**GPS and Wi-Fi Location Tracker**

**Pow Rei Young**

**Supervisor: Mr Teo Tse Tsong**

**Abstract**

*This project is about designing and developing a reliable and user-friendly software Android application that helps to locate the elderly with GPS and Wi-Fi information. Android Studio is used to code the application and software. The objectives of the project were achieved with two different applications developed using two different methods of getting information, SMS and OpenShift server.*

**1 INTRODUCTION**

* 1. **Background**

Mental illnesses are very common among the elderly and sometimes they pose a huge problem when they want to do daily tasks. Common mental illnesses like dementia, may cause a deterioration in memory and thinking behavior [1]. They may panic or get loss in a neighbourhood that they always go to, they may even forget where they live. Simple everyday activities can cause a huge worry for their family members. That is why a tracking application is needed to know exactly where they are, allowing family members or friends to locate and help them anytime.

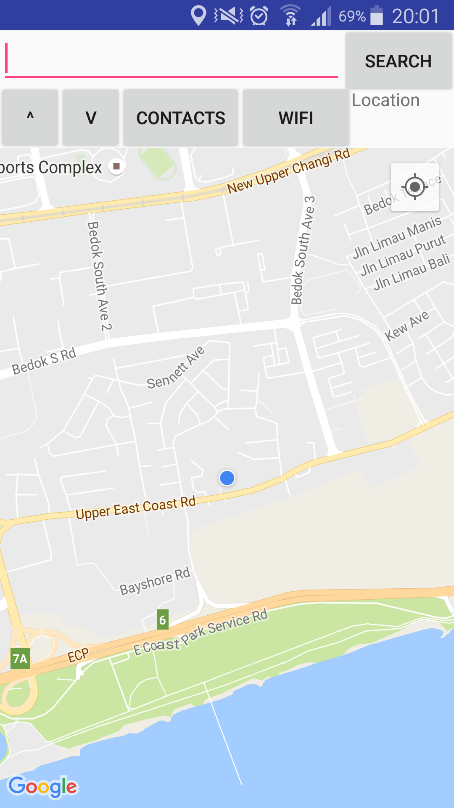
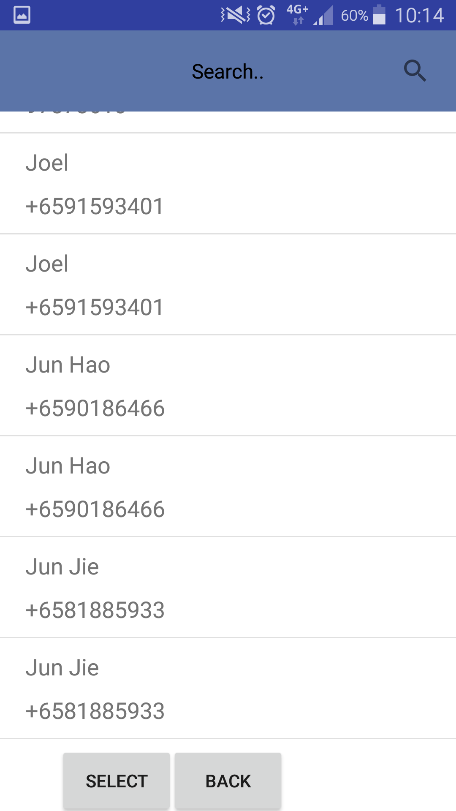
* 1. **Objectives**

The aim of the project is to design an Android tracking software that can display the location and Wi-Fi information of the elderly phone without the elderly doing anything. Wi-Fi information is needed if GPS coordinates are unavailable, allowing the user to see the closest SSIDs to the phone, like seeing Tampines Mall Free Wi-Fi and indicating that the elderly is in Tampines Mall. The application should be user-friendly and easy to use for anybody.

1. **Project Description**

**2.1 SMS Application**

The first application that was developed was the SMS application. When the application is opened, the main page will be shown (Figure 1), containing a map, a textbox and some buttons. If the phone GPS is enabled, their own location will be displayed on the map as a blue dot. From the main page, the user can search an address with the text box or zoom in or zoom out on the map. The user can also go to the contact list page to select someone to send a SMS request for collecting information or the Wi-Fi page to see the SSIDs information.

