Anti-Theft Application for Android-Based Smartphone

Sandy Darmowinoto[1], Seno Hadi Wibowo[2]
[1,2]Faculty of computing, President University
Jl. Ki Hajar Dewantara, Cikarang Baru, Bekasi, 17550
[1]sandy.darmowinoto@president.ac.id

Abstract— **In daily life, almost everyone uses smart phone with different types of operating system and one of them is android. People intention to steal someone else smartphone were very high. This is one of the problems for all Android smart phone users. While it was happening, they could not do anything to get it back. The objective of this research is to create an application that will help android smart phone users to get information about their phone. The information will be obtained by the user which is about the location of their android smart phones. Another feature, it can lock android smart phone while the SIM changed by other person. Only one registered SIM card that can access the android smart phone. Last thing, their phone can be checked by the trigger alarm feature. Those features can be obtained by sending the keywords via SMS remotely from another phone user. This application needs two devices in order to run perfectly as this application runs remotely by another device. Also, these applications need enough amount of pulse to send the keywords. It needs GPS and network connection to obtain the right location. This application will make the android smart phone users feel safe, secure, and comfortable and not have to worry about their phone.**

Keywords— **anti-theft application, android**

I. Introduction

Nowadays, the era of communication is really developing. At first, phone is only used at home or office and has a cable. As time goes by people have higher mobility and also the communication. Based on that problem, the innovator made the phone that usually stay in one place to become which can be held on everywhere, it called mobile phone. The functions of mobile phone were calling, sending a message, setting the alarm etc.

The innovation of mobile phone is really fast. There are so many features provided which are camera, radio, music player and the newest one is internet. Internet is the one a big influence that give a revolution from mobile phone into smart phone. Everyone uses smart phone with different types of operating system and one of them is android.

The research is intended to assist the android smart phone users to monitor and get information on their android smart phones remotely. This research developed the mobile application by using two mobile phones; android smart phone and regular phone. This research implements the idea to be able to get information from their android smart phones in real-time.

The first step to generate information from android smart phone users is sending the keyword that already set in the Anti-Theft application from another mobile phone user using SMS remotely. There is various ways which the user can get the result of this application. First, it will automatically get a reply SMS from android smart phone users to get information of location and configuration. Second, for trigger the alarm, it will ring on android smart phone user automatically. The last is to lock the android smart phone user after the SIM changed, it will lock the phone automatically.

Table 1.1 shows the comparison among those three applications (Prey-Anti Theft, Where's My Android, and Anti-Theft Alarm) with Anti-Theft. As a result the Anti-Theft application is the most complete features to protect Android smart phone than the others application.

TABLE 1.1
COMPARISON ANTI-THEFT APPLICATION WITH RELATED WORK APPLICATIONS

Features	Anti-Theft	Prey Anti-Theft	Where's My Droid	Anti-Theft Alarm
Location Keyword	v	v	v	x
Alarm Keyword	v	x	x	v
Configuration Keyword	v	v	v	v
Disalarm Password	v	x	x	v
Apps Access Password	v	v	v	v
Phone Access Password	v	v	v	x
SIM Lock Password	v	v	v	x

I. Methodology

The development of the application implements the concept of RAD (Rapid Application Development), considering the limited amount of time from design phase until deployment phase of the application.

II. Results

The development of this application involves the creation of dynamic user interface which is displayed through the view of the system. The user interface design is created based on the use case and activity diagrams developed earlier.

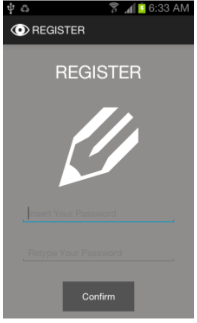


Figure 3.1 Registration Page

Figure 3.1 shows the register form of the application. Register page is only for the first user installed this application. Because this application requires a password for access to the application, because only users who can access this application. So, users must register a password before using this application.

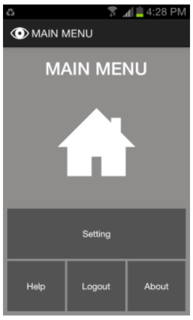


Figure 3.2 Main Menu Page

Figure 3.2 shows the main menu page. The main menu page of the application contains the four buttons that are intended to run the features according to their functions. It shows the page shown to interacting user directly after the user has been logged in successfully. As shown in Figure 3.2, there are several menus.

