

SMART LOCATION ALARM ON MOBILE PLATFORM

LAI JIA YONG


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

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

September 2020

DECLARATION

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

Signature : 
Name : Lai Jia Yong
ID No. : 1800546
Date : 30-9-2020

APPROVAL FOR SUBMISSION

I certify that this project report entitled “**SMART LOCATION ALARM ON MOBILE PLATFORM**” was prepared by **LAI JIA YONG** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : 
Supervisor : Ms. Chean Swee Ling
Date : 30 Sept 2020

The copyright of this report belongs to the author under the terms of the copyright Act 1987 as qualified by Intellectual Property Policy of Universiti Tunku Abdul Rahman. Due acknowledgement shall always be made of the use of any material contained in, or derived from, this report.

© 2020, Lai Jia Yong. All right reserved.

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my research supervisor, Ms. Chean Swee Ling for her invaluable advice, guidance and her enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents and friends who had helped and given me encouragement.

ABSTRACT

Travelling to a place to do something is what most people had experienced. Many people have to work, attend formal meetings or regular events in different places, date and time. With Smart Location Alarm application, it allows the users to mark down the places where the users need to be reminded. Thus, they have lower chances to miss their destinations while driving or taking public transportation. The users can inform their friends by using the application without messaging them manually, once they are reaching the marked places. For daily usage, the users can select their favourite weekdays to trigger the location alarms. Besides, the users can also choose a date and time to schedule the time alarms. With Kanban development methodology, the project had proceeded smoothly and well-organized. Tasks had scheduled and executed on time, which increased the efficiency in the development process. As a result, the application reached the deployment stage and it is available for Android users. Due to some limitations, iOS users are not able to use the application. In short, Smart Location Alarm is a tool that improves convenience and reduces burden for human memory.

TABLE OF CONTENTS

DECLARATION	ii
APPROVAL FOR SUBMISSION	iii
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xvi
LIST OF APPENDICES	xviii

CHAPTER

1	INTRODUCTION	1
	1.1 Introduction	1
	1.2 Problem Statement	1
	1.3 Project Objectives	2
	1.4 Justification/Significance	2
	1.5 Project Scope	3
	1.6 Proposed Solution	4
	1.7 Proposed Approach	6
	1.8 Assumptions and Dependencies	7

CHAPTER

2	LITERATURE REVIEW	8
	2.1 Introduction	8
	2.2 Compare and Contrast Similar Mobile Applications	8
	2.2.1 Tsada-Mobiminder: A Location Based Alarm Mobile Reminder	9
	2.2.2 Map Alarm: Location based Alarm using Mobile Device	10
	2.2.3 Location Based Task Reminder Android Application	12

2.3	Compare and Contrast Software Development Methodologies	13
2.3.1	Waterfall vs Agile	14
2.3.2	Kanban vs Scrum	17
2.4	Suitable Software Development Tools	18
2.4.1	Front-end development	18
2.4.2	Back-end development	19
2.5	Comparison Matrices for Each Topic	21
2.6	Conclusion	22
CHAPTER		
3	METHODOLOGY AND WORK PLAN	24
3.1	Introduction	24
3.2	Development Methodology	24
3.2.1	Kanban Board Planning	24
3.3	Research Methodology	28
3.3.1	Online Survey Questions	28
3.3.2	Online Survey Responses	28
3.4	Work Plan	29
3.4.1	Work Breakdown Structure	29
3.4.2	Gantt Chart	29
3.5	Test Plan	29
3.6	Technologies Involved	30
CHAPTER		
4	PROJECT INITIAL SPECIFICATION	34
4.1	Introduction	34
4.2	Fact Findings	34
4.2.1	Online Survey Analysis	34
4.3	Project Requirements	40
4.3.1	Functional Requirements	40
4.3.2	Non-Functional Requirements	41
4.4	Use Case Diagram	42

4.5	Use Case Descriptions	43
4.6	Activity Diagrams	53
4.7	Conceptual Design	58
4.7.1	Logical Entity Relationship Diagram	58
4.7.2	Logical Data Flow Diagram	59
4.8	Prototype Design	60
4.8.1	Low Fidelity Prototype	60
4.8.2	High Fidelity Prototype	64
CHAPTER		
5	SYSTEM DESIGN	69
5.1	Introduction	69
5.2	User Interface Design	69
5.2.1	Home Page	69
5.2.2	Login Page and Register Page	71
5.2.3	Settings Page	72
5.2.4	Profile Page and Edit Profile Page	73
5.2.5	Friends Page and Friend Profile Page	74
5.2.6	Calendar Page	78
5.2.7	Location Page	79
5.3	Class Diagram	84
5.4	Physical Entity Relationship Diagram	85
5.5	Physical Data Flow Diagram	87
CHAPTER		
6	SYSTEM IMPLEMENTATION	88
6.1	Introduction	88
6.2	Time-based Alarm Module	89
6.3	Location-based Alarm Module	90
6.4	Settings Module	94
6.5	Social Profile Module	95
6.6	Friends System Module	96
CHAPTER		
7	SYSTEM TESTING	97

7.1	Introduction	97
7.2	Test Code	97
7.2.1	Unit Testing	97
7.2.2	Integration Testing	101
7.2.3	Test Suite	104
7.3	System Testing	104
7.4	Usability Testing	116
7.4.1	Usability Testing Survey	117
7.4.2	Usability Testing Result	119
7.5	User Acceptance Testing	120
7.5.1	UAT Survey	120
7.5.2	UAT Result	120
CHAPTER		
8	CONCLUSIONS AND RECOMMENDATIONS	121
8.1	Conclusions	121
8.2	Recommendations	121
REFERENCES		123
APPENDICES		125

LIST OF TABLES

Table 2.1: Comparison Matrix of Similar Mobile Applications	21
Table 2.2: Comparison Matrix of Software Development Methodologies	21
Table 2.3: Comparison Matrix of Front-end Development Tools	22
Table 2.4: Comparison Matrix of Back-end Development Tools	22
Table 3.1: Periods of PLC Stages and SDLC Activities	28
Table 4.1: Use Case Description of View DateTime Pickers	43
Table 4.2: Use Case Description of Set Time-based Alarm	44
Table 4.3: Use Case Description of View Map	45
Table 4.4: Use Case Description of Set Location-based Alarm	46
Table 4.5: Use Case Description of Delete Alarm	47
Table 4.6: Use Case Description of Login As User	48
Table 4.7: Use Case Description of View User Profile	49
Table 4.8: Use Case Description of Update User Profile	50
Table 4.9: Use Case Description of Send Friend Request	51
Table 4.10: Use Case Description of Change Settings	52
Table 4.11: Logical ERD Description	58
Table 5.1: Physical ERD Description	85
Table 6.1: Features available for Anonymous User and Email User	88
Table 7.1: Unit Test Cases	98
Table 7.2: Integration Test Cases	101
Table 7.3: System Test Case of Create a User Account	105
Table 7.4: System Test Case of Login to an Account	105
Table 7.5: System Test Case of Logout from an Account	106
Table 7.6: System Test Case of Edit Profile	106
Table 7.7: System Test Case of Reset Password	106
Table 7.8: System Test Case of Search Friend and Send/Remove Friend Request	107
Table 7.9: System Test Case of Accept Friend Request	108
Table 7.10: System Test Case of Reject Friend Request	108
Table 7.11: System Test Case of Unfriend	109
Table 7.12: System Test Case of Set Time-based Alarm	109

Table 7.13: System Test Case of Update Time-based Alarm	110
Table 7.14: System Test Case of Delete Time-based Alarm	110
Table 7.15: System Test Case of Set Location-based Alarm	111
Table 7.16: System Test Case of Update Location-based Alarm	111
Table 7.17: System Test Case of Delete Location-based Alarm	112
Table 7.18: System Test Case of Search Location and Go to My Location	113
Table 7.19: System Test Case of Show/Hide Friend Shared Location	113
Table 7.20: System Test Case of Enable/Disable Share Location	114
Table 7.21: System Test Case of Filter Location Alarm with Favourite Weekdays	114
Table 7.22: System Test Case of Trigger Location Alarm to Chosen Friends	115
Table 7.23: System Test Case of Settings	115
Table 7.24: User Satisfaction Survey	117
Table 7.25: Usability Testing Result	119
Table 8.1: Project Limitations, Justifications and Suggestions	121

LIST OF FIGURES

Figure 1.1: System Overview	5
Figure 1.2: Kanban Board	6
Figure 2.1: UI of Tsada-Mobiminder	9
Figure 2.2: UI of Map Alarm	11
Figure 2.3: UI of Location Based Task Reminder Android Application	12
Figure 2.4: Waterfall Model	14
Figure 2.5: Agile Model	15
Figure 3.1: Kanban Board in the Early Stage of the PLC	25
Figure 3.2: Kanban Board in the Early Middle Stage of the PLC	25
Figure 3.3: Kanban Board in the Middle Stage of the PLC	26
Figure 3.4: Kanban Board in the Last Middle Stage of the PLC	26
Figure 3.5: Kanban Board in the Last Stage of the PLC	27
Figure 4.1: Different Group of Target Users	34
Figure 4.2: Mobile Platforms that is Widely Used	35
Figure 4.3: Usage of Reminder Application	36
Figure 4.4: Reasons of Using Reminder Application	36
Figure 4.5: Frequency of Forgetfulness from Different Target Users	37
Figure 4.6: Importance of Using Reminder Application	38
Figure 4.7: Criticality of Each Reminder Application Feature	38
Figure 4.8: Conditions to use Smart Location Alarm System	39
Figure 4.9: Use Case Diagram	42
Figure 4.10: Activity Diagram for Set Time-based Alarm	53
Figure 4.11: Activity Diagram for Set Location-based Alarm	54
Figure 4.12: Activity Diagram for Login as User	55
Figure 4.13: Activity Diagram for Update User Profile	56
Figure 4.14: Activity Diagram for Change Settings	57
Figure 4.15: Logical Entity Relationship Diagram	58
Figure 4.16: Logical Data Flow Diagram	59
Figure 4.17: Low Fidelity Prototype Part 1	60
Figure 4.18: Low Fidelity Prototype Part 2	61

Figure 4.19: Low Fidelity Prototype Part 3	62
Figure 4.20: Low Fidelity Prototype Part 4	63
Figure 4.21: High Fidelity Prototype Part 1	64
Figure 4.22: High Fidelity Prototype Part 2	65
Figure 4.23: High Fidelity Prototype Part 3	66
Figure 4.24: High Fidelity Prototype Part 4	67
Figure 4.25: High Fidelity Prototype Part 5	68
Figure 5.1: UI of Home Page as Anonymous (Left), Home Page as User (Right)	69
Figure 5.2: UI of Home Page with Delete (Left), Home Page with Menu (Right)	70
Figure 5.3: UI of Login Page (Left), Register Page (Right)	71
Figure 5.4: UI of Settings Page	72
Figure 5.5: UI of Profile Page (Left), Edit Profile Page (Right)	73
Figure 5.6: UI of Friends Page (Left), Friends Page with Search Result (Right)	74
Figure 5.7: UI of Friend Profile Page	75
Figure 5.8: UI of Send Friend Request (Left), Remove Friend Request (Right)	76
Figure 5.9: UI of Friends Page with Friend Request (Left), Accept or Reject Friend Request (Right)	77
Figure 5.10: UI of Calendar Page (Left), Calendar Alarm (Right)	78
Figure 5.11: UI of Location Page as Anonymous (Left), Location Marker as Anonymous (Right)	79
Figure 5.12: UI of Location Page as User (Left), Location Marker as User (Right)	80
Figure 5.13: UI of Location Page with Search Result (Left), Location Page with Friend Shared Location (Right)	81
Figure 5.14: UI of Favourite Page (Left), Notification Page (Right)	82
Figure 5.15: UI of Location Alarm (Left), Location Alarm from a Friend (Right)	83
Figure 5.16: Class Diagram	84
Figure 5.17: Physical Entity Relationship Diagram	85
Figure 5.18: Physical Data Flow Diagram	87
Figure 6.1: Schedule Time Alarm Code Segment	89
Figure 6.2: Background Tracking Code Segment	90
Figure 6.3: Foreground Tracking Code Segment	91
Figure 6.4: Calculate Distance Code Segment	91

Figure 6.5: Trigger Location Alarm Code Segment	92
Figure 6.6: Receive Notification Code Segment	93
Figure 6.7: Send Notification Code Segment	94
Figure 6.8: Choose Ringtone Code Segment	94
Figure 6.9: Form Validators Code Segment	95
Figure 6.10: Search Friend Code Segment	96
Figure 7.1: Actual Result of Unit Testing	97
Figure 7.2: Actual Result of Integration Testing	101
Figure 7.3: Actual Result of Test Suite	104

LIST OF ABBREVIATIONS

API	Application Programming Interface
CRUD	Create, Read, Update and Delete
CSS	Cascading Style Sheets
DFD	Data Flow Diagram
DBMS	Database Management System
ERD	Entity Relationship Diagram
FCM	Firebase Cloud Messaging
GPS	Global Positioning System
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
iOS	iPhone Operating System
ISP	Internet Service Provider
OS	Operating System
PHP	Hypertext Preprocessor
PLC	Project Life Cycle
SDLC	Software Development Life Cycle
SQL	Structured Query Language
SUS	System Usability Scale
UAT	User Acceptance Testing
UI	User Interface

UML	Unified Modeling Language
WBS	Work Breakdown Structure
XML	Extensible Markup Language

LIST OF APPENDICES

APPENDIX A: Online Survey (Questions)	125
APPENDIX B: Online Survey (Responses)	128
APPENDIX C: Work Breakdown Structure	132
APPENDIX D: Gantt Chart	134
APPENDIX E: User Satisfaction Survey (Responses)	135
APPENDIX F: UAT Survey (Questions)	147
APPENDIX G: UAT Survey (Responses)	151
APPENDIX H: Ratings and Reviews on Deployment Platform	161

CHAPTER 1

INTRODUCTION

1.1 Introduction

Nowadays, people are tending to forget some tasks which are waiting to be done as they are mentally exhausted after a busy working day, especially office workers who live in a hustle and bustle city. Therefore, people are using some ways to record down the future tasks for reminding purpose by making a to-do list or a timetable. However, in the age of digital technology, there are many useful mobile applications that remind people to make a move for a specific task when the scheduled time is up by alerting the people with a customizable ringtone or vibration. For this project, it is about developing a cross-platform mobile application called ‘Smart Location Alarm’ that brings many benefits and offers convenience for the users. One of the differences between a normal reminder application and Smart Location Alarm application is that the normal reminder application is time-based whereas the Smart Location Alarm application belongs to both time-based and location-based. Time-based reminder application means that there is a countdown timer and calendar inside the mobile application so when the time is up it will alert the user. On the other hand, location-based reminder application means that the mobile application is using GPS technology to track the user’s location with its main feature, notify user.

1.2 Problem Statement

Malaysia provides many public transport services like Rapid-KL Bus Service, Keretapi Tanah Melayu, Mass Rapid Transit, Light Rail Transit, Electric Train Service, Monorail and so on. These services are frequently used by the people now to travel around the country conveniently and efficiently. However, digital distraction is a bad phenomenon and a common problem that might cause passengers to miss their bus or train stop (Innes, 2013). Thus, with Smart Location Alarm application, the people will no longer or less likely to miss their bus or train stop as the reminder application will notify the users with a customizable ringtone or vibration.

People who use public transport usually have no idea about the time to arrive at the destination as there might be a delay for public transport. On the other hand, people who drive their own cars may not know about the condition of the traffic whether the route is having a traffic jam issue or a road block by the traffic police that might lead to miscalculation of arrival time. With the help of GPS in Smart Location Alarm application, it can detect the users' current locations and alert them as well as their friends who might be waiting for them, when they are reaching their desired locations. This is what a normal time-based alarm application cannot achieve.

For students and working adults, most of them need to go to the schools or working places every weekdays but not weekends. The problem is the location-based alarms might be triggered if the users passed by the schools or working places on the weekends. Therefore, Smart Location Alarm allows the users to set their desired weekdays to ring. Besides, the users are able to activate or deactivate both time-based and location-based alarms manually. Hence, the application can remind the users when they are reaching their frequently visited places accurately and consistently.

1.3 Project Objective

1. To develop a reminder mobile application which avoids problems that are derived from the limitations of human memory.
2. To realize the concept of time-based and location-based reminder system into a cross-platform mobile application.
3. To obtain good user ratings from Usability Testing, User Acceptance Testing and deployment platform by meeting their satisfaction on the features available.

1.4 Justification/Significance

Although there are many calendar and reminder applications that are available for the users to download from the Google Play Store and Apple App Store, the project of

developing a Smart Location Alarm mobile application is still important to be carried out. This is because many reminder applications are time-based as it contains a feature with set alarm duration to alert users only. Besides, some of the reminder applications are only grouped as location-based because it uses GPS to notify users only. However, Smart Location Alarm is not only time-based but also location-based.

This project brings benefits to the users especially when they are using public transports like buses, trains and so on. Users can depend on the application to have a rest or concentrate on doing their stuff while taking the public transport as Smart Location Alarm will alert them when they reach their destinations. Furthermore, when users have no idea where to go for diet, entertainment or exercise like having dinner, watching movies or playing badminton, users are able to use the Smart Location Alarm application to search for suitable places. Hence, this project allows users to decide a favourable location to go in an easy way.

Moreover, Smart Location Application allows users to sign up for an account as the project consists of a friends system. This system provides a platform to show their friends' current locations when they turn on their sharing. The big difference between users with accounts and users without accounts is users with accounts can permit their friends to view their shared locations whereas users without accounts cannot use the friends system to share locations and view their friends' shared locations. In addition, users with accounts are the only ones that are able to send their notifications to their friends when they reach their desired location.

1.5 Project Scope

This project aims to develop a time-based and location-based alarm system which requires some useful modules in this cross-platform mobile application. First and foremost, the target users for this mobile application are mostly from these three groups of people, the first group is the students, second group is the working adults and last but not least, the third group is the senior citizens. The reason for doing so is these three groups of people might have some differences on their needs and mindsets.

For the data gathering part, an online survey is conducted on these three groups of people. Quantitative research methodology is used in this online survey which focused on a large number of respondents. A target of 40 respondents which stands from 20 students, 10 working adults and 10 senior citizens are needed to participate in this survey. Thus, a variety of data will be collected and analysed according to the opinions of different groups of people.

There are 5 main modules in this project, which are time-based alarm module, location-based alarm module, settings module, social profile module and friends system module. Each module delivers different features for the user. For example, the time-based alarm module allows the user to schedule an alarm which rings on time, whereas the location-based alarm module tracks the user and triggers the alarm when the user is inside the marker area.

1.6 Proposed Solution

For front-end development, there are many programming languages and frameworks can be adopted to develop a cross-platform mobile application such as Ionic, React Native, Flutter, NativeScript and so on. For back-end development, there are also many databases that can be used to manage the data from the application like Firebase, SQLite, MySQL, SQL Server and many more. For the location tracking service, the most common and popular one is by using Google Maps with GPS technology.

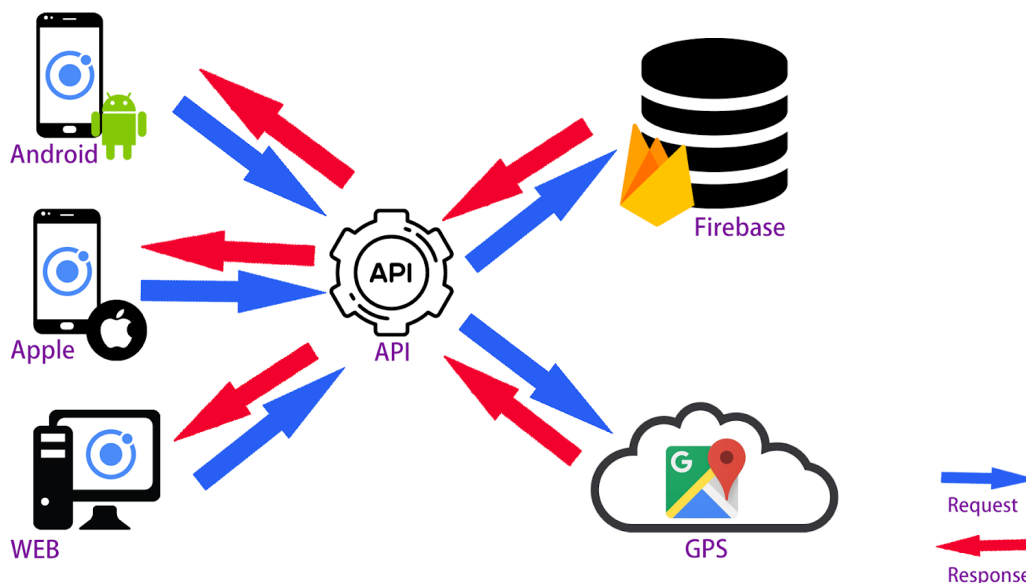


Figure 1.1: System Overview

Figure 1.1 shows the system overview of Smart Location Alarm mobile application in this project. Laptop/Computer is used to develop the mobile application by using Ionic framework with the association of other frameworks like Angular and Apache Cordova. Ionic framework is about producing a hybrid mobile application that uses web technologies and languages to implement, so a web application will also be one of the outputs of the project. However, cross-platform mobile application is the main product of this project, thus Apache Cordova is used to translate the source code which is implemented in HTML, SCSS, TypeScript and so on, to Android and iOS source code.

Firestore is the database used to store the users data like the profile details and avatars as well as the alarms data like the date, time and location, whereas SQLite is the database used to store the settings data like the ringtones. Besides, Firebase Storage is used to store the image files, which are the uploaded profile pictures from the users' phone devices. For the authentication part, Firebase Authentication also provides an easy and secure sign in service that is able to be integrated to any cross-platform application.

The mobile application works on the smartphones that contain GPS function only. Maps JavaScript API is used to activate the Google Maps service in this application. On the other hand, Firebase Cloud Messaging API is also used to send and receive HTTP requests. Thus, the user is able to send notification to one or more users when he or she triggers the alarm. More APIs, components, libraries and plugins involved will be covered in Chapter 6.

1.7 Proposed Approach

There are many categories for software development methodologies including Structured Design, Rapid Application Development and Agile Development. The methodology used in this project is under the Agile Development category, that is Kanban Agile methodology. This methodology is a scheduling system which uses the Kanban Board, Kanban Cards as a to-do list. The first appearance of the Kanban System is from the Toyota Production System.

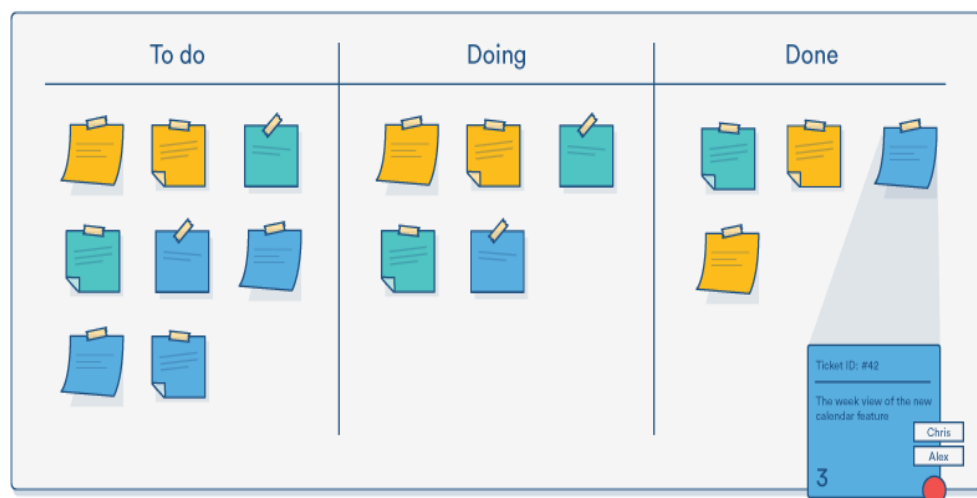


Figure 1.2: Kanban Board (“Kanban Board”, n.d.)

Figure 1.2 shows the Kanban Board that consists of 3 columns which have the headings of “To Do”, “Doing” and “Done”. The naming of these headings is not fixed, it can be changed according to the main activities in the SDLC. For example, Design, Coding, Testing which can be used as the heading names for the columns as they are the phases of the SDLC. Moreover, visual signals like Kanban Cards,

stickies and tickets that are each filled with a smaller task will be placed under one of these columns based on the workflow of the project.

One of the reasons for choosing Kanban methodology is it enables developers to group the project works or tasks and visualize the progress in a convenient way. Hence, a better management of the tasks in the software development project is able to be achieved. The reason is the tasks need to be done that might lead to procrastination can be seen quickly. Besides, Kanban is more flexible as it encourages continuous improvement by showing the progress or delivering an incomplete product to the client continuously. Therefore, developers can gain information and new requests from the client and make modifications to the project as early as possible.

Last but not least, Kanban limits the amount of work in the “Doing” column or Work in Progress is another reason for choosing this software development methodology. By using this methodology, it makes developers concentrate on doing one task instead of many tasks at the same time. Thus, anxiety due to multitasking can be avoided and the workflow of the project cycle is able to go smoothly.

1.8 Assumptions and Dependencies

There are a few assumptions and dependencies for this project. Firstly, smartphones with GPS technology are required in order to use the full functionalities of Smart Location Alarm mobile application. This is because GPS is used to locate the target users that is the crucial part for a location-based reminder application to work successfully. Furthermore, Smart Location Alarm mobile application only supports Android version 5.1 - 10.0.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, 3 topics that are ‘Compare and Contrast Similar Mobile Applications’, ‘Compare and Contrast Software Development Methodologies’ and ‘Suitable Software Development Tools’ will be reviewed and researched. For the first topic, the features and technologies of similar location-based alarm mobile applications will be compared and evaluated to have a clearer view on what features should be included in the project. Next, a comparison between software development methodologies will be made in order to select one of the methodologies for this project. As a result, Kanban Agile methodology which is very flexible and limits Work in Progress tasks is chosen for this project. For the last topic, it is a discussion on the frameworks and services for front-end and back-end development. Ionic with other frameworks like Apache Cordova and Angular is used for front-end development as it is very efficient due to the auto translation to Android and iOS coding languages from web programming languages. On the other hand, Google Firebase is used for back-end development as it is a real-time database system which is more suitable for a reminder application which implements Google Maps to track user’s location.

2.2 Compare and Contrast Similar Mobile Applications

Three similar mobile applications are studied from different journals or articles in this section. One of the mobile applications is from Mindanao University of Science and Technology, Philippines, followed by another one is from Muscat College, Sultanate of Oman and the last mobile application that has been studied is from Sri Ramakrishna Engineering College, India.

2.2.1 Tsada-Mobiminder: A Location Based Alarm Mobile Reminder (Eder, 2015)

The target users for 'Tsada-Mobiminder' are the tourists who travel around in Cagayan de Oro, Northern Mindanao of Philippines. The technology aspects of this project are by using Google Maps Geolocation API and Google Places API. For the user part, a mobile phone with good internet connection is required to use the mobile application. For administrator part, web application is used to add, update, delete and search data of Cagayan de Oro. However, it requires a long-term maintenance service to update the latest information of Cagayan de Oro from the admin manually until the application has halted on long term support or services.



Figure 2.1: UI of Tsada-Mobiminder (Eder, 2015)

Main Features:

1. Provides detailed information about the tourist spots, hotels and restaurants in Cagayan de Oro.

2. Users can use this application to search for other places with Google Places API but not in much detail.
3. Users can set alarm notification by selecting the desired destination from the map.

Strengths:

1. Supports older versions (version 4.0 and above) of Android. Thus, more Android users are available to use this mobile application.
2. Advertises the famous tourist spots and provides useful information about Cagayan de Oro.
3. Offers a user guide that enables the first time user has an easy start to use the application.

Limitations:

1. The UI of the home page is not attractive enough.
2. It has a set range of alarm notifications.
3. Edit and delete alarm functions are not available.

To sum up, users who plan to visit Cagayan de Oro are recommended to use this application as it provides a lot of useful tourist guides for the users. It brings the reminder function to the users who may miss their stop while travelling.

2.2.2 Map Alarm: Location based Alarm using Mobile Device (Rashid and Jhawari, 2018)

The target users for 'Map Alarm' are the long-distance travelers who go to new places frequently. The database involved in this project is SQLite and Java, XML are the programming languages used in the development process. The main services used in this project are Google Play services that help in updating the features of Google Maps to the latest automatically. However, the sharing of the alarm or location is still not available in this prototype.

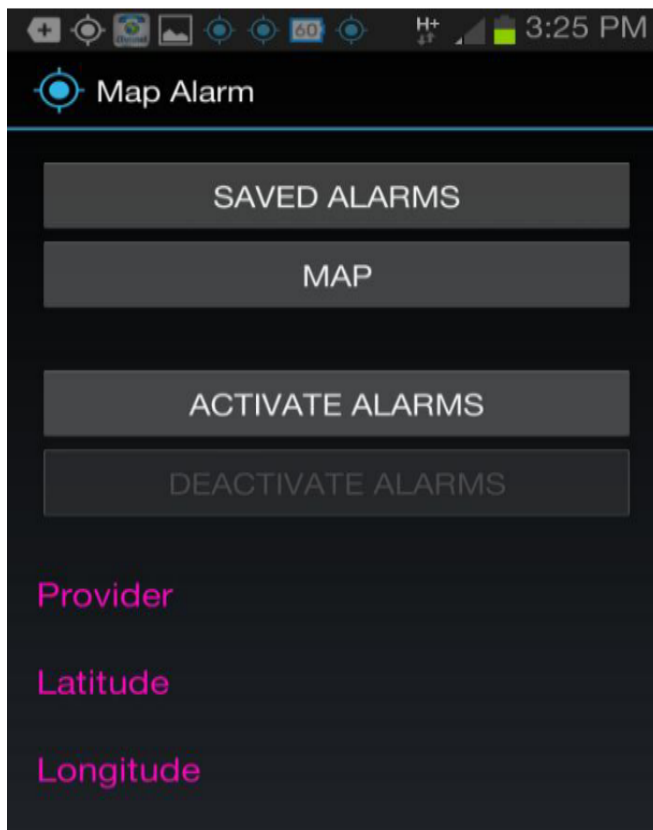


Figure 2.2: UI of Map Alarm (Rashid and Jhawari, 2018)

Main Features:

1. Searches for the nearest place to the places like market, hospital, hotel and so on.
2. Users can add, edit, delete, save the alarms according to the desired places.
3. Users can activate and deactivate the alarm directly.

Strengths:

1. Customizable UI descriptions, marker and audio ringtones.
2. Simple UI for users to make use of the mobile application.
3. Users can view their saved location list and turn on/off ringing alerts as well as delete the alarm conveniently from the view list.

Limitations:

1. Users cannot specify the center location of the user area by using GPS.
2. Users cannot see how far or estimation time of arrival to their target location.

In a nutshell, users who like to travel long distances are suggested to use this application as it will alert the users when they have arrived at the destination. It demonstrates a reminder application to a state of easy and customizable tool.

2.2.3 Location Based Task Reminder Android Application (Nethra et al., 2019)

The target users for this mobile application are the busy workers. The front-end programming languages and DBMS used in this project are the same as 'Map Alarm', which are Java, XML and SQLite respectively. This mobile application consists of a time-based and location-based alarm system. However, this mobile application lacks many useful features like determining the distance and arrival time to the specified location, changing the alarm settings, share location and other functionalities.

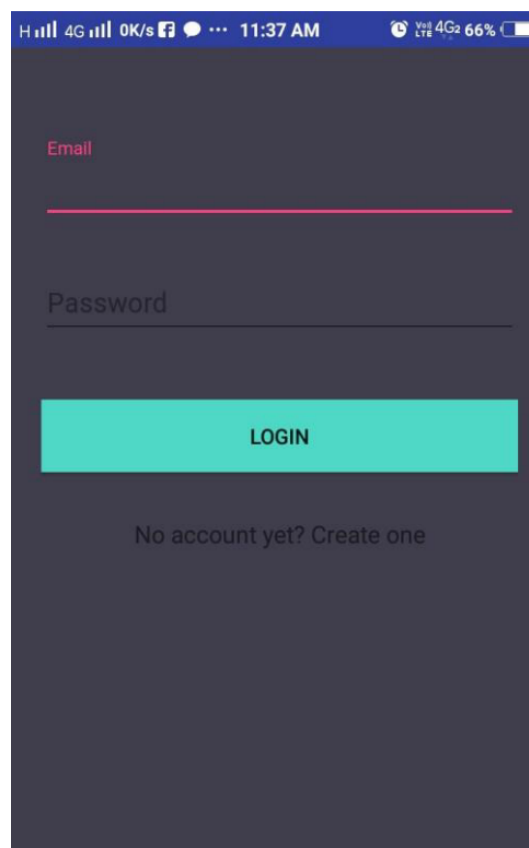


Figure 2.3: UI of Location Based Task Reminder Android Application (Nethra et al., 2019)

Main Features:

1. Provides information about the weather of the selected location.
2. Users can select the date, time and add task description to time-based alarm.
3. Users can choose the location and add task description to location-based alarm.

Strengths:

1. Flexible on switching in between time-based alarm and location-based alarm functions.
2. Allows users to decide to execute their tasks based on the weather's condition.

Limitations:

1. Location-based reminder only works for users who are at a 5km radius of the place specified.
2. Notification is triggered automatically after 2 days when the user is more than 5km radius from the target place.
3. Edit and delete alarm functions are not available.
4. Users cannot see how far or estimation time of arrival to their destination.

In conclusion, users who want to use traditional time-based alarm systems and integrate with location-based features can try this application. It added a creative function in location-based alarm that is showing the weather condition to support users on making decisions.

2.3 Compare and Contrast Software Development Methodologies

Several software development methodologies including traditional Waterfall methodology and Agile methodology that can be subdivided into Extreme Programming, Scrum, Kanban and Crystal are discussed in this section.

2.3.1 Waterfall vs Agile

According to Patil, Panicker and Kv (2016), Waterfall methodology defines the requirements before the project continues to other stages like analysis and development. However, collecting requirements is not an easy process as the users sometimes found themselves are not clear on what they need. On the other hand, Patil, Panicker and Kv (2016) discussed that Agile methodology is breaking down the programming, development and project management into many stages. Thus, developers can proceed to other stages first and integrate the documentation afterward.

