MULTI OBJECTS PROXIMITY MONITORING USING ANDRIOD SMART PHONES VIA BLUETOOH
BY
LEE YEN NEE

A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF INFORMATION SYSTEM ENGINEERING (HONS)
Faculty of Information and Communication Technology
(Perak Campus)

MAY 2012

DECLARATION OF ORIGINALITY

I declare that this report entitled "MULTI OBJECTS PROXIMITY MONITORING USING ANDROID SMART PHONES VIA BT" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : _____

Name : LEE YEN NEE

Date : 13/8/2012

Acknowledgements

I would like to express my sincere appreciation and special thanks to my project supervisor Mr. Ooi Boon Yaik , who gave me this wonderful opportunity to do this wonderful project on the topic Bluetooth Tracker. He also gave me guidance in completing the project and in the process I am also able to gain a lot of new and invaluable knowledge. I am really thankful to him.

 I would also like to thank my parents who give me love, encouragement and support in completing the project within the limited time and throughout the course.

Lastly I would also like to thank my friends who in one way or other helped me in completing the projects

Abstracts

This project is to develop an android application to help user keep track of one or multiple connected devices which can help user to reduce lost of belonging. Although there are a lot of similar applications found in the market, all those applications are not suitable for our scenario. In this project, we will develop the application by adding some new features to it and make it a bit different from the other existing applications. The application will include some features such as allowing multiple tacking and providing functions that will notify connected devices when user devices battery power is low. Since most of the devices nowadays provide Bluetooth technology, the application will use Bluetooth technology to communication between two or more devices. By using Bluetooth technology, it allow user Bluetooth device to connect to more than one device at a time. We will adapt and follow spiral model to develop the application. This application is suitable for those users who are concern about the loss of their belongings or kids. The application will help the user to keep track of their belongings and be aware that their belonging is near or far from them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Introduction and Motivation

This project is to develop an android application that will keep track of one or multiple connected Bluetooth devices such as mobile phones. This application help user to keep track of their children, friends and family members at anytime and anywhere. The user is allowed to keep track of one or multiple Bluetooth devices at the same time. When the connected device is out of range or disconnected, the device will alert the user by triggering the device alarm and vibrate. This will alert the user that the device is disconnected or is out of user range. By triggering the alarm, it can help user to avoid lost of belonging or warn user when one of the user friend is out of his/her range. Besides that, this application will also notify other connected devices when the device battery power is low. This will enable other user to be aware that the connected device battery power is low and may automatically shut down. In addition, when user faces any emergency situation/problem and requires help, they can just shake the mobile phone and the application will send out emergency message to all other connected devices.

Bluetooth technology is intended to replace cable with wireless connection. Bluetooth is a technology that allows devices to communicate with each other over a short distance wirelessly. Bluetooth technology transmits data via low-power radio waves (Franklin, C. and Layton, J. 2000). The communication between two or more devices can be successful only if the devices have Bluetooth technology. Bluetooth only covers very short distances, typically up to 30 feet or 10 meters (Compnetworking.about.com, n.d.). Bluetooth devices normally communicate about less than 1 Mbps (Compnetworking.about.com, n.d.). Besides that, Bluetooth technology allows those devices to communicate with each other even if both devices are in different building, the wall does not stop the Bluetooth signal (Franklin, C. and Layton, J. 2000).

Bluetooth pairing is needed when two Bluetooth device are connected to each other in order to communicate. To ensure that the pair device is accepted, the device user is normally required to key in a passkey. Once the Bluetooth device is pair for the first time,

it will automatically save the device in a list (Munson, M. 2002). Bluetooth is commonly used for transfer of sound or byte data such as picture, song and file.

There are several benefits and advantages of using Bluetooth. One of the benefits is that Bluetooth technology is available in most of the hand phones and other devices such as computers, printers and Bluetooth headsets. Bluetooth is very easy to use especially for those who are not so familiar with technology, It does not require any technical skills. The user is require to click on the "ON' button and Bluetooth will configure itself. Since Bluetooth device uses low power signal, it does not consume a lot of energy. This will help mobile devices to save more battery power when using it.

1.2 Problem Statement

Nowadays, there are a lot of Bluetooth finder applications that is available in the android market. Each application has its own functions and advantages. These existing applications also have disadvantages which can be further improve.

- ♣ Provide only one way notification.

    One of the disadvantages of the existing application is that they do not provide two way notifications between both connected devices. Most of the existing applications that are available in the market only provide one way notification. Using one way notifications, the application can either send notification or receive notification from connected device when pair device is out of range or disconnected. It does not allow the application to receive and send notification to connected device at the same time. Two way notifications enable the user application to notify other connected device and also get notification from other devices at the same time. For example, in two way notifications device A is connected to device B, when device A is out of range or disconnected, both device A and B get the notification about this situation.

- ♣ Do not allow multiple object tracking.

    Besides that, most of the applications only allow one-to-one tracking. They do not allow one Bluetooth device to keep track of multiple devices. This mean that the user is only allow to control and keep track of one device at a time. Userill be at a disadvantage when they need to keep track of more than one device at the same time.

- ♣ Do not notify other connected devices when user devices battery power is low.

    The disadvantage of most of the existing applications is that they do not provide notification to notify other user devices when one of the connected device battery power is low. Without this function, the user sometimes may not be aware that the battery power of the other connected device is getting low or have already auto shut down due to low battery power. This will cause some problem because in the

BIS (Hons) Information Systems Engineering

event the user device auto shut down, other connected device will assume that the user device is out of range or disconnected and will send out notification even if both devices is actually beside each other. By sending SMS to notify user that pair device battery power is low, user will know and be aware of it even through the device shut down and send out of range or disconnected notification.

♣ Do not provide function which send emergency message through SMS.

Another disadvantage of the existing application is that they do not provide function which send emergency message through SMS. Without this function, the user is unable to send emergency message to other connected device when the user face an emergency situation.

1.3 Objective

Since the existing application that is available in the android market still have some disadvantages/problems, we are going to make some improvements by building several features that will improves/solves these disadvantages/problems.

♣ Provide two way notifications.

The feature that we are going to add in will be two way notifications. By adding two way notifications to the application, user will be able to send notification to other pair device and also get notify from other pair devices at the same time when one of them is out of range or disconnected.

♣ Allow multiple tacking.

Besides that, we also will add another feature that allow user to keep track of multiple Bluetooth devices at the same time. The user can either keep track of one or multiple devices base on the user choices. This will help user to solve the problem when the user wants to monitor or keep track of multiple Bluetooth devices at the same time.

♣ Provide function that notifies connected devices when user devices battery power is low.

We also will add on another feature that will notify other connected device when the user device battery power is getting low. By adding this feature, it will allow other user to be aware that one of the connected device battery power is low and may auto shut down soon.

♣ Provide function which send emergency message through SMS.

In addition, we will also include another feature which will enable user to send emergency messages through SMS. When user faces any emergency situation, they just need to shake the mobile phone and the application will automatically send out help messages through SMS to all other connected devices.

1.4 Contribution

By building this application, it will help user to reduce lost of belonging. With the function of notifying user when pair device is out of range or disconnected, it can help user to reduce lost of property or lost sight of their kids. For example, when one of the user property is taken by other people, the device will notify the user that the connected device is out of range or disconnected and user can be aware of the situation.

Besides that, with two way notification function, it allows both connected device to be aware that pair device is out of range or disconnected. With this, not only one device will be aware that connected device is out of range or disconnected but both devices will be aware. When both side of user get out of range or disconnected notification, the user can immediately try to trace/contact each other.

Other than that, the user is allowed to keep track of more than one Bluetooth device. This allow user not only to keep track of one device per time but able to keep track of multiple devices simultaneously at a time. For example, user is not only able to monitor his/her kids, at the same time he/she is also able to keep track of his/her belonging and keep track of his/her friends. The application can be used anywhere and at any time such as while hiking, it can keep track of group mates thus helping to prevent separation of group mates. While in a mall, it can be used to keep track of kids while parents/guardians are shopping.

With the function that allows user to send emergency messages through SMS via shaking the phone, this provides benefit to the user. In the event that the user is alone and in an emergency situation, the user can just shake the phone rather than call for assistances (which in certain situation the user is unable to make a call e.g. while being robbed) and other connected device user will receives the help message. When the other connected user gets the help message, they can estimate whether the user is within their range and immediately provide help.

1.5 Background

Nowadays, we can easily find Bluetooth finder application that is available in the android market. Bluetooth finder application which can be obtain at the android market such as BabyBluetooth Alarm, Bluetooth Tracker, BlueUnity FDL, Stay Close 2 Me, Bluetooth Alarm Free, and STAY HERE! BT. All those applications have their own advantages and disadvantages depend on the user intention to use for what situation.

Most of the applications only provide one-on-one paring, which means that only one device can connect to one device at a time. For example, Baby Bluetooth Alarm, Bluetooth Tracker, Stay Close 2 Me, Bluetooth Alarm Free (FREE), STAY HERE! BT. This allow user to keep track on only one device at a time which is very inconvenient for user who want to monitor multiple devices at a time.

Most of the applications only provide one way notification function when connected device is out of range or disconnected. When connected device is out of range or disconnected, only one user will get the notification but the other connected user will not receive the notification such as Baby Bluetooth Alarm, Bluetooth Tracker, Stay Close 2 Me, Bluetooth Alarm Free (FREE), STAY HERE! BT.

Besides that, most of the applications do not provide function which will notify other connected devices when user device battery power is low. For example, Baby Bluetooth Alarm, Bluetooth Tracker, Stay Close 2 Me, Bluetooth Alarm Free (FREE), STAY HERE! BT.

Furthermore, most of the application also do not provide feature which allow user to shake the phone in and emergency situation and the application will automatically send emergency message to other connected device through SMS such as Baby Bluetooth Alarm, Bluetooth Tracker, Stay Close 2 Me, Bluetooth Alarm Free (FREE), STAY HERE! BT.

There are still some improvements which can be make to improve the application in order to be more useful, user friendly and suitable for our scenario. To solve these problems, we will add on some new features into the application which cannot be found on most of the existing application available in the market.

