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_identity);
}

public void DietitianLogin(View view) {
    Intent intents = new Intent( packageContext: this, DietitianLogin.class);
    startActivity(intents);
}

public void Login(View view) {
    Intent intents = new Intent( packageContext: this, MainActivity.class);
    startActivity(intents);
}
```

Figure 4.2.4: Code of Identity Module

4.2.4 Dietitian Login Module

In Dietitian Login Module, dietitians are required to login with verified email and password. They will receive a verification email for the first-time login to verify their account. When the dietitian successfully login into the system, the first interface appears would be the dietitian menu page.



Figure 4.2.5: Dietitian Login Module Interface

```
mAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
    @Override  
    public void onComplete(@NonNull Task<AuthResult> task) {  
  
        if(task.isSuccessful){  
            FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();  
  
            if(user.isEmailVerified()){  
                //redirect to user profile  
                startActivity(new Intent( packageContext: DietitianLogin.this, DietitianMenu.class));  
            }else{  
                user.sendEmailVerification();  
                Toast.makeText( context: DietitianLogin.this, text: "Check your email to verify your account!", Toast.LENGTH_SHORT).show();  
            }  
        }else{  
            Toast.makeText( context: DietitianLogin.this, text: "Failed to login! Please check your credentials", Toast.LENGTH_LONG).show();  
        }  
    }  
});
```

Figure 4.2.6: Code of Dietitian Login Module

4.2.5 User Login Module

In User Login Module, users are required to login with verified email and password. User will receive a verification email for the first-time login to verify their account. When the user successfully login into the system, the first interface appears would be the user menu page.



Figure 4.2.7: User Login Module Interface

```
mAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
    @Override  
    public void onComplete(@NonNull Task<AuthResult> task) {  
        if(task.isSuccessful){  
            FirebaseUser user = FirebaseAuth.getInstance().getCurrentUser();  
            if(user.isEmailVerified()){  
                //redirect to user profile  
                startActivity(new Intent( packageContext: MainActivity.this, UserMenu.class));  
            }else{  
                user.sendEmailVerification();  
                Toast.makeText( context: MainActivity.this, text: "Check your email to verify your account!", Toast.LENGTH_SHORT).show();  
            }  
        }else{  
            Toast.makeText( context: MainActivity.this, text: "Failed to login! Please check your credentials", Toast.LENGTH_LONG).show();  
        }  
    }  
});
```

Figure 4.2.8: Code of User Login Module

4.2.6 User and Dietitian Registration Module

Users/Dietitians who have not registered for an account must first register as a user in the application. Users/Dietitians are required to insert their personal information such as name, mobile phone number, email, password and confirm password. Users must fill up all the information otherwise they cannot register the account. If the email is already existed, the system will display wrong registration message.



The screenshot displays the registration interface for the HSBR application. At the top, a purple header bar contains the text "HSBR". Below the header, the title "Healthy Smart Baking Recipes" is prominently displayed in a large, black, serif font. Underneath the title, the word "Welcome" is written in a smaller, italicized font. The registration form consists of six white input fields with rounded corners, stacked vertically. The fields are labeled "Full Name", "Age", "Email", "Phone Number", "Password", and "Confirm Password". At the bottom of the form, there is a teal-colored button with the text "REGISTER" in white capital letters. The background of the interface features abstract, overlapping shapes in shades of orange, brown, and grey. The top of the screen shows a status bar with the time "3:01 PM" and various system icons. At the bottom, there are three small navigation icons: a back arrow, a home circle, and a recent apps square.

Figure 4.2.9: Registration Module Interface

```

 mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {

            if(task.isSuccessful()){

                User user = new User(fullName, age, email, phone, FirebaseAuth.getInstance().getUid());

                FirebaseDatabase.getInstance().getReference( path: "Dietitians")
                    .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                    .setValue(user).addOnCompleteListener(new OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void> task) {

                            if(task.isSuccessful()){
                                Toast.makeText( context: DietitianRegistration.this, text: "User has been registered successfully",
                                    progressBar.setVisibility(View.GONE);
                                    startActivity(new Intent( packageContext: DietitianRegistration.this, DietitianLogin.class));
                            }else{

                                Toast.makeText( context: DietitianRegistration.this, text: "Failed to register! Try Again!", Toast.
                                    progressBar.setVisibility(View.GONE);
                                }
                            }
                        }
                    })
            }
        }
    })

```

```

 mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {

            if(task.isSuccessful()){

                User user = new User(fullName, age, email, phone, FirebaseAuth.getInstance().getUid()); //FirebaseAuth.getInstance().getCur

                FirebaseDatabase.getInstance().getReference( path: "Users")
                    .child(FirebaseAuth.getInstance().getCurrentUser().getUid())
                    .setValue(user).addOnCompleteListener(new OnCompleteListener<Void>() {
                        @Override
                        public void onComplete(@NonNull Task<Void> task) {

                            if(task.isSuccessful()){
                                Toast.makeText( context: RegisterUser.this, text: "User has been registered successfully", Toast.LENGTH_
                                    progressBar.setVisibility(View.GONE);
                                    startActivity(new Intent( packageContext: RegisterUser.this, MainActivity.class));
                            }else{

                                Toast.makeText( context: RegisterUser.this, text: "Failed to register! Try Again!", Toast.LENGTH_SHORT).
                                    progressBar.setVisibility(View.GONE);
                                }
                            }
                        }
                    })
            }
        }
    })

```

Figure 4.2.10: Code for Registration Module

4.2.7 Forgot Password Page

If users/dietitians forgot their password, they could click Forgot Password button. The users require to key in their email and submit to get a new password reset chance.

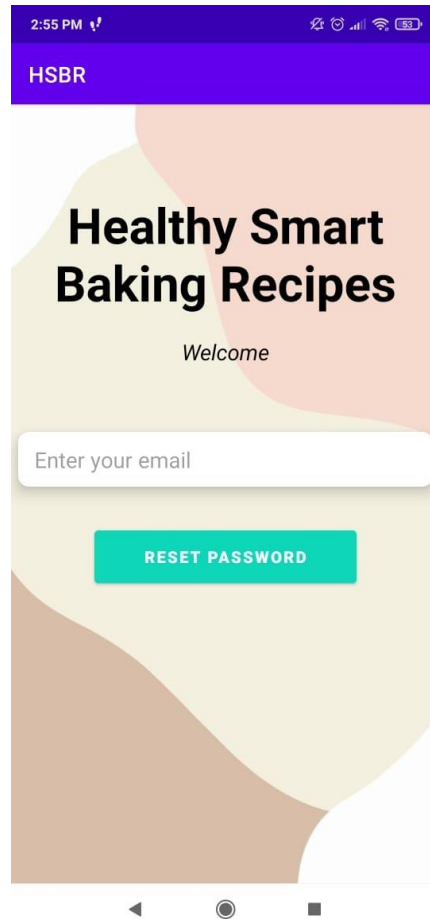


Figure 4.2.11: Forgot Password Page

```
auth.sendPasswordResetEmail(email).addOnCompleteListener(new OnCompleteListener<Void>() {  
    @Override  
    public void onComplete(@NonNull Task<Void> task) {  
  
        if(task.isSuccessful()){  
            Toast.makeText(context, ForgotPassword.this, text: "Check your email to reset your password!", Toast.LENGTH_LONG).show();  
        }else{  
            Toast.makeText(context, ForgotPassword.this, text: "Please try again!", Toast.LENGTH_LONG).show();  
        }  
    }  
});
```

Figure 4.2.12: Code for Forgot Password Page

4.2.8 User Menu Page

After successfully login, the user will be directed to their user menu. There are few functions in the user menu which are profile, recipes, BMI, chat, and logout functions.



Figure 4.2.13: User Menu Page

```

goProfile.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents = new Intent( packageContext: UserMenu.this, ProfileActivity.class);
        startActivity(intents);
    }
});

goMain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents = new Intent( packageContext: UserMenu.this, UserRecipe.class);
        startActivity(intents);
    }
});

goOut.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent( packageContext: UserMenu.this, Identity.class));
    }
});

goBMI.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents =(new Intent( packageContext: UserMenu.this, CalcBMI.class));
        startActivity(intents);
    }
});

goChat.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents =(new Intent( packageContext: UserMenu.this, UserSendMessage.class));
        startActivity(intents);
    }
});

```

Figure 4.2.14: Code for User Menu Page

4.2.9 Dietitian Menu Page

After successfully login, the dietitian will be directed to their dietitian menu. There are few functions in the dietitian menu which are profile, recipes, edit, chat, and logout functions.



Figure 4.2.15: Dietitian Menu Page