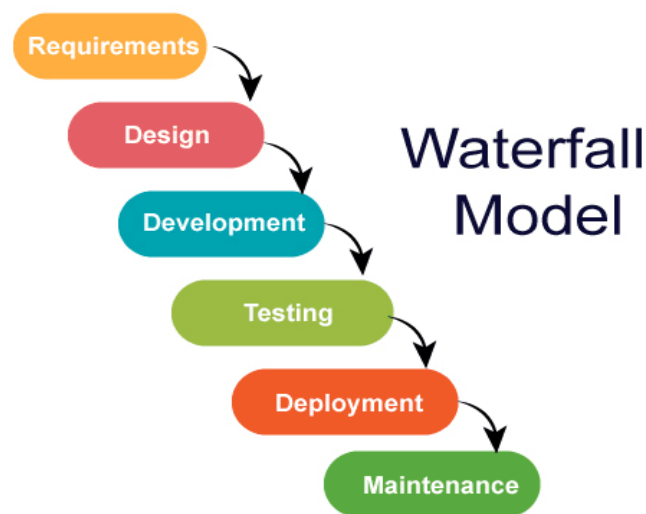


Figure 2.4: Waterfall Model (“Waterfall Model”, n.d.)

Figure 2.4 shows the six stages of Waterfall methodology model that are requirements, design, development, testing, deployment and maintenance. Each stage proceeds one-by-one sequentially.

Supporting Evidence:

1. Instagram used agile methodologies to swift alterations, add or remove features based on user requirements (Patil, Panicker and Kv, 2016).
2. Spotify abandoned Scrum framework which is one of the agile methodologies and adopted a basic agile approach which is more flexible and experimental (Patil, Panicker and Kv, 2016).

Agile methodology is highly suggested to be used for development of Mobile Applications. However, it consists of many frameworks which can be referred and customized to suit the project needs. This can be proved as Altameem (2015) clarified that Agile methodology has many lightweight methods like Scrum, Extreme Programming, Crystal and so on. These methods break the tasks into smaller tasks called iterations. All phases of SDLC should go through one time to complete each iteration. Besides, Agile allows developers to start with predicted requirements and then enhance or change them throughout the development process. However, Extreme Programming focuses on pair programming which is a concept of having two programmers to work together (Yasvi, Yadav and Shubhika, 2019) and Crystal method only emphasizes on the interactions within software development projects rather than processes and tools (Altameem, 2015).

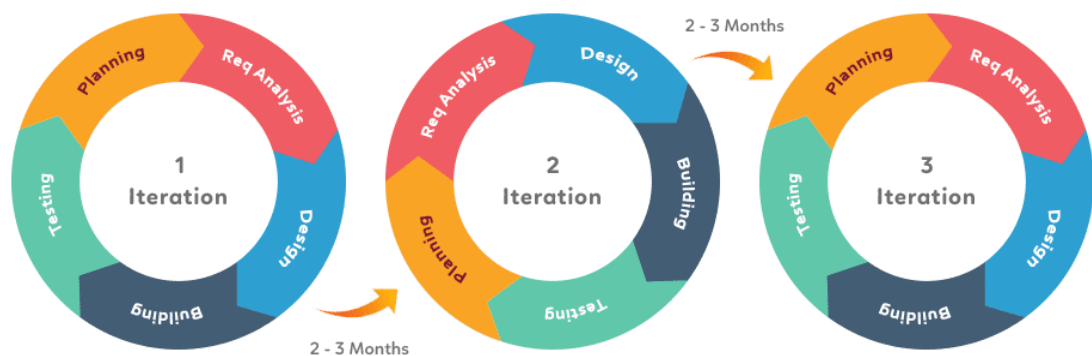


Figure 2.5: Agile Model (“Agile Model”, n.d.)

Figure 2.5 displays the iteration process of Agile methodology. Each iteration will be delivered to the customers for them to check and specify the requirements for the next release (Casteren, 2017).

Sample Characteristics:

1. Extreme Programming is responsive with the changing of client requirements. It has 5 important ways which are Planning, Managing, Coding, Designing and Testing.

2. Scrum is applicable for complex projects that are full of deadlines. Each iteration in this framework is also known as a sprint that is usually 2 to 4 weeks long. It can scale the team to hundreds of people easily.
3. The categorization of Crystal method depends on the team size and criticality of the project.

Gechman (2019) stated the strengths of Waterfall methodology are that every aspect of the project are well planned and documented. The development process of the application from documentation to deployment part can be carried out according to plan. Besides, it avoids the project costs for altering requirements during project execution as the requirements of the project are well defined.

Gechman (2019) also explained the limitations of Waterfall methodology are difficult in adjusting the feature set in the middle stage of development, which happens frequently as the changing of business environments and problems that are unpredictable or uncovered in the early stage of development. According to Patil, Panicker and Kv (2016), the limitations of Waterfall methodology are the risk of failure is high when the early project requirement specifications are not clear .

The strengths of Agile methodology are the development process is open to change. Besides, Agile methodology focuses on people including the clients, end users and the developers. Agile methodology is suitable for developing Mobile Applications as it can cope with the rapid pace of change in the market (Patil, Panicker and Kv, 2016). Agile methodology is good in detecting the faults as testing is gone through every iteration. Software functionality progress can be checked as every process in software development is updated incrementally. By using Agile methodology, developers are flexible on changing the design or requirements (Altameem, 2015).

However, Agile methodology also has some limitations. Due to many iterations are needed in Agile methodology, it causes some difficulties on scalability to large projects (Altameem, 2015). According to Altameem (2015); Patil, Panicker

and Kv (2016), Agile methodology requires a lot of time just to complete a single feature or carry out testing as the development of Mobile Applications requiring frequent revision to meet customer needs that might be subject to change. Product design will be neglected easily as Agile methodology mainly focuses on developing code (Altameem, 2015).

In a nutshell, Agile methodology is a popular software development methodology that brings many positive impacts to the quality of the software and the efficiency of the development process. Thus, Agile methodology is chosen for this final year project. The comparisons between these two agile methods Kanban and Scrum will be discussed in the following part.

2.3.2 Kanban vs Scrum

Scrum and Kanban belong to the agile family. According to Brezočnik and Majer (2016), Scrum helps the team to release a new product as soon as possible and it consists of three roles that are Product Owner, Scrum Master and Team. However, these roles are not applicable for individual projects. Furthermore, Scrum has a sprint that is a fixed-length iteration and procedures or documents like Sprint Planning, Sprint Backlog, Sprint Review and Sprint Retrospective take place in each sprint. On the other hand, Kanban does not consist of iteration and it uses Just-In-Time technique to schedule tasks to avoid re-planning from occurring. Kanban has three basic guidelines which enable developers to visualize workflow, limit Work in Progress and Principle 'Pull' that helps in avoiding delays by limiting new tasks in certain stages.

Sample Characteristics:

1. Visualization workflow for Scrum is through Scrum Board whereas for Kanban is through Kanban Board.
2. Scrum resets the tasks with each sprint whereas Kanban works with constant flow.

One of the strengths of Kanban methodology is Kanban limits Work in Progress tasks. Thus, developers can focus on one task instead of working on too many tasks at a time. Kanban allows developers to change the workflow at any project cycle time. Bug-fixing can be executed immediately after the bug has been discovered with Kanban methodology (Brezočnik and Majer, 2016). The limitations of Scrum methodology are Scrum requires more effort compared with Kanban like in preparing Sprint Backlog, Burndown Chart that is not required in Kanban. Alteration in the work plan is not allowed in Scrum when the sprint is running. Bug-fixing in Scrum is planned, and it is not subject to change the sprint plan (Brezočnik and Majer, 2016).

In conclusion, Kanban is suitable for a high maturity level or self-initiative team or individual to use whereas Scrum is recommended for a team that requires time limit or deadlines to gain motivation to proceed on software development. Thus, Kanban methodology is selected in this final year project.

2.4 Suitable Software Development Tools

To develop a mobile application, studies on suitable software development tools are very important to the project. Technology aspects like programming languages, frameworks, databases and services are covered in this section.

2.4.1 Front-end development

For the front-end development, comparison is made in two different frameworks that are Ionic and React Native. Ionic is used with two pre-existing frameworks which are Angular and Cordova. Angular is a web framework that includes more concepts than React. Whereas, Cordova allows developers to use native functionality like running the mobile application in the background. On the other hand, React Native is based on React. Cross-platform and the native components are linked tightly together, thus React Native codebase is more suitable to develop a cross-platform mobile application (Borick, 2018). However, React Native is complicated as well as harder

to build. Conversely, Ionic is simple to build and it consists of Cordova framework which drastically helps in using native components.

Sample Characteristics:

1. Ionic is close to the trend of abstraction as the native implementation has been extracted out.
2. React Native is far from the trend of abstraction as it is with the native implementation.

Strengths:

1. Angular provides more concepts such as Bindings, Components, Routing and Services.
2. Ionic adds extra Angular components that will change appearance depending on different devices.
3. Ionic is easy to build and maintain.
4. React Native can build a higher quality application.

Limitations:

1. React includes concepts of Bindings and Components only.
2. The barrier of entry for React Native is higher.
3. Ionic separates it's framework from the native side, as a result in building processes become intensive.

In a nutshell, Ionic framework is chosen to build the Smart Location Alarm which is a cross-platform mobile application with reminding purpose. The reasons for choosing this framework are it is easier to get started and it also allows developers to use the native components which is very important to the background functionalities.

2.4.2 Back-end development

For the back-end development, two types of DBMS, Firebase and MySQL are compared. Firebase is a real-time database from Google company. Besides, Firebase

uses NoSQL programming language and it does not need any server-side code. On the other hand, MySQL is a relational database from Oracle Corporation which uses SQL programming language. According to Ohyver et al. (2019), MySQL requires server-side code which can be implemented with other programming languages like PHP, Ruby and so on.

Sample Characteristics:

1. Firebase can sync the data across connected devices and it is available without network connectivity via local cache.
2. MySQL can store data into tables consistently. When a table requires connection with another table, foreign key is used.

Strengths:

1. Firebase has a better performance in CRUD functions.
2. Firebase allows users to update the data to the server immediately, thus synchronization is achieved easily.
3. Firebase provides developers to concentrate in building the best apps as server-side code is not required.
4. MySQL can handle complex queries.

Limitations:

1. MySQL has a slower performance in CRUD functions.
2. MySQL takes time in implementation of server-side code and configuration in order to access data in real time.
3. Difficulties in query and migrate complex data with Firebase.

To sum up, Firebase DBMS is selected as the system to manage the back-end database of Smart Location Alarm mobile application with reminder purpose. This is because this mobile application does not need to store complex data. Furthermore, a real-time database is good for synchronization, so users with different platforms can retrieve the same data.

2.5 Comparison Matrices for Each Topic

Applications/ Features	Tsada-Mobiminder	Map Alarm	Location Based Task Reminder
Add Location-based Alarm	Yes	Yes	Yes
Edit Alarm	No	Yes	No
Delete Alarm	No	Yes	No
Enable and Disable Alarm	No	Yes	No
Search Location	Yes	Yes	No
Provide Location Information	Yes	No	Yes
Show Route Path	Yes	No	No
Add Message	No	Yes	Yes
Customizable UI and Ringtone	No	Yes	No
Account Login	No	No	Yes
Integrate Time-based Alarm	No	No	Yes

Table 2.1: Comparison Matrix of Similar Mobile Applications

Methodologies/ Features	Waterfall	Agile	Scrum	Kanban
Requirement Gathering	Before Analysis	Any Changeable Stage	Any Changeable Stage	Any Changeable Stage
Iteration	No	Yes	Yes	No (use Just-In-Time technique)
Alteration Flexibility	No	Yes	No (when sprint is running)	Yes
Visualization Workflow	No	No	Yes	Yes

Limits WIP	No	No	No	Yes
Time Consuming	Yes (if changes occur not in early stage)	Yes (frequent revise with customer)	Yes (requires many efforts on charts)	No
Risk of Failure	High (if project requirements are not clear)	Low	Low	Low

Table 2.2: Comparison Matrix of Software Development Methodologies

Front-end Tools/ Features	Ionic	React Native
Concepts Involved	More	Less
Build and Maintain	Easy	Difficult
Native Components	Yes (separated out)	Yes (linked together)
Cross-platform	Yes	Yes

Table 2.3: Comparison Matrix of Front-end Development Tools

Back-end Tools/ Features	Firebase	MySQL
Server Side Code	Not Required	Required
Data Synchronization and Real Time Configuration	Easy	Difficult
CRUD Performance	Fast	Slow
Complex Data Query and Migration	Difficult	Easy

Table 2.4: Comparison Matrix of Back-end Development Tools

2.6 Conclusion

In conclusion, all reminder applications which have been compared and contrasted above, have their own strengths and limitations. None of them is flawless as there is no perfect application or software existing in this world. However, the strengths of these mobile applications can be learned and applied in this project. For example,

enable and disable alarm, customizable ringtone and other useful features are added into Smart Location Alarm application. Conversely, the limitations acted as the negative characteristics of an application to be avoided in this project. For instance, the radius of each location alarm should be adjustable as well as basic edit and delete of the alarms must be implemented as they are essential for the user experience.

Kanban Agile methodology has a good overview of grouped tasks and uses Just-In-Time technique instead of iteration which is very time consuming on repetition of all SDLC processes. Consequently, it is chosen as the software development methodology in this project to determine the to-do tasks, ongoing tasks and completed tasks in 5 stages of PLC that will be shown in Chapter 3. In addition, Ionic framework with other frameworks including Apache Cordova and Angular is used for the front-end development whereas Firebase DBMS from Google is used for the back-end development.

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

This chapter covers the details of development methodology and research methodology. Work Plan including the WBS and Gantt Chart as well as Test Plan will be presented in this chapter. Besides, the technologies involved will also be discussed here, such as the hardwares, softwares, services, programming languages and frameworks required in this final year project.

3.2 Development Methodology

There are many types of software development methodology from the traditional approach Waterfall methodology to modern approach Agile methodology. Kanban is one of the methodologies from Agile Family and it is selected in this project. It is a useful methodology which developers can visualize and organize the workflow conveniently.

The online platform or tool that is used for creating the Kanban Board in order to organize the project tasks is Kanban Tool. A Kanban Board consists of many Kanban Cards with one of them taking one task. Each Kanban Card can set its task description, priority and dateline. Besides, it can also allow developers to customise the card type based on different card colours and assign other team members into the project.

3.2.1 Kanban Board Planning

Kanban Boards in 5 different stages of PLC which covered a variety of activities in SDLC will be shown in this subsection.

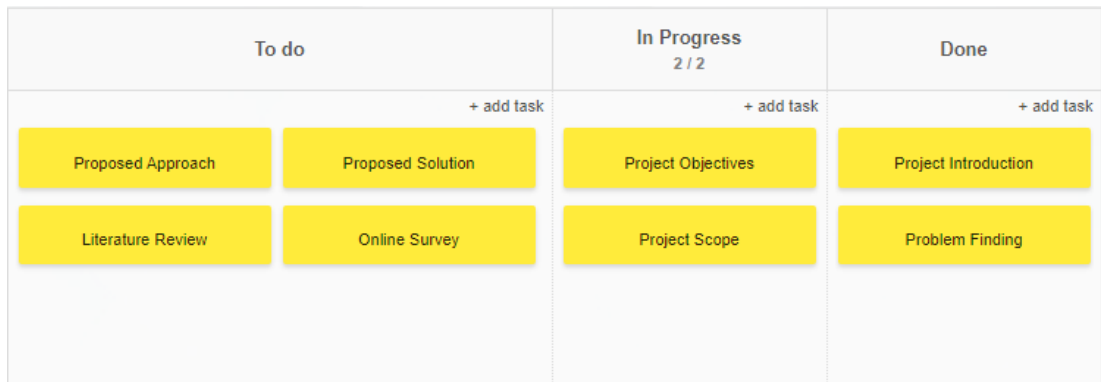


Figure 3.1: Kanban Board in the Early Stage of the PLC

Figure 3.1 shows the Kanban Board in the early stage of the PLC. This Kanban Board has three columns that are “To Do”, “In Progress” and “Done”. It has a Work in Progress limit of 2 tasks in this Kanban system. In the early stage of the PLC, Project Introduction and Problem Finding tasks are completed around week 3 of the semester. Project Objectives and Project Scope which required a longer time to prepare are placed into the “In Progress” column whereas other parts of Chapter 1, Chapter 2 Literature Review and Online Survey tasks are pending to be executed.

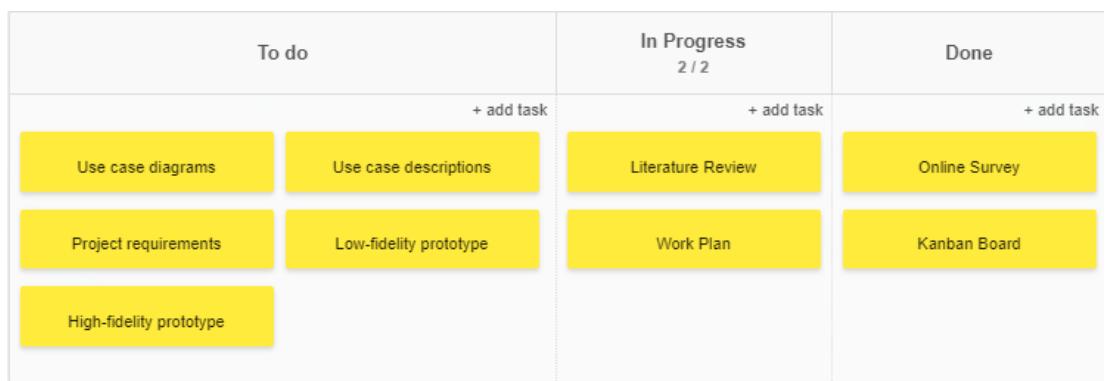


Figure 3.2: Kanban Board in the Early Middle Stage of the PLC

Figure 3.2 illustrates the Kanban Board in the early middle stage of the PLC. In the early middle stage of the PLC, use case modeling, defining project requirements and developing prototype tasks are waiting to be carried out whereas Online Survey tasks including questions constructing, responses collecting and graphs analysing as well as Kanban Board planning are accomplished around week 10 of the semester. On the other hand, tasks from Literature Review which takes a lot

of time to study the articles and perform critical writing plus tasks from Work Plan or Project Schedule which consists of WBS and Gantt Chart are still ongoing.

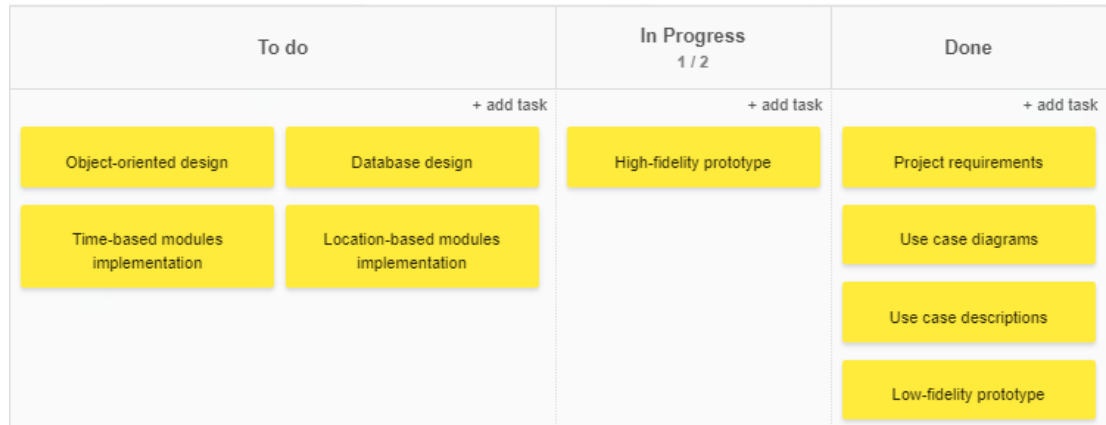


Figure 3.3: Kanban Board in the Middle Stage of the PLC

Figure 3.3 presents the Kanban Board in the middle stage of the PLC. In the middle stage of PLC, defining functional and non-functional requirements, use case modeling and preparing a low-fidelity prototype are completed around week 13 of the semester. On the other hand, Object-oriented design like Class Diagram and Sequence Diagrams as well as Database design like ERD and DFD are waiting to be done after tasks from high-fidelity prototype are finished. Besides, the implementation or coding part of the project also lies into the “To do” column.

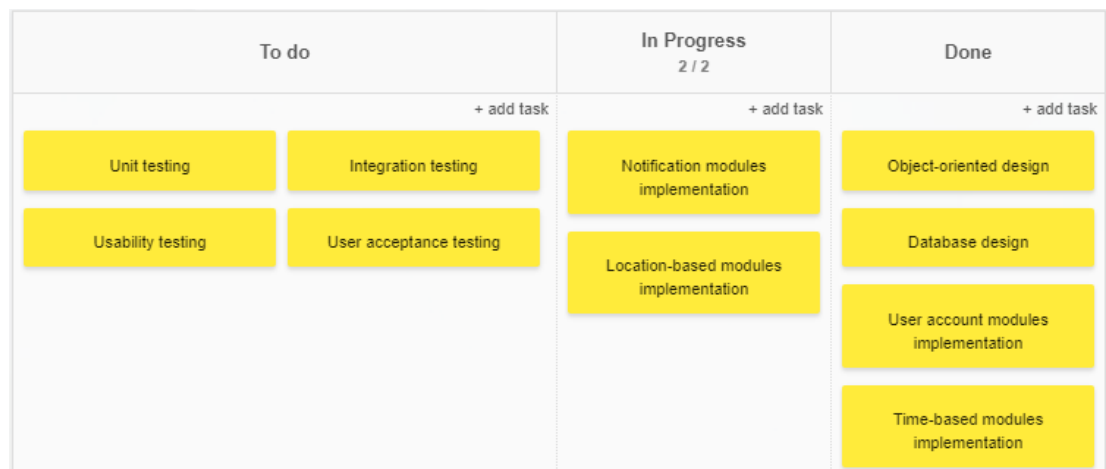


Figure 3.4: Kanban Board in the Late Middle Stage of the PLC

Figure 3.4 displays the Kanban Board in the late middle stage of the PLC. In the late middle stage of PLC, the diagrams design tasks and some coding tasks have been accomplished. The remaining coding tasks are still in progress until the mobile application is fully implemented. Furthermore, testing of the mobile application including Unit Testing, Integration Testing, Usability Testing and User Acceptance Testing are queueing to be carried out.

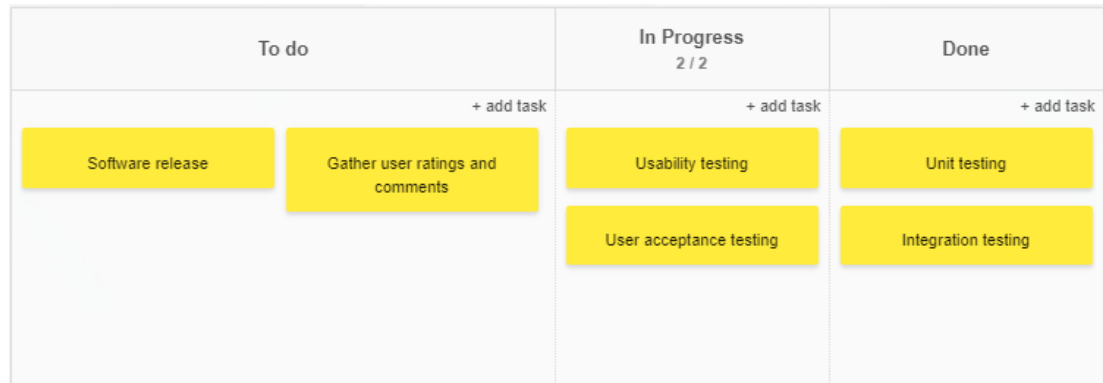


Figure 3.5: Kanban Board in the Late Stage of the PLC

Figure 3.5 shows the Kanban Board in the late stage of the PLC. In the late stage of the PLC, Usability Testing and User Acceptance Testing are almost completed whereas Unit Testing and Integration Testing are already completed. After all of the testing tasks have been executed and the mobile application works well without any high severity error, software deployment or software release is going to be implemented. End users are able to give ratings and comments after they try out and use the mobile application.

Although all these Kanban Board Planning should be scheduled in Just-In-Time which means the tasks should only be moved from one column to another when the time is up. The example above is just a guide on using the Kanban Board to view the tasks status whether the tasks are on hold, still going or already fulfilled. The start dates and end dates or periods of the PLC stages are defined in the table below.

PLC Stages	Start Dates (dd/mm/yyyy)	End Dates (dd/mm/yyyy)	SDLC Activities
Early Stage	15/1/2020	17/2/2020	Planning to Analysis
Early Middle Stage	18/2/2020	25/3/2020	Analysis to Design
Middle Stage	26/3/2020	12/6/2020	Design to Coding
Late Middle Stage	13/6/2020	31/7/2020	Coding to Testing
Late Stage	1/8/2020	4/9/2020	Testing to Deployment

Table 3.1: Periods of PLC Stages and SDLC Activities

3.3 Research Methodology

There are two main types of research methodology, one is qualitative research like interview and focus group while another one is quantitative research like questionnaire or online survey. Quantitative research is chosen as the research methodology as well as data gathering technique in this project. Thus, an online survey is conducted on three groups of target users that are students, working adults and senior citizens respectively.

The aim of this online survey is to figure out the purpose of using a reminder application, collect user requirements and suggested features for Smart Location Alarm mobile application. The survey consists of 9 questions including open-ended questions, close-ended questions and multiple-choice questions. Besides, Google Form is used as an online platform or tool to conduct the survey. The questions and responses of the online survey will be stated in the subsections below respectively. The analysis of the responses gathered is conducted in the next chapter, Chapter 4.

3.3.1 Online Survey Questions

The questions of the Online Survey are attached as Appendix A.

3.3.2 Online Survey Responses

The responses of the Online Survey are attached as Appendix B.

3.4 Work Plan

In this section, the work plan of the entire project will be presented in the WBS and Gantt Chart. WBS shows the milestones and the smaller tasks expanded from them, whereas the date scheduled for the execution of each task and milestone can be visualised in the Gantt Chart.

3.4.1 Work Breakdown Structure

Work Breakdown Structure is attached as Appendix C.

3.4.2 Gantt Chart

Gantt Chart is attached as Appendix D.

3.5 Test Plan

Testing Strategy:

First and foremost, Unit Testing will be executed to test each feature separately and Integration Testing will be the next stage to test all of the features combined in a group. After Integration Testing is completed, System Testing that runs the application on Android and iOS will be conducted. Usability Testing will be carried out to collect the feedback from the experts and novices. Last but not least, User Acceptance Testing will be implemented to collect the feedback from the end users before the deployment of Smart Location Alarm mobile application.

Testing Approach:

Unit testing and Integration Testing will be carried out by following the guidelines and testing principle provided in the Ionic Documentation. For instance, by using mock objects and testing scripts. System Testing is taking place on both Android phone and Xcode simulator as the signing certificate for iOS phone is too expensive. For Usability Testing, User Satisfaction Survey will be used to record the test scenario and test result from each participant whereas Usability Test Report will be prepared after all participants have tested the mobile application. The scope of Usability Testing is about 3 novices and 3 experts are chosen to participate in this testing. Other than that, the testers of the User Acceptance Test are the normal end

users or target users like the students, working adults and senior citizens. Thus, a goal of 10 users or more are required to undergo the User Acceptance Testing. After they use the application, they are highly suggested to fill up the UAT Survey that will be used to evaluate the application.

Features to be tested:

CRUD time-based alarm, CRUD location-based alarm, login and register account, edit user profile including update avatar and account details, share location to friends, view friends' shared locations, favourite weekdays to ring, alarm notifying chosen friends, alarm ringing on time or on place and application settings such as change ringtone, ringtone duration, ringtone vibration as well as do not disturb mode.

Features not to be tested:

None.

Criteria of item passed:

The feature is carried out successfully and the actual output is matched with the expected output.

Criteria of item failed:

The feature is carried out unsuccessfully or the actual output is not matched with the expected output.

3.6 Technologies Involved

Hardwares:

- Laptop/Computer
 - A laptop or computer is used for project development and documentation. The development processes from planning to deployment require a laptop or computer with coding softwares, documentation softwares and other softwares.
- Mobile Phones with GPS

- One Android OS mobile phone and another iOS mobile phone are used for testing and deployment purposes. Both mobile phones should have the GPS function. Smart Location Alarm application will be installed in both mobile phones first and tested before and after the deployment of the application.

Softwares:

- Visual Studio Code
 - Visual Studio Code acted as the source code editor for this project. The blank Ionic and Firebase projects will be installed with the assistance of Command Prompt and modified in this code editor. The implementation part of the project mainly takes place in this software.
- Google Chrome
 - Google Chrome is used to preview the output of the mobile application from the local host and set up the environment for the Firebase. When Ionic is serving the services, the browser will pop up the output of the project. Jasmine, the testing framework that requires Karma, the test runner also controls Google Chrome as a tool to show the actual test results.
- Android Studio
 - Android Studio is used to run or emulate the mobile application on the Android platform. Android Virtual Device Manager from this software is used only as it can show the output of the project in an Android virtual mobile phone after the command has been executed.
- Xcode
 - Xcode is used to run or emulate the mobile application on the iOS/Apple platform. Similar to the purpose of Android Studio, this software will show the output of the project in an iOS virtual mobile phone after the command has been executed.
- Microsoft Project
 - Microsoft Project is used to prepare the Work Breakdown Structure and the Gantt Chart. The milestones and the tasks are inserted one-by-one into the WBS and each of them will be scheduled wisely. The Gantt Chart will be generated automatically when the WBS has been updated.

- Axure RP 8
 - Axure RP 8 acted as the wireframing tool that is used to make the high-fidelity functional prototype. The interactions between the user and the system will be prepared in this software in order to show the features of the completed application with the prototype.
- Enterprise Architect
 - Enterprise Architect is used to draw the Use Case Diagram, Activity Diagrams, Class Diagram, Entity Relationship Diagram and Data Flow Diagrams. All of the UML Diagrams are prepared by using this modeling software except for physical ERD that is drawn on a website called 'draw.io'.

Programming and Markup Languages:

- HTML
 - HTML is a markup language that is used to create a web based mobile application. It is mainly implemented in the views of the Ionic project.
- SCSS
 - SCSS is a style sheet language with the extension of CSS preprocessor which is used to design the user interface of the application. It can only be implemented in the views and style sheets of the Ionic project.
- TypeScript
 - TypeScript is the superset of JavaScript that is used with the Angular framework. It is mostly implemented in the routes and other modules of the Ionic project.

Frameworks and Services:

- Ionic
 - Ionic is a hybrid mobile application framework which is good to build a cross-platform mobile application with the help of Apache Cordova and Angular frameworks. It used HTML, SCSS and TypeScript languages to develop the application.
- Apache Cordova

- Apache Cordova acted as a translator or converter to translate or convert the HTML, SCSS, TypeScript to other programming languages that suit Android and iOS platforms. For Android, it will convert these languages to JavaScript and Java. As for iOS, it will convert these languages to Objective-C and Swift.
- Angular
 - Angular is a web application framework which provides many web components that allow these components to be used in the development of mobile applications. For example, components like Platforms, Modules, Providers and so on.
- Google Firebase
 - Google Firebase provides a real-time database and back-end as a service. It is required to store the real-time data like the current time, date and location of the user. In order to use this service, a Firebase project is needed to setup externally and integrate it into the base project.
- SQLite
 - SQLite is a DBMS that is used to save the settings of the application to the user's phone locally. Thus, settings will not be synced with the user account but it allows the user to use back the settings that are stored inside the mobile phone locally.
- GitHub
 - GitHub is used as the backup tool that offers web hosting, so it can prevent accidental deletion of the Smart Location Alarm source code. The edited files can be committed to the GitHub server with different branches or merged with the master branch once it is ready.
- Google Play Store
 - Google Play Store is the place where most of the Android users download their applications from. It is the deployment platform chosen for this project, that allows Smart Location Alarm to be published and downloaded by the public users.

CHAPTER 4

PROJECT INITIAL SPECIFICATION

4.1 Introduction

This chapter covers the analysis of the online survey, states the project requirements and also shows the Use Case Diagram, Use Case Descriptions, Activity Diagrams, conceptual design such as logical ERD and logical DFD as well as prototype design in both low-fidelity storyboard or screen sketches and high-fidelity functional prototype screen captures.

4.2 Fact Findings

Fact findings or the analysis of the requirement gathering technique, online survey will be described in the following subsection.

4.2.1 Online Survey Analysis

After collecting the responses from 40 respondents, analysis of the online survey will be discussed in this subsection. For the last question of the online survey, some of the responses will be selected as the project functional requirements and non-functional requirements directly.

Are you a student, a working adult or a senior citizen?

40 responses

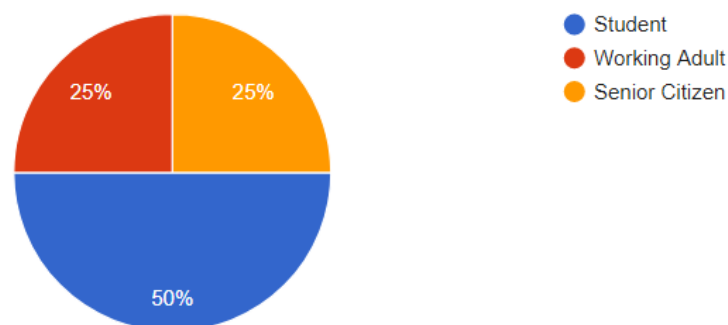


Figure 4.1: Different Group of Target Users

Figure 4.1 presents the different group of target users who participated in this online survey. Three main groups including students, working adults and senior citizens are the respondents and target users for this project. A total of 40 respondents participated in this online survey. Half of them are represented by the students and another half represented by 10 working adults and 10 senior citizens. This is because different opinions can be collected from people with different life stages and this mobile application is suitable for almost everyone.

Which mobile platform are you using now?

40 responses

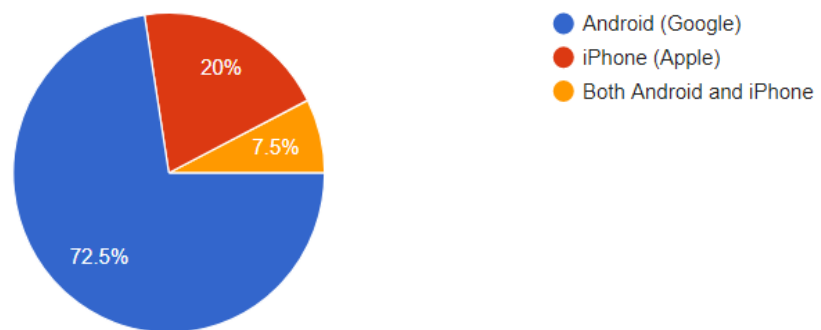


Figure 4.2: Mobile Platforms that is Widely Used

According to Figure 4.2, different mobile platforms are used by the target users and the majority of them are using Android. iPhone has a portion of 20% from the 40 respondents which is 8 of them. The minority of the users have both Android and iPhone, thus a cross-platform reminder application should be implemented to achieve data synchronization. However, there is no respondent selecting other mobile platforms like Windows, hence this project will emphasize in developing the application on Android and iPhone mobile platforms.