CHAPTER 2 LITERATURE REVIEW

2.1 Literature Review

There are a lot of similar applications that are available in the market which allow user to either pay for it or download it for free. All those application have it own benefits and also weaknesses.

BabyBluetooth Alarm allows the user to keep track or to find a connected Bluetooth headset (Play.google.com 2011). In this application it provide two different modes for the user to choose, which are "Tracker Mode" and "Finder Mode". In the "Tracker Mode" if the connected Bluetooth headset goes out of range an alarm will sound. And in the "Finder Mode" if the connected Bluetooth headset comes within range an alarm will sound. This allows the user to get to know once the connected Bluetooth device is either within range or out of range that enable the user to keep track of their kids, friends, family or children at a shopping mall, picnic, friend's party or any other event. Other than that, it also allow the user to find lost item/s which is attached with any Bluetooth headset, the alarm will ring when the lost item/s is within the user range. Since the alarm will sound once the connect Bluetooth headset is within range or out of range, it will be quite confusing if the alarm is with the same sound. The user needs to check whether the connected Bluetooth headset is within range or out of range each time the alarm sound. Besides that, BabyBluetooth Alarm does not support multiple Bluetooth devices to connect to each other. It only allows two Bluetooth headsets to connect with each other.

Bluetooth Tracker is an application that allows the user to keep track of another Bluetooth device to determine if it is within range or out of range (Bowl, P. 2011). When the connected Bluetooth device is out of range, the device will notify the user by enabling the phone to vibrate. Besides that, if the user set to "vibrate within range", the phone alert the user when the connected Bluetooth device is come within range. This allow the user to keep track of the connected device, it will notify the user when the connected device is taken certain distance away from users. It also will alert the user when the connected device comes within range, so that the user knows that the connected device is near to him/her. The weaknesses of the Bluetooth Tracker are that it only allows one Bluetooth

9

device to connect to another Bluetooth device at the same time. This means that the user only can keep track of only one device at a time. Bluetooth Tracker notify it user when the connected device is within or out of range by vibrating the device. Vibrating the device may not be a good choice to notify the user. This is because in certain condition the user may not notice that the phone is vibrating and thus is not aware that the connected device is already out of range.

BlueUnity FDL is an application that allows user to keep track of Bluetooth enabled headsets and also other mobile phones (Play.google.com 2010). This application also allow user to save a Bluetooth device or phone to a friend list. The user will receive alerts when friends are near or out of range. This application allow the user to keep track of all his/her friends, the device will notify the user once the user friend is near to the user or the user friend is out of the user ranges. Other than that, BlueUnity FDL also allow user to keep track of multiple Bluetooth enabled device at the same time. It also allow user to save the Bluetooth device into a friend list. This is more convenient, because the user does not need to pair with other Bluetooth device each time when the user wants to keep track of that device. The user just need to pair with the device once and save the device into friend list, the user then can keep track of the device anytime the users want. Although the user is able to keep track of the device that is save in the friend list at anytime, but this only can be done when the Bluetooth enabled device is turn is onn. If the device that is save in the friend list is on turn off in Bluetooth, the user will be unabled to keep track of the device.

Stay Close 2 Me is an application that will show list of all BT devices that have paired with user phone before (Benny, G. ,n.d.) . Once the user selects the device it wants to keep track of, the application will start to keep track on that device. When the connected device get out of range, the device will trigger a powerful alarm sound and start vibrating the phone, alerting the user that the device is out of range. This application help user to keep track of the connected device and it will notify the user when the pair device is far away from user device or out of range. By generating a powerful alarm sound and vibrating the phone it is easy to get user attention and notice that the pair device is out of

range, rather than just vibrating the phone which may not get the attention by the user. But this application only allow user device to connect to only one Bluetooth enable device, which means that it can only keep track of only one device each a time.

Bluetooth Alarm Free is an application that enables the user to keep track of the connected Bluetooth devices (SSPARK 2011) .The alarm will trigger to notify the user when the registered connected Bluetooth devices are away from certain distance. It will also notify the user when the devices get close to him/her by changing mode. This enable the user to keep track of the connected device, allow the user to know whether the pair device is near or far from the user which can help to avoid losing properties. By triggering the alarm when connected device is within the range of user device enable the user to know that the connected device is very near to the user. In order to get notification either when the pair device is out of range or within the range, the user need to change the mode within them. This make things very inconvenient, because the user need to change between mode each time the device goes out of range and change back to the mode within the user range. Other than that, Bluetooth Alarm Free only allows one-on-one connection. This means that the user device can only be connect to one of the Bluetooth devices each a time. This is not applicable when user needs to keep track of multiple Bluetooth devices.

STAY HERE! BT is an application that uses Bluetooth headset connection to keep track other Bluetooth device (Hdop 2012). An alarm will be triggered when the connected Bluetooth device goes out of range. This application help user to keep track of the connected device, when the connected device is out of range it will notify the user. This allows the user to know that the connected device is out of ranger or far from the user. Besides that, sending out the notification by triggering the alarm will get user attention and enable the user to notice about what is happening. But, this application only allows the user to connect to only one device at a time. This may not be convenient to some users that wish to keep track or connect to more than one device at a time.

	Notify user when Bluetooth headset goes out of range	Two way notifica-tions	One way notifica-tions	Pair with Multiple Bluetooth Device	Notify other connected device when user device battery is low	Automatically send emergency message through SMS
Baby Bluetooth Alarm	Yes	No	Yes	No	No	No
Bluetooth Tracker	Yes	No	Yes	No	No	No
Blue Unity FDL (Legacy)	Yes	No	Yes	Yes	No	No
Stay Close 2 Me	Yes	No	Yes	No	No	No
Bluetooth Alarm Free (Free)	Yes	No	Yes	No	No	No
STAY HERE! BT	Yes	No	Yes	No	No	No

Table 2.1: Comparison between existing applications

Based on table 2.1, it shows the existing applications with each of their own attributes. Each of the attributes has its own functionality and benefits. For the notify user when Bluetooth headset goes out of range, it provide the function which will alert or send notification to the user by triggering the alarm when the connected Bluetooth devices is out of the user range or disconnected. This is important because without the notification, the user may not notice that the pair device is out of the user range or disconnected. Besides that, the user will need to check the phone frequently to know whether the connected device is out of range or disconnected which will cause a lot of inconvenience.

Two way notifications enable both sides of device to receive notification at the same time when one of them is out of range or disconnected. Both sides of the application can receive notification from the connected device when it is out of the user range. By using two way notifications, the user application can receive notification from the pair device and at the same time send out the notification to the pair device. This is important because all the connected device need to get notification from each other during the time that one of the connected devices is out of range or disconnected. Without two way notifications, only one side of the device can get the notification and other connected device will not get any notification and will not be aware that they have out of connected device range.

One way notifications only allow one side of the device to receive notification. One side of the device will send out notification to alert the user that the connected device is out of user range. While another side of the device will receive the notification. This mean that only the user device will be aware that the pair device is out of range or disconnected but the pair device will not aware about this situation. This is important when the user wants to keep track of other devices and the other devices do not keep track of the user device. So that when the pair device is out of range or disconnected, only the user device will get notification while the pair device will not get the notification.

For pair with multiple Bluetooth devices, it enables the user Bluetooth device to pair with one or more Bluetooth devices at the same time. This is important when the user want to keep track of multiple devices at the same time. Without this, the user is unable to keep track of multiple devices at the same time. They can only keep track and monitor one device at a times.

Notifying other connected device when user device battery power is low is to provide notification or to warn other connected user that the user device battery power is getting low. This is important because other user may not be aware that the user device battery power is low while using the application and the user device may auto shut down without any other user knowing about it.

Lastly, for the automatically send emergency message through SMS, this is to allow user to get help urgently when the user face any dangerous situation. With this function, user can just shake the phone and the application will automatically send out help message to all other connected devices. Once the other connected devices receives the SMS, they will be aware that the user is in danger and can give him/her assistance.

There are a lot of similar applications that are available in the Android market. But all these existing applications do not meet our requirement. We would like to have an application that have all these functions such as notify user when Bluetooth headset goes out of range, two way notifications, pair with multiple Bluetooth device, connected device when user device battery power is low and automatically send emergency message through SMS. Based on the table above, this type of application is still not available in the market yet.

2.2 Application Features

This application is mainly use to keep track of one or multiple Bluetooth enable headset and devices. This application enables the user to keeping track of their children, friends, family or kids at anywhere and anytime.

Features:

1) One Bluetooth devices is connected to one or more pair devices. If one of the pair devices goes out of range or disconnected from the connected devices, both the user and the pair device alarm will ring notifying them that they are far away from each other.

   Example: DEVICE 1 is connected with DEVICE 2 and DEVICE 3, if DEVICE 1 is disconnected with DEVICE 2, DEVICE 1 and DEVICE 2 phone alarm will trigger. If DEVICE 1 is disconnected with DEVICE 3, DEVICE 1 and DEVICE 3 phone alarm will trigger.

2) While using this application, if the user device battery power is found to be low, a notification will send out by SMS to warm the other connected user about this issue.

3) While using this application, user can just shake the phone and the application will automatically send out emergency message through SMS to other connected device.

With this application it is possible, for instance:

♣ To prevent your phone been lost/stolen

♣ To prevent lost of properties by putting Bluetooth device in the bags.

♣ To prevent losing sight of your kids

♣ To prevent getting lost in a group.

♣ Allow you to determine whether your friend is near or far from you.

CHAPTER 3 METHODOLOGY AND TECHNOLOGY INVOLVED

3.1 Methodology

It involves a few different stages in order to complete the whole project. We follow spiral model to develop and carry out our project.

Stage 1

♣ Involves collecting information about existing products which is similar to the application that we are going to build. After getting all the information we need, we will understand how those products work and find out their advantages and disadvantages. Then we will decide which features to add on to our project.

Stage 2

♣ To show a large picture of how our application really works after adding in the new features. We will come out with flow charts to show the flow of the application.

