Development of a Multi Sensor Android Application

Veerababu Velugubantla
Sasanka Maddala

Department of Applied Signal Processing (TISB)
Department of Mathematics and Natural Sciences (TIMN)
Blekinge Institute of Technology
SE-371 79 Karlskrona Sweden

This thesis is submitted at Blekinge Institute of Technology in partial fulfilment of the requirements for the degree of Master of Science in Electrical Engineering with emphasis on Signal Processing. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Author(s):
Veerababu Velugubantla
E-mail: veve15@student.bth.se
Sasanka Maddala
E-mail: sama15@student.bth.se

University advisor:
Benny Lövström
Department of Applied Signal Processing (TISB)
Department of Mathematics and Natural Sciences (TIMN)

University examiner:
Dr. Sven Johansson
Department of Applied Signal Processing (TISB)
Department of Technology and Aesthetics (DITE)

Blekinge Institute of Technology
SE-371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

ABSTRACT

There has been an enormous growth in usage of smartphones in recent times. Smartphones are not limited for communication purpose. It has various applications designed as per daily requirement of human such as for web searching, online shopping, bank transactions, games, etc. With the increase in the usage of the smartphone, the more user information is captured and stored by it, which raises the question of security. The goal of this research is to develop two android applications. One is sensor detector application and the second is screen lock application.

The first application will help the user to identify all the hidden sensors and working sensors in the mobile phone. This application even describes the features and usage of every sensor in detail. Using a graphical description of each sensor which depicts the behaviour of each sensor as per environment/movement. The second application is designed using a combination of two sensors. Screen lock applications contains two main factors. One is to work properly in all cases and efficiently do the functions that are required to do. The second is to maintain a smooth inner system interaction because in addition to locking the screen this application should make sure to hide the display of all the other applications without closing the process of those applications. With the increase in the usage of the smartphone, it becomes difficult for older generations to memorize the security pattern techniques and use them. This thesis develops a simple technique in the mobile authentication android application.

Thesis is developed on Android studio platform. The background functionality of the app is coded in java using android SDK tool and frontend of the application is designed using XML files. The GENYMOTION emulator and a mobile phone are used to test the output.

ACKNOWLEDGEMENTS

We are thankful to our mentor who is also our supervisor Benny Lövström. Their knowledge regarding android development and system interaction really helped us a lot to understand the working of system working on smart phones. Their guidance helped us to give the proper outlook and framework to our application, so it could be presentable to the user. Their guidance is a limelight for us.

On the second number we thank our parents. Without their support and motivation, we would never be able to reach on this position. Parents have supported us on every level. Even when didn't believed in ourselves they believed in us and pushed us to work harder and further. There constant support is the reason that we have come to this position that we are making a whole android application with our own knowledge.

Thank you all.

CONTENTS

LIST OF FIGURES

1 INTRODUCTION

Android is currently the largest smartphone operating system all around the world. This operation system is used so much widely that every person at some extent has its knowledge. When a system is so much popular and as many people are familiar with the system then the security risks also get higher for the system. The security measurement should on a level that no user should be able to fetch the data of the other user [1]. With all the data securities we must take care of many other things in the android system. This thesis works on development of two android applications. The first one is sensor detector and the second is a lock screen application which aims in security of an android phone.

The sensor detector application has an inventory of all the sensors and it will give the information regarding only the sensors available in the mobile phone after testing the presence of each sensor in the mobile phone. After installation of this application in an android phone and once the application is opened, it reads the complete mobile phone system and lists the available sensors in the mobile phone. In addition, the application provides an option to test the functionality of the sensors.

The second application focuses on providing the security. The front end we use is made of xml files. The back-end working is done with java files but the button we click on the smart phone screen are the part of xml files. Our job is to close that xml file with our application. When you will click on the power button, the application will close the screen and it will become dark. To turn the phone screen on, need to click the main power button or double tap the screen (applicable in some android phones) which will turn the phone on. This application gives you double option at the same time you can either use the application or the main feature if the smart phone to operate the on and off function of the screen. The application has a grey background and two white colour flat buttons on it. On button is to turn off the screen. The second button is to calibrate. It helps you to test the functioning of the application. The application will interact with the operating system to turn off the screen and this gives a need to maintain the security of the system.

1.1 Thesis statement

The main purpose of this thesis is to develop two android applications. The applications designed in thesis are a sensor detector and a lock screen application. Sensor detector application lists out the sensors present in the mobile phone. A daily requirement application is developed using these detected sensors combinations. In this thesis, a lock screen application is designed using a combination of accelerometer sensor and proximity sensor. The applications are tested on android operating system mobile phone.

1.2 Research questions

The research focus on the development of sensors related android applications. The research questions are listed below.

- Can the sensor detector application list out all the sensors available in a mobile phone?
- Are the sensors functionality test outputs being correct?
- Does the developed lock screen application give better performance compared to regular one?
- Can the lock screen application make the screen off and sensing feature faster and reliable?

1.3 Objectives

The objectives of this thesis are:

- Literature reviews on the existing sensor related android applications.
- Finding different types of sensors and studying their purposes.
- Eventual modification of the applications design as per the thesis goal.
- Verify the system measurements to give a proper security to the phone.
- Verify the optimized way in design to get the security features in android phone.
- Implementation of designed algorithms for applications on Java using android SDK tool.
- Measurement of the performance by testing the applications on an android mobile phone.
- Improvement of applications design to maintain the required security level.

1.4 Research methodology

In this thesis, the main application is to design a complex application based on sensors available in an android mobile phone. We will use two types of research methodologies. One is constructive research and the second is nomothetic research.

1.4.1 Constructive research

This type will be used to understand and discover the frame works used in the android application. It will help us to understand and find out the reason to use a specific framework and give a proper outlook to the user to user and decide the way of working of the front end.

1.4.2 Nomothetic Research

This research will help us to test our idea that we have made regarding testing the sensors and locking the screen in the mobile phone. With this approach it will be easy for us to work and identify the procedure of locking and unlocking the screen.

1.5 Thesis organization

Chapter 1 Introduction is the introduction to the project that includes problem statement, research questions and a brief research methodology of this thesis. Chapter 2 Related Work deals with background work and previous research work related to this thesis. Chapter 3 Theory includes a brief theory about the sensors and frameworks used for development of thesis. Chapter 4 Experimental Setup discusses the interaction of java and xml files and procedures used in this thesis work in development of the two applications. Chapter 5 Results and Discussions contains the results of sensors features and its graphs. This chapter also includes discussions on the screen lock application results. Chapter 6 Conclusion and Future Work includes conclusions derived from the analysis of experiment results, answers to the research questions and scope for future research work.

2 RELATED WORK

Research in android applications is a popular topic with the development in several algorithms and requirements for humans. Significant research has been carried out in the development of an android application area for different types of uses. The choice of the application and its development method changes with the application purpose. However, there exist many applications in practice according to the needs. Some published research papers related to android applications developments are presented as follows.

Authors in [1] presents a study of how different sensor fusion algorithms perform in low cost hardware and in high acceleration scenarios. For this purpose, an Arduino MKR1000 is used together with an accelerometer, gyroscope and magnetometer. The objective of this thesis is to choose the most suitable algorithm for the purposed practical application, which consists on attaching the device to a moving object, such as a skate board or a bike. Once the orientation is estimated, a movement recognition algorithm that was developed is able to match what trick or movement was performed. The algorithm chosen was the Madgwick one with some minor adjustments, which uses quaternions for the estimation and is very resilient when the device is under strong external accelerations.

Authors in [2, 2] presents an innovative research approach to design and develop mobile consumer applications across communication network. The design involves mobile client that receives services provided by the base station server connected to database server. Target consumer applications implemented are Electronic Health Record (EHR) and Utility billing application. These applications are tested for correctness and analysed for performance with varying input data type and size. The applications are also verified and validated by porting on a mobile device that runs android operating system. The proposed research finds use in upcoming mobile based consumer applications for achieving increased cost-effectiveness, scalability and ease of use.