Have you ever used a reminder application?

40 responses

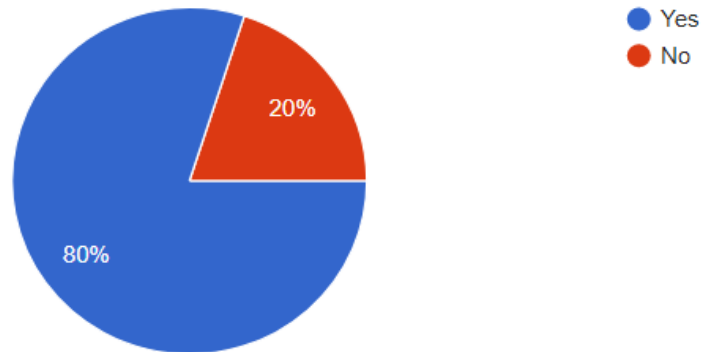


Figure 4.3: Usage of Reminder Application

Figure 4.3 displays the usage of reminder applications in these 40 respondents. 32 people from the 40 respondents stated that they have the experience of using a reminder application. This result tells that a reminder application has a huge demand for people to use it as it serves many useful features in a convenient way. On the other hand, 8 of them might be using traditional handwritten to-do lists or they are good at remembering.

If yes, why would you like to download and use the reminder application?

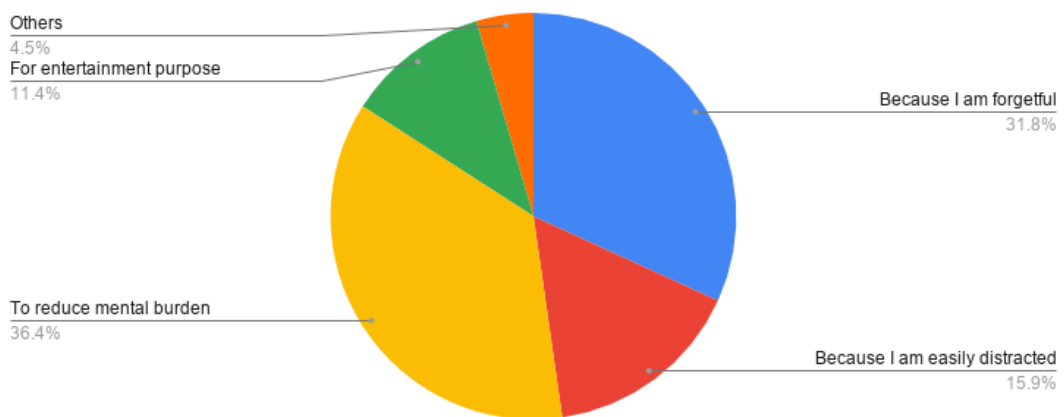


Figure 4.4: Reasons of Using Reminder Application

From the 32 respondents who have used a reminder application before, there are some main reasons for them to use it based on Figure 4.4. Most of them stated that they want to reduce the mental burden followed by owing to the fact that they are forgetful. There are other reasons like to remind them about the important events and dates. Thus, a reminder application helps in providing many features for the target users to solve their problems and meet their needs.

How often do you forget things?

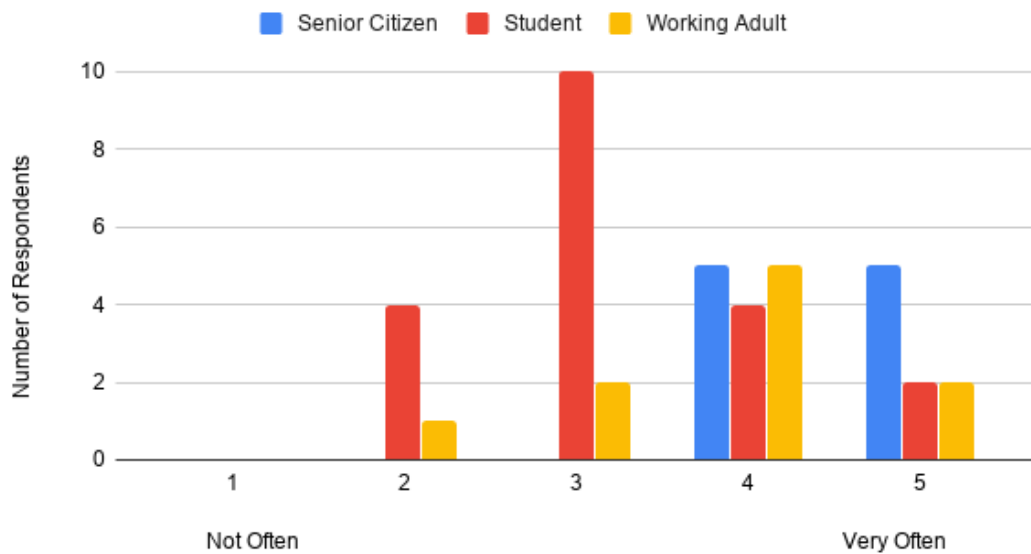


Figure 4.5: Frequency of Forgetfulness from Different Target Users

Figure 4.5 shows the frequency of forgetfulness from different target users. Most of the students seldom forget things. This is because students usually have a better memory compared with working adults and senior citizens who are older. This can be explained as working adults and senior citizens have greater proportions in always forgetting things. However, the frequency of forgetfulness also depends on the different strength of human memory.

To what extent do you agree that it is necessary for every student, working adult and senior citizen to use a reminder application?

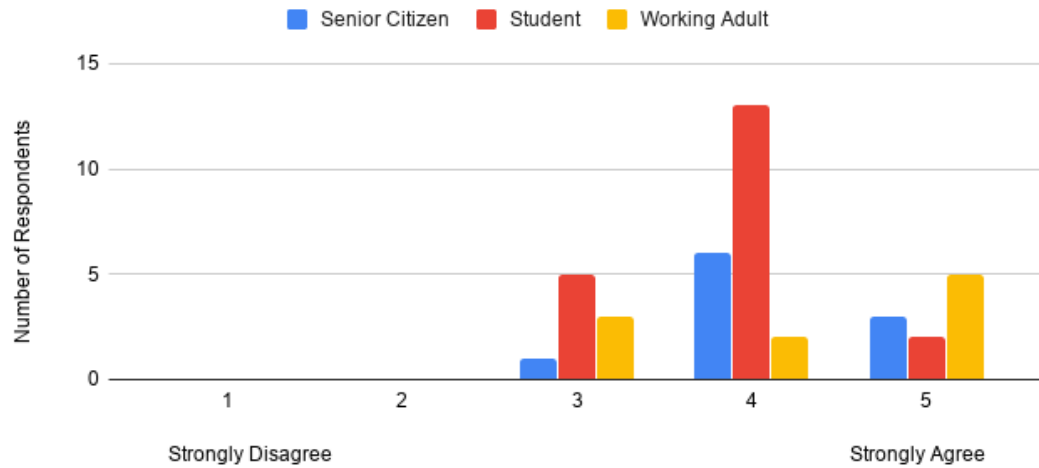


Figure 4.6: Importance of Using Reminder Application

Figure 4.6 illustrates the importance of using a reminder application from different target users. None of the respondents disagrees on the importance of using a reminder application for every student, working adult and senior citizen. In contrast, 41 respondents agree on the necessity of using a reminder application. On the other hand, 9 of them imply that the choice of whether to use a reminder application is depending on the people's needs.

Which function do you think is the most important for a reminder application?

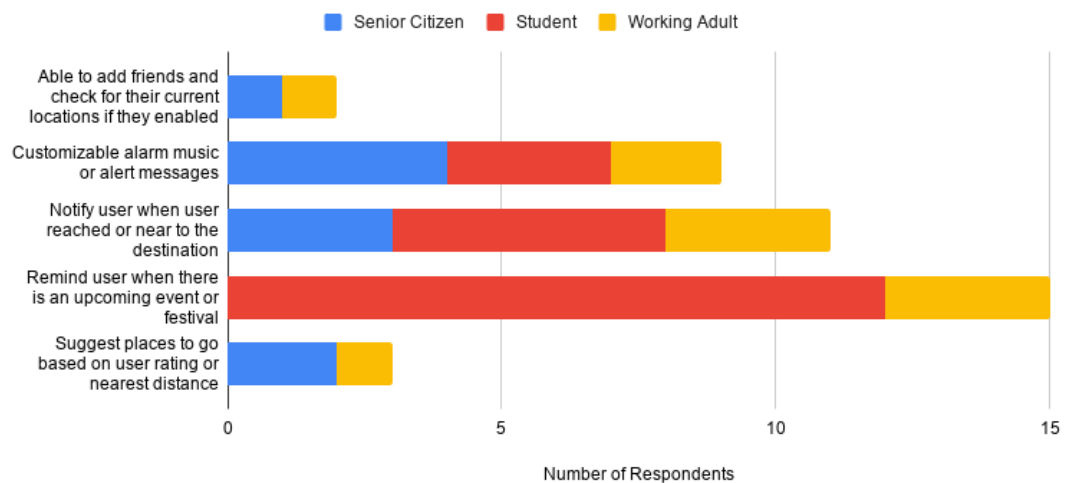


Figure 4.7: Criticality of Each Reminder Application Feature

According to Figure 4.7, features of reminder application are presented with the levels of criticality. More than half of the 20 students indicated that reminding about upcoming events or festivals is the most important feature for a reminder application. Besides, notification about the preset location and alarm customization features are also quite important to be included into a reminder application. Therefore, these 3 features are mandatory in this project.

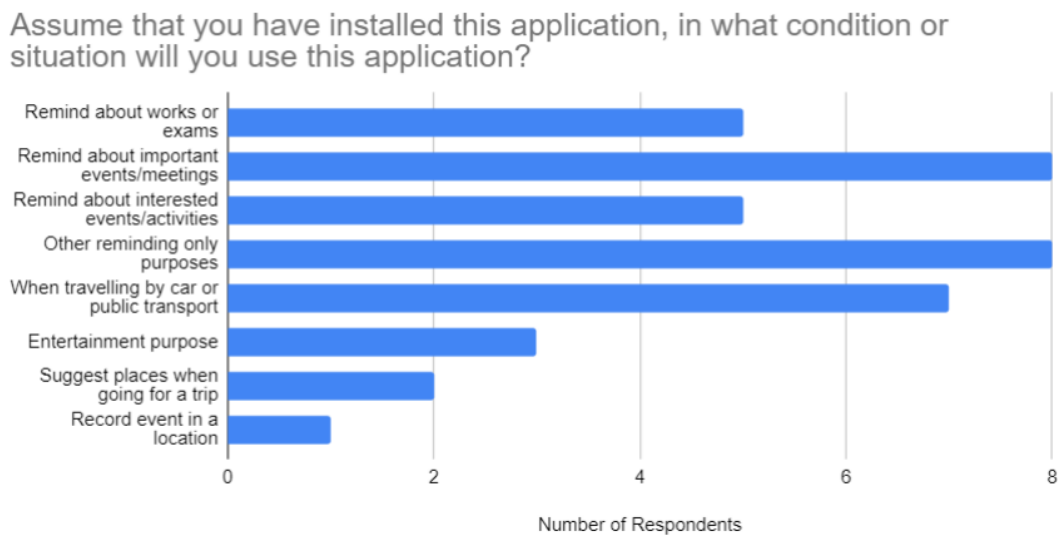


Figure 4.8: Conditions to use Smart Location Alarm System

Figure 4.8 shows the conditions to use Smart Location Alarm mobile application from the respondents' assumption. Majority of the respondents will use it for the purpose of reminding only. 5 of them stated they will use it for reminding works, exams or assignments whereas another 5 respondents will use it for reminding events or activities that they are interested in. Besides, 8 of the respondents are planning to use it for important events or meetings which cannot be missed. Another 8 respondents also stated they will use it to alert them with notification messages about cooking time, things they want to buy and any other conditions that they might forget.

Other than that, 7 of the respondents stated they will use the mobile application when going to unfamiliar places by driving or sleeping in the public

transport. Therefore, the application can show their current location and route as well as remind the users when they almost reach their destinations. Furthermore, 3 of the respondents will use this system for entertainment purposes by adding friends, relieving boredom.

In addition, 2 of the respondents will use this application when going out for a trip especially outstation to search information about the tourist attractions. One of them stated he will use this application to record the event that will take place in a far location. Hence, this respondent will expect to use the map to set the event's location and also set the date and time to remind him.

4.3 Project Requirements

This section discusses the functional requirements and non-functional requirements of Smart Location Alarm on Mobile Platform project.

4.3.1 Functional Requirements

1. The system shall allow the user to set and update the alarms.
2. The system shall allow the user to enable or disable the alarms.
3. The system shall allow the user to delete the alarms.
4. The system shall allow the user to view the map with information.
5. The system shall allow the user to search for places.
6. The system shall allow the user to change the ringtone music.
7. The system shall allow the user to choose the alarm ringing duration.
8. The system shall allow the user to enable or disable the ringtone vibration.
9. The system shall allow the user to enable or disable do not disturb mode.
10. The system shall allow the user to write the notification message.
11. The system shall allow the user to share location to friends.
12. The system shall allow the user to show or hide their friends' shared location.
13. The system shall allow the user to send location-based alarm notifications to chosen friends.
14. The system shall allow the user to set the location-based alarms to ring on selected weekdays.
15. The system shall allow the user to use some features without an account.

16. The system shall allow the user to login and register an account.
17. The system shall allow the user to logout and reset password.
18. The system shall allow the user to upload profile pictures.
19. The system shall allow the user to edit profile details.
20. The system shall allow the user to search for friends.
21. The system shall allow the user to send or remove friend requests.
22. The system shall allow the user to accept or reject friend requests.
23. The system shall allow the user to unfriend users.

4.3.2 Non-Functional Requirements

1. The time-based alarms shall ring on time.
2. The location foreground tracking and location-based alarms reminding shall be accurate when the user's device has good and stable internet connection.
3. The system shall prevent the unauthorized login attempts with warnings and actions.
4. The notification shall be able to show information like date, time and message for time-based alarm, latitude, longitude and radius for location-based alarm.
5. Location-based alarm notifications from friends will not be received if do not disturb mode is turned on.
6. The system shall send notification to the user if the friend request has been received and the friend request has been accepted.
7. The system shall send location-based alarm notifications to chosen friends when the user is reaching the friends selected marker.
8. The system shall ring the location-based alarms on the selected weekdays only when the user is reaching the weekdays selected marker.
9. The disabled alarms shall not ring, the enabled alarms shall ring if certain criterias have been met.
10. The alarms shall be disabled right after ringing.

4.4 Use Case Diagram

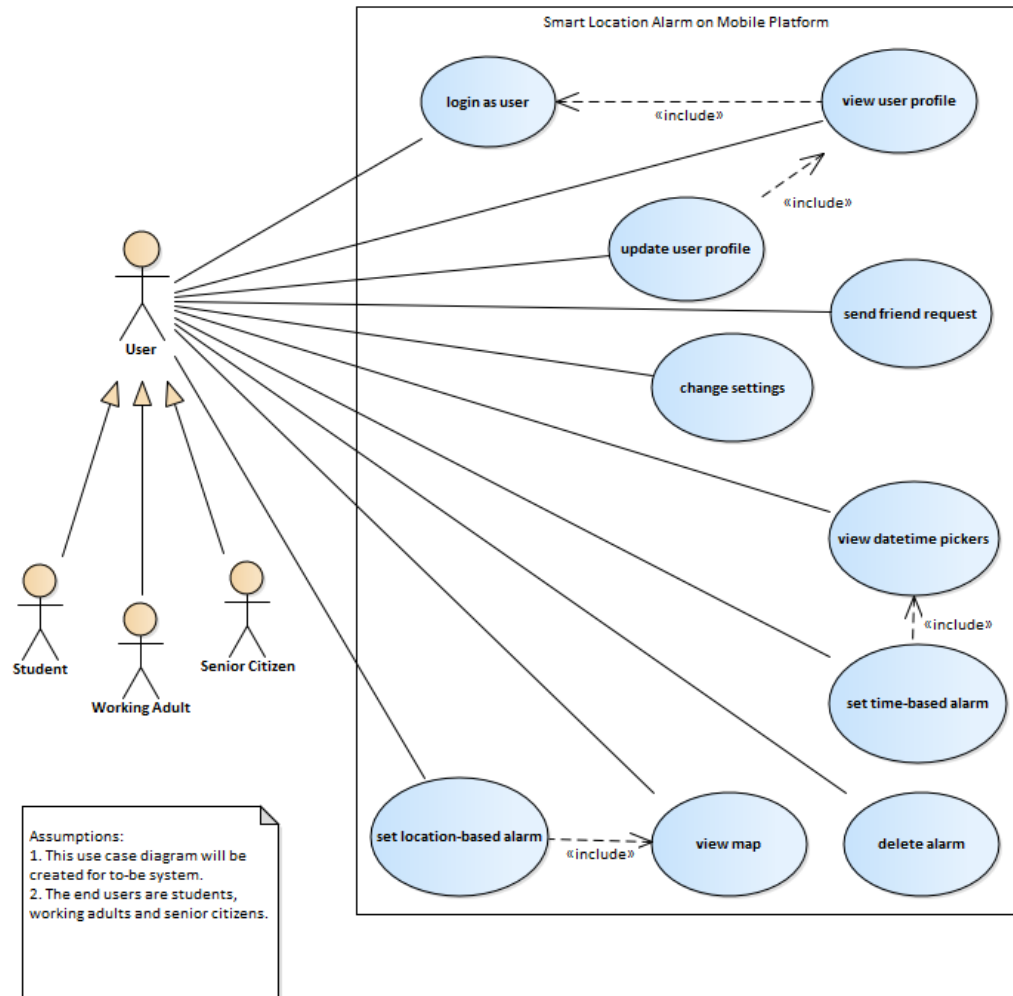


Figure 4.9: Use Case Diagram

4.5 Use Case Descriptions

Use Case ID: 1
Use Case Name: view datetime pickers
Actor: User
Brief Description: This use case describes how a user views the datepicker and timepicker to choose his/her desired date and time.
Trigger: User wants to select the date and time for reminding purpose.
Relationships: Association: User Include: set time-based alarm Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. User sets or edits the time-based alarm. 2. User selects the date input area. 3. System displays the datepicker. 4. User selects the time input area. 5. System displays the timepicker.

Table 4.1: Use Case Description of View DateTime Pickers

Use Case ID: 2
Use Case Name: set time-based alarm
Actor: User
Brief Description: This use case describes how a user sets the alarm with the date and time.
Trigger: User wants to set or update the time-based alarm.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. User selects the date and time. 2. System shows the selected date and time. 3. User enters the notification message. 4. User confirms the time-based alarm. 5. System saves and enables the time-based alarm. 6. System shows the alarm in the alarm list. 7. System displays an alarm added or updated message.
Sub Flows: <ol style="list-style-type: none"> 1.1 User changes the date and time. <ol style="list-style-type: none"> 1.1.1 System shows a picked datepicker. 1.1.2 User chooses a new date. 1.1.3 System shows a picked timepicker. 1.1.4 User chooses a new time. 3.1 User changes the notification message. <ol style="list-style-type: none"> 3.1.1 System shows a filled text area. 3.1.2 User enters a new notification message.

Table 4.2: Use Case Description of Set Time-based Alarm

Use Case ID: 3
Use Case Name: view map
Actor: User
Brief Description: This use case describes how a user views the Google Maps to choose his/her desired location.
Trigger: User wants to select the location for reminding purpose.
Relationships: Association: User Include: set location-based alarm Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. User sets or edits the location-based alarm. 2. System displays Google Maps.
Sub Flows: <ol style="list-style-type: none"> 2.1 User selects to track his/her current location. <ol style="list-style-type: none"> 2.1.1 User chooses to go to my location. 2.1.2 System shows the current location of the user. 2.2 User chooses to search places. <ol style="list-style-type: none"> 2.2.1 User enters a place name. 2.2.2 System displays a list of search results. 2.2.3 User presses one of the search results. 2.2.4 System pans to the searched location on Google Maps. 2.3 User shares his/her location <ol style="list-style-type: none"> 2.3.1 User chooses to share the location to friends. 2.3.2 System updates the user share location. 2.3.1 System displays a share location complete message. 2.4 User views friends' shared locations. <ol style="list-style-type: none"> 2.4.1 User chooses to show friends' shared locations. 2.4.2 System drops the friends' profile picture on Google Maps.

Table 4.3: Use Case Description of View Map

Use Case ID: 4
Use Case Name: set location-based alarm
Actor: User
Brief Description: This use case describes how a user sets the alarm with the location.
Trigger: User wants to set or update the location-based alarm.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. User selects the coordination or location from Google Maps. 2. System shows the information of the coordination or location. 3. User long presses on the desired location. 4. System drops a marker to Google Maps. 5. System saves and enables the location-based alarm. 6. System shows the alarm in the alarm list. 7. System displays an alarm added or updated message.
Sub Flows: <ol style="list-style-type: none"> 3.1 User changes the radius. <ol style="list-style-type: none"> 3.1.1 User presses the alarm. 3.1.2 System shows the information of the alarm. 3.1.3 User enters a new radius. 3.1.4 User applies the new radius. 3.1.5 System changes the marker area based on the radius. 5.1 User chooses friends to notify. <ol style="list-style-type: none"> 5.1.1 User presses the alarm. 5.1.2 System shows the information of the alarm. 5.1.3 User chooses one or more friends. 5.1.4 User confirms the chosen friends. 5.1.5 System saves the friends to notify. 5.2 User selects weekdays to notify. <ol style="list-style-type: none"> 5.2.1 User presses the alarm. 5.2.2 System shows the information of the alarm. 5.2.3 User chooses one or more weekdays. 5.2.4 User confirms the selected weekdays. 5.2.5 System saves the weekdays to notify.

Table 4.4: Use Case Description of Set Location-based Alarm

Use Case ID: 5
Use Case Name: delete alarm
Actor: User
Brief Description: This use case describes how a user deletes the time or location based alarm.
Trigger: User wants to delete the time-based or location-based alarm.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. System shows a list of enabled or disabled alarms. 2. User swipes one of the alarms to the left. 3. User chooses to delete the alarm. 4. System removes the alarm. 5. System displays an alarm deleted message.

Table 4.5: Use Case Description of Delete Alarm

Use Case ID: 6
Use Case Name: login as user
Actor: User
Brief Description: This use case describes how a user login or sign up for an account.
Trigger: User wants to login or sign up for an account.
Relationships: Association: User Include: view user profile Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. System displays a reactive form. 2. User enters an email and password. 3. User signs in to the system with the filled email and password. 4. System shows the user profile picture or default profile picture. 5. System grants the features of location sharing, updating user profiles and so on.
Sub Flows: <ol style="list-style-type: none"> 1.1 User registers for a new account. <ol style="list-style-type: none"> 1.1.1 User enters an email and password. 1.1.2 User enters his/her personal details. 1.1.3 User confirms the filled details. 1.1.4 System records and creates the new user profile. 1.1.5 System helps the user to sign in automatically. 1.1.6 System shows an account registration successful message. 1.1.7 Perform Normal Flow 4 onwards. 3.1 System authentication failed. <ol style="list-style-type: none"> 3.1.1 Perform Normal Flow 1 onwards with error messages.

Table 4.6: Use Case Description of Login As User

Use Case ID: 7
Use Case Name: view user profile
Actor: User
Brief Description: This use case describes how a user views the user profile.
Trigger: User wants to view his/her own or friend's user profile.
Relationships: Association: User Include: update user profile Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. User selects the friends. 2. System prompts a list of friends. 3. User chooses to view the user profile. 4. System displays the information of the user profile.
Sub Flows: <ol style="list-style-type: none"> 2.1 User chooses to search friends. <ol style="list-style-type: none"> 2.1.1 User enters a username. 2.1.2 System displays a list of search results. 2.2 User views the friend requesters. <ol style="list-style-type: none"> 2.2.1 User receives friend requests from other users. 2.2.2 System shows a list of friend requesters.

Table 4.7: Use Case Description of View User Profile

Use Case ID: 8
Use Case Name: update user profile
Actor: User
Brief Description: This use case describes how a user updates the user profile.
Trigger: User wants to update his/her own user profile.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. System shows the information of the user profile. 2. User edits the information that he or she wants to change. 3. User confirms the new information of the user profile. 4. System records the new information of the user profile.
Sub Flows: <ol style="list-style-type: none"> 1.1 User uploads a new profile picture. <ol style="list-style-type: none"> 1.1.1 User selects a new image from the file directory. 1.1.2 System shows a loading bar for the upload process. 1.1.3 System replaces the old profile picture with the new one. 1.1.4 System shows an upload image complete message. 1.2 User alters his/her own account password. <ol style="list-style-type: none"> 1.2.1 User chooses to reset password. 1.2.2 System sends a reset password email to the user's mailbox. 1.2.3 User enters a new password. 1.2.4 System saves the new password.

Table 4.8: Use Case Description of Update User Profile

Use Case ID: 9
Use Case Name: send friend request
Actor: User
Brief Description: This use case describes how the friends system works.
Trigger: User wants to send, remove, accept, reject a friend request or unfriend.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. System shows the information of the user profile. 2. When both users are not friends, the user sends a friend request. 3. System records the friend request. 4. System displays a friend request sent message.
Alternate/Exceptional Flows: <ol style="list-style-type: none"> 2a. When the friend request has been sent, the user removes the friend request. 3a. System deletes the friend request. 4a. System displays a friend request removed message. 2b. When the friend request has been received, the user accepts the friend request. 3b. System deletes the friend request and records both users as friends. 4b. System displays a friend request accepted message. 2c. When the friend request has been received, the user rejects the friend request. 3c. System deletes the friend request. 4c. System displays a friend request rejected message. 2d. When both users are friends, the user unfriends the other user. 3d. System shows an unfriend alert message. 4d. User confirms the unfriend process. 5d. System removes both users from friends. 6d. System displays an unfriend complete message.

Table 4.9: Use Case Description of Send Friend Request

Use Case ID: 10
Use Case Name: change settings
Actor: User
Brief Description: This use case describes how a user changes the settings.
Trigger: User wants to change the application settings.
Relationships: Association: User Include: n/a Extend: n/a Generalization: Student, Working Adult, Senior Citizen
Normal Flow of Events: <ol style="list-style-type: none"> 1. System shows the default settings. 2. User selects a music file as the alarm ringtone. 3. User chooses a ringtone duration. 4. User enables or disables the ringtone vibration. 5. User enables or disables the do not disturb mode. 6. User confirms the settings. 7. System saves the settings to the phone device. 8. System shows a settings updated message.

