**Software Description**

Title: **Android Personal Data Collector Application**

Our research related with user personal behavior. This application created because we need to collected user log data in theirs smartphone as much as possible without forgetting about the quality of data. In context of collecting personal user data we have two methods, first is participatory method which need user intervention for collecting the data. The second method is opportunistic method which do not need user intervention for collecting data. So, this application following opportunistic method which is after user install this application they do not need to doing anything. After installation this application automatically collect the all of user data in theirs smartphone.

The list of probes that we implemented in this application as follows:

1. **HardwareInfoProbe.java**

This file contain function that we use to get the device information such as device brand, model, device ID, and etc. The size of this file is 3 KB and with 68 lines code.

1. **BatteryProbe.java**

This file contain function that used for detecting action battery changed. It means such as when user charge and uncharge, USB, or directly to power, etc. The size of this file is 2 KB and with 60 lines code.

1. **AccountsProbe.java**

This file contain function that we use to get the account information in user smartphone. New smartphone store information about accounts that user use such as facebook, twitter, or another social media that user used. The number of lines code is 69 with size of file 2.3 KB

1. **ApplicationProbe.java**

This file contain function that we use to get information about installed application in user smartphone. This information is very useful for our research, because our research focus on to define user behavoir based on their smartphone log data so application is one of features that can be used to define user behavior. The number of lines code is 130 with size 4.7 KB.

1. **AudioMediaProbe.java**

This file contain function that can be used for getting information about what kind of audio file in user smartphone. The data that we collect with this function such as date add, date modified, display name, mime type, size, album, title, artist, composer, duration, is ringtone, etc. The number of the code is 86 lines with size 3.3 KB.

1. **BrowserBookmarkProbes.java**

This file contain function that we use for collecting Bookmark. Based on user bookmark, we can know about what kind of site that user like. The data that we collect with this function such as title, url, visit, date, created, and bookmark itself. The number of code is 62 lines with size 2.3 KB

1. **ImageMediaProbes.java**

The functionality of this function is similar with AudioMediaProbe.java, but this file is for collecting information related with image file in user’s phone. The data that we collect using this function such as date added, date modified, display name, size, title, date taken, description, latitude, longitude, orientation, and etc. The number of code is 76 lines with size 2.7 KB.

1. **RunningApplicationProbe.java**

The function of this file is to retrive what kind of current running application in user’s smartphone. The number of code is 190 lines with size 6 KB.

1. **ScreenProbe.java**

The function of this file is to store when user screen on their phone and when the phone screen of. Base on this data, we can know user intensity of phone used. The number of code is 92 lines with size is 2.6 KB.

1. **VideoMediaProbe.java**

This file is similar with AudioMediaProbes and ImageMediaProbes but the functionality of this code is for collecting information related with video in user’s phone. The data that collected by this code such as date added, date modified, display name, mime type, size, title, album, artist, bookmark, catefory, date taken, description, duration, latitude, longitude, resolution, and etc. The line of code is 84 lines and size 3.2 KB.

1. **GravitySensorProbe.java**

Those functions are to get the information about device position. Based on information that we get from those sensors, we could define the device position. The size of this file is 2.8 KB with code 71 lines.

1. **LightSensorProbe.java**

This file contain code that exploit ilumination sensor. The data that we get from this function is light sensor data such as the brightness, the light condition near with mobile device. The number of code is 43 lines with size 1.4 KB.

1. **OrientationSensorProbe.java**

The oriantation in android are azimutch, pitch, roll, with this function we can get those of data. The number of code is 47 lines with size 1.6 KB.

1. **PressureSensorProbe.java**

Some of news type of android supported with pressure sensor. In this code we implement code to get the value of pressure. The number of code in this file is 43 lines with size 1.4 KB.

1. **ProximitySensorProbe.java**

A proximity sensor is a sensor able to detect the presence of nearby objects without any physical contact. In this file contain code for collecting the data from this sensor. The value that we get in this file is distance, the distance between phone and another object. The number of code is 52 with size 1.8 KB.

1. **TemperatureSensorProbe.java**

Not all devices has temperature and pressure sensor but the newest device version already provided those sensors. In this application we also tried to develop code that can collect the information about the temperature and ambient air pressure. The number of code is 51 lines with size 1.8 KB.

1. **AccelerometerSensorProbe.java**

This function is for getting accelerometer values (X,Y, abn Z axis). The number of code is 46 lines with size 1.5 KB.

1. **ActivityProbe.java**

This file contain code that processing accelerometer value to three type of user activity, there are : high, low, and none/idle. So, based on this function we can know what kind of user activities such as high or low activity or maybe idle means no activity. The number of code is 148 lines with size 4.9 KB.

1. **GyroscopeSensorProbe.java**

This file is for getting Gyroscope values (X,Y,Z axis). The number of code is 57 lines with size 2.3 KB.

1. **LinearAccelerationProbe.java**

If we looking information in Android reference we will see something like this “acceleration = gravity + linear-acceleration”. So this function is for getting value of linear acceleration. The number of code is 51 lines with size 2 KB.

1. **RotationVectorSensor.java**

The rotation vector represents the orientation of the device as a combination of an angle and an axis, in which the device has rotated through an angle θ around an axis (x, y, or z). The number of code in this file is 46 lines with size 1.7 KB.

1. **BluetoothProbe.java**

This file contain function that we use to get the nearby bluetooth signal. This data can be used as features for our research. The number of code is 140 lines with size 4.5 KB.

1. **LocationProbe.java**

This file using GPS to get the current user location. The main values that we want to collect are latitude and longitude. The number of code is 185 lines with size 5.8 KB.

1. **WiFiProbe.java**

This file has similar function with BluetoothProbe.java but in this file we use WiFi to get the nearby Access points. The number of code is 199 with size 6 KB.

1. **CallLogProbe.java**

This file contain function that we use to get call log information. Android system store all user of call log activity. In this function we get the all of call log information such as the number caller, duration, type (answer, miss, etc). The number of code is 64 lines with size 2.2 KB.

1. **ContactProbe.java**

This file contain function that can be used for collecting contact database in user smartphone. Android system store all of contact database. So, we can get it, the information related with contact are : phone number, email, name of contact, etc. The number of code is 297 lines with size 14 KB.

1. **SmsProbe.java**

This file contatin function that can be used for collecting SMS log such as when user sent SMS, length of message, to whom, etc. The number of code is 78 lines and size 2.8 KB.