Authors in [3] presents the architecture of Android platform, including the classes and methods in developing it. Android is an open and free operating system based on Linux, which is mainly used for mobile terminals, such as smart phones and panel computer. It is developed by Open Handset Alliance composed of more than 30 technology companies and mobile phone companies. Android tries to allow users experience the best service quality, and allow developers get a more open level for more convenient software developing. Thus mobile applications with more convenient functions can be developed via Android. This thesis takes audio/video file procurement as an example to introduce the Android program design and development, including classes application, program design, development and analysis.

Authors in [4] enhances the speech quality analysis process used when testing the GSM (Global System for Mobile communication) network. One goal was to analyse if it was possible to record and inject sound into a phone call and verify the possibility to execute the PESQ (Perceptual Evaluation of Speech Quality) algorithm using a smartphone. Another goal was to develop a smartphone application to perform the described tasks. The development process included an investigation whether the Android platform supplies the functionality required. The investigation showed that no general solution for recording and injecting audio into a phone call could be found. Recording was possible on some smartphones, but the characteristics of the recorded audio were not as desired. No solution for injecting audio into a phone call could be found for any of the tested smartphones.

Authors in [5] examines the use of sensors available in smartphones to implement an application for fall detection. The proposed solution is to use accelerometer data coming from the embedded sensors available in smartphones. This data can be fed into a finite state machine to detect possible fall candidates. Properties are extracted from the data, which is analysed by a pre-trained neural network that perform a classification of the event. The evaluation of the accuracy shows that the iOS and Android implementation reached a success rate in classifying events correctly of 91% and 83%, respectively. The evaluation of battery life shows that this solution can be implemented without consuming to much battery power.

Authors in [6] developed an application for smart parking. People are using cars for transportation every day all over the world. This leads to a constant demand for parking. In public areas there are usually designated places for parking that many people share. The shared parking spots with the best location and cheapest fee thereby become attractive. Some motorists cruise around hoping to find an attractive spot that is not occupied. This cruising is generating a lot of unnecessary Carbon Dioxide as well as causing other problems on the streets. This thesis proposes a solution to these problems by providing motorists with real-time information about the availability of spots in parking houses through a smartphone application. In order to reach as many potential users as possible the smartphone application was developed with PhoneGap and standard web technologies. To achieve scalability on the back end Google App Engine was used. By focusing on multi-story car parks and parking lots the cost for deploying the system is inexpensive due to the low demand for sensors compared to those that require a sensor for each parking spot.

Authors in [7] presents an implementation for a typical Android application to support Plug-ins or an add-on application to enhance functionality at run time. Such a framework allows an application to offer new functionality without updating the base application every time in the application store. Our implementation uses advanced bound services and AIDL concepts to provide communication between the application and its plugins. It implements an installed plugin discovery mechanism and also provides deployment from Plugin SDK perspectives. The plug-ins developed with this framework implementation use the typical Android method for application development, which is the Android SDK. Therefore, no extra skill set is required for first or third party developers to write such plug-ins. This thesis presents how the plug-ins are discovered and launched from the parent application, and also how the parent application can expose APIs which are utilized by the plug-ins. It considers typical native Android applications as well as web applications as plugins. Identifies key problems in implementing web applications as plugins and their solutions. In addition, analysis of how to open the plug-in development kit to third party developers i.e. deployment scenarios of the plugin SDK. Finally, the time taken to execute an API which transfers various sizes of data over bound service inter process communication is determined.

3 THEORY

In this chapter, an explanation about the sensors considered in this thesis are presented. The sensors considered in this thesis are Accelerometer, Barometer, Compass, Gravity sensor, Gyroscope sensor, Linear accelerometer sensor, Light sensor, Orientation sensor, proximity sensor and Rotation sensor. A brief description about the sensors and frameworks used for development is given in the following subsections.

3.1 Java and XML

Java is a computer coding language. In development of an android application, it handles all the backend functions. Java is simple to use and there are no pointers in this language. The main objective of this thesis is to develop two android applications and these two applications need both front end and back end development. The front-end development is done using XML language. XML language is more similar to HTML, but it is widely used in development of android systems. Using this language, the opening screen of the application and buttons required for operations are developed. The functions of the buttons and its operations for executing its purpose are developed using java.

3.2 Android Operating System

This thesis implementation is based on Android operating system for a mobile device. Android is currently the most used mobile operating system in the world, has more users, more phones and more tablets worldwide than any other mobile operating system. In addition to popularity of the android phones, the open source code is available for everyone for free to use. The following figure is the description of Android framework.

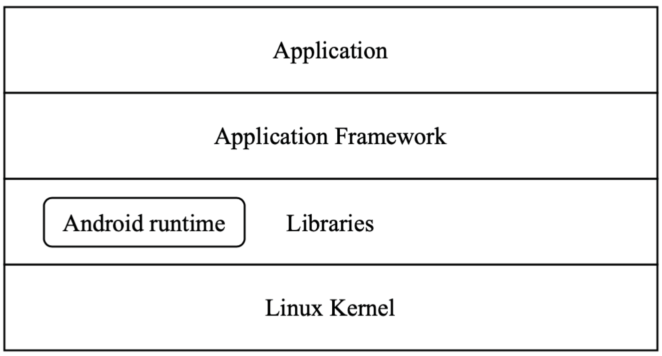

Figure 1 Android framework

As it is illustrated in Figure 1, the top layer of the framework is application interface to the user. Examples are contact address book, phone keypad used to dial. The second layer is the application framework. In this layer the operations of the application are carried out as per user needs. This layer gives high level Java class services to applications such as activity manager, resource manager, notification manager and content provider. The third layer is for all the libraries set such as SQLite database, Web browser web kit, etc. This layer is a set of Java based libraries. In this same layer one section is reserved for android runtime which provides a component called DVM. DVM enables each android application to run on its own process and instance. It also provides some libraries which allows developers to write code using Java language. The bottom layer is a customized embedded Linux system which interacts with the phone hardware and contains hardware drivers of camera, keypad etc.

3.3 Sensors

A sensor is a converter that measures a physical quantity and converts it into a signal which can be read by a user or an instrument [6]. Most of the modern smartphones devices are equipped with various kind of sensors like accelerometer, barometer, compass, gravity, gyroscope, magnetometer, light, GPS, etc. A sensor can be hardware-based like Accelerometer or Gyroscope sensors. These sensors has a physical component built into the device and it obtains the data by directly measuring specific properties.

Gyro sensor mainly detects the earth's gravity and based on that one can determine phone's orientation. This is a free rotating disc which is mounted on the spinning wheel. It could also be a software-based sensor sometime called virtual sensor and it obtains the data from one or more hardware-based sensors such as gravity sensor.

3.3.1 Sensor Categories

There are three different kind of sensors in an android platform. They are explained in the following subsections.

3.3.1.1 Motion Sensors

Motion sensors are primarily used to monitor the movement of the device. These sensors can be hardware based or software based. Hardware based sensors are gyroscope and accelerometer sensor. On the other hand, gravity, linear acceleration and rotation vector sensors can be either hardware based or software based sensors. Motion sensors are very useful for detecting the shaking, rotation, swing and tilting movements of mobile device. Motion sensors are generally used with some other type of sensors such as position sensors to show the relative position frame of reference.

3.3.1.2 Position Sensors

Position sensors are primarily used for determining physical position of mobile phone. There are two types of position sensors. They are

1. Geomagnetic field sensor
2. Orientation sensors

The geomagnetic sensor is a hardware based sensor while orientation sensor is a software based sensor. Orientation sensor derives the data from hardware based sensors such as geomagnetic sensor and accelerometer sensor.

3.3.1.3 Environment Sensors

Environment sensors are hardware based sensors and these are not usually available in all the mobile devices. There are four types of environment sensors. They are

1. Ambient humidity
2. Illuminance
3. Ambient pressure
4. Ambient temperature

These sensors are used to measure the environmental pressure, humidity, air temperature etc.

3.3.2 Sensors Descriptions

A brief description of each sensor [8] used in this thesis is given below.