Table 4.10: Use Case Description of Change Settings

4.6 Activity Diagrams

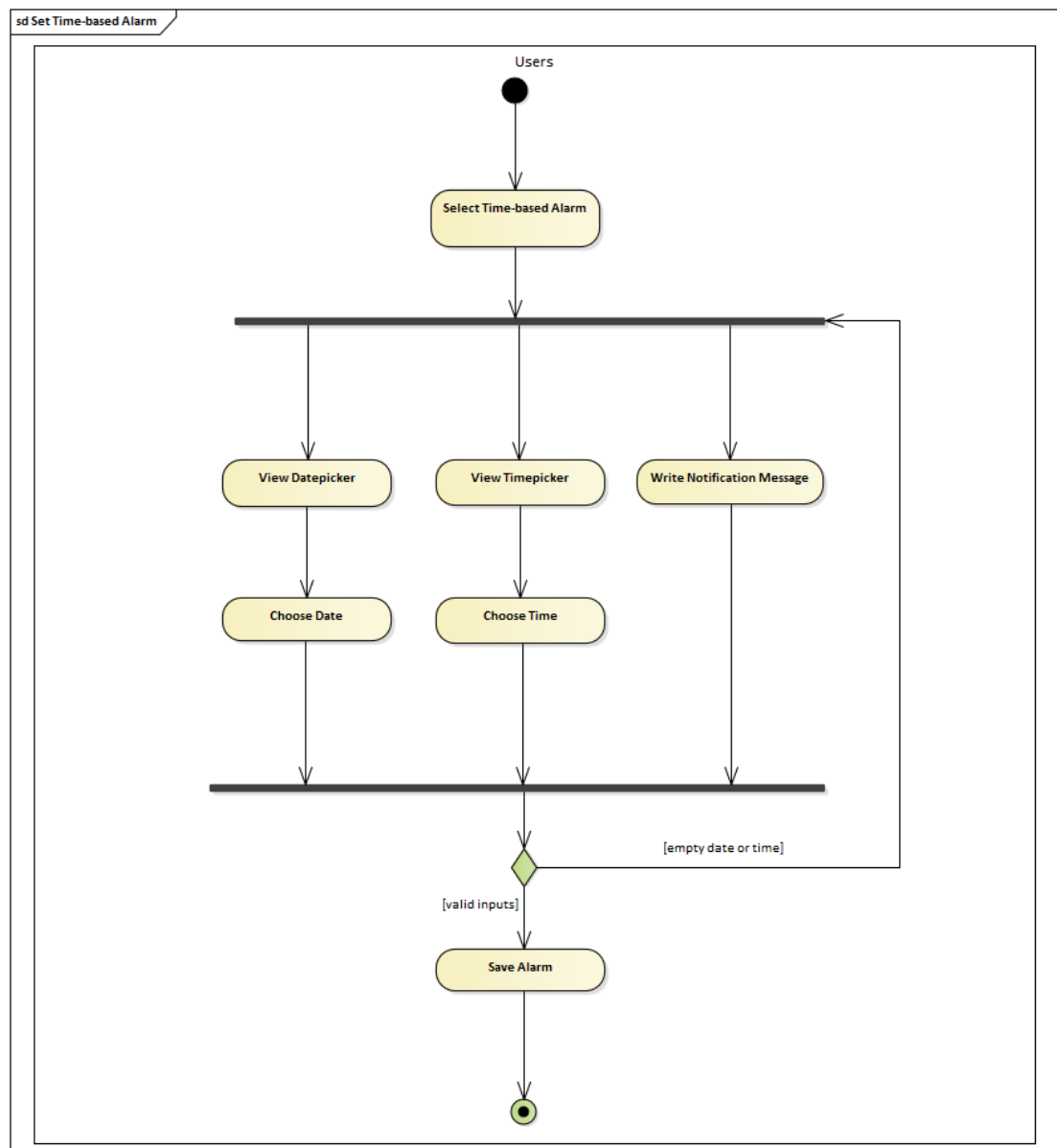


Figure 4.10: Activity Diagram for Set Time-based Alarm

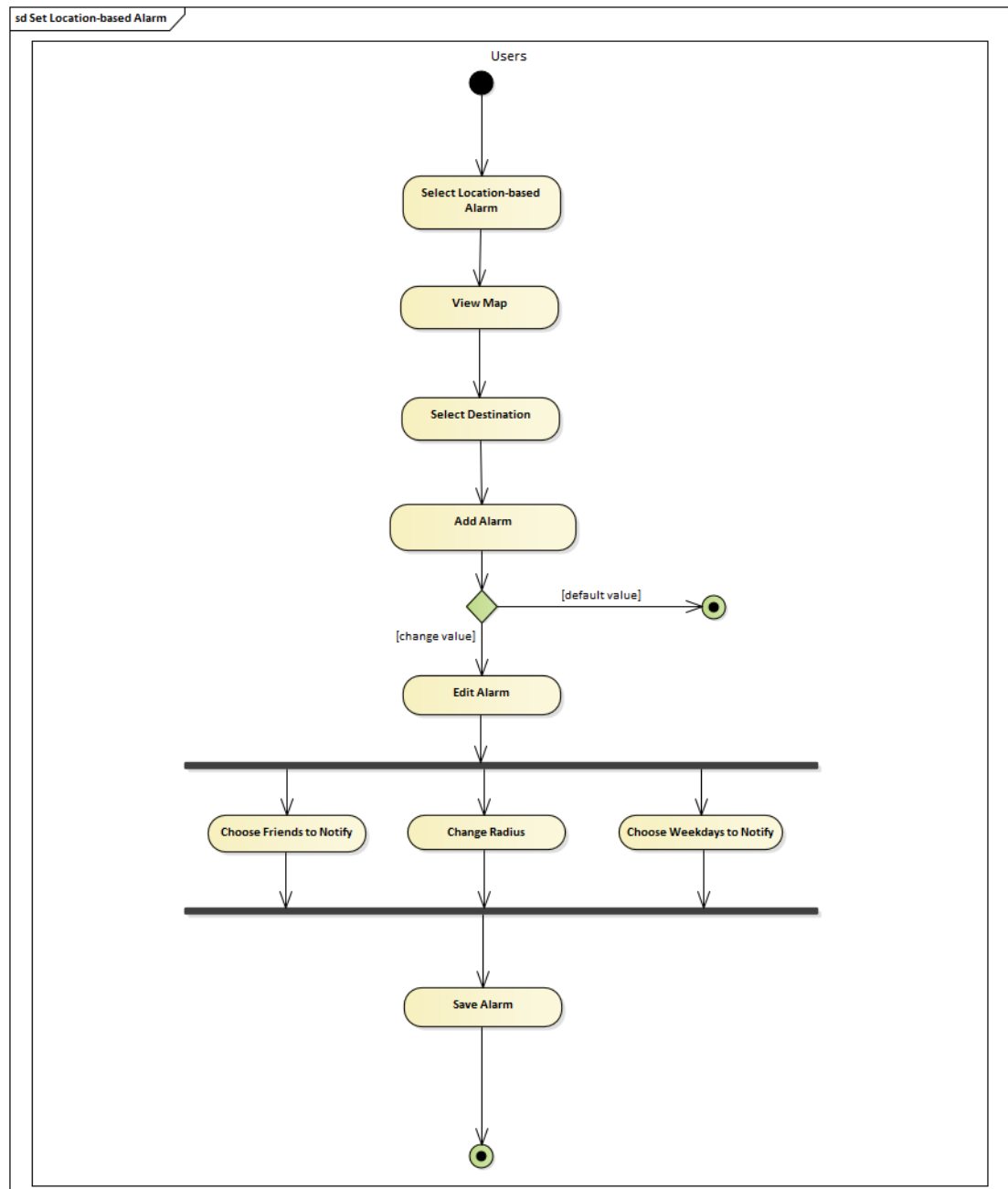


Figure 4.11: Activity Diagram for Set Location-based Alarm

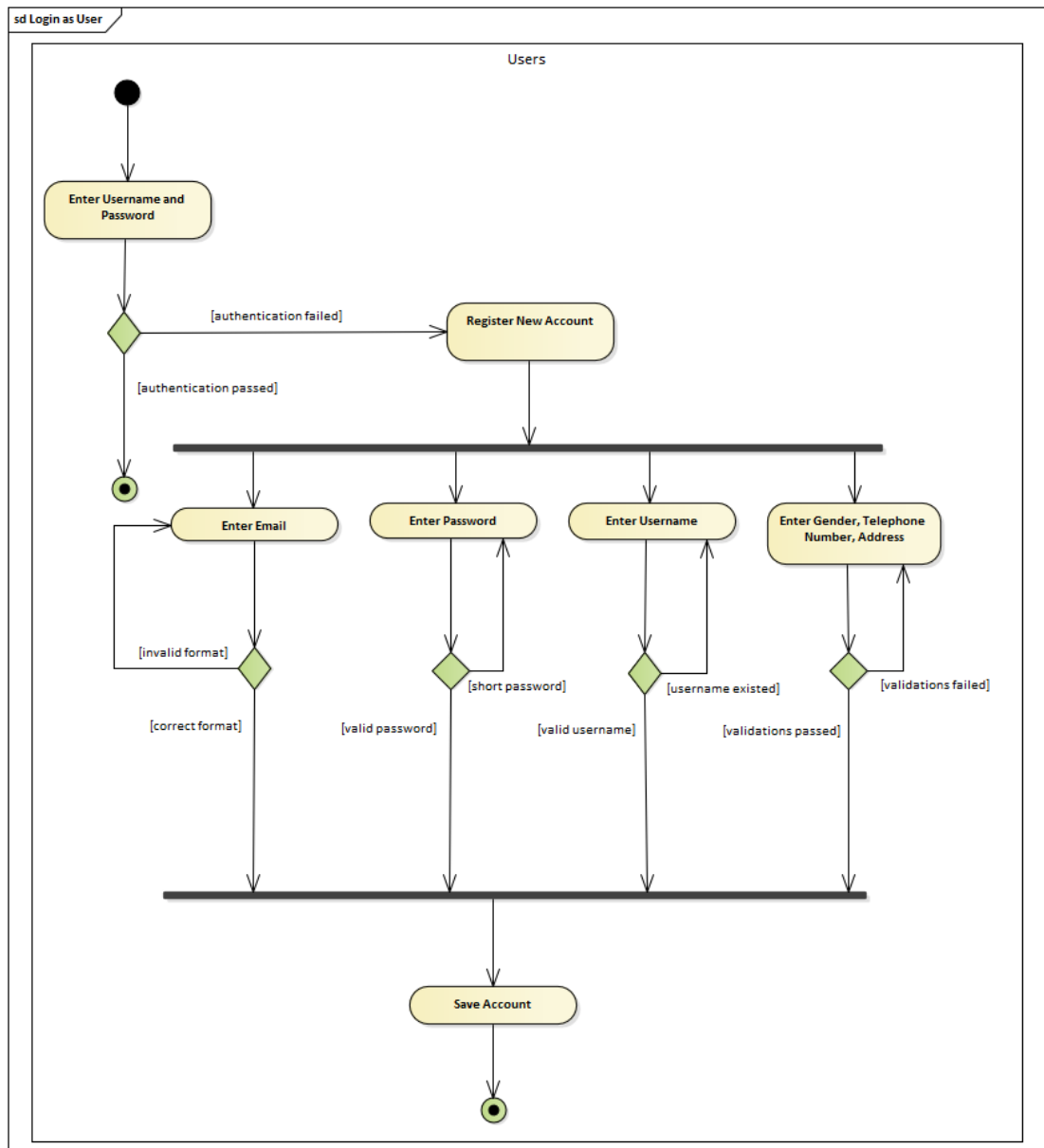


Figure 4.12: Activity Diagram for Login as User

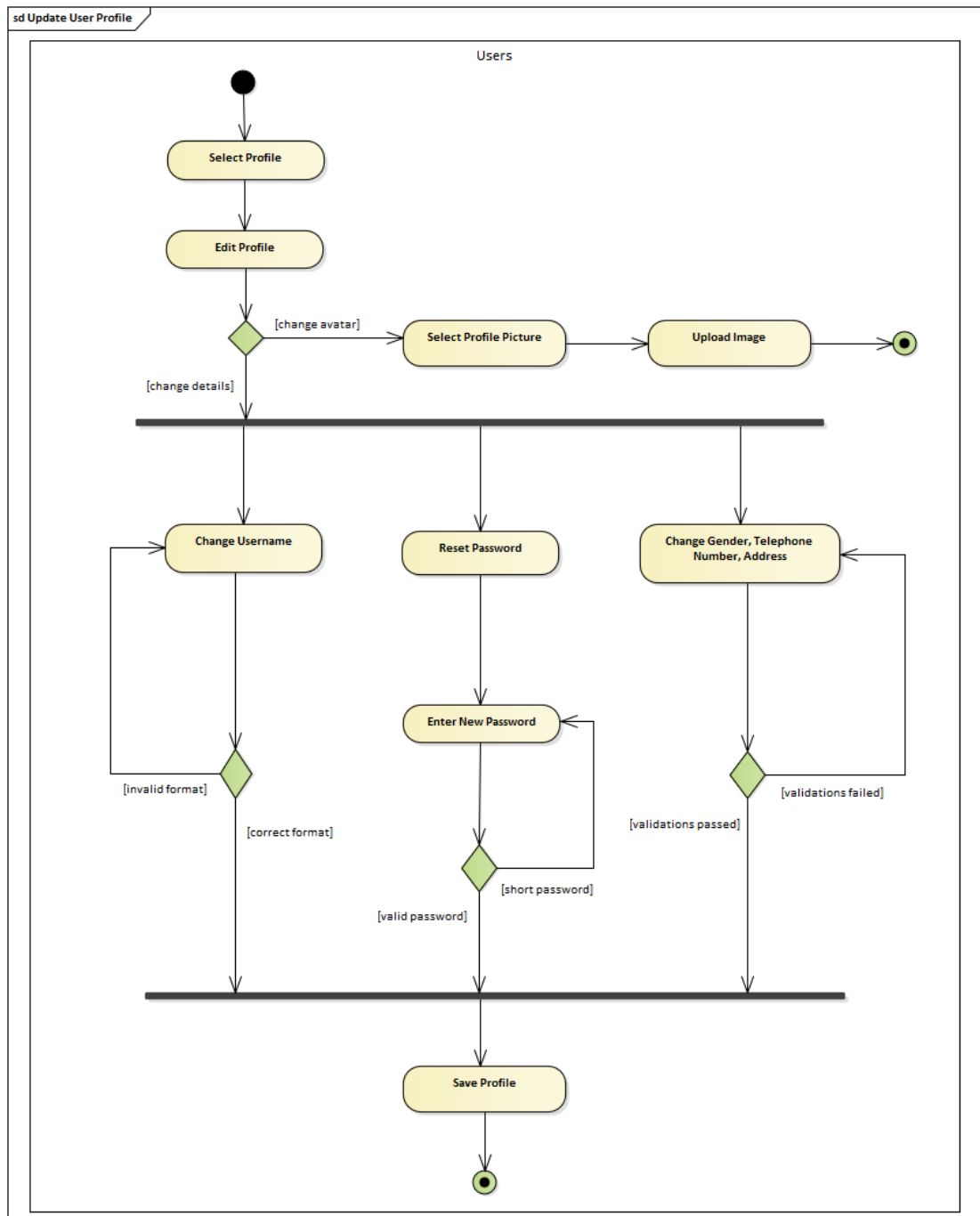


Figure 4.13: Activity Diagram for Update User Profile

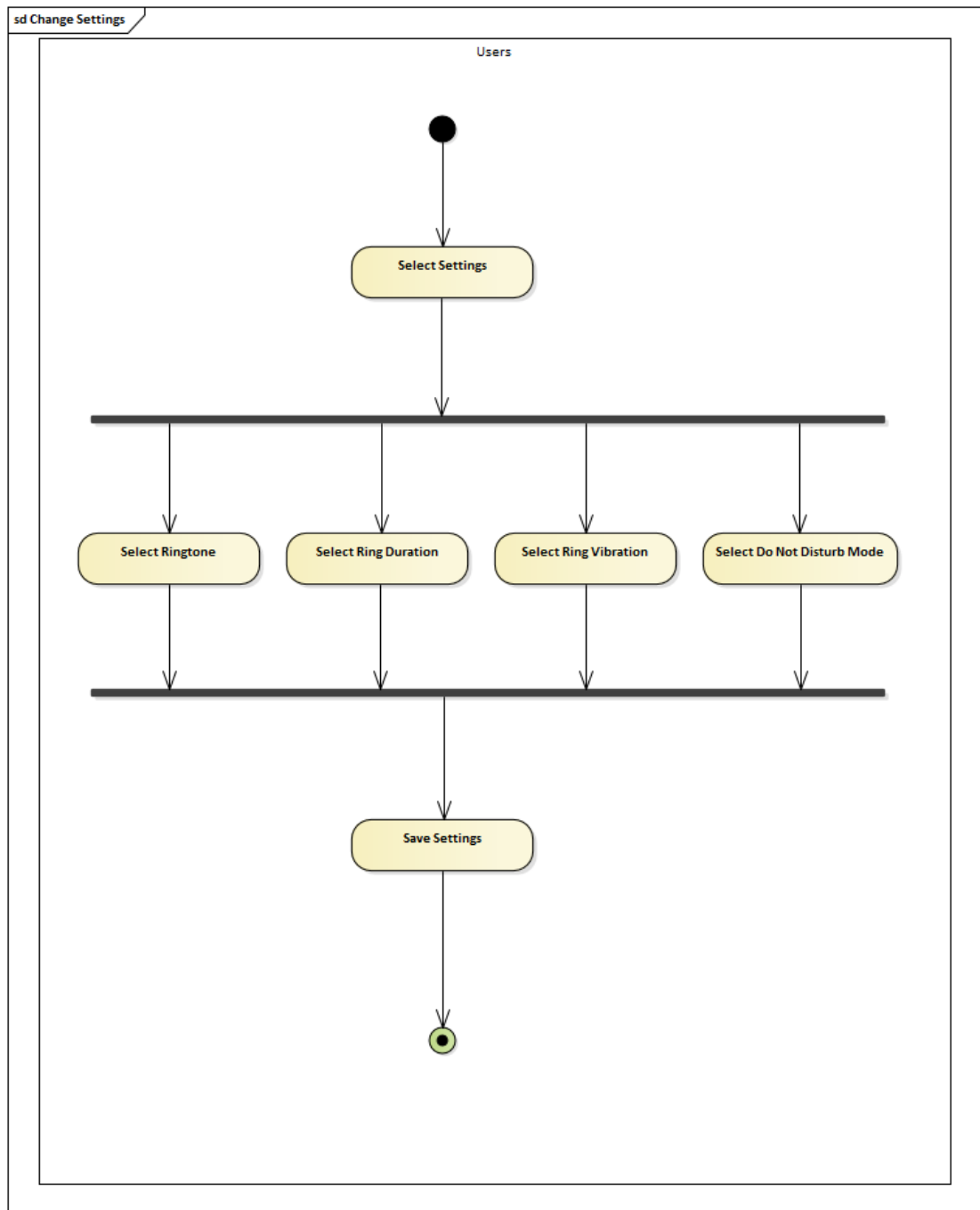


Figure 4.14: Activity Diagram for Change Settings

4.7 Conceptual Design

This section shows the logical ERD and logical DFD which acted as an overview for each diagram. These logical diagrams will help in developing physical ERD and physical DFD.

4.7.1 Logical Entity Relationship Diagram

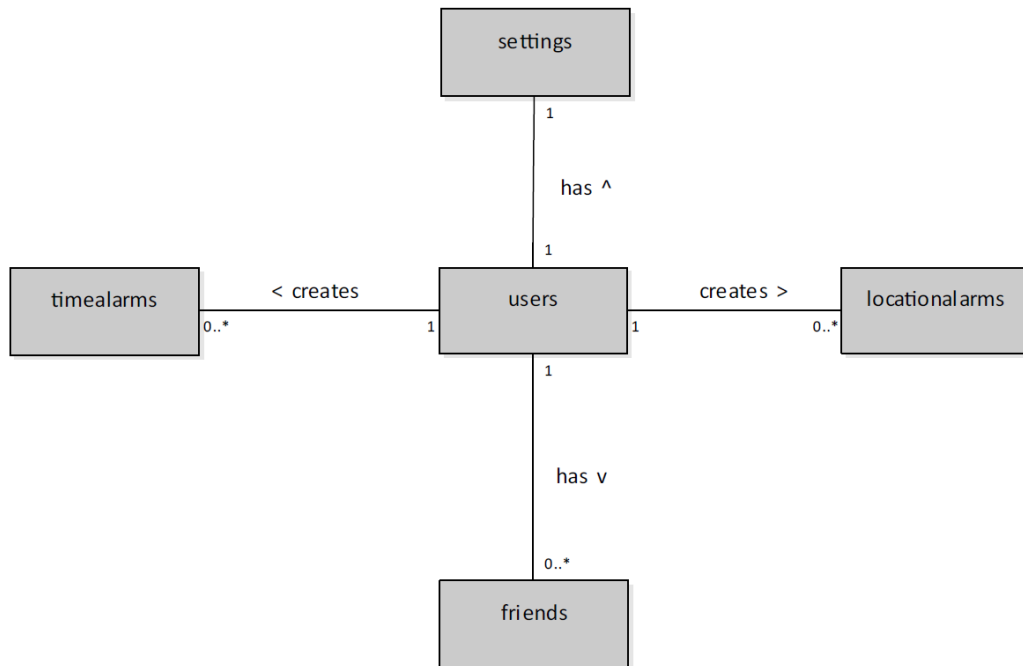


Figure 4.15: Logical Entity Relationship Diagram

Tables	Attributes
users	id, email, password, username, gender, telephone number, address, profile picture, share latitude and share longitude.
timealarms	id, date, time and message.
locationalarms	id, latitude, longitude and radius.
settings	id, ringtone, duration and vibration.
friends	id and username.

Table 4.11: Logical ERD Description

4.7.2 Logical Data Flow Diagram

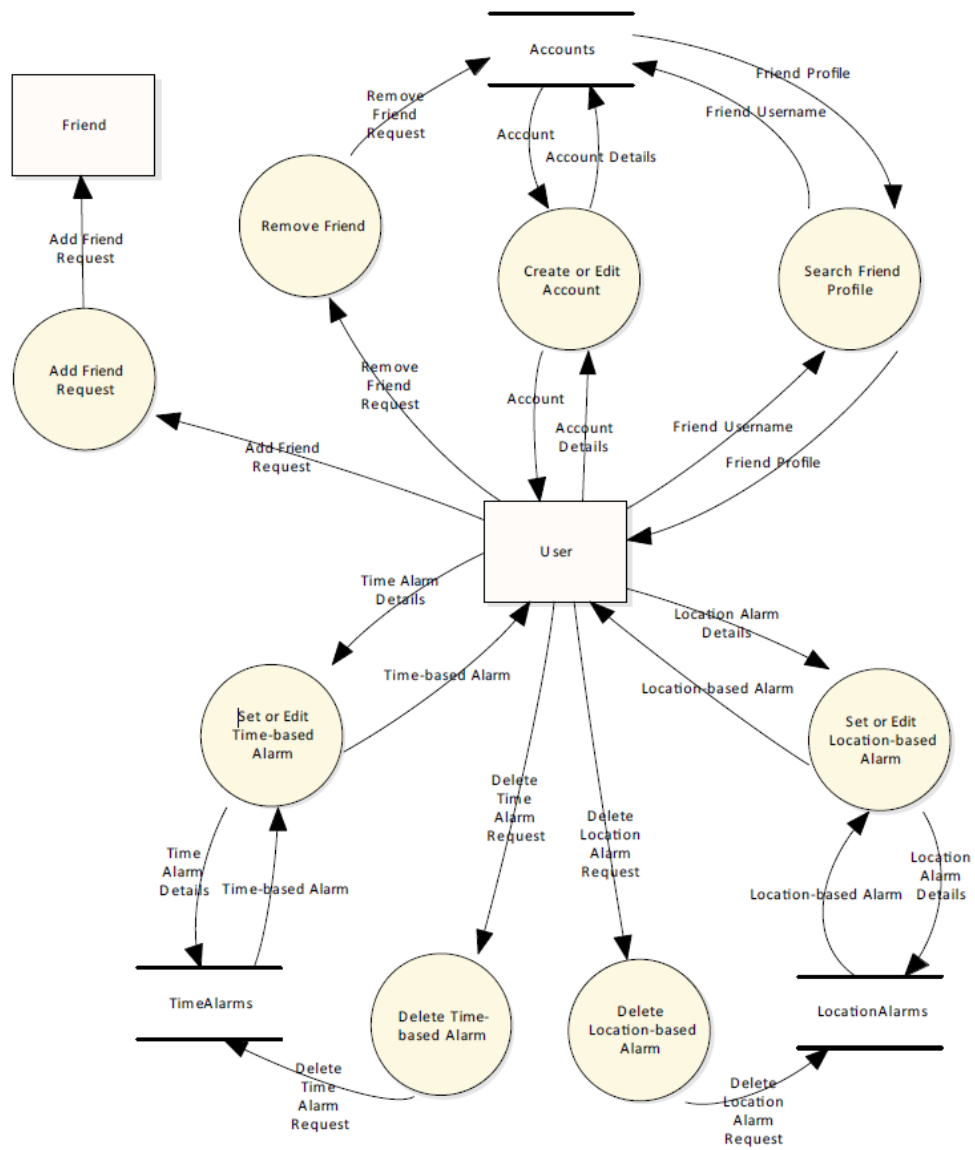


Figure 4.16: Logical Data Flow Diagram

4.8 Prototype Design

This section will display the user interface design of the application in both low fidelity prototype screen sketches and high fidelity prototype screen captures. The functional prototype will be demonstrated during the presentation session.

