

MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING

BY

CHOONG WIN SHEN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONOURS)

INFORMATION SYSTEMS ENGINEERING

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2022

REPORT STATUS DECLARATION FORM

Title: MOBILE APPLICATION FOR MEET UP HOSTING AND
GATHERING

Academic Session: Jan 2022

I CHOONG WIN SHEN
(CAPITAL LETTER)

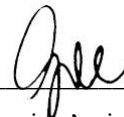
declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,



(Author's signature)



(Supervisor's signature)

Address:

129A, Persiaran Desa Rishah 1,

Desa Rishah,

30100 Ipoh, Perak.

Ms. Yap Seok Gee

Supervisor's name

Date: 17/4/2022

Date: 17/4/2022

Universiti Tunku Abdul Rahman			
Form Title : Sample of Submission Sheet for FYP/Dissertation/Thesis			
Form Number: FM-IAD-004	Rev No.: 0	Effective Date: 21 JUNE 2011	Page No.: 1 of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TUNKU ABDUL RAHMAN

Date: 17/4/2022

SUBMISSION OF FINAL YEAR PROJECT

It is hereby certified that CHOONG WIN SHEN (ID No: 18ACB01717) has completed this final year project/ dissertation/ thesis* entitled “MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING” under the supervision of MS. YAP SEOK GEE (Supervisor) from the Department of Information Systems, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

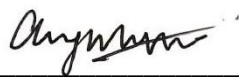
Yours truly,



(CHOONG WIN SHEN)

DECLARATION OF ORIGINALITY

I declare that this report entitled “**MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : CHOONG WIN SHEN

Date : 17/4/2022

ACKNOWLEDGEMENTS

I am extremely grateful to my supervisor Ms. Yap Seok Gee for guiding me throughout the entire journey of this project. Her patience, enthusiastic encouragement, useful recommendations, and advice have greatly helped me to complete the project successfully. Not to mention that she is always there to enlighten me with a fresh perspectives whenever I have confusions or obstacles. I am very grateful to have her as my supervisor in my university life.

I am also thankful to the project moderator Mr. Lim Jit Theam for taking efforts to evaluate the quality of this project. His valuable feedbacks are very useful to me in making better improvement for this project.

I would like to express my deepest thanks to my family and friends for constantly supporting me in every stages of my life. Their encouragement and caring are the biggest help to me in achieving my milestones. I am able to reach at this moment of life is all because of my family that supports me to pursue my dream without hesitations. There are no words to express my love and appreciation to them.

ABSTRACT

This project is a mobile application development project that facilitate users in meeting up with stranger. Some of the problems with traditional dating application can be resolved by this proposed project, where user could find it easy to use the application and meet up with others in real person easily. There will be some features assisting user in scheduling and initiating a meet up such as host, join or accept application of a meet up. This will change the way of people engage with each other differently, unlike any other traditional dating applications. Business owner can now take a part in it for their business promotion with booking and profiling function. This project is developed with Extreme Programming methodology which will be easier for an individual or small team. The technology for backend will be using Android studio in Java language, while Firebase will be the main cloud database for any data of users in the application. This project is initialized in Jan 2021 for the first part and it is proceeded in Jan 2022 for the second part.

TABLE OF CONTENTS

TITLE PAGE	i
REPORT STATUS DECLARATION FORM	ii
FYP THESIS SUBMISSION FORM	iii
DECLARATION OF ORIGINALITY	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES	ix
LIST OF TABLES	xii
LIST OF ABBREVIATIONS	xiv
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement and Motivation	2
1.2 Objectives	3
1.3 Project Scope and Direction	4
1.4 Contributions	4
1.5 Report Organization	5
CHAPTER 2 LITERATURE REVIEW	7
2.1 Existing Applications Review	7
2.1.1 Tinder	7
2.1.2 Bumble	8
2.1.3 Tantan	9
2.2 Methodologies Review	10
2.1.1 Rapid Application Development (RAD)	10
2.1.2 Agile Methodology – Extreme Programming (XP)	11
2.1.3 Spiral Methodology	12
CHAPTER 3 SYSTEM METHODOLOGY/APPROACH	14
3.1 System Design Diagram/Equation	14
3.1.1 Use Case Diagram	14
3.1.2 Activity Diagram	30
	vii

3.1.3	User Interface (UI) Design	36
CHAPTER 4	SYSTEM DESIGN	43
4.1	System Block Diagram	43
4.2	System Components Specifications	44
4.3	Components Design	53
4.4	System Components Interaction Operations	55
CHAPTER 5	SYSTEM IMPLEMENTATION	66
5.1	Hardware Setup	66
5.2	Software Setup	66
5.3	Setting and Configuration	67
5.4	System Operation	67
5.5	Concluding Remark	72
CHAPTER 6	SYSTEM EVALUATION AND DISCUSSION	73
6.1	System Testing and Performance Metrics	73
6.2	Testing Setup and Result	73
6.3	Implementation Issues and Challenges	78
6.4	Objectives Evaluation	78
6.5	Concluding Remark	79
CHAPTER 7	CONCLUSION AND RECOMMENDATION	80
7.1	Conclusion	80
7.2	Recommendation	80
REFERENCES		82
WEEKLY LOG		85
POSTER		96
PLAGIARISM CHECK RESULT		97
FYP2 CHECKLIST		100

LIST OF FIGURES

Figure Number	Title	Page
Figure 2.1	Swipe left or right feature of Tinder	7
Figure 2.2	Woman on Bumble has to initiate conversation first	8
Figure 2.3	Swipe left or right feature on Tantan	9
Figure 3.1	Normal User Use Case Diagram	14
Figure 3.2	Business User Use Case Diagram	23
Figure 3.3	Log in	30
Figure 3.4	Sign Up	30
Figure 3.5	Log out	30
Figure 3.6	Host Meet up	30
Figure 3.7	Join Meet up	30
Figure 3.8	View User	31
Figure 3.9	View Business	31
Figure 3.10	Write Review	31
Figure 3.11	Give Rating	31
Figure 3.12	View Profile	31
Figure 3.13	Edit Profile	31
Figure 3.14	View Review	32
Figure 3.15	View Meet up	32
Figure 3.16	Check Booking	32
Figure 3.17	Check Apply	32
Figure 3.18	Accept Application	32
Figure 3.19	Reject Application	32
Figure 3.20	Close Registration	33
Figure 3.21	Send Booking	33
Figure 3.22	View Chat	33
Figure 3.23	Send Chat	33
Figure 3.24	Open Chat Room	33
Figure 3.25	Add Friend	33
Figure 3.26	Respond Friend Request	34
Figure 3.27	View Notification	34

Figure 3.28	View Business Profile	34
Figure 3.29	Edit Business Profile	34
Figure 3.30	View Approved Booking	34
Figure 3.31	View Pending Booking	34
Figure 3.32	Accept Booking	35
Figure 3.33	Reject Booking	35
Figure 3.34	View Business Review	35
Figure 3.35	View Business Notification	35
Figure 3.36	View Business Chat	35
Figure 3.37	Send Business Chat	35
Figure 3.38	Open Business Chat Room	36
Figure 3.39	View Normal User Profile	36
Figure 3.40	Turn On Booking	36
Figure 3.41	Turn Off Booking	36
Figure 3.42	Sign Up and Login	36
Figure 3.43	Sign up options for different account	36
Figure 3.44	Sign up for normal user	37
Figure 3.45	Sign up for business user	37
Figure 3.46	Home page for normal user	37
Figure 3.47	Notification Page of normal user	38
Figure 3.48	Chat page for normal user	38
Figure 3.49	Host meet up	38
Figure 3.50	Profile Page of normal user (user's own profile)	38
Figure 3.51	Review Page of normal user (user's own review)	39
Figure 3.52	Review page of business user	39
Figure 3.53	Meet up Page of normal user (user's own meet up)	39
Figure 3.54	Booking detail	39
Figure 3.55	Meet up requested	39
Figure 3.56	Meet up approved or ended	40
Figure 3.57	Profile Page of normal user (other users' profile)	40
Figure 3.58	Review Page of normal user (other users' review)	41
Figure 3.59	Meet up Page of normal user (other users' meet up)	41
Figure 3.60	Business profile in normal user's view	41

Figure 3.61	Home page of business user	41
Figure 3.62	Booking detail page of business user	41
Figure 3.63	Notification page of business user	41
Figure 3.64	Chat page of business user	42
Figure 3.65	Profile page of business user	42
Figure 4.1	System Block Diagram	43
Figure 4.2	Firebase Cloud Storage Structure	51
Figure 4.3	The usage of different layouts	52
Figure 4.4	BottomNavigationView in home screen	52
Figure 4.5	ToolBar on top of screen	53
Figure 4.6	Normal User Flow Chart Diagram	53
Figure 4.7	Business User Flow Chart Diagram	54
Figure 4.8	Navigation Graph	55
Figure 5.1	Setting and Configuration	67
Figure 5.2	User Manual: Home page	67
Figure 5.3	User Manual: Host meet up	68
Figure 5.4	User Manual: Host meet up	68
Figure 5.5	User Manual: Host meet up	69
Figure 5.6	User Manual: Host meet up	69
Figure 5.7	User Manual: Edit profile	70
Figure 5.8	User Manual: Join meet up	70
Figure 5.9	User Manual: Respond friend request	71
Figure 5.10	User Manual: Business home page	71
Figure 5.11	User Manual: Booking details	72

LIST OF TABLES

Table Number	Title	Page
Table 2.1	Summary of application review	10
Table 3.1	Host meet up	14
Table 3.2	Join meet up	15
Table 3.3	View user	15
Table 3.4	View business	16
Table 3.5	Give rating	16
Table 3.6	Write review	17
Table 3.7	View profile	17
Table 3.8	Edit profile	17
Table 3.9	View review	18
Table 3.10	Check booking	18
Table 3.11	Check apply	19
Table 3.12	Send booking	19
Table 3.13	Accept application	20
Table 3.14	Reject application	20
Table 3.15	Close registration	20
Table 3.16	View chat	21
Table 3.17	Send chat	21
Table 3.18	Open chat room	21
Table 3.19	Add friend	22
Table 3.20	Respond friend request	22
Table 3.21	View notification	23
Table 3.22	View business profile	24
Table 3.23	Edit business profile	24
Table 3.24	View history booking	24
Table 3.25	View pending booking	25
Table 3.26	Accept booking	25
Table 3.27	Reject booking	25
Table 3.28	View business review	26
Table 3.29	View business notification	26

Table 3.30	View business chat	26
Table 3.31	Send business chat	27
Table 3.32	Open business chat room	27
Table 3.33	View normal user profile	27
Table 3.34	Turn on booking	28
Table 3.35	Turn off booking	28
Table 3.36	Log in	28
Table 3.37	Log out	29
Table 3.38	Sign up	29
Table 4.1	Log in and sign up	55
Table 4.2	Host a meet up	57
Table 4.3	Join a meet up	58
Table 4.4	Meet up is accepted, ended or closed	58
Table 4.5	Host manages the meet up and visits business profile	58
Table 4.6	User views chat page	59
Table 4.7	User view notification	60
Table 4.8	Normal user view own user profile	61
Table 4.9	Normal user view other user profile	62
Table 4.10	Business user visits home page	63
Table 4.11	Business user visits notification page	63
Table 4.12	Business user visits chat page	63
Table 4.13	Business user visits own user profile	63
Table 4.14	Business user visits normal user profile	64
Table 4.15	Log out	65
Table 5.1	Requirements for user's device	66
Table 5.2	Requirements for developer's device	66
Table 6.1	Functional testing	66
Table 6.2	Objectives Evaluation	73

LIST OF ABBREVIATIONS

RAD	Rapid Application Development
JAD	Joint Application Development
CASE	Computer-Aided Software Engineering
XP	Extreme Programming

CHAPTER 1 INTRODUCTION

Online dating is gradually becoming common and consists of nearly 980,000 users in 2020 including pay and non-pay users in every range of age, especially the group of 18 to 34 years old [1]. There are various dating applications in the market such as Tinder and Tantan which providing a platform for users to have casual communication through texting, video or photo sharing with a stranger met online. It is widely used; however, people tend to dislike using it. Based on the statistics, around 59% of women and 55% of men have negative opinions towards the online dating services because of the risk of engaging strangers online and worried of leaking personal information [2]. It is normal to have this worried as we cannot judge one through an online communication since people can easily construct their personal information and personal qualities as we cannot observe them, unless meeting up in real person. However, there are other more critical problems such as these dating applications are having a consistent, old and traditional concept where user can only swipe left to pass and right to like on someone's pictures. In order to be matched, they have to wait for the other user to swipe right on them too. Furthermore, most of the users are only communicate with strangers on the application, but not many of them successfully meet each other in-person due to few reasons. For example, the users will hesitate to meet up even though they have been communicating through texting for a long period as they do not wish to ruin the texting chemistry [3].

Hence, this project is to develop a mobile application that connecting strangers through face-to-face communication. When user first time get on the application, there will be a list of meet up created by other users that allows user to join. Then, if they are accepted, they will be meeting each other in a particular venue. It gives conveniency for users to schedule a meet up with others. They can meet up with multiple person that sharing similar interest at the same time. They can also review other users' rating and feedback to avoid meeting up some users with inappropriate behaviour. Not only normal user, but business users can also promote their businesses on the application to attract more people to schedule an event at their business location.

1.1 Problem Statements and Motivations

a. Cognitive overload due to swipe left or right feature

In Tinder, Bumble or Tantan, there is millions of profile for users to swipe left or right for the one they dislike or like. However, there is no history record on which profile has the user swiped on before. This means that user is forced to recall on which profiles they have swiped. This could be cumbersome as well since there is a case where user accidentally swipe left on someone. For instance, those profile being swiped left will nearly never show up again as Tinder will deprioritize them, except for some special circumstances such as the user deletes the account and comeback [4]. Besides, user might experience memory load on the user they have swipe right before as it will not instantly shows up after user swiped, unless they are matched. Overall, user might have trouble in recalling what have they done on the application. Since there is a lot of alternatives and options of user on Tinder, users will keep swiping and swiping lots of profile until they cannot even recall and even brain overloading [5]. Hence, this is a motivation to build an application that reduce the cognitive and memory load for a better user experience.

b. It is hard to schedule for meet-up due to no supported function built in

As most of the online dating applications are targeting for user to match with the one they like, they are more focusing on building the algorithm for user matching such as Elo Score algorithm for Tinder and Gale-Shapley algorithm for Hinge [5]. Hence, there is no function support the entire process of how users meeting up once they are matched to have further engagement. Whenever users feel the need of meeting up with one and another, they have to arrange the meet up such as location, date, and time themselves which is not easy since one might have different time schedule for their daily lives. In addition, not everyone would like to meet up with you as some of them may not be ready yet or they simply do not want to ruin the texting chemistry [3]. Hence, there is a motivation to create a meet up function that would greatly facilitate the process of meeting up in order to let users have a better experience where they could be efficient and effective through this application.

c. Difficult in arranging meet-up for multiple person with shared interest

As mentioned above, there is no meet up function for users to facilitate the entire meeting up process. Especially when user not only wants to meet up one person, but

CHAPTER 1 INTRODUCTION

multiple person at the same time. Under this circumstance, user has to direct message to each and everyone for meet up invitation. Furthermore, users have to keep swiping in order to find one that match their own interests or hobbies. Even when they did find it, there will be no guarantee that the user would like to go out for a meet up. Hence, it causes a lot of time wasting and mental fatigues as they swipe on it for a long period to find the matched. However, according to Tinder matching algorithm for instances, high frequency in swiping left will ruin the user's Elo score - a score that determine user's potential matches [6]. The higher score means this particular user is more attractive and pickier, so the system will show this profile to another user with even higher Elo Score. Hence, this will cause the user hard to be matched as the other users are picky too [7]. Overall, find one with shared interest is hard and to find a group of people with shared interest is even harder. This happened to be a problem because for those busy schedule working user, it is hard to spare time in meeting up with others one-by-one [8]. If there is a function allows user to gather a group of people with shared interest, they could make their time more valuable. Hence, there is a motivation to build up an application that enable user hosting a meet-up that allows other interested users to join along.

1.2 Objectives

a. To analyse various dating application in the market based on their functions, strengths, and weaknesses in providing dating services.

This project is aimed to find out the shortcoming and the strengths of the various market applications in order to provide a new perspective of view and inspiration that should be accomplished in this project for a better user experience.

b. To design and develop an application that provides function hosting meet up and gathering to enable user have chances to meet each other in real person.

One of the main objectives in this project is to develop an application that enable user to interact with other users in person through hosting meet up functions and features. These functions are inspired from the analysis and review on various market application.

c. To evaluate and test the deliverable of the project in the aspect of functionality

Another objective is to review and evaluate the end deliverable of this project, which is the mobile application in functionality perspective in order to ensure error free.

1.3 Project Scope

The scope of this project is to develop and deliver a social networking and scheduling application that allows user to host a meet up to meet other users in person. This project will be broken down into two types of users, which are normal user and business user. Both type of user will have different functions to perform.

Normal user is expected to host meet up and gathering. With this function, user can schedule meet up by setting up necessary details such as date, time, number of participants, payment method, description and other personal preferences. Other users can choose to join the meet up, while the host will have the choice of accepting or rejecting application. Users can also edit their profile such as interest, preferences, self-description as references for other users.

User can also give feedback and review other users by rating stars and comments. Each user is expected to have their own rating shown in their profile. In the application, user is expected to have the function of chatting with other users too.

Host needs to close the registration for the meet up in order to create a group chat room with other joined users. Once the meet up registration is closed, user can also make booking if the business location is one of the registered user in our system and it is available for booking. For example, user can book the targeted restaurant or places that allows booking in the application.

Business user will have the function of receiving booking requests, accept or reject them, and promoting their business by enhancing their profile.

1.4 Impact, Significance and Contribution

The impact of this project is changing the way how a social networking application can be and how people can utilize it to expand their network connection, find friends, and life partner through this application. Among most of the popular online dating application, there is more than 25 million users all around the world engaging on the online dating application and this is only in United States area only [9]. This kind of application already be part of their lives and affect them in the way of finding life partners and expanding social network. Hence, the way of a dating application operates

CHAPTER 1 INTRODUCTION

will have huge impact on their users. However, the process of using the dating application is still in the traditional way which is swiping left and right on photos of other users to find matches.

By having this application, users can have a different way to make friends through the main function of this application, which is the meet up or gathering hosting. Whoever is interested in a meet up, they can apply and join it. Hence, no more swiping left and right, waiting the other one to swipe right on you, or keep wasting time to find a better option by swiping addictively. Users that are sharing the same interests can meet each other in an easy way. This could encourage users to engage each other by having face-to-face communication, unlike traditional dating application that most of the users are having texting chemistry only. The rating system in the application can also be one of the selling points of this application as it can be a tool for other users to avoid other users with inappropriate intentions and behaviours.

In addition, not only the users who wish to finding friend and partners, business user can also take part and get benefits from this application. This proposed application could possibly contribute to the local economy as users will have the function to choose a registered business as meet up venue and make booking within the application. Hence, business users can use it to receive booking and also promote their business on this platform.

1.5 Report Organization

This report is organized into 7 Chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Methodology, Chapter 4 System Design, Chapter 5 System Implementation, Chapter 6 System Evaluation and Discussion, Chapter 7 Conclusion and Recommendation.

The first chapter covers the introduction of this project including the problem statement, objectives, motivation and contribution, project scopes and report organization. The second chapter covers the literature review which sectioned into two part: the first part is where a few similar applications in market are reviewed based on their strengths, weaknesses and features; the second part is about the methodology reviews based on their strengths and weaknesses. The third chapter covers the system methodology with diagrams such as use case diagram, activity diagram and user interface designs. The fourth chapter is about the system design including the block diagram and system components interaction. The fifth chapter covers the system

CHAPTER 1 INTRODUCTION

implementation with a proper explanation on system operation and setup. The sixth chapter is about the system evaluation and discussion which included the system testing cases and results, objectives evaluation and project challenges. The seventh chapter covers the conclusion and recommendation for the future enhancement of this project.

CHAPTER 2 LITERATURE REVIEW

This literature review is split into two parts which are application reviews and methodology reviews. Application reviewed are Tinder, Bumble and Tantan. Methodologies reviewed are Rapid Application Development, Agile methodology and Spiral Methodology.

2.1 Existing Applications Review

2.1.1 Tinder

Tinder is a mobile application that enable user to build up social network connection, dating and meeting new people [10]. There are some features supported which are swipe left and right that enable users to match with other interested users that liked them back. It has more features can be unlocked by paying the subscription fees ranging from RM64.90 to RM359.90. Features under subscription such as swipe more a day, unlimited rewinds, see who likes you and etcetera [11],

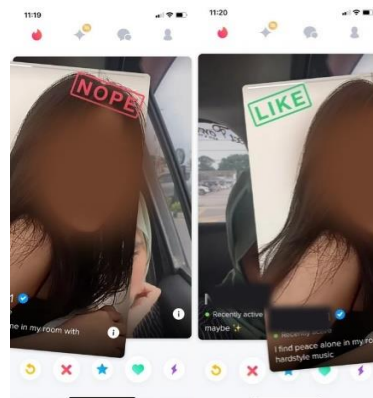


Figure 2.1 Swipe left or right feature of Tinder

Strengths:

Tinder is the world leading and most famous online dating application which recorded a total number of 7.86 million of audience size [9]. This has given an advantage where the user has more chances to be likely match with someone that they really interested due to a large size of users. Besides, its clean and simple user interface design enables user to know clearly on what to click and where it will be navigated to within the application. In addition, although there is subscription packages, user can still chat with the one they matched successfully without paying any cents. Besides, Tinder provides a function of photo verification which will match the user selfies and pose geometries from the photo uploaded in order to verify one is a real user. Tinder

CHAPTER 2 LITERATURE REVIEW

also comes with video calling function which could help user to avoid those fake account.

Weaknesses:

User will need to pay for subscription fees in order to unlock certain function and features which it always shows up as an advertisement. This could give a bad user experience since user will always see the advertisement and have to close it. Besides, user is not forced to perform the photo verification function. This might give an opportunity to those that trying to impersonate on the platform. It also will create a loophole for some party to create bots and fake accounts with fake professional photo. The most important feature which is the swipe left and right could also create a problem where user might judge the user based on their profile photos uploaded. This might arise a problem for those who does not upload any attractive photos.

2.1.2 Bumble

Bumble is the first online dating application that challenge the rules of dating where female user will have to chance to make the first move [12]. The CEO of Bumble stated that it is time to flip the table where women have to wait for a man to date and ask them out. It is the time for women to make their first move in building relationship [13]. Men users cannot directly chat with woman unless the opposite one liked them and initiate a conversation. The match will dispose after 24 hours if the woman user does not initiate any conversation.

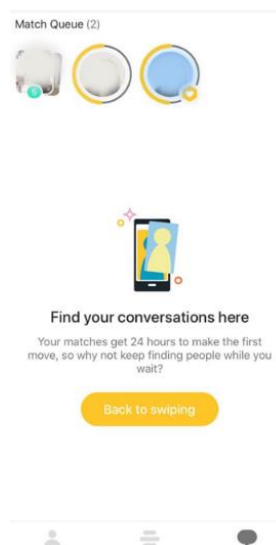


Figure 2.2 Woman on Bumble has to initiate conversation first

CHAPTER 2 LITERATURE REVIEW

Strengths:

It is a platform that empower the woman in finding their interested one instead of being picked by man user. It has a feature that will automatically blur out the nude image that sent by other users which give a protection to user from harassment. User can decide to see the image or not. Besides, Bumble has a moderator that will check and block the messages or account contains any inappropriate content that meets the criteria as stated in Bumble terms and condition. This could help to prevent users sending inappropriate or harass message on the platform [14]. Bumble also comes with an account verification function that allows user to upload a one-time selfie in order to verify oneself. User can also set up the filter such as education and hobbies to increase the chances of swiping the user with same interest.

Weaknesses:

Male user has to wait for female user to initiate the conversation, which the match will automatically dispose after 24 hours in worst case. Besides, users can only see who they liked before by paying subscription fees. In addition, male user without subscribing the package, has to wait for female user initiate conversation and cannot see who they have liked. Only two free advanced filters for user without subscription.

2.1.3 Tantan

Tantan is also a location-based dating application that pretty much similar to Tinder, where the only difference is Tantan is a Chinese dating application [15]. Tantan also consist of the swipe left and right feature to let user find their interested one. It has more than 300 million registered user on the application with more than 10 million active users in daily usage [16]. It also comes with subscription package which can unlock features such as swipe more, unlimited likes, rewinds, hide last seen and etcetera.



Figure 2.3 Swipe left or right feature on Tantan

Strengths:

Tantan has a feature of blocking the contacts on user phone which can give a private social life to user. Its photo verification allows user to upload selfie for self-verification and they will obtain a badge after verified. This could help another user to identify whether the user they liked is a real person or not.

Weaknesses:

Video calling is not supported within the application itself where it could lead to inconvenience for user since they have to video call using other application. It is not forcing user to take the photo verification functions which create a loophole for bots and fake account. Its swipe left and right feature is the only way that user to get connect with others, which is a disadvantage to those does not upload attractive photos.

Summary:

	Tinder	Bumble	Tantan
Swipe left or right feature	√	√	√
Free chat function	√	√	√
Video call supported	√	-	-
Subscription fee for advanced feature per month	RM36.90	RM84.90	RM35.90
Photo verification	√	√	√
Block contact	-	-	√
Advanced filter for matching	-	√	-

Table 2.1 Summary of application review

2.2 Methodologies Review

2.2.1 Rapid Application Development (RAD)

RAD is a methodology that emphasis more on development rather than planning phase. The project will be broken down into smaller modules which allows a faster development process. It involves several components such as Joint Application Development (JAD) session, timebox, incremental prototyping and Computer-Aided Software Engineering (CASE) tools. The prototyping is the most important component in RAD as it focus on iterative model. The development team will produce prototype and present it to customer for feedback and review. At least three time of repetition for inspection, discussion, and amendment in order to product final model that satisfy all

CHAPTER 2 LITERATURE REVIEW

key stakeholders and client [17]. As stated by Geambasu [18], there are 5 main stages in RAD which are initialization, requirement, design, implementation, and deployment. Each stages will go through several phases and each phases will go through three steps which are preparation, session, and conclusion.

Strengths:

It promotes user involvement and participation in any of the development phases. For example, a JAD session is hold for communication between project team, stakeholders and customers. It allows the requirement to be defined more details during each of the iterative session with the help of prototype. Besides, it is suitable for low budget project since the prototype produced is reusable. The whole project development time is also shorter as RAD focus on timebox approach which can deliver the product within the due date. The team would rather cut off some requirement instead of extending the deadline [18].

Weaknesses:

Although it promotes high user participation, but the client also required to have experience and knowledge in the field or problem domain. The final product may miss out some significant requirement since the development team is working quickly in each of the iteration to produce the prototype. RAD is not suitable for large and complex system as it allows poor defined of user requirement at the early stage, which is not recommended for complex project [18]. Since the project is broken down into modules and smaller segment, the team has to put more attention in the integration of coding [19].

2.2.2 Agile Methodology – Extreme Programming (XP)

Agile literally means easy and quick moving. Hence, this methodology allows project development process able to adapt to new changes very quickly in order to ensure the cost and customer satisfaction. This could be easily fulfilled in Agile as it is an iterative and incremental model approach. Extreme Programming (XP) is one of the frameworks under Agile. It allows direct communication between clients and extreme programmer team. Extreme Programmers will work together on codes production and keep the system constantly integrating to improve on it as needed after communicating with the clients [20]. There are 5 main stages which are planning, design, coding, testing

CHAPTER 2 LITERATURE REVIEW

and release [21]. Paulk [22] pointed out that this XP cycle is included the four core and basic activities that developer should carry out which is designing, coding, testing, and listening.

Strengths:

It allows client with little requirement defined at early stage since it promotes communication and feedback between client and XP team to obtain requirement during development process. Since it allows project team to constantly work with client, it is more flexible to adapt new changes of client requirement into the system at any stages of the life cycle. It is suitable for low budget or small project as the XP team required lesser human resources [18]. Since it is a lighter team, with only few developer, it is easy to perform iteration and testing on the system.

Weaknesses:

Difficult to use in large scale project due to insufficient planning and poor up-front documentation as it focus on early result of project model [23]. It is also difficult for development effort estimation of the whole project as not all requirements are defined clearly in early stages of XP.