3.3.2.1 Accelerometer

This sensor is a type of hardware based motion sensor. This sensor measures the acceleration force in m/s which is applied on the mobile phone on all the three physical directions (x, y and z). These directions even include the direction of gravity. This sensor is used to detect movements of the mobile phone like shaking, tilting, etc.

3.3.2.2 Barometer

This sensor is a type of environmental sensor. This sensor is used to measure the air pressure in certain environment. Many measurements of air pressure are used within surface weather analysis to help find surface troughs, pressure systems and frontal boundaries.

3.3.2.3 Compass

This sensor is a type of position sensor. This sensor is used to give the information regarding right directions with respect to north and south magnetic poles of earth. The needle present in the compass sensor always points towards geometric north direction of earth. Digital Compass Sensor is actually a magnetometer that can measure the Earth’s magnetic field.

3.3.2.4 Gravity Sensor

This sensor is a type of motion sensor. This can be either a hardware or software based type sensor. This sensor measures the gravitational force applied on the mobile phone in all the three physical directions (x, y and z) in m/s. This sensor also captures the movement data of mobile phone.

3.3.2.5 Gyroscope Sensor

This sensor is a type of motion sensor. This is a hardware based type sensor. This sensor measures the rate of rotation of the mobile phone around each of three physical directions (x, y and z) in rad/s. This sensor derives data like spinning of the mobile phone.

3.3.2.6 Linear Acceleration Sensor

This sensor is a type of motion sensor. This can be either a hardware or software based type sensor. This sensor measures the acceleration force in m/s^2 which is applied on the mobile phone on all the three physical directions (x, y and z). The functioning of this sensor depends on the presence of gyroscope on a device. If gyroscope is present then the sensor uses gyroscope and acceleration as input and if there is no gyroscope it uses the input of magnetometer instead of gyroscope.

3.3.2.7 Light Sensor

This sensor is a type of environmental sensor. This sensor is a photodetector used to sense the amount of ambient light present in the mobile phone environment and appropriately dim or bright the phones screen to match the exterior environment.

3.3.2.8 Orientation Sensor

This sensor is a type of software based position sensor. This sensor measures the degrees of rotation that a device makes around all the three physical directions (x, y and z). The mobile phone position can be retrieved using this sensor in the mobile phone.

3.3.2.9 Proximity Sensor

This sensor is a type of environmental sensor. This sensor is used to detect the presence of nearby objects without any physical contact. This sensor in a smartphone is primarily used to detect the user's face is close to the screen. Due to this sensor, the phone screen operates to turn off itself when the user holds it up to ear during a phone call which prevents any errant button presses.

3.3.2.10 Rotation Sensor

This sensor is a type of motion sensor. This sensor can be either a hardware or software based type sensor. This sensor measures the orientation of a device by providing the three components of a rotation vector of the mobile phone. The three components are obtained with respect to the three physical directions (x, y and z).

3.3.3 Sensor Parameters

There are some parameters of sensors which help in choosing the sensors as per the application requirement [6]. The study of these parameters helps in determining the capacities and limitations of the sensor. Few parameters are given a brief description in following subsections.

3.3.3.1 Range

A range in a sensor which indicates the minimum and maximum values that the sensor can measure. If the real variable goes above or below that threshold values, the output value will be limited to the maximum. It is expressed in the same units that the sensor measures, g for accelerometer, degrees/s for gyroscope and gauss for magnetometer.

3.3.3.2 Sensitivity

Sensitivity is defined as the slope of the characteristic curve of the sensor. This is the relation between the input variable (i.e. an acceleration) and the output variable (i.e. a voltage). So its unit is determined by the variable that the sensor measures and the output that it gives. In digital sensors, it is common to give the sensibility in LSB/u o u/LSB, where u is the unit that the sensor measures. This means that when the LSB bit changes, the real variable changes in u units.

3.3.3.3 Precision

Precision parameter is related to the degree of reproducibility of a measurement. Therefore a very precise sensor will output the same (i.e. voltage) when the same input is given. Note that is not the same as accuracy.

3.3.3.4 Accuracy

The accuracy parameter is the difference that exist between the real value and the measured value for every sensor. Note that one sensor can be very precise but not accurate.

3.3.3.5 Noise

Noise parameter will depend on the internal characteristics of the sensor and the operation mode. It is typically measured in LSB rms but in some applications it is given as a function of the operation frequency.

3.3.3.6 Rate

Rate defines the sampling frequency of the device, measured in Hz. So it is the number of samples that the sensor can take per second. This parameter can usually be chosen among different values.

There are also other characteristics such as power or current consumption, temperature of operation, supply voltages, connection interface, dimensions or weight. They all can be checked in the data sheet of the sensor.

4 EXPERIMENTAL SETUP

This chapter gives description about the methodology and application development procedures used in this thesis implementation.

4.1 Methodology

There are several stages in this thesis development. The following figure is a flowchart of the steps involved to complete this thesis.

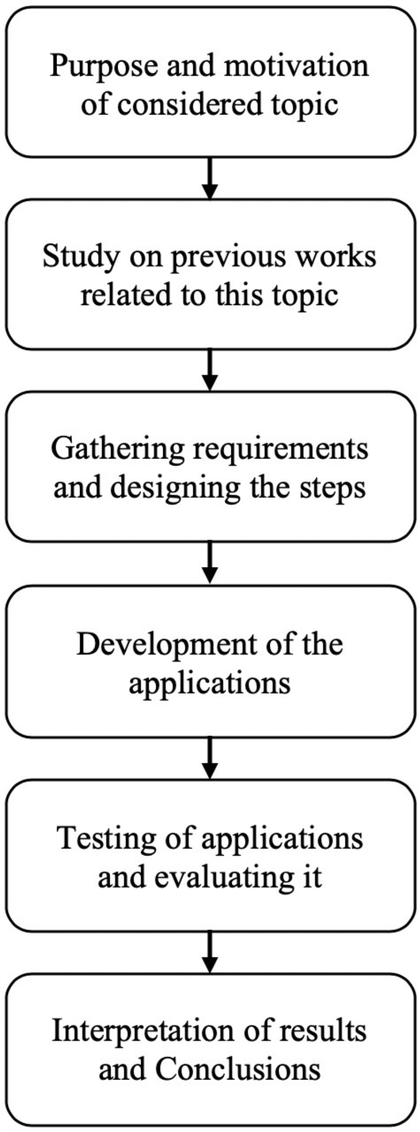

Figure 2 Workflow steps

The Figure 2 shows an overview of the different stages in the development of thesis. As the figure depicts, initially started with finding motivation and usages of considered topic and then reading previous similar works. The next phase was to generate different ideas through prototyping and finalize the applications to be made. Then was to implement the applications. The next stage includes the testing of the applications and then final stage includes the results and the suggested future work of the thesis.

4.2 Interaction between Java and XML

The two programming languages play equal role in the development of the two android applications. As mentioned before, the front end development is implemented using XML language and the back end is developed using java language. The main challenge is to make a good connection between the XML files and the java files. Compared to XML language, the java language has more functionalities. In addition, the coding part of java is more complex compared to the XML coding part. This is because java works on the main part of development of the android application which involves the internal analysis of whole mobile phone in both the android applications.

4.3 Applications Development

In this thesis, two android applications were developed. The first one is sensor detector application. The second one is lock screen application. The stages or process undertaken during the development of these two applications are explained in this section.

4.3.1 Sensor Detector Application

This application will list out all the sensors available in the mobile phone in which this application is installed. The various stages involved during the development of the application are explained in the following subsections.

4.3.1.1 Methodology

To list out all the sensors available in a mobile phone, this sensor detector application has to access all the data available in the mobile phone. After reading the data, this application will show the result of the sensors available in the application page. The application will automatically analyse all the data available in the mobile phone when the application is installed for the first time. Then the application will find the sensors located in any place of the mobile phone by detecting the file types that are carried by a sensor and then it will add the sensor name as a button in the first display page of the application. When the user clicks on the button of a particular sensor, it will redirect to the properties graph of that particular sensor. The application has a lengthy execution process.