```

goProfile.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents = new Intent( packageContext: DietitianMenu.this, DietitianProfile.class);
        startActivity(intents);
    }
});

goMain.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents = new Intent( packageContext: DietitianMenu.this, UserRecipe.class);
        startActivity(intents);
    }
});

goEdit.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents = new Intent( packageContext: DietitianMenu.this, DietitianRecipe.class);
        startActivity(intents);
    }
});

goOut.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        FirebaseAuth.getInstance().signOut();
        startActivity(new Intent( packageContext: DietitianMenu.this, Identity.class));
    }
});

goChat.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intents =(new Intent( packageContext: DietitianMenu.this, DietitianSendMessage.class));
        startActivity(intents);
    }
});
};

```

Figure 4.2.16: Code for Dietitian Menu Page

4.2.10 User Profile Page

When user click on the profile button, they will direct to the user profile page. The user profile page show the personal information of the user and they can upload their profile picture in this page. Besides, they can also edit their information by clicking edit profile button.



Figure 4.2.17: User Profile Page

```

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == 1000){
        if (resultCode == Activity.RESULT_OK){
            Uri imageUri = data.getData();

            //profileImage.setImageURI(imageUri);

            uploadImageToFirebase(imageUri);
        }
    }
}

private void uploadImageToFirebase(Uri imageUri) {
    //upload image to firebase

    StorageReference fileRef= storageReference.child("Users/" +user.getId()+"/profile.jpg");
    fileRef.putFile(imageUri).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

            fileRef.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {
                @Override
                public void onSuccess(Uri uri) { Picasso.get().load(uri).into(profileImage); }
            });
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {

            Toast.makeText( context: ProfileActivity.this, text: "Failed", Toast.LENGTH_SHORT).show();
        }
    });
}

```

Figure 4.2.18: Code of User Profile Page

4.2.11 Dietitian Profile Page

When dietitian click on the profile button, they will direct to the dietitian profile page. The dietitian profile page show the personal information of the dietitian and they can upload their profile picture in this page. Dietitians are required to upload their professional certificate within 7 days for the purpose of identity verification. Besides, they can also edit their information by clicking edit profile button.

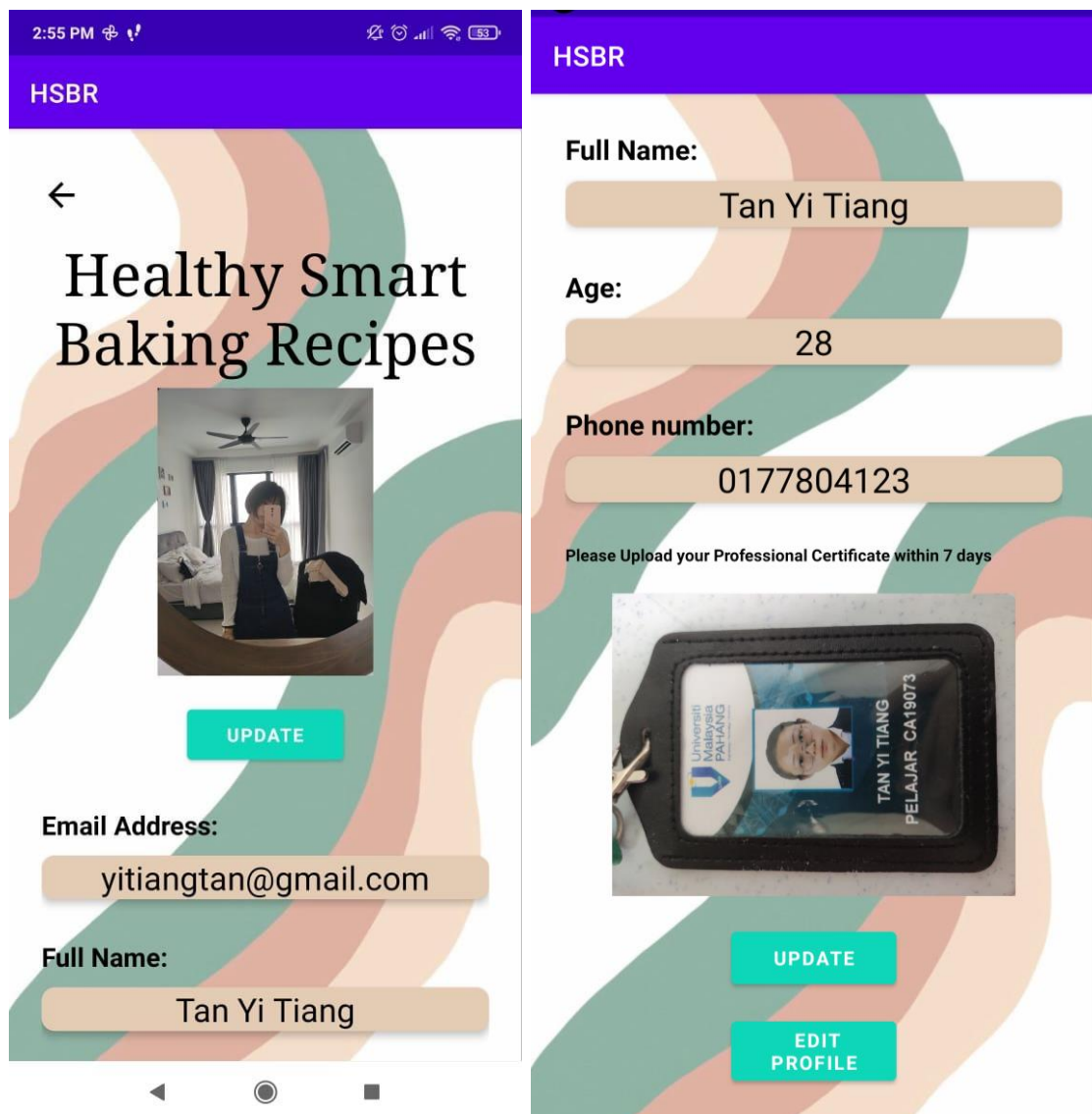


Figure 4.2.19: Dietitian Profile Page

```

private void uploadImageToFirebase(Uri imageUri) {
    //upload image to firebase

    StorageReference fileRef = storageReference.child("Dietitians/" + user.getUid() + "/profile.jpg");
    fileRef.putFile(imageUri).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

            fileRef.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {
                @Override
                public void onSuccess(Uri uri) { Picasso.get().load(uri).into(profileImage); }
            });
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {

            Toast.makeText(context, DietitianProfile.this, text: "Failed", Toast.LENGTH_SHORT).show();
        }
    });
}

private void uploadImageToFirebase2(Uri imageUri2) {
    //upload image to firebase

    StorageReference fileRef2 = storageReference.child("Dietitians/" + user.getUid() + "/professional.jpg");
    fileRef2.putFile(imageUri2).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

            fileRef2.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {
                @Override
                public void onSuccess(Uri uri2) {
                    Picasso.get().load(uri2).into(dietitianProImage);
                }
            });
        }
    });
}

```

Figure 4.2.20: Code for Dietitian Profile Page

4.2.12 Manage User Profile Page

User can update their name, age, and phone number in the update profile page. After clicking update info button, the data will be saved and data display in profile page will be updated.



Figure 4.2.21: Manage User Profile Page

```

editbtn = (Button) findViewById(R.id.btnupdate);
editbtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        String Name = fullName.getText().toString().trim();
        String Age = age.getText().toString().trim();
        String Phone = phone.getText().toString().trim();

        HashMap hashMap = new HashMap();
        hashMap.put("fullName", Name);
        hashMap.put("age", Age);
        hashMap.put("phone", Phone);

        reference.child(Uid).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {
            @Override
            public void onSuccess(Object o) {
                Toast.makeText( context: update_profile.this, text: "Edit Successfully", Toast.LENGTH_SHORT).show();
                startActivity(new Intent( packageContext: update_profile.this, ProfileActivity.class));
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText( context: update_profile.this, text: "Edit Failure", Toast.LENGTH_SHORT).show();
            }
        });
    }
});
}
}

```

Figure 4.2.22: Code for Manage User Profile Page

4.2.13 Manage Dietitian Profile Page

Dietitian can update their name, age, and phone number in the update profile page. After clicking update info button, the data will be saved and data display in profile page will be updated.



Figure 4.2.23: Manage Dietitian Profile Page