2.2.3 Spiral Methodology

Spiral development methodology includes both of the characteristic of Agile and Waterfall methodology as it combines top-down and bottom-up approach [24]. It is a risk-driven approach where all the activities are planned to reduce risk. Each iteration of spiral model consists of 4 phases which are planning, risk analysis, development, and evaluation. A model is developed at the end of each iteration and it is being reviewed by the key stakeholders. The model will not be discarded, instead, it will be extended to next iteration to adjust and extend from it until the final model meets the user requirement and satisfaction [19].

Strengths:

It does not require fully defined user requirement at the early stage as it is an iterative approach where user requirement can be specifically increasing [25]. Besides, it has a high ability in identifying project risk as risk management is the main focus of this methodology. It also allows stakeholders to obtain functional model to review and

CHAPTER 2 LITERATURE REVIEW

give feedback for a better model in the end of next iteration. In addition, it also provides a better time and cost analysis. Time and cost is required to be analysed in each of the iteration during risk analysis phase as it is one of the important criteria to determine the project risk [26].

Weaknesses:

The success of project is highly dependent on the accuracy of risk analysis from experts since it is the main focus of this methodology. Otherwise, a non-expert team will be providing an opposite report from the expert and causing wrong decision for whoever review on that [27]. This also link to the next weakness where it is a costly methodology since it required experienced experts to perform risk analysis. It is also not suitable to low-risk or small project aiming for low budget.

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

3.1 System Design Diagram

3.1.1 Use-case Diagram

a) Normal User Module

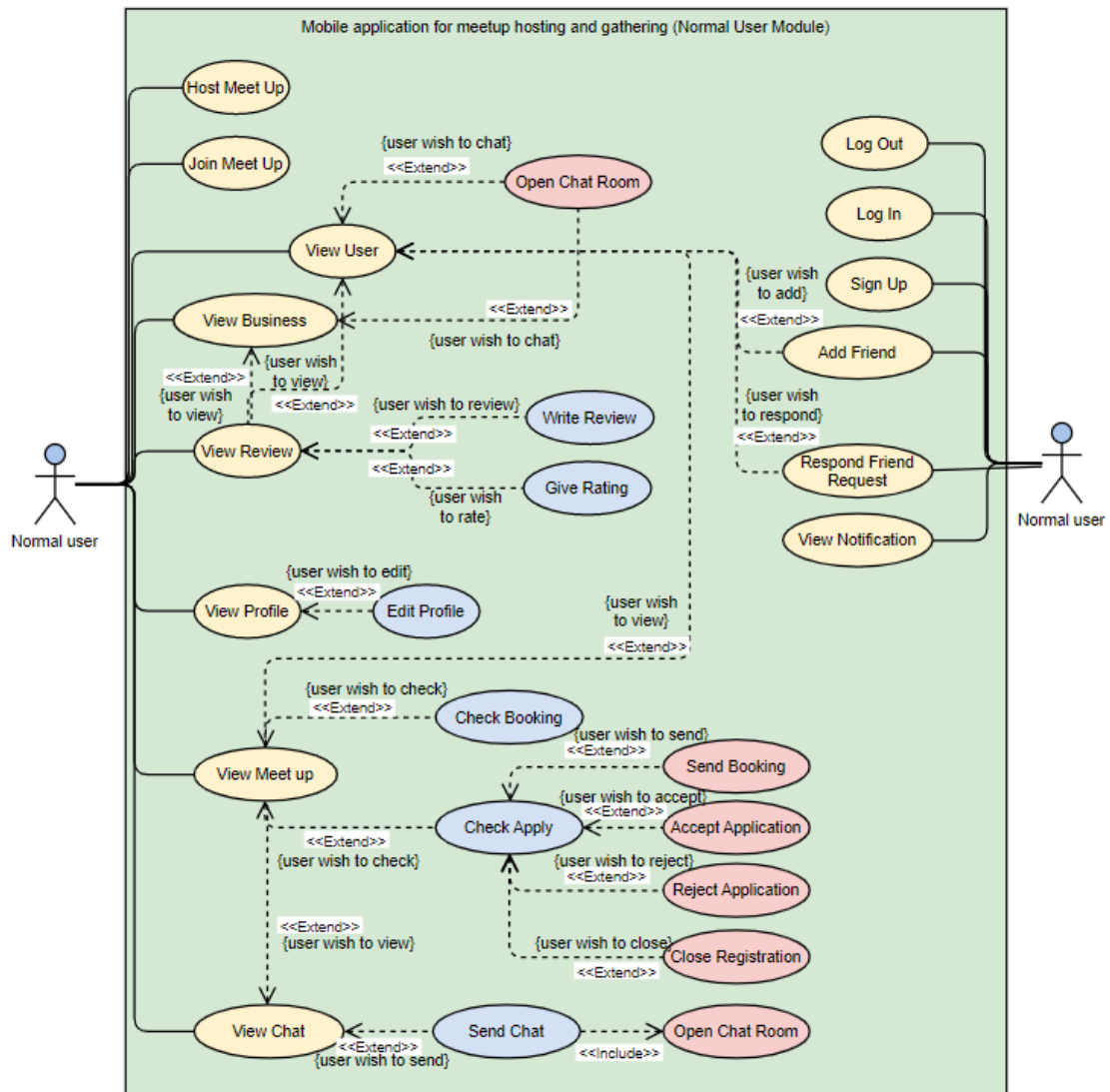


Figure 3.1 Normal User Use Case Diagram

Host meet up	
Description	User wants to host a meet up
Actor	Normal user
Assumption(s)	User wants to host a meet up
Precondition(s)	User is logged into the home page
Postcondition(s)	User hosted a meet up and back to home page
Primary Path	1. User clicks on host meet up button

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

	<ol style="list-style-type: none"> 2. User keys in necessary input 3. User clicks publish button 4. System stores the meet up information 5. User is returned to the home page
Alternate Path(s)	2. User customized location from the maps. Use case resumes at 3.
Exception Path(s)	User is warned by not filling up all the required input at 2.

Table 3.1 Host meet up

Join meet up	
Description	User wants to join a meet up
Actor	Normal user
Assumption(s)	User saw an interesting meet up
Precondition(s)	User is logged into the home page
Postcondition(s)	User joined a meet up
Primary Path	<ol style="list-style-type: none"> 1. User clicks on any meet up 2. User clicks on the join button 3. System stores the data 4. System shows the cancel button
Alternate Path(s)	4. User clicks cancel button. System stores the data and shows join button. Use case resumes at 2.

Table 3.2 Join meet up

View user	
Description	User wants to view another user's profile
Actor	Normal user
Assumption(s)	User wants to view another user's profile
Precondition(s)	User is logged in
Postcondition(s)	User is landed in an user's profile page
Primary Path	<ol style="list-style-type: none"> 1. User clicks on user's header icon 2. System gets the data 3. User is landed in the user's profile page
Alternate Path(s)	3. User clicks review tab. Use case proceeds with View review case.

	<ol style="list-style-type: none"> 3. User clicks meet up tab. Use case proceeds with View meet up case. 3. User clicks add friend button. Use case proceeds with Add friend case. 3. User clicks message button. Use case proceeds with Open chat room case. 3. User clicks respond button. Use case proceeds with Respond friend request case.
--	--

Table 3.3 View user

View business	
Description	User wants to view a business user's profile
Actor	Normal user
Assumption(s)	User wants to view a business user's profile
Precondition(s)	User is logged in
Postcondition(s)	User is landed in a business user's profile page
Primary Path	<ol style="list-style-type: none"> 1. User clicks on business user's header 2. System gets the data 3. User is landed in the business user's profile page
Alternate Path(s)	<ol style="list-style-type: none"> 1. User clicks the meet up location in the meet up page. Use case resumes at 2. 3. User clicks review tab. Use case proceeds with View review case. 3. User clicks message button. Use case proceeds with Open chat room case.

Table 3.4 View business

Give rating	
Description	User wants to rate a business or user
Actor	Normal user
Assumption(s)	User wants to rate a business or user
Precondition(s)	User is logged in
Postcondition(s)	User rated a business or user
Primary Path	<ol style="list-style-type: none"> 1. User input the number of stars 2. User clicks send button

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

	3. System stores the data
Exception Path(s)	User is warned if they click send button without giving stars

Table 3.5 Give rating

Write review	
Description	User wants to write a review for business or user
Actor	Normal user
Assumption(s)	User wants to write a review for business or user
Precondition(s)	User is logged in
Postcondition(s)	User gave a review for business or user
Primary Path	<ol style="list-style-type: none"> 1. User input reviews 2. User clicks send button 3. System stores the data
Exception Path(s)	User is warned if they click send button without input review

Table 3.6 Write review

View profile	
Description	User wants to view own profile
Actor	Normal user
Assumption(s)	User wants to view own profile
Precondition(s)	User is logged in
Postcondition(s)	User is landed in their own profile
Primary Path	<ol style="list-style-type: none"> 1. User clicks on profile icon in bottom navigation bar 2. System gets the data 3. User is landed in own profile page
Alternate Path(s)	3. User clicks edit button. Use case proceeds at Edit profile case.

Table 3.7 View profile

Edit profile	
Description	User wants to edit profile
Actor	Normal user
Assumption(s)	User wants to edit profile
Precondition(s)	User is logged in and landed in their own profile page
Postcondition(s)	User edited their information
Primary Path	<ol style="list-style-type: none"> 1. User clicks on edit button

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

	<ol style="list-style-type: none"> 2. System gets the data 3. User edits the information 4. User clicks done 5. System stores the data 6. User is returned to profile page
Exception Path(s)	User is warned by submitting empty information for required fields.

Table 3.8 Edit profile

View review	
Description	User wants to view review of own, other users or business user
Actor	Normal user
Assumption(s)	User wants to view review of own, other users or business user
Precondition(s)	User is logged in and landed in profile page
Postcondition(s)	User is landed in review page
Primary Path	<ol style="list-style-type: none"> 1. User clicks review tab 2. System gets the data 3. System displays the data in review page
Alternate Path(s)	<ol style="list-style-type: none"> 3. User wants to rate. Use case proceeds with Give rating case. 3. User wants to write review. Use case proceeds with Write review case.

Table 3.9 View review

Check booking	
Description	User wants to check booking detail
Actor	Normal user
Assumption(s)	User has sent booking before
Precondition(s)	User is in the meet up application page
Postcondition(s)	User is landed in booking detail page
Primary Path	<ol style="list-style-type: none"> 1. User clicks check booking button 2. System gets the data 3. User is landed in booking detail page
Alternate Path(s)	<ol style="list-style-type: none"> 1. User clicks from the booking notification. Use case resumes at 2.

Table 3.10 Check booking

Check apply	
Description	User wants to check application list of meet up
Actor	Normal user
Assumption(s)	User has received application from other users
Precondition(s)	User is logged in and in the meet up page
Postcondition(s)	User is landed in application page
Primary Path	<ol style="list-style-type: none"> 1. User clicks check button in meet up page 2. System gets the data 3. User is landed in application page
Alternate Path(s)	<ol style="list-style-type: none"> 3. User clicks send booking. Use case proceeds at Send booking case. 3. User clicks close registration. Use case proceeds at Close registration case. 3. User clicks accept or reject. Use case proceeds at Accept or Reject application case.

Table 3.11 Check apply

Send booking	
Description	User wants to send a booking request for the meet up location
Actor	Normal user
Assumption(s)	User selected a registered business user as location
Precondition(s)	<p>User is logged in and in the application page.</p> <p>The location selected is a registered business user and allow for booking.</p> <p>User has closed registration.</p>
Postcondition(s)	User has sent booking request
Primary Path	<ol style="list-style-type: none"> 1. User click send booking button 2. User keys in information 3. User clicks send button 4. System stores the data 5. User is returned to application page
Alternate Path(s)	<ol style="list-style-type: none"> 2. User edits some preloaded information. Use case resumes at 3.

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Exception Path(s)	User is warned by sending empty information.
-------------------	--

Table 3.12 Send booking

Accept application	
Description	User wants to accept application of other users.
Actor	Normal user
Assumption(s)	User has received application from other users.
Precondition(s)	User is logged in and in the meet up page
Postcondition(s)	User has accepted application(s)
Primary Path	<ol style="list-style-type: none"> 1. User clicks accept button 2. System stores the data
Exception Path(s)	User is warned if the amount of guest accepted is exceeded.

Table 3.13 Accept application

Reject application	
Description	User wants to reject application of other users.
Actor	Normal user
Assumption(s)	User has received application from other users.
Precondition(s)	User is logged in and in the meet up page
Postcondition(s)	User has rejected application(s)
Primary Path	<ol style="list-style-type: none"> 1. User clicks reject button 2. System stores the data

Table 3.14 Reject application

Close registration	
Description	User wants to close registration
Actor	Normal user
Assumption(s)	User is satisfied with the number of guest accepted
Precondition(s)	User is logged in and in the application page. There is at least one application accepted.
Postcondition(s)	User has closed registration and a group chat is created
Primary Path	<ol style="list-style-type: none"> 1. User clicks close button 2. System stores the data 3. System generated a group chat 4. System checks if the meet up venue is allowed for booking

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

	5. System displays send booking button
Alternate Path(s)	4. The venue is not available for booking. Use case stops at 4
Exception Path(s)	User is warned if there is no accepted guest at step 1

Table 3.15 Close registration

View chat	
Description	User wants to view the chat records
Actor	Normal user
Assumption(s)	User wants to view the chat records
Precondition(s)	User is logged in
Postcondition(s)	User is landed in chat page
Primary Path	<ol style="list-style-type: none"> 1. User clicks the chat icon in bottom navigation bar 2. System gets the data 3. User is landed in chat page
Alternative Path(s)	<ol style="list-style-type: none"> 3. User clicks any chat item. System gets the data and land user in chat room.

Table 3.156 View chat

Send chat	
Description	User wants to send a message to the other users.
Actor	Normal user
Assumption(s)	User wants to send a message
Precondition(s)	User is logged in and in the chat room
Postcondition(s)	User sent message
Primary Path	<ol style="list-style-type: none"> 1. User input messages 2. User click send 3. System stores the data
Alternative Path(s)	<ol style="list-style-type: none"> 3. User wants to send again. Use case resumes at 1.

Table 3.17 Send chat

Open chat room	
Description	User wants to chat with the other users
Actor	Normal user
Assumption(s)	User wants to view the message records and chat with users
Precondition(s)	User is logged in

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Postcondition(s)	User is landed in chat room
Primary Path	<ol style="list-style-type: none"> 1. User clicks any chat in chat page 2. System gets the data 3. User is landed in the chat room
Alternate Path(s)	<ol style="list-style-type: none"> 1. User clicks on other users' icon. Use case resumes at 2. 1. User clicks on message button. Use case resumes at 2. 2. System did not find related data. System creates a new chat room. Use case resumes at 3.

Table 3.18 Open chat room

Add friend	
Description	User wants to add the other users as friend
Actor	Normal user
Assumption(s)	User wants to be friend with other users
Precondition(s)	User is logged in and in the other users' profile page
Postcondition(s)	User has sent friend request
Primary Path	<ol style="list-style-type: none"> 1. User clicks add friend button 2. System stores the data 3. System displays cancel button

Table 3.19 Add friend

Respond friend request	
Description	User wants to accept or reject friend request
Actor	Normal user
Assumption(s)	User has received friend request
Precondition(s)	User is logged in and in the profile page of requested user
Postcondition(s)	User responded the friend request
Primary Path	<ol style="list-style-type: none"> 1. User clicks respond button 2. System shows a bottom sheet with accept or reject button 3. User clicks accept button 4. System stores the data 5. System displays message button
Alternate Path(s)	<ol style="list-style-type: none"> 3. User clicks reject button. Use case resumes at 4 and ended.

Table 3.20 Respond friend request

View notification	
Description	User wants to view the notification records
Actor	Normal user
Assumption(s)	User wants to view the notification records
Precondition(s)	User is logged in
Postcondition(s)	User is able to see notification records
Primary Path	<ol style="list-style-type: none"> 1. User clicks notification icon at bottom navigation bar 2. System gets the data 3. User is landed in notification page 4. User clicks any of the notification item 5. System gets the data 6. User is landed in relevant pages
Alternate Path(s)	4. User did not click any item. Use case stops at 3.

Table 3.21 View notification

b) Business User Module

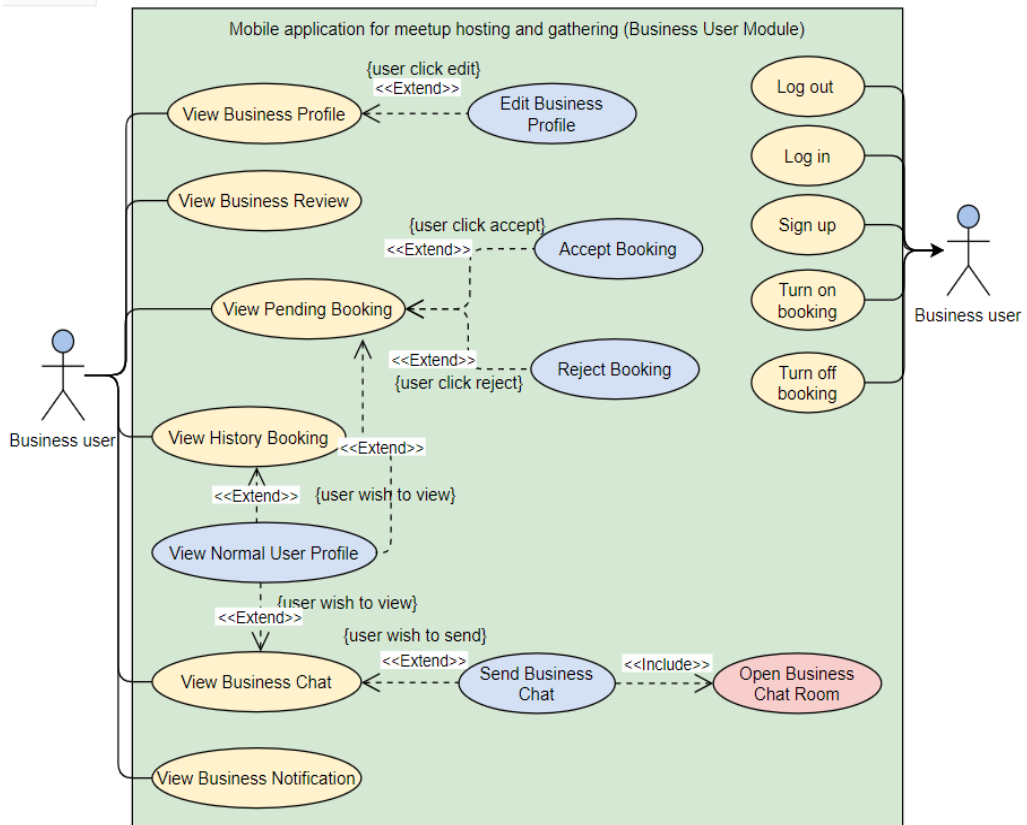


Figure 3.2 Business User Use Case Diagram

View business profile	
Description	User wants view business profile information
Actor	Business user
Assumption(s)	User wants to view business information
Precondition(s)	User is logged in
Postcondition(s)	User is landed in profile page
Primary Path	<ol style="list-style-type: none"> 1. User clicks profile icon in bottom navigation bar 2. System gets the data 3. User is landed in profile page
Alternate Path(s)	<ol style="list-style-type: none"> 3. User wants to edit information. System gets the data and open edit page.

Table 3.22 View business profile

Edit business profile	
Description	User wants to edit business information
Actor	Business user
Assumption(s)	User wants to edit business information
Precondition(s)	User is logged in and in the profile page
Postcondition(s)	User edited the business information
Primary Path	<ol style="list-style-type: none"> 1. User clicks edit button 2. System gets the data 3. User edits information 4. User clicks done 5. System stores the data
Exception Path(s)	User is warned by not filling up all the required input at 4.

Table 3.23 Edit business profile

View history booking	
Description	User wants to view history booking request
Actor	Business user
Assumption(s)	User has responded booking requests before
Precondition(s)	User is logged into the home page
Postcondition(s)	User can see a list of history booking request
Primary Path	<ol style="list-style-type: none"> 1. User clicks the history tab 2. System gets the data

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

	3. System display request details
Alternate Path(s)	3. User clicks the requester's header. System gets the data and user is landed in profile page.

Table 3.24 View history booking

View pending booking	
Description	User wants to view pending booking request
Actor	Business user
Assumption(s)	User has received booking applications
Precondition(s)	User is logged into the home page
Postcondition(s)	User can see a list of pending booking request
Primary Path	<ol style="list-style-type: none"> 1. User clicks the pending tab 2. System gets the data 3. System display request details
Alternate Path(s)	<ol style="list-style-type: none"> 3. User clicks the requester's header. System gets the data and user is landed in profile page. 3. User clicks the accept or reject button. System stores the data.

Table 3.25 View pending booking

Accept booking	
Description	User wants to accept booking request
Actor	Business user
Assumption(s)	User has received booking request
Precondition(s)	User is logged in and in the booking detail page
Postcondition(s)	User has accepted booking request
Primary Path	<ol style="list-style-type: none"> 1. User clicks accept button 2. System stores the data

Table 3.26 Accept booking

Reject booking	
Description	User wants to reject booking request
Actor	Business user
Assumption(s)	User has received booking request
Precondition(s)	User is logged in and in the booking detail page
Postcondition(s)	User has rejected booking request

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Primary Path	<ol style="list-style-type: none"> 1. User clicks accept button 2. System stores the data
--------------	---

Table 3.27 Reject booking

View business review	
Description	User wants to view review of own or other users
Actor	Business user
Assumption(s)	User wants to view review of own or other users
Precondition(s)	User is logged in and landed in profile page
Postcondition(s)	User is landed in review page
Primary Path	<ol style="list-style-type: none"> 1. User clicks review tab 2. System gets the data 3. System displays the data in review page

Table 3.28 View business review

View business notification	
Description	User wants to view the notification records
Actor	Business user
Assumption(s)	User wants to view the notification records
Precondition(s)	User is logged in
Postcondition(s)	User is able to see notification records
Primary Path	<ol style="list-style-type: none"> 1. User clicks notification icon at bottom navigation bar 2. System gets the data 3. User is landed in notification page 4. User clicks any of the notification item 5. System gets the data 6. User is landed in relevant pages
Alternate Path(s)	<ol style="list-style-type: none"> 4. User did not click any item. Use case stops at 3.

Table 3.29 View business notification

View business chat	
Description	User wants to view the chat records
Actor	Business user
Assumption(s)	User wants to view the chat records
Precondition(s)	User is logged in
Postcondition(s)	User is landed in chat page

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Primary Path	<ol style="list-style-type: none"> 1. User clicks the chat icon in bottom navigation bar 2. System gets the data 3. User is landed in chat page
--------------	--

Table 3.30 View business chat

Send business chat	
Description	User wants to chat with other users
Actor	Business user
Assumption(s)	User wants to send a message to another user
Precondition(s)	User is logged in and in the chat room
Postcondition(s)	User sent a message
Primary Path	<ol style="list-style-type: none"> 1. User types messages 2. User clicks send button 3. System stores the data
Alternate Path(s)	<ol style="list-style-type: none"> 3. User wants to send again. Use case resumes at 1.

Table 3.31 Send business chat

Open business chat room	
Description	User wants to chat with normal user
Actor	Business user
Assumption(s)	User wants to chat with normal user
Precondition(s)	User is logged in
Postcondition(s)	User is landed in the chat room
Primary Path	<ol style="list-style-type: none"> 1. User clicks on any chat records item 2. System gets the data 3. User is landed in chat room
Alternate Path(s)	<ol style="list-style-type: none"> 1. User clicks on message button. Use case resumes at 2. 2. System did not find related data. System creates a new chat room. Use case resumes at 3.

Table 3.32 Open business chat room

View normal user profile	
Description	User wants to view normal user's profile
Actor	Business user
Assumption(s)	User wants to view normal user's profile
Precondition(s)	User is logged in

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Postcondition(s)	User is landed in profile page
Primary Path	<ol style="list-style-type: none"> 1. User clicks any user's header icon 2. System gets the data 3. User is landed in normal user profile
Alternate Path(s)	<ol style="list-style-type: none"> 3. User clicks message button. System gets the data. User is landed in chat room.

Table 3.33 View normal user profile

Turn on booking	
Description	User wants to turn on booking option
Actor	Business user
Assumption(s)	User wants to turn on booking option
Precondition(s)	User is logged into the home page
Postcondition(s)	User turned on booking option
Primary Path	<ol style="list-style-type: none"> 1. User turn on the booking switch 2. System stores the data

Table 3.34 Turn on booking

Turn off booking	
Description	User wants to turn off booking option
Actor	Business user
Assumption(s)	User wants to turn off booking option
Precondition(s)	<p>User is logged into the home page.</p> <p>There is no pending booking.</p>
Postcondition(s)	User turned off the booking option
Primary Path	<ol style="list-style-type: none"> 1. User turns off the booking switch 2. System stores the data
Exception Path(s)	User is warned because there are pending bookings at step 1.

Table 3.35 Turn off booking

c) Shared use case of normal and business user:

Log in	
Description	User wants to log in to the application
Actor	Normal or business user
Assumption(s)	User has account and wants to log in

CHAPTER 3 SYSTEM METHODOLOGY/APPROACH

Precondition(s)	User has account
Postcondition(s)	User logged into the home page
Primary Path	<ol style="list-style-type: none"> 1. User input credentials 2. System logs in the account 3. User is landed in the home page
Alternative Path(s)	<ol style="list-style-type: none"> 3. Normal user is redirected to normal user dashboard 3. Business user is redirected to business user dashboard
Exception Path(s)	User is warned because the credentials are wrong at step 1.