The coding is developed on android studio IDE and then the GENYMOTION emulator for testing purpose. This emulator is much faster than the android's original emulator and even more reliable. Firstly, the application is tested on the emulator in a computer. The results contains all the sensors available in the emulator. Then the code is dumped on the mobile phone of models OPPO NEO7, SAMSUNG S8 and J7 max. After that, the application is tested and even in this case the results have the list of all the sensors available in the mobile phone. After seeing the detected sensors list, the ratio of accuracy provided by our application is even tested. The results are mostly similar and accurate to the sensor properties.

4.3.1.2 Intent Technique

The intent query in java is used to switch from one page to another page. In the sensor detector application it is working dynamically. In any application most of the time the intent query is kept constant and we do not need to generate pages during run time but in this sensor detector application as the main purpose is to detect the sensors, there is a need to generate different number of graphs on different pages to show the output of each sensor. So in this application intent is working dynamically. Intent will generate as much pages as much the sensors are in the application. The result of the number of sensors will be shown to the user as a list and every sensor name be shown to the user as a button and as we will click the button, this will direct the user to the next page and the graph of sensor properties will be displayed to the user on the next page.

4.3.2 Lock Screen Application

This application will lock the mobile phone screen when the specific instructions from user is given. These specific instructions are registered while designing the application. The sensors used to design this application are acceleration sensor and proximity sensor. The process of this application development is explained in the following subsections.

4.3.2.1 Methodology

To lock the screen in this application, the instruction is given using the proximity sensor. The function of the proximity sensor is to detect presence of nearby objects without any physical contact. The proximity sensor makes sure that there is no physical contact of user on the front side of the mobile screen. As soon as we hover our hand/any object over the proximity sensor the device gets locked. To unlock the device, we shake the device thereby unlocking it using data from accelerometer sensor.

The first screen of the application has a button named ‘CALIBRATE’. This screen appears only for the first time when the user is using this application. Clicking on this ‘CALIBRATE’ button, the mobile phone is calibrated and displays a sensor value with the help of the proximity sensor. It only requires for the device to calibrate once.

All the security features of the mobile device like pin, pattern lock need to be disabled by the user before using this application as these security features will override this application functionality. Once the proximity sensor is calibrated, then we can lock the device with proximity sensor data and unlock it using accelerometer data.

4.3.2.2 Intent Technique

In the proximity sensor based lock screen application, the intent is used in constant number. There are fixed number of pages and every accessible button on the application takes you to another page. The intent technique ends as soon as the proximity lock functionality is initialised by clicking the start button. The intent works only 3 to 5 times in the application. The rest of the work is executed directly by the application.

5 RESULTS AND DISCUSSIONS

5.1 Sensor Detector Application

This section discuss the results obtained for this application in both the testing cases. The first testing case is testing the application on emulator where the sensors are detected from Genymotion and the second testing case is testing the application on an android mobile phone.

5.1.1 Testing on Emulator

The following figure is the screenshot after the compilation of the sensor detector application.

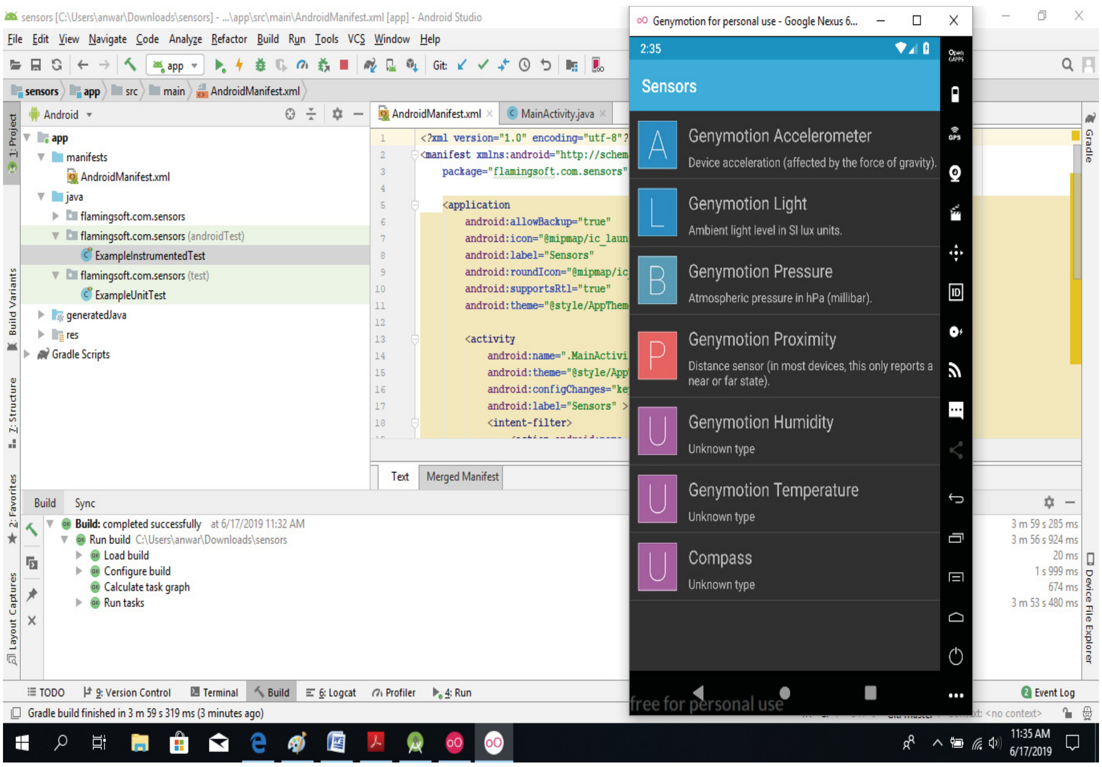

Figure 3 Sensors detected in the emulator

The Figure 3 is the emulator screenshot. After running the application, the output gives the list of detected sensors. In this case, the emulator is working as a mobile phone for the application. Therefore the application has detected all the sensors that are shown in the emulator.

After displaying the sensors available, this application give us the option to check the properties or features of every listed out sensor. The application will be able to access the performance of the sensor by itself.