♣ Design the interface for the application.

Stage 3

♣ Start the programming and coding part. We will use eclipse software to develop our android application.

Stage 4

♣ Once the coding for the application has been completed, we will then proceed to testing part. Testing stage is to check the application whether it got any problem or error.
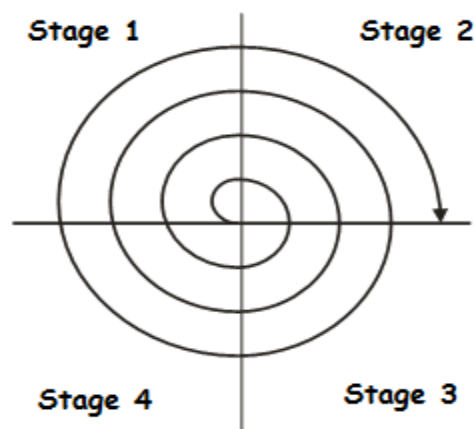
Figure 3.1: Spiral model methodology

3.2 Timeline

	Task Mode ▼	Task Name ▼	Duration ▼	Start ▼	Finish ▼	Predecessors ▼
1	✈	Meet supervisor and identify title	3 days	Mon 30/1/12	Wed 1/2/12	
2	✈	Start and submit preliminary proposal reports	28 days	Thu 2/2/12	Mon 12/3/12	1
3	✈	Pass up report for Project 1	20 days	Tue 13/3/12	Mon 9/4/12	2
4	✈	Oral presentation and submit poster	10 days	Tue 10/4/12	Mon 23/4/12	3
5	✈	Meet supervisor and discuss about Project 2	2 days	Tue 24/4/12	Wed 25/4/12	4
6	✈	Design and write coding	34 days	Thu 26/4/12	Tue 12/6/12	5
7	✈	Testing and Documentation	21 days	Wed 13/6/12	Wed 11/7/12	6
8	✈	Submit final report 2	1 day	Thu 12/7/12	Thu 12/7/12	7
9	✈	Oral presentation and Product demonstrration	12 days	Fri 13/7/12	Mon 30/7/12	8
10	✈	Poster submission	1 day	Tue 31/7/12	Tue 31/7/12	9

Table 3.1: Gantt Chart

Figure 3.2: Gantt Chart

BIS (Hons) Information Systems Engineering
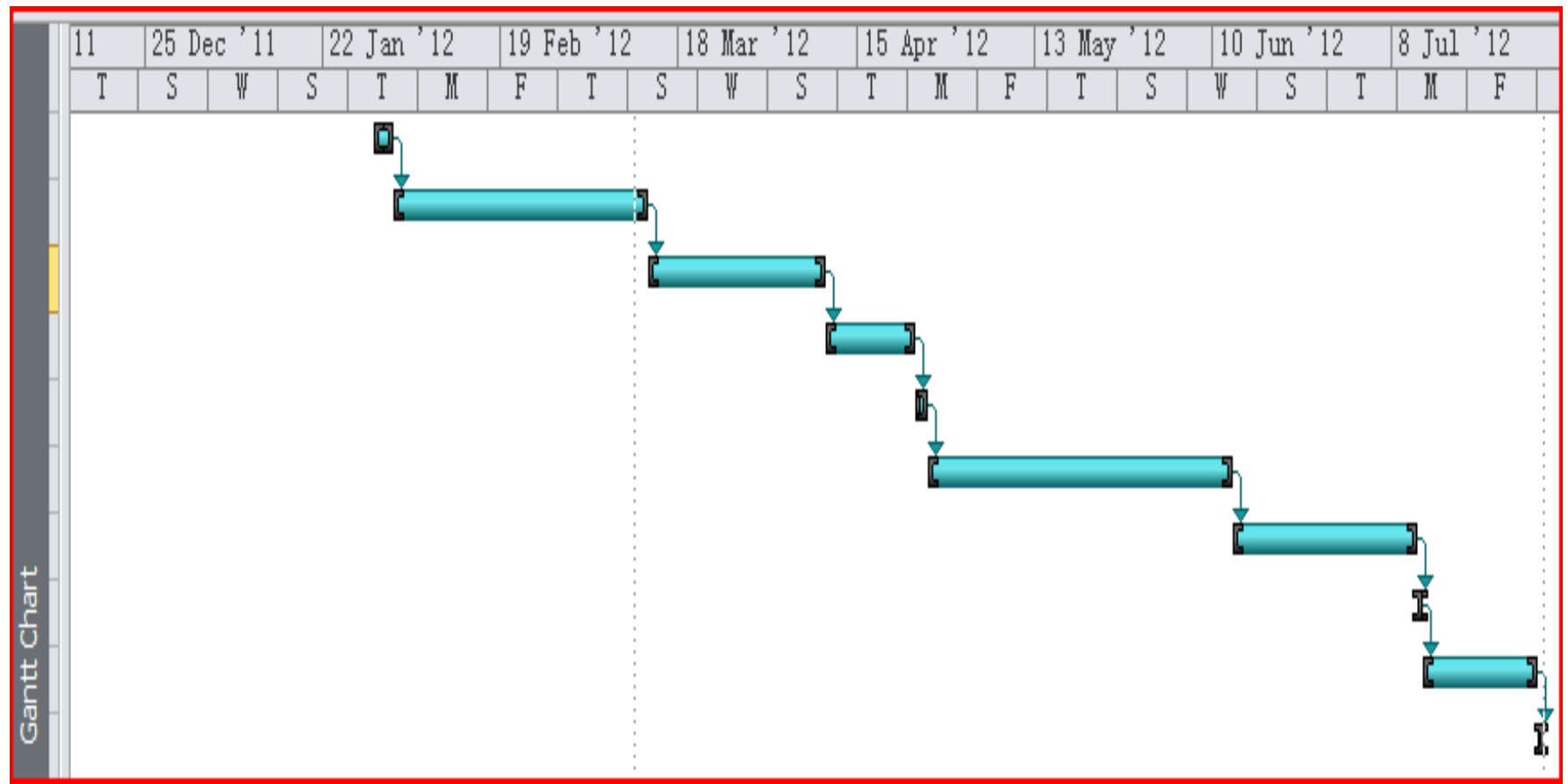
3.3 Technology Involved

Bluetooth will be the technology that will be use in building this project. Bluetooth is a technology that allows devices to communicate with each other over a short distance wirelessly (Franklin, C. and Layton, J. 2000). Bluetooth technology transmits data via low-power radio waves (Compnetworking.about.com, n.d.).

Other than that, the application will run on mobile phone which will use android operating system. Android is an operating system for mobile phones and tablets, which is maintained by Google (Bolton, M. 2012). The application can run on a few devices such as smart phones or tablets which run on android operating system.

3.4 Requirement Specifications

**Entity Relationship Diagram**
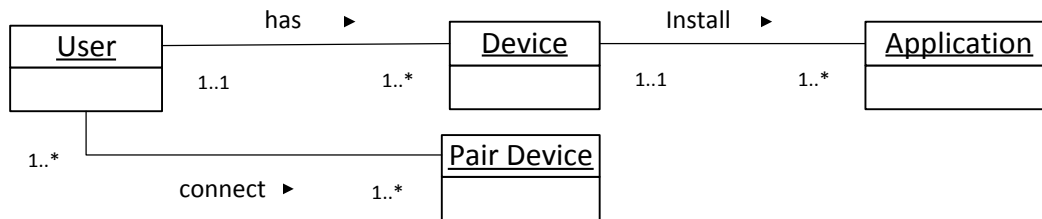
Figure 3.3: Entity Relationship Diagram of Bluetooth Pairing

The Figure 3.3 shows the entity relationship diagram. It consists of entities as well as relationships between entities which are use in our project. There are total of 4 entities c in the diagram and each is related to one another. The 4 entities are User, Device, Application, and Pair Device. For the User entity, user may own one or more device at a time, and each device is only own by one user. While for the Device entity, each device can only install one of our project applications, and one application can be install to one or more device. For the Application entity, one application can connect to one or more pair device and one pair device can work with one or more applications. Besides that, an

application can notify one or more user, and a user can get more than one notification at a time.

Class Diagram
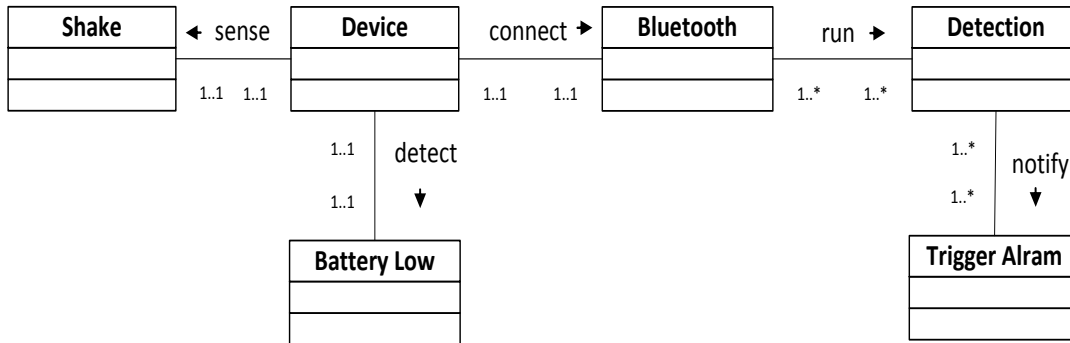
Figure 3.4: Class Diagram of the Application

The Figure 3.4 shows the class diagram and their relationship which are use in our project. The class diagram contains total of 6 classes which are Device, Bluetooth, Detection, Trigger Alarm, Battery Low and Shake. Each class has its own function and attributes. For the Devices class, it is responsible to run and startup the application and also responsible in closing or terminating the application. While for the Bluetooth class, it deals with pairing between devices and also detecting the range between the connected devices. Other than that, for the Detection class it is used to detect whether the connected device is out of range or disconnected. Besides that, for the Trigger Alarm class, it used to handle the notification when a connected device is out of range or disconnected. It will send out a notification to trigger other connected device alarm when the user device is out of range or disconnected. Furthermore, for the Battery Low class, it is responsible to send SMS to notify other connected user when user device battery is low. Lastly, for the shake class, it is use to send out emergency message through SMS when the device detect shacking.