While the setting menu page of the application contains the four buttons that are intended to run the features according to their functions. It shows the page shown to interacting user directly after the user has chosen the setting button. There are several menus; Keyword, Password, Contact and Uninstall. Inside the after main menu page, it will display list of the setting menu, so users can see the features that available.

Figure 3.3 shows the keyword menu page. The setting menu page of the application contains the four buttons that are intended to run the features according to their functions. It shows the page shown to interacting user directly after the user has chosen the keyword button in the setting menu. Inside the after setting menu page, it will display list of the keyword menu, so users can see the features

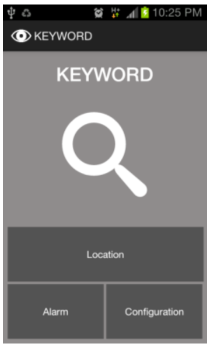


Figure 3.3 Keyword Page

Figure3.4 shows the password menu page. The password menu page of the application contains the three buttons that are intended to run the features according to their functions. It shows the page shown to interacting user directly after the user has chosen the password button in the setting menu.

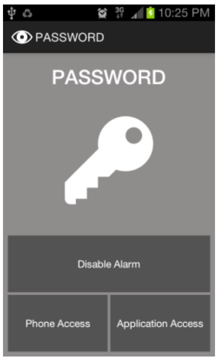


Figure 3.4 Password Menu Page

Figure 3.5 shows the contact menu page. The contact menu page of the application contains only one button that is intended to run the feature according to its function. It shows the page shown to interacting user directly after the user has chosen the contact button in the setting menu. As shown in Figure 3.5, there is only one menu. That is: Extra Phone Number. Inside the after setting menu page, it will display the contact menu, so users can see the feature that available.

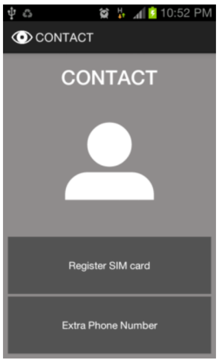
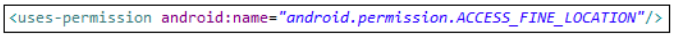


Figure 3.5 Contact Menu Page

GPS for Finding Location

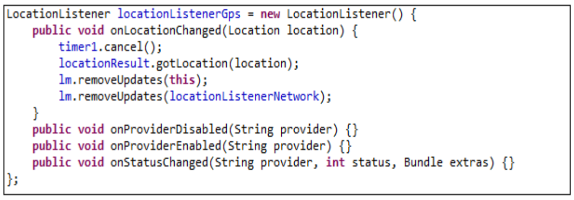
In order to shows the user's location the application needs to obtain user's location through GPS connection in android. This method implements Location Listener. In order to receive any information from GPS, android requires us to set some permission in Android Manifest that will be asked to the users when installing this system to their device. The permission required for receiving location updates is shown in Figure 3.6 and need to be put inside the Android Manifest file.



```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

Figure 3.6 Permission to access current location using GPS connection

Figure 3.7 shows how to use the Location Listener in the application. This code is responding the updates location in the application by implementing the Location Listener in the java class file.