```

editbtn = (Button) findViewById(R.id.btnupdate);
editbtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String Name = fullName.getText().toString().trim();
        String Age = age.getText().toString().trim();
        String Phone = phone.getText().toString().trim();

        HashMap hashMap = new HashMap();
        hashMap.put("fullName", Name);
        hashMap.put("age", Age);
        hashMap.put("phone", Phone);

        reference.child(Uid).updateChildren(hashMap).addOnSuccessListener(new OnSuccessListener() {
            @Override
            public void onSuccess(Object o) {
                Toast.makeText(context, update_dietitianProfile.this, "Edit Successfully", Toast.LENGTH_SHORT).show();
                startActivity(new Intent(context, update_dietitianProfile.this, DietitianProfile.class));
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText(context, update_dietitianProfile.this, "Edit Failure", Toast.LENGTH_SHORT).show();
            }
        });
    }
});
}
});

```

Figure 4.2.24: Code for Manage Dietitian Profile Page

4.2.14 View Recipe Page

After clicking on the Recipes button, user/dietitian will be directed to the view recipe page which display recipes with photo and recipes name.

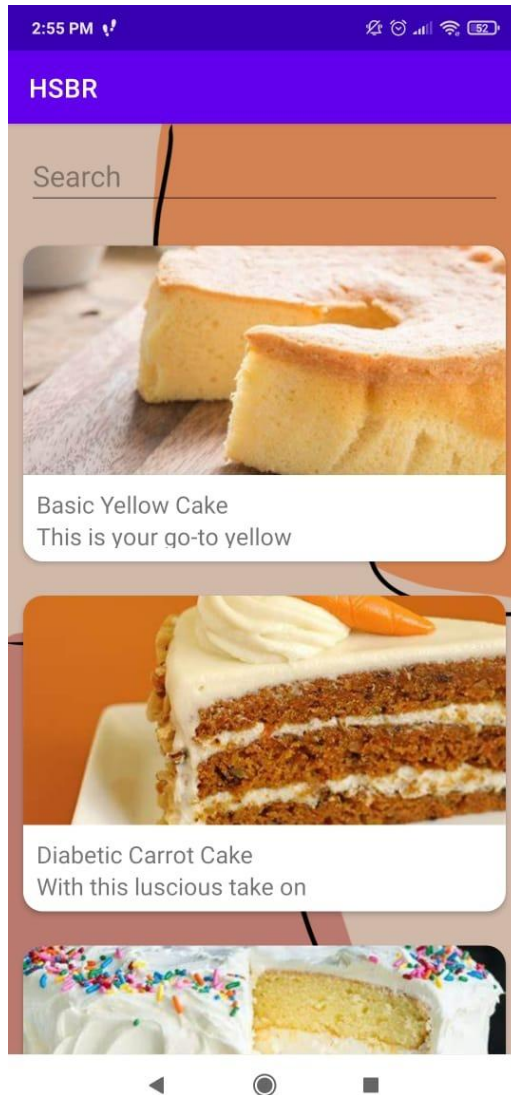


Figure 4.2.25: View Recipe Page

```

databaseReference.child("Recipe").addValueEventListener(new ValueEventListener() {
    @SuppressWarnings("NotifyDataSetChanged")
    @Override
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

        myFoodList2.clear();
        //List<FoodData> myFoodList2 = new ArrayList<>();

        for(DataSnapshot snapshot: dataSnapshot.getChildren()){

            for(DataSnapshot itemSnapshot: snapshot.child("Recipess").getChildren()){

                FoodData foodData = itemSnapshot.getValue(FoodData.class);
                myFoodList2.add(foodData);

            }

        }
        myAdapter2.notifyDataSetChanged();
        progressDialog.dismiss();

    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {

        progressDialog.dismiss();

    }

});

```

Figure 4.2.26: Sample code of View Recipe page

4.2.15 Search Recipe

User/ Dietitian can search recipes they need with the keywords on recipes name.

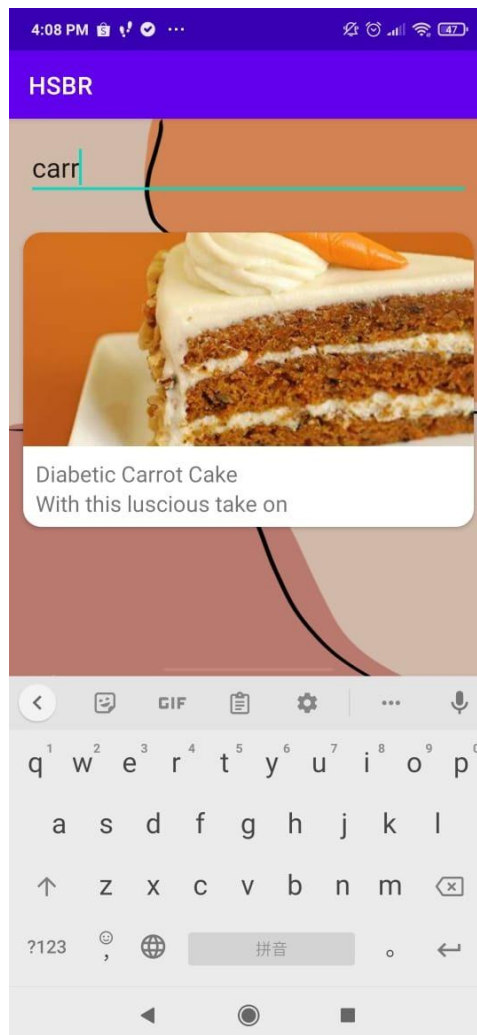


Figure 4.2.27: Search Module

```
txt_Search.addTextChangedListener(new TextWatcher() {  
    @Override  
    public void beforeTextChanged(CharSequence s, int start, int count, int after) {  
    }  
  
    @Override  
    public void onTextChanged(CharSequence s, int start, int before, int count) {  
    }  
  
    @Override  
    public void afterTextChanged(Editable s) {  
        filter(s.toString());  
    }  
});
```

Figure 4.2.28: Code for Search Module

4.2.16 Recipe Details Page

By clicking the picture of the baked goods, the user will be directed to the recipe details page which displays all the information they need in baking. The top of the recipe details page will be the recipe uploader, followed by the recipe's picture, serving size, time taken to prepare, description, steps, ingredients, and nutritional value. If user has any question, they can click find the dietitian button to chat with the dietitian.

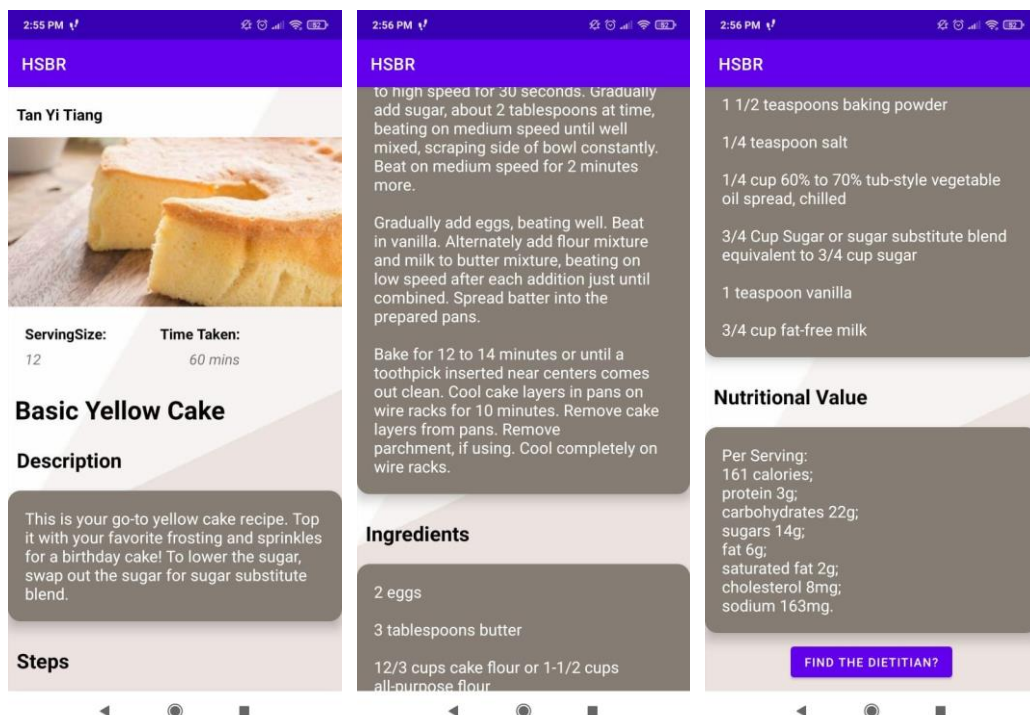


Figure 4.2.29: Recipe Details Page