Table 3.36 Log in

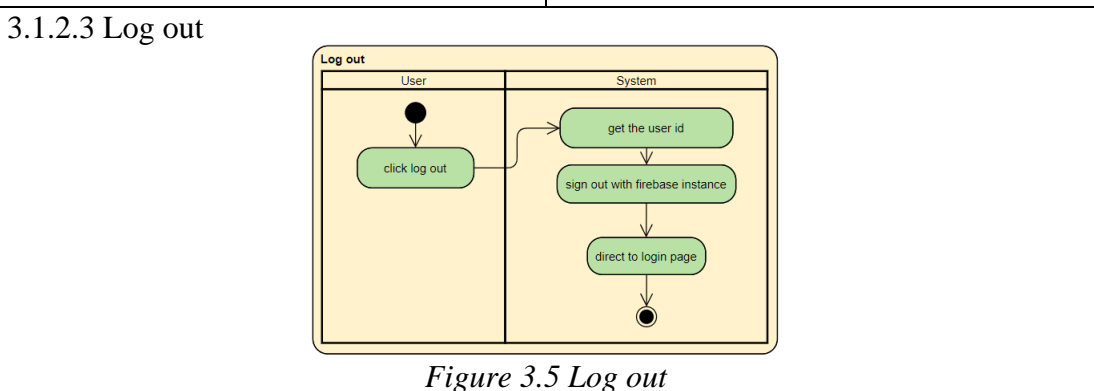
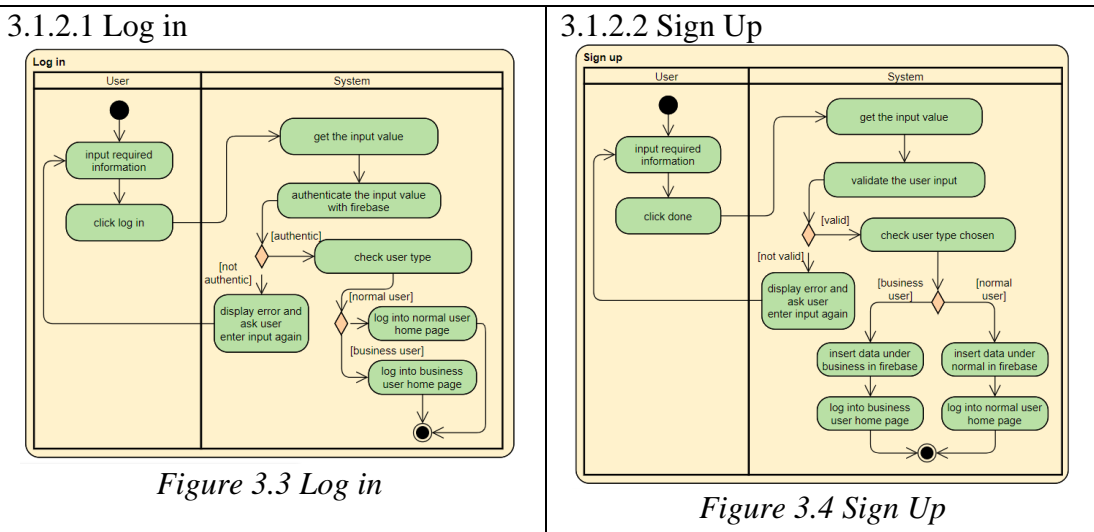
Log out	
Description	User wants to log out of application
Actor	Normal or business user
Assumption(s)	User wants to log out of application
Precondition(s)	User is logged in
Postcondition(s)	User is logged out
Primary Path	<ol style="list-style-type: none"> 1. User clicks log out button 2. System logs out the account 3. User is redirected to log in page

Table 3.37 Log out

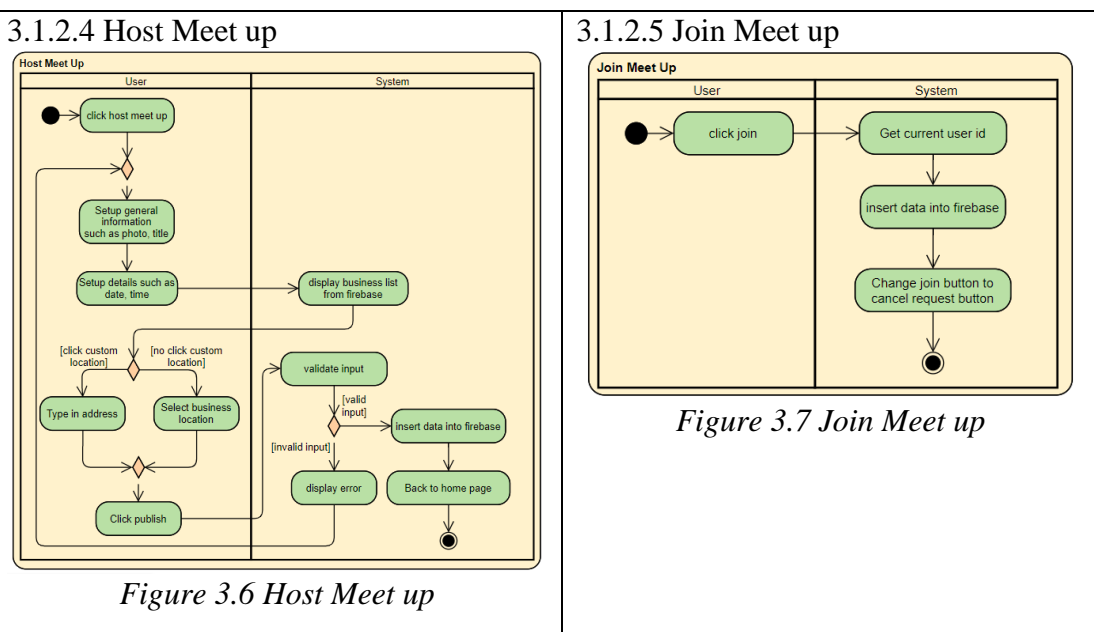
Sign up	
Description	User wants to sign up an account
Actor	Normal or business user
Assumption(s)	User wants to sign up an account
Precondition(s)	
Postcondition(s)	User signed up an account and landed in home page
Primary Path	<ol style="list-style-type: none"> 1. User clicks sign up button 2. User key in required data 3. User clicks done 4. System gets the data and verify 5. User is redirected to home page
Alternative Path(s)	<ol style="list-style-type: none"> 5. Normal user is redirected to normal user dashboard 5. Business user is redirected to business user dashboard
Exception Path(s)	User is warned because there are invalid input at step 2.

Table 3.38 Sign up

3.1.2 Activity Diagram



a) Normal user module:



3.1.2.6 View User

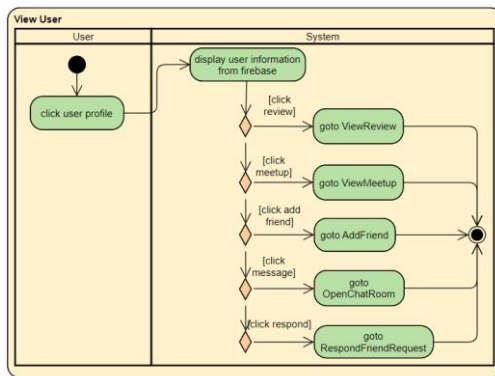


Figure 3.8 View User

3.1.2.7 View Business

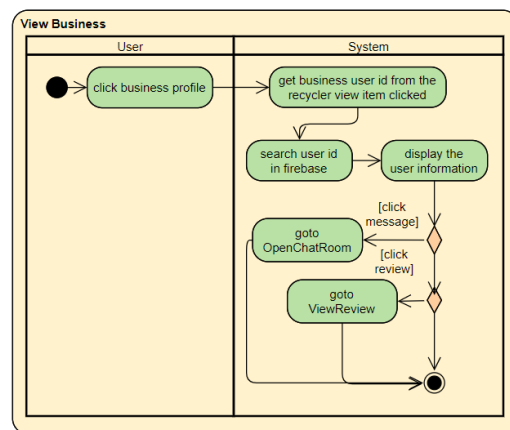


Figure 3.9 View Business

3.1.2.8 Write Review

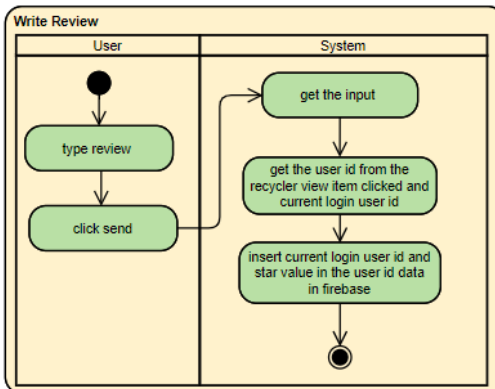


Figure 3.10 Write Review

3.1.2.9 Give Rating

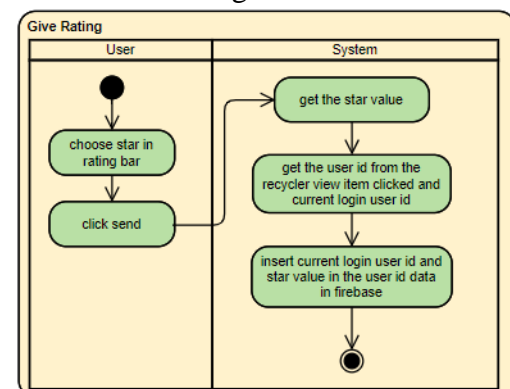


Figure 3.11 Give Rating

3.1.2.10 View Profile

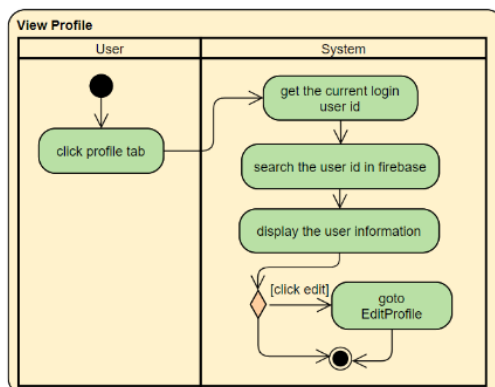


Figure 3.12 View Profile

3.1.2.11 Edit Profile

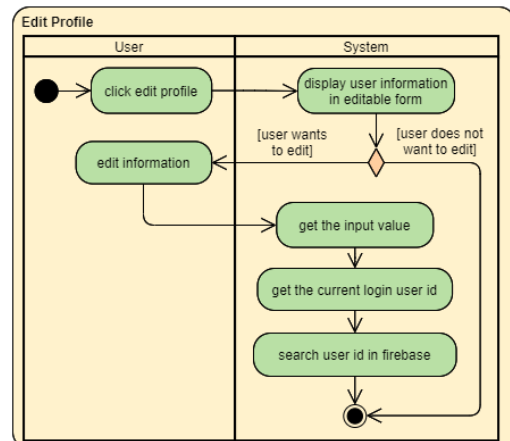


Figure 3.3 Edit Profile

3.1.2.12 View Review

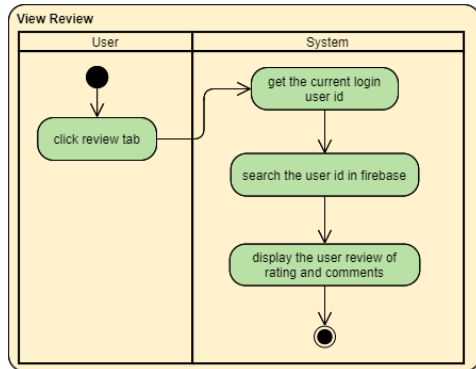


Figure 3.14 View Review

3.1.2.13 View Meet up

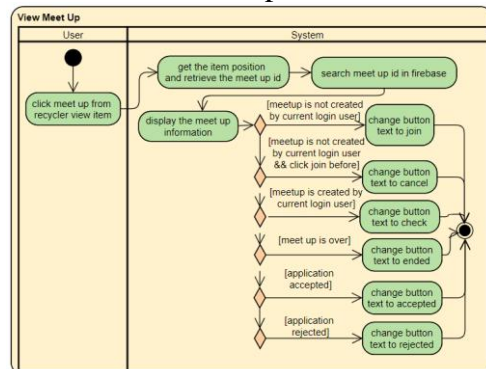


Figure 3.15 View Meet up

3.1.2.14 Check Booking

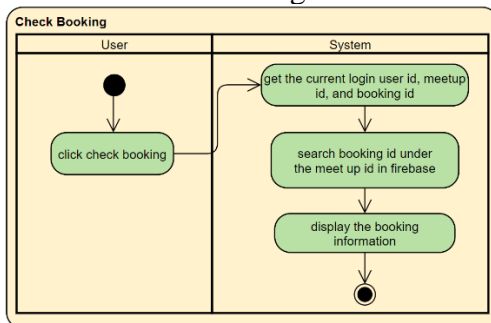


Figure 3.5 Check Booking

3.1.2.15 Check Apply

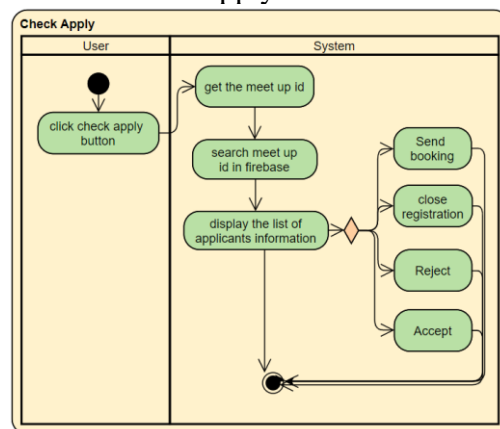


Figure 3.17 Check Apply

3.1.2.16 Accept Application

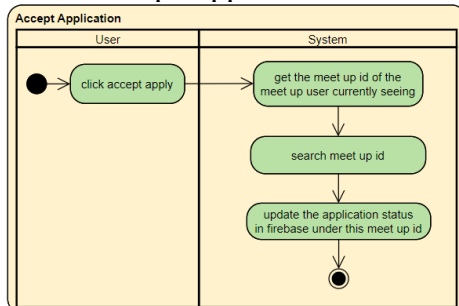


Figure 3.18 Accept Application

3.1.2.17 Reject Application

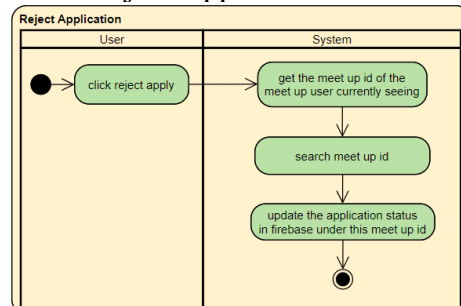


Figure 3.19 Reject Application

3.1.2.18 Close Registration

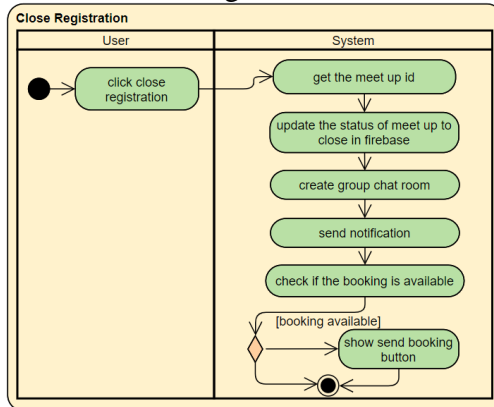


Figure 3.20 Close Registration

3.1.2.19 Send Booking

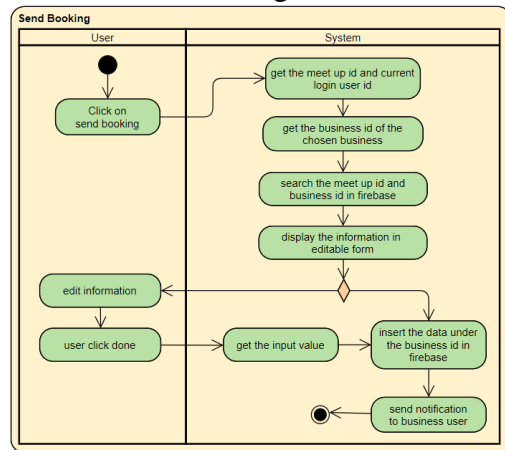


Figure 3.21 Send Booking

3.1.2.20 View Chat

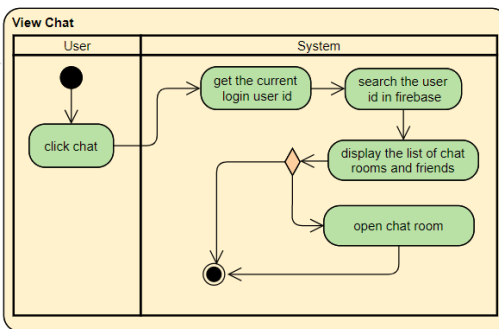


Figure 3.22 View Chat

3.1.2.21 Send Chat

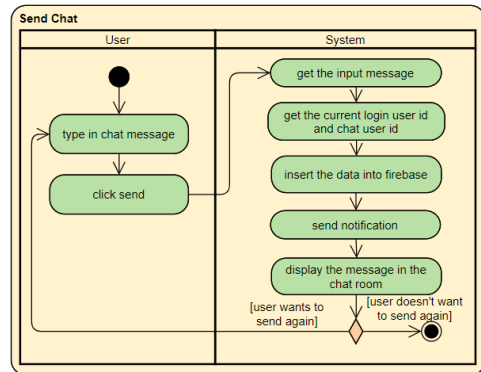


Figure 3.23 Send Chat

3.1.2.22 Open Chat Room

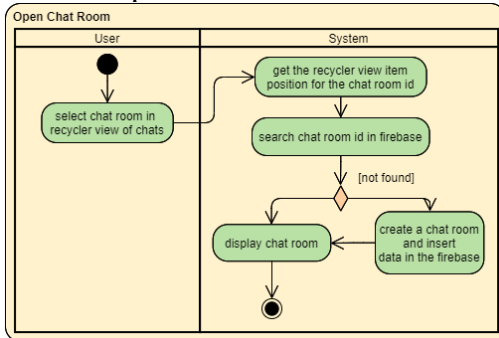


Figure 3.24 Open Chat Room

3.1.2.23 Add Friend

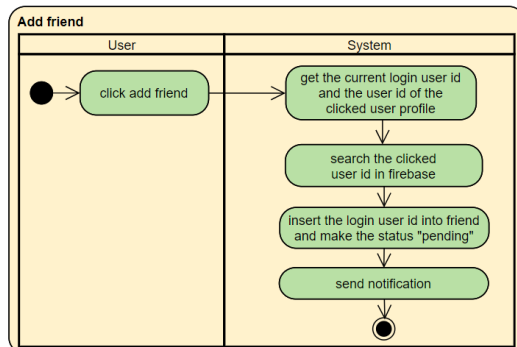


Figure 3.25 Add Friend

3.1.2.24 Respond Friend Request

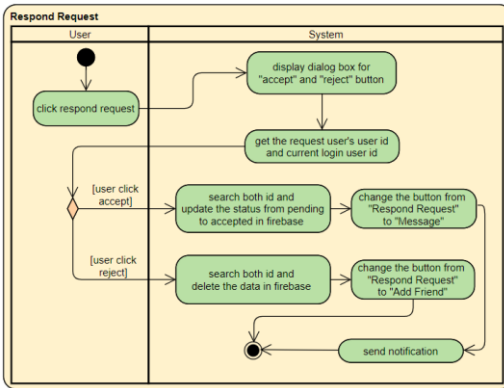


Figure 3.26 Respond Friend Request

3.1.2.25 View Notification

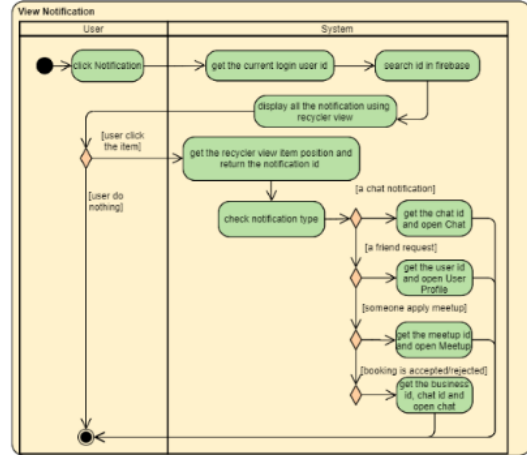


Figure 3.27 View Notification

b) Business user module

3.1.2.26 View Business Profile

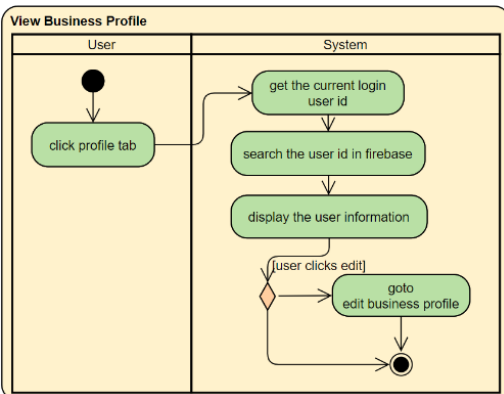


Figure 3.28 View Business Profile

3.1.2.27 Edit Business Profile

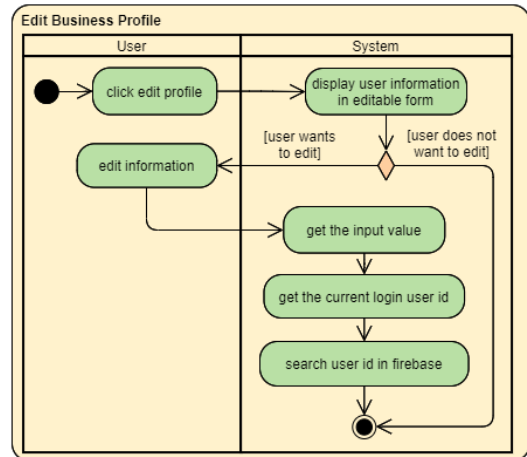


Figure 3.29 Edit Business Profile

3.1.2.28 View History Booking

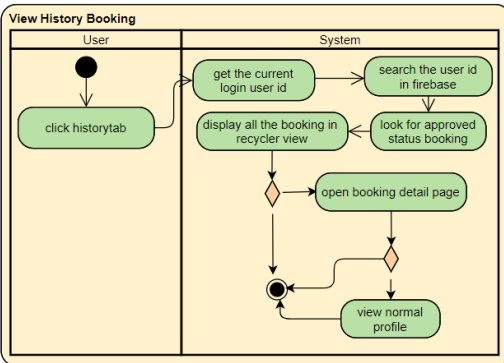


Figure 3.30 View History Booking

3.1.2.29 View Pending Booking

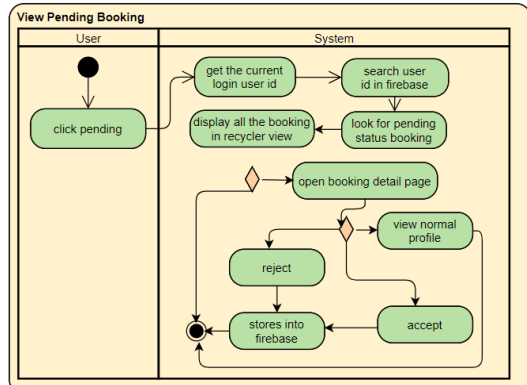


Figure 3.31 View Pending Booking

3.1.2.30 Accept Booking

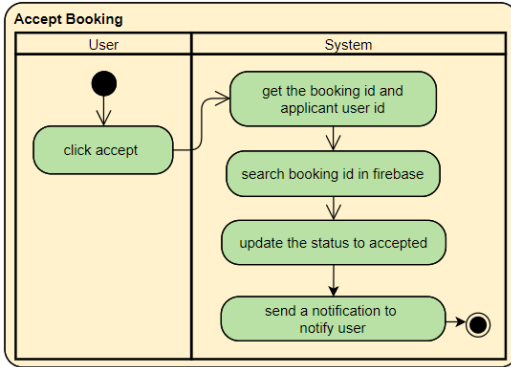


Figure 3.32 Accept Booking

3.1.2.31 Reject Booking

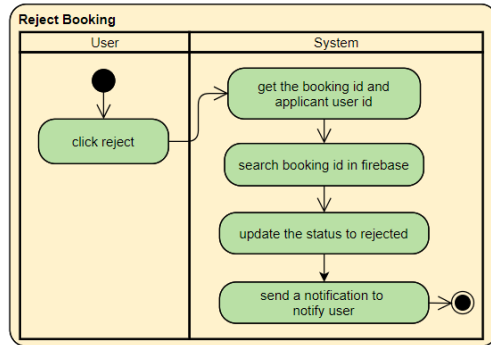


Figure 3.33 Reject Booking

3.1.2.32 View Business Review

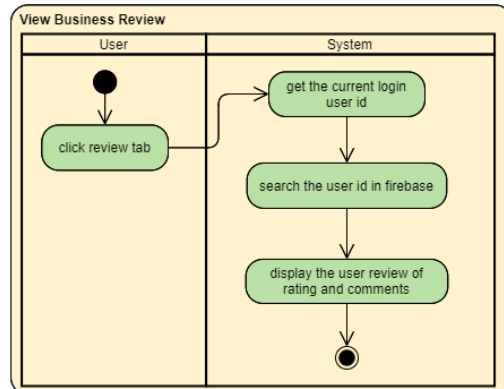


Figure 3.34 View Business Review

3.1.2.33 View Business Notification

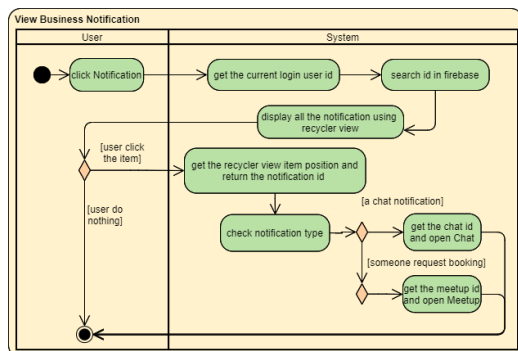


Figure 3.35 View Business Notification

3.1.2.34 View Business Chat

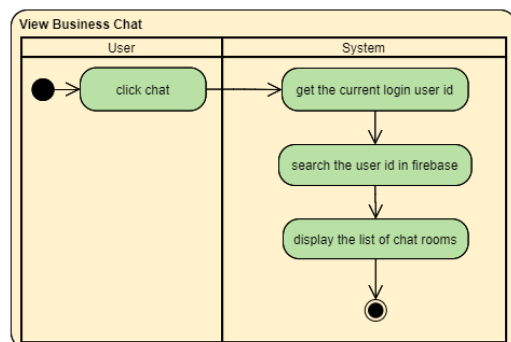


Figure 3.36 View Business Chat

3.1.2.35 Send Business Chat

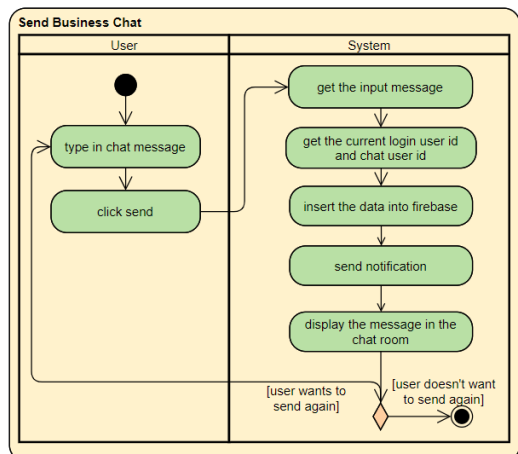


Figure 3.37 Send Business Chat

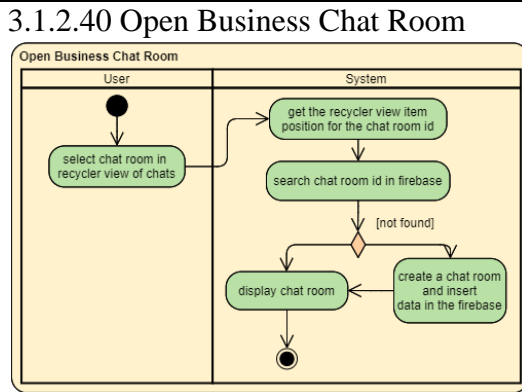


Figure 3.38 Open Business Chat Room

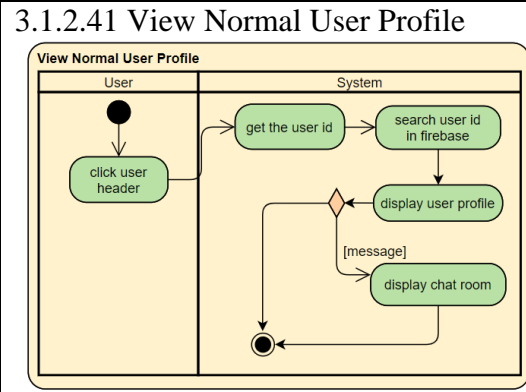


Figure 3.39 View Normal User Profile

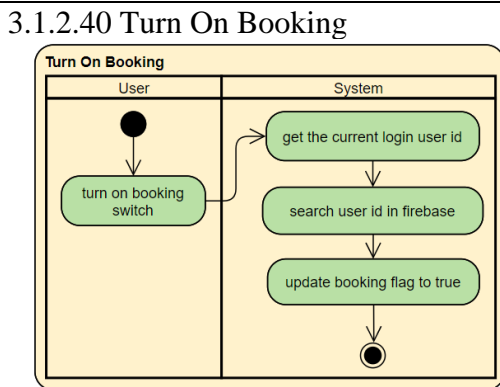


Figure 3.40 Turn On Booking

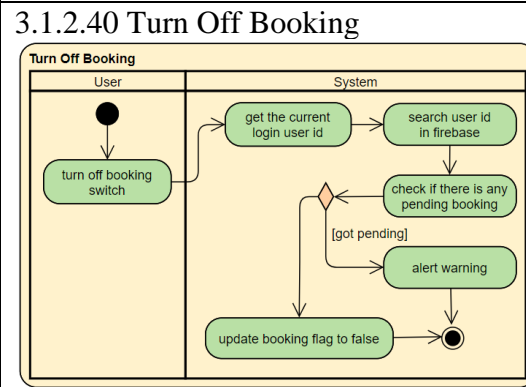


Figure 3.41 Turn Off Booking

3.1.3 User Interface (UI) Design

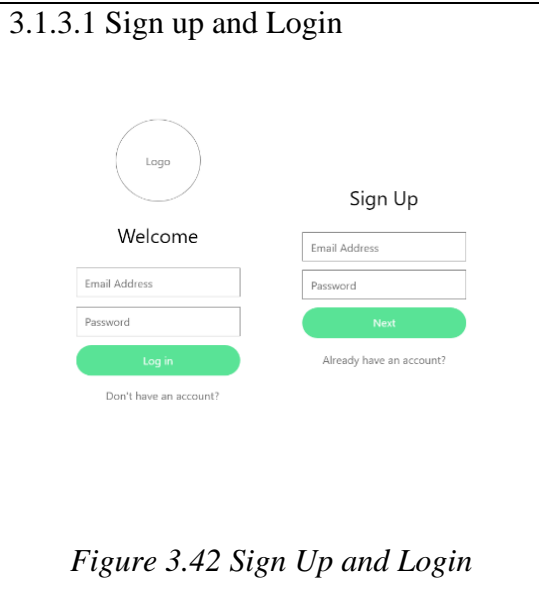


Figure 3.42 Sign Up and Login

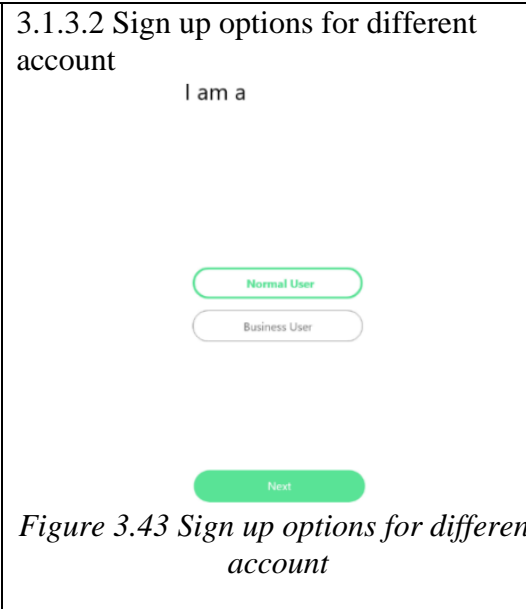


Figure 3.43 Sign up options for different account

3.1.3.3 Sign up for normal user

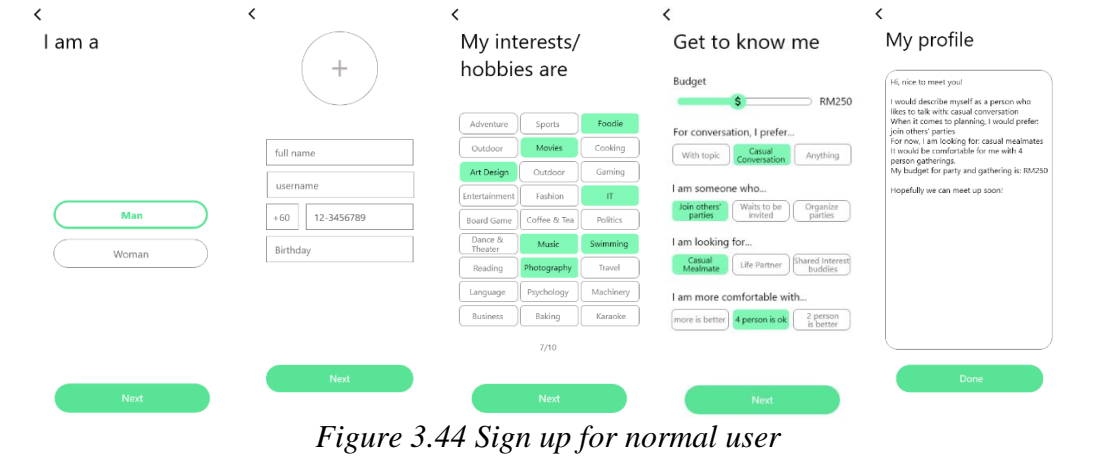


Figure 3.44 Sign up for normal user

3.1.3.4 Sign up for business user

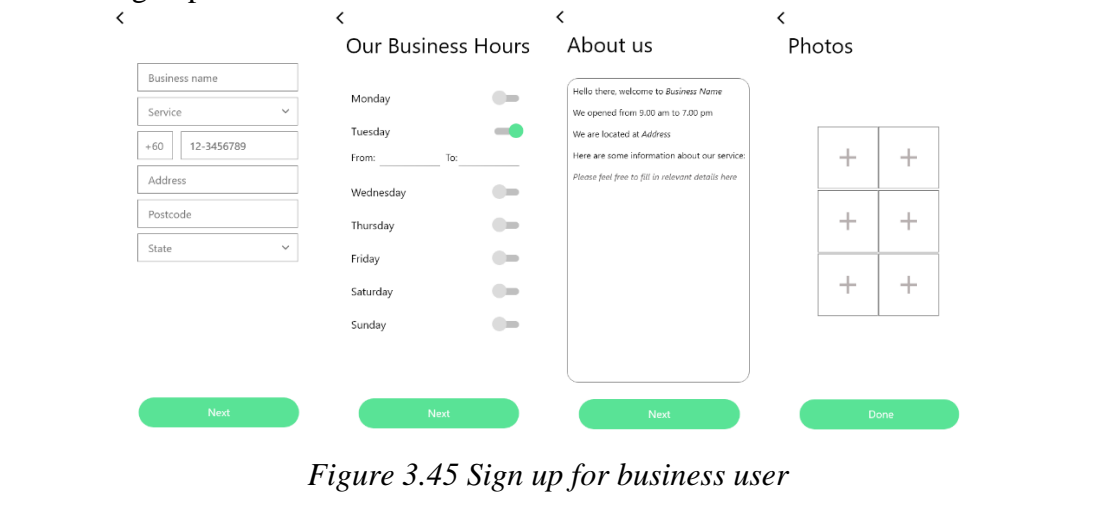


Figure 3.45 Sign up for business user

3.1.3.5 Home page of normal user

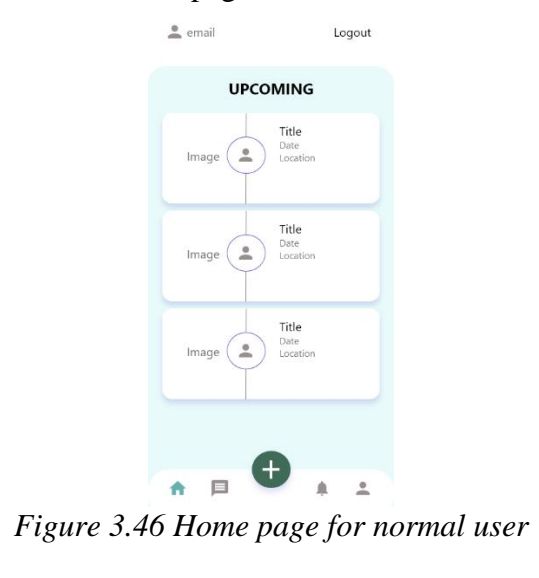


Figure 3.46 Home page for normal user

3.1.3.6 Notification Page of normal user

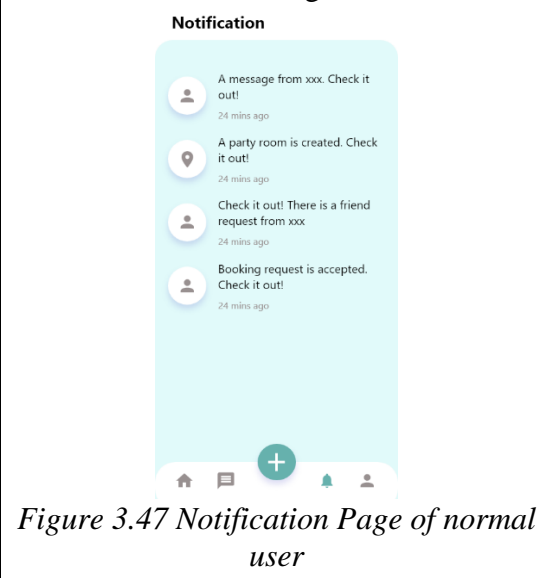


Figure 3.47 Notification Page of normal user

3.1.3.7 Chat page of normal user

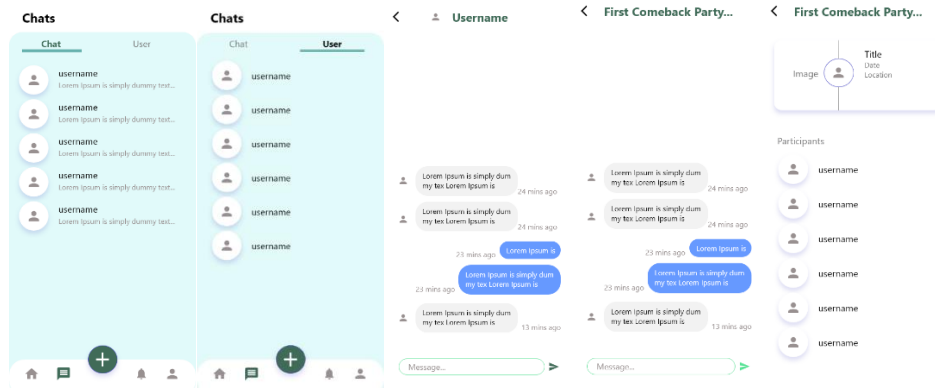


Figure 3.48 Chat page for normal user

3.1.3.8 Host Meet up

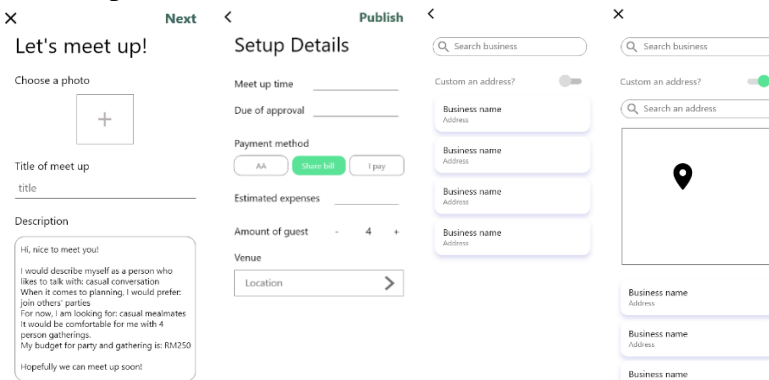


Figure 3.49 Host meet up

3.1.3.9 Profile Page of normal user (user's own profile)

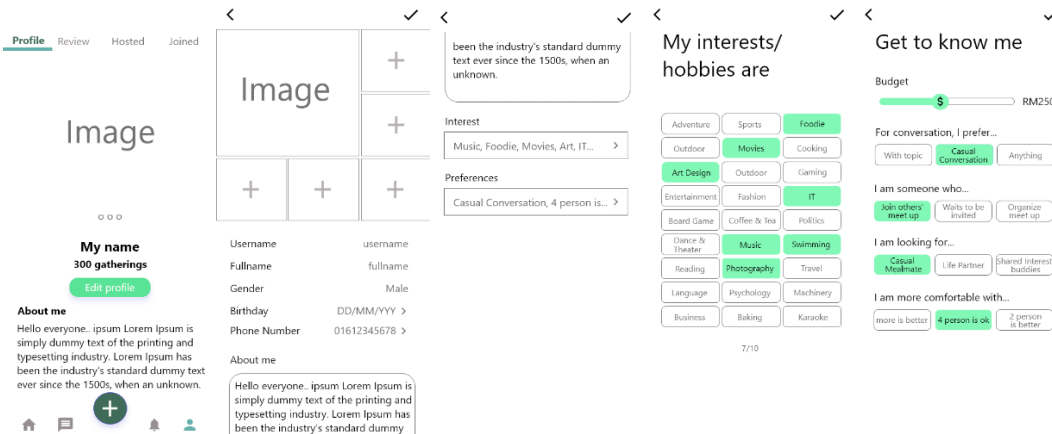
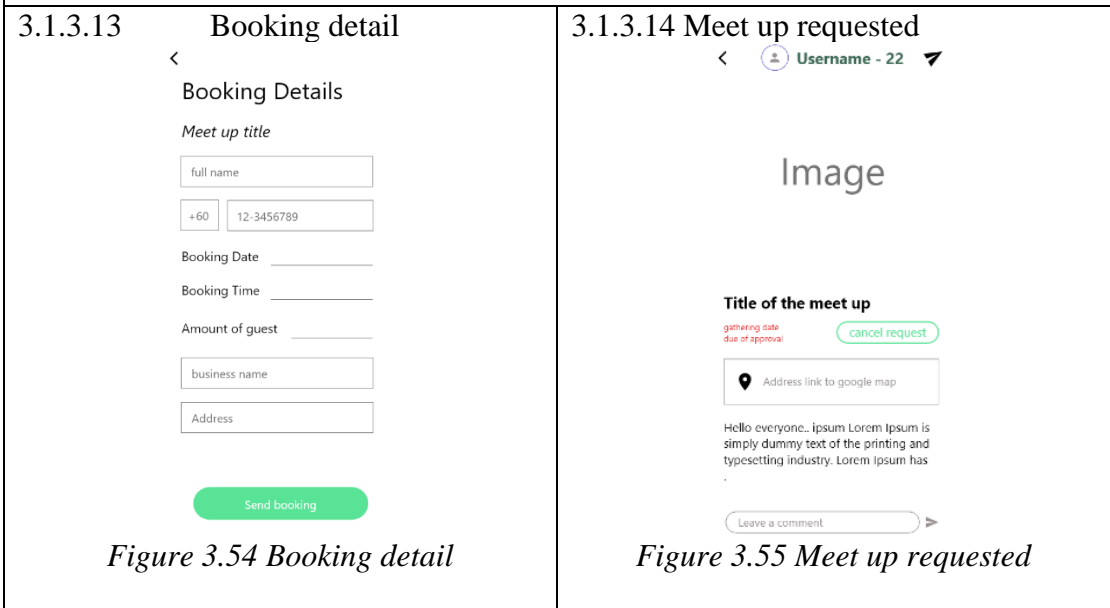
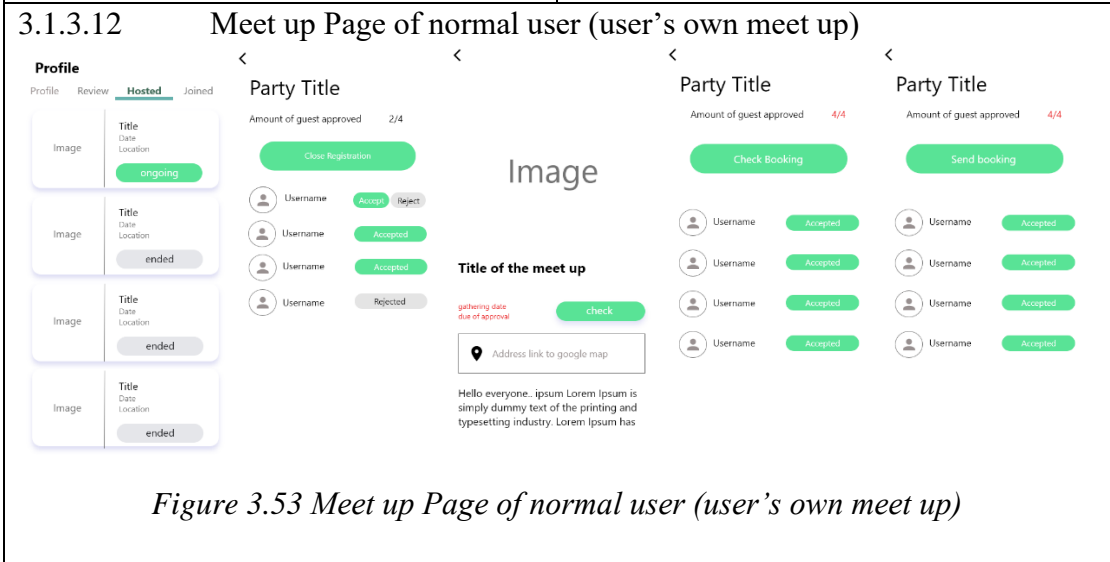
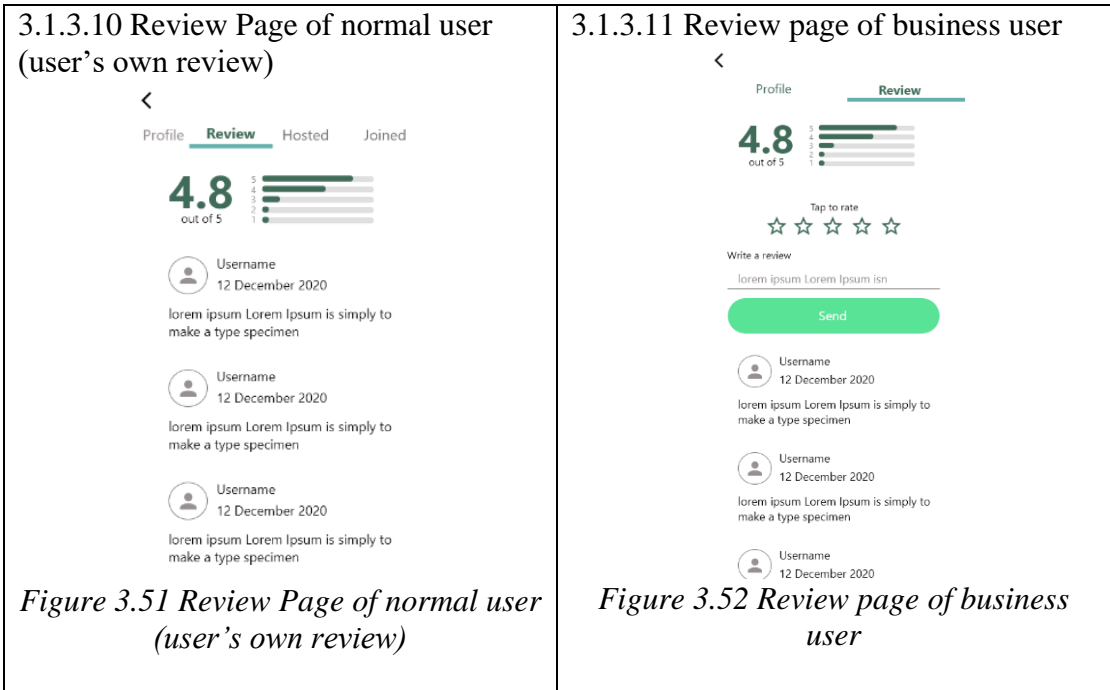


Figure 3.50 Profile Page of normal user (user's own profile)



3.1.3.15 Join Meet up and Meet up is accepted/ended/closed

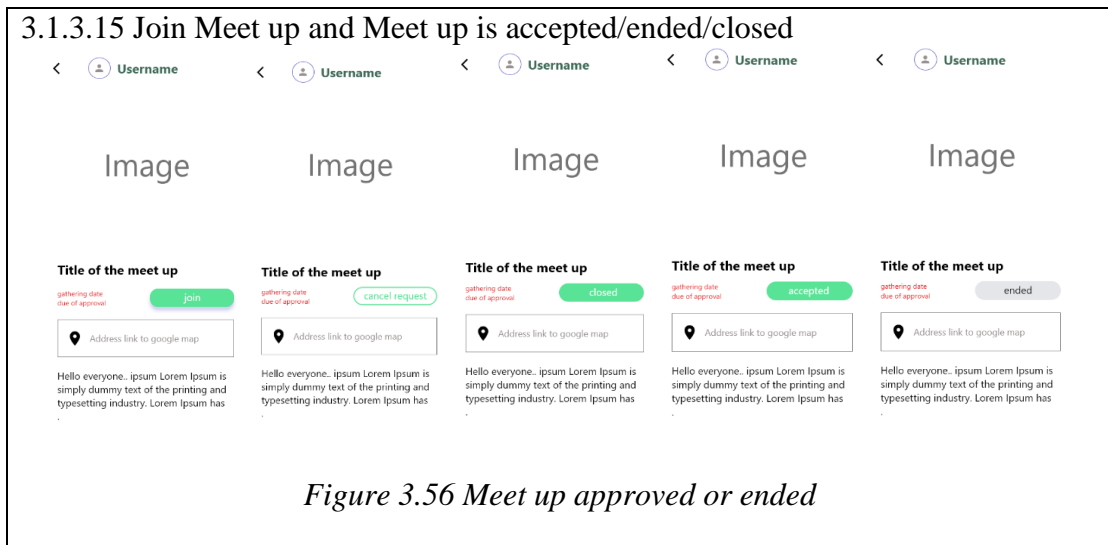


Figure 3.56 Meet up approved or ended

3.1.3.16 Profile Page of normal user (other users' profile)

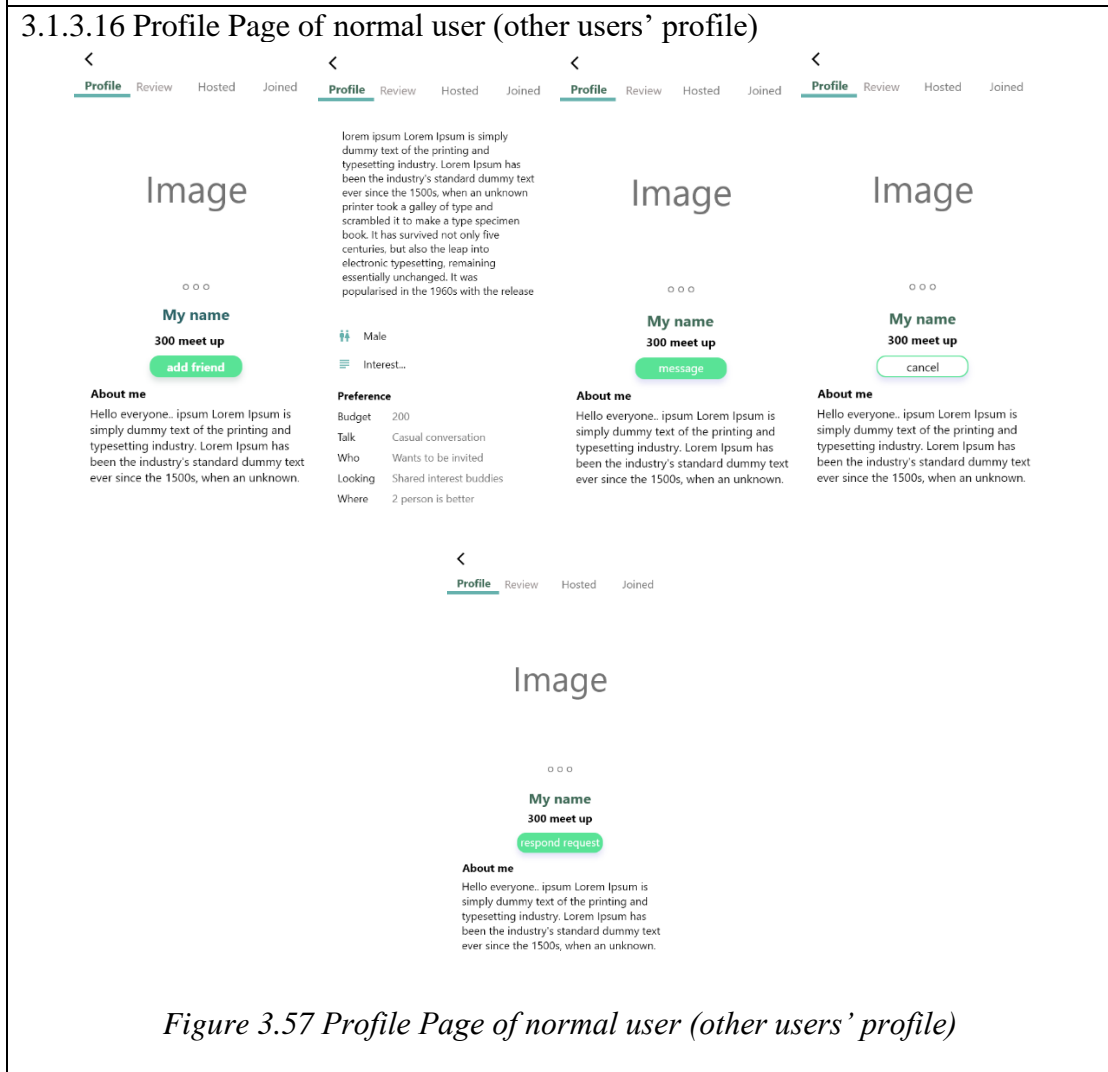


Figure 3.57 Profile Page of normal user (other users' profile)

3.1.3.17 Review Page of normal user (other users' review)

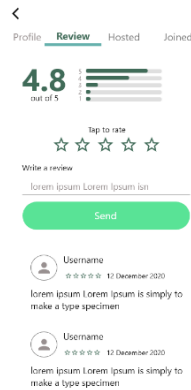


Figure 3.58 Review Page of normal user (other users' review)

3.1.3.18 Meet up Page of normal user (other users' meet up)

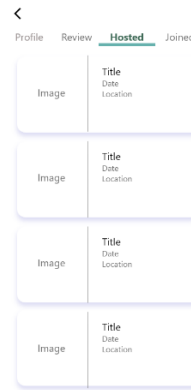


Figure 3.59 Meet up Page of normal user (other users' meet up)

3.1.3.19 Business profile in normal user's view

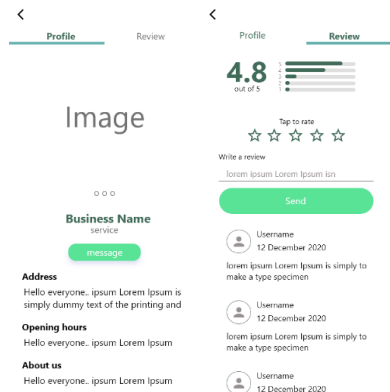


Figure 3.60 Business profile in normal user's view

3.1.3.20 Home page of business user

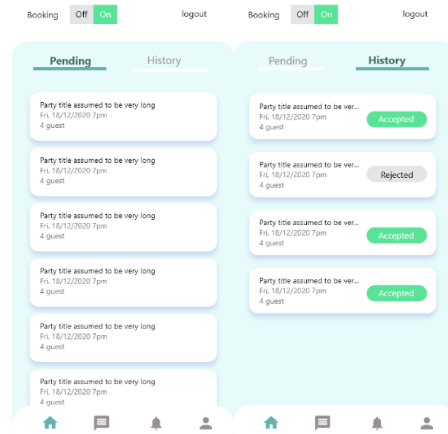


Figure 3.61 Home page of business user

3.1.3.21 Booking detail page of business user

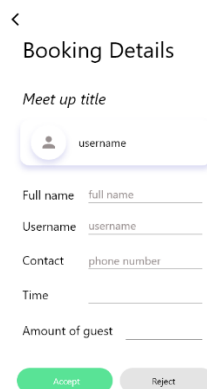


Figure 3.62 Booking detail page of business user

3.1.3.22 Notification page of business user

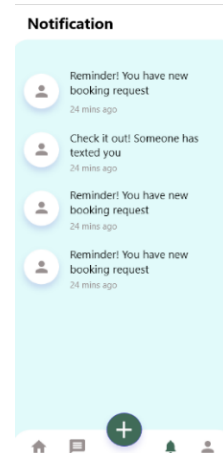


Figure 3.63 Notification page of business user

3.1.3.23 Chat page of business user

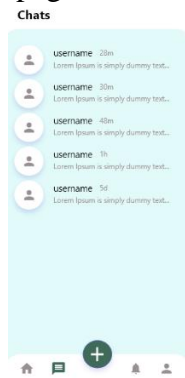


Figure 3.64 Chat page of business user

3.1.3.24 Profile page of business user

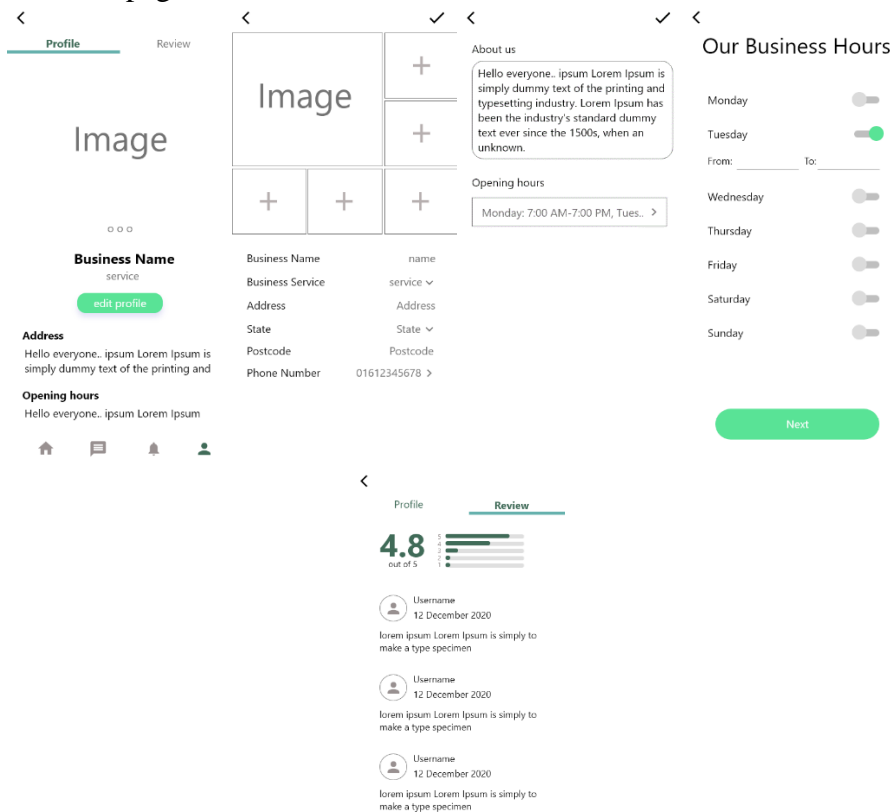


Figure 3.65 Profile page of business user

CHAPTER 4 SYSTEM DESIGN

4.1 System Block Diagram

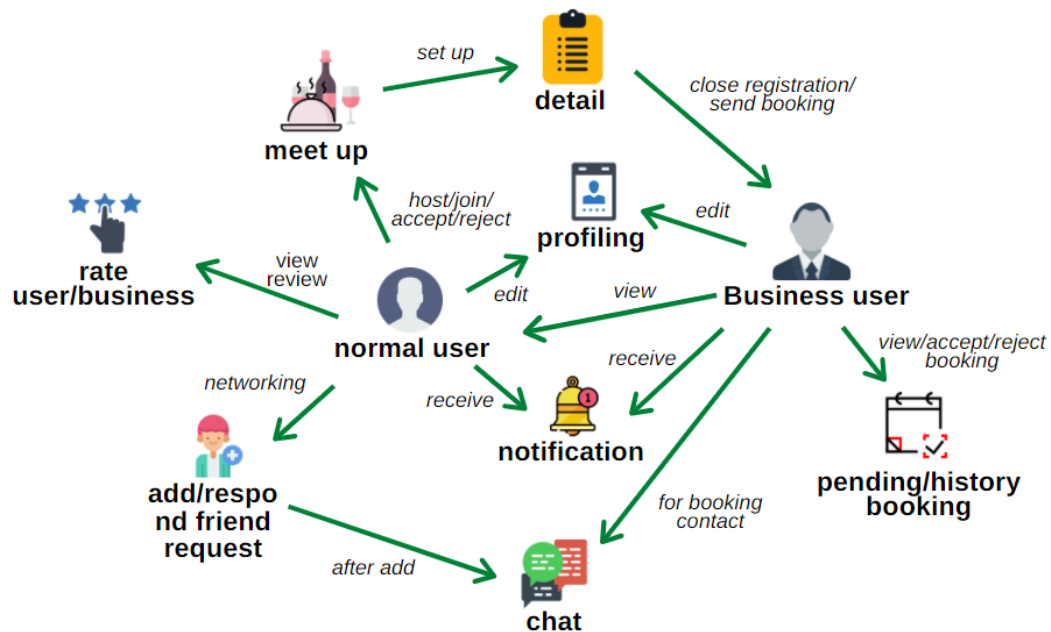


Figure 4.1 System Block Diagram

Normal user:

For meet up, user can host or join a meet up hosted by other normal users. There are some information needed to be setup such as photo, description, gathering date and time, approval date and time to accept or reject the application and meet up location. For meet up location, the users can either choose the business locations which are registered business user in the system, or they can custom a location by searching the address in the map. When all the information is setup, they can publish it and other users can start applying to join the meet up.