4.8.1 Low Fidelity Prototype

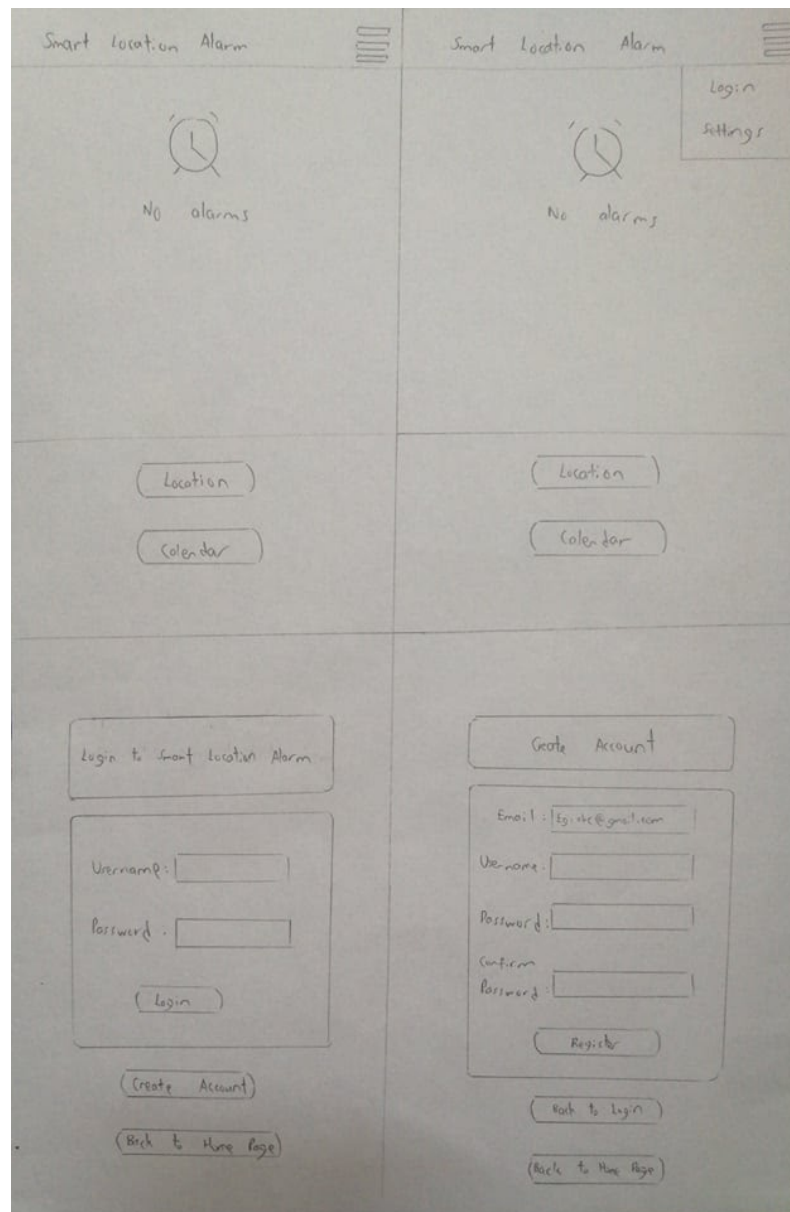


Figure 4.17: Low Fidelity Prototype Part 1

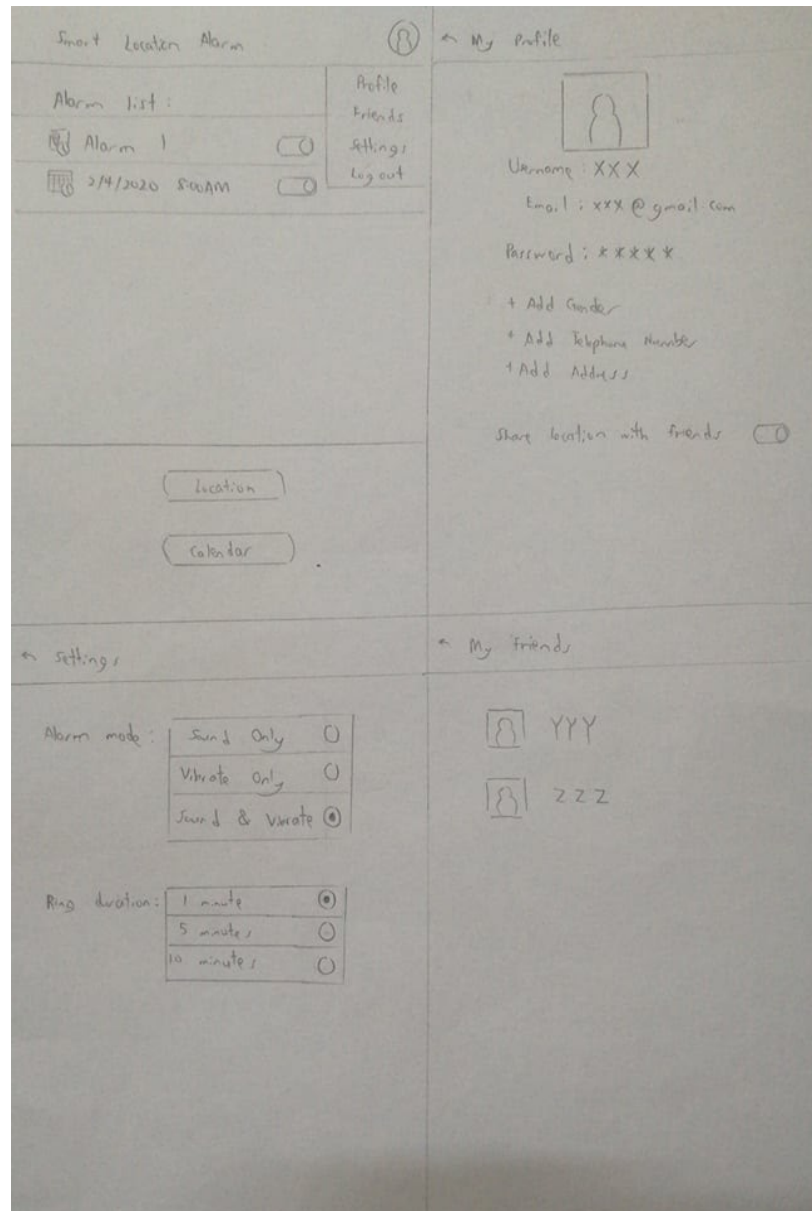


Figure 4.18: Low Fidelity Prototype Part 2

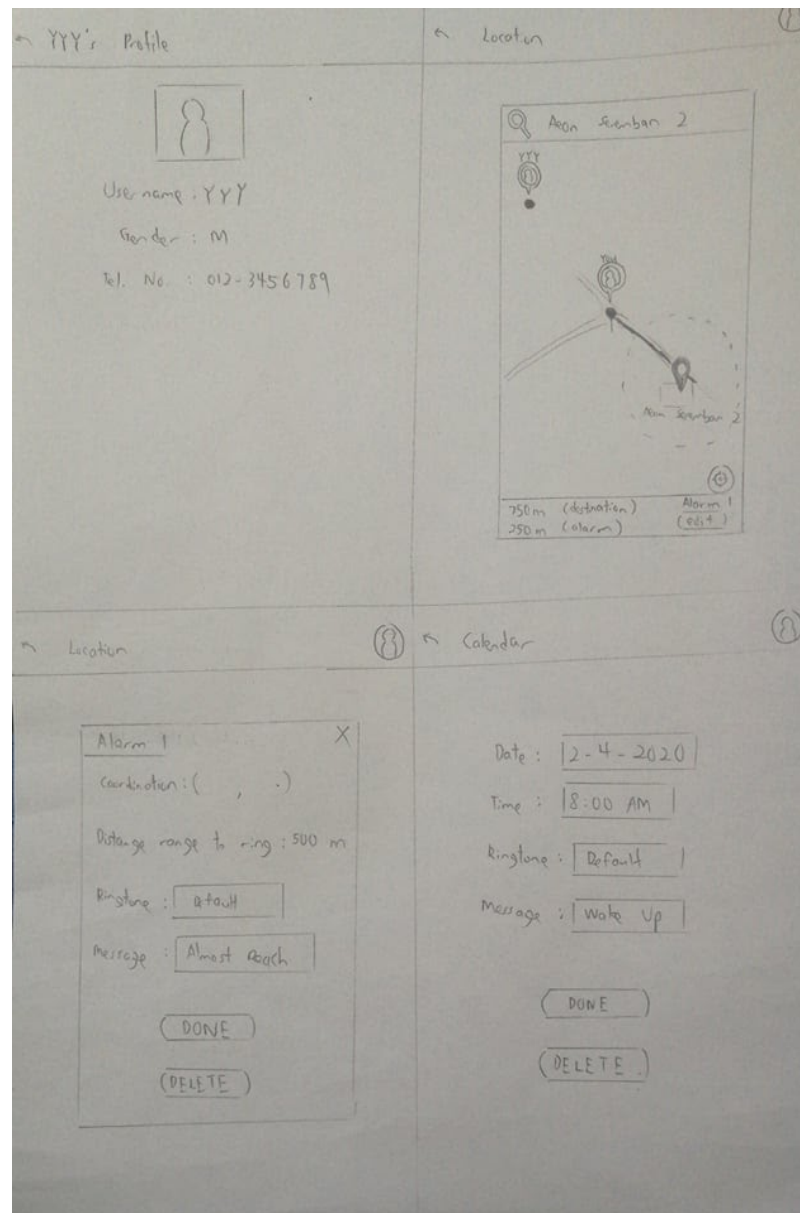


Figure 4.19: Low Fidelity Prototype Part 3

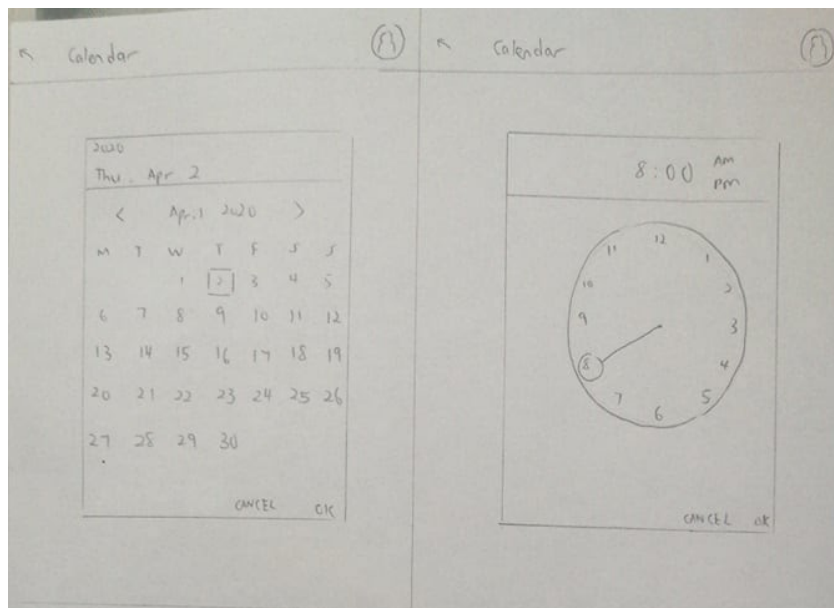


Figure 4.20: Low Fidelity Prototype Part 4

4.8.2 High Fidelity Prototype

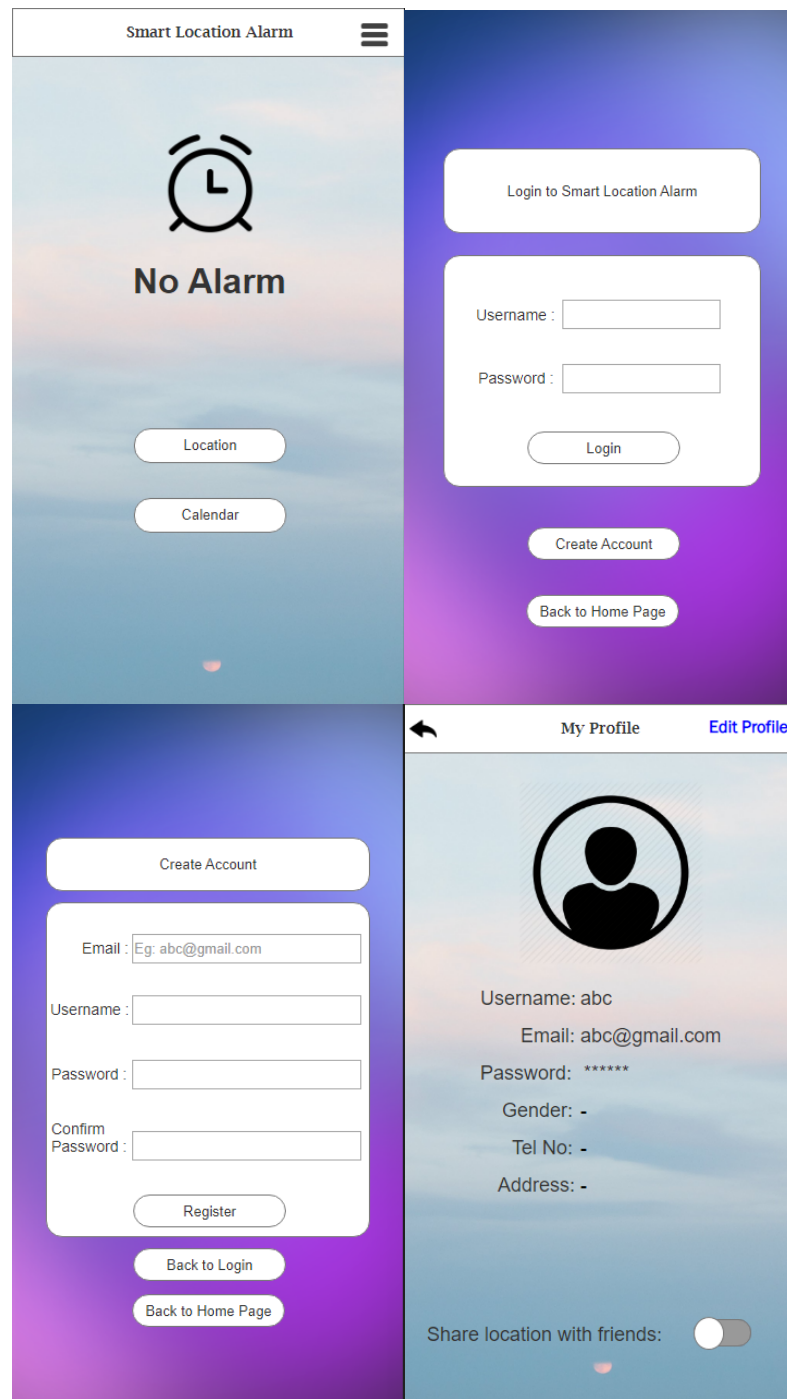


Figure 4.21: High-Fidelity Prototype Part 1

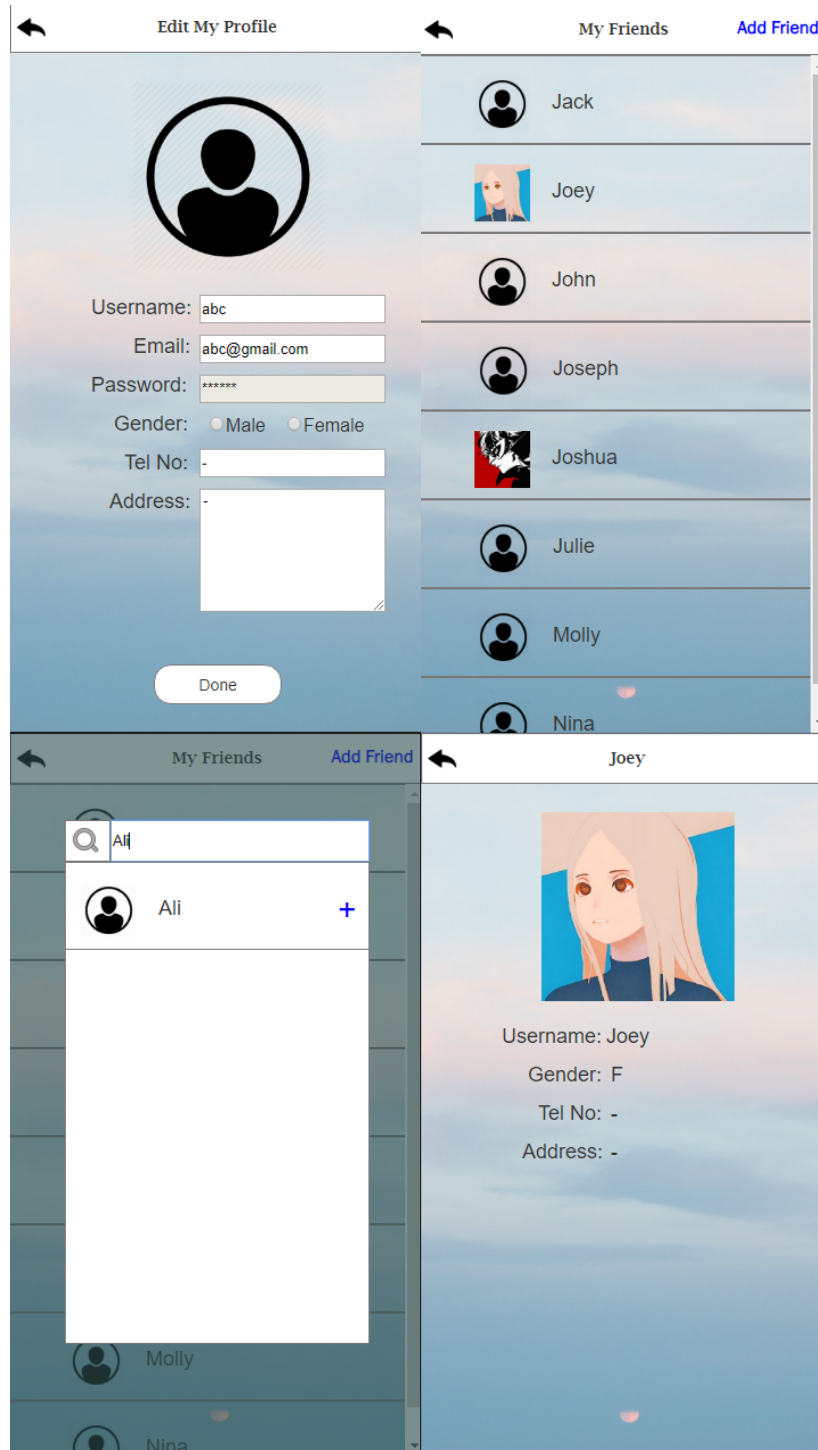


Figure 4.22: High-Fidelity Prototype Part 2

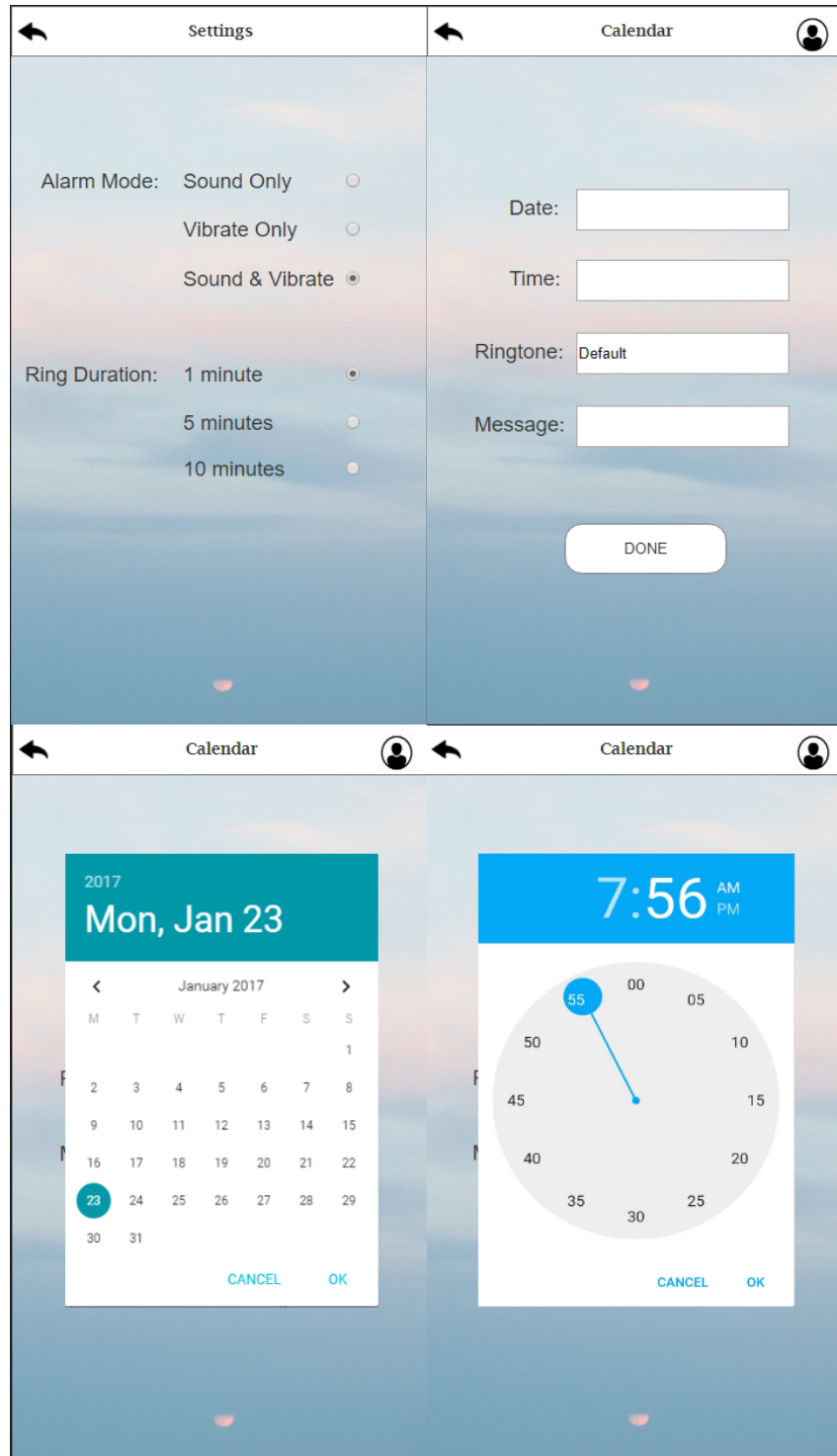


Figure 4.23: High-Fidelity Prototype Part 3

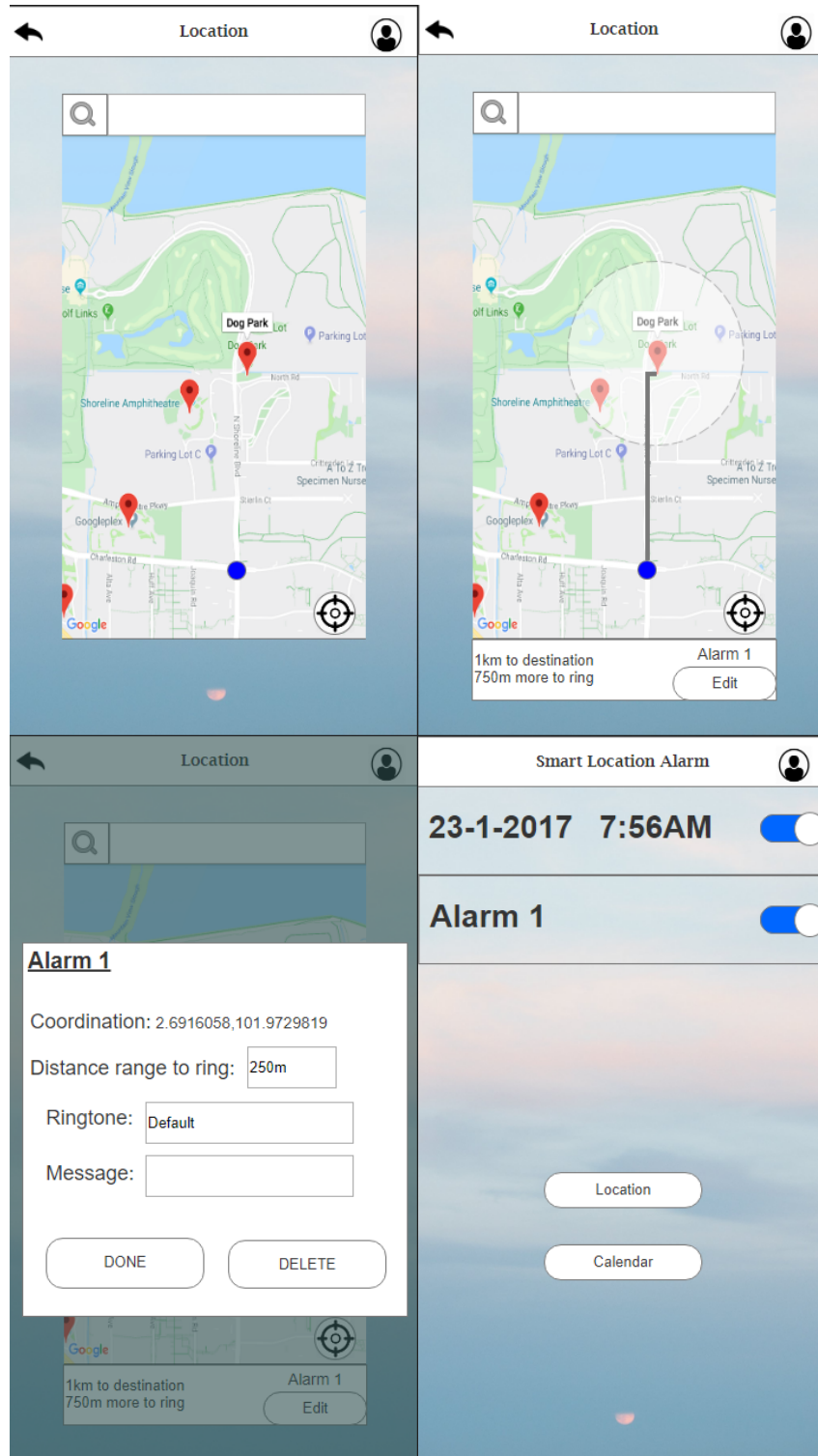


Figure 4.24: High-Fidelity Prototype Part 4

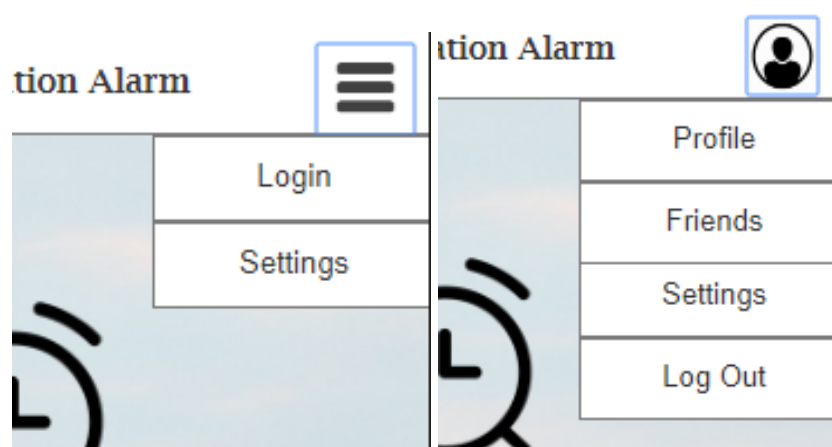


Figure 4.25: High Fidelity Prototype Part 5

CHAPTER 5

SYSTEM DESIGN

5.1 Introduction

This chapter comprises the actual user interface design of the Smart Location Alarm application, Class Diagram, physical ERD as well as physical DFD.

5.2 User Interface Design

This section shows the user interface of every page from the application. They are mostly made from Ionic UI components, HTML elements with styling of SCSS and also assets like background image and custom icons.

5.2.1 Home Page

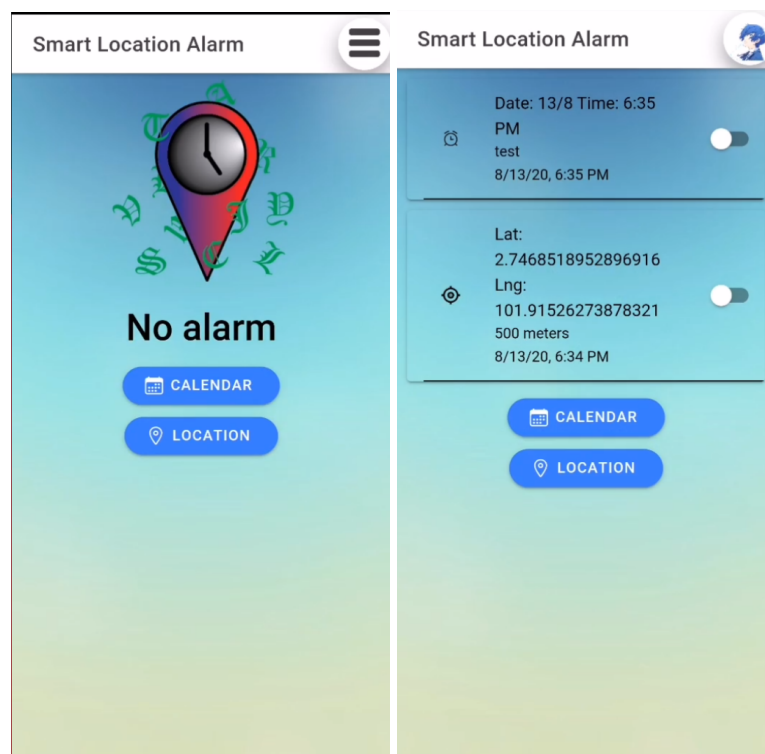


Figure 5.1: UI of Home Page as Anonymous (Left), Home Page as User (Right)

First of all, when the app is initialised, Home Page is the first page that will be shown on the app. If the user has already logged in and set some alarms, Home Page will be looking like the right figure.

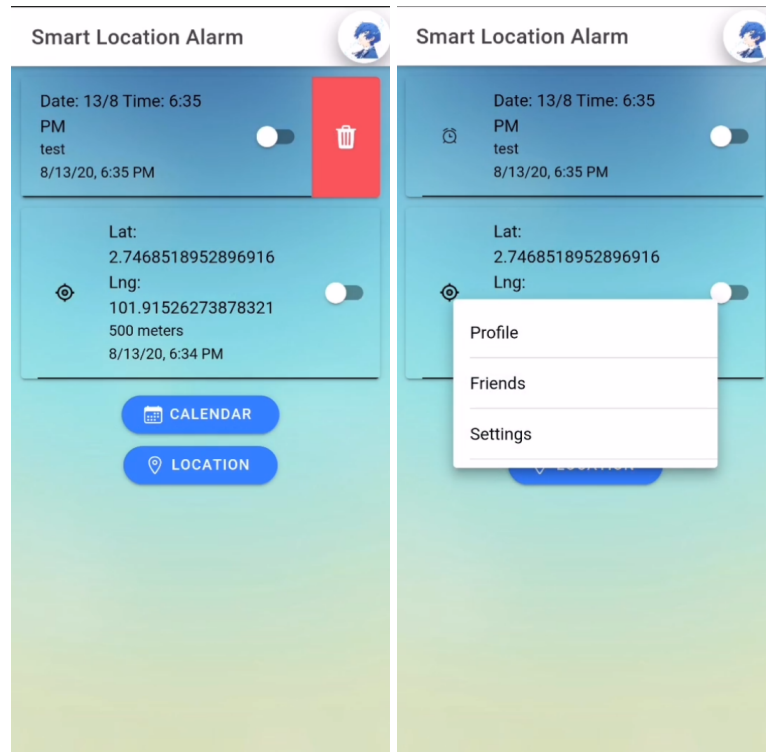
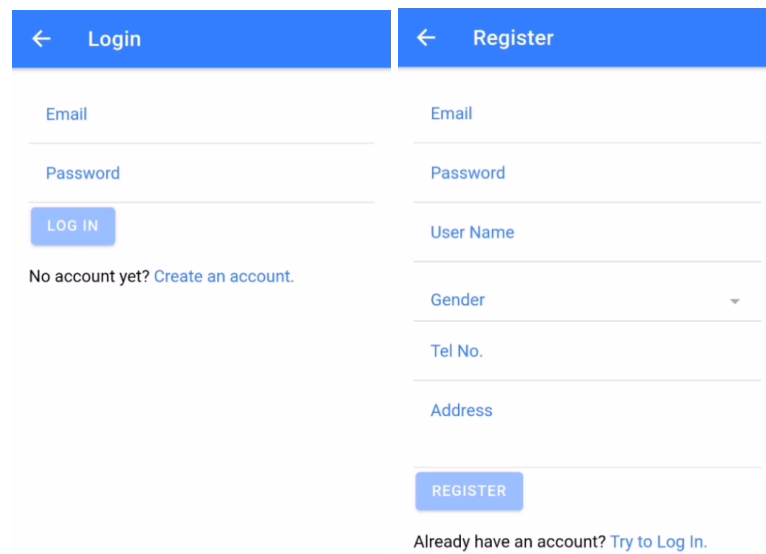


Figure 5.2: UI of Home Page with Delete (Left), Home Page with Menu (Right)

When the user swipes the alarm to the left side, a dustbin icon will appear. By clicking the dustbin icon, the alarm will be deleted. The right figure is when the user pressed the upper right image, then a menu will appear on the screen.

5.2.2 Login Page and Register Page



The image displays two side-by-side mobile application screens. The left screen is the 'Login' page, featuring a blue header with a back arrow and the text 'Login'. Below the header are two input fields labeled 'Email' and 'Password'. A blue button labeled 'LOG IN' is positioned below the password field. At the bottom, there is a link: 'No account yet? [Create an account.](#)' The right screen is the 'Register' page, also with a blue header containing a back arrow and the text 'Register'. It includes input fields for 'Email', 'Password', 'User Name', 'Gender' (with a dropdown arrow), 'Tel No.', and 'Address'. A blue button labeled 'REGISTER' is located below the address field. At the bottom, there is a link: 'Already have an account? [Try to Log In.](#)'

Figure 5.3: UI of Login Page (Left), Register Page (Right)

When the user has logged out, the menu contains Login and Register items that replaced Profile and Friends items. Login page is visible when the user pressed the Login item from the menu, whereas the Register Page is visible when the user pressed the Register item from the menu.

5.2.3 Settings Page

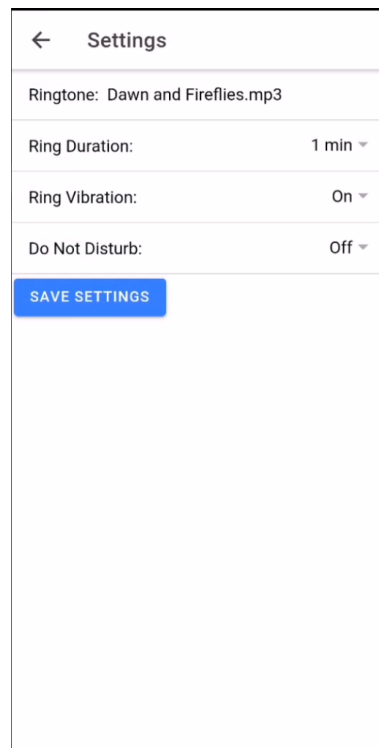


Figure 5.4: UI of Settings Page

Same goes to the Settings Page, it is also from the menu navigation. The user has more alarm customization from the Settings Page.

5.2.4 Profile Page and Edit Profile Page

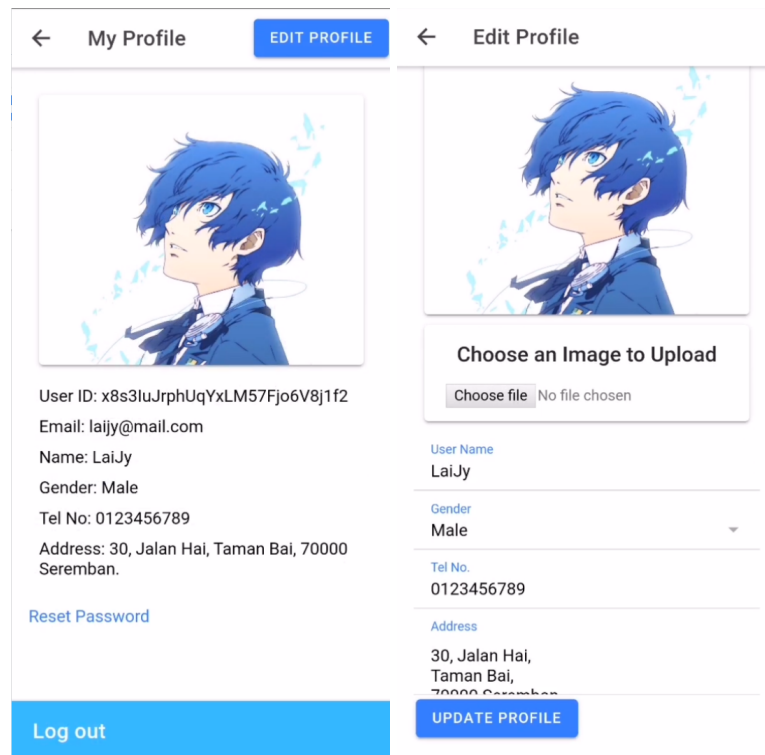


Figure 5.5: UI of Profile Page (Left), Edit Profile Page (Right)

When the user has logged in, the menu contains Profile and Friends items that replaced Login and Register items. If the user pressed the Profile item, personal details will be shown on Profile Page. Besides, the user is able to edit some personal details from the Edit Profile Page.

5.2.5 Friends Page and Friend Profile Page

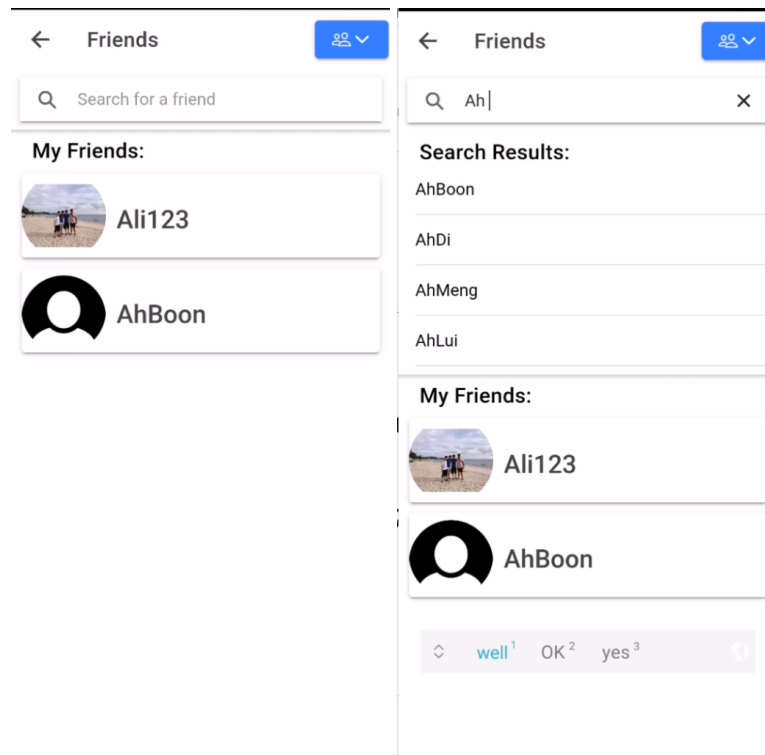


Figure 5.6: UI of Friends Page (Left), Friends Page with Search Result (Right)

If the user pressed the Friends item, the friend list can be seen on Friends Page. The user can use the search bar to search for other users.



Figure 5.7: UI of Friend Profile Page

If the user pressed on one of the friend items from the friend list, it will prompt to the friend profile page.

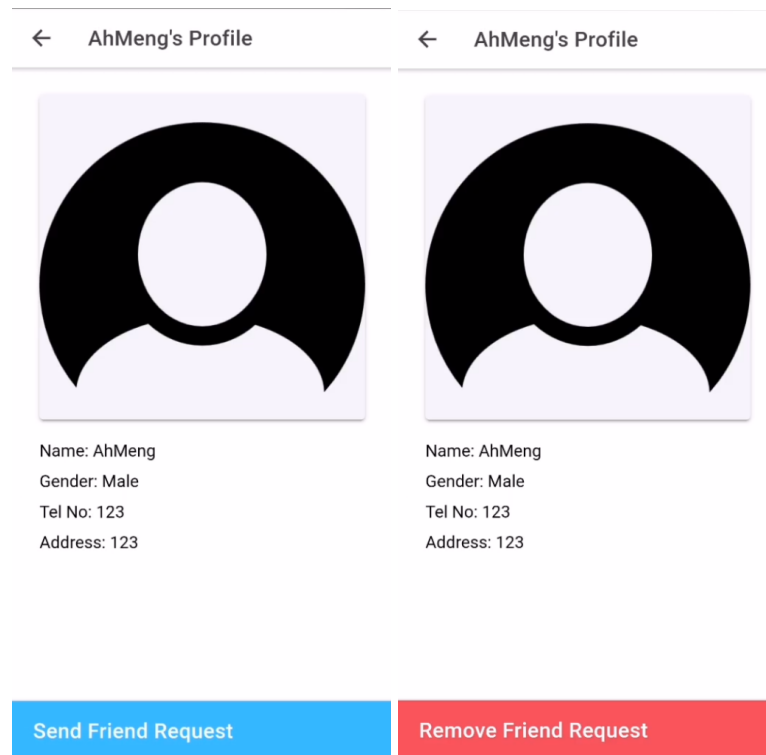


Figure 5.8: UI of Send Friend Request (Left), Remove Friend Request (Right)

If the user pressed on one of the users from the search results, it will navigate to the user profile. Thus, the user can send a friend request to him or her and remove it once the user regrets to do so.

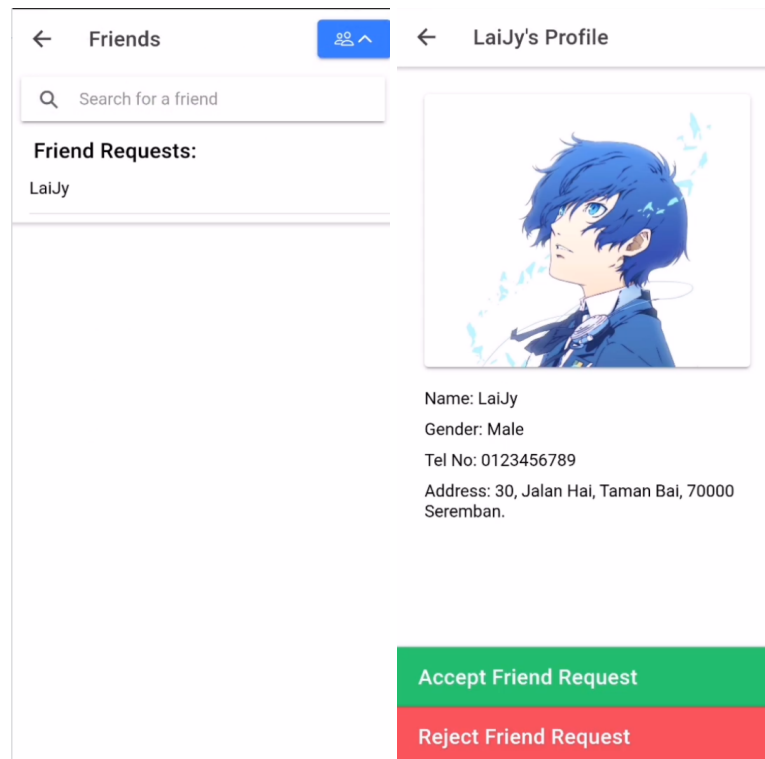


Figure 5.9: UI of Friends Page with Friend Request (Left), Accept or Reject Friend Request (Right)

For the user who gets the friend request, he or she is able to view the friend request on the Friends Page after pressing the upper right friend icon. Then, the user can choose to accept or reject the friend request once the user pressed the requester name from the friend requests.

5.2.6 Calendar Page

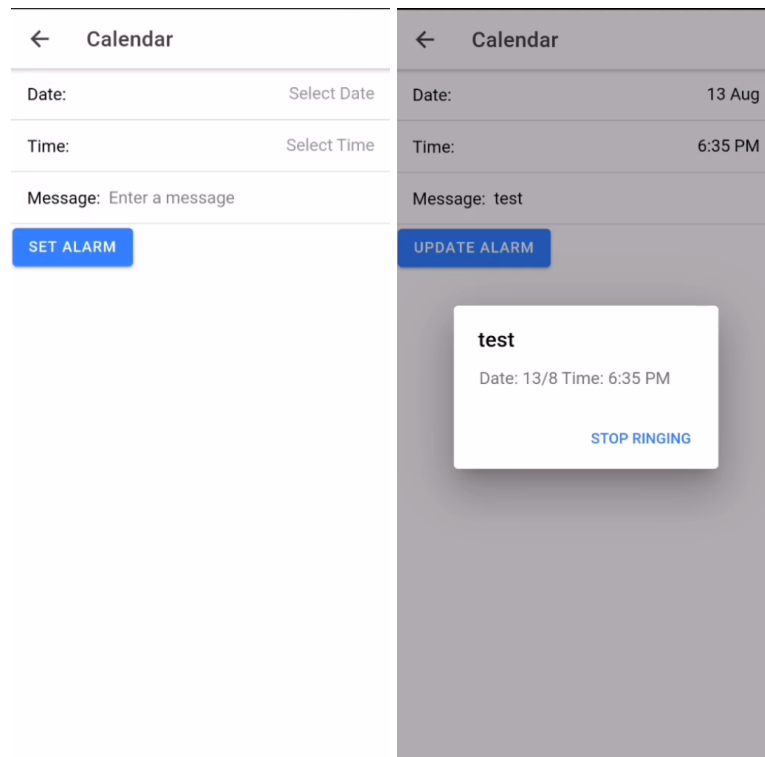


Figure 5.10: UI of Calendar Page (Left), Calendar Alarm (Right)

There are Calendar and Location buttons on the Home Page. The user can go to the Calendar Page by pressing the Calendar button. Next, the user can set the time-based alarm from the page. When the date is matched and the time is up, the time-based alarm is triggered as shown in the right figure.

5.2.7 Location Page

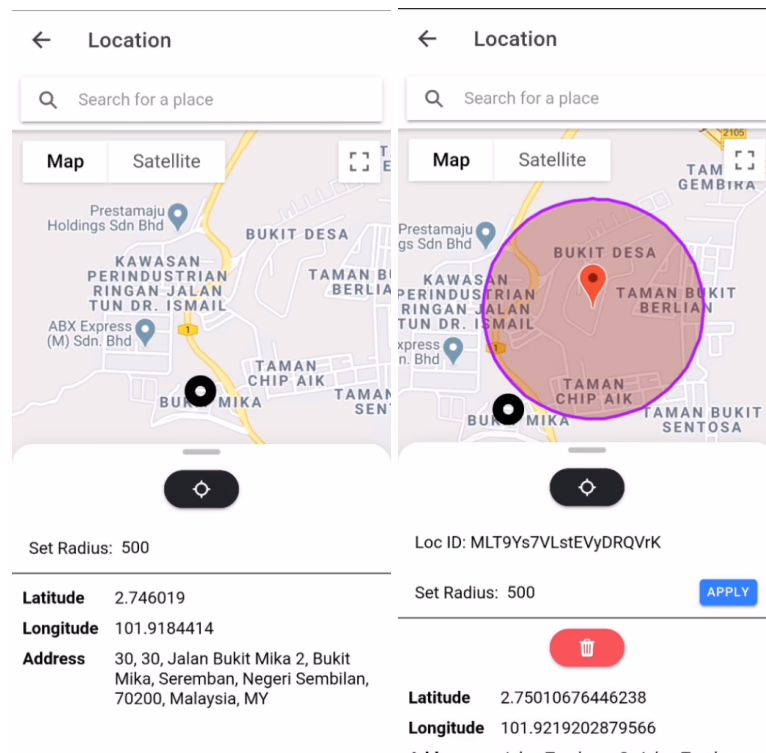


Figure 5.11: UI of Location Page as Anonymous (Left), Location Marker as Anonymous (Right)

Likewise, the user can go to the Location Page by pressing the Location button. These figures are the Location Page without login as an email user. And, there are some limitations that restrict the user to use some good features. Thus, it is advised to create an account in order to unlock and use all of the features from this app.

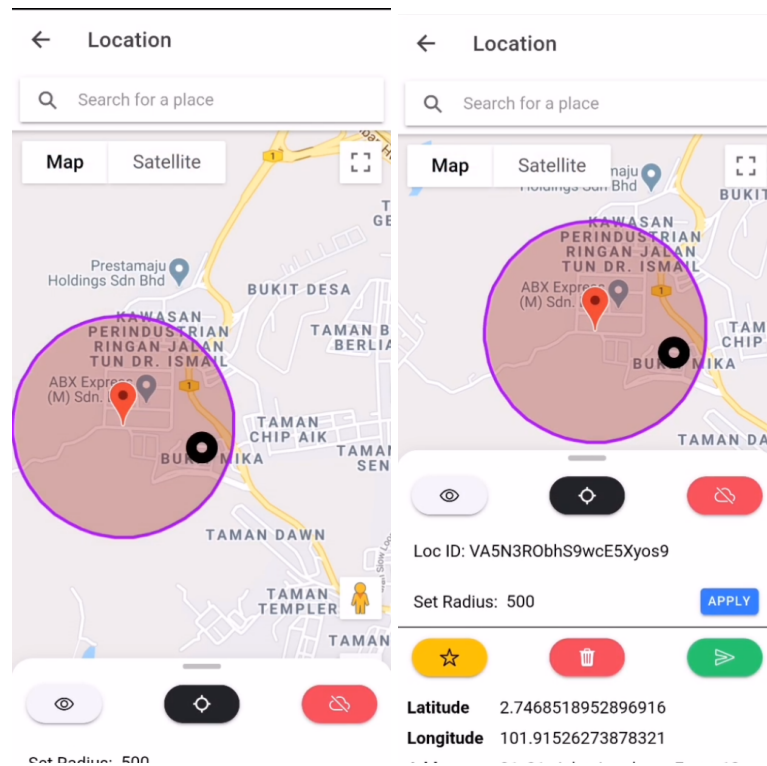


Figure 5.12: UI of Location Page as User (Left), Location Marker as User (Right)

These figures are the Location Page after login as an email user. 2 extra buttons are visible now from the left figure and there are 2 more buttons that appear if the user pressed one of the markers which is shown on the right figure.