```

RecipeName = (TextView) findViewById(R.id.txtRecipeName);
foodDescription = (TextView) findViewById(R.id.txtDescription);
foodImage = (ImageView) findViewById(R.id.ivImage2);
steps = (TextView) findViewById(R.id.txtSteps);
serving = (TextView) findViewById(R.id.txtServing);
time = (TextView) findViewById(R.id.txtTime);
nutrition = (TextView) findViewById(R.id.txtNutrition);
ingredients = (TextView) findViewById(R.id.txtIngredients);
creditBy = (TextView) findViewById(R.id.postName);

//listView = findViewById(R.id.listView);
Bundle mBundle = getIntent().getExtras();

if (mBundle != null) {

    RecipeName.setText(mBundle.getString( key: "RecipeName"));
    //foodImage.setImageResource(mBundle.getInt("Image"));
    foodDescription.setText(mBundle.getString( key: "Description"));
    steps.setText(mBundle.getString( key: "RecipeSteps"));
    serving.setText(mBundle.getString( key: "RecipeServingSize"));
    time.setText(mBundle.getString( key: "RecipeTime"));
    nutrition.setText(mBundle.getString( key: "RecipeNutritionalValue"));
    ingredients.setText(mBundle.getString( key: "RecipeIngredients"));
    creditBy.setText(mBundle.getString( key: "CreditBy"));

    key = mBundle.getString( key: "KeyValue");

    imageUrl = mBundle.getString( key: "Image");

    Glide.with( activity: this).load(mBundle.getString( key: "Image")).into(foodImage);
}

```

Figure 4.2.30: Example Code of Recipe Details Page

4.2.17 Calculate BMI Page

User can calculate their BMI when they click on BMI button in the user menu page. They just need to insert weight and height, then click on the calculate button. The result will show at the bottom part of the page.



Figure 4.2.31: Calculate BMI Page

```

public void calcBMI(View view) {

    String w = weight.getText().toString();
    String h = height.getText().toString();
    if(w != null && h != null) {

        float weight_value = Float.parseFloat(w);
        float height_value = Float.parseFloat(h) / 100;

        float bmi = weight_value / (height_value * height_value);
        try {
            if (bmi < 16) {
                BMIresult = "Severly under weight";
            } else if (bmi < 18.5) {
                BMIresult = "Under weight";
            } else if (bmi >= 18.5 && bmi <= 24.9) {
                BMIresult = "Normal Weight";
            } else if (bmi >= 25 && bmi <= 29.9) {
                BMIresult = "Overweight";
            } else {
                BMIresult = "Obese";
            }
            //displaying a result
            calculation = "Result: \n \n" + bmi + "\n" + BMIresult;

            resulttext.setText(calculation);
        } else {BMIresult="n0"; }
    }
}

```

Figure 4.2.32: Code for Calculate BMI Page

4.2.18 Dietitian Edit Recipe Page

The dietitian edit recipe page displays all the recipes uploaded by the dietitian. There is a floating button which allows dietitians to add new recipes. After clicking the floating button, dietitian will be directed to the Dietitian Upload Page.

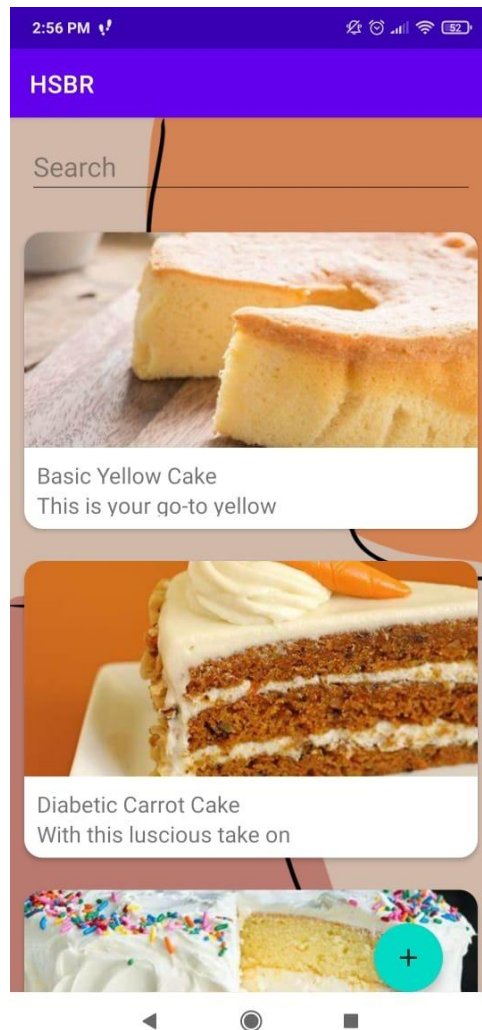


Figure 4.2.33: Dietitian Edit Recipe Page Interface

```
<com.google.android.material.floatingactionbutton.FloatingActionButton  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignParentRight="true"  
    android:layout_alignParentBottom="true"  
    android:layout_marginStart="40dp"  
    android:layout_marginTop="40dp"  
    android:layout_marginEnd="40dp"  
    android:layout_marginRight="40dp"  
    android:layout_marginBottom="40dp"  
    android:onClick="btn_uploadActivity"  
    android:src="@drawable/add" />
```

Figure 4.2.34: Code of Add Floating button

4.2.19 Dietitian Upload Page

When dietitian click on the floating button, they can add recipe to the firebase. Select image button allows them to add image and after they fill in all the information, they can upload it by clicking upload button. The uploaded recipes will be display in the view recipe page for user.

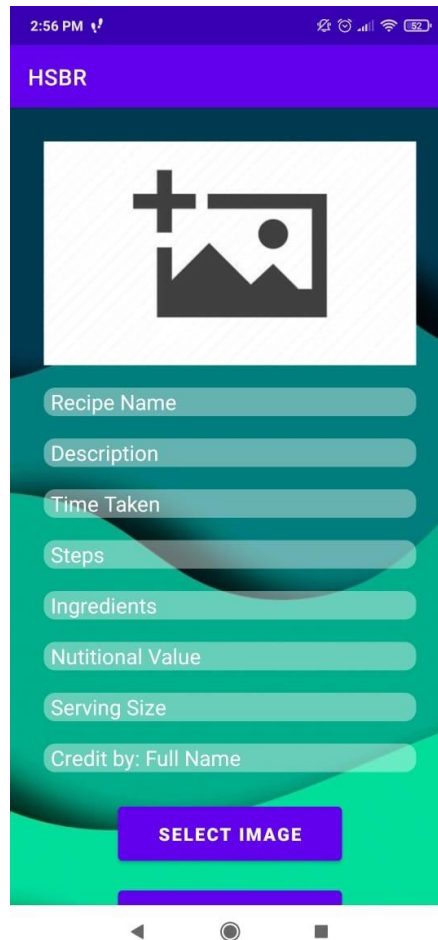


Figure 4.2.35: Dietitian Upload Page Interface

```

private void uploadRecipe() {
    FoodData foodData = new FoodData(
        txt_name.getText().toString(),
        txt_description.getText().toString(),
        txt_time.getText().toString(),
        imageUrl,
        txt_steps.getText().toString(),
        txt_ingredients.getText().toString(),
        txt_nutrition.getText().toString(),
        txt_serving.getText().toString(),
        txt_creditBy.getText().toString()
    );

    user = FirebaseAuth.getInstance().getCurrentUser();
    Uid= user.getId();