When the host checks the application of the meet up, they can close the registration whenever there is at least one accepted guest. If the location chosen is one of the registered business user in the system and booking is available, the host can send a booking request.

For networking, users can send friend request to each other and respond them by accepting or rejecting. After added as friend, they can chat via the chat room. They can also review and give rating to each other. They can also contact with business user for booking enquiry.

Users can edit their own profile such as adding photos and modifying their personal information.

CHAPTER 4 SYSTEM DESIGN

For notification, it will only happen in the following scenarios: chat message is received, booking is accepted, meet up registration is closed and a group chat room is created, friend request is received and friend request is responded.

Business Users:

For booking, they can view the history or pending approval booking in home page. Inside the booking detail page, they can either accept or reject the booking. They can also click on the host's header to view user profile. They can also chat with the host to communicate on the booking request.

They can also edit their business profile such as adding photos and modifying business information to enhance their business page.

For notification, it will only happen in the following scenarios: booking request is received and chat message is received.

4.2 System Components Specification

Classes and Objects Specification:

In this system, multiple java libraries and classes are utilized to build up the functionality of the application. For the most basic classes that required are Activity and Fragment. The lifecycle of the Activity and Fragment will control the pages and content switching of the application. For example, there are a couple of callback supported to monitor the state changes such as onCreate() – the system is creating, onStart() – the system is starting, onDestroy() – the system process is destroyed.

For fragment, there are some different extensions of the class such as BottomSheetDialogFragment. It is a fragment that provides a modal bottom sheet view on the application. For example, it is used in the function of responding friend request in this application where there will be a rectangular modal pop up at the bottom of the screen with some button options. Another kind of fragment will be AppCompatActivity which is a class that support pop up dialog on the screen. It is useful in allowing user to edit their phone number on the pop-up screen without directing them to another page. Another class that is related to fragment is PagerAdapter, which is a class that keep the fragments persistently as a set of tabs. For example, it is useful in allowing users to swipe right to view more profile photos.

Besides, another common class being used in this application is the RecyclerView. It mainly helps in generating a dynamic list of output after retrieving

the data from database. It is a view supporting scrolling function where users can scroll up and down to discover contents. RecyclerView works with ViewHolder – create a view for each item of data, and Adapter – binding the data to the view. These three classes work together to provide the functionality of viewing contents in a scrollable screen.

Another common class in this application is ViewModel. It is a class that is dedicated to hold and store the data controlling by a lifecycle. For example, data that hold by ViewModel will not be loss once users rotated the screen. Another example will be the data will not be lost when users trying to go back and forth of a series of process. This is particularly useful in the application's signing up process, where users might be going back to the previous page to amend the sign-up information, then move forward again without loss of data.

Database, Cloud Storage, and Messaging Specification:

The database used across the application is Firebase Realtime Database. It stores the users' data in JSON format and provide real-time data retrieving services. The structure of the database is as shown below:

```
{
  "Tokens" : {
    "userID" : {
      "token" : "value"
    }
  },
  "booking" : {
    "bookingID" : {
      "booking_datetime" : "value",
      "booking_id" : "value",
      "business_id" : "value",
      "customer_contact" : "value",
      "customer_id" : "value",
      "customer_name" : "value",
      "guest_amount" : "value",
      "party_id" : "value",
      "status" : "value"
    }
  }
}
```

CHAPTER 4 SYSTEM DESIGN

```
    }  
  },  
  "chatMessage" : {  
    "roomID" : {  
      "chatMessageID" : {  
        "message" : "value",  
        "messageDateTime" : "value",  
        "messageID" : "value",  
        "senderID" : "value"  
      }  
    }  
  },  
  "friends" : {  
    "userID" : {  
      "friendID" : {  
        "status" : "value",  
        "user_id" : "value"  
      }  
    }  
  },  
  "notification" : {  
    "userID" : {  
      "notificationID" : {  
        "data" : {  
          "body" : "value",  
          "icon" : "value",  
          "sent" : "value",  
          "title" : "value",  
          "type" : "value",  
          "user" : "value",  
          "misc" : "value"  
        },  
        "datetime" : "value",  
        "key" : "value"  
      }  
    }  
  }  
}
```

CHAPTER 4 SYSTEM DESIGN

```
    }
  }
},
"party" : {
  "partyID" : {
    "party_approvalDate" : "value",
    "party_description" : "value",
    "party_expenses" : "value",
    "party_gatheringDate" : "value",
    "party_hostUser" : "value",
    "party_id" : "value",
    "party_image" : "value",
    "party_joinedUser" : {
      "partyJoinedUserID" : {
        "status" : "value",
        "user_id" : "value"
      }
    },
    "party_location" : "value",
    "party_numGuest" : "value",
    "party_paymentMethod" : "value",
    "party_title" : "value"
  }
},
"party_location" : {
  "partyLocationID" : {
    "businessID" : "value",
    "partyID" : "value",
    "partyLocationID" : "value"
  }
},
"reviews" : {
  "userID" : {
    "reviewID" : {
```

CHAPTER 4 SYSTEM DESIGN

```
"message" : "value",
"rating" : "value",
"reviewDateTime" : "value",
"reviewID" : "value",
"userID" : "value"
}
}
},
"userChat" : {
  "userID" : {
    "roomID" : {
      "member" : {
        "userID" : "value",
        "userID" : "value"
      },
      "message" : "value",
      "roomid" : "value"
    },
    "roomID" : {
      "member" : {
        "userID" : "value",
        "userID" : "value"
      },
      "message" : "value",
      "partyid" : "value",
      "roomid" : "value"
    }
  }
},
"users" : {
  "business" : {
    "userID" : {
      "bu_address" : "value",
      "bu_booking" : "value",
```

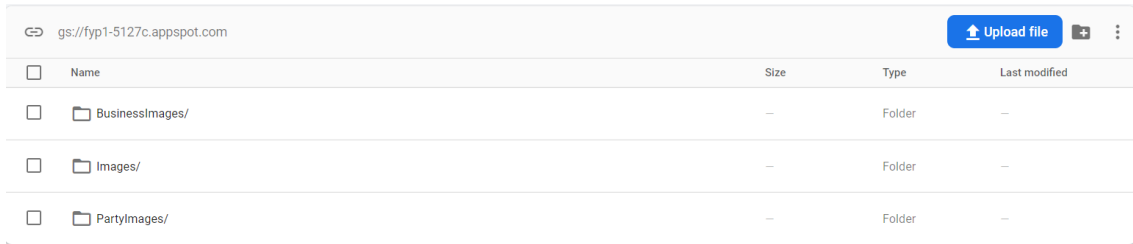
CHAPTER 4 SYSTEM DESIGN

```
"bu_businessName" : "value",
"bu_description" : "value",
"bu_email" : "value",
"bu_hour" : {
  "value" : "value",
  "value" : "value",
  "value" : "value",
  "value" : "value",
  "value" : "value"
},
"bu_imgUri" : {
  "imageID" : {
    "imgUri" : "value",
    "key" : "value"
  }
},
"bu_password" : "value",
"bu_phoneNum" : "value",
"bu_postcode" : "value",
"bu_service" : "value",
"bu_state" : "value",
"bu_userid" : "value"
},
"normal" : {
  "userID" : {
    "nu_birth" : "value",
    "nu_budget" : "value",
    "nu_description" : "value",
    "nu_email" : "value",
    "nu_fullname" : "value",
    "nu_gender" : "value",
    "nu_imgUri" : {
      "imageID" : {
```

CHAPTER 4 SYSTEM DESIGN

```
    "imgUri" : "value",
    "key" : "value"
  }
},
"nu_interest" : [ "value", "value", "value", "value" ],
"nu_partyCreated" : {
  "partyCreatedID" : {
    "party_id" : "value"
  }
},
"nu_partyJoined" : {
  "partyJoinedID" : {
    "party_id" : "value",
    "status" : "value"
  }
},
"nu_password" : "value",
"nu_phoneNum" : "value",
"nu_preference" : [ "value", "value", "value", "value" ],
"nu_userid" : "value",
"nu_username" : "value"
}
}
}
}
```

For cloud storage, Firebase Cloud Storage is used as it supports user to upload and download file from a cloud-based storage which supported together with the real-time database services. The cloud storage structure is as shown below where the images uploaded by the business user will be stored to BusinessImages/, normal user will be stored to Images/ and meet up images will be stored to PartyImages/.



<input type="checkbox"/>	Name	Size	Type	Last modified
<input type="checkbox"/>	BusinessImages/	–	Folder	–
<input type="checkbox"/>	Images/	–	Folder	–
<input type="checkbox"/>	PartyImages/	–	Folder	–

Figure 4.2 Firebase Cloud Storage Structure

For the messaging service that enable users to send and receive notification, Firebase Cloud Messaging is used in this application. There are few classes needed to build up the entire service. An instance class that extends the `FirebaseMessagingService` class is needed with the override methods such as `onNewToken()` – create a token to grant the service and `onMessageReceived()` – controls the behaviour when the message is received. Another self-defined method could be `sendNotification()` – to send out the message. Other classes such as `Client` – contains `getClient()` method for the messaging service to get the current client, `DataMessage` class – a model for the message, `FCMToken` class that contains `getToken()` – to get the token for service granted, `Response` class – indicating the success or fail of the service, `Sender` class – to create an instance of the message sender. There is an interface needed `APIServiceFCM` – contains a header and HTTP POST method to send notification over the internet.

User Interface Specification:

The types of layout that are commonly used across the application are `LinearLayout`, `ConstraintLayout`, and `RelativeLayout`. These three layouts provide different flexibility to arrange different View class in the application. For `LinearLayout`, it is mostly used to arrange and group the relevant views in horizontal or vertical layout. `ConstraintLayout` is also frequently used as the base layout of the screen as it allows to position the children view in flexible way by setting the constraint weight. While `RelativeLayout` is used when there is a need to display the children view in a relative way such as overlapping views. All these layouts are often used together. For example, in sign up page, there is a base layout using `ConstraintLayout`, then the children view using `LinearLayout` to organize multiple views in vertical orientation to ask for user's information. An example for `RelativeLayout` will be displaying the user image by overlapping it with another image view.

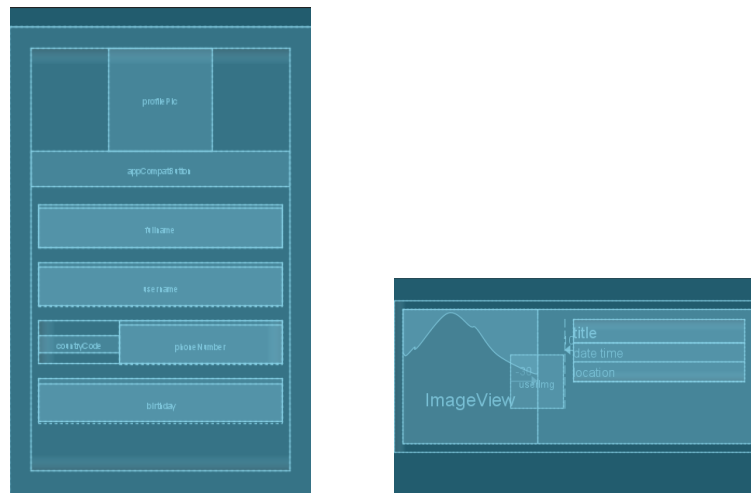


Figure 4.3 The usage of different layouts

For the design of bottom navigation, a view called BottomNavigationView is used to populate the menu items for users to switch to different tab of pages. For example, there are 4 pages – Home, Chat, Notification and Profile that can be switched by users at the bottom of the screen, which is hold by this BottomNavigationView. The options of items are populated by a menu – a XML file that specify all the menu items.

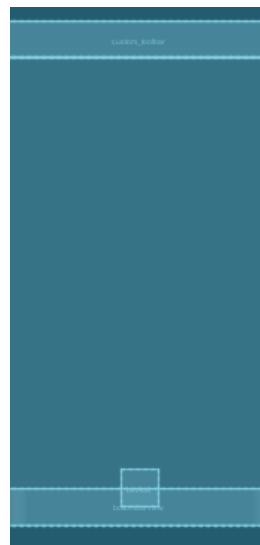


Figure 4.4 BottomNavigationView in home screen

There is a header used across the application mainly for back-pressed function and displaying the title of the pages. It is a customized Toolbar view that contains all the necessary view that shows the title of the pages. Whenever users land in a particular page, the Toolbar will handle which title view to show on the top of the page. Sometimes it is also useful for displaying a back-pressed button at the left of the Toolbar to let user navigates to previous page.

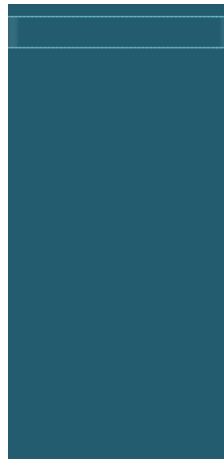


Figure 4.5 Toolbar on top of screen

4.3 Components Design

System Flow Chart Diagram:

a) Normal User Module

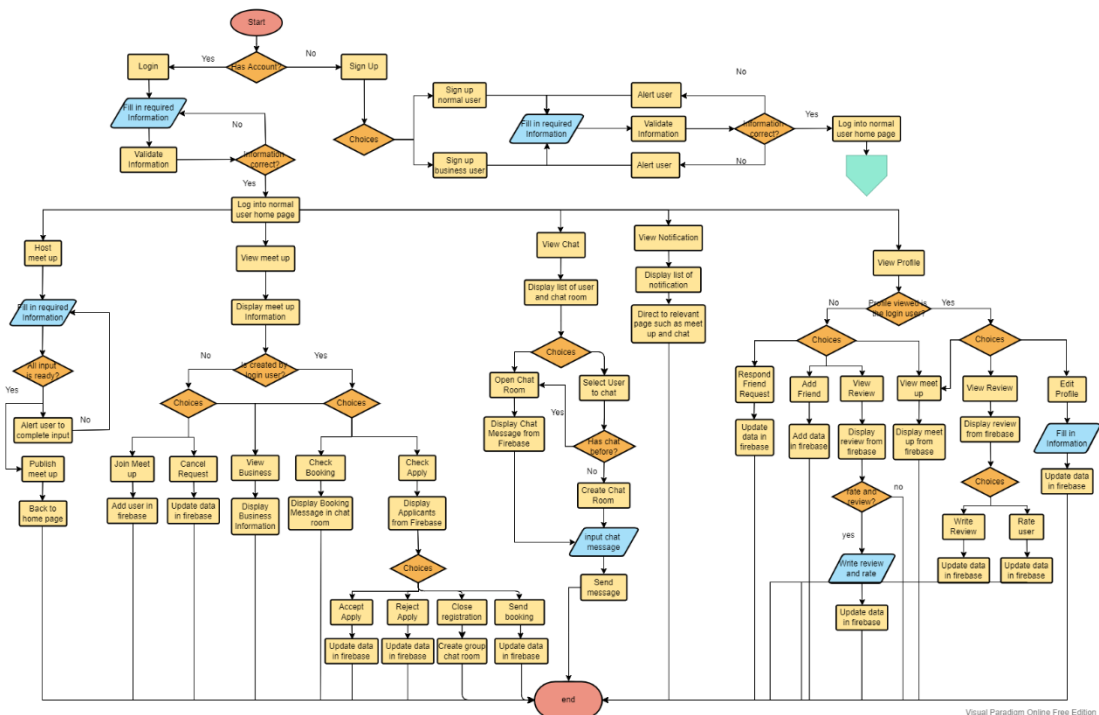


Figure 4.6 Normal User Flow Chart Diagram

b) Business user module

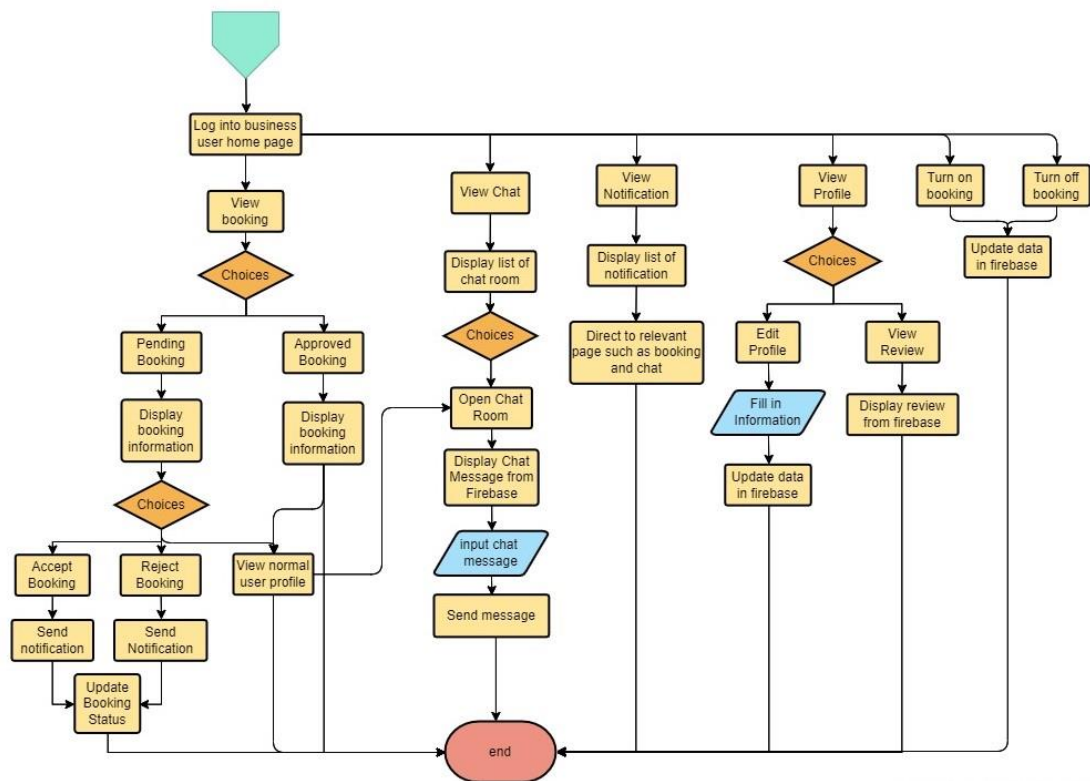


Figure 4.7 Business User Flow Chart Diagram

User Interface Navigation Design:

There is an important component that is designed to control the flow of multiple related fragments. It is a Navigation components supported by Android that allows user to navigate across different fragments in the application. For example, since the sign-up process is split into multiple pages, these pages can be group together with the help of Navigation Graph that helps navigating one to another. A Navigation Graph is an XML file that contains relevant fragments that has a navigating destination. As shown in below picture, this is a Navigation Graph for sign up process, where a fragment can be navigated to another fragment as destination and going back or forth.

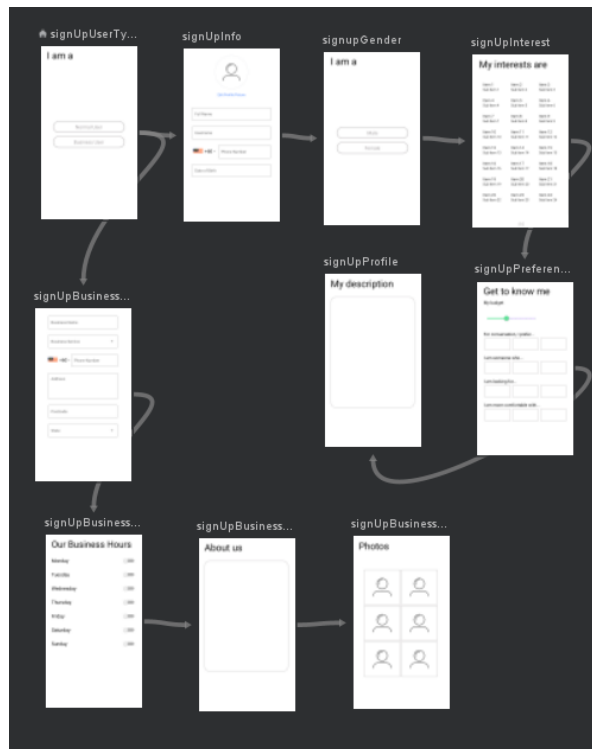


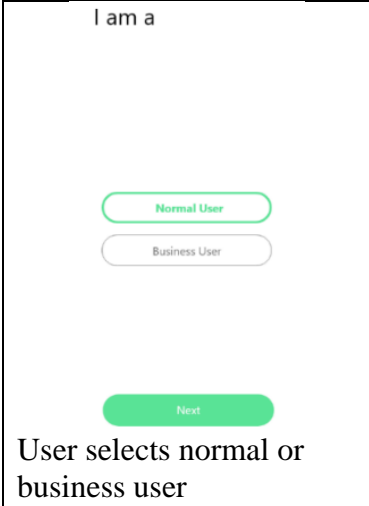
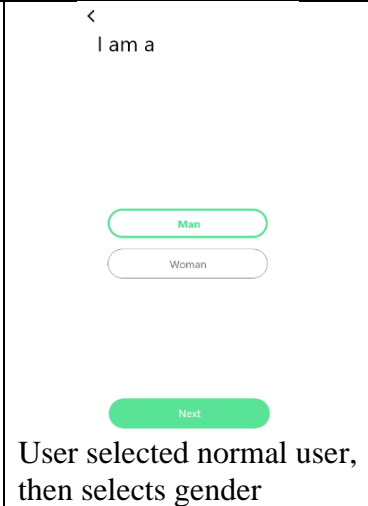
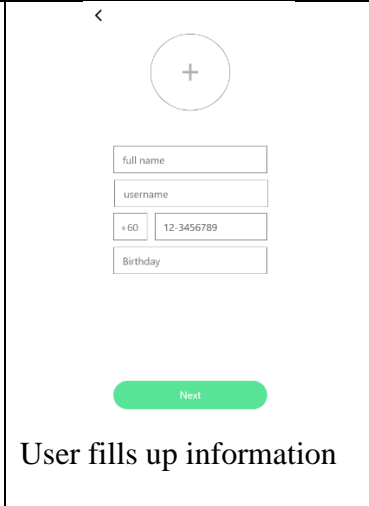
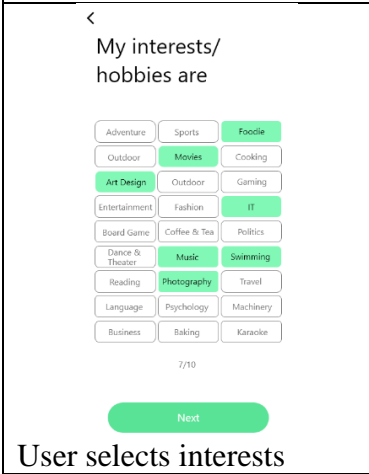
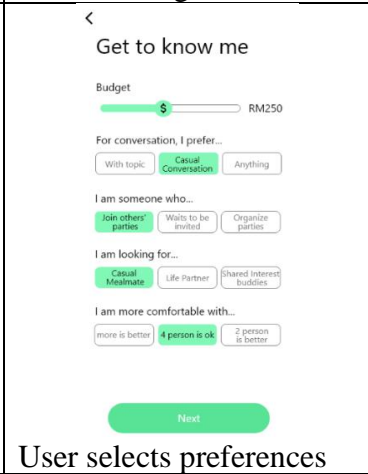
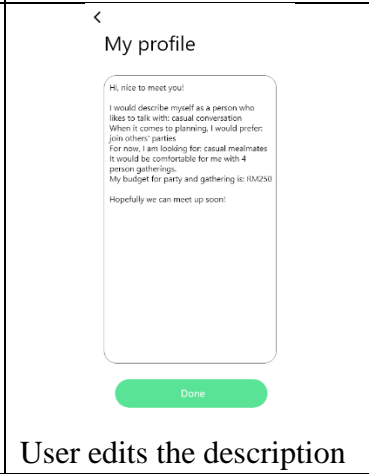
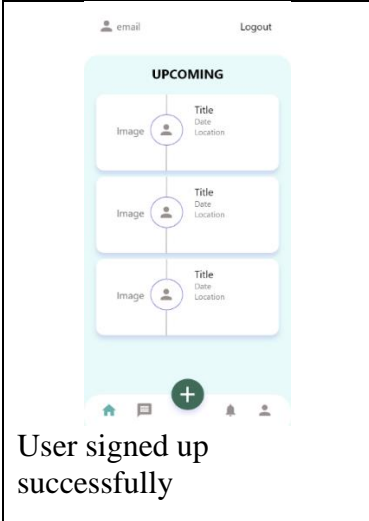
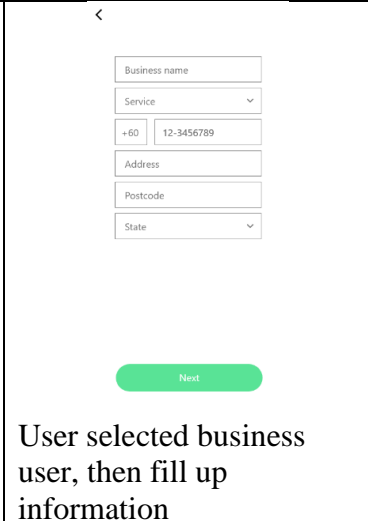
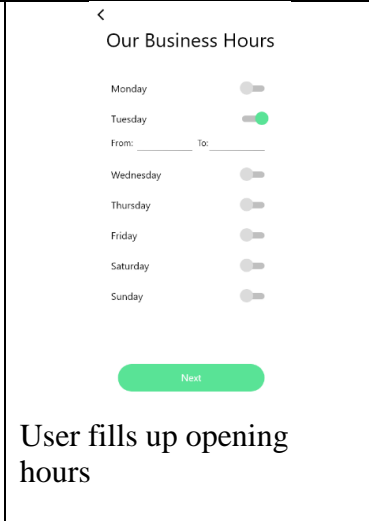
Figure 4.8 Navigation Graph