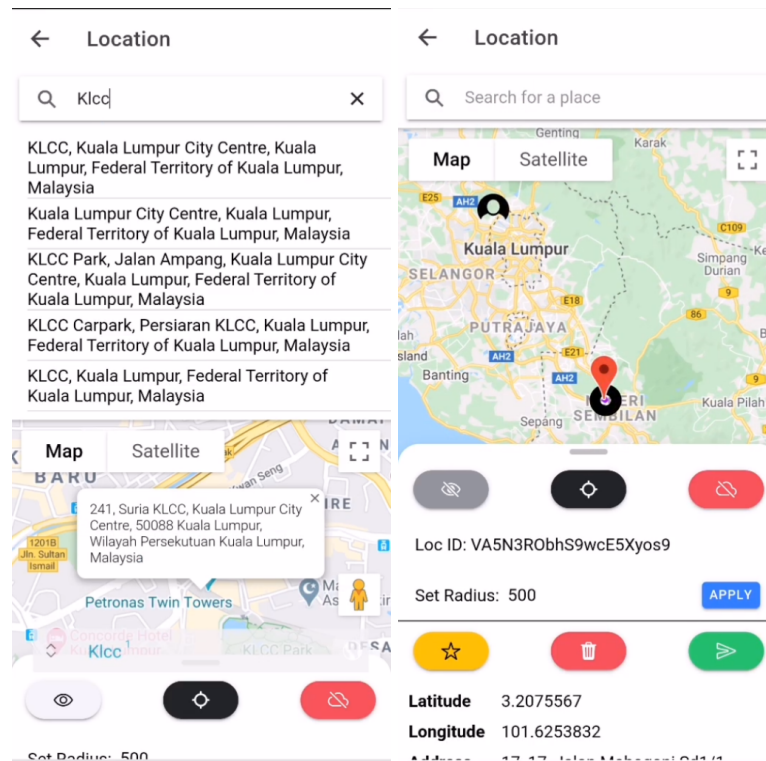


Figure 5.13: UI of Location Page with Search Result (Left), Location Page with Friend Shared Location (Right)

The user can search for places from the search bar and by pressing one of the search results, it will pan to the searched location on the map. After pressing the enable eye icon button, the user is able to view his/her friends' shared locations.

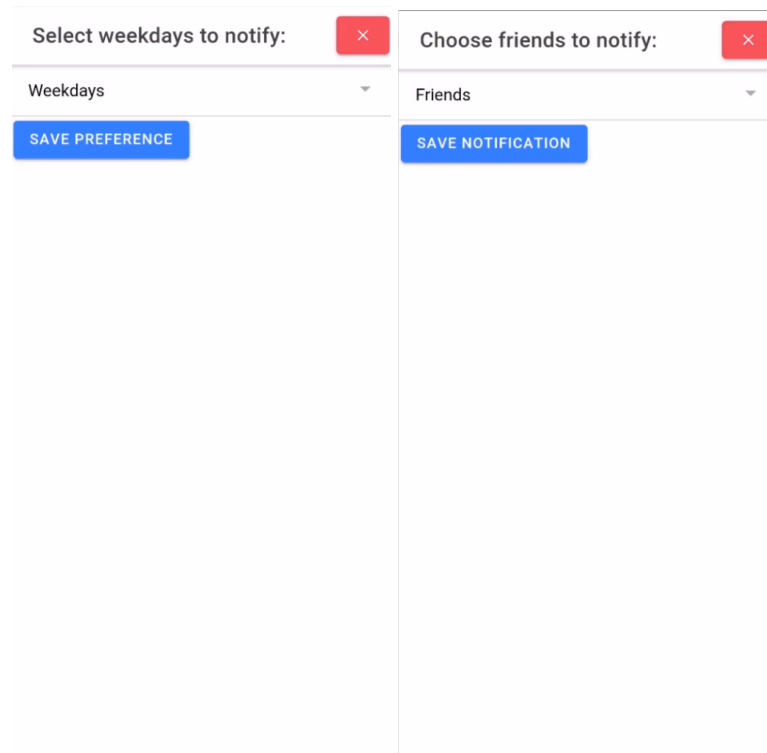


Figure 5.14: UI of Favourite Page (Left), Notification Page (Right)

After pressing the star icon button, the user can select the favourite weekdays to ring for the location-based alarm. Furthermore, the user can choose to notify friends when he or she reaches the destination marker after pressing the message icon button and saving the friends to notify.

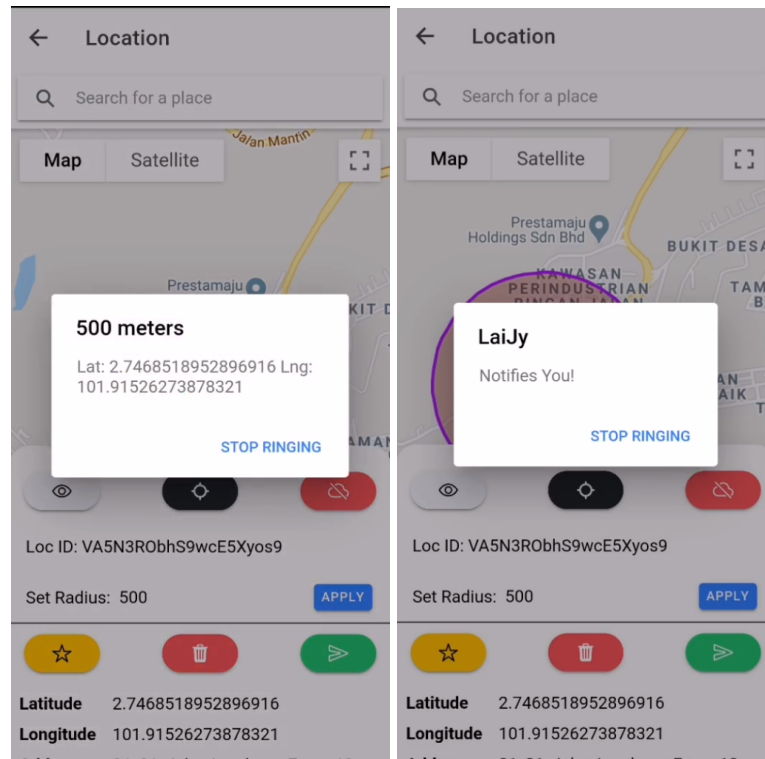


Figure 5.15: UI of Location Alarm (Left), Location Alarm from a Friend (Right)

The location-based alarm will trigger if the user reaches the marker area or the circle of the marker. On the other hand, the friend who is getting chosen will be notified too.

5.3 Class Diagram

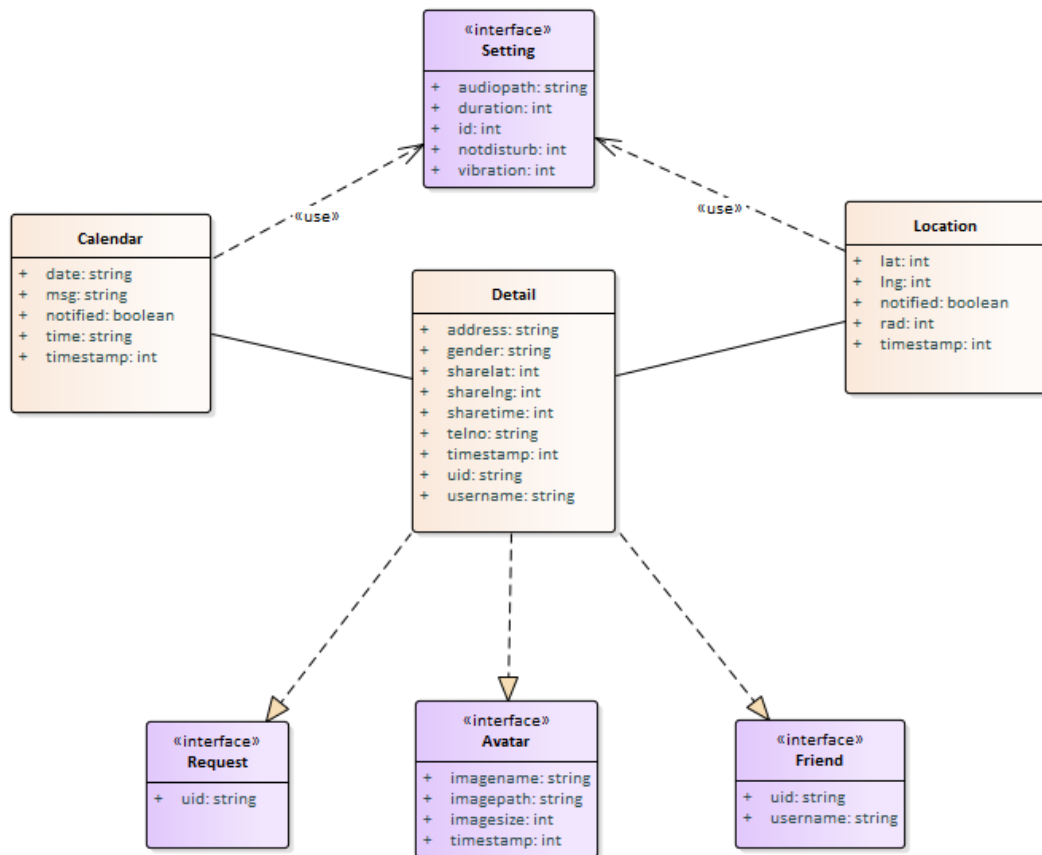


Figure 5.16: Class Diagram

5.4 Physical Entity Relationship Diagram

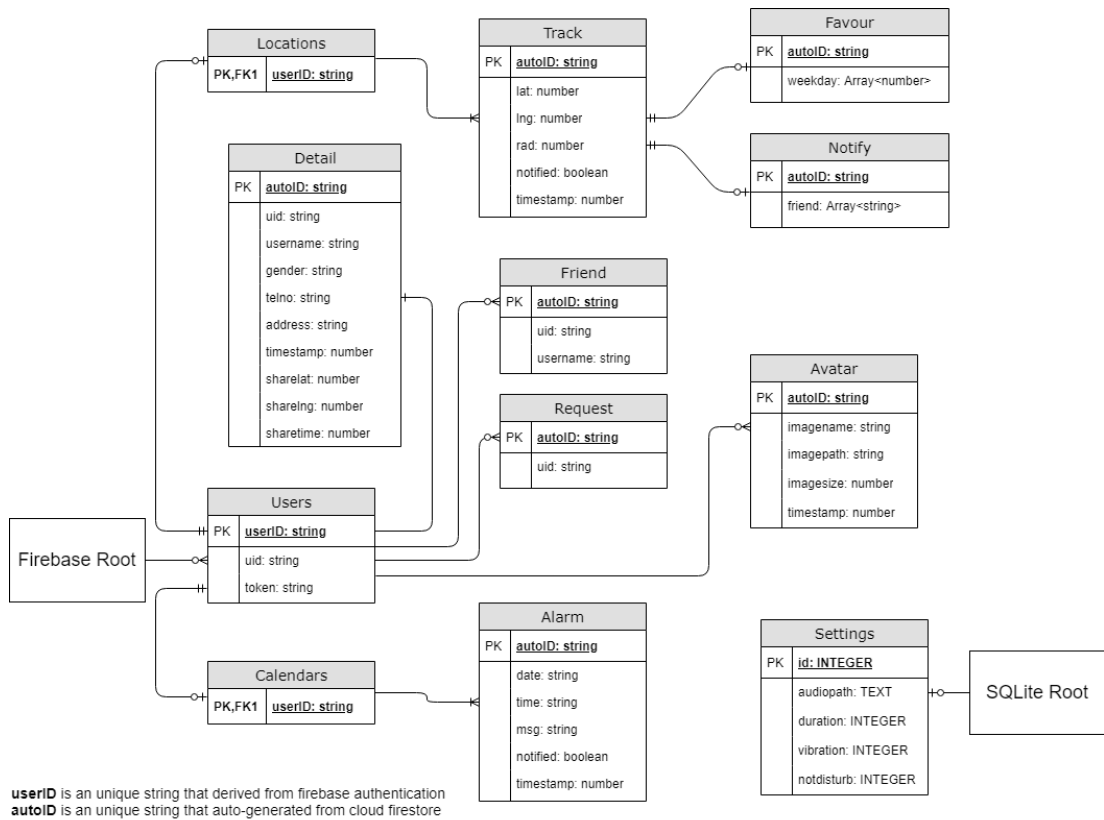


Figure 5.17: Physical Entity Relationship Diagram

Firebase		
Collections	Documents	Fields
Users	userID	uid and token
• Detail	autoID	uid, username, gender, telno, address, timestamp, sharelat, sharelng and sharetime
• Avatar	autoID	imagename, imagepath, imagesize and timestamp
• Friend	autoID	uid and username
• Request	autoID	uid
Calendars	userID	
• Alarm	autoID	date, time, msg, notified and timestamp
Locations	userID	

● Track	autoID	lat, lng, rad, notified and timestamp
○ Favour	autoID	weekday
○ Notify	autoID	friend
SQLite		
Tables	Attributes	
Settings	id, audiopath, duration, vibration and notdisturb	

Table 5.1: Physical ERD Description

- (Sub Collection)
- (Nested Sub Collection)

5.5 Physical Data Flow Diagram

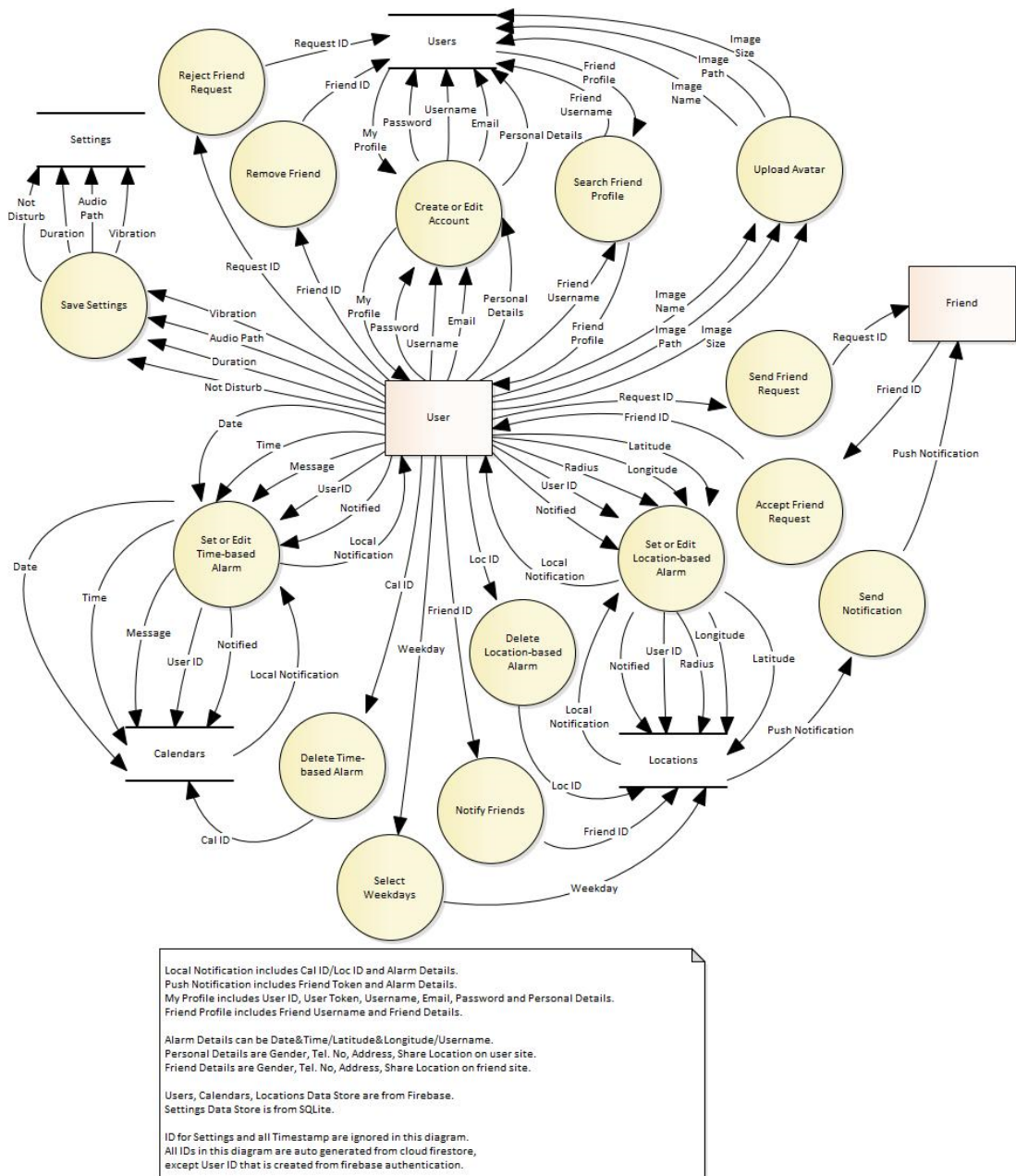


Figure 5.18: Physical Data Flow Diagram

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 Introduction

In this chapter, some essential code segments are provided and explained. Smart Location Alarm application is developed for almost everyone whether he or she is a student, working adult or senior citizen. Every user can choose to create an account and use it as an email user or use it plainly without registration.

Anonymous User (Without Account)	Email User (With Account)
Time-based Alarm Module <ul style="list-style-type: none"> - Set Alarm - Update Alarm - Delete Alarm 	Time-based Alarm Module <ul style="list-style-type: none"> - Set Alarm - Update Alarm - Delete Alarm
Location-based Alarm Module <ul style="list-style-type: none"> - Set Alarm - Update Alarm - Delete Alarm - Search Place - Satellites View - Foreground Tracking - Background Tracking - View Coordinates and Address 	Location-based Alarm Module <ul style="list-style-type: none"> - Set Alarm - Update Alarm - Delete Alarm - Search Place - Satellites View - Foreground Tracking - Background Tracking - View Coordinates and Address - View Friends' Shared Locations - Share Location to Friends - Filter Alarm with Weekdays - Notify Friends when triggered Alarm
Settings Module <ul style="list-style-type: none"> - Choose Ringtone - Change Ringtone Duration - Change Ringtone Vibration 	Settings Module <ul style="list-style-type: none"> - Select Ringtone - Change Ringtone Duration - Change Ringtone Vibration - Do Not Disturb Mode
	Social Profile Module <ul style="list-style-type: none"> - Upload Avatar - Edit Profile Details
	Friends System Module

	<ul style="list-style-type: none"> - Search Friend - View Friend Profile - Add Friend - Remove Friend
--	---

Table 6.1: Features available for Anonymous User and Email User

Available features between a user without an account also called as an anonymous user and a user with an account also known as an email user are listed based on Table 6.1.

6.2 Time-based Alarm Module

In this module, the user is able to select a date, time and enter a message to set a time alarm. The time alarm will ring along with the notification once the date is matched and the time is up. Besides, update and delete of the time alarm are also available for the user.

```

scheduleCalAlarm(calendars) {
  for( let cal of calendars){
    let s1 = cal.date.indexOf('/');
    let day = parseInt(cal.date.substring(0 , s1));
    let month = parseInt(cal.date.substring(s1 + 1));

    let c1 = cal.time.indexOf(':');
    let c2 = cal.time.indexOf(' ');
    let hour = parseInt(cal.time.substring(0 , c1));
    let minute = parseInt(cal.time.substring(c1 + 1, c2));
    let apm = cal.time.substring(c2 + 1);
    if (apm == 'AM' && hour == 12){ //Convert 12AM to 0
      hour = 0;
    }
    if (apm == 'PM' && hour != 12){ //Convert 1PM-11PM to 13-23 except 12PM
      hour = hour + 12;
    }
    console.log(month,day,hour,minute);

    if(cal.notified == false){
      let calTitle = cal.msg;
      let calBody = "Date: " + cal.date + " Time: " + cal.time;
      this.lclNoti.schedule({
        id: 1,
        title: calTitle,
        text: calBody,
        data: cal.id,
        trigger: { every: { month: month, day: day, hour: hour, minute: minute } , count: 1} //Schedule alarm ringing
      });
    }
  }
}

```

Figure 6.1: Schedule Time Alarm Code Segment

Figure 6.1 shows the schedule time alarm code segment, which user input is trimmed and converted to integer, in order to obtain the values like month, day, hour,

minute and meridiem. Local Notification plugin is used to schedule the notification with a callback that plays the ringtone based on the application settings with Media, Vibration plugins as well as shows an alert to the user for alarm stopping.

6.3 Location-based Alarm Module

Similar to the time-based alarm module, this module also allows the user to set, update and delete the location alarm. However, it consists of more features than the time-based alarm module, such as search places that use Geocoder plugin and Places library, share location to friends, view friends' shared location and so on. For Google Maps itself, Maps JavaScript API is used that supports every platform needed in this project such as Android, iOS and browser.

Geocoder plugin is used to convert coordinates to an exact address. Thus, the user can know the location with information. Whereas, Places library is an autocomplete feature, which returns place information after the request has been made by searching place name. Therefore, the user is able to use this feature to search for location that he or she wants to set the location alarm nearby.

```
//TRACK USER BACKGROUND
const config: BackgroundGeolocationConfig = {
  locationProvider: 0, //BackgroundGeolocationLocationProvider, 0 = DISTANCE_FILTER_PROVIDER(Default), 1 = ACTIVITY_PROVIDER, 2 = RAW_PROVIDER
  desiredAccuracy: 0, //BackgroundGeolocationAccuracy, 0 = HIGH, 10 = MEDIUM(Default), 100 = LOW
  stationaryRadius: 10,
  distanceFilter: 10,
  interval: 1000,
  fastestInterval: 1000,
  activitiesInterval: 1000,
  debug: false, //Enable will play a sound
  stopOnTerminate: false //Enable will stop when app halted
};
this.bgGeolocation.configure(config).then(() => {
  if(this.plt.is("ios"))
    this.bgGeolocation.finish(); //For iOS only
  this.bgGeoSub = this.bgGeolocation.on(BackgroundGeolocationEvents.location).subscribe((res: BackgroundGeolocationResponse) => {
    console.log("Background Geolocation: " + res.latitude + ', ' + res.longitude);
    let latLng = new google.maps.LatLng(res.latitude, res.longitude);
    if(this.user != null)
      this.calcService.startCalculation(this.user, latLng); //When background tracking updated, it calculates distance again
  });
});
this.bgGeolocation.start();
});
```

Figure 6.2: Background Tracking Code Segment

Tracking the user location is one of necessary parts to allow the location alarm to function well. In this project, the Background Geolocation plugin is used to track the user's location when the application is suspended. However, background

tracking is not as accurate as foreground tracking. Hence, it has less battery power consumption.

```
//TRACK USER FOREGROUND
let watchOptions = {
  poll: 1000, //Check for changes every 1 sec
  enableHighAccuracy: true //True = GPS (might cause error), False = ISP (not accurate)
};
this.watchSub = this.geolocation.watchPosition(watchOptions).pipe(filter((p: any) => p.code === undefined)).subscribe((data: Geoposition) => {
  console.log('Foreground Geolocation: ' + data.coords.latitude + ', ' + data.coords.longitude);
  let latLng = new google.maps.LatLng(data.coords.latitude, data.coords.longitude);
  this.selfMarker.setPosition(latLng);
  this.calcService.startCalculation(this.user, latLng); //When watchPosition subscribe updated, it calculates distance again
});
```

Figure 6.3: Foreground Tracking Code Segment

On the other hand, the Geolocation plugin is used to track the user's location when the application is running. It is more accurate because it uses GPS technology instead of claiming the ISP location as the user's location.

To clarify further, when the user launches the app, background tracking is used first. When the user navigates to the Location Page, it switches to foreground tracking and is available throughout the running app, until the app has been closed or suspended. If the app is suspended, it switches back to background tracking and once the user continues using the app without closing it in between, it changes to foreground tracking again. Both trackings are unavailable when the app is closed.

```
calculateDistance(selfPos, locations){
  for(let loc of locations){
    let xn = loc.lat;
    let yn = loc.lng;
    let r = loc.rad;
    let locPos = new google.maps.LatLng(xn,yn);
    console.log("Loc: ",xn,yn)
    let d = google.maps.geometry.spherical.computeDistanceBetween(selfPos, locPos); //Return in meters
    console.log("Distance between user and added marker: ", d);
    this.trigger = d-r;
    console.log("Ans: ", this.trigger);
    if(this.trigger < 0 && loc.notified == false){
      this.pullNotification(loc);
    }
  }
}
```

Figure 6.4: Calculate Distance Code Segment

Figure 6.4 presents the distance calculation code segment that uses the Geometry library to return the distance between two coordinates. The distance stated

here is the distance between the user marker and the location alarm marker. Thus, when the user marker is inside the location alarm circle, the subtraction of the distance and the location alarm circle's radius will return a negative value. If it is negative then further notification checking will be taking place. For instance, check for favourite weekdays to ring and the chosen friends to notify.

In addition, there is a plugin called Geofence. It provides a quick way to implement the detection of user location in marked areas. However, it requires many dependencies that might lead to the project build failure. Hence, by implementing the distance calculation and the location alarm circle without plugin is the better way for this project.

```
triggerLocAlarm(loc){
  let locTitle = loc.rad + " meters" ;
  let locBody = "Lat: " + loc.lat + "   Lng: " + loc.lng;
  this.lclNoti.schedule({
    id: 2,
    title: locTitle,
    text: locBody ,
    data: loc.id,
    trigger: {count: 1} //Trigger alarm ringing
  });
}

triggerLocAlarmWithFren(loc,userID,username){
  let locTitle = loc.rad + " meters" ;
  let locBody = "Lat: " + loc.lat + "   Lng: " + loc.lng;
  this.lclNoti.schedule({
    id: 3,
    title: locTitle,
    text: locBody,
    data: {locID: loc.id, frenID: this.friendList, userID: userID, username: username},
    trigger: {count: 1} //Trigger alarm ringing
  });
}
```

Figure 6.5: Trigger Location Alarm Code Segment

Figure 6.5 shows two types of triggering location alarm functions. The first one with ID equals to 2 is implemented for self notifying only, whereas the second one with ID equals to 3 is implemented for self notifying and also notifying the user's chosen friends.

```

if(res.id == 3 && this.locDelay == 0){
  this.locDelay++;
  this.afs.firestore.collection(`locations/${this.user.uid}/track`).doc(res.data.locID)
  .get().then(doc=>{
    if(doc.data().notified == false){
      this.afs.firestore.collection(`locations/${this.user.uid}/track`).doc(res.data.locID).update({
        notified: true
      }).then(()=>{
        for(let uid of res.data.frenID)
          this.searchFToken(uid,res.data.userID,res.data.username,"Notifies You!");
        var pattern = [];
        var n = this.confService.getDuration()/1000;
        for (var i = 1; i <= n; i++) {
          pattern = pattern.concat(1000);
        };
        if(this.confService.getVibration() == 1)
          this.vib.vibrate(pattern); //Vibrate duration is ignored on iOS, maximum vibration streak is 10000 on android

        this.locMusic = this.media.create(this.confService.getAudio());
        this.locMusic.play();
        window.setTimeout(() => {
          this.locMusic.stop();
          this.vib.vibrate(0); //Stop vibration, not supported on iOS
        }, this.confService.getDuration());

        this.alertCtrl.create({
          header: res.title,
          message: res.text,
          buttons: [{
            text: 'Stop Ringing',
            handler: () => {
              this.locMusic.stop();
              this.vib.vibrate(0);
            }
          }],
          backdropDismiss: false //Force user to press button
        }).then(alert=> alert.present());

        window.setTimeout(()=>{this.resetLocDelay();},1000);
      });
    }
  });
}

```

Figure 6.6: Receive Notification Code Segment

In this code segment, it just shows the alarm with ID = 3 callback. It is corresponding to time alarm callback (ID = 1) and location alarm callback (ID = 2) except for calling one more function to search for chosen friends' tokens. This is to send the push notification to one or more friends right after the local notification is triggered. Moreover, the push notification callbacks are also similar to the local notification callbacks.

Push notification requires Firebase X plugin, which allows the application to generate token for receiving notification. It is compatible with both Android and iOS but it requires a higher version of Xcode and upgrading from Xcode 10 to Xcode 11 requires a better Mac OS. Thus, another plugin called FCM is used for push notification in iOS.

```

sendNotification(token,title,body) {
  let postData = {
    "notification":{
      "title":title,
      "body":body,
      "sound":"default"
    },
    "to":token,
    "priority":"high",
    "restricted_package_name":""
  }
  let httpOptions = {
    headers: new HttpHeaders({
      'Content-Type': 'application/json',
      'Authorization': 'key=AAAAiI2wTbE:APA91bEczx-NnVznxWmKSpMwXE3JvIkMA1UwtSqCEDOZm8KGcNgjwGt7Y
    })
  }
  this.httpSub = this.httpClient.post('https://fcm.googleapis.com/fcm/send',postData,httpOptions)
  .subscribe(postData => { //Subscribe to make httpClient works
    console.log(postData)
  }, error => {
    console.log(error);
  });
}

```

Figure 6.7: Send Notification Code Segment

Send the push notification is done by using HttpClient component from Angular framework. It is similar to an application like Postman, which allows the user to send HTTP requests.

6.4 Settings Module

For the settings module, the biggest difference from other modules is it used SQLite instead of Firebase. This is because settings data is usually stored on phone devices locally. Once the user has uninstalled the application, the settings data will be eliminated as well. Default settings data is used if the application is installed.

```

loadAudio() {
  if (this.plt.is("android")) {
    let filter = {"mime": "audio/wav,audio/mp3,audio/aac,audio/ogg,audio/3gp,audio/wma,audio/m4a"}
    this.fileChooser.open(filter).then(uri => {
      this.filePath.resolveNativePath(uri).then((nativepath)=>{
        this.audiopath = nativepath;
        let ls = this.audiopath.lastIndexOf('/');
        this.audioname = this.audiopath.substring(ls + 1);
      })
    })
  }
  else if (this.plt.is("ios")) {
    this.filePicker.pickFile("public.audio").then(uri => {
      this.audiopath = uri;
      let ls = this.audiopath.lastIndexOf('/');
      this.audioname = this.audiopath.substring(ls + 1);
    })
    .catch(err => this.showToast(err));
  }
}

```

Figure 6.8: Choose Ringtone Code Segment

Figure 6.8 shows the code segment of choosing ringtone from the mobile phone file directory. Different plugins are used for different platforms due to compatibility issues. Audio path is stored to SQLite but not for audio name. Audio name is trimmed and displayed to allow the user to see which song or music is selected for the alarm ringtone only.

6.5 Social Profile Module

Social profile module or account module that includes login and register is the only module that uses Reactive Form component. Therefore, form and input validations are important in this module.

```

noWhitespaceValidator(control: FormControl) {
  if (control && control.value) {
    const isSpace = (control.value || '').match(/\s/g);
    return isSpace ? {'whitespace': true} : null;
  }
}

noAlphabetValidator(control: FormControl) {
  if (control && control.value) {
    if (!isNumeric(control.value)){
      return {'alphabet': true};
    }
    else
      return null;
  }
}

```

Figure 6.9: Form Validators Code Segment

There are many validations for this project. For example, input required, input length, input pattern and so on. Figure above shows the custom validations for username and Tel. No respectively. Username is unique and it cannot contain any whitespace in this application, so it helps in searching users. Besides, alphabets are not allowed when entering the Tel. No.

Apart from editing the user details that require validations in the social profile module, the user is able to upload an image as the avatar. The image upload process can be resumed, paused or cancelled. Most of the data is stored in Firebase Firestore except for the image files that are larger size, are stored in Firebase Storage.

6.6 Friends System Module

After implementing the social profile module, the friends system is also an inevitable module for the application. With this module, the user can interact with other users by adding them as friends.

```
//SEARCH FRIEND WITH USERNAME
filterUsernames(searchInput) {
  return this.usernameList.filter(uname => {
    return uname.toLowerCase().indexOf(searchInput.toLowerCase()) > -1;
  });
}

getSearchResult(searchResult){
  this.searchResult = searchResult;
  this.getOneFriendDetail();
}

getOneFriendDetail(){
  for(let friend of this.detailList){
    if(friend.username == this.searchResult){
      console.log("Friend username and search input matched!",friend)
      this.navigateFriendProfile(friend);
    }
  }
}
```

Figure 6.10: Search Friend Code Segment

Figure 6.10 is the code segment of searching a friend by entering his/her username. The system will retrieve every username from the database and check for the searched username. If the username matched, the system will obtain the user details so that the user can navigate to the user profile and do further action like send or remove the friend request.

If the user navigates to another user profile from friend requests list but not from search friends result, the user is able to accept or reject the friend request. If the user navigates to another user profile from friends list, the user is able to unfriend the user. After pressing the unfriend button, an alert will pop up to warn the user about the location alarms which are bound with the user will be reset.

CHAPTER 7

SYSTEM TESTING

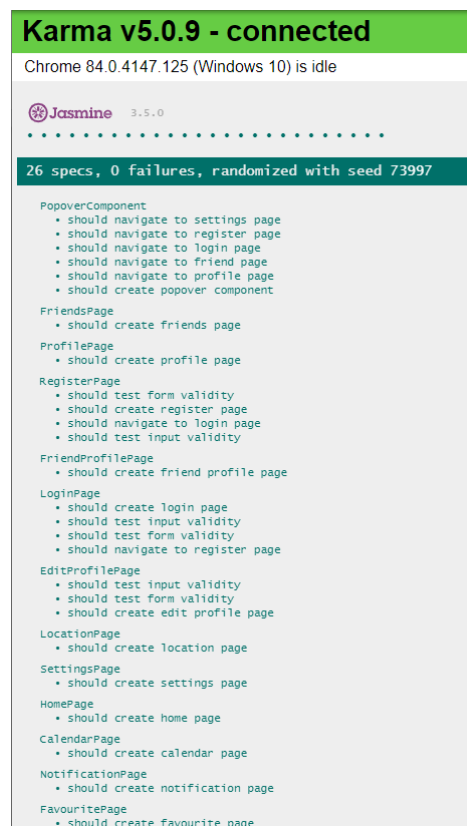
7.1 Introduction

This chapter includes different types of testing procedures like Unit Testing, Integration Testing, System Testing, Usability Testing and User Acceptance Testing.

7.2 Test Code

Testing framework used in this project is Jasmine framework and the test runner is Karma that controls the browser like Google Chrome to show the test result. The testing framework and runner are integrated with Ionic and Angular framework. The test code is written in the spec files provided and the test will be executed after typing the 'npm test' command.