    String myCurrentDateTime = DateFormat.getDateTimeInstance().format(Calendar.getInstance().getTime());
    String name= txt_name.getText().toString();

    FirebaseDatabase.getInstance().getReference("Recipe").child(Uid).child("Recipes").child(name).setValue(foodData).addOnCompleteListener(new OnCompleteListener<Void>()
    @Override
    public void onComplete(@NonNull Task<Void> task) {
        if(task.isSuccessful()){
            Toast.makeText(context, Upload_Recipe.this, text: "Recipe Uploaded", Toast.LENGTH_SHORT).show();
            finish();
        }
    }
    }).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        Toast.makeText(context, Upload_Recipe.this, e.getMessage().toString(), Toast.LENGTH_SHORT).show();
    }
    });
}

```

Figure 4.2.36: Sample Code of Dietitian Upload Page

4.2.20 Dietitian Edit Recipe Details Page

In recipe details of dietitian, they can choose to delete the old recipes, they just need to click on the delete button then the recipes will eliminate from the database. They can also update the current information of the recipes by clicking update button. By clicking update button, they will be directed to the Dietitian Update Page.

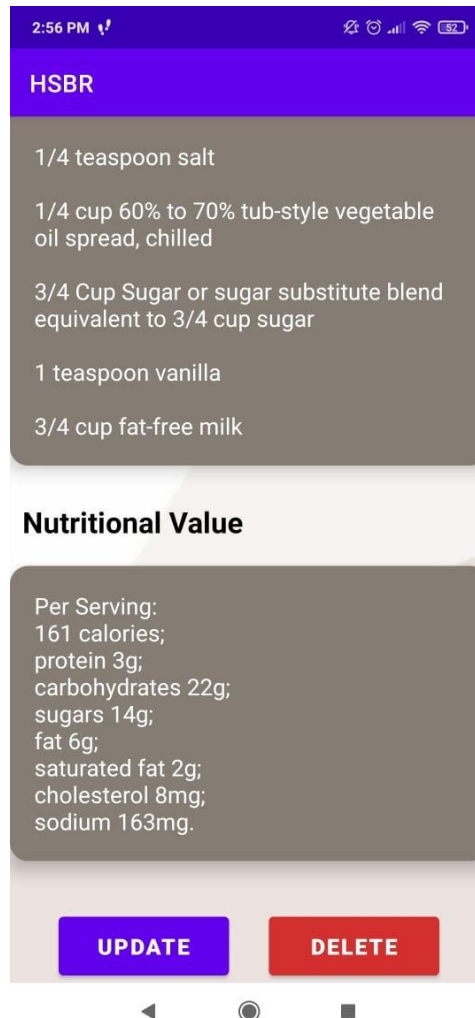


Figure 4.2.37: Dietitian Edit Recipe Details Page

```

public void btnDeleteRecipe(View view) {

    user = FirebaseAuth.getInstance().getCurrentUser();
    Uid= user.getUid();

    final DatabaseReference reference = FirebaseDatabase.getInstance().getReference("Recipe").child(Uid).child("Recipess");
    FirebaseStorage storage = FirebaseStorage.getInstance();

    StorageReference storageReference = storage.getReferenceFromUrl(imageUrl);

    storageReference.delete().addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {

            reference.child(key).removeValue();
            Toast.makeText(context: DetailedActivity.this, text: "Recipe Deleted", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(getApplicationContext(), DietitianRecipe.class));
            finish();

        }
    });
}

public void btnUpdateRecipe(View view) {

    startActivity(new Intent(getApplicationContext(), UpdateRecipeActivity.class)
        .putExtra( name: "recipeNameKey", RecipeName.getText().toString())
        .putExtra( name: "descriptionKey", foodDescription.getText().toString())
        .putExtra( name: "timeKey", time.getText().toString())
        .putExtra( name: "oldimageUrl", imageUrl)
        .putExtra( name: "stepsKey", steps.getText().toString())
        .putExtra( name: "ingredientsKey", ingredients.getText().toString())
        .putExtra( name: "nutritionKey", nutrition.getText().toString())
        .putExtra( name: "servingKey", serving.getText().toString())
        .putExtra( name: "creditByKey", creditBy.getText().toString())
        .putExtra( name: "key", key)
    );
}

```

Figure 4.2.38: Code of Dietitian Edit Recipe Details Page

4.2.21 Dietitian Update Page

When dietitian click on the update button, they can edit recipe details in the firebase. Select image button allows them to add new image and after they done editing the information, they can update it by clicking update button. The updated recipes will be display in the view recipe page for user.

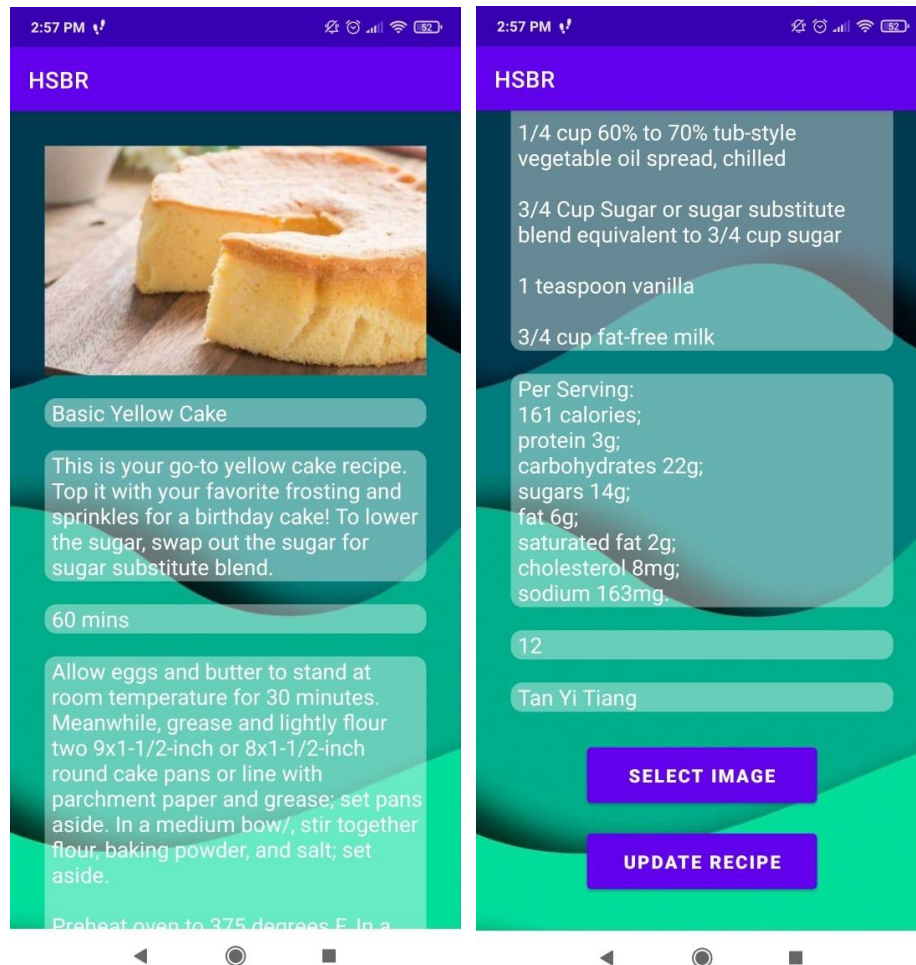


Figure 4.2.39: Dietitian Update Page

```

public void btnUpdateRecipe(View view) {

    recipeName = txt_name.getText().toString().trim();
    recipeDescription = txt_description.getText().toString().trim();
    recipeTime = txt_time.getText().toString().trim();
    recipeSteps = txt_steps.getText().toString().trim();
    recipeIngredients = txt_ingredients.getText().toString().trim();
    recipeNutrition = txt_nutrition.getText().toString().trim();
    recipeServing = txt_serving.getText().toString().trim();
    creditBy = txt_creditBy.getText().toString().trim();

    final ProgressDialog progressDialog = new ProgressDialog( context: this);
    progressDialog.setMessage("Recipe Uploading...");
    progressDialog.show();
    storageReference = FirebaseStorage.getInstance()
        .getReference().child("RecipeImage").child(uri.getLastPathSegment());
    storageReference.putFile(uri).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