Use Case Diagram
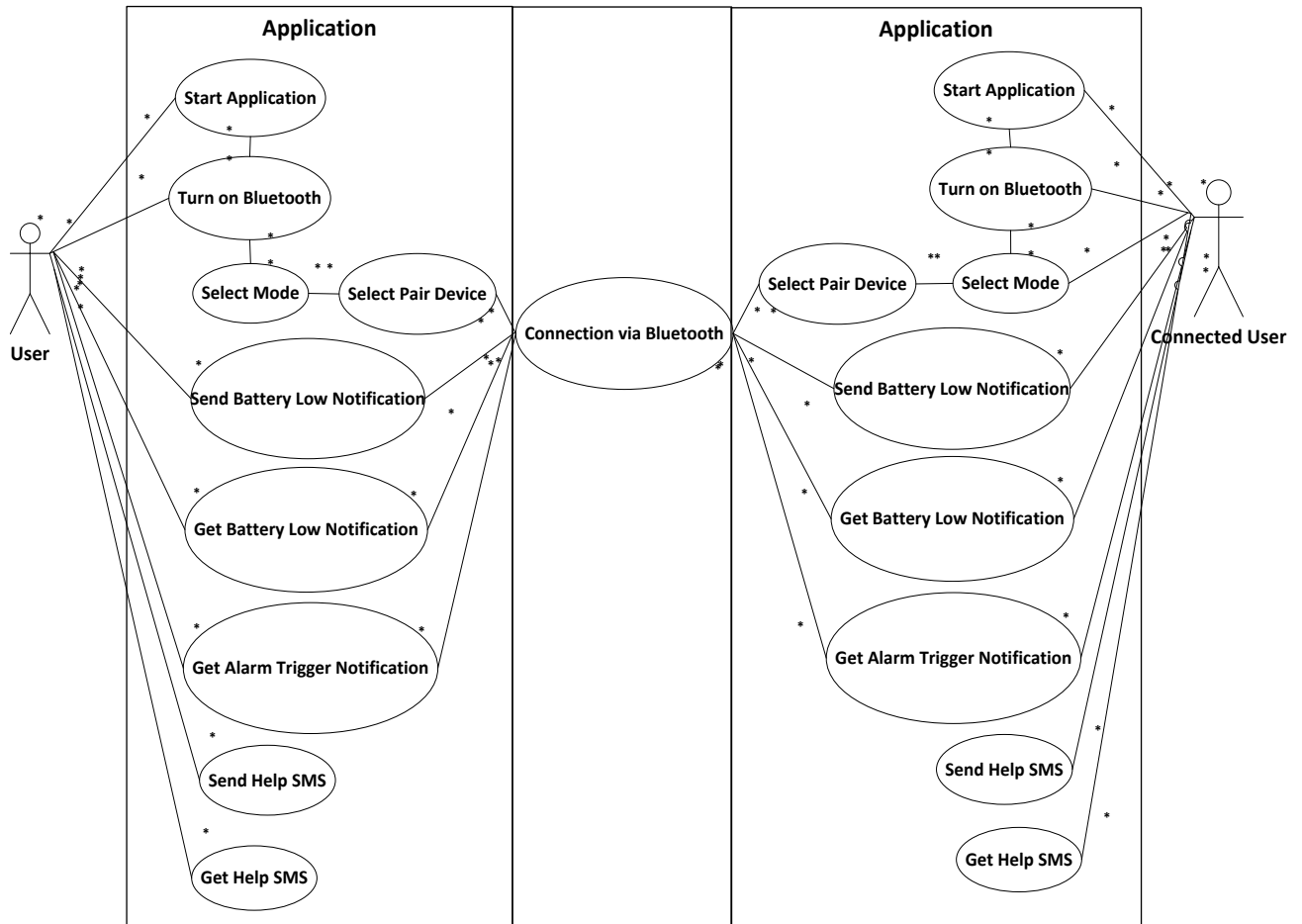
Figure 3.5: Use Case Diagram of the Interaction between User and the Application

The Figure 3.5 shows use case diagram of our project. It shows how user interacts with the application. User can first start the application and turn on the Bluetooth. After turning on the Bluetooth, user can select the mode they want and then select the pair device they wish to pair with. Besides that, user can also get alarm trigger notification when connected device is out of user range. Other than that, user is able to send out battery low notification to connected devices when user device battery power is low or user can get battery low notification from other connected devices when the connected device battery power is low. Lastly, user can also send out help SMS to connect devices by shaking the mobile phone. Besides sending, user can also receive help SMS from other connected devices.

Sequence Diagram
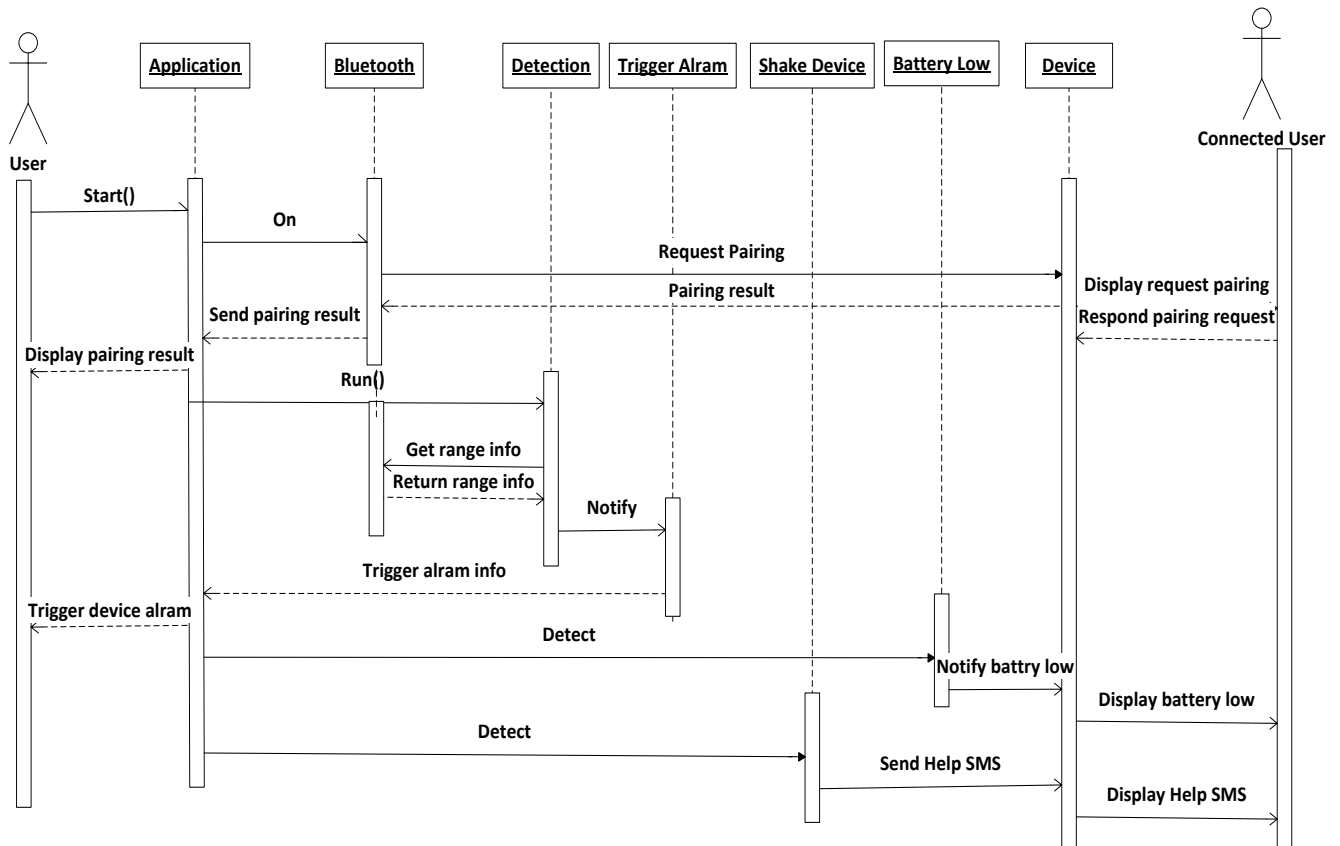
Figure 3.6: Sequence Diagram of sequence of the interaction between user and the application

The Figure 3.6 shows the sequence diagram of our project. It provides illustration of the order in which messages are sent between objects. The sequence diagram contains of 6 objects which are Application, Bluetooth, Detection, Trigger Alarm, Battery Low and Device. It also contains 2 actors in this diagram which are User and Connected User. First the User start the Application and then the Application allow user to turn on the Bluetooth. The Bluetooth will send pairing request to the Connected User. Once the Connected User accepts the pairing request, the result will send back and display to the User. After the devices have successful connected to each other, the Application object will send a message to trigger the Detection object. The Detection object will then get range information between the devices for the Bluetooth object to determine the range between the connected devices. When the connected devices are either out of range or