7.2.1 Unit Testing



```
Karma v5.0.9 - connected
Chrome 84.0.4147.125 (Windows 10) is idle
Jasmine 3.5.0
26 specs, 0 failures, randomized with seed 73997

PopoverComponent
  • should navigate to settings page
  • should navigate to register page
  • should navigate to login page
  • should navigate to friend page
  • should navigate to profile page
  • should create popover component

FriendsPage
  • should create friends page

ProfilePage
  • should create profile page

RegisterPage
  • should test form validity
  • should create register page
  • should navigate to login page
  • should test input validity

FriendProfilePage
  • should create friend profile page

LoginPage
  • should create login page
  • should test input validity
  • should test form validity
  • should navigate to register page

EditProfilePage
  • should test input validity
  • should test form validity
  • should create edit profile page

LocationPage
  • should create location page

SettingsPage
  • should create settings page

HomePage
  • should create home page

CalendarPage
  • should create calendar page

NotificationPage
  • should create notification page

FavouritePage
  • should create favourite page
```

Figure 7.1: Actual Result of Unit Testing

Spec File	Test Case ID	Test Case	Input	Expected Result	Test Result
Home Page	1	Create home page	-	Home Page is created	Pass
Login Page	2	Create login page	-	Login Page is created	
	3	Test form validity	Valid Email and Password	Login Form validation is true	Pass
	4	Test input validity	Valid Email and Password	Email, Password validations are true	Pass
			Invalid Email and Password	Email, Password validations are false	
	5	Navigate to register page	Click 'Create an Account' hyperlink	Navigate forward to Register Page has been called	Pass
Register Page	6	Create register page	-	Register Page is created	Pass
	7	Test form validity	Valid Email, Password, Username, Gender, Tel. No and Address	Register Form validation is true	Pass
	8	Test input validity	Valid Email, Password, Username, Gender, Tel. No and Address	Email, Password, Username, Gender, Tel. No, Address validations are true	Pass
			Invalid Email, Password, Username, Gender, Tel. No and Address	Email, Password, Username, Gender, Tel. No, Address validations are false	
	9	Navigate to	Click 'Already	Navigate forward	Pass

		login page	have an account?’ hyperlink	to Login Page has been called	
Settings Page	10	Create settings page	-	Settings Page is created	Pass
Profile Page	11	Create profile page	-	Profile Page is created	Pass
Edit Profile Page	12	Create edit profile page	-	Edit Profile Page is created	Pass
	13	Test form validity	Valid Username, Gender, Tel. No and Address	Edit Profile Form validation is true	Pass
	14	Test input validity	Valid Username, Gender, Tel. No and Address	Username, Gender, Tel. No, Address validations are true	Pass
			Invalid Username, Gender, Tel. No and Address	Username, Gender, Tel. No, Address validations are false	
Friends Page	15	Create friends page	-	Friends Page is created	Pass
Friend Profile Page	16	Create friend profile page	-	Friend Profile Page is created	Pass
Calendar Page	17	Create calendar page	-	Calendar Page is created	Pass
Location Page	18	Create location page	-	Location Page is created	Pass
Favourite Page	19	Create favourite page	-	Favourite Page is created	Pass
Notification	20	Create	-	Notification Page	Pass

Page		notification page		is created	
Popover Component	21	Create popover component	-	Popover Component is created	Pass
	22	Navigate to login page	Click 'Login' item	Router navigate to Login Page has been called	Pass
	23	Navigate to register page	Click 'Register' item	Router navigate to Register Page has been called	Pass
	24	Navigate to settings page	Click 'Settings' item	Router navigate to Settings Page has been called	Pass
	25	Navigate to profile page	Click 'Profile' item	Router navigate to Profile Page has been called	Pass
	26	Navigate to friends page	Click 'Friends' item	Router navigate to Friends Page has been called	Pass

Table 7.1: Unit Test Cases

7.2.2 Integration Testing

```

Karma v5.0.9 - connected
Chrome 84.0.4147.125 (Windows 10) is idle
Jasmine 3.5.0
.....
26 specs, 0 failures, randomized with seed 97319

IntegrationTesting
  • should create settings page
RegisterPage
  • should navigate to login page
  • should test input validity
  • should test form validity
  • should create register page
  • should create profile page
  • should create home page
EditProfilePage
  • should test input validity
  • should test form validity
  • should create edit profile page
  • should create calendar page
  • should create friend profile page
  • should create location page
PopoverComponent
  • should create popover component
  • should navigate to login page
  • should navigate to register page
  • should navigate to profile page
  • should navigate to friend page
  • should navigate to settings page
  • should create friends page
LoginPage
  • should test input validity
  • should create login page
  • should test form validity
  • should navigate to register page
  • should create notification page
  • should create favourite page

```

Figure 7.2: Actual Result of Integration Testing

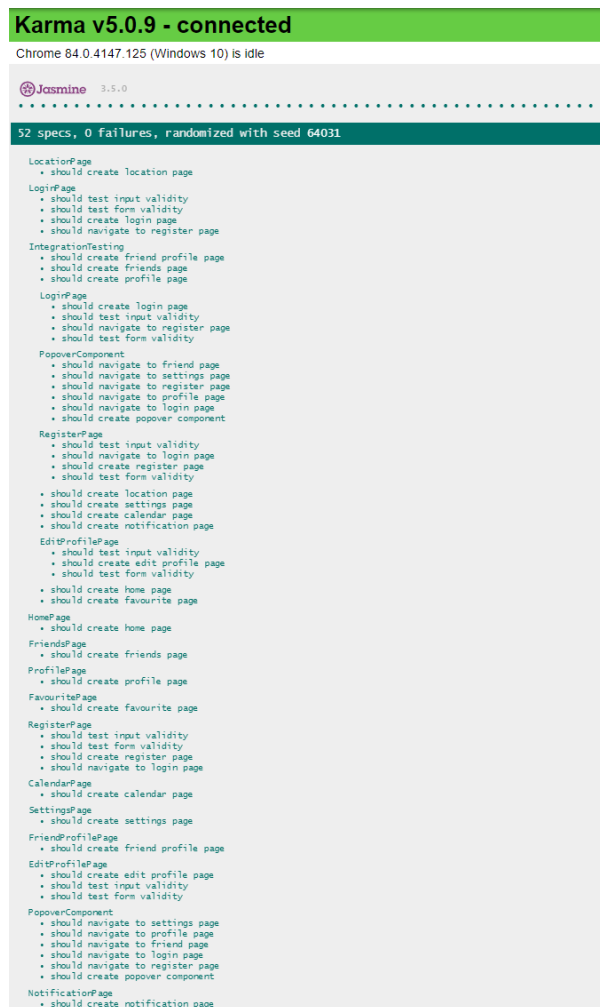
Spec File	Test Case ID	Test Case	Input	Expected Result	Test Result
Integration	1	Create home page	-	Home Page is created	Pass
	2	Create login page	-	Login Page is created	Pass
		Test form validity	Valid Email and Password	Login Form validation is true	
		Test input validity	Valid Email and Password	Email, Password validations are true	
			Invalid Email and Password	Email, Password validations are false	
		Navigate to register page	Click 'Create an Account' hyperlink	Navigate forward to Register Page has been called	

3	Create register page	-	Register Page is created	Pass
	Test form validity	Valid Email, Password, Username, Gender, Tel. No and Address	Register Form validation is true	
	Test input validity	Valid Email, Password, Username, Gender, Tel. No and Address	Email, Password, Username, Gender, Tel. No, Address validations are true	
		Invalid Email, Password, Username, Gender, Tel. No and Address	Email, Password, Username, Gender, Tel. No, Address validations are false	
	Navigate to login page	Click 'Already have an account?' hyperlink	Navigate forward to Login Page has been called	
4	Create settings page	-	Settings Page is created	Pass
5	Create profile page	-	Profile Page is created	Pass
6	Create edit profile page	-	Edit Profile Page is created	Pass
	Test form validity	Valid Username, Gender, Tel. No and Address	Edit Profile Form validation is true	
	Test input validity	Valid Username, Gender, Tel. No and Address	Username, Gender, Tel. No, Address validations are true	
		Invalid Username, Gender, Tel. No	Username, Gender, Tel. No, Address	

			and Address	validations are false	
7	Create friends page	-		Friends Page is created	Pass
8	Create friend profile page	-		Friend Profile Page is created	Pass
9	Create calendar page	-		Calendar Page is created	Pass
10	Create location page	-		Location Page is created	Pass
11	Create favourite page	-		Favourite Page is created	Pass
12	Create notification page	-		Notification Page is created	Pass
13	Create popover component	-		Popover Component is created	Pass
	Navigate to login page	Click 'Login' item		Router navigate to Login Page has been called	
	Navigate to register page	Click 'Register' item		Router navigate to Register Page has been called	
	Navigate to settings page	Click 'Settings' item		Router navigate to Settings Page has been called	
	Navigate to profile page	Click 'Profile' item		Router navigate to Profile Page has been called	
	Navigate to friends page	Click 'Friends' item		Router navigate to Friends Page has been called	

Table 7.2: Integration Test Cases

7.2.3 Test Suite



```

Karma v5.0.9 - connected
Chrome 84.0.4147.125 (Windows 10) is idle
Jasmine 3.5.0
52 specs, 0 failures, randomized with seed 64031

LocationPage
  • should create location page
LoginPage
  • should test input validity
  • should test form validity
  • should create login page
  • should navigate to register page
IntegrationTesting
  • should create friend profile page
  • should create friends page
  • should create profile page
LoginPage
  • should create login page
  • should test input validity
  • should navigate to register page
  • should test form validity
PopoverComponent
  • should navigate to friend page
  • should navigate to settings page
  • should navigate to register page
  • should navigate to profile page
  • should navigate to login page
  • should create popover component
RegisterPage
  • should test input validity
  • should navigate to login page
  • should create register page
  • should test form validity
  • should create location page
  • should create settings page
  • should create calendar page
  • should create notification page
EditProfilePage
  • should test input validity
  • should create edit profile page
  • should test form validity
  • should create home page
  • should create favourite page
HomePage
  • should create home page
FriendsPage
  • should create friends page
ProfilePage
  • should create profile page
FavouritePage
  • should create favourite page
RegisterPage
  • should test input validity
  • should test form validity
  • should create register page
  • should navigate to login page
CalendarPage
  • should create calendar page
SettingsPage
  • should create settings page
FriendProfilePage
  • should create friend profile page
EditProfilePage
  • should create edit profile page
  • should test input validity
  • should test form validity
PopoverComponent
  • should navigate to settings page
  • should navigate to friend page
  • should navigate to login page
  • should navigate to register page
  • should create popover component
NotificationPage
  • should create notification page

```

Figure 7.3: Actual Result of Test Suite

Test suite is used to run both unit testing and integration testing together. Thus, every case is tested twice in the test suite.

7.3 System Testing

Due to the application being cross-platform, system testing has to be done in both Android and iOS. To run the application on an Android device, a command ‘ionic cordova run android --device’ has to be entered. Besides, the project has to be loaded in Xcode in order to run the application on an iOS simulator.

Test Case ID 1 to 2 must test before login as an email user.

Test Case ID 3 to 9 and 17 to 20 must test after login as an email user.

Test Case ID 10 to 16 and 21 can be tested before login or after login.

Test Case ID:	1	
Test Case:	Create a user account	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Register Page 3. Fill in valid Email, Password, Username, Gender, Tel. No and Address 4. Press Register 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Your account has been created and logged in.' will be displayed. 2. Account has been created and logged in. 	<ol style="list-style-type: none"> 1. A message of 'Your account has been created and logged in.' will be displayed. 2. Account has been created and logged in.

Table 7.3: System Test Case of Create a User Account

Test Case ID:	2	
Test Case:	Login to an account	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Login Page 3. Fill in valid Email and Password 4. Press Login 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Navigate to Home Page. 2. If the user has existing alarms, they will be displayed out. 3. Menu icon will be changed to user icon or avatar. 	<ol style="list-style-type: none"> 1. Navigate to Home Page. 2. If the user has existing alarms, they will be displayed out. 3. Menu icon will be changed to user icon or avatar.

Table 7.4: System Test Case of Login to an Account

Test Case ID:	3	
Test Case:	Logout from an account	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Profile Page 3. Press Logout 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Navigate to Home Page. 2. The alarms and menu icon are reset back to default. 	<ol style="list-style-type: none"> 1. Navigate to Home Page. 2. The alarms and menu icon are reset back to default.

Table 7.5: System Test Case of Logout from an Account

Test Case ID:	4	
Test Case:	Edit Profile	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Profile Page 3. Press Edit Profile 4. Press Choose File to upload new avatar 5. Fill in new valid Username, Gender, Tel. No or Address 6. Press Update Profile 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Navigate to Profile Page. 2. New profile details and avatar are visible. 	<ol style="list-style-type: none"> 1. Navigate to Profile Page. 2. New profile details and avatar are visible.

Table 7.6: System Test Case of Edit Profile

Test Case ID:	5	
Test Case:	Reset Password	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Profile Page 3. Press Reset Password 	

Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Reset password email has been sent to your mailbox' will be displayed. 2. Mailbox has the auto generated email from firebase. 	<ol style="list-style-type: none"> 1. A message of 'Reset password email has been sent to your mailbox' will be displayed. 2. Mailbox has the auto generated email from firebase.

Table 7.7: System Test Case of Reset Password

Test Case ID:	6	
Test Case:	Search Friend and Send/Remove Friend Request	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Friends Page 3. Fill in search bar with username 4. Press the search result 5. Press Send Friend Request 6. Press Remove Friend Request 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. If step 5 is executed, the sender will get a message of 'Your request has been sent to receiver name' and Send Friend Request button will change to Remove Friend Request button. 2. If step 6 is executed, the sender will get a message of 'Your request has been removed from receiver name' and Remove Friend Request button will change to Send Friend Request button. 3. The receiver will get 	<ol style="list-style-type: none"> 1. If step 5 is executed, the sender will get a message of 'Your request has been sent to receiver name' and Send Friend Request button will change to Remove Friend Request button. 2. If step 6 is executed, the sender will get a message of 'Your request has been removed from receiver name' and Remove Friend Request button will change to Send Friend Request button. 3. The receiver will get

	<p>a notification or message when the request has been received.</p> <p>4. The receiver will not get a notification or message when the request has been removed.</p>	<p>a notification or message when the request has been received.</p> <p>4. The receiver will not get a notification or message when the request has been removed.</p>
--	---	---

Table 7.8: System Test Case of Search Friend and Send/Remove Friend Request

Test Case ID:	7	
Test Case:	Accept Friend Request	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Friends Page 3. Press the friends icon 4. Press the friend request 5. Press Accept Friend Request 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Accept or Reject buttons will change to Unfriend button. 2. A message of 'You accepted the friend request from sender name' will be displayed. 3. Friends Page will show the new friend user. 	<ol style="list-style-type: none"> 1. Accept or Reject buttons will change to Unfriend button. 2. A message of 'You accepted the friend request from sender name' will be displayed. 3. Friends Page will show the new friend user.

Table 7.9: System Test Case of Accept Friend Request

Test Case ID:	8	
Test Case:	Reject Friend Request	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Friends Page 3. Press the friends icon 4. Press the friend request 5. Press Reject Friend Request 	

Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Accept or Reject buttons will change to Unfriend button. 2. A message of 'You rejected the friend request from sender name' will be displayed. 	<ol style="list-style-type: none"> 1. Accept or Reject buttons will change to Unfriend button. 2. A message of 'You rejected the friend request from sender name' will be displayed.

Table 7.10: System Test Case of Reject Friend Request

Test Case ID:	9	
Test Case:	Unfriend	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Go to Friends Page 3. Press any one of the friends 4. Press Unfriend 5. Press Confirm 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Unfriend button will change to Send Friend Request button. 2. A message of 'You removed friend name from your friend list in this app' will be displayed. 3. Friends Page won't show the unfriended user. 	<ol style="list-style-type: none"> 1. Unfriend button will change to Send Friend Request button. 2. A message of 'You removed friend name from your friend list in this app' will be displayed. 3. Friends Page won't show the unfriended user.

Table 7.11: System Test Case of Unfriend

Test Case ID:	10
Test Case:	Set Time-based Alarm
Test Steps:	<ol style="list-style-type: none"> 1. Press Calendar 2. Choose Date, Time

	3. Add Message (Optional) 4. Press Set Alarm	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Time alarm added successfully' will appear. 2. Home Page will show the activated time-based alarm. 3. Time Alarm will ring when the date is matched and the time is up. 	<ol style="list-style-type: none"> 1. A message of 'Time alarm added successfully' will appear. 2. Home Page will show the activated time-based alarm. 3. Time Alarm will ring when the date is matched and the time is up.

Table 7.12: System Test Case of Set Time-based Alarm

Test Case ID:	11	
Test Case:	Update Time-based Alarm	
Test Steps:	<ol style="list-style-type: none"> 1. Press one of the existing time-based alarms 2. Change Date, Time 3. Change Message (Optional) 4. Press Update Alarm 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Time alarm updated successfully' will appear. 2. Home Page will show the activated time-based alarm with updated info. 3. Time Alarm will ring when the date is matched and the time is up. 	<ol style="list-style-type: none"> 1. A message of 'Time alarm updated successfully' will appear. 2. Home Page will show the activated time-based alarm with updated info. 3. Time Alarm will ring when the date is matched and the time is up.

Table 7.13: System Test Case of Update Time-based Alarm

Test Case ID:	12
---------------	----

Test Case:	Delete Time-based Alarm	
Test Steps:	<ol style="list-style-type: none"> 1. Swipe one of the existing time-based alarms to left side 2. Press the dustbin icon 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Time alarm deleted successfully' will appear. 2. Home Page will not display the time-based alarm. 	<ol style="list-style-type: none"> 1. A message of 'Time alarm deleted successfully' will appear. 2. Home Page will not display the time-based alarm.

Table 7.14: System Test Case of Delete Time-based Alarm

Test Case ID:	13	
Test Case:	Set Location-based Alarm	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Long Press on the map 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Location alarm added successfully' will appear. 2. A marker with a circle will be dropped on the map. 3. Home Page will show the activated location-based alarm. 4. Location Alarm will ring when the user is inside the marker's circle. 	<ol style="list-style-type: none"> 1. A message of 'Location alarm added successfully' will appear. 2. A marker with a circle will be dropped on the map. 3. Home Page will show the activated location-based alarm. 4. Location Alarm will ring when the user is inside the marker's circle.

Table 7.15: System Test Case of Set Location-based Alarm

Test Case ID:	14
Test Case:	Update Location-based Alarm

Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Press one of the Markers 3. Change the radius 4. Press apply OR <ol style="list-style-type: none"> 1. Press one of the existing location-based alarms 2. Change the radius 3. Press apply 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Location alarm updated successfully' will appear. 2. The marker's circle size will change on the map. 3. Home Page will show the activated location-based alarm with updated info. 4. Location Alarm will ring when the user is inside the marker's circle. 	<ol style="list-style-type: none"> 1. A message of 'Location alarm updated successfully' will appear. 2. The marker's circle size will change on the map. 3. Home Page will show the activated location-based alarm with updated info. 4. Location Alarm will ring when the user is inside the marker's circle.

Table 7.16: System Test Case of Update Location-based Alarm

Test Case ID:	15	
Test Case:	Delete Location-based Alarm	
Test Steps:	<ol style="list-style-type: none"> 1. Swipe one of the existing location-based alarms to left side 2. Press the dustbin icon OR <ol style="list-style-type: none"> 1. Press Location 2. Press one of the Markers 3. Press the dustbin icon 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Location alarm deleted successfully' will appear. 	<ol style="list-style-type: none"> 1. A message of 'Location alarm deleted successfully' will appear.

	2. Home Page will not display the location-based alarm.	2. Home Page will not display the location-based alarm.
--	---	---

Table 7.17: System Test Case of Delete Location-based Alarm

Test Case ID:	16	
Test Case:	Search Location and Go to My Location	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Fill in search bar with place name 3. Press the search result 4. Press the location icon (black) 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. If step 3 is executed, the map will pan to the location of the searched place. 2. If step 4 is executed, the map will pan back to the user current location. 	<ol style="list-style-type: none"> 1. If step 3 is executed, the map will pan to the location of the searched place. 2. If step 4 is executed, the map will pan back to the user current location.

Table 7.18: System Test Case of Search Location and Go to My Location

Test Case ID:	17	
Test Case:	Show/Hide Friend Shared Location	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Press the enable eye icon (white) 3. Press the disable eye icon (gray) 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. If step 2 is executed, friends' avatars will be shown on the map if the user's friends have shared their locations and the enable eye icon will change to disable eye icon. 	<ol style="list-style-type: none"> 1. If step 2 is executed, friends' avatars will be shown on the map if the user's friends have shared their locations and the enable eye icon will change to disable eye icon.

	2. If step 3 is executed, friends' avatars will not be shown on the map and the disable eye icon will change to enable eye icon.	2. If step 3 is executed, friends' avatars will not be shown on the map and the disable eye icon will change to enable eye icon.
--	--	--

Table 7.19: System Test Case of Show/Hide Friend Shared Location

Test Case ID:	18	
Test Case:	Enable/Disable Share Location	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Press the enable cloud icon (blue) 3. Press the disable cloud icon (red) 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. If step 2 is executed, a message of 'You updated and shared your current location to your friends' will appear and the enable cloud icon will change to disable cloud icon. 2. If step 3 is executed, a message of 'You cleared the last shared location and stopped sharing' will appear and the disable cloud icon will change to enable cloud icon. 3. The friends can see the user's shared location after pressing the enable eye icon. 	<ol style="list-style-type: none"> 1. If step 2 is executed, a message of 'You updated and shared your current location to your friends' will appear and the enable cloud icon will change to disable cloud icon. 2. If step 3 is executed, a message of 'You cleared the last shared location and stopped sharing' will appear and the disable cloud icon will change to enable cloud icon. 3. The friends can see the user's shared location after pressing the enable eye icon.

Table 7.20: System Test Case of Enable/Disable Share Location

Test Case ID:	19
---------------	----

Test Case:	Filter Location Alarm with Favourite Weekdays	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Press one of the Markers 3. Press the star icon (yellow) 4. Choose weekdays 5. Press Save Preference 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. Back to Location Page. 2. Activated Location Alarm will ring when the user is inside the marker's circle and test day is matched with the selected weekday. 	<ol style="list-style-type: none"> 1. Back to Location Page. 2. Activated Location Alarm will ring when the user is inside the marker's circle and test day is matched with the selected weekday

Table 7.21: System Test Case of Filter Location Alarm with Favourite Weekdays

Test Case ID:	20	
Test Case:	Trigger Location Alarm to Chosen Friends	
Test Steps:	<ol style="list-style-type: none"> 1. Press Location 2. Press one of the Markers 3. Press the message icon (green) 4. Choose friends 5. Press Save Notification 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Fail	<ol style="list-style-type: none"> 1. Back to Location Page. 2. Activated Location Alarm will ring when the user is inside the marker's circle and the alarm will notify the selected friends too. 	<ol style="list-style-type: none"> 1. Back to Location Page. 2. Activated Location Alarm will ring when the user is inside the marker's circle and the alarm will notify the selected friends too.

Table 7.22: System Test Case of Trigger Location Alarm to Chosen Friends

Test Case ID:	21
---------------	----

Test Case:	Settings	
Test Steps:	<ol style="list-style-type: none"> 1. Press Menu 2. Press Settings 3. Select Ringtone 4. Select Ring Duration 5. Change Ring Vibration 6. Change Do Not Disturb 7. Press Save Settings 	
Test Result	Expected Result	Actual Result
Android: Pass iOS: Pass	<ol style="list-style-type: none"> 1. A message of 'Settings have been updated successfully' will be displayed. 2. Time Alarm and Location Alarm will play the selected ringtone. 3. Time Alarm and Location Alarm will ring with the desired Ring Duration. 4. Phone will vibrate if Ring Vibration is turned on and the app is ringing on the foreground. 5. Users will not receive friends' Location Alarm notifications if Do Not Disturb is turned on. 	<ol style="list-style-type: none"> 1. A message of 'Settings have been updated successfully' will be displayed. 2. Time Alarm and Location Alarm will play the selected ringtone. 3. Time Alarm and Location Alarm will ring with the desired Ring Duration. 4. Phone will vibrate if Ring Vibration is turned on and the app is ringing on the foreground. 5. Users will not receive friends' Location Alarm notifications if Do Not Disturb is turned on.

Table 7.23: System Test Case of Settings

7.4 Usability Testing

Usability Testing is conducted by distributing the User Satisfaction Survey to 3 novices and 3 experts respectively. After they tested the Smart Location Alarm application, they will fill up the survey by providing their honest ratings and opinions for the application. Then, the SUS score will be calculated out by using the formula in section 7.4.2.

7.4.1 Usability Testing Survey

	Strongly Disagree 1	2	3	4	Strongly Agree 5
1. I think that I would like to use this mobile application for reminding purpose.					
2. I found the mobile application unnecessarily complex.					
3. I thought the mobile application was easy to use					
4. I think that I would need the support of a technical person to be able to use this mobile application.					
5. I found this mobile application was easily moved through without a lot of					

backtracking or data re-entry.					
6. I thought there was too much inconsistency in this mobile application.					
7. I would imagine that most people would learn to use this mobile application very quickly.					
8. I found the mobile application very awkward to use.					
9. I felt very confident using the mobile application.					
10. I needed to learn a lot of things before I could get going with this mobile application.					

Table 7.24: User Satisfaction Survey (“System Usability Scale”, Brooke, 1986)

What did you like best
about the application?

What did you like least
about the application?

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

Do you have any other final
comments or questions?

7.4.2 Usability Testing Result

Questions	Evaluators						Average
	Novice 1	Novice 2	Novice 3	Expert 1	Expert 2	Expert 3	
1.	4	4	3	3	4	2	3.3/4.0
2.	2	3	4	1	4	2	2.7/4.0
3.	2	3	4	3	3	4	3.2/4.0
4.	4	4	4	3	3	4	3.7/4.0
5.	4	4	4	4	4	3	3.8/4.0
6.	4	4	4	4	4	3	3.8/4.0
7.	3	3	2	3	3	3	2.8/4.0
8.	4	4	3	3	4	2	3.3/4.0
9.	3	4	4	4	4	4	3.8/4.0

10.	4	4	4	3	4	4	3.8/4.0
Total	34	37	36	31	37	31	34.3/40.0
SUS score	85.0	92.5	90.0	77.5	92.5	77.5	85.8

Table 7.25: Usability Testing Result

This satisfaction user survey scoring system is adapted from System Usability Scale, Brooke, J. (1986). The formula of SUS score calculation is as below:

1. For odd numbered questions, the user response is subtracted by 1.
2. For even-numbered questions, the user response is subtracted from 5.
3. This scales all values from 0 to 4 with 4 being the most positive response.
4. The total of the converted responses is multiplied by 2.5, so that the range of possible values is from 0 to 100 instead of from 0 to 40.

The responses of the User Satisfaction Survey are attached as Appendix E.

7.5 User Acceptance Testing

User Acceptance Testing is the final phase of the software testing in this project before the application is ready to be released. A minimum target of 10 users are needed to fully participate in the User Acceptance Testing. Their feedback can verify the functionalities of the application.

7.5.1 UAT Survey

The questions of the UAT Survey are attached as Appendix F.

7.5.2 UAT Result

The responses of the UAT Survey are attached as Appendix G.

CHAPTER 8

CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions

Smart Location Alarm is a cross-platform mobile application that allows users to be reminded when they reach certain locations or the time is up. Therefore, the first and second objectives are achieved as the time alarm and location alarm reminding features work like a charm. For the third objective, different users' ratings and feedback have been collected from Usability Testing, User Acceptance Testing and Google Play Store, the deployment platform. Hence, the application can be measured and evaluated based on them. Last but not least, the application that has been published to Google Play Store successfully is applying the minimum viable product concept. As a result, the users are able to give feedback about the application in many perspectives like the UI, features, performance and so on, as soon as possible. The ratings and reviews for the application on the deployment platform are attached as Appendix H.

8.2 Recommendations

Despite Smart Location Alarm application fulfills many specifications and has been deployed to the public, there are still some limitations in this project. Table below lists out the limitations with justifications and suggestions for the current situation and the future work.

Limitations	Justifications	Suggestions
Although the application is cross-platform, it can only be run on Android devices but not iOS devices.	An iOS signing certificate for push notification requires expensive enrollment in Apple Developer Program Membership, which is up to RM412. Thus, system testing of trigger location alarm to chosen friends	Xcode simulator is used to demonstrate the application. In future, it will be better to pay for the program's fee, then the application can be used on the real iOS devices.

	failed on iOS platform and the application cannot be tested on iOS devices.	
Users are able to spam the push notifications to their friends.	As the users can set the location at anywhere and send push notifications to their friends anytime when they reach the destination.	Do not disturb mode is implemented in the application that allows the users to not receive the notifications from their friends. However, it is still not the best way to solve it. In future, it is better to restrict the users from keep sending the push notifications to friends instead of stop receiving the push notifications from friends.
Route navigation, estimation of distance and time to arrival and other useful features are not implemented.	As these features might require more plugins and integrate more services, which increase chances of causing project build failure. Besides, they might be high battery draining and time consuming features.	Minimum viable product is developed for now. In future, the application should implement more features as well as battery optimization is highly recommended.
Sequence Diagrams are not provided, Unit Testing and Integration Testing are not complete.	Due to the time constraint and complexity of the code in this project, Class Diagram is provided for Object-oriented design only and the test code is focusing on creating the pages as well as the forms validations only.	System Testing that is not planned in the WBS is provided, which is totally in contrast with the Sequence Diagrams. Besides, writing more test code to test for each important feature is strongly advised.

Table 8.1: Project Limitations, Justifications and Suggestions

REFERENCES

- Agile Model*. [online]. Available at: <<https://www.hiteshi.com/agile-vs-waterfall/>> [Accessed 19 March 2020].
- Altameem, E. (2015) 'Impact of Agile Methodology on Software Development', *Computer and Information Science*, 8(2). doi: 10.5539/cis.v8n2p9.
- Borick, S. (2018) 'Comparing the Usage of React Native and Ionic'.
- Brezočnik, L. and Majer, Č. (2016) 'Comparison of agile methods: Scrum, Kanban, and Scrumban', *Proceedings of the 19th International Multiconference Information Society*, C(July), pp. 1–5. Available at: <https://www.researchgate.net/publication/326441463>.
- Brooke, J. (1986). 'SUS: a quick and dirty usability scale'. In P. W. Jordan; B. Thomas; B. A. Weerdmeester; A. L. McClelland (eds.). *Usability Evaluation in Industry*. London: Taylor and Francis.
- Casteren, W. Van (2017) 'The Waterfall Model And The Agile Methodologies : A Comparison By Project Characteristics-Short The Waterfall Model and Agile Methodologies', *Academic Competences in the Bachelor*, (February), pp. 10–13. doi: 10.13140/RG.2.2.10021.50403.
- Eder, M. S. (2015) 'Tsada-Mobiminder: A Location Based Alarm Mobile Reminder', *International Journal of Recent Contributions from Engineering, Science & IT (iJES)*, 3(4), p. 43. doi: 10.3991/ijes.v3i4.5226.
- Gechman, M. (2019) 'Software Development Methodologies', *Project Management of Large Software-Intensive Systems*, (August 2013), pp. 49–66. doi: 10.1201/9780429027932-4.

Innes, E. (2013). '*Digital distraction' causes 20 million passengers to miss their bus or train stop every year.* [online]. Available at:
 <<https://www.dailymail.co.uk/sciencetech/article-2341068/Digital-distraction-causes-20-million-passengers-miss-bus-train-stop-year.html>> [Accessed 20 February 2020].

Kanban Board. [online]. Available at:
 <<https://sites.google.com/site/wcfpandu/what-is-kanban?tmpl=%2Fsystem%2Fapp%2Ftemplates%2Fprint%2F>> [Accessed 22 February 2020].

Nethra, V. et al. (2019) 'Location Based Task Reminder Android Application', (September).

Ohyver, M. et al. (2019) 'The comparison firebase realtime database and MySQL database performance using wilcoxon signed-rank test', *Procedia Computer Science*, 157, pp. 396–405. doi: 10.1016/j.procs.2019.08.231.

Patil, V., Panicker, S. and Kv, M. (2016) 'Use of Agile Methodology for Mobile Applications', *International Journal of Latest Technology in Engineering, Management & Applied Science*, 5(10), pp. 73–77.

Rashid, F. and Jhawari, A. (2018) 'Location based alarm using mobile device', *International Journal of Computer Applications*, (January), pp. 975–8887. Available at: <https://research.ijcaonline.org/mic/number1/mic1411.pdf>.

Waterfall Model. [online]. Available at:
 <<https://www.javatpoint.com/jira-waterfall-model>> [Accessed 19 March 2020].

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

APPENDICES

APPENDIX A: Online Survey (Questions)

Online Survey - Smart Location Alarm on Mobile Platform

I would like to thank you for taking part in this survey on 'Smart Location Alarm on Mobile Platform' as I am required to do this survey for my final year project.

In this project, I would like to find out the purpose and gather more information about the requirements and features for a mobile application called 'Smart Location Alarm'.

It is a time-based and location-based reminder application that used both countdown timer/calendar and Global Position System (GPS) technology to track and notify user.

**Required*

1. Are you a student, a working adult or a senior citizen? *

Mark only one oval.

- Student
 Working Adult
 Senior Citizen

2. Which mobile platform are you using now? *

Mark only one oval.

- Android (Google)
 iPhone (Apple)
 Both Android and iPhone
 Other: _____

3. Have you ever used a reminder application? *

Mark only one oval.

- Yes
 No

7. Which function do you think is the most important for a reminder application? *

Mark only one oval.

- Notify user when user reached or near to the destination
- Suggest places to go based on user rating or nearest distance
- Able to add friends and check for their current locations if they enabled location sharing
- Customizable alarm music or alert messages
- Remind user when there is an upcoming event or festival
- Other: _____

8. Assume that you have installed this application, in what condition or situation will you use this application? *

9. Please suggest one important or interesting feature that you think 'Smart Location Alarm' should have included. *

This content is neither created nor endorsed by Google.



APPENDIX B: Online Survey (Responses)

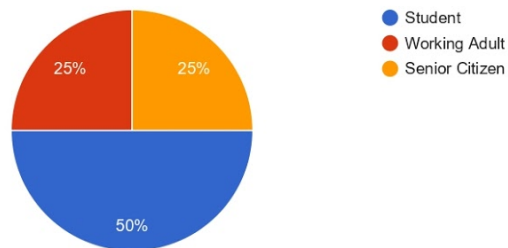
Online Survey - Smart Location Alarm on Mobile Platform

40 responses

[Publish analytics](#)

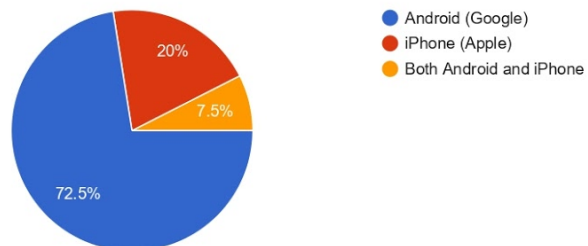
Are you a student, a working adult or a senior citizen?

40 responses



Which mobile platform are you using now?