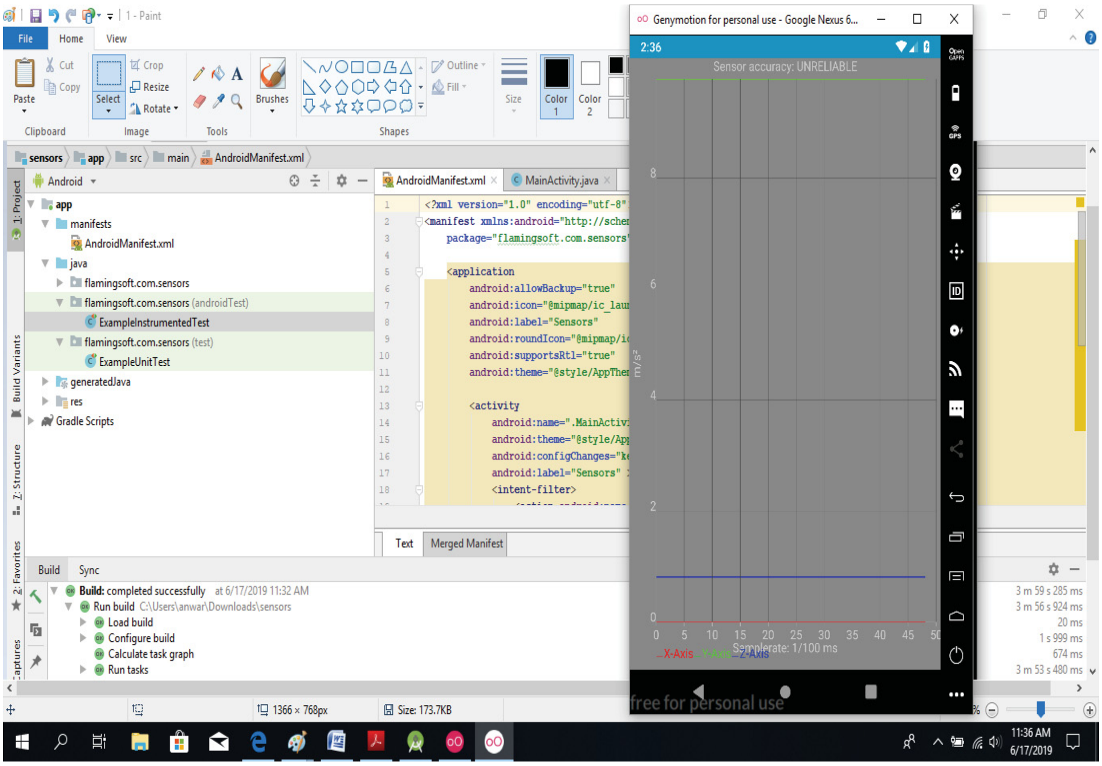

Figure 4 Graphical representation of genymotion accelerometer sensor properties

In the Figure 4, the red, blue and green straight lines represents the performance of the sensor. To measure the performance of the sensor, a test case is made to run on the sensor by the application. This process is done by the application by itself. This helps in generating the output regarding the working module of the sensor. The graph shows the acceleration in 3 dimension when a phone is kept still. This application helps the user to know the performance of phone based on the sensors performances.

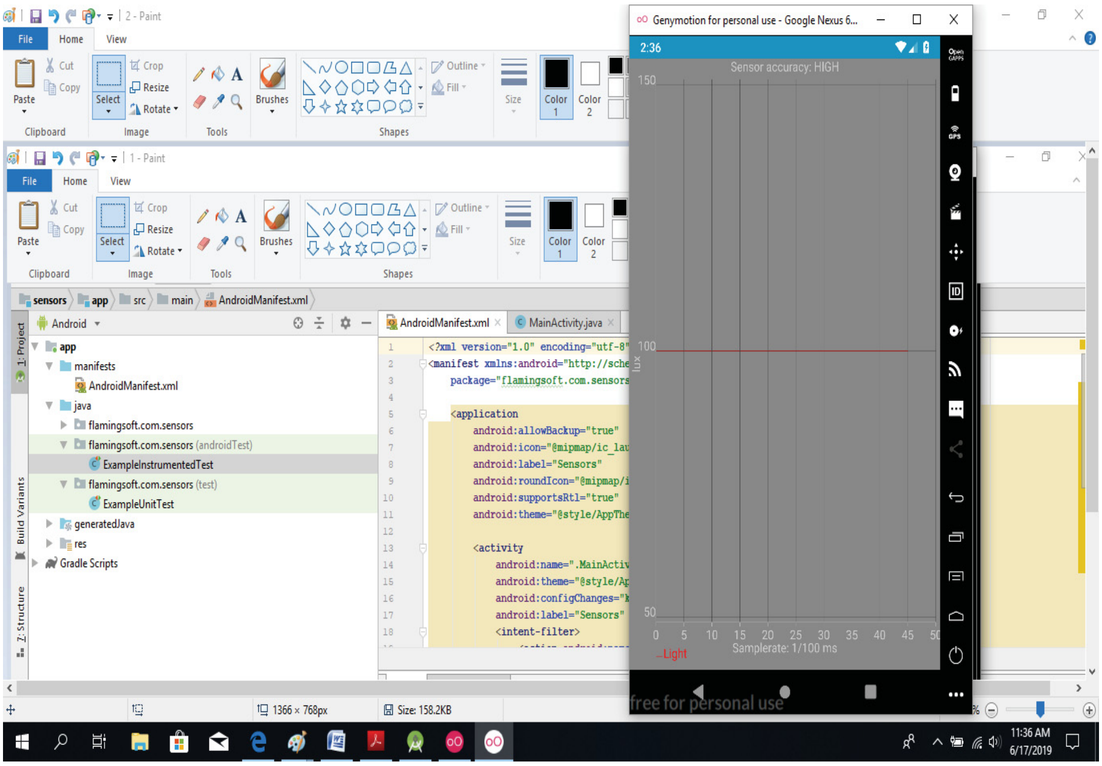

Figure 5 Graphical representation of genymotion light sensor properties

In the Figure 5, the graphical representation of genymotion light sensor is shown. It detects the ambient light in the area where the mobile unit is present. The red line indicates the light detected by the sensor. In this case it is flat as this is an emulator. The unit of measurement is lux (l).

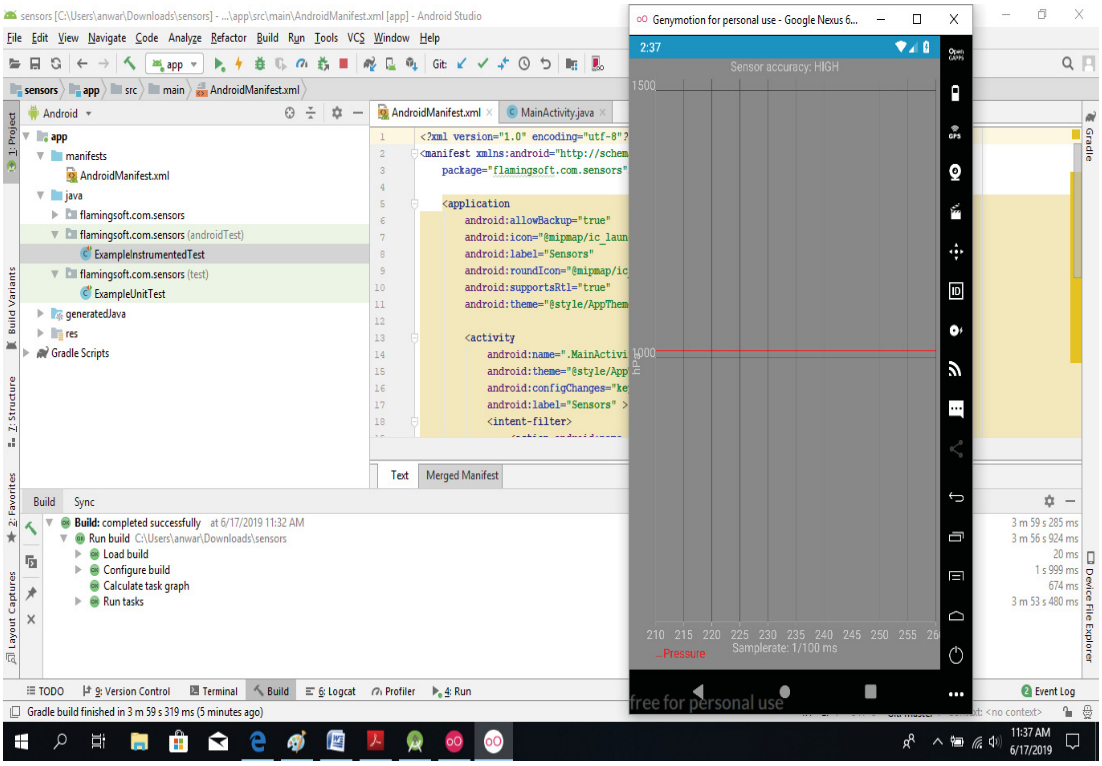

Figure 6 Graphical representation of genymotion pressure sensor properties

The Figure 6 is the graphical representation of genymotion pressure sensor. This sensor measures the ambient air pressure around the mobile unit. It is used to monitor any air pressure changes in the environment. The ambient air pressure is measured in hPa or mbar.