22

disconnected, the Detection object will send a message to notify the Trigger Alarm object. The Trigger Alarm object will then notify the Application and the Application will warn the user about the issue by triggering the alarm. At the same time, the Application object will also detect if the user device battery power is low. If the device is low in battery, the Battery Low object will notify the device and it will notify and display the message to the user. Other than that, the application also can detect shaking motion. If the device detects any shaking motion, the Shake Device object will send out Help SMS and the device will display the SMS to the user.
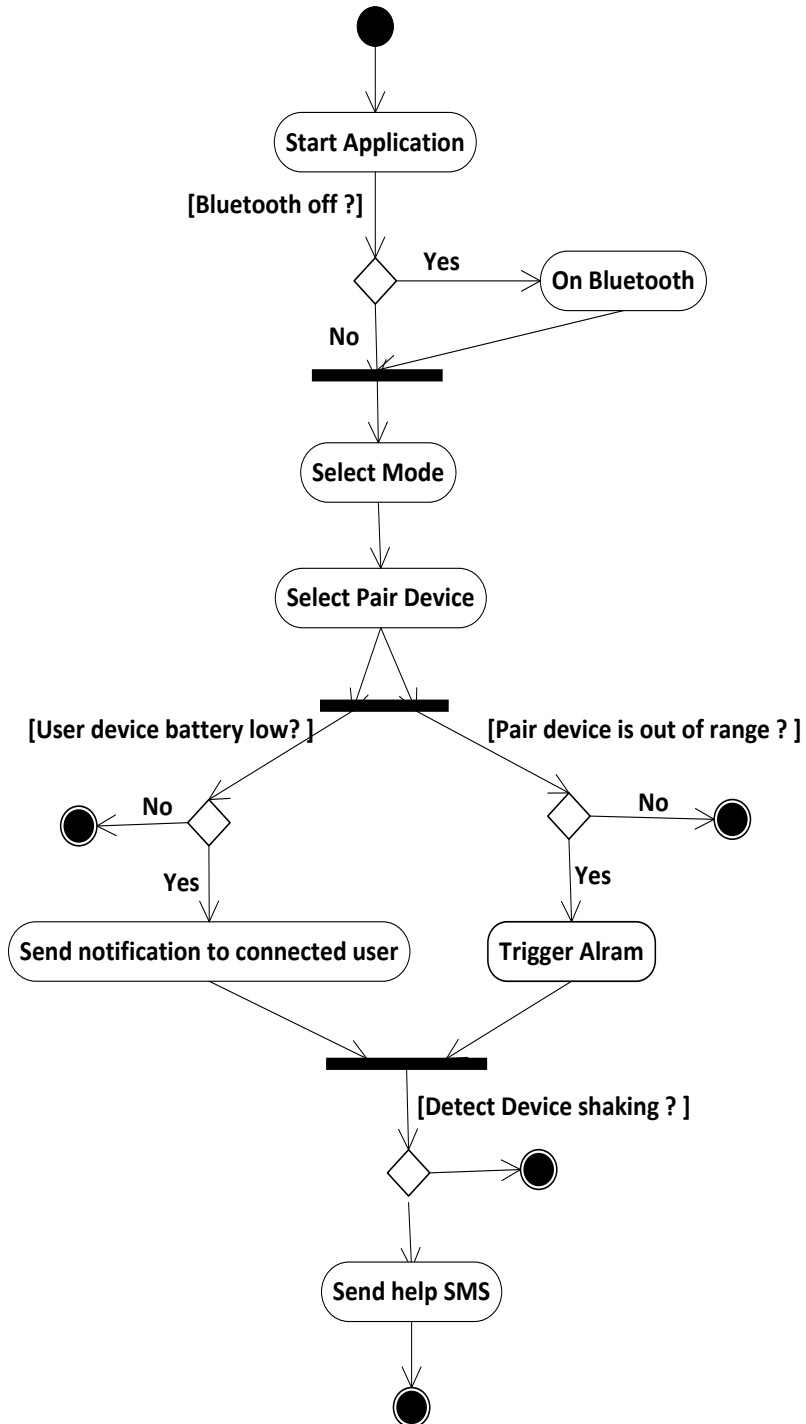
Activity Diagram

Figure 3.7: Activity Diagram of the Application

The Figure 3.7 shows the activity diagram of our project.It show how the system's execute and how it changes directions based on different conditions. The diagram contains 6 activities which are Start Application, On Bluetooth, Select Mode, Select Pair Device, Alarm Trigger, SMS to notify other connected user and Send help SMS. The activity is started at the start state and the application is start, it then makes decision on whether the Bluetooth is on or off. If the Bluetooth is off, it will allow user to turn on the Bluetooth. The application then allows user to select pair device and then the process is then divided into two decision point which run at the same time. At one of the decision point, it checks whether the pair device is out of range or disconnected. If the answer is yes, it will trigger the alarm. While for another decision point, it checks whether the user battery is low. If the user battery is low, it will send SMS to notify the connected pair device about the issue. It then come to another decision point to check whether the device detect any shaking motion. If the answer is yes, it will send out SMS to notify other connected device. After all the activity has done, it will end the state.

3.5 Test Plan

Black box testing is also known as functional testing (Webopedia.com 2012). Black box testing only test the functionality of an application or a system but will not test its internal structures. For black box testing, the tester will not need to examine the programming code for the system and does not need any knowledge about the program other than its specifications. For Example, the tester only knows the inputs and what the expected output should be and not how the programs generate those outputs. We will test the application to check if it perform or behave as expected using black box testing. We will test the application by performing a simple action such as clicking a button or a drop down menu and observe is the application performing as what we expected.

Next we will do testing on the detection part using black box testing. We will test on when the connected device is out of the user device range, can user device receive any notification about the situation. Besides that, we also will do testing on when the user device battery is low, does other connected devices receive SMS that generated by user device to inform them about this situation. In addition, we also will test that when user shake the phone does other connect devices receives SMS.

CHAPTER 4 IMPLEMENTATION

4.1 Overview of the Implementation

We used Eclipse program to develop our android application. This is because most of the android application is develop using Eclipse software. Besides that, we also used Android 2.3 platform to develop our application. This is because most of the android device which is available will be using platform 2.3 or above. This means that most of the available device can support or use our application. Other than that, the Bluetooth version that we use will be Bluetooth 2.0. This is because most of the android device is using Bluetooth 2.0.

4.2 Implementation of the Bluetooth Discovery Process
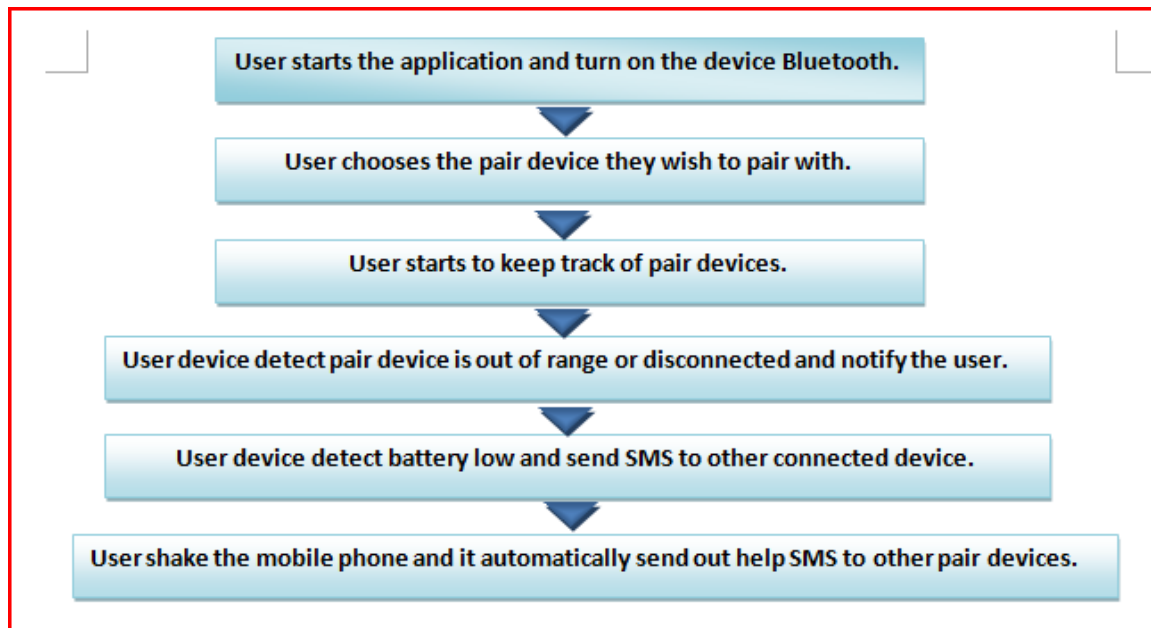
Figure 4.1: Sequence of interaction between user and the device for the application.

The Figure 4.1 above shows the sequence of interaction between user and the device while using the application. To build this application, we have to combine few different classes and part of coding into one complete application. The main function of this application is to establish Bluetooth connection between two or more devices. Without this function we will fail and unable to meet our goal and objective. The Figure 4.2 below

show the code which allow the application to establish Bluetooth connection between devices.

```java
private void connectDevice(Intent data, boolean secure) {
    // Get the device MAC address
    String address = data.getExtras()
        .getString(DeviceListActivity.EXTRA_DEVICE_ADDRESS);
    // Get the BLuetoothDevice object
    BluetoothDevice device = mBluetoothAdapter.getRemoteDevice(address);
    // Attempt to connect to the device
    MAC[usernumber] = address;
    mChatService.connect(device, secure);

}
```

Figure 4.2: Source code of establish Bluetooth connection

Before establishing Bluetooth connection between devices, user needs to choose which device they wish to pair with. The Figure 4.3 below shows the code which allow user to choose which device to pair with.

```java
// The on-click listener for all devices in the ListViews
private OnItemClickListener mDeviceClickListener = new OnItemClickListener() {
    public void onItemClick(AdapterView<?> av, View v, int arg2, long arg3) {
        // Cancel discovery because it's costly and we're about to connect
        mBtAdapter.cancelDiscovery();

        // Get the device MAC address, which is the last 17 chars in the View
        String info = ((TextView) v).getText().toString();
        String address = info.substring(info.length() - 17);

        // Create the result Intent and include the MAC address
        Intent intent = new Intent();
        intent.putExtra(EXTRA_DEVICE_ADDRESS, address);

        // Set result and finish this Activity
        setResult(Activity.RESULT_OK, intent);
        finish();
    }
};
```

Figure 4.3: Source code of displaying available device

Once user chooses the device they wishes to pair with, the user need to press the start tracking button and the application will keep track of the connected device. The class which keeps track of the connected device is named as tracking (). The figure 4.4 shows the coding which keep track of pair device.

```
private void tracking(){
    registerReceiver( batteryReceiver, new IntentFilter(Intent.ACTION_BATTERY_CHANGED) );
    // Get the BLuetoothDevice object
    BluetoothDevice device = mBluetoothAdapter.getRemoteDevice(MAC[numberofdevice]);
    // Attempt to connect to the device
    mChatService.connect(device, true);
    }
```