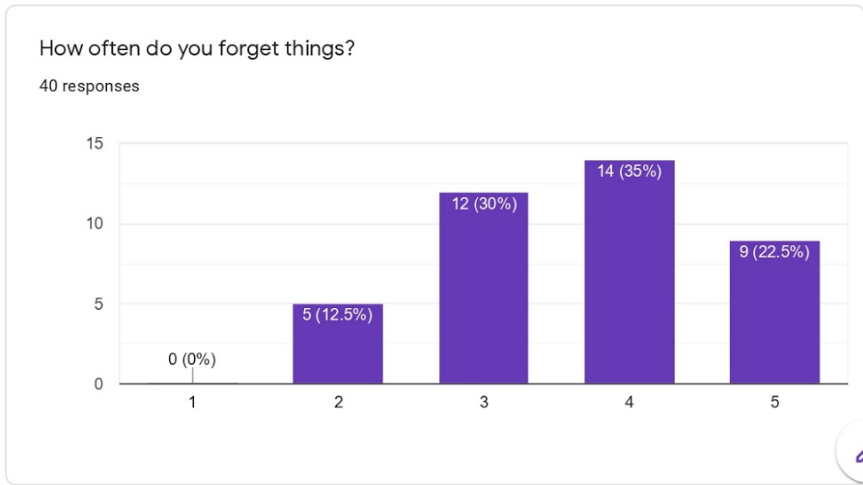
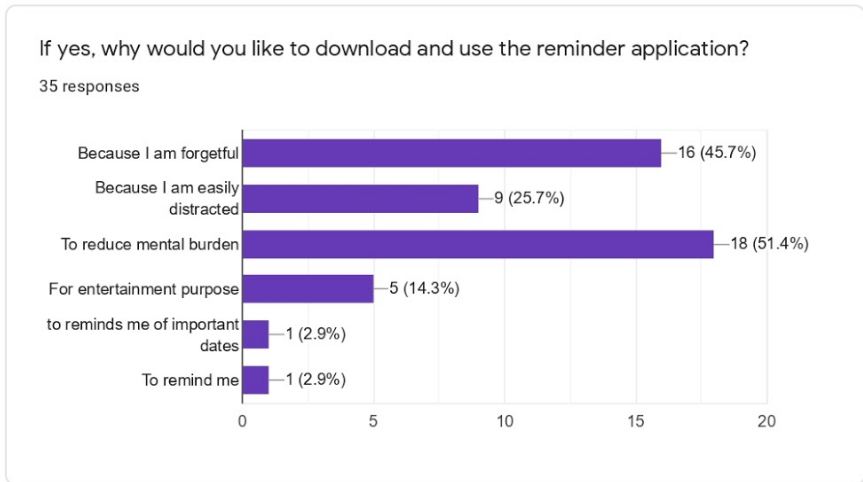
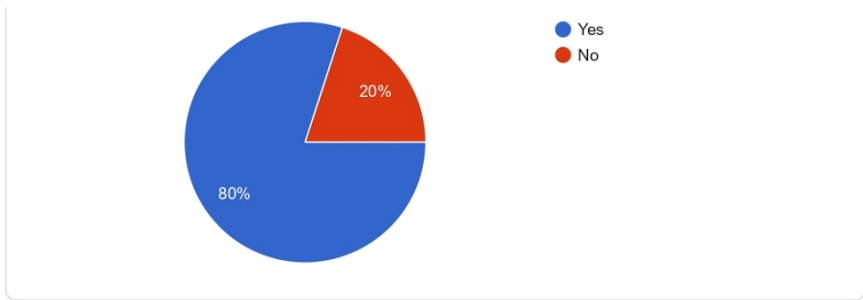
40 responses



Have you ever used a reminder application?

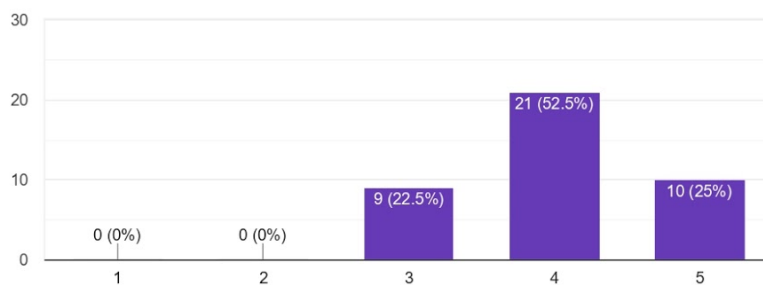
40 responses





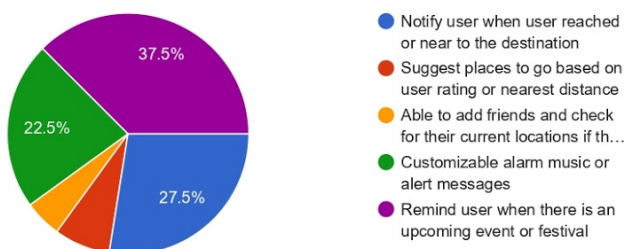
To what extent do you agree that it is necessary for every student, working adult and senior citizen to use a reminder application?

40 responses



Which function do you think is the most important for a reminder application?

40 responses



Assume that you have installed this application, in what condition or situation will you use this application?

40 responses

when exam is near and preparation and schedule need to distribute accordingly.

Remind me to participate some events

For studying and meeting



While taking note of something important

Sleeping in public transport

For working and personal usage

for event/place that i'm interested but still thinking whether want to go or not

To remind me when and where is the event that i am interested.

to record some event that at far location

Please suggest one important or interesting feature that you think 'Smart Location Alarm' should have included.

40 responses

notification alarm tone should be loud and clear.

Includes many locations and music alarms

Automatic tell the user about the situation and info of the destination

Customizing alarm for issues of different level of importance

Can ring in background

Will remain notify if the user didnt read the reminder

link with social media accounts so that i can create reminder straightaway if needed.

Private location/ event sharing

Must detect current location accurate

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Privacy Policy](#).

Google Forms

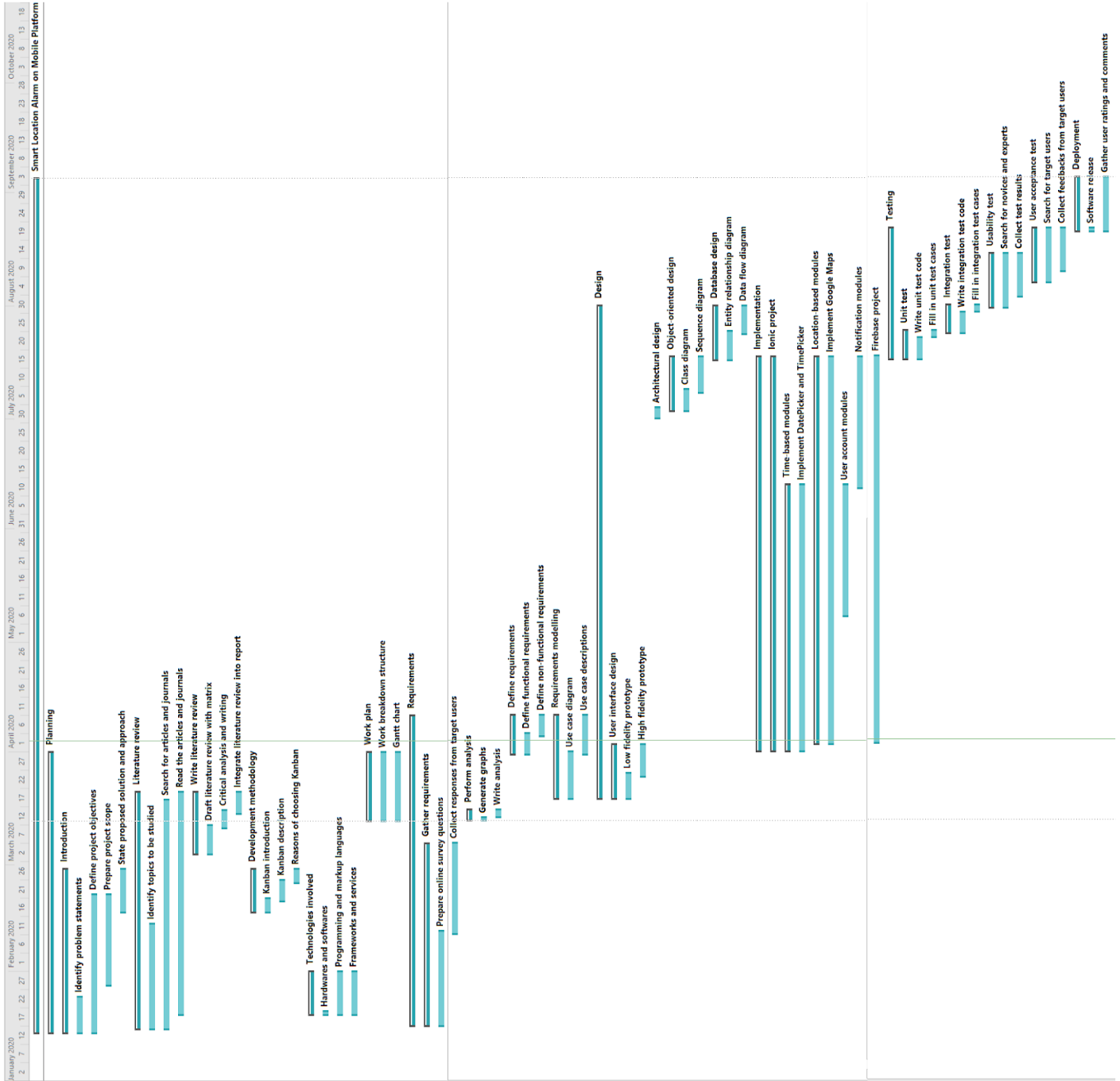


APPENDIX C: Work Breakdown Structure

Task Name	Duration	Start	Finish
1 Smart Location Alarm on Mobile Platform	234 days	January 15, 2020	September 4, 2020
1.1 Planning	77 days	January 15, 2020	March 31, 2020
1.1.1 Introduction	45 days	January 15, 2020	February 28, 2020
1.1.1.1 Identify problem statements	10 days	January 15, 2020	January 24, 2020
1.1.1.2 Define project objectives	38 days	January 15, 2020	February 21, 2020
1.1.1.3 Prepare project scope	25 days	January 28, 2020	February 21, 2020
1.1.1.4 State proposed solution and approach	12 days	February 17, 2020	February 28, 2020
1.1.2 Literature review	65 days	January 16, 2020	March 20, 2020
1.1.2.1 Identify topics to be studied	29 days	January 16, 2020	February 13, 2020
1.1.2.2 Search for articles and journals	63 days	January 16, 2020	March 18, 2020
1.1.2.3 Read the articles and journals	61 days	January 20, 2020	March 20, 2020
1.1.2.4 Write literature review	17 days	March 4, 2020	March 20, 2020
1.1.2.4.1 Draft literature review with matrix	8 days	March 4, 2020	March 11, 2020
1.1.2.4.2 Critical analysis and writing	5 days	March 11, 2020	March 15, 2020
1.1.2.4.3 Integrate literature review into report	6 days	March 15, 2020	March 20, 2020
1.1.3 Development methodology	12 days	February 17, 2020	February 28, 2020
1.1.3.1 Kanban introduction	4 days	February 17, 2020	February 20, 2020
1.1.3.2 Kanban description	6 days	February 20, 2020	February 25, 2020
1.1.3.3 Reasons of choosing Kanban	4 days	February 25, 2020	February 28, 2020
1.1.4 Technologies involved	12 days	January 20, 2020	January 31, 2020
1.1.4.1 Hardwares and softwares	1 day	January 20, 2020	January 20, 2020
1.1.4.2 Programming and markup languages	12 days	January 20, 2020	January 31, 2020
1.1.4.3 Frameworks and services	12 days	January 20, 2020	January 31, 2020
1.1.5 Work plan	19 days	March 13, 2020	March 31, 2020
1.1.5.1 Work breakdown structure	19 days	March 13, 2020	March 31, 2020
1.1.5.2 Gantt chart	19 days	March 13, 2020	March 31, 2020
1.2 Requirements	85 days	January 17, 2020	April 10, 2020
1.2.1 Gather requirements	50 days	January 17, 2020	March 6, 2020
1.2.1.1 Prepare online survey questions	26 days	January 17, 2020	February 11, 2020
1.2.1.2 Collect responses from target users	25 days	February 11, 2020	March 6, 2020
1.2.2 Perform analysis	3 days	March 13, 2020	March 15, 2020
1.2.2.1 Generate graphs	1 day	March 13, 2020	March 13, 2020
1.2.2.2 Write analysis	2 days	March 14, 2020	March 15, 2020
1.2.3 Define requirements	11 days	March 31, 2020	April 10, 2020
1.2.3.1 Define functional requirements	6 days	March 31, 2020	April 5, 2020
1.2.3.2 Define non-functional requirements	6 days	April 5, 2020	April 10, 2020
1.2.4 Requirements modelling	23 days	March 19, 2020	April 10, 2020
1.2.4.1 Use case diagram	13 days	March 19, 2020	March 31, 2020
1.2.4.2 Use case descriptions	11 days	March 31, 2020	April 10, 2020
1.3 Design	135 days	March 19, 2020	July 31, 2020
1.3.1 User interface design	15 days	March 19, 2020	April 2, 2020
1.3.1.1 Low fidelity prototype	7 days	March 19, 2020	March 25, 2020
1.3.1.2 High fidelity prototype	9 days	March 25, 2020	April 2, 2020
1.3.2 Architectural design	3 days	July 1, 2020	July 3, 2020
1.3.3 Object-oriented design	15 days	July 3, 2020	July 17, 2020
1.3.3.1 Class diagram	6 days	July 3, 2020	July 8, 2020
1.3.3.2 Sequence diagram	10 days	July 8, 2020	July 17, 2020
1.3.4 Database design	15 days	July 17, 2020	July 31, 2020
1.3.4.1 Entity relationship diagram	8 days	July 17, 2020	July 24, 2020
1.3.4.2 Data flow diagram	8 days	July 24, 2020	July 31, 2020
1.4 Implementation	108 days	April 1, 2020	July 17, 2020
1.4.1 Ionic project	108 days	April 1, 2020	July 17, 2020
1.4.1.1 Time-based modules	73 days	April 1, 2020	June 12, 2020
1.4.1.1.1 Implement DatePicker and TimePicker	73 days	April 1, 2020	June 12, 2020
1.4.1.2 Location-based modules	106 days	April 3, 2020	July 17, 2020
1.4.1.2.1 Implement Google Maps	106 days	April 3, 2020	July 17, 2020
1.4.1.3 User account modules	36 days	May 8, 2020	June 12, 2020
1.4.1.4 Notification modules	36 days	June 12, 2020	July 17, 2020
1.4.2 Firebase project	106 days	April 3, 2020	July 17, 2020

▸ 1.5 Testing	36 days	July 17, 2020	August 21, 2020
▸ 1.5.1 Unit test	8 days	July 17, 2020	July 24, 2020
1.5.1.1 Write unit test code	6 days	July 17, 2020	July 22, 2020
1.5.1.2 Fill in unit test cases	2 days	July 23, 2020	July 24, 2020
▸ 1.5.2 Integration test	8 days	July 24, 2020	July 31, 2020
1.5.2.1 Write integration test code	6 days	July 24, 2020	July 29, 2020
1.5.2.2 Fill in integration test cases	2 days	July 30, 2020	July 31, 2020
▸ 1.5.3 Usability test	15 days	July 31, 2020	August 14, 2020
1.5.3.1 Search for novices and experts	15 days	July 31, 2020	August 14, 2020
1.5.3.2 Collect test results	12 days	August 3, 2020	August 14, 2020
▸ 1.5.4 User acceptance test	15 days	August 7, 2020	August 21, 2020
1.5.4.1 Search for target users	15 days	August 7, 2020	August 21, 2020
1.5.4.2 Collect feedbacks from target users	12 days	August 10, 2020	August 21, 2020
▸ 1.6 Deployment	15 days	August 21, 2020	September 4, 2020
1.6.1 Software release	1 day	August 21, 2020	August 21, 2020
1.6.2 Gather user ratings and comments	15 days	August 21, 2020	September 4, 2020

APPENDIX D: Gantt Chart



APPENDIX E: User Satisfaction Survey (Responses)

Novice 1

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

9. I felt very confident using the mobile application.				/	
10. I needed to learn a lot of things before I could get going with this mobile application.	/				

What did you like best about the application?

Notify friends when reaching a place without having to call them, which is dangerous during driving.

What did you like least about the application?

The ability of spamming notifications to friends.

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

An application that provides convenience to the user.

Do you have any other final comments or questions?

No.

Novice 2

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

9. I felt very confident using the mobile application.					/
10. I needed to learn a lot of things before I could get going with this mobile application.	/				

What did you like best about the application?

Able to set alarm in different locations.

What did you like least about the application?

Nothing special about the time alarm.

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

No.

Do you have any other final comments or questions?

No.

Novice 3

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

9. I felt very confident using the mobile application.					/
10. I needed to learn a lot of things before I could get going with this mobile application.	/				

What did you like best about the application?

Ringtone settings.

What did you like least about the application?

No.

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

Interesting application that can track and alert the user.

Do you have any other final comments or questions?

No.

Expert 1

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

9. I felt very confident using the mobile application.					/
10. I needed to learn a lot of things before I could get going with this mobile application.		/			

What did you like best about the application?

The ability to track and show the location of your friends you added throughout the map.

What did you like least about the application?

It is better not to show telephone number and address in profile before adding he or she as a friend.

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

No.

Do you have any other final comments or questions?

No.

Expert 2

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

9. I felt very confident using the mobile application.					/
10. I needed to learn a lot of things before I could get going with this mobile application.	/				

What did you like best about the application?

The ability to show the location of your friends you added and notify you when they are within range.

What did you like least about the application?

The calendar feature as my phone already have in built calendar and notification function.

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

An easy to use app that can let the driver notify the person that located at the destination that he is within range automatically.

Do you have any other final comments or questions?

No.

Expert 3

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

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

What did you like best about the application?

Tracking system of the user in a certain place. It will be useful for tracking worker whether they come to work or not therefore does not require extra machine that check for the attendance of the workers.

What did you like least about the application?

Method to set alarm. It will be better if there are voice control function that can change the location of interest without having to set it by hand especially while driving.

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

Nice and smart app that can be used to replace calling and messaging to notify people.

Do you have any other final comments or questions?

An app which have infinite potential that could be develop further into a more functional and useful communication social media.

APPENDIX F: UAT Survey (Questions)

User Acceptance Testing Survey - Smart Location Alarm

I would like to express my gratitude to you for taking part in the User Acceptance Testing of Smart Location Alarm application. Through this final stage testing, I would like to gather valuable feedback from you, whether the process flow of the application are going well on your mobile device.

***Required**

Personal Details

This section is optional.

1. Your Name:

2. Your Age:

3. Device Name:

User Acceptance Testing
Survey - Smart Location Alarm

I would like to express my gratitude to you for taking part in the User Acceptance Testing of Smart Location Alarm application. Through this final stage testing, I would like to gather valuable feedback from you, whether the process flow of the application are going well on your mobile device.

UAT Questions

This section is compulsory.

Register Page, Login Page and Profile Page

4. Register an account successfully. *

Mark only one oval.

Pass

Fail

5. Login and logout successfully. *

Mark only one oval.

Pass

Fail

6. Edit profile details successfully. *

Mark only one oval.

- Pass
 Fail

7. Upload a new avatar or profile picture successfully. *

Mark only one oval.

- Pass
 Fail

Friends Page, Friend Profile Page

8. Search user(s) and add friend(s) successfully. *

Mark only one oval.

- Pass
 Fail

Calendar Page and Location Page

9. Set time alarm(s) and ring on correct date and time successfully. *

Mark only one oval.

- Pass
 Fail

10. Set location alarm(s) and ring on correct destination successfully. *

Mark only one oval.

- Pass
 Fail

11. Update time or location alarm(s) successfully. *

Mark only one oval.

- Pass
 Fail

12. Delete time or location alarm(s) successfully. *

Mark only one oval.

- Pass
 Fail

13. Search place(s) and move to the selected place on the map successfully. *

Mark only one oval.

- Pass
 Fail

14. Show and hide friend shared location(s) successfully. (Your friend(s) must enable sharing location(s) first) *

Mark only one oval.

- Pass
 Fail

15. Enable and disable sharing location to friends successfully. (Your friends must be able to see your shared location) *

Mark only one oval.

- Pass
 Fail

16. Location alarm rings on your selected weekday(s) and not rings on the non-selected weekday(s) successfully. (Must saved the weekdays to notify first) *

Mark only one oval.

- Pass
 Fail

17. Location alarm notifies your chosen friend(s) or you received notification(s) from your friend(s) successfully. (Must saved the friends to notify first) *

Mark only one oval.

- Pass
 Fail

Home Page

18. Activate and deactivate the alarm(s) successfully. *

Mark only one oval.

- Pass
 Fail

Settings Page

19. Ringtone music, duration, vibration and do not disturb mode can be changed successfully. *

Mark only one oval.

- Pass
 Fail

Others

20. Satisfaction level for the application. *

Mark only one oval.

	1	2	3	4	5	
Not satisfied	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very satisfied

21. Comment or feedback for the application. *

APPENDIX G: UAT Survey (Responses)

User Acceptance Testing Survey - Smart Location Alarm

10 responses

[Publish analytics](#)

Personal Details

Your Name:

10 responses

Leon Kang

Jennifer

Siew Keok Ying

Loh Yi Hao

IC Liew

Lai Jia Qi

Lim Chiah Chern

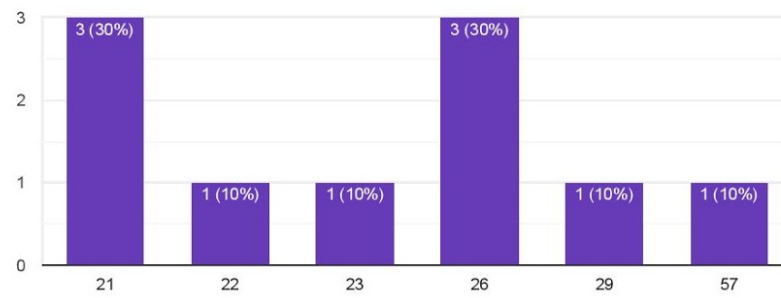
You Xu Bin

Fu Chu Bin



Your Age:

10 responses



Device Name:

10 responses

Redmi 5

samsung

Redmi Note 7

Xiaomi Mix2s

Samsung A51

Samsung Note 10 lite

Honor play

Huawei Nova 2i

Samsung Note 10

User Acceptance Testing Survey - Smart Location Alarm

UAT Questions

Register Page, Login Page and Profile Page



Register an account successfully.

10 responses



● Pass
● Fail

Login and logout successfully.

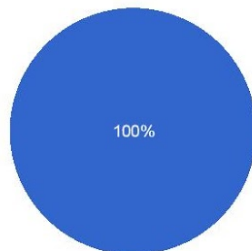
10 responses



● Pass
● Fail

Edit profile details successfully.

10 responses

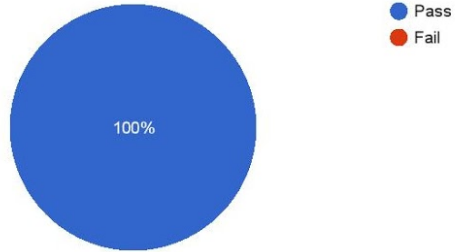


● Pass
● Fail



Upload a new avatar or profile picture successfully.

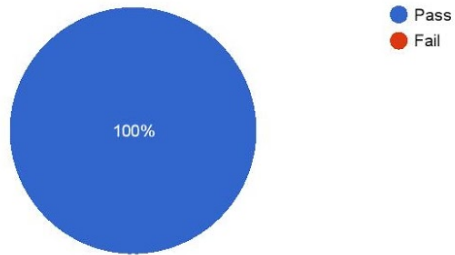
10 responses



Friends Page, Friend Profile Page

Search user(s) and add friend(s) successfully.

10 responses

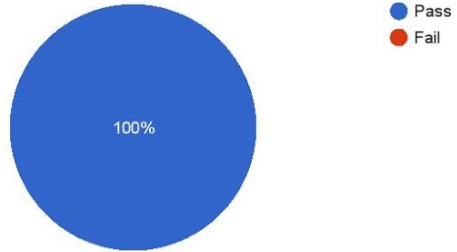


Calendar Page and Location Page



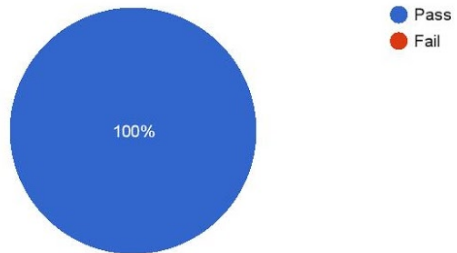
Set time alarm(s) and ring on correct date and time successfully.

10 responses



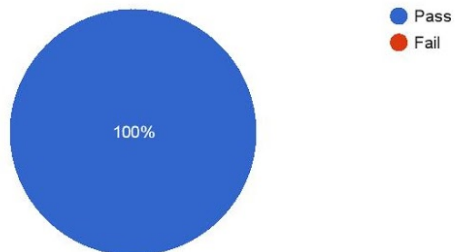
Set location alarm(s) and ring on correct destination successfully.

10 responses



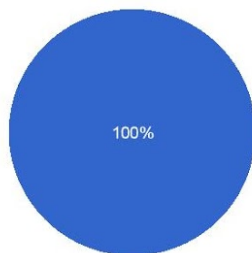
Update time or location alarm(s) successfully.

10 responses



Delete time or location alarm(s) successfully.

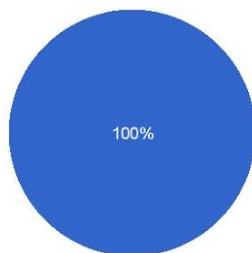
10 responses



● Pass
● Fail

Search place(s) and move to the selected place on the map successfully.

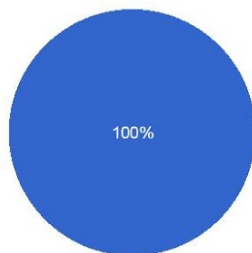
10 responses



● Pass
● Fail

Show and hide friend shared location(s) successfully. (Your friend(s) must enable sharing location(s) first)

10 responses

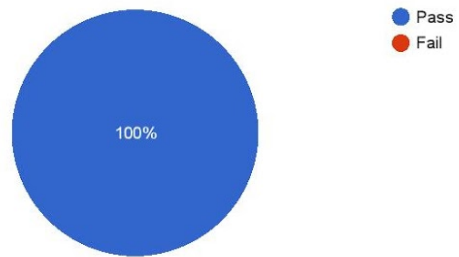


● Pass
● Fail



Enable and disable sharing location to friends successfully. (Your friends must be able to see your shared location)

10 responses



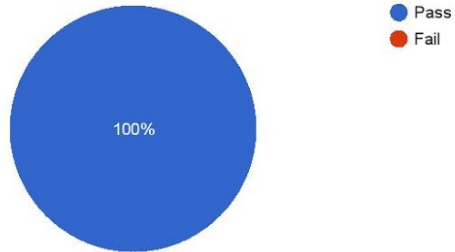
Location alarm rings on your selected weekday(s) and not rings on the non-selected weekday(s) successfully. (Must saved the weekdays to notify first)

10 responses



Location alarm notifies your chosen friend(s) or you received notification(s) from your friend(s) successfully. (Must saved the friends to notify first)

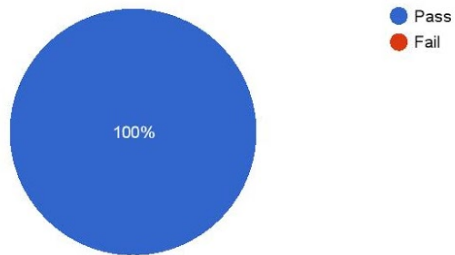
10 responses



Home Page

Activate and deactivate the alarm(s) successfully.

10 responses

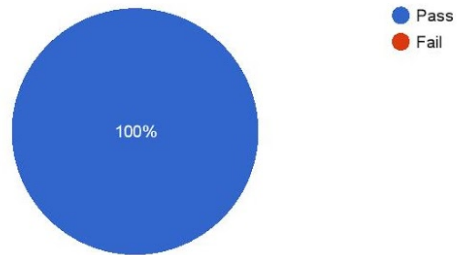


Settings Page



Ringtone music, duration, vibration and do not disturb mode can be changed successfully.

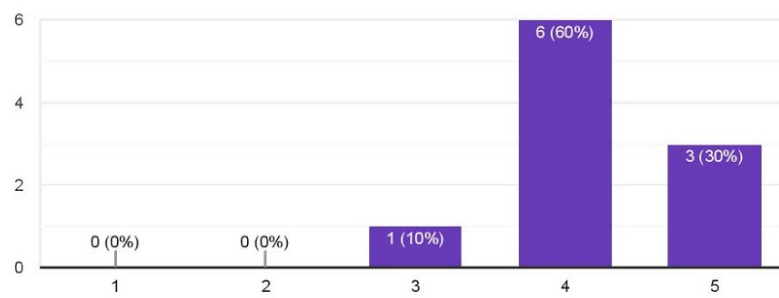
10 responses



Others

Satisfaction level for the application.

10 responses



Comment or feedback for the application.

10 responses

No comment

the search user section, guideline is not clear stated how to search,although lastly i can find lah, only develepor name. Then, for the alarm, not every minute that i want can set, the minute is fixed with range and if the time is already pass, eg. set 31 Aug, time is 3.00pm, the system still can successful add to list, this feels strange...

Excellent

Convinient to use!

Good App!

No

Useful app


Nope

This content is neither created nor endorsed by Google. [Report Abuse](#) - [Terms of Service](#) - [Privacy Policy](#)

Google Forms



APPENDIX H: Ratings and Reviews on Deployment Platform



Smart Location Alarm

LaiJy Maps & Navigation ★★★★★ 7

3+


i This app is compatible with your device.

Installed

REVIEWS Review Policy


Most relevant ▾ All devices ▾ All ratings ▾

User reviews




XB You
★★★★★ 31 August 2020

Nice app with easy to follow instruction and functions. Can be used to notify others when u reach the destination. There are some minor bugs however. Keep up the good works !!




happy sad
★★★★★ 31 August 2020

App provide tutorial with picture which is easy to understand and learn. Highly recommended for its quick notifications to selected people with simple preset setting 😊😊




Wong Phang Yong
★★★★★ 31 August 2020

Easy to use and friendly app




Loh Yi Hao
★★★★★ 3 September 2020

This app is easy to use, and is especially convenient while going to a friend's house. When the app is fully developed, I believe that it is another Waze with amazing alarm function! Hope that the developer will perfect the app as soon as possible!



Jennifer Yew
★★★★★ 31 August 2020

Interesting app!



Siew Keok Ying
★★★★★ 31 August 2020

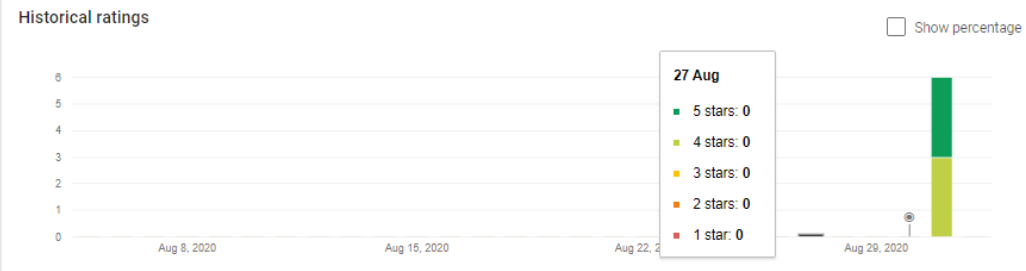
Excellent 👍👍👍👍

4.571 ★ Google Play rating ⓘ
4.571 ★ Lifetime rating ⓘ
7 Total ratings

0 ★ Your daily average
 vs peers' median
 Peers' median

Users can rate your app on Google Play with a star rating out of 5 and a written review. Users can update these at any time. [Learn more](#)

Custom peer group: 0 apps [Edit peer group](#)



Ratings breakdown

Any number of ratings ▾ Last month ▾

Country

[SEE DETAILS](#)

1. Malaysia	6 ratings	4.50 ★
-------------	-----------	--------

Language

[SEE DETAILS](#)

1. English	6 ratings	4.50 ★
------------	-----------	--------

App Version

[SEE DETAILS](#)

1. Unknown	1 rating	5.00 ★
2. 4	5 ratings	4.40 ★

Android Version

[SEE DETAILS](#)

1. Android 9	3 ratings	5.00 ★
2. Android 8.0	2 ratings	4.00 ★
3. Android 10	1 rating	4.00 ★

Device

[SEE DETAILS](#)

1. Huawei HUAWEI P20 Lite (HWANE)	1 rating	5.00 ★
2. Huawei Honor Play (HWCOR)	1 rating	5.00 ★
3. Redmi Redmi Note 7 (lavender)	1 rating	5.00 ★
4. Huawei 华为畅享8 (HWLDN-Q)	1 rating	4.00 ★
5. Huawei Mate 10 lite (HWRNE)	1 rating	4.00 ★
6. Samsung Galaxy A70 (a70q)	1 rating	4.00 ★

Operator

[SEE DETAILS](#)

1. U Mobile	2 ratings	5.00 ★
2. Maxis/Hotlink	2 ratings	4.50 ★
3. Digi	1 rating	4.00 ★
4. Celcom	1 rating	4.00 ★

Updated Ratings Lifetime ▾

Updated Ratings shows the way your ratings and reviews were updated, including the effect your replies had on those updates. [Learn more](#)

Reviews and ratings ⓘ	Returning users ⓘ	Updates to ratings ⓘ	Average rating change ⓘ
With replies	0		- *
Without replies	0		- *

Benchmarks & topics All APK versions ▾ 5 Sep 2019 – 3 Sep 2020 ▾

Identify and understand trends in your app's reviews. Review analysis features use ratings from reviews and aren't shown to users. [Learn more](#)

Benchmarks

Benchmarks display your app's ratings for a fixed set of topics and compare these ratings with similar apps in the **Maps & Navigation** category. Available for reviews written in English.

▲ Common topic ⓘ	Average rating	Rating versus peers	Number of reviews	Number versus peers	Effect on rating ⓘ
speed ⓘ	5.000 *	+1.000 *	1	9.62 x	
stability ⓘ	4.000 *	+2.058 *	1	6.17 x	
usability ⓘ	4.667 *	+0.073 *	3	17.09 x	

Topics

Topics display your app's ratings for a dynamic set of words mentioned in your app's reviews. Available for reviews written on devices using the languages shown below.

ALL ENGLISH GERMAN HINDI ITALIAN JAPANESE KOREAN PORTUGUESE SPANISH

Translated topic	Original Topic ⓘ	Average rating	Number of reviews	▲ Effect on rating ⓘ
bugs	bugs (English)	4.000 *	1	
others	others (English)	4.000 *	1	
people	people (English)	5.000 *	1	
picture	picture (English)	5.000 *	1	