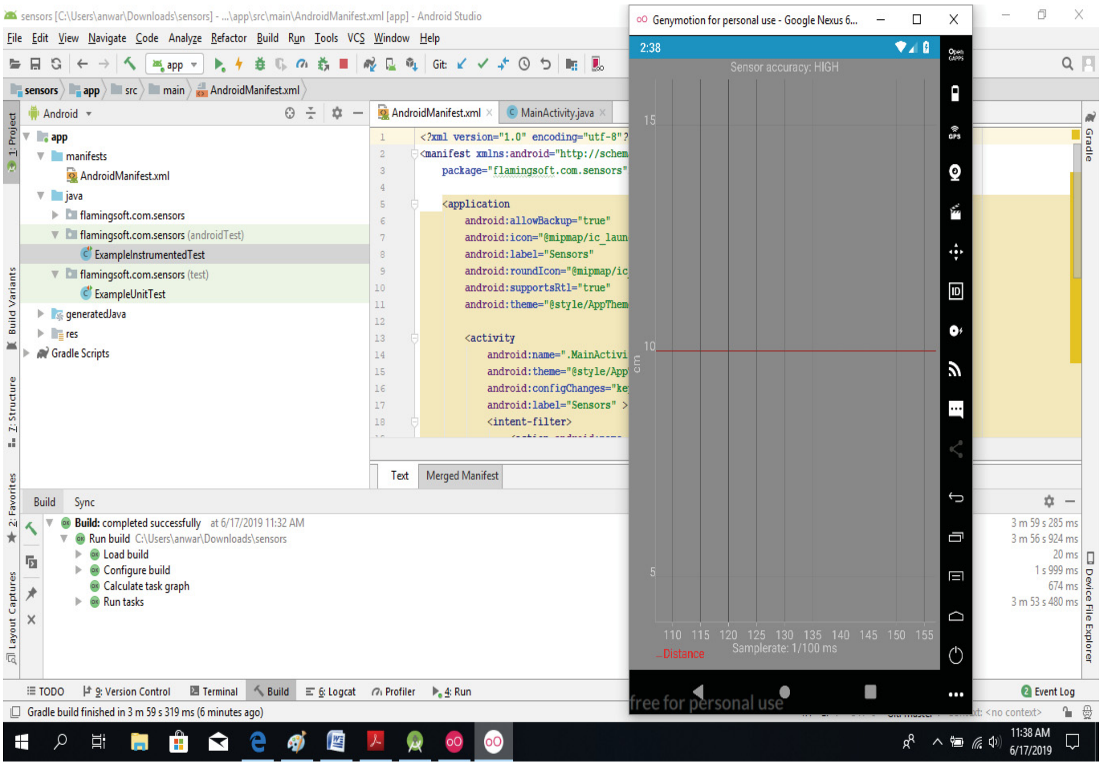

Figure 7 Graphical representation of genymotion proximity sensor properties

In the Figure 7, the graphical representation of genymotion proximity sensor is shown. This sensor is used to detect the distance of a foreign object from the mobile unit. Main use of this sensor is to detect the distance with user's ear during a phone call and the display of user's mobile screen will be closed based on that distance. The distance is measured in cm.

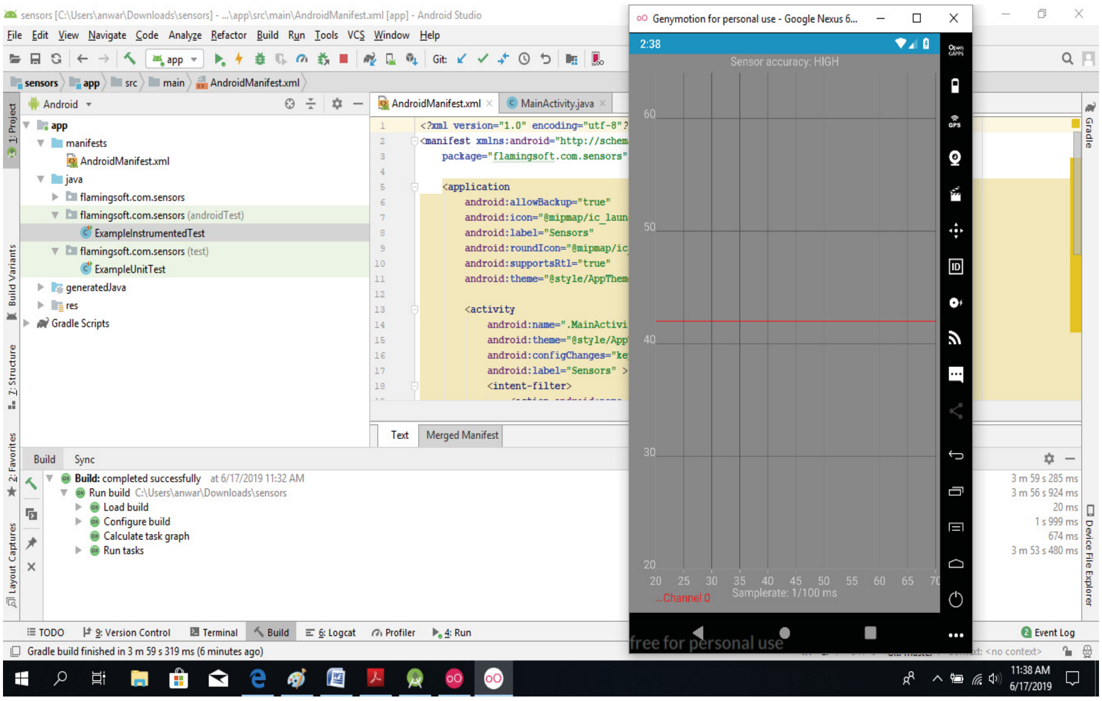

Figure 8 Graphical representation of genymotion humidity sensor properties

In the Figure 8, the graphical representation of genymotion humidity sensor is presented and this sensor is used to monitor dewpoint, absolute and relative humidity. Unit of measurement for humidity sensor is percentage (%).

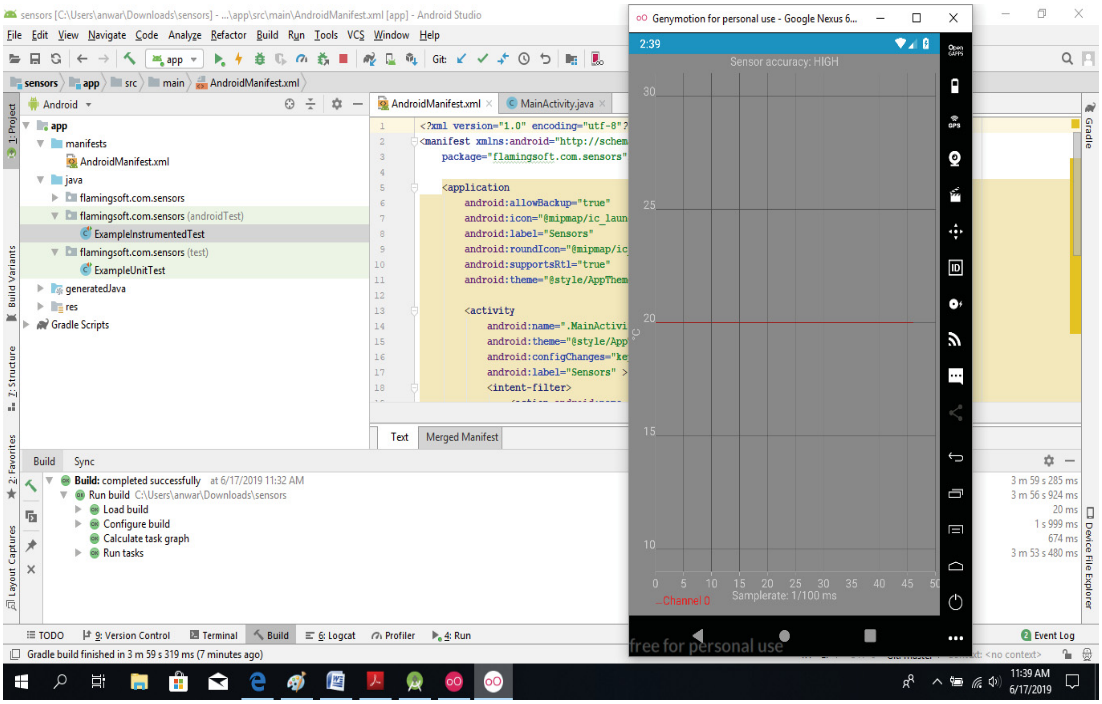

Figure 9 Graphical representation of genymotion temperature sensor properties

The Figure 9 is graphical representation of genymotion temperature sensor which is used for monitoring temperatures. The temperature of the mobile is done in Celsius (°C).