Figure 4.4: Source code for tracking device

When the application detected that one of the connected device is out of range or disconnected, the device will vibrate and the alarm will ring to notify the user that one of the connected device is out of range or disconnected. The code of this part will be show at figure 4.5.

```
if (msg.getData().getString(TOAST).equals("Unable to connect device")
        && callconnectionbutton == true && destry == true&& disconnect==true){
    vibrator.vibrate(7000);
    MediaPlayer mediaPlayer = MediaPlayer.create(getBaseContext() , R.raw.baby);
    mediaPlayer.start();
    senddisconnectedSMS(numberofdevice);
    reconnect();
    disconnect=false;
```

Figure 4.5: Source code for detecting pair device out of range or disconnected

Besides that, the application also will detect the device battery level. Once the battery reach certain level, the device will send out SMS to notify other connected device that it device battery is getting low. The figure 4.6 below shows the code which detects the device battery level.

```
//------------battery---------------------
public void updateBatteryCondition(int level)
{
    if(level==20){
        if(batterystate==0){
        sendSMS();
        batterystate=1;
        }else{
            Log.i(TAG, "THE STATE IS FALSE>>>>>>>>>>>>>>>>>>>>");
        }
    }
}

private BroadcastReceiver batteryReceiver = new BroadcastReceiver() {
@Override
public void onReceive( Context context, Intent intent )
{
int level = intent.getIntExtra( "level", 0 );
updateBatteryCondition(level);
}
};
```

Figure 4.6: Source code for detecting battery level

In addition, we also add in shake detection feature into the application. When the user shake the phone and it meet certain shake level, the application will automatically send help message to other connected device. The figure 4.7 below shows the coding of the shake detection feature.

```java
public void onSensorChanged(SensorEvent event) {

    float x = event.values[0];
    float y = event.values[1];
    float z = event.values[2];
    if (!mInitialized) {
    mInitialized = true;
    } else {
    float deltaX = Math.abs(mLastX - x);
    float deltaY = Math.abs(mLastY - y);
    float deltaZ = Math.abs(mLastZ - z);
    if (deltaX < NOISE) deltaX = (float)0.0;
    if (deltaY < NOISE) deltaY = (float)0.0;
    if (deltaZ < NOISE) deltaZ = (float)0.0;
    mLastX = x;
    mLastY = y;
    mLastZ = z;

    float more = (float) 20.0;
    if(deltaX >more && shakestate==true){
        sendshakeSMS();
        MediaPlayer mediaPlayer = MediaPlayer.create(getBaseContext() , R.raw.coin);
        mediaPlayer.start();
        vibrator.vibrate(2000);
        shakestate=false;
    }else if (deltaY >more && shakestate==true){
        sendshakeSMS();
        MediaPlayer mediaPlayer = MediaPlayer.create(getBaseContext() , R.raw.coin);
        mediaPlayer.start();
        vibrator.vibrate(2000);
        shakestate=false;
    }
  }
}
```

Figure 4.7: Source code of shake detection

Furthermore, we also include SMS messaging function into our application. When the connected device is out of range or disconnected, it will send SMS message to other connected device. Besides that, when user shake the phone, it will automatically send out SMS message to other connected device. The figure 4.8 below shows the coding of one of the SMS classes which include in our application.

```java
public void sendshakeSMS()
{
    for(int i=0;i<phonenum;i++){
    SmsManager sm = SmsManager.getDefault();
    try {
    sm.sendTextMessage(phone[i], null, "User "
    + Devicename + " is asking for help!", null, null);
        Thread.sleep(3000);
    } catch (InterruptedException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
   }
}
```

Figure 4.8: Source code for sending SMS

BIS (Hons) Information Systems Engineering
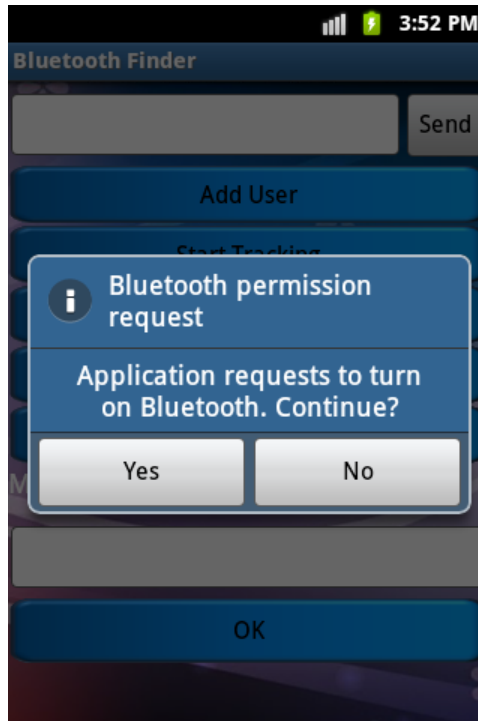
4.3 User Interface

Figure 4.9: User Interface for Bluetooth permission request

Figure 4.9 show the user interface when the application start up, it require user to turn on device Bluetooth before the application can proceed. If the user chooses to turn on the device Bluetooth, the application will proceed to the main menu. If the user chooses not to turn on the Bluetooth, the application will turn off.

Figure 4.1-: Main Menu of the application

After the user chooses to turn on the device Bluetooth, the application will display the main menu as show in Figure 4.10. Add User button is to allow user to choose the device they wish to connect with. Start Tracking button is to allow the user to start keep track of connected device. If user wishes to stop keeping track of connected device, they can press the stop button. Once the Alarm button is press by user, other connected device alarm will be trigger. Exit button allow user to exit the application.
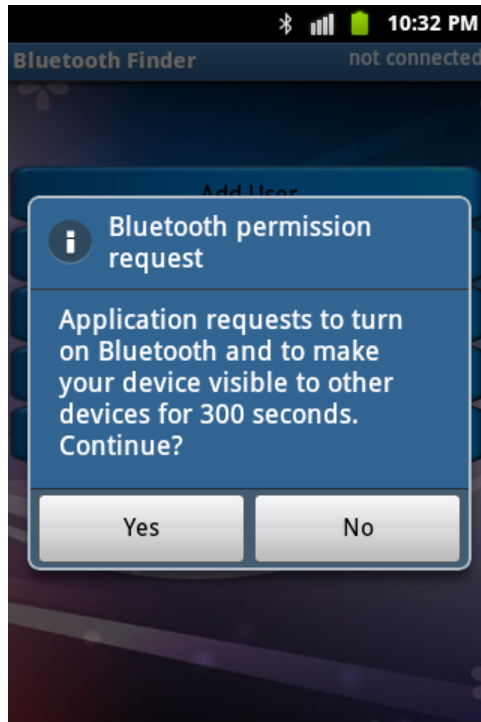
Figure 4.11: User interface request device to be visible.

User can also allow it device to be visible to other Bluetooth devices for 300 second as show in Figure 4.11.
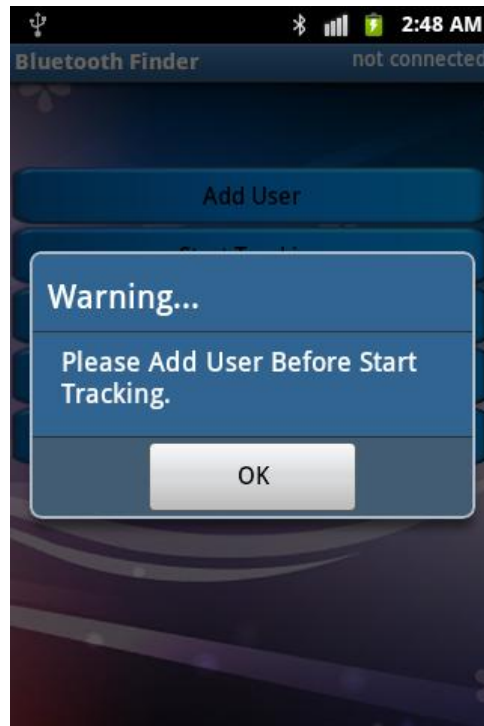
Figure 4.12: Warning alert require user to add in user before tracking.

User need to add in user before they can start to keep track or monitor other user device. A warning alert will pop up if the user starts to keep track other device without adding in any user as show in Figure 4.12.

Figure 4.13: User Interface of list of paired devices

Once the user press the Add User button, an interface will display as show as figure 4.13. This user interface show a list of previously paired device and it allow user to choose the connected device they wish to pair with.