4.4 System Components Interaction Operation

Log in and sign up:

<p>User key in the input to sign in or choose to create new account</p>	<p>User logged in successfully.</p>	<p>User chose to create new account</p>
---	-------------------------------------	---

CHAPTER 4 SYSTEM DESIGN

 <p>I am a</p> <p>Normal User</p> <p>Business User</p> <p>Next</p>	 <p>I am a</p> <p>Man</p> <p>Woman</p> <p>Next</p>	 <p>+</p> <p>full name</p> <p>username</p> <p>+60 12-3456789</p> <p>Birthday</p> <p>Next</p>
 <p>My interests/ hobbies are</p> <p>Adventure Sports Foodie</p> <p>Outdoor Movies Cooking</p> <p>Art Design Outdoor Gaming</p> <p>Entertainment Fashion IT</p> <p>Board Game Coffee & Tea Politics</p> <p>Dance & Theater Music Swimming</p> <p>Reading Photography Travel</p> <p>Language Psychology Machinery</p> <p>Business Baking Karaoke</p> <p>7/10</p> <p>Next</p>	 <p>Get to know me</p> <p>Budget \$ RM250</p> <p>For conversation, I prefer...</p> <p>With topic: Casual Conversation Anything</p> <p>I am someone who...</p> <p>Join others' parties Wants to be invited Organize parties</p> <p>I am looking for...</p> <p>Casual Mealmate Life Partner Shared Interest Buddies</p> <p>I am more comfortable with...</p> <p>more is better 4 person is ok 3 person is better</p> <p>Next</p>	 <p>My profile</p> <p>Hi, nice to meet you!</p> <p>I would describe myself as a person who likes to talk with casual conversation. When it comes to planning, I would prefer: join others' parties. For now, I am looking for casual mealmates. It would be comfortable for me with 4 person gatherings. My budget for party and gathering is: RM250. Hopefully we can meet up soon!</p> <p>Done</p>
 <p>email Logout</p> <p>UPCOMING</p> <p>Title Date Location</p> <p>Image</p> <p>Title Date Location</p> <p>Image</p> <p>Title Date Location</p> <p>Image</p> <p>+</p>	 <p>Business name</p> <p>Service</p> <p>+60 12-3456789</p> <p>Address</p> <p>Postcode</p> <p>State</p> <p>Next</p>	 <p>Our Business Hours</p> <p>Monday</p> <p>Tuesday</p> <p>From: To:</p> <p>Wednesday</p> <p>Thursday</p> <p>Friday</p> <p>Saturday</p> <p>Sunday</p> <p>Next</p>

User selects normal or business user

User selected normal user, then selects gender

User fills up information

User selects interests

User selects preferences

User edits the description

User signed up successfully

User selected business user, then fill up information

User fills up opening hours

CHAPTER 4 SYSTEM DESIGN

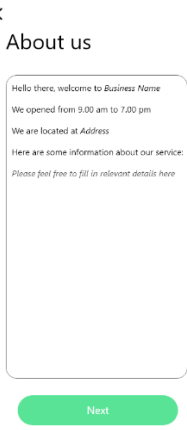
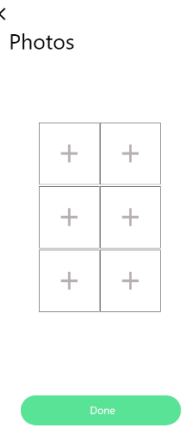
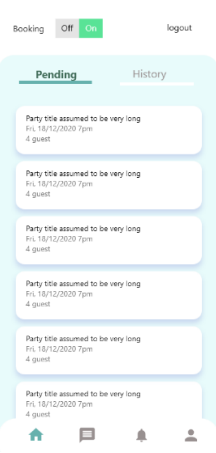
 <p>User edits description</p>	 <p>User uploads photos</p>	 <p>User signed up successfully</p>
---	--	--

Table 4.1 Log in and sign up

Host meet up:


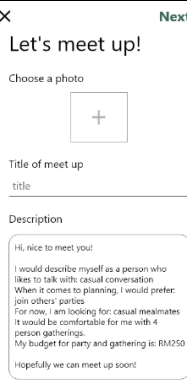
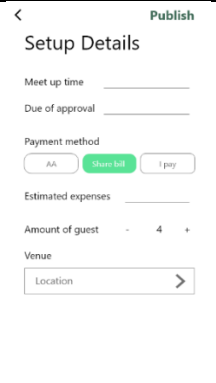
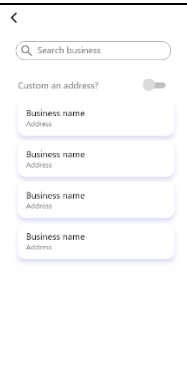


 <p>User clicks the “+” button</p>	 <p>User fills up information</p>	 <p>User fills up details</p>
 <p>User selects business location</p>	 <p>Users wants to customize a location from map</p>	 <p>User published the meet up and back to home</p>

Table 4.2 Host a meet up

Join a meet up:



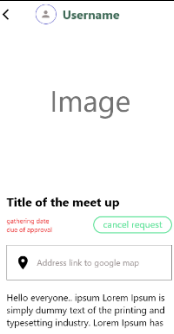
 <p>User selects a meet up in home screen</p>	 <p>User clicks join</p>	 <p>User can cancel request</p>
--	---	--

Table 4.3 Join a meet up

Meet up is accepted, ended or closed





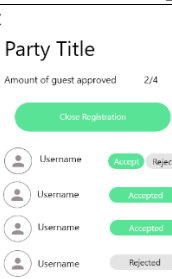
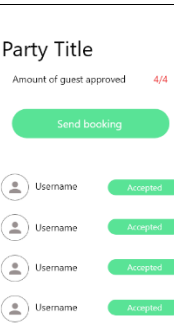
 <p>User checks meet up and found that it is closed</p>	 <p>User checks meet up and found that it is accepted</p>	 <p>User checks meet up and found that it is ended</p>
---	---	--

Table 4.4 Meet up is accepted, ended or closed

Host manages the meet up and visits business profile

 <p>User clicks check button</p>	 <p>User accepts or rejects the applications</p>	 <p>User clicks close registration button and the location is allowed for booking</p>
---	---	--

CHAPTER 4 SYSTEM DESIGN

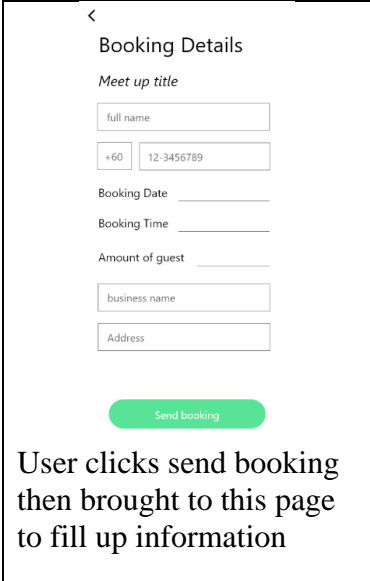
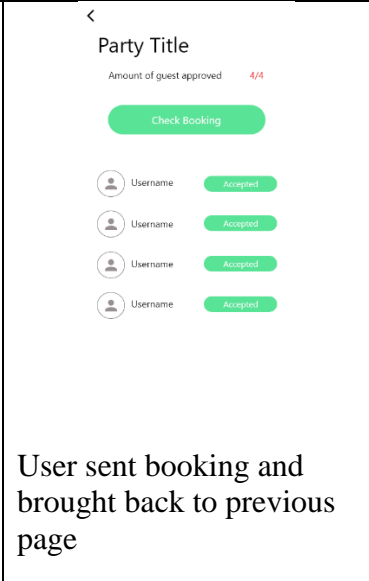
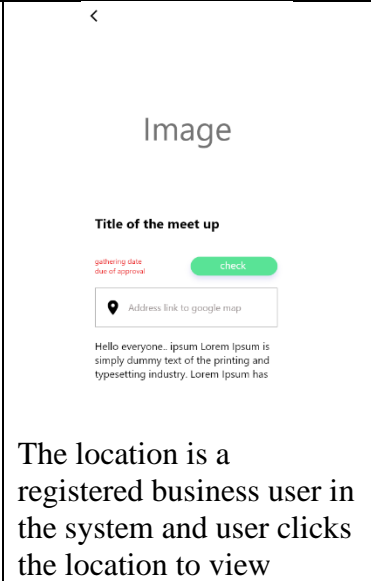
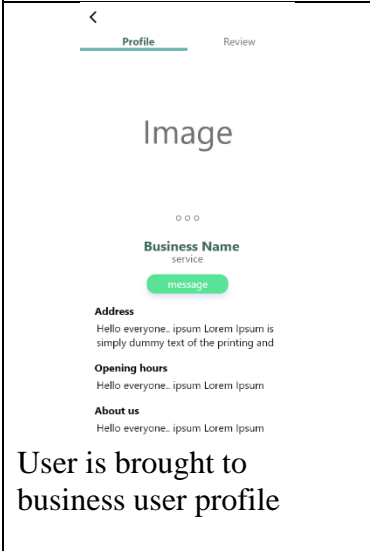
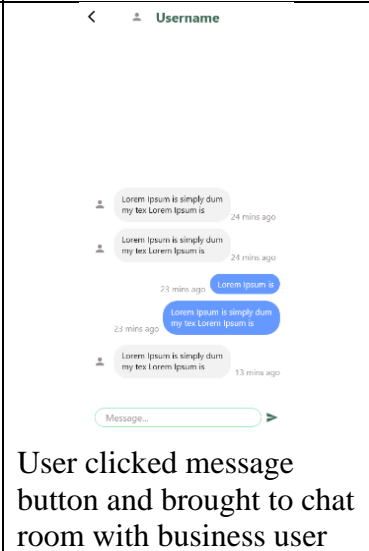
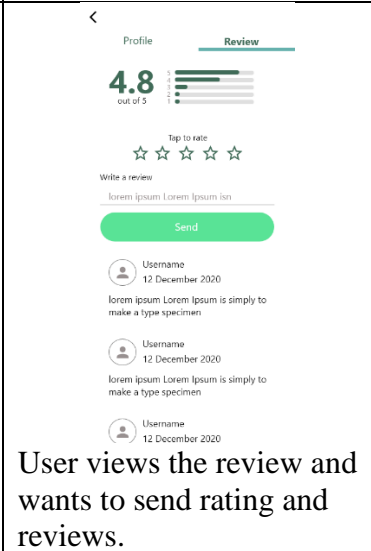
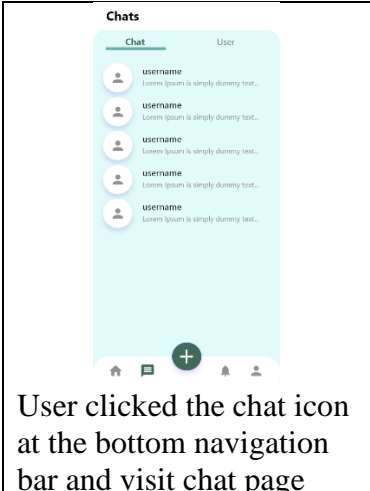
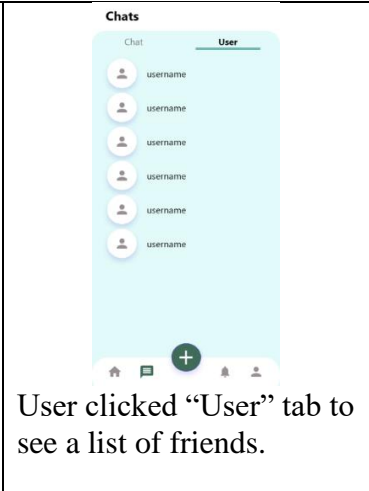
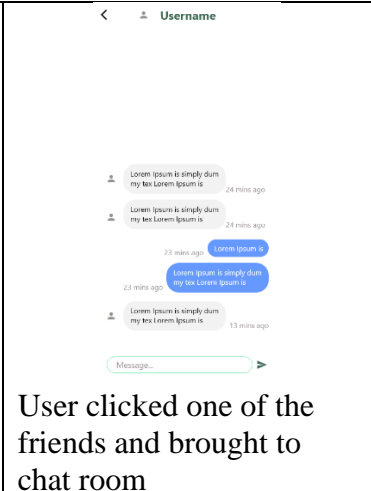
 <p>User clicks send booking then brought to this page to fill up information</p>	 <p>User sent booking and brought back to previous page</p>	 <p>The location is a registered business user in the system and user clicks the location to view</p>
 <p>User is brought to business user profile</p>	 <p>User clicked message button and brought to chat room with business user</p>	 <p>User views the review and wants to send rating and reviews.</p>

Table 4.5 Host manages the meet up and visits business profile

User views chat page

 <p>User clicked the chat icon at the bottom navigation bar and visit chat page</p>	 <p>User clicked "User" tab to see a list of friends.</p>	 <p>User clicked one of the friends and brought to chat room</p>
--	---	---

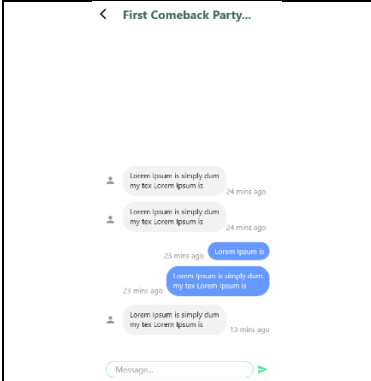
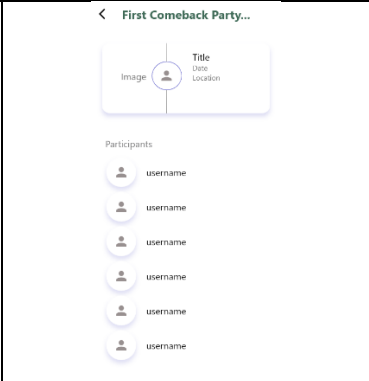
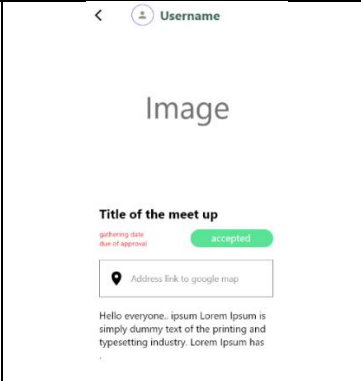
 <p>User clicked a meet up group chat and brought to the room</p>	 <p>User clicked the header and view the meet up information and participants</p>	 <p>User clicked the meet up card and visit the meet up page</p>
--	---	---

Table 4.6 User views chat page

User view notification

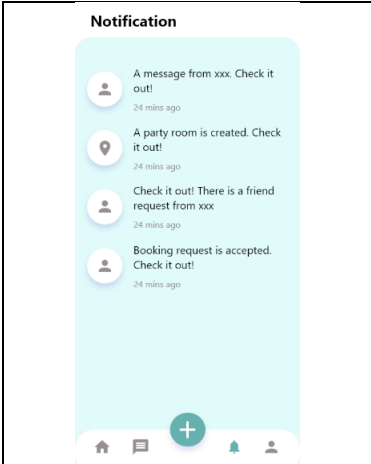
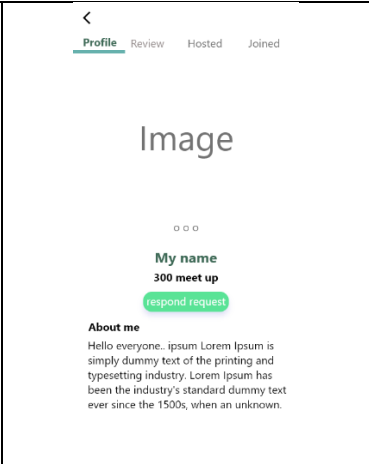
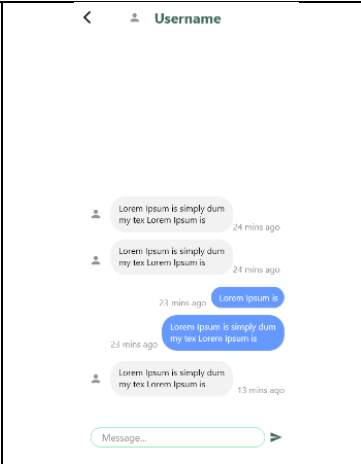
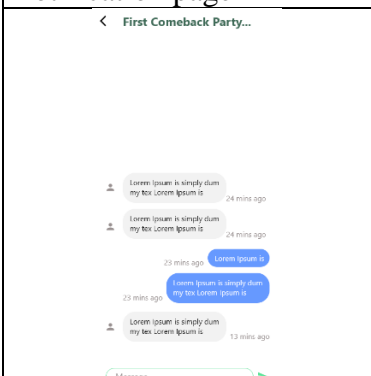
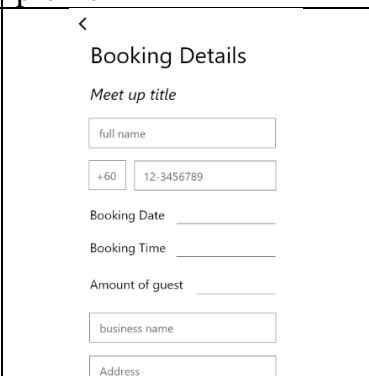
 <p>User clicked notification icon at the bottom navigation bar and visit notification page</p>	 <p>User clicked on friend request notification and visit the requested user profile</p>	 <p>User clicked on chat notification and visit chat room</p>
 <p>User clicked on meet up ready notification and visit group chat room</p>	 <p>User clicked booking response notification and visit booking detail page</p>	

Table 4.7 User view notification

Normal user view own user profile

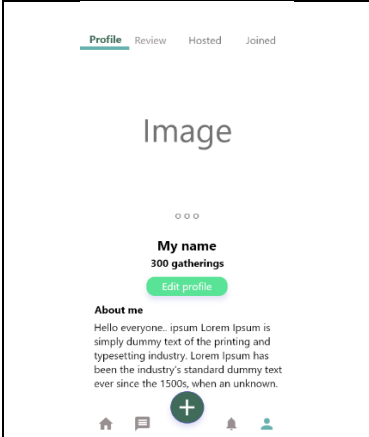
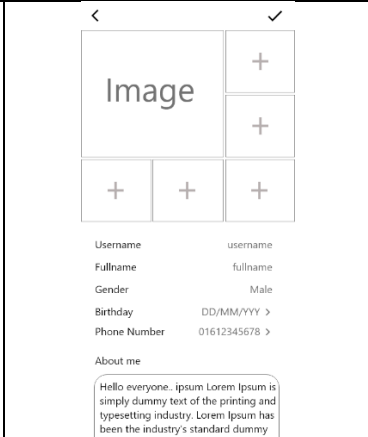

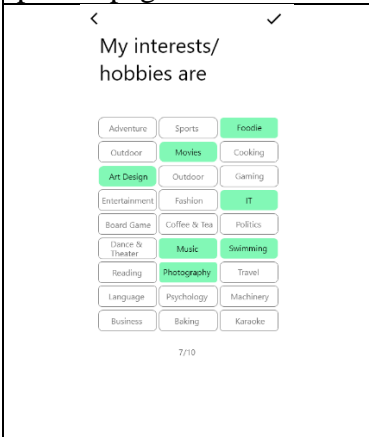
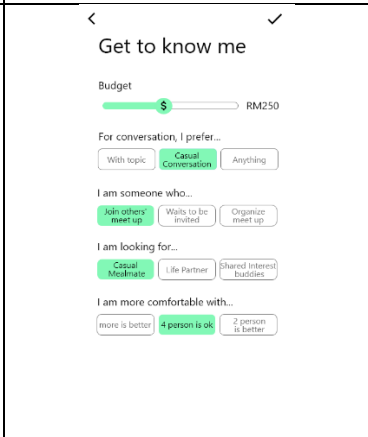
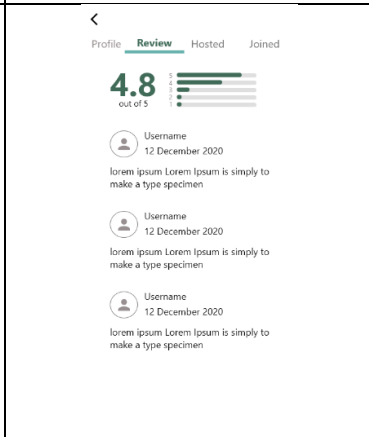
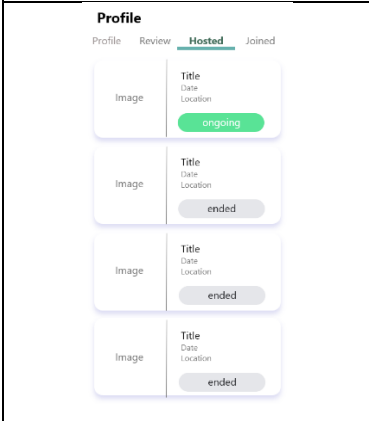
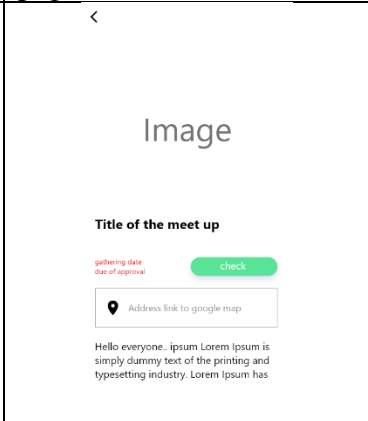
 <p>User clicked profile icon at the bottom navigation bar and visit own user profile page</p>	 <p>User clicked edit button and visit edit page</p>	 <p>User can also edit interest or preferences</p>
 <p>User clicked interest and visit edit interest page</p>	 <p>User clicked preferences and visit edit preference page</p>	 <p>User clicked Review tab</p>
 <p>User clicked Hosted or Joined tab</p>	 <p>User clicked any meet up and visit meet up information page</p>	

Table 4.8 Normal user view own user profile

Normal user view other user profile:

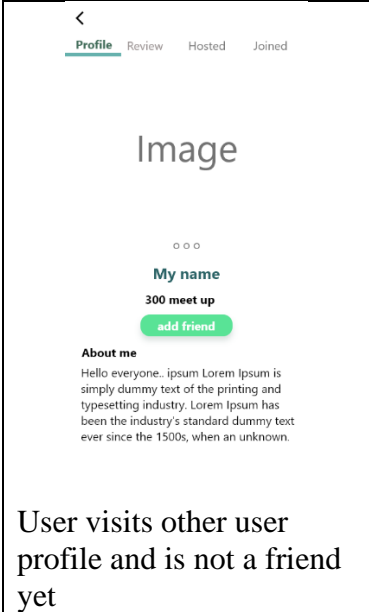
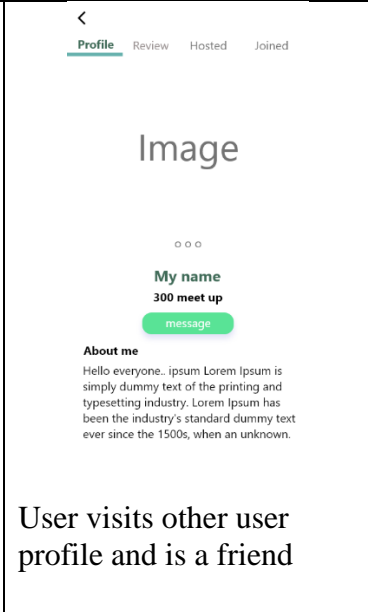
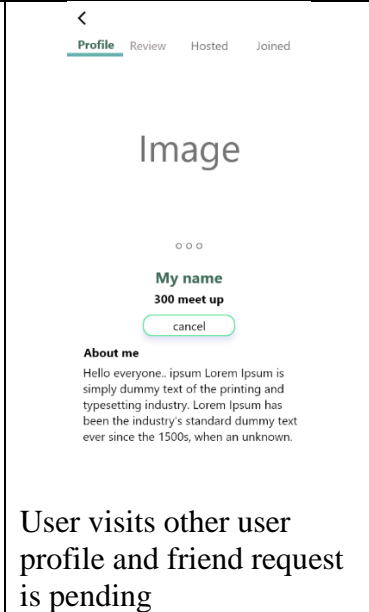
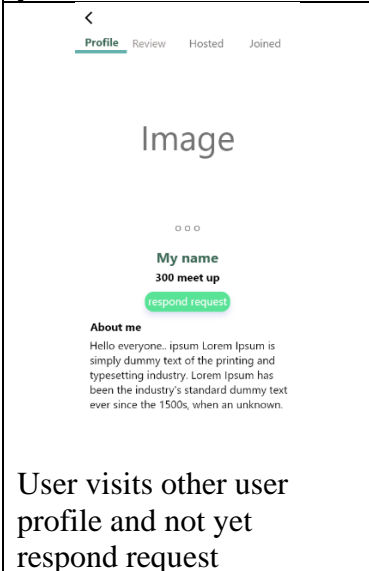
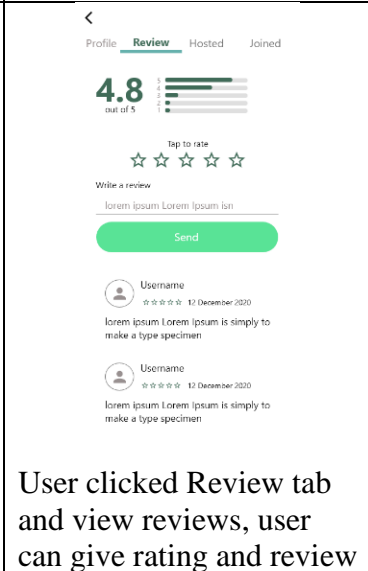

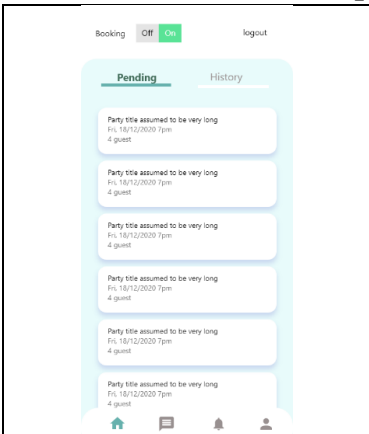
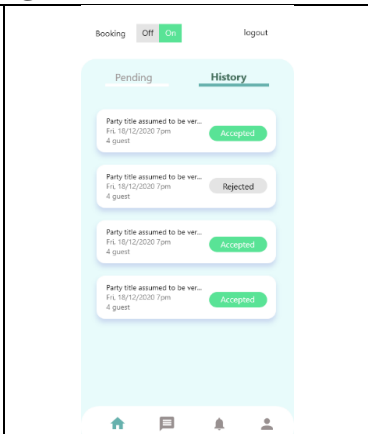
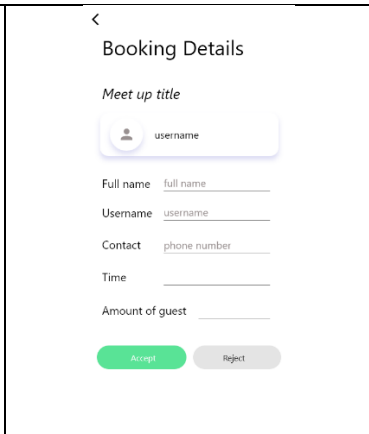
 <p>User visits other user profile and is not a friend yet</p>	 <p>User visits other user profile and is a friend</p>	 <p>User visits other user profile and friend request is pending</p>
 <p>User visits other user profile and not yet respond request</p>	 <p>User clicked Review tab and view reviews, user can give rating and review</p>	 <p>User clicked Hosted or Joined tab, user cannot view the meet up detail</p>

Table 4.9 Normal user view other user profile

Business user visits home page

		
---	--	---

CHAPTER 4 SYSTEM DESIGN

<p>User can switch on or off for “booking”. User needs to accept or reject all the pending booking before switching off</p>	<p>User clicked History tab</p>	<p>User clicked any of the booking in home page and visit booking detail page</p>
---	---------------------------------	---