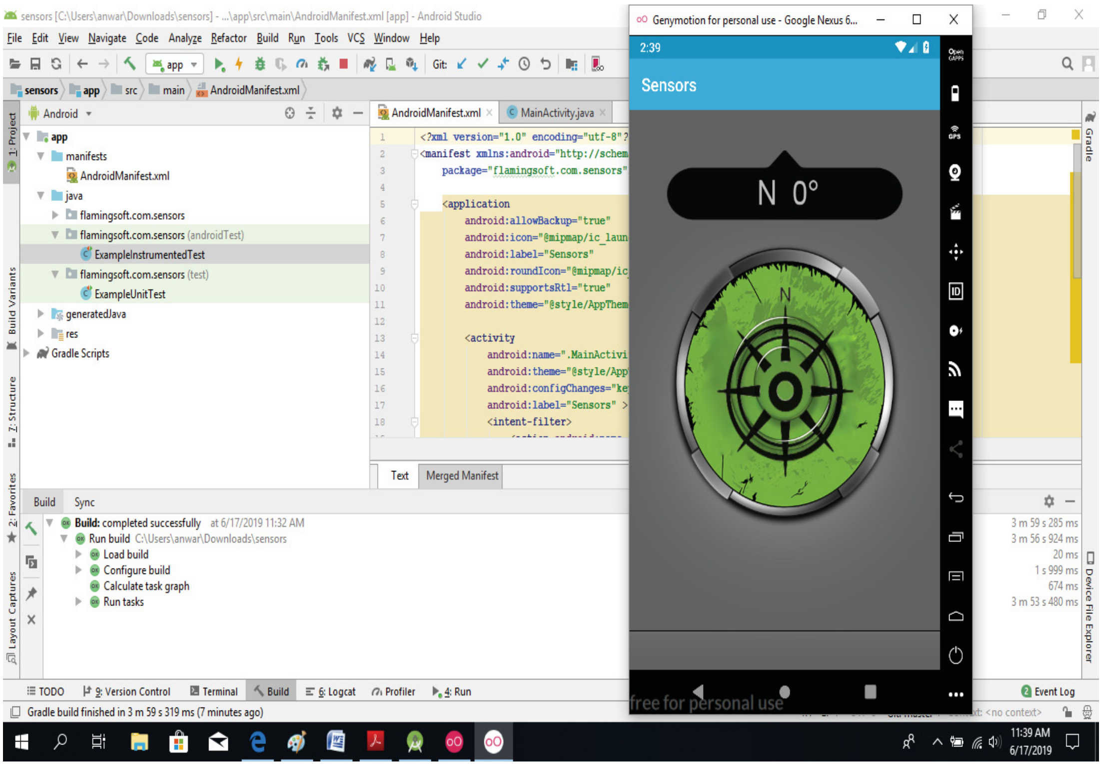

Figure 10 Graphical representation of compass sensor properties

In the Figure 10, the graphical representation of compass sensor is shown. This sensor uses magnetometer to measure the strength and direction of magnetic fields. It basically displays the cardinal directions of the earth's magnetic field.

5.1.2 Testing on Mobile Phone

The developed application is also tested on an android mobile phone. The results are shown in the form of screenshots captured from the mobile phone. They are displayed in the following figures.

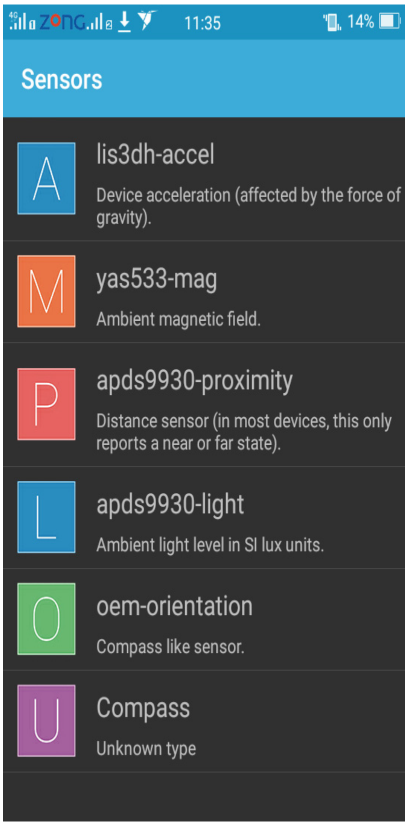

Figure 11 List of sensors detected in a mobile phone

The Figure 11 is the screenshot of a mobile phone viewing the list of detected sensors when the application is tested on a mobile phone.

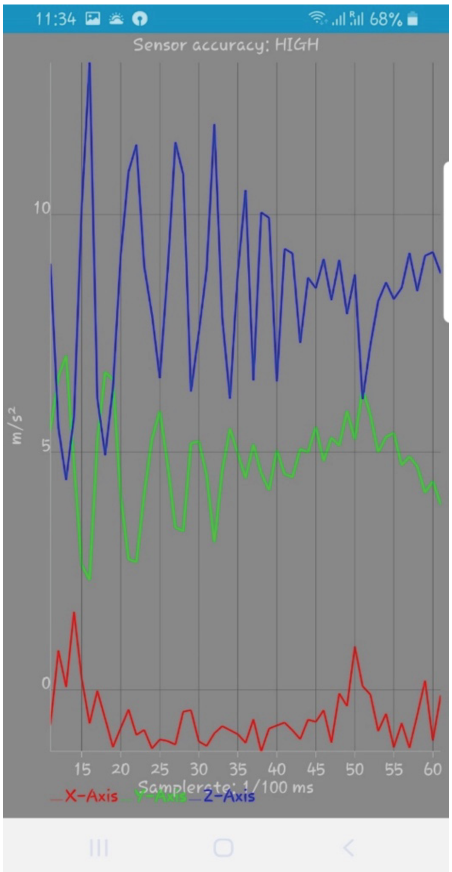

Figure 12 Acceleration sensor properties graph

In the Figure 12, accelerometer sensor measures the acceleration force applied on a particular device. The measurement is done in three-dimensional axis x, y and z. Red line represents x-axis, Green line represents y-axis and Blue line represents z-axis of the mobile unit. When the device is moved up and down with some external force, sensor measures in z-axis direction and when the device is moved sideways, sensor measures in x-axis direction. Similarly, when the device is moved towards and away sensor measures in y-axis direction.