Figure 4.14: User Interface of list of other available device

If the paired device list does not display the devices that the user wish to pair with, the user can press the scan for device button and a list of other available device will be display as show in Figure 4.14.
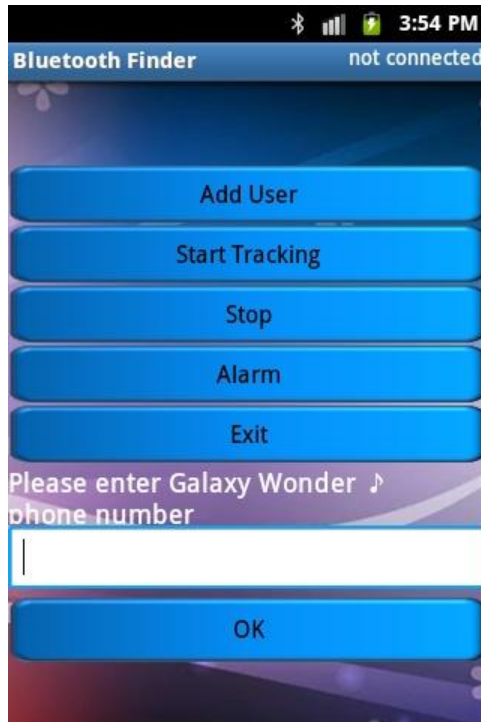
Figure 4.15: User Interface that require user to enter connected device phone number

Once the user choose the pair device they wish to connect with, the application will require the user to enter the phone number of the device which the user choose as show in Figure 4.15.
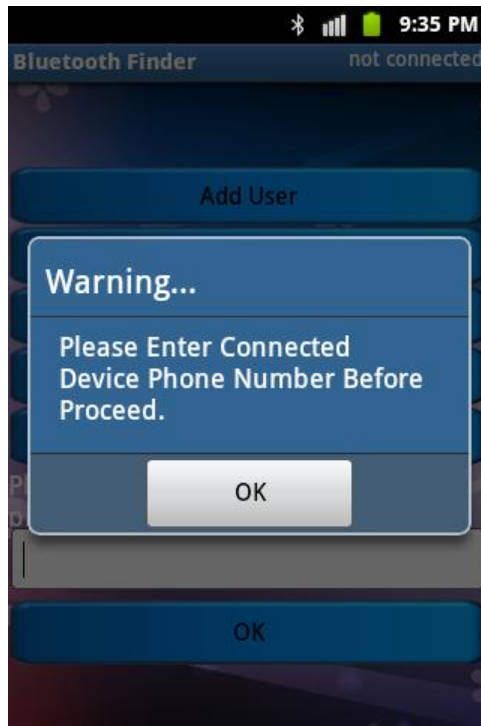
Figure 4.16: Warning alert require user to enter user phone number before proceed

If the user wish to add in another user device or start to keep track connected device without entering the connected device phone number, a warning alert will pop up as show in Figure 4.16.

Figure 4.17: Warning alert which notify user that connected device is out of range or disconnected.

Once the user started to keep track of connected device and connected device is out of range or disconnected, a warning alert will pop up as show in Figure 4.17.

Figure 4.18: User Interface for exit confirmation

If the user wishes to exit the application, a confirmation alert will pop up as show in Figure 4.18.

CHAPTER 5 TESTING

5.1 Testing

Once the application had been develop, we will proceed and continue with the testing stage. Testing is an important stage that we need to go through for every software development. This is because testing allows us to find out or detect previously any undetected bug or error. The reason that we do testing is to check and determine whether the application perform according to the specification. Besides that, testing also allow us to check whether the application contain any error or bug. The Table 5.1 below show the result of testing conducted on the application.

Application Features	Results	Remarks(Done By)
Able to establish connection between devices.	✅	Lee Yen Nee
Able to connect more than 1 device.	✅	Lee Yen Nee
Master Side		
Slave device able to keep track of Master device which is disconnected.	❌	Lee Yen Nee
Slave device able to keep track of Master device which is out of range.	❌	Lee Yen Nee
Master and Slave device alarm ring when Master device is out of range.	❌	Lee Yen Nee
Master and Slave device alarm ring when Master device is disconnected.	❌	Lee Yen Nee
Master and Slave device vibrate when Master device is out of range.	❌	Lee Yen Nee
Master and Slave device vibrate when Master device is disconnected.	❌	Lee Yen Nee
Master and Slave device receive SMS	❌	

when Master device is out of range.		Lee Yen Nee
Master and Slave device receive SMS when Master device is disconnected.	❌	Lee Yen Nee
Master device reconnect back to Slave device when Master device is nearby.	✅	Lim Yuen Nee
Master device send SMS to get back location of Slave device.	✅	Lim Yuen Nee
Master device receive location SMS from Slave device.	✅	Lim Yuen Nee
Master device sound the alarm of Slave device when Slave device is nearby.	✅	Lim Yuen Nee
Slave Side		
Master device able to keep track of Slave device which is disconnected.	✅	Lee Yen Nee
Master device able to keep track of Slave device which is out of range.	✅	Lee Yen Nee
Master and Slave device alarm ring when Slave device is out of range.	✅	Lee Yen Nee
Master and Slave device alarm ring when Slave device is disconnected.	✅	Lee Yen Nee
Master and Slave device vibrate when Slave device is out of range.	✅	Lee Yen Nee
Master and Slave device vibrate when Slave device is disconnected.	✅	Lee Yen Nee
Master and Slave device receive SMS when Slave device is out of range.	✅	Lee Yen Nee
Master and Slave device receive SMS when Slave device is disconnected.	✅	Lee Yen Nee

BIS (Hons) Information Systems Engineering

Master device receive SMS when Slave device shake the phone.	✅	Lee Yen Nee
Slave device receive SMS when Master device shake the phone.	✅	Lee Yen Nee
Master device receive SMS when Slave device battery is low.	✅	Lee Yen Nee
Slave device receive SMS when Master device battery is low.	✅	Lee Yen Nee
Slave device reconnect back to Master device when Slave device is nearby.	✅	Lim Yuen Nee
Slave device send SMS to get back location of Master device.	✅	Lim Yuen Nee
Slave device receive location SMS from Master device.	✅	Lim Yuen Nee
Slave device sound the alarm of Master device when Master device is nearby.	✅	Lim Yuen Nee

**Table 5.1: Test result of the application**

Besides the testing that we conduct as show as Table 5.1 above, we also do extra testing as show in Figure 5.1 and Figure 5.2.
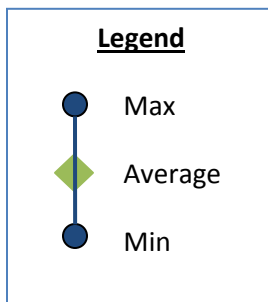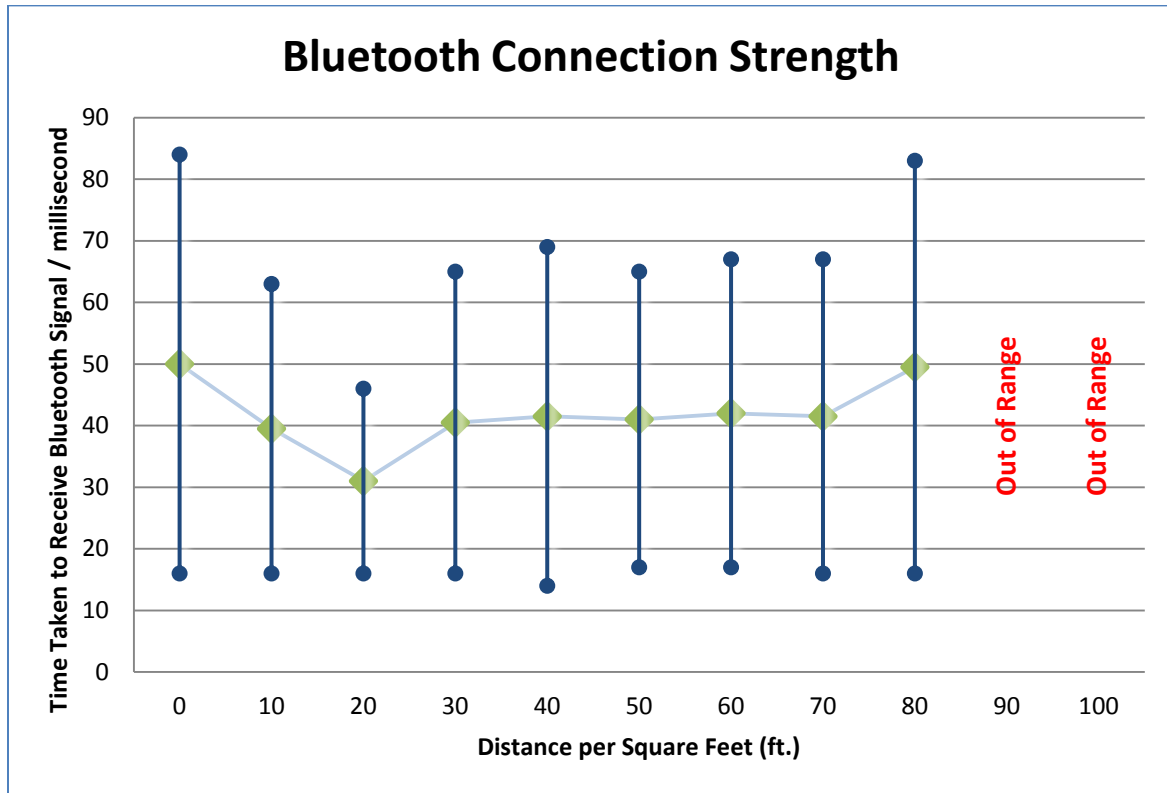
Figure 5.1: Test result for Bluetooth connection strength

The Table 5.1 above show the test result that we conduct to test the Bluetooth connection strength. We did some modification to the application in order to carry out this testing. The modification that we did to the application in order to conduct this testing is to add in a button. Once the user presses this button, it will display the maximum and minimum value which it takes to send out a single string to other connected device. During this testing, user needs to press the button once every 10 feet from the starting point until the