Zoom Buttons

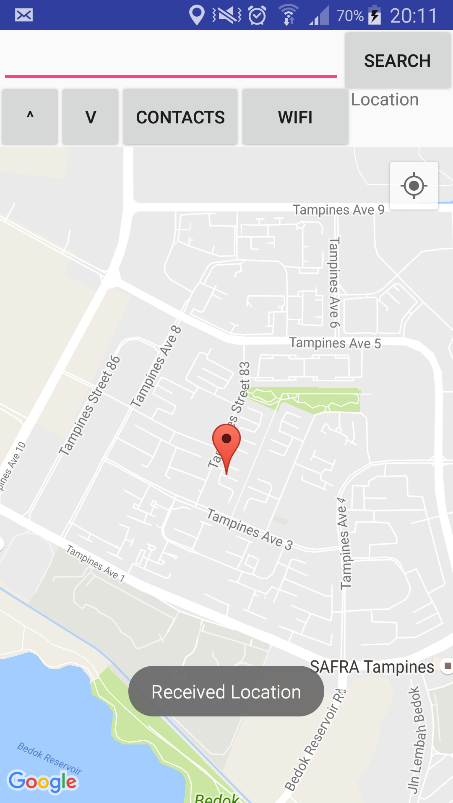
Contact List

Wi-Fi List

Search Address

**Figure 1** Main Page of SMS App **Figure 2** Contact List of SMS App

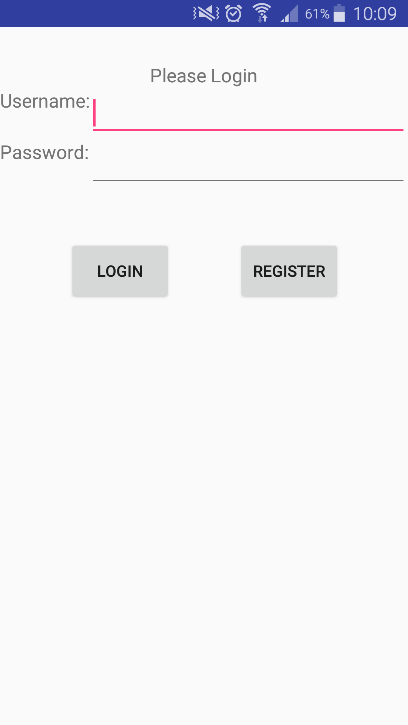
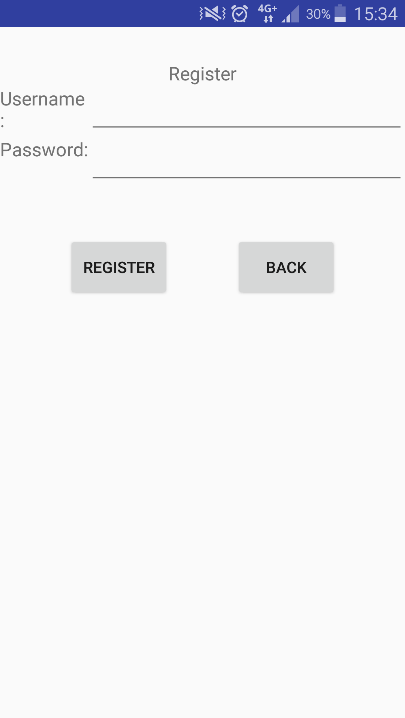
The contact list page will display all the contacts in the phone (Figure 2), allowing the user to select a contact to send a SMS with a request code inside to collect the GPS and Wi-Fi information. The elderly’s application will detect the request code from the user and send back another SMS with the information. The user’s phone will receive the SMS and read the info from it, displaying the info on the map with a red marker and the SSIDs on the Wi-Fi page respectively (Figure 3 and Figure 4).

**Figure 3** Wi-Fi page with info **Figure 4** Main Page with info

**2.2 Server Application**

The second application that was developed uses an OpenShift server to store and receive information. The application uses PHP files to connect to the server and every info, account details or friend info will go through the PHP files and the server. When the application opens, a login screen will be presented to the user (Figure 5). If the user does not have an account, they may register by going to the register page from the login page (Figure 6). Once the user logins, they will be directed to the main page of the server app, this also starts uploading the GPS and Wi-Fi info to the server in a continuous loop. The main page has an address bar for searching addresses, zoom in and zoom out buttons and a Wi-Fi button to display the Wi-Fi page, which is the same as the SMS application. The main page now has two additional options, add friend and friends list (Figure 7).