            Task<Uri> uriTask = taskSnapshot.getStorage().getDownloadUrl();
            while(!uriTask.isComplete() );
            Uri urlImage = uriTask.getResult();
            imageUrl = urlImage.toString();
            uploadRecipe();
            progressDialog.dismiss();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) { progressDialog.dismiss(); }
    });
}

```

Figure 4.2.40: Code of Dietitian Update Page

4.2.22 Chat Name List Page

When user/dietitian clicks on the chat button in menu page, they will be directed to the chat name list which display user/dietitian name. They can choose their name and chat with each other.

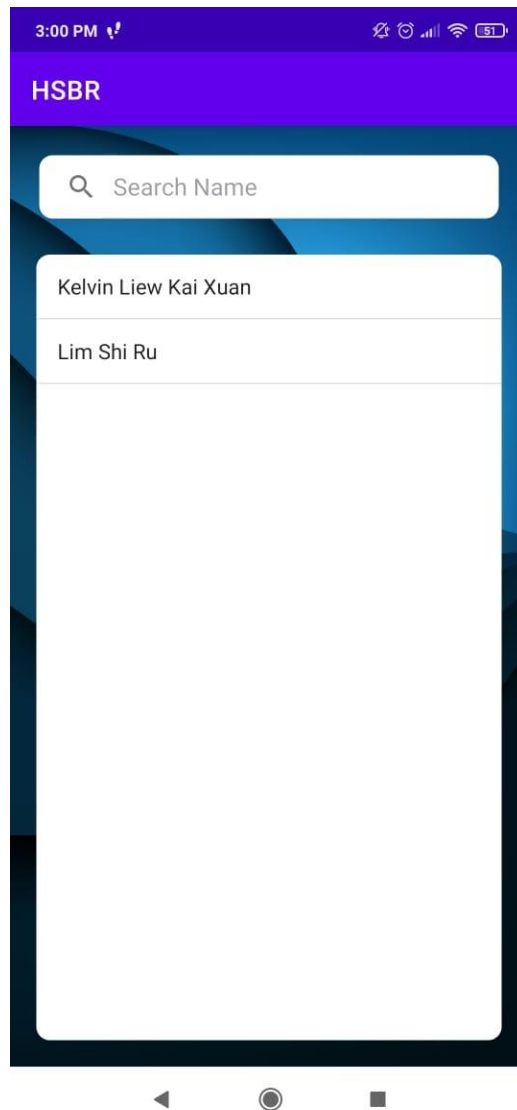


Figure 4.2.41: Chat Name List Page

```

databaseReference.child("Dietitians").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        if (snapshot.exists()){
            for (DataSnapshot dataSnapshot : snapshot.getChildren()){
                if (!dataSnapshot.child("fullName").getValue().toString().equals(firebaseAuth.getCurrentUser().getDisplayName())){
                    User Dietitians = dataSnapshot.getValue(User.class);
                    User.add(Dietitians);

                    username.add(dataSnapshot.child("fullName").getValue().toString());
                }
            }
            arrayAdapter = new ArrayAdapter( context: UserSendMessage.this, android.R.layout.simple_list_item_1, username);
            userListView.setAdapter(arrayAdapter);
        }
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText( context: UserSendMessage.this, text: "Failed to load users", Toast.LENGTH_SHORT).show();
    }
});

```

Figure 4.2.42: Code for Chat Name List Page

4.2.23 Chat Page

User can send message to the dietitian through the chat page. They just need to type the question or message in the chat box and sent. Dietitian can also reply to the chat.



Figure 4.2.43: Chat Page

```

@NonNull
@Override
public RecyclerView.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
    if (viewType == ITEM_SEND){
        View v = LayoutInflater.from(context).inflate(R.layout.schatlayout,parent, attachToRoot: false);
        return new SenderViewHolder(v);
    }else {
        View v = LayoutInflater.from(context).inflate(R.layout.rchatlayout,parent, attachToRoot: false);
        return new ReceiverViewHolder(v);
    }
}

@Override
public void onBindViewHolder(@NonNull RecyclerView.ViewHolder holder, int position) {
    Messages messages = messagesArrayList.get(position);

    if (getItemViewType(position) == ITEM_SEND){
        SenderViewHolder viewHolder = (SenderViewHolder) holder;
        viewHolder.textViewmessage.setText(messages.getMessage());
        viewHolder.timeofmessage.setText(messages.getCurrentntime());
    } else {
        ReceiverViewHolder viewHolder = (ReceiverViewHolder) holder;
        viewHolder.textViewmessage.setText(messages.getMessage());
        viewHolder.timeofmessage.setText(messages.getCurrentntime());
    }
}

@Override
public int getItemViewType(int position) {
    Messages messages = messagesArrayList.get(position);