```
LocationListener locationListenerGps = new LocationListener() {
    public void onLocationChanged(Location location) {
        timer1.cancel();
        locationResult.gotLocation(location);
        lm.removeUpdates(this);
        lm.removeUpdates(locationListenerNetwork);
    }
    public void onProviderDisabled(String provider) {}
    public void onProviderEnabled(String provider) {}
    public void onStatusChanged(String provider, int status, Bundle extras) {}
};
```

Figure 3.7 Getting location using GPS connection

Getting Location of Android Smart Phone

If the device provides a GPS and enable, the application is run using a GPS. If not, then the application uses the network connection. Since this application is not turn on the GPS automatically.

Even if the user is not moving, the application will update user's location. The time required to lock the location is 60000 miliseconds. Sometimes the current location will be different 100-1500 meters if using network connection and 3-15 meters if using GPS because of the magnetic field around the user that will affect the GPS to detect the location. In another cases, users might not be in motion for quite a long time, and for this case the system also still could access the latest location updates by calling GetLastLocation() method.

Triggering the Alarm

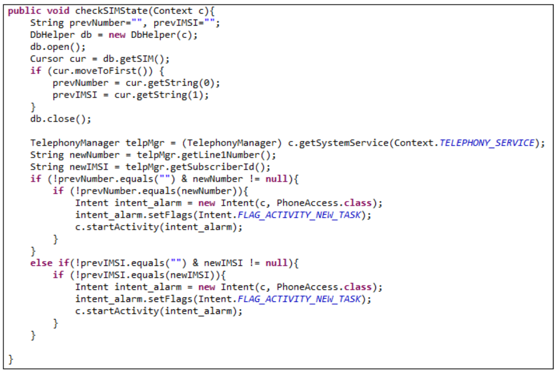
A sound file format used in this application is .Ogg format with the file name sirene. Alarm sound will continue playing until the user enters the password to disable the alarm.

If the user wants to disable the alarm, it must enter the password disable alarm on Android smart phone users correctly. If the password entered by the user is correct the sound will stop and if it is not, sounds of alarm will continue to play even in conditions Android smart phone users is locked.

SIM Card Registration and Detecting

For detecting phone and IMSI number on SIM card, user needs to register the SIM card first. In order to lock the Android smart phone user by SIM card, android requires us to set some permission in Android Manifest that will be asked to the users when installing this system to their device.

As shown in Figure 3.8, the application will run and detect the SIM card number automatically after reboot. This function will check phone and IMSI number on the new SIM card with the previous SIM card that already registered in the database. If phone and IMSI number is same with the data in the database, the application and database will close. But, if the phone and IMSI number is different with the data in the database, the application will call the PhoneAccess class file. After that, the alarm will ringing automatically until the user insert the phone access password correctly to disable the alarm and then the user can access the Android smart phone.



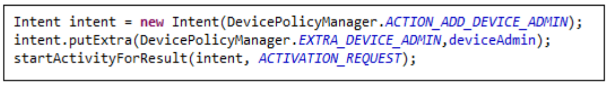
```
public void checkSIMState(Context c){
    String prevNumber="", prevIMSI="";
    DbHelper db = new DbHelper(c);
    db.open();
    Cursor cur = db.getSIM();
    if (cur.moveToFirst()) {
        prevNumber = cur.getString(0);
        prevIMSI = cur.getString(1);
    }
    db.close();

    TelephonyManager telpMgr = (TelephonyManager) c.getSystemService(Context.TELEPHONY_SERVICE);
    String newNumber = telpMgr.getLine1Number();
    String newIMSI = telpMgr.getSubscriberId();
    if (!prevNumber.equals("") & newNumber != null){
        if (!prevNumber.equals(newNumber)){
            Intent intent_alarm = new Intent(c, PhoneAccess.class);
            intent_alarm.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            c.startActivity(intent_alarm);
        }
    }
    else if(!prevIMSI.equals("") & newIMSI != null){
        if (!prevIMSI.equals(newIMSI)){
            Intent intent_alarm = new Intent(c, PhoneAccess.class);
            intent_alarm.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            c.startActivity(intent_alarm);
        }
    }

}
```

Figure 3.8 Detecting SIM card number after reboot

Register the Application as Device Administrator

To set the application as device administrator, user needs to activate it from inside the application. In order to the application cannot be uninstalled from the application list, Android requires us to set some permission in Android Manifest that will be asked to the users when installing this system to their device.

As shown in Figure 3.9, this code is used to register the application as device administrator. The variable of ACTIVATION_REQUEST is asked the user to confirm the policy of device administrator. If the user disagree with the policy, then the application can not be used. Because this application need to be enabled as device administrator in order to avoid the another person to uninstall this application. But if the user has agreed with the policy, the application will runs and enable to uninstall the application from inside the application using uninstall feature.

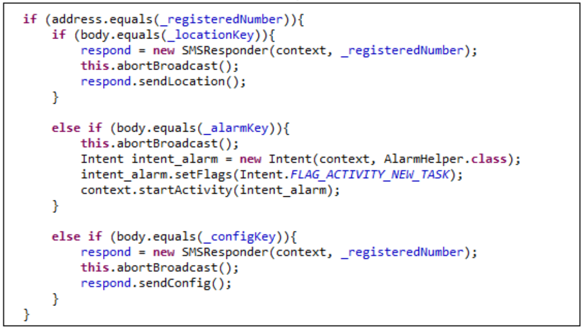