Wi-Fi Page

Search Address

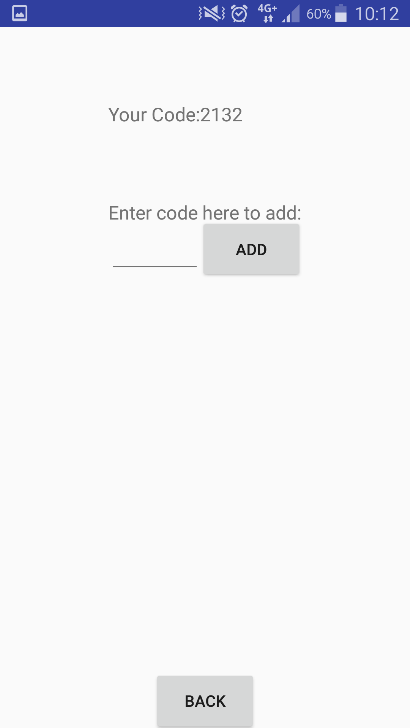
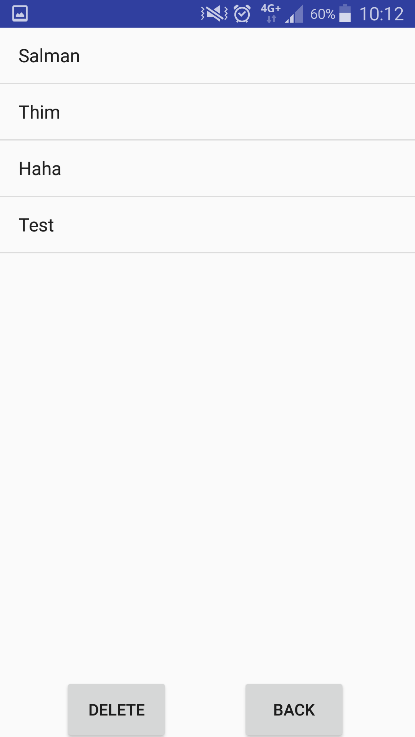
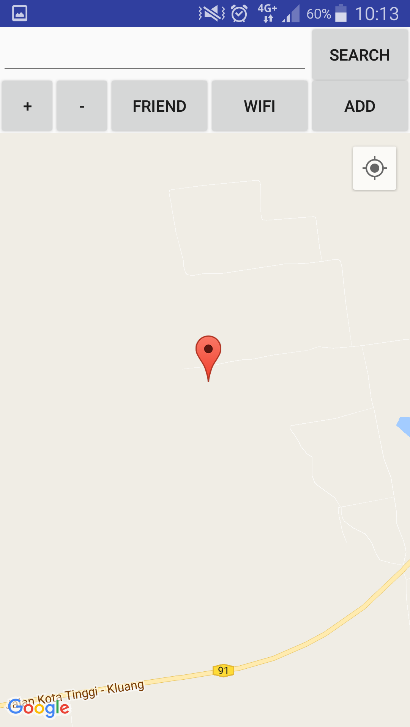
Zoom in/out

Friend List

Add Friend

**Figure 5** Login Page **Figure 6** Register Page **Figure 7** Server App Main Page

The add friend option will display the add friend page, it will contain the user’s unique code and a textbox to add a friend (Figure 8). To add a friend, the friend’s code has to be input into the textbox and entered with the “add” button, the user will be taken back to the main page if it is added successfully. If an incorrect or a duplicate code has been added, an error message will be shown to tell the user to enter again. The friends list option will display the friends list page, showing all the friends by their usernames in a listview (Figure 9). When a friend is selected, it will receive the friend’s info through the OpenShift database and start a loop to always receive the info when the info gets updated. Friends can be deleted from the user’s friend list by pressing the delete button and choosing the friend they wish to delete. The application will display the main page again and it will display the friend’s GPS location on the map with a red marker (Figure 10). The marker will update with the friend’s new location if the info gets updated showing where they are at all times. The info in the Wi-Fi page will also be updated when the friend’s info changes (Figure 3). If the user wants to see another friend’s info, they just need to move back to the friends list and choose another friend, the info will be replaced with the new friend’s info and it will receive updates when the new friend’s info changes.