    if (sendername.equals(messages.getSenderName())){
        return ITEM_SEND;
    }else {
        return ITEM_RECEIVE;
    }
}

```

Figure 4.2.44: Code for Chat Page

4.3 Testing

This application was tested according to system and user standards to determine whether it was acceptable for a general audience. A mobile app development process cannot be completed without testing. By completing this phase, the developer will be able to enhance and improve the application they have created. As a result of feedback and survey results, the developer may be able to troubleshoot and improve the application.

4.3.1 User Acceptance Test

User Acceptance Test (UAT) form

No	Module	Activities	Status			Comments
1.	Registration	User Sign Up	Yes	/	No	User/Dietitian can sign up an account in the application.
2.	Login	User Login	Yes	/	No	User/ Dietitian with account can login to their account in the application.
3.		User Forgot Password	Yes	/	No	User/ Dietitian able to reset password.

4.		User Logout	Yes	/	No	User/ Dietitian able to logout from their account.
5.	Manage User and Dietitian Profile	Change Profile Picture	Yes	/	No	User/ Dietitian able to upload profile picture.
6.		Update Name	Yes	/	No	User/ Dietitian able to update name.
7.		Update Age	Yes	/	No	User/ Dietitian able to update age.
8.		Update Phone Number	Yes	/	No	User/ Dietitian able to update phone number.
9.		(Dietitian) Upload Professional Certificate	Yes	/	No	Dietitian able to upload professional certificate.
10.	Search Recipe	Search Bar	Yes	/	No	User/ Dietitian able to search recipe using recipe name.

11.	Select and View Recipes	View Recipe Details	Yes	/	No	User/Dietitian able to select and view recipe details.
12.		Chat a Dietitian	Yes	/	No	User able to send chat.
13.	Add, Edit, Delete Recipes	Add Recipes	Yes	/	No	Dietitian able to add recipe and inserting recipe details.
14.		Edit Recipes	Yes	/	No	Dietitian able to edit recipe and inserting recipe details.
15.		Delete Recipes	Yes	/	No	Dietitian able to delete recipe.
16.	Chat	Select name to Chat	Yes	/	No	User/Dietitian able to select name.
17.		Sent Messages	Yes	/	No	User/Dietitian able to chat.

Table 4.3.1 User Acceptance Test

This test has been performed by:

Name : Tan Yi Tiang

Signature : 

Date : 16/01/2023

4.3.2 Result and Discussion

An acceptance test is conducted to determine whether the Healthy Smart Baking recipe mobile application is able to meet the user's needs. This survey will evaluate all features implemented, and users will provide feedback through surveys. Testing the usability of the application is conducted with the help of a questionnaire. The results and analysis of the study are shown below.

Do you think Healthy Smart Baking Recipe Mobile Application interface is easy to understand?
12 responses

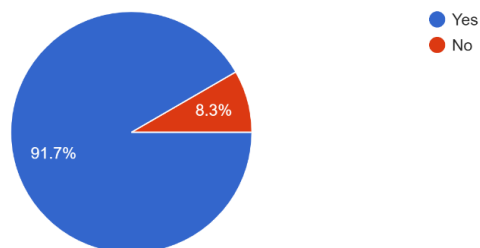


Figure 4.3.1: User Acceptance Test Question 1

Based on the survey, there are 11 respondents (91.7%) think that Healthy Smart Baking Recipe Mobile Application interface is easy to understand will 1 respondent (8.3%) disagree with the statement.

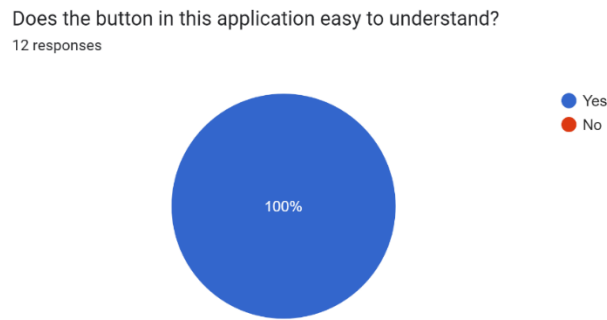


Figure 4.3.2: User Acceptance Test Question 2

Based on the survey, all the respondents (100%) think that button in the Healthy Smart Baking Recipe Mobile Application is easy to understand.

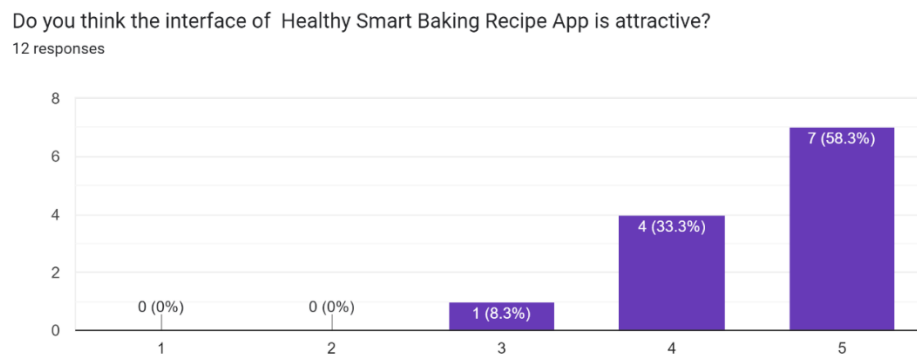


Figure 4.3.3: User Acceptance Test Question 3

Based on the survey, there are 7 (58.3%) respondents give a rate of 5 for the interface, 4 (33.3%) respondents give a rate of 4 and 1 (8.3%) respondent give a rate of 3 for the interfaces designed.

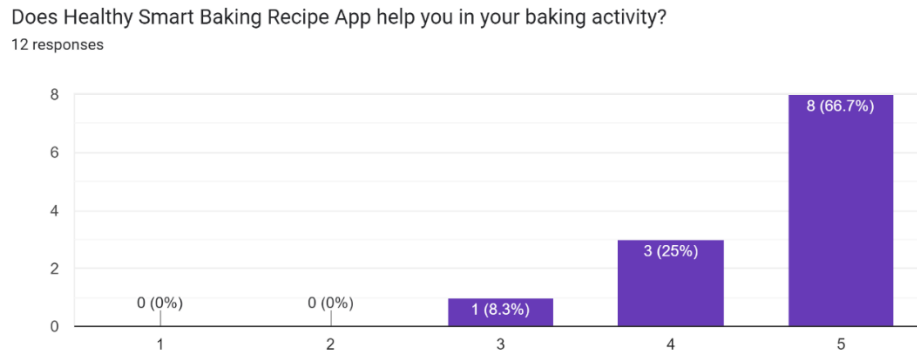


Figure 4.3.4: User Acceptance Test Question 4

Based on the survey, there are 8 (66.7%) respondents give a rate of 5 for the claim Healthy Smart Baking Recipe App help in their baking activity, 3 (25%) respondents give a rate of 4 and 1 (8.3%) respondent give a rate of 3 for the claim.

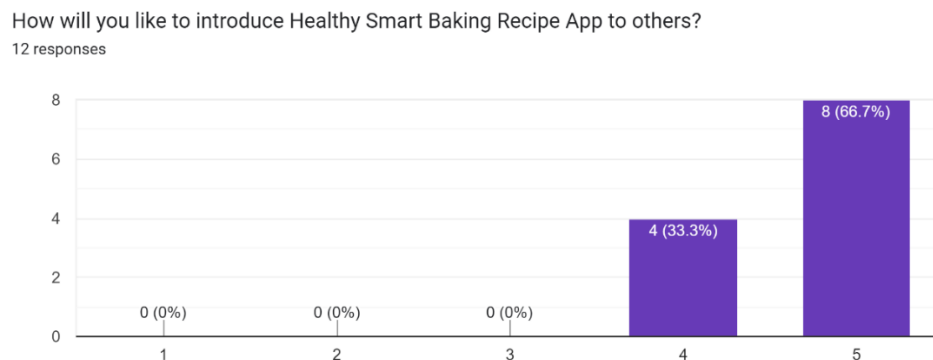


Figure 4.3.5: User Acceptance Test Question 5

Based on the survey, there are 8 (66.7%) respondents give a rate of 5 and 4 (33.3%) respondents rate 4 that they would like to introduce Healthy Smart Baking Recipes App to others.

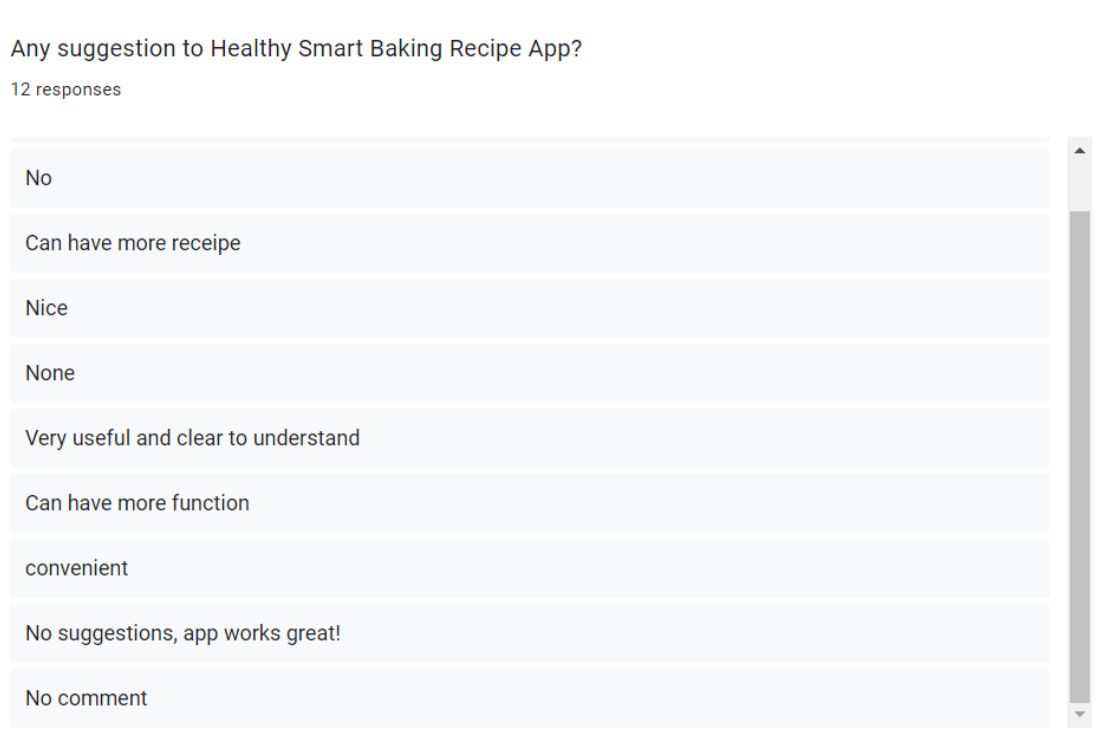


Figure4.3.6: User Acceptance Test Question 6

Based on the survey, there are few suggestions from the respondents. There are few suggestions for the Healthy Smart Baking Recipes such as the application can have more function and more recipes. Few respondents comment that the application is clear to understand and works great. Overall, the application is usable and acceptable for the respondents.

4.4 Chapter Summary

In conclusion, Healthy Smart Baking recipe mobile application is successfully developed in this chapter. In the implementation and development phases, screenshots of coding and an interface for a specific function are provided to illustrate the details. The Healthy Smart Baking recipe mobile application is developed using Android Studio and the language used is java programming language. The results of the testing are presented in this chapter, along with a discussion of the results. As a result of the testing response, the results are displayed in graph form to illustrate the usability and acceptability of the application to the user.

CHAPTER 5

CONCLUSION

5.1 Objective Revisited

In the first part of the thesis, the project's objective is to digitalize baking recipes and provide nutrition information for baked goods through a mobile application. The Healthy Smart Baking Recipe Mobile Application has completed all its objectives throughout the entire project development process from planning to design to construction and finally cut-over. In this application, users can log in to their own accounts and get the recipes they need directly from the application, receive step-by-step instructions and the estimated preparation time, view nutritional information and calories per serving based on the recipes they choose, and consult with dietitians about the recipes.

5.2 Limitation

Limitations refer to weaknesses that can't be addressed by the current version of the system. It is important to note that this project has a few limitations:

1. The system developed is currently only compatible with Android phones; IOS users will not be able to use the system with their phone.