device is out of range. To get more accurate result, the testing is conduct at an open air area with no interference which cannot affect the Bluetooth signal. There are two purposes for us to conduct this testing. One of the purposes is to test how far the connected device can go before it become out of range. Based on the Table 5.1, it shows that the maximum distance that the connected device can go is beyond 80 feet. When the connected device goes more than 80 feet, the device connection will terminate. Another purpose is to determine how long it takes for a device to send a 512Kb string to another connected device while there is still a connection. Based on the Table 5.1, it shows that the average time it takes for a device to send a sting is around 30 to 50 milliseconds. Based on the result we found that the second testing is not feasible. In conclusion, the maximum distance that a connected device can go before out of range will be beyond 80 feet.
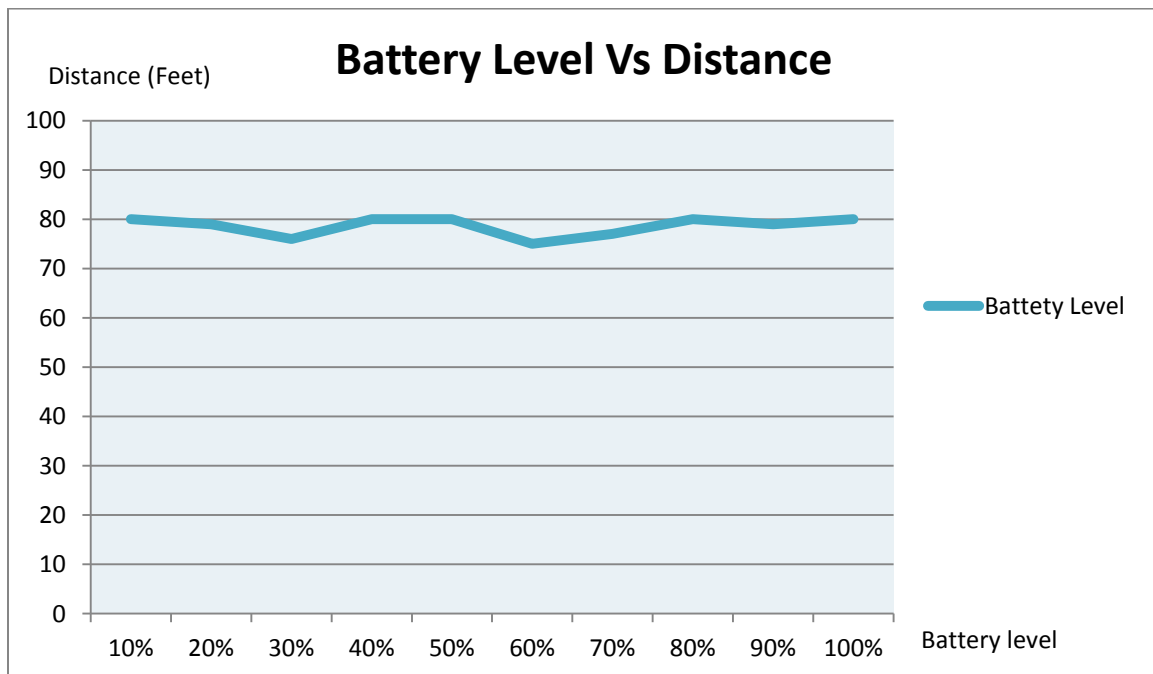
Figure 5.2: Test result for battery level versus distance

The Table 5.2 above shows the test result that we conduct to test the battery level versus the distance that a connected device can go before out of range. To do this testing we also did some simple modification to the application. Once the user presses the button that we added to the application, it will return the maximum and minimum value it takes to send

46

a string to pair device. At certain battery level, we will press the button at every 10 feet from the starting point. If the device is not out of range, the application will display the maximum and minimum value to the user. If the device out of user range, the application will not return the value and it will notify user that device is out of range. The purpose of this testing is to test whether the battery level can affect the device out of range distance. The Table 5.2 shows that, the distance that a connected device can go is beyond 80 feet. Once the device goes more than 80 feet, the connection between both devices is disconnected. This testing is not feasible to us as the battery level do not affect the distance of the connected devices.

CHAPTER 6 DISCUSSION AND CONCLUSION

6.1 Conclusion

Although there are a lot of similar applications which are available in the market, but all those applications still have some weaknesses which are not suitable to be use in our cases. In order to solve the problem, we decided to add on some new features such as providing two way notification function, allowing multiple tacking, providing functions that notifies connected devices when user devices battery is low and automatically send emergency message through SMS. After adding those features, it may be more useful and convenient to the user bases on our case. By adding in those features into our application, we are able to fulfill and meet our objective. This application will allow user to keep track and monitor any Bluetooth device they want at anytime and anywhere. It does not only allow user to keep track on only one connected devices but more than one device which mean that user can monitor multiple devices at the same time. The application also will help user to reduce lost of belonging and also reduce the need to keep their eyes on their kid every seconds or minutes. Besides that, when user face any emergency situation they can just shake the mobile device and the application will automatically send help message to other connected devices. Since the pair device user is near to the user, they can provide immediate help.

6.2 Limitation

After adding in some new features into the application, the application becomes more useful and suitable to be use in our scenario. Although adding in those features allow us to meet our objective and goal but there is still some limitation which can be improve. One of the limitations is that it only allows the user to connect maximum up to 3 devices at a time. This means that the user can only keep track or monitor maximum 3 devices at a times.

Besides that, delay in receiving notification is also one of the limitations of the application. When pair device is out of the user range, user device will vibrate, device alarm will trigger and user device will also receive SMS that notify the user that pair device is out of range or disconnected. The problem is that once the pair device is out of range or disconnected, user device will not get the notification immediately without any delay. There are few second or minutes of delay before the user gets the notification and become aware that the connected device is out of range or disconnected.

Other than that, device battery draining too fast is also one of the limitations in this application. Once the user starts up the connection between the pair device, the user device will keep track of the pair device frequently. This will cause the device battery to drain very fast.

Furthermore, one of the limitations of the application is that it only provides client-server function. Client-server function only allows the device to act either server or client at a time. With client-server function, the server device can keep track of the client device but the client device is unable to keep track of the server device.

6.3 Future Work

There are still some weaknesses or limitations in the application which can be improved, even though the application meets our objective and can be use in our scenario. One of the improvements which can be made to the application is to allow the user to connect to more than 3 Bluetooth devices at a time. This is because our application only allow user to connect to maximum of 3 device at a time, which mean that the user is only allow to keep track and monitor maximum 3 device at a time. This will cause inconvenient to the user if they want to keep track or monitor more than 3 devices. Our improvement will allow user to connect not only limit to 3 devices but more than 3 devices at a time.

Other than that, reduce delay time in receiving notification is also one of the improvements which can be made. This is because when the pair device is out of the user range, there is a delay before user receives notification and aware about this situation. An improvement can be made to reduce the delay time, so that the user can receive more accurate and efficient notification. With this improvement, our application will be more useful to the users.

Another improvement which can be made to the application is to reduce battery consumption. This is because when using the application, it needs to turn on the device Bluetooth and keep track of pair device frequently. Keeping the Bluetooth activated for a long time can also drain battery life of the mobile phone (Blogengage.com 2012), this can cause device battery to drain very fast. It is very inconvenient for the user if the battery drain too fast, the user need to make frequently recharge the device battery each time they use the application. By reducing the battery consumption, the device battery can last longer even through user is using the application.

In addition, adding peer-to-peer function is also one of the improvements which can be made to the application. With peer-to-peer function, the device can act as a client and server at the same time. This mean that one of the user devices can keep track of connected device and the connected device also can keep track of the user device at the

BIS (Hons) Information Systems Engineering

same time. Peer-to-peer function allows both devices to keep track of each other at the same time.

BIS (Hons) Information Systems Engineering

BIBLIOGRAPHY

<u>**References**</u>

Benny, G. (n.d.) *Stay Close 2 Me - Android*. [online] Available at:
http://www.androidzoom.com/android_applications/tools/stay-close-2-me_tdkf.html
[Accessed: 12 Aug 2012].

Blogengage.com (2012) *Tips and tricks that will prevent your Android's Battery from
draining quickly. | Blog Engage*. [online] Available at: http://www.blogengage.com/
blogger/prevent-your-android-battery-from-draining-quickly/ [Accessed: 12 Aug 2012].

Bolton, M. (2012) *What is Android? A beginner's guide | News | TechRadar*. [online]
Available at: http://www.techradar.com/news/phone-and-communications/mobile-
phones/what-is-android-a-beginner-s-guide-975482 [Accessed: 12 Aug 2012].

Bowl, P. (2011) *Bluetooth Tracker - Android Apps on Google Play*. [online] Available at:
https://play.google.com/store/apps/details?id=com.purplebowl.BluetoothTracker#?t=W2
51bGwsMSwxLDIxMiwiY29tLnB1cnBsZWJvd2wuQmx1ZXRvb3RoVHJhY2tlciJd
[Accessed: 12 Aug 2012].

Compnetworking.about.com (n.d.) *Bluetooth - Wireless Technology - Blue Tooth Phones
PDAs*. [online] Available at: http://compnetworking.about.com/od/bluetooth/
Bluetooth_Wireless_Technology.htm [Accessed: 12 Aug 2012].

Franklin, C. and Layton, J. (2000) *HowStuffWorks "How Bluetooth Operates"*. [online]
Available at: http://electronics.howstuffworks.com/bluetooth2.htm [Accessed: 12 Aug
2012].

Hdop (2012) *STAY HERE! BT - Android Apps on Google Play*. [online] Available at:
https://play.google.com/store/apps/details?id=hdop.hiergeblieben_bt&feature=search_res
ult#?t=W251bGwsMSwxLDEsImhkb3AuaGllcmdlYmxpZWJlbl9idCJd [Accessed: 12
Aug 2012].

BIS (Hons) Information Systems Engineering

Munson, M. (2002) *How Does Bluetooth Pairing Work? | eHow.com*. [online] Available at: http://www.ehow.com/how-does_4964578_bluetooth-pairing-work.html [Accessed: 12 Aug 2012].

Play.google.com (2011) *Baby Bluetooth Alarm - Android Apps on Google Play*. [online] Available at: https://play.google.com/store/apps/details?id=com.app.babybluetooth &hl=en[Accessed: 12 Aug 2012].

Play.google.com (2010) *BlueUnity FDL (Legacy) - Android Apps on Google Play*. [online] Available at: https://play.google.com/store/apps/details?id=com.etg.Finder.os15 [Accessed: 12 Aug 2012].

SSPARK (2011) *Bluetooth Alarm Free - Android Apps on Google Play*. [online] Available at: https://play.google.com/store/apps/details?id=jp.sspark.bluetooth.free&%20feature=search_result [Accessed: 12 Aug 2012].

Webopedia.com (2012) *What is White Box Testing? - A Word Definition From the Webopedia Computer Dictionary*. [online] Available at: http://www.webopedia.com /TERM/W/white_box_testing.html [Accessed: 12 Aug 2012].

Webopedia.com (2012) *What is Black Box Testing? - A Word Definition From the Webopedia Computer Dictionary*. [online] Available at: http://www.webopedia.com /TERM/B/Black_Box_Testing.html [Accessed: 12 Aug 2012].

BIS (Hons) Information Systems Engineering