```
Intent intent = new Intent(DevicePolicyManager.ACTION_ADD_DEVICE_ADMIN);
intent.putExtra(DevicePolicyManager.EXTRA_DEVICE_ADMIN,deviceAdmin);
startActivityForResult(intent, ACTIVATION_REQUEST);
```

Figure 3.8 Register the application as device admin

SMS for Receiving the Keyword

If the user wants to send a keyword, the user has to register the extra phone number, can be seen with the parameter _registeredNumber. Each function of feature will be execute on this class. In this case SMSReceiver class will call some of the functions that have been made on SMSResponder class as shown in Figure 3.9.



```
if (address.equals(_registeredNumber)){
    if (body.equals(_locationKey)){
        respond = new SMSResponder(context, _registeredNumber);
        this.abortBroadcast();
        respond.sendLocation();
    }

    else if (body.equals(_alarmKey)){
        this.abortBroadcast();
        Intent intent_alarm = new Intent(context, AlarmHelper.class);
        intent_alarm.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        context.startActivity(intent_alarm);
    }

    else if (body.equals(_configKey)){
        respond = new SMSResponder(context, _registeredNumber);
        this.abortBroadcast();
        respond.sendConfig();
    }
}
```

Figure 3.9 Respond the keyword

III. CONCLUSIONS

Developing Anti-Theft application can help android smart phone users around the world who want to feel the security and comfort when their android smart phone was stolen by thieves. Of course, this application is better, simple, and easy to use. The concern by android smart phone users when they lost their phone, now was replaced by an android application that installed on their android smart phones. With only need mobile phone, GPS or network connection and a few pulses, users can feel comfortable in using this application. Based on that, the application offers advanced features but easy to use.

This application helps the users to track the location of their android smart phone with appropriate accuracy. Users can also trigger alarm on their Android smart phone and the alarm can only be turned off by the user with enter the disable alarm password. This application can also lock the android smart phone users when they replaced the SIM card by the other person after the SIM card registered. And if the users forgot about the keyword and password, the user can uses the configuration feature to return the keyword and password from the database. All these features it is very easy to use, because it is only by using GPS or network connection, alarm, SIM card, and SMS. Thus, the convenience and security of android smart phone users will not be difficult anymore to obtain.

REFERENCES

[1] Alienman Tech (2012, October). Where's My Droid. Retrieved November 2012, from: https://play.google.com/store/search?q=wheres+my+droid&c=apps

[2] Catherine. (n.d.). What is GPS?. Retrieved November 2012, from http://www.wisegeek.com/what-is-gps.htm

[1] Dennis. (n.d.). What is SMS?. Retrieved November 2012, from http://www.wisegeek.org/what-is-sms.htm

[2] Digital Dividend. (n.d.). Mobile SMS and Data Roaming Explained. Retrieved September 2012, from http://serving.webgen.gsm.org/

[3] Fork Ltd. (2012, July). Prey Anti-Theft. Retrieved November 2012, from https://play.google.com/store/search?q=prey+anti+theft&c=apps

[4] Kathink. (2010, October). International Mobile Subscriber Identity (IMSI). Retrieved December 2012, from: http://learntelecom.com/telephony/gsm/international-mobile-subscriber-identity-imsi mmapps mobile.(2012, August). Anti-Theft Alarm. Retrieved November 2012, from: https://play.google.com/store/apps/details?id=mmapps.mobile.anti.theft.alarm

[5] Rouse, M. (2007, February). Rapid Application Development. Retrieved September 2012, from: http://searchsoftwarequality.techtarget.com/definition/rapid-application-development

[6] TELECOM BLOG. (n.d.). MSI, format, definition with examples. Retrieved December 2012. From: http://www.imsi.biz/

[7] UK Android Forum. (n.d.). What is an Android Phone?. Retrieved November 2012, from http://androidphone.org.uk/guides/what-is-an-android-phone