2. English is the only language supported. All information will be displayed in English only. There are no other languages to choose from. Thus, non-native English speakers might have trouble using the application.

3. There is no classification function in the system for categorizing recipes by ingredients. Without the categorizing function, users might need to search by recipe name, but not by ingredient, to find the recipes they need.

5.3 Future Work

Due to the project's limitations, there is some scope for improvement in the future. With the number of IOS users growing yearly, it would be helpful if this system could be used by all IOS users. This would enable them to view a suitable baking recipe and get the information they need for their baking. Next is the language. In order to enable people who speak different languages to use multilingual systems effectively, we need to develop languages such as Malay, Chinese, and others. In addition to a classification function, the system should have the ability to sort recipes by ingredients. Users will be able to find recipes quickly and easily by categorizing them by ingredients. With the improvement, the application can be used by the public, dietitian and nutritionist can make used this application to provide baking recipes for their patients and people with diabetes and obesity can used this application to get their sweet treats.

REFERENCES

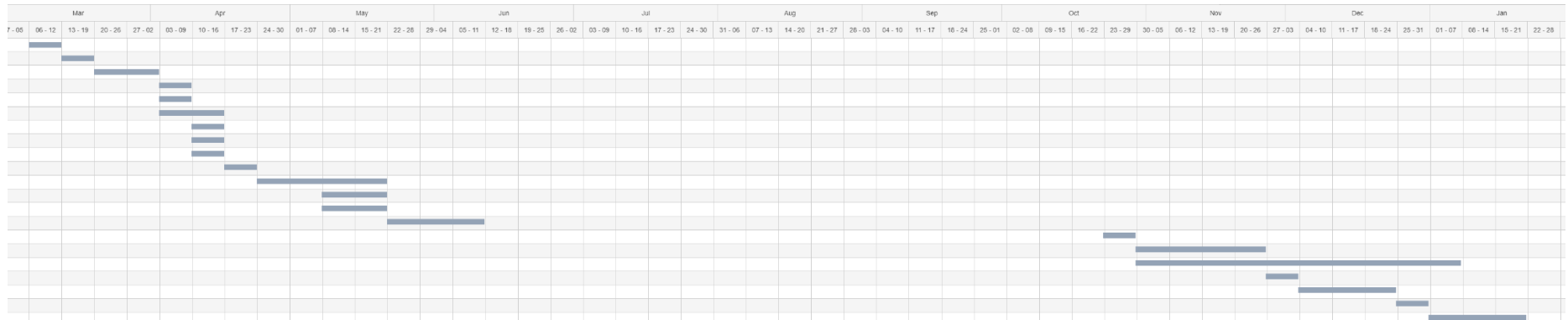
1. *Obesity and overweight*. (2021, June 9). <https://www.who.int/news-room/factsheets/detail/obesity-and-overweight>
2. *5 Reasons Baking is Good for Mental Health - Goodnet*. (2013, October 8). <https://www.goodnet.org/articles/5-reasons-baking-good-for-mental-health>
3. Harnil Oza. (2017, March 28). *The Importance Of Mobile Applications In Everyday Life!* <https://www.hyperlinkinfosystem.com/blog/the-importance-of-mobile-applications-in-everyday-life>
4. L.Ceci. (2022, February 25). • *Time spent on average on a smartphone in the U.S. 2021* | Statista. <https://www.statista.com/statistics/1224510/time-spent-per-day-on-smartphone-us/>
5. R.Paul Singh & Samuel A. Matz. (2021, December 23). *baking | Description, History, Types, & Facts | Britannica*. <https://www.britannica.com/topic/baking>
6. *The importance of Literature Review in Research Writing | Elsevier Blog*. (n.d.). Retrieved April 10, 2022, from <https://scientific-publishing.webshop.elsevier.com/research-process/importance-literature-review-research-writing/>
7. Priya Pedamkar. (n.d.). *What is Methodology? | 5 Types of Software Development Methodology*. Retrieved April 14, 2022, from <https://www.educba.com/what-is-methodology/>
8. *Agile Model (Software Engineering) - javatpoint*. (n.d.). Retrieved January 9, 2023, from <https://www.javatpoint.com/software-engineering-agile-model>
9. *The Phases Of Agile Software Development Life Cycle & Workflow And Project Management*. (2022, April 26). <https://bitbytesoft.com/phases-of-agile-software-development-life-cycle/>

10. Thomas Hamilton. (2022, December 10). What is Agile Testing? Process & Life Cycle. <https://www.guru99.com/agile-testing-a-beginner-s-guide.html>

APPENDIX A
SAMPLE APPENDIX 1

Gantt Chart

ID	Title	Start Time	End Time
1	Meeting with Supervisor	03/06/2022	03/13/2022
2	Brainstorming idea for proje	03/13/2022	03/20/2022
3	Writing report Chapter 1	03/20/2022	04/03/2022
4	Do correction on report Cha	04/03/2022	04/10/2022
5	Search Existing System for C	04/03/2022	04/10/2022
6	Writing Report - Chapter 2	04/03/2022	04/17/2022
7	Do correction for Chapter 2	04/10/2022	04/17/2022
8	Meeting with supervisor	04/10/2022	04/17/2022
9	Writing report - Draft Chapte	04/10/2022	04/17/2022
10	First Evaluation	04/17/2022	04/24/2022
11	Design Flowchart and Diagra	04/24/2022	05/22/2022
12	Design User Interface	05/08/2022	05/22/2022
13	Do Project Prototype	05/08/2022	05/22/2022
14	Finalize report and submissi	05/22/2022	06/12/2022
15	Brief PSM 2	10/23/2022	10/30/2022
16	Draft Chapter 4 Report	10/30/2022	11/27/2022
17	Develop app	10/30/2022	01/08/2023
18	Meeting with supervisor and	11/27/2022	12/04/2022
19	Chapter 4 & Chapter 5 Repo	12/04/2022	12/25/2022
20	Meet supervisor and submit	12/25/2022	01/01/2023
21	Complete app and report for	01/01/2023	01/22/2023



APPENDIX B

USER ACCEPTANCE TEST

User Acceptance Test of Healthy Smart Baking Recipe App

Hi, my name is Tan Yi Tiang (Faculty of Computing), Bachelor of Computer Science (Computer System and Networking) from Universiti Malaysia Pahang (UMP) who currently developing my FYP Project. I have conducted a survey on the user acceptance survey of Healthy Smart Baking Recipes Mobile Application. The purpose of this survey is to collect responses of application usability. All of the information collected would be confidential and anonymous, only for academic purpose. Thank you for spare your time to fill up the survey. Stay safe and healthy.

Link of the APK file:
https://drive.google.com/file/d/1nXZohpelcyiG2hmottxO78Ht35Jf08EI/view?usp=share_link

Thank you in advance for your valuable time and cooperation.

 yitiangtan@gmail.com (not shared) [Switch accounts](#) 

*Required

Do you think Healthy Smart Baking Recipe Mobile Application interface is easy to understand? *



- Yes
- No

Does the button in this application easy to understand? *

- Yes
- No

Do you think the interface of Healthy Smart Baking Recipe App is attractive? *

- | | | | | | | |
|----------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------|
| | 1 | 2 | 3 | 4 | 5 | |
| Very bad | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | Very Good |

Does Healthy Smart Baking Recipe App help you in your baking activity? *

- | | | | | | | |
|------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-------------|
| | 1 | 2 | 3 | 4 | 5 | |
| Not useful | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | Very Useful |

How will you like to introduce Healthy Smart Baking Recipe App to others? *

- | | | | | | | |
|-------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|----------------|
| | 1 | 2 | 3 | 4 | 5 | |
| Strongly Disagree | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | Strongly Agree |

Any suggestion to Healthy Smart Baking Recipe App? *

Your answer _____

Submit

Clear form