**Figure 8** Add Friend Page **Figure 9** Friend List Page **Figure 10** Main Page with info

1. **Project Development**
   1. **Key Considerations**

It was decided to make an Android application that can track the elderly with any method. It needs to display the GPS and Wi-Fi information in the application. I suggested creating a hardware device that transmits GPS and Wi-Fi data for the elderly to carry, if they do not have a phone. But the idea was scrapped, as the device may be bulky and more annoying to carry around than a phone. My supervisor suggested creating an application that can transmit data using SMS, while I suggested using the method of server for transmitting data. We decided to create an application that can transmit data using SMS first and if they are enough time, create an application that uses server to store and receive data

* 1. **Stages of Development**

There are 4 stages of this project: Research, SMS Application, Server and Server Application.

* + 1. **Research**

The first stage of the project was research. Research was done before the development of the application, so I can know other tracking applications do. I researched on 4 different tracking applications on the Play Store, I noted down their delay of the updated information and how they relay information. This allows me to get a good idea about how tracking applications should be. I also researched on GitHub and SourceTree, which allows me to learn how to back up my data and it also allows my supervisor to see the constant progress that was made.

* + 1. **SMS Application**

The second stage of the project is the development of the SMS application. I had to learn how to use Android Studio and the Java language again, building a strong foundation for my coding. Many websites, like StackOverflow and YouTube, were very useful in teaching me about the different methods and features available. I managed to use the different methods I learned to generate the map and make the application detect codes in the SMS, using the SMSListener method. I wanted to delete the SMS that the application sends out automatically, but the last Android system that allowed the deletion of SMS was made in 2011 and now all the Android systems will not allow deleting of SMS. I managed to create the application with the help of video tutorials and some help I got from StackOverflow. I finished the application the week after the term break.

* + 1. **Server**

For the creation of a server, my senior offered me to use his SQL server and I accepted. I created my own tables inside his server and managed to learn the codes to direct connect my application with his server. I had almost finished creating all the codes for the connection to his server, until my supervisor advised me to create my own server with OpenShift. I decided it was for the best as I also do not want to rely on his server all the time. So I deleted most of the codes in the application and I learned how to make an OpenShift server. Thanks to my experience and research in GitHub, it was easy for me as OpenShift also uses git databases. I learned some PHP coding online, and I was able to upload PHP files to the server allowing me to connect my application with the server.

* + 1. **Server Application**

I learned some connection codes for PHP online and I made use of Android Studio built in features to connect to the uploaded PHP files. I made sure every account that was created from the application receives a unique code for adding them to the friend list. I also added many error messages if a duplicate or wrong info was added, like similar username when registering or the wrong code when adding a friend. I finished the application before the deadline and I still had some time to test it.

* 1. **Testing**

I tested the applications by trying to make them crash and seeing if they will still send data if they are closed. I also checked the update delay of the info, by using a GPS spoofing application on another phone that also has the application and checking if the info that gets updated is correct and how accurate is it. If any of the application crashes, I will try to find out why to fix it or create a try/catch statement with an error message to solve the issue.

1. **Results**

For the SMS application, info can be sent and received without the use of any internet. The SMS code can still be detected when the app is closed and the info SMS will still be sent. The SMS application cannot update as the SMS info can only show the info at the time it was requested.

For the Server application, internet is needed to relay information through the server. The info is still being updated to the server when the application is closed and the info will be updated with a 0.5 second delay. Both info that are displayed by the applications are very accurate. The SMS application is recommended for users without any data as it does not require the constant use of internet, while the Server application does.

1. **Conclusion**

Overall, the objectives of the project have been achieved, as I managed to create two Android applications that are able to track the GPS and Wi-Fi information of the elderly. Users have a choice between two different methods and I wish these applications will be used by people in the future. I wish I could create a hardware device for tracking information, but it would be a hassle as it would be too big for convenience.

**Acknowledgments**

The author would like to thank Mr Teo Tse Tsong (MP supervisor) for his guidance and his help with this project

**References**

[1] World Health Organization Site. (2008). Mental health and older adults. [Online] Available:

<http://www.who.int/mediacentre/factsheets/fs381/en/>

**Appendices**

Appendix A: User Manual

Appendix B: Block Diagrams

Appendix C: Project Poster and Flyer

Appendix D: Sequence Diagrams