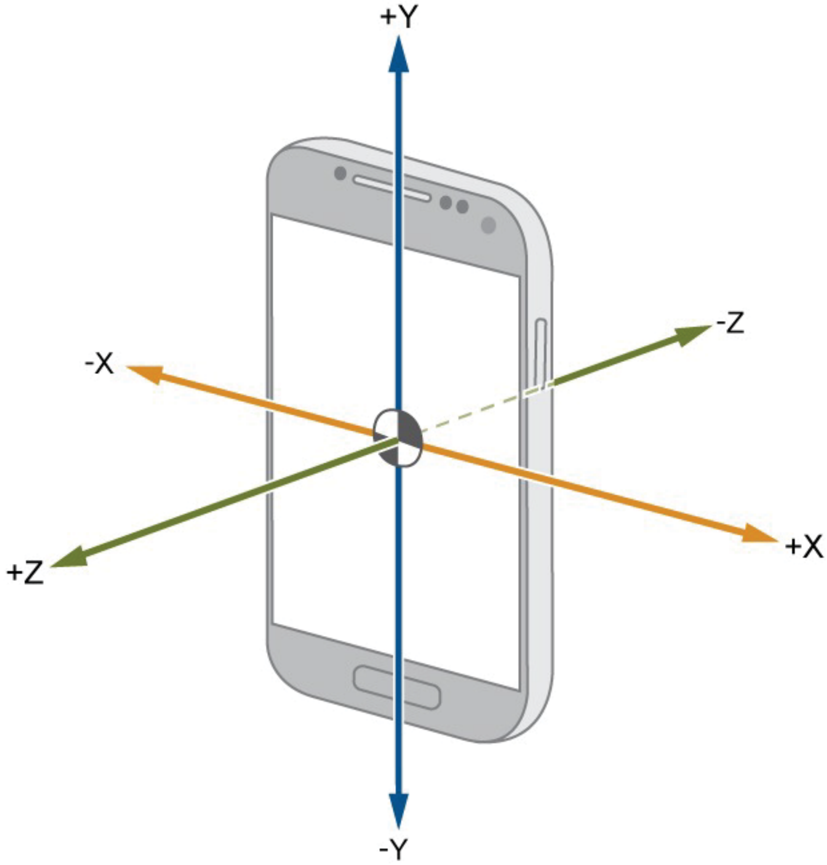

Figure 13 Representation of X, Y and Z axis directions

In the Figure 13, the x, y and z directions relative the mobile phone is shown. The motion sensors measure the data as per these directions.

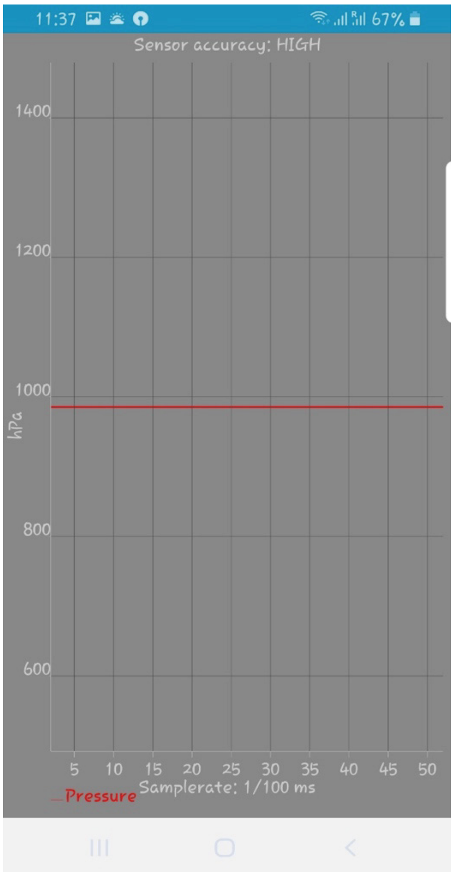

Figure 14 Barometer sensor properties graph

In the Figure 14, Barometer sensor is used to detect the altitude of the mobile unit from the sea level. This uses the GPS chip inside the mobile unit to provide the altitude data.

Figure 15 Compass sensor properties graph

In the Figure 15, Compass shows the cardinal directions of earth's magnetic field. For example, in figure it shows that the mobile unit is in pointing towards 163° South.

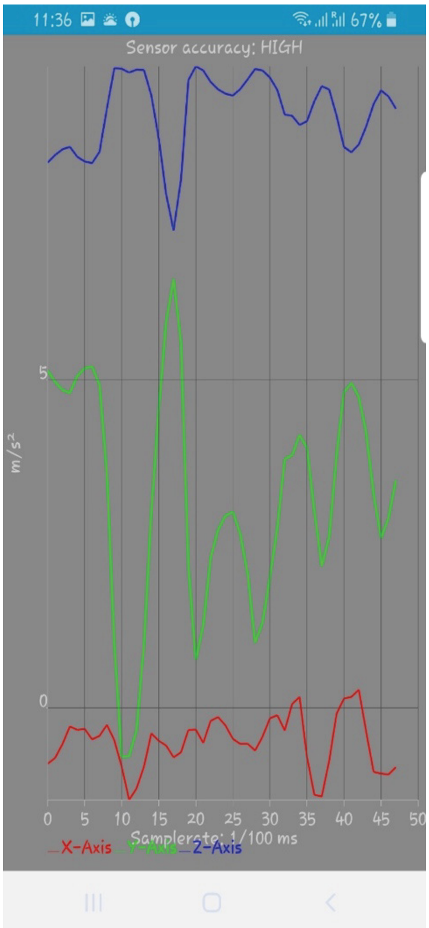

Figure 16 Gravity sensor properties graph when mobile phone is moving

In the Figure 16, Gravity sensor measures the gravitational force applied on the mobile unit on all physical axes x, y and z in m/s^2 . The graphical data shown in figure gives the force applies on the mobile unit when it is shaken or tilted.

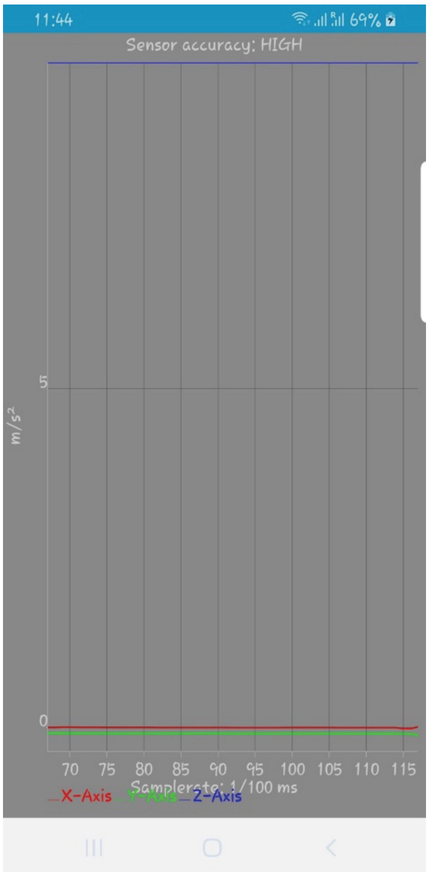

Figure 17 Gravity sensor properties graph when mobile phone is stable

In the Figure 17, the graphical data of gravity sensor when the mobile unit is stable i.e., when the mobile placed on a desk or on ground but not in a user's hand. In this case, the gravity sensor measures the data only in z-axis as the acceleration towards earth gravity. Its value is 9.81 m/s^2. The x-axis and y-axis are perpendicular to earth gravity acceleration. Therefore, no data is measured in these directions. This can be clearly shown in above Figure 17.

Figure 18 Gyroscope sensor properties graph

In the Figure 18, gyroscope measures the rate of rotation of mobile unit on all physical axes in rad/s. In the graph, Red line represents x-axis direction measures, Green line represents y-axis direction measures and Blue line represents z-axis direction measures of the mobile unit. These lines are changed based upon the spins or tilts of the mobile unit.

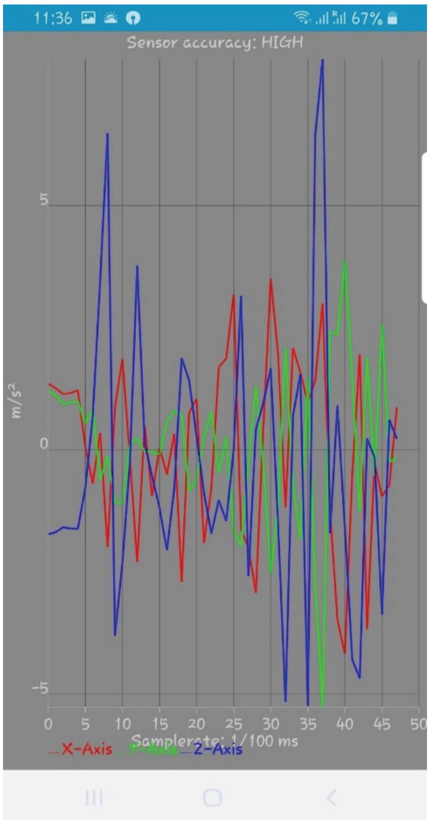

Figure 19 Linear acceleration sensor properties graph when mobile phone is moving

In the Figure