Table 4.10 Business user visits home page

Business user visits notification page

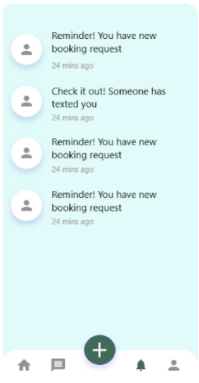


 <p>User clicked notification icon at the bottom navigation bar and visits notification page</p>	 <p>User clicked booking request notification and visits booking detail page</p>	 <p>User clicked on chat notification and visit chat room</p>
---	---	--

Table 4.11 Business user visits notification page

Business user visits chat page

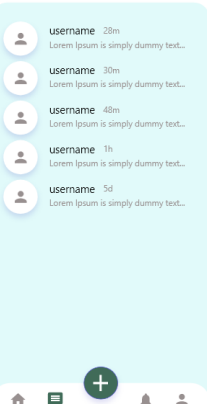
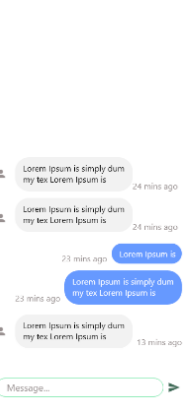
 <p>User clicked chat icon at the bottom navigation bar and visits chat page</p>	 <p>User clicked on any of the chats and visit chat room</p>	
---	---	--

Table 4.12 Business user visits chat page

Business user visits own user profile

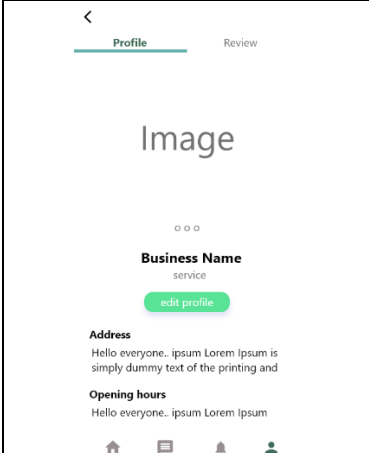
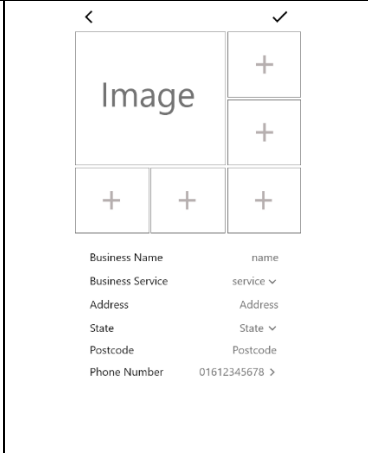
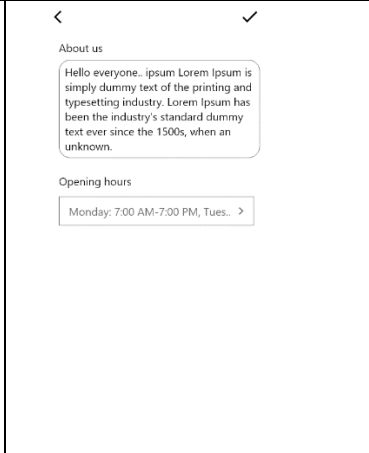
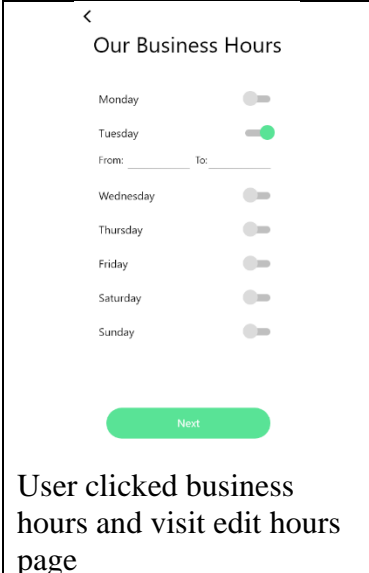
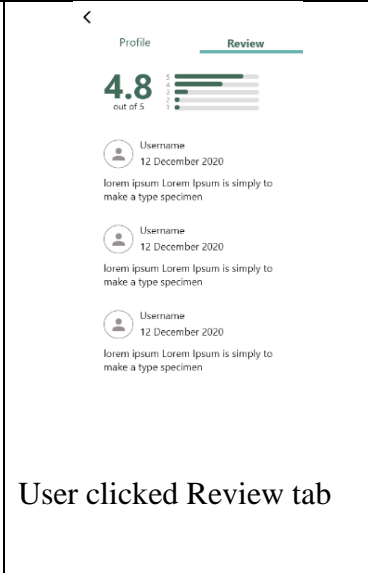
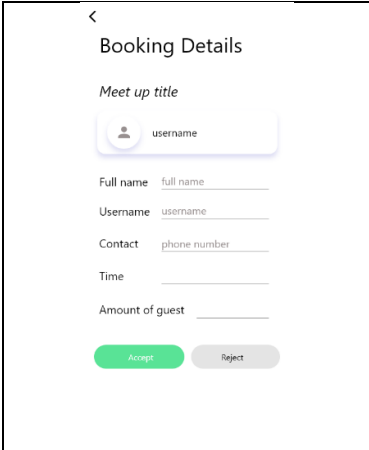
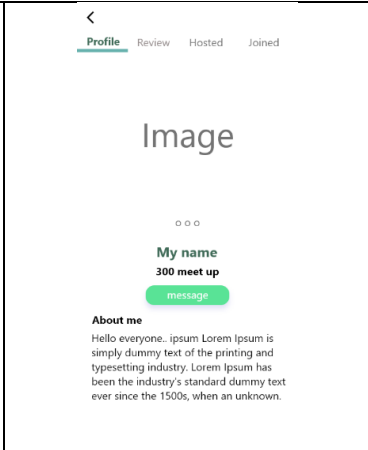
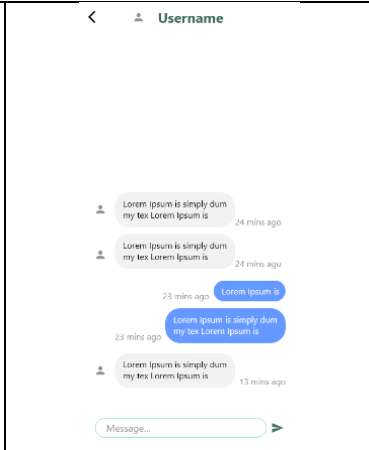
 <p>User clicked profile icon at the bottom navigation bar and visit own user profile page</p>	 <p>User clicked edit button and visit edit page</p>	 <p>User can edit business hours</p>
 <p>User clicked business hours and visit edit hours page</p>	 <p>User clicked Review tab</p>	

Table 4.13 Business user visits own user profile

Business user visits normal user profile

		
---	--	---

CHAPTER 4 SYSTEM DESIGN

User clicked any booking and visit booking detail page	User clicked user header and visit profile page	User clicked message button and visit chat room
--	---	---

Table 4.14 Business user visits normal user profile

Log out

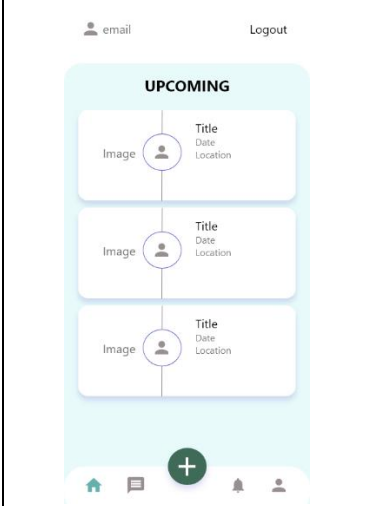
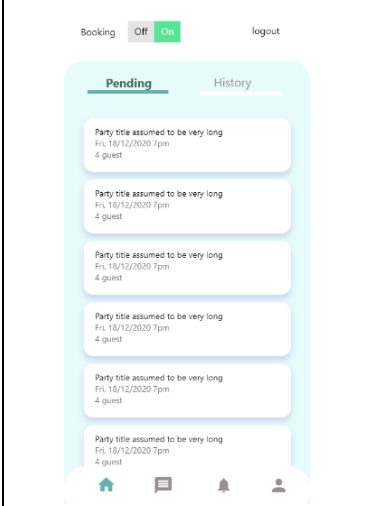
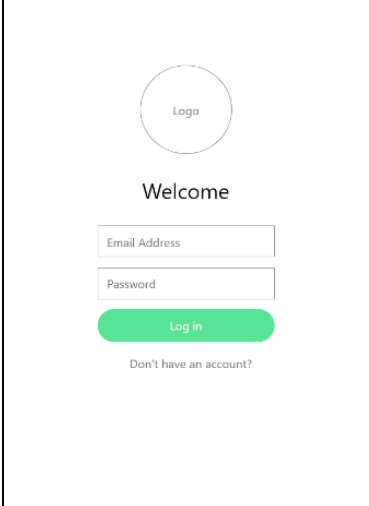
 <p>Normal user clicked logout</p>	 <p>Business user clicked logout</p>	 <p>User redirected to login page</p>
---	--	--

Table 4.15 Log out

CHAPTER 5 SYSTEM IMPLEMENTATION

5.1 Hardware Setup

Requirements for user's device

Aspect	Description
Operating system (OS)	Android Kitkat 4.4 or above
Random Access Memory (RAM)	2GB or above
Internal Storage	1GB or above
Internet Connection	Required for cloud database access

Table 5.1 Requirements for user's device

Requirements for developer's device

Aspect	Description
Operating System (OS)	Window 7 or above, Window 10 is recommended
Processor	Intel core i3 8 th Gen or above
Random Access Memory (RAM)	4GB or above, 8GB is recommended
Disk Space	2GB or above, 4GB is recommended
Disk Drive	SSD is recommended
Screen Resolution	1280 x 800 or above

Table 5.2 Requirements for developer's device

5.2 Software Setup

There is a file with extension “.apk” which indicating that it is an Android Package File which created for Android operating system.

For using it in a physical smartphone operated by Android, download the APK file and open it in file manager, then wait for the download and installation process to be completed.

For using it in Android Studio software, follow the instruction and download the software at the following website: https://developer.android.com/studio?gclid=Cj0KCQiAybaRBhDtARIsAIEG3km5x88Sjx577c_7fvg8NODcxQy2bCDu1YFNHddHVCPTTw3TlcETFmIaApjzEALw_wcB&gclsrc=aw.ds. After that, follow the instruction to set up the emulator at the following website: <https://developer.android.com/studio/run/emulator>. After the emulator is set up, drag

CHAPTER 5 SYSTEM IMPLEMENTATION

the APK file to the emulator and wait for the download and installation process to be completed.

5.3 Setting and Configuration

There is only a simple setting which is to turn either the Wi-Fi or mobile data on since it required user to be online in using this application.

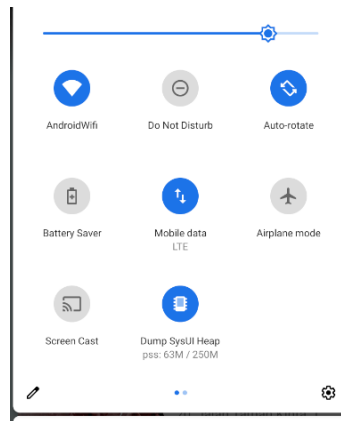


Figure 5.1 Setting and Configuration

5.4 System Operation

Normal user:

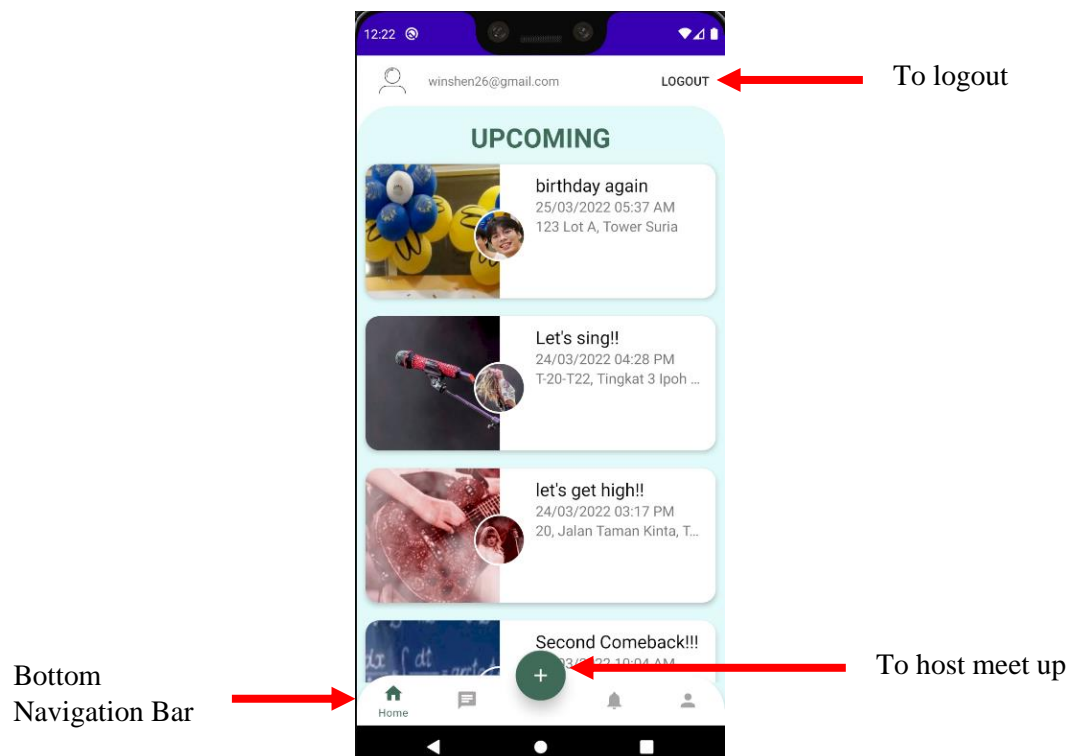


Figure 5.2 User Manual: Home page

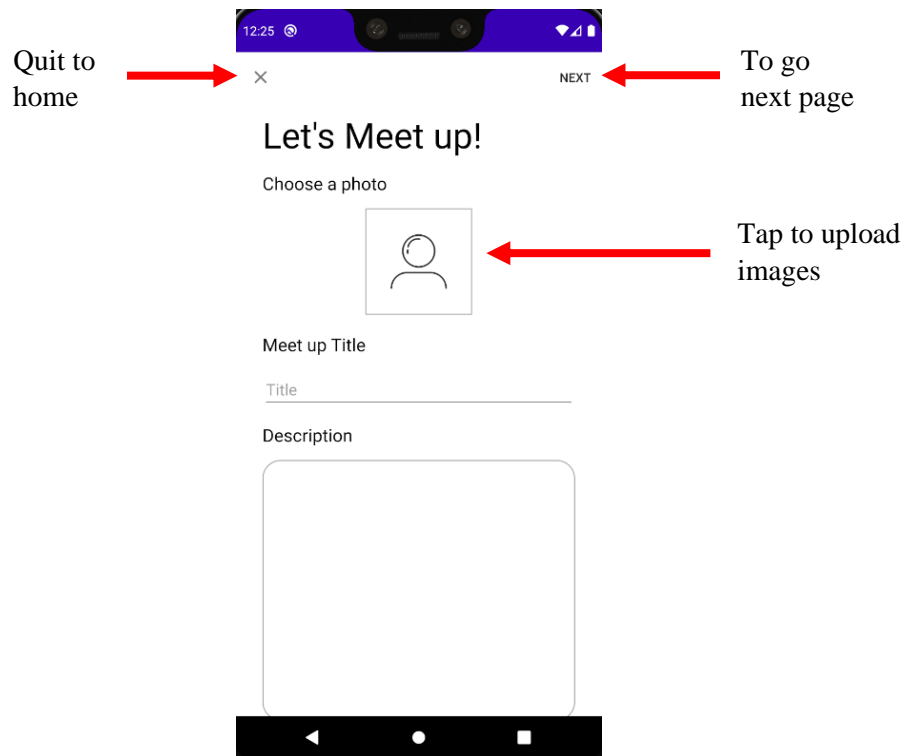


Figure 5.3 User Manual: Host meet up

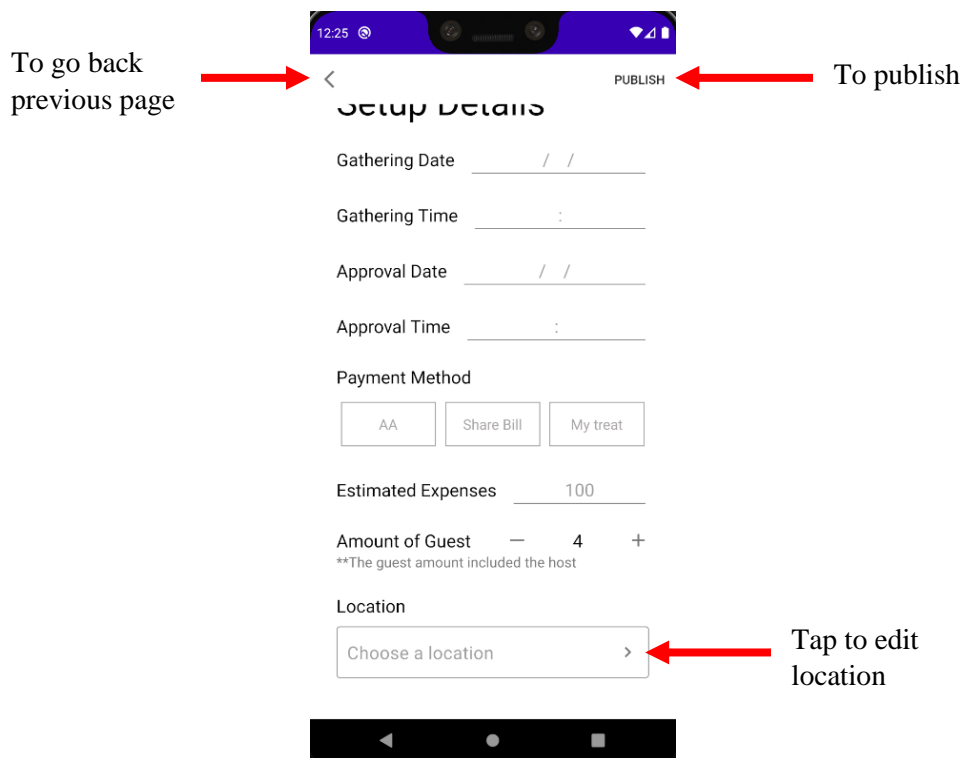


Figure 5.4 User Manual: Host meet up

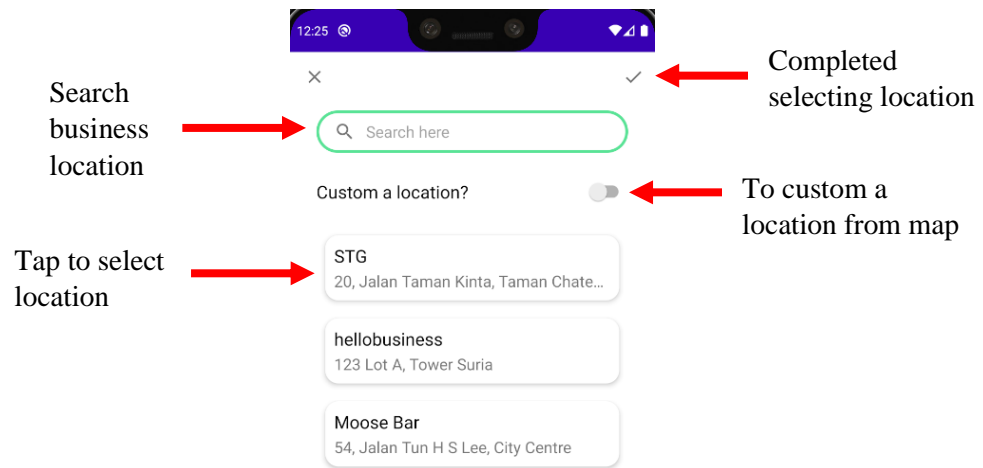


Figure 5.5 User Manual: Host meet up

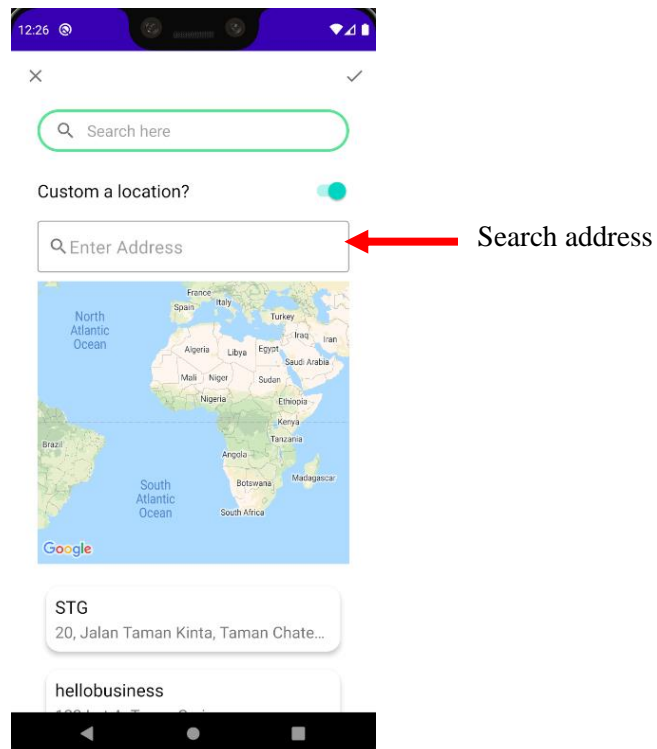


Figure 5.6 User Manual: Host meet up

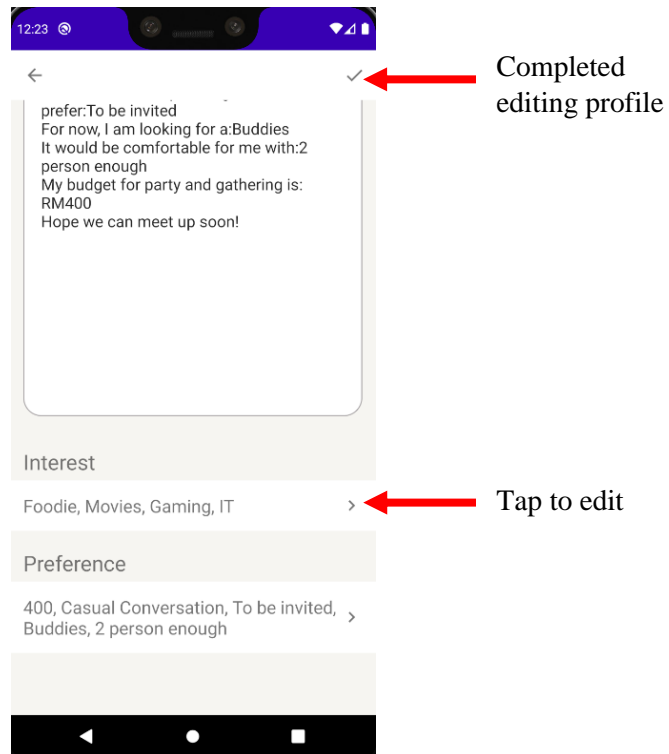


Figure 5.7 User Manual: Edit profile

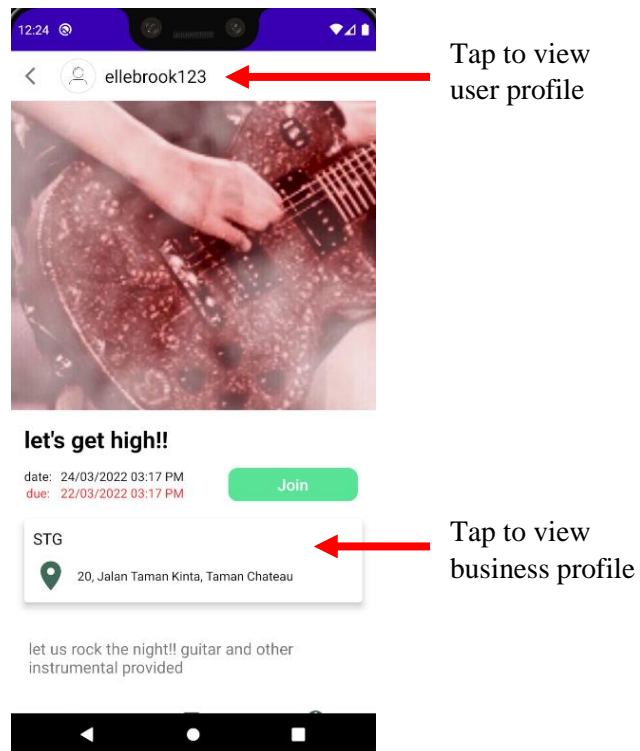


Figure 5.8 User Manual: Join meet up

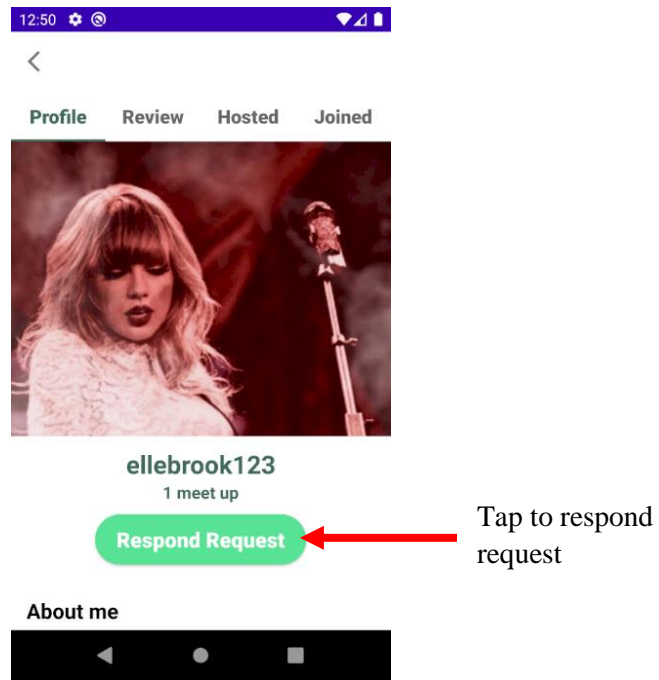


Figure 5.9 User Manual: Respond friend request

Business user:

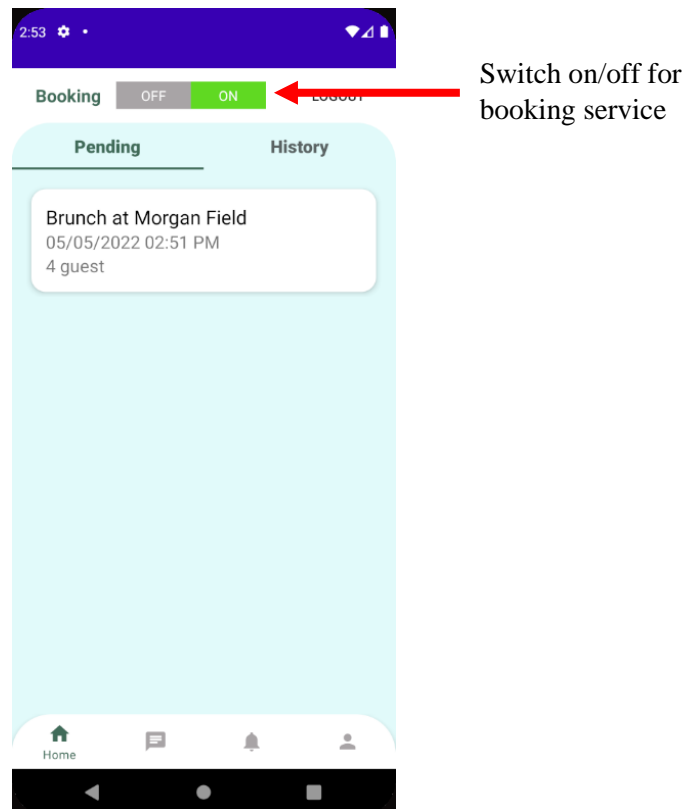


Figure 5.10 User Manual: Business home page

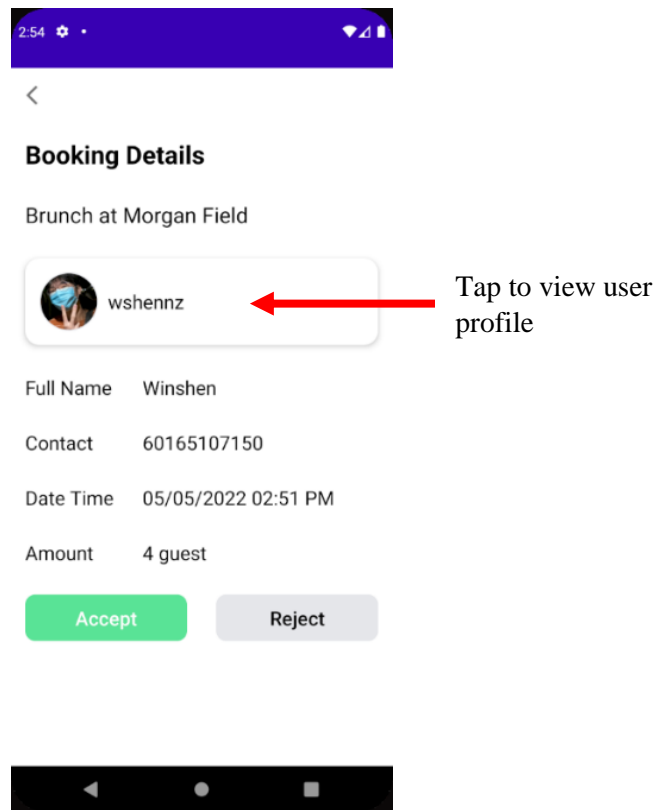


Figure 5.11 User Manual: Booking details

5.5 Concluding Remarks

This section covers the setup, device's requirements as well as a simple user manual to guide the interested user in using the application. User's device required to be operating on system Android Kitkat 4.4 or above. However, if the user wishes to use this application in Android Studio, there is a built-in emulator provided by Android Studio and the user can drag the application file into the emulator. A brief user manual is provided in this section to guide the user in using the main function of the application.

CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

6.1 System Testing and Performance Metrics

System testing is used to verify the developed application meets and satisfy all the expected requirement. There will be several testing to be performed for verification which are functional testing and user interface testing.

Functional testing is to test the functions or project scopes are delivered as expected and with correct output. All the units are integrated together become a function as expected. Usually, this testing involved black box testing without the concern of the codes. For example, when user clicks host meet up, the whole process from setting up details, publish the meet up and to update the database record should be operated without error and wrong output. The key metric in this testing is to verify whether all the functions are working properly (actual outcome meets the expected outcome).

6.2 Testing Setup and Result

Functional Testing:

Test case	Description	Step	Expected Outcome	Actual Outcome	Result
Normal User					
Login as normal user	User can login as normal user	1. Input credentials	Login successfully into normal user dashboard	Login successfully	Pass
Sign up as normal user	User can sign up as normal user	1. Input credentials 2. Select account type 3. Input information 4. Select gender 5. Select interests 6. Select preferences 7. Edit description	Sign up successfully and login into normal user dashboard	Sign up and login successfully	Pass
Host meet up	User can host a meet up	1. Input information 2. Input meet up details	User published meet up and return the home page successfully	Meet up is successfully hosted and user is returned to home page	Pass

CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

		3. Select location			
Join meet up	User can join the interested meet up	1. Select meet up 2. Click join	User sent out application successfully and can cancel request before host accepted	User joined successfully and has the option to cancel joining	Pass
View user	User can view another user's profile	1. Click user's icon or header	User can view other user's profile details	User can view other user's profile successfully	Pass
View review	User can view another user's review	1. Click user's icon or header	User can view other user's review	User can view other user's review successfully	Pass
Give rating	User can rate another user	1. Tap on the star rating 2. Click send	User gave rating successfully	User can rate the other user successfully	Pass
Write review	User can write review for another user	1. Input review 2. Click send	User sent review successfully	User can review the other user successfully	Pass
View business	User can view business profile	1. Click business user header or location card	User can view business profile details successfully	User can view the business profile details successfully	Pass
View profile	User can view own user profile	1. Tap profile icon at bottom navigation bar	User can view own user profile detail successfully	User can view own user profile detail successfully	Pass
Edit profile	User can edit own user profile details	1. Click edit button 2. Edit the information	User can edit the information successfully	User can edit the information successfully	Pass
View meet up	User can view meet up either through profile page or home page	1. Click a meet up	User can view the meet up details successfully	User can view the meet up details successfully	Pass
Check apply	Host can check meet up application	1. Click a hosted meet up 2. Click check apply button	User can see a list of applications	User can see a list of applications successfully	Pass

CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

Accept application	Host can accept application	1. Click check apply button 2. Click accept button	User accepted the application successfully	User accepted the application successfully	Pass
Reject application	Host can reject application	1. Click check apply button 2. Click reject button	User rejected the application successfully	User rejected the application successfully	Pass
Close registration	Host can close registration if there is at least one accepted application	1. Click check apply button 2. Click close registration button	User closed registration successfully	User closed registration successfully	Pass
Send Booking	Host can send booking if the selected location is a registered business user and allow booking	1. Click send booking button 2. Input information 3. Click send button	User can send a booking successfully	User can send a booking successfully	Pass
Check booking	Host can check booking details	1. Click check apply button 2. Click check booking button	User can check the booking details	User can check the booking details successfully	Pass
View chat	User can view chat page	1. Tap chat icon at bottom navigation bar 2. Click any chat record	User can view a list of chat and the chat room message	User can view a list of chat and the chat room message successfully	Pass
Send chat	User can send chat to another user	1. Click any chat record or send message button 2. Input message 3. Click send icon	User sent out a message successfully	User sent out a message successfully	Pass
Add friend	User can add another user as friend	1. Click user's header or icon	User sent out friend request	User sent out friend request successfully	Pass

CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

		2. Click add friend button	and can cancel request		
Respond friend request	User can respond the friend request	1. Click user's header or icon 2. Click respond request button	User accepted or rejected friend request successfully	User accepted or rejected friend request successfully	Pass
View notification	User can view notification	1. Tap notification icon at bottom navigation bar 2. Click any notification record	User can view notification at relevant pages successfully	User can view notification at relevant pages successfully	Pass
Business User					
Login as business user	User can login as business user	1. Input credentials	Login successfully into business user dashboard	Login successfully	Pass
Sign up as business user	User can sign up as business user	1. Input credentials 2. Select account type 3. Input information 4. Setup business hour 5. Edit description 6. Upload photos	Sign up successfully and login into business user dashboard	Sign up and login successfully	Pass
View business profile	User can view own user profile	1. Tap profile icon at bottom navigation bar	User can view own user profile detail successfully	User can view own user profile detail successfully	Pass
Edit business profile	User can edit own user profile details	1. Click edit button 2. Edit the information	User can edit the information successfully	User can edit the information successfully	Pass
View business review	User can view own review	1. Tap Review tab	User can view own review	User can view own review	Pass
View pending booking	User can view pending	1. Tap Pending tab	User can view the list of pending	User can view the list of pending	Pass

CHAPTER 6 SYSTEM EVALUATION AND DISCUSSION

	booking in home page	2. Click any of the pending booking	booking and the details	booking and the details	
Accept booking	User can accept booking	1. Click any pending booking 2. Click accept button	User accepted the booking request successfully	User accepted the booking request successfully	Pass
Reject booking	User can reject booking	1. Click any pending booking 2. Click reject button	User rejected the booking request successfully	User rejected the booking request successfully	Pass
View history booking	User can view history booking	1. Tap History tab 2. Click any history booking	User can view the list of history booking and the details	User can view the list of history booking and the details	Pass
View normal user profile	User can view normal user profile	1. Click user's header or icon	User can view normal user profile details	User can view normal user profile details	Pass
View business chat	User can view chat records	1. Tap chat icon at bottom navigation bar 2. Click any chat record	User can view the list of chat and chat records	User can view the list of chat and chat records	Pass
Send business chat	User can send message	1. Click any chat record or send message button 2. Input message 3. Click send icon	User sent chat message successfully	User sent chat message successfully	Pass
View business notification	User can view notification records	1. Tap notification icon at the bottom navigation bar 2. Click any notification record	User can view notification at relevant pages successfully	User can view notification at relevant pages successfully	Pass
Turn on booking	User can turn on booking availability	1. Click the on button at the top of home page	User turned on booking successfully	User turned on booking successfully	Pass

Turn off booking	User can turn off booking availability if there is no pending booking	1. Click the off button at the top of home page	User turned off booking successfully	User turned off booking successfully	Pass
------------------	---	---	--------------------------------------	--------------------------------------	------

Table 6.1 Functional testing

6.3 Implementation Issues and Challenges:

One of the challenges is that the Firebase Realtime Database structure has to be well planned and structured. As suggested by Google, it is better not to nest the data under each nodes (Structure Your Database, 2021). In another words, it is better to not create many levels of data for each nodes. This is because when the system wants to read and access a particular nodes, all the child of that nodes will be granted access. Hence, to have a better readability and security, it is always recommended to flatten out the data structures. Flatten is also known as denormalization where data are separated into different paths so that they can be called in separate called. However, this is not easy to fully achieve as a beginner to the firebase service. Hence, there will be some flaws and issues whenever structuring the database.

Another challenge is to create a seamless and smooth screen navigation across the application. It is needed to ensure user can access the certain screen and return back to the previous screen seamlessly and smoothly. A lot of configuration needed to be done whenever user is landing on a certain page from another certain page. Especially for the notification, when user clicks on it, the system needs to navigate the user to the relevant page. Hence, this condition is needed to be controlled and configured in the system.

6.4 Objectives Evaluation

Objectives	Evaluation	Achievement
To analyse various dating application in the market based on their functions, strengths, and weaknesses in providing dating services.	This objective is met as the Chapter 2 Literature Review has covered the detail analysis of various similar application in the market based on their functions, strengths, and weakness.	Achieved

To design and develop an application that provides function hosting party and gathering to enable user have chances to meet each other in real person.	This objective is met as the user can host a meet up in the application by setting up relevant information such as title, description, gathering date and time, guest amount and also location.	Achieved
To evaluate and test the deliverable of the project in the aspect of functionality	This objective is met as the Chapter 6 has covered the system testing and results by carried out functional testing.	Achieved

Table 6.2 Objectives Evaluation

6.5 Concluding Remark

This section covers the system testing plan, testing cases, testing results, challenges as well as the objectives' evaluation. The testing results show that the functions are working properly and met the expected outcomes. It is carried out by black box testing which focus on the input and output of the test cases. In order to pass the test, the actual outcome should meet the expected outcome. The last part of this section covers the evaluation of objectives which reflect the success or failure of this project.

CHAPTER 7 CONCLUSION AND RECOMMENDATION

7.1 Conclusion

This project is dedicated to bring users meet up in real life in an easier and efficient way by utilizing the in-application function. Unlike other dating applications that focus on matching then texting, this project is mainly encouraging the real connection and in-person feeling with one and another. Not only meeting one, but meeting in a group of people brings different experience and chemistry among each other through this propose application. On the other hand, It also dedicated to resolve one of the problem in most of the swipe-to-match dating application, which is the cognitive or memory overloading of user's brain while swiping away the other users in the application. This project is carried out successfully where all of the objectives are achieved by using Agile methodology – Extreme Programming (XP). Two types of users – Normal and Business user can interact with each other under the same roof. A detail system methodology, design and diagram are provided in the Chapter 3 and 4. This project was initialized in Jan 2021 for part one and proceeded in May 2022 for part two.

An overview for this application, user can now host meet up and join any meet up hosted by other users. They can also use it as social application to add friend and chat with them. Editing their profile to enhance their online personal presence is also doable in this application. Business user can enhance their profile presence in this application and facilitate their operation through the booking function. It solves the cognitive load problem since there is no swiping feature and users can always look back their joined and hosted meet up history.

7.2 Recommendation

Since this project is carried out by Agile methodology, there is always improvement and enhancement can be done in the next incremental. There are some recommendations for the next increment version. For example, the user cannot delete any record now in the application, this can be improved by adding a delete function for meet up history, chat record, and notification record. Another recommendation is to add a comment section under the meet up page where users interact with each other to decide whether joining the meet up or not. From technical viewpoint, it is suggested to implement alternative authentication methods such as Google and Facebook

CHAPTER 7 CONCLUSION AND RECOMMENDATION

authentication. In the future version, it is suggested that only those users who have attended a meet up together will have the privilege to rate and review each other. It is suggested to provide safety guidelines to the host and joiners for the meet up.

REFERENCES

- [1] “Online Dating - Malaysia | Statista Market Forecast” Statista.com. <https://www.statista.com/outlook/372/122/online-dating/malaysia> (accessed Nov. 17, 2020).
- [2] J. Gitlin. “Dating Apps Are Common, Useful—And Widely Disliked | SurveyMonkey.” SurveyMonkey.com. <https://www.surveymonkey.com/curiosity/dating-apps-and-sites-are-almost-as-common-as-they-are-disliked/#:~:text=They%20allow%20you%20to%20conveniently,story%20shares%20a%20similar%20beginning.> (accessed Nov. 17, 2020).
- [3] L. Moore. “5 reasons people message on dating apps but never meet up” Washingtonpost.com. <https://www.washingtonpost.com/news/soloish/wp/2018/10/08/5-reasons-people-message-on-dating-apps-but-never-meet-up/> (accessed Nov. 18, 2020).
- [4] I. Partanen. “What happens when you swipe left on Tinder?” Cupidhaven.com. <https://cupidhaven.com/what-happens-when-you-swipe-left-on-tinder/#:~:text=Swiping%20left%20on%20Tinder%20means,swiped%20left%20the%20first%20time.> (accessed Jan. 28, 2021).
- [5] K. Tiffany. “How the Tinder algorithm actually works.” Vox.com. <https://www.vox.com/2019/2/7/18210998/tinder-algorithm-swiping-tips-dating-app-science> (accessed Jan. 28, 2021).
- [6] “Powering Tinder® — The Method Behind Our Matching.” Blog.gotinder.com. <https://blog.gotinder.com/powering-tinder-r-the-method-behind-our-matching/> (accessed Jan. 28, 2021).
- [7] A. Abraham. “11 Brutal Reasons Why You’re Not Getting Dates From Tinder.” Medium.com. <https://medium.com/@alexanderabraham5648/11-brutal-reasons-why-youre-not-getting-dates-from-tinder-b925732a496f> (accessed Jan. 28, 2021).
- [8] K. Antoniadis. “10 Ways To Make Time For Friends When You Work A Lot - Corporette.Com.” Corporette.com. <https://corporette.com/ways-to-make-time-for-friends-work-a-lot/> (accessed Dec. 8, 2020).
- [9] “U.S. Dating Apps By Audience Size 2019 | Statista.” Statista.com. <https://www.statista.com/statistics/826778/most-popular-dating-apps-by-audience-size-usa/> (accessed Nov. 30, 2020).

- [10] *Tinder – Dating & Meet People*. (Version 11.29.0). Tinder Inc. Accessed: Nov. 11, 2020. [Online]. Available: <https://play.google.com/store/apps/details?id=com.tinder&hl=en&gl=US>
- [11] “Tinder | Dating, Make Friends & Meet New People” Tinder.com. <https://tinder.com/?lang=en> (accessed Nov. 30, 2020).
- [12] *Bumble – Dating, Make New Friends & Networking*. (Version 5.198.0). Bumble Trading Inc. Accessed: Nov. 20, 2020. [Online]. Available: <https://play.google.com/store/apps/details?id=com.bumble.app&hl=en&gl=US>
- [13] “Bumble - Bumble's Founder And CEO Whitney Wolfe Herd Talks Building Bumble — And Fighting For Gender Equality.” Bumble.com. <https://bumble.com/the-buzz/a-letter-from-whitney-wolfe-herd-founder-and-ceo> (accessed Dec. 1, 2020).
- [14] “Bumble - Privacy.” Bumble.com. <https://bumble.com/en/privacy> (accessed Dec. 1, 2020).
- [15] *Tantan*. (Version 4.3.8). Tantan Cultural Development (Beijing) Co., Ltd.. Accessed: Nov. 23, 2020. [Online]. Available: <https://apps.apple.com/us/app/tantan/id861891048>
- [16] “About us” Tantanapp.com. <http://tantanapp.com/en/about> (accessed Dec. 1, 2020).
- [17] P. Beynon-Davies, C. Carne, H. Mackay, and D. Tudhope, “Rapid application development (RAD): an empirical review,” in *European Journal of Information Systems*, Dec. 2017. [Online]. Available: <https://doi.org/10.1057/palgrave.ejis.3000325>
- [18] C.V. Geambaşu, I. Jianu, I. Jianu, and A. Gavrilă, “Influence factors for the choice of a software development methodology,” in *Accounting and Management Information Systems*, Jan. 2011. [Online]. Available: https://www.researchgate.net/publication/227367934_Influence_Factors_for_the_Choice_of_a_Software_Development_Methodology
- [19] M.L. Despa, “Comparative study on software development methodologies,” in *Database systems journal*, Dec. 2014. [Online]. Available: <https://www.semanticscholar.org/paper/Comparative-study-on-software-development-Despa/28d32b9375e9125624ec614a2cec85c9f0716b13>
- [20] L. Lindstrom and R. Jeffries, “Extreme programming and agile software development methodologies,” in *Information systems management*, Dec. 2006.

- [Online]. Available: <https://doi.org/10.1201/1078/44432.21.3.20040601/82476.7>
- [21] B.G. Sudarsono, S.P. Lestari, A.U. Bani, J. Chandra, and J.F. Andry, “Using an Extreme Programming Method for Hotel Reservation System Development,” in *International Journal*, June. 2020. [Online]. Available: <http://dx.doi.org/10.30534/ijeter/2020/01862020>
- [22] M.C. Paulk, “Extreme programming from a CMM perspective,” in *IEEE software*, Dec. 2001. [Online]. Available: <https://doi.org/10.1109/52.965798>
- [23] L. Cao, K. Mohan, P. Xu, and B. Ramesh, “How extreme does extreme programming have to be? Adapting XP practices to large-scale projects,” in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, 5-8 Jan. 2004. [Online]. Available: <https://doi.org/10.1109/HICSS.2004.1265237>
- [24] V. Öztürk, “Selection of appropriate software development life cycle using fuzzy logic,” in *Journal of Intelligent & Fuzzy Systems*, May. 2013. [Online]. Available: <http://dx.doi.org/10.3233/IFS-120686>
- [25] C.J. West, “A comparison of software project architectures: agile, waterfall, spiral, and set-based,” in *Massachusetts Institute of Technology*, Jan. 2018. [Online]. Available: <http://hdl.handle.net/1721.1/118510>
- [26] H. Hijazi, T. Khdour, and A. Alarabeyyat, “A review of risk management in different software development methodologies,” in *International Journal of Computer Applications*, May. 2012. [Online]. Available: https://www.researchgate.net/publication/266144516_A_Review_of_Risk_Management_in_Different_Software_Development_Methodologies
- [27] S.H. Kan, “The Spiral Model,” in *Metrics And Models In Software Quality Engineering* (2Nd Edition). [Online]. Available: https://flylib.com/books/en/1.428.1/the_spiral_model.html

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 1
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Developing navigation from chat room to user profile
Developing review and rating function

2. WORK TO BE DONE

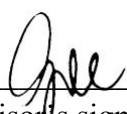
Enhance notification
Develop navigation from notification to specific page

3. PROBLEMS ENCOUNTERED

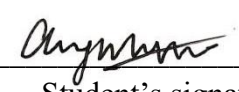
Need to host a user fragment in activity for easy navigation
Need to check if the guest amount included the host

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 2
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Enhancing notification

Developing navigation from notification to specific page

2. WORK TO BE DONE

Develop sign in and sign up for business user

Develop function to choose business user as meet up location

3. PROBLEMS ENCOUNTERED

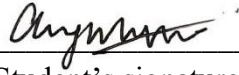
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 3
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Developing sign in and sign up for business user
Developing function to choose business user as meet up location

2. WORK TO BE DONE

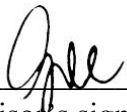
Develop function to view booking information
Develop business profile page

3. PROBLEMS ENCOUNTERED


None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 4
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Developing function to view booking information
Developing business profile page

2. WORK TO BE DONE


Develop chat function to chat with normal user
Enhance notification

3. PROBLEMS ENCOUNTERED

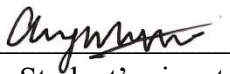
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 5
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Developing chat function to chat with normal user
Enhancing notification

2. WORK TO BE DONE


Develop function to respond booking request
Develop navigation from business user to normal user

3. PROBLEMS ENCOUNTERED


None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 6
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Developing function to respond booking request
Developing navigation from business user to normal user

2. WORK TO BE DONE


Enhance meet up page
Develop booking and close registration function
Write report for chapter 1 and 2

3. PROBLEMS ENCOUNTERED


None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 7
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Enhancing meet up page
Developing booking and close registration function
Writing report for chapter 1 and 2

2. WORK TO BE DONE

Enhance review page
Enhance meet up record page
Test and fix bugs
Write report for chapter 3 and 4

3. PROBLEMS ENCOUNTERED


None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 8
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Enhancing review page
Enhancing meet up record page
Testing and fixing bugs
Writing report for chapter 3 and 4

2. WORK TO BE DONE


Enhance chat room
Enhance navigation across application
Test and fix bugs
Write report for chapter 5 and 6

3. PROBLEMS ENCOUNTERED

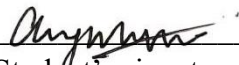
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 9
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Enhancing review page
Enhancing meet up record page
Testing and fixing bugs
Writing report for chapter 5 and 6

2. WORK TO BE DONE


Enhance chat room
Enhance navigation across application
Test and fix bugs
Write report for chapter 7

3. PROBLEMS ENCOUNTERED

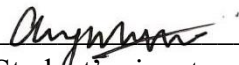
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 10
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Enhancing chat room
Enhancing navigation across application
Testing and fixing bugs
Writing report for chapter 7

2. WORK TO BE DONE

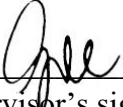
Test and fix bugs

3. PROBLEMS ENCOUNTERED

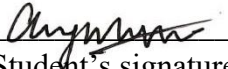
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.: 11
Student Name & ID: Choong Win Shen, 18ACB01717	
Supervisor: Ms. Yap Seok Gee	
Project Title: MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Testing and fixing bugs

2. WORK TO BE DONE


Test and fix bugs

3. PROBLEMS ENCOUNTERED

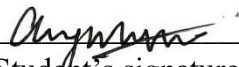
None

4. SELF EVALUATION OF THE PROGRESS

Good in progressing



Supervisor's signature

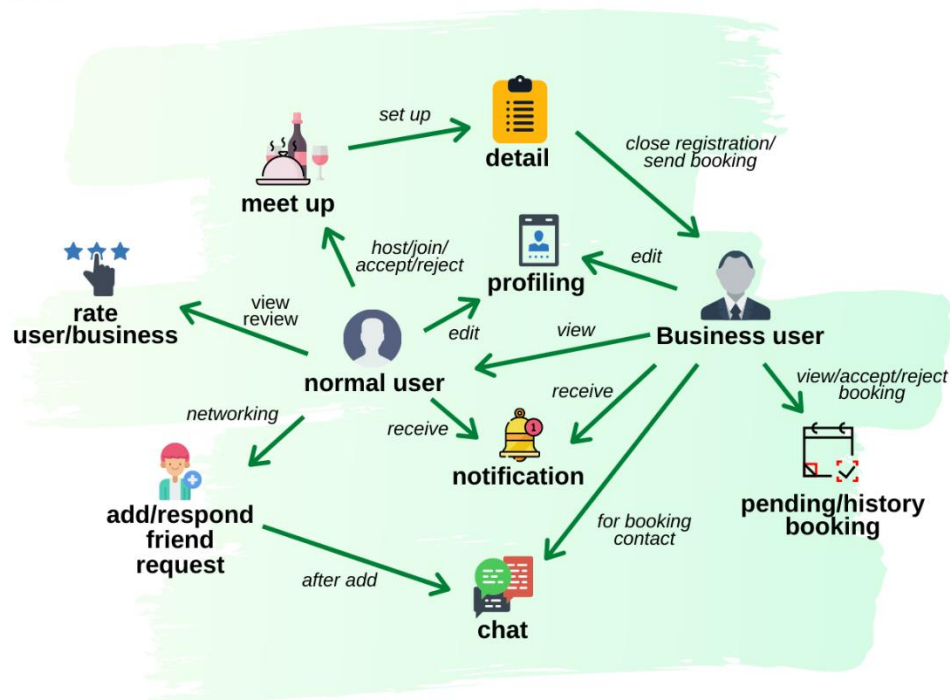


Student's signature

POSTER



A MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING



LET'S MEET UP AND TALK

Ever wonder why is it so hard to meet users on other dating applications? Not only looking for a dating partner but a group of friends or networking? Host a meet-up using this application to meet strangers with shared interests in real life easily and efficiently. Business is given chances to attract more customers through our booking and profiling function.



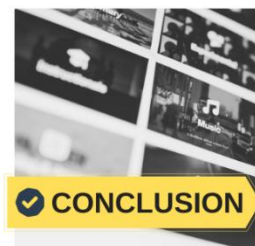
METHODS

Develop using Java. Store and retrieve data from Firebase. Implement eXtreme Programming (XP) methodology



DISCUSSION

Encourage face-to-face and in-person engagement to make new friends. An opportunity for business to attract more customer



CONCLUSION

A bridge or tool to let you meet strangers in real life easier and efficient for friends, partner and networking.

Developed By: Choong Win Shen

Supervised By: Ms. Yap Seok Gee

PLAGIARISM CHECK RESULT

FYP2_ChoongWinShen_turnitin

ORIGINALITY REPORT

3 %	1 %	0 %	2 %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.utar.edu.my Internet Source	1 %
2	Submitted to Heriot-Watt University Student Paper	<1 %
3	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1 %
4	Submitted to Asia Pacific Institute of Information Technology Student Paper	<1 %
5	Submitted to Royal Melbourne Institute of Technology Student Paper	<1 %
6	fict.utar.edu.my Internet Source	<1 %
7	Submitted to Binus University International Student Paper	<1 %
8	Submitted to Federation University Student Paper	<1 %

9	Patrice Chalin, Daniel Sinnig, Kianoush Torkzadeh. "Capturing business transaction requirements in use case models", Proceedings of the 2008 ACM symposium on Applied computing - SAC '08, 2008 Publication	<1 %
10	Submitted to University of Greenwich Student Paper	<1 %
11	Submitted to University of Westminster Student Paper	<1 %
12	docplayer.net Internet Source	<1 %
13	Submitted to Harrisburg University of Science and Technology Student Paper	<1 %
14	Submitted to CSU, San Jose State University Student Paper	<1 %
15	Submitted to HELP UNIVERSITY Student Paper	<1 %
16	docs.fedoraproject.org Internet Source	<1 %
17	Andre Luiz Satoshi Kawamoto, Tereza Gonçaves Kirner, Claudio Kirner. "Experience on the Implementation of a Collaborative Virtual Environment for Educational Applications", 2006 IEEE International Conference on Systems, Man and Cybernetics, 2006 Publication	<1 %
18	speakerdeck.com Internet Source	<1 %
19	www.slideshare.net Internet Source	<1 %

Exclude quotes On Exclude matches Off
Exclude bibliography On

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date:	Page No.: 1 of 1




FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	CHOONG WIN SHEN
ID Number(s)	18ACB01717
Programme / Course	BACHELOR OF INFORMATION SYSTEMS (HONOURS) INFORMATION SYSTEMS ENGINEERING
Title of Final Year Project	MOBILE APPLICATION FOR MEET UP HOSTING AND GATHERING

Similarity	Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)
Overall similarity index: <u> 3 </u> % Similarity by source Internet Sources: <u> 1 </u> % Publications: <u> 0 </u> % Student Papers: <u> 2 </u> %	
Number of individual sources listed of more than 3% similarity: <u> 0 </u>	
Parameters of originality required and limits approved by UTAR are as Follows: (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.



Signature of Supervisor

Name: Ms. Yap Seok Gee

Date: 17/4/2022

Signature of Co-Supervisor

Name: _____

Date: _____



UNIVERSITI TUNKU ABDUL RAHMAN

**FACULTY OF INFORMATION & COMMUNICATION
TECHNOLOGY (KAMPAR CAMPUS)**

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB01717
Student Name	Choong Win Shen
Supervisor Name	Yap Seok Gee

TICK (√)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)

Date: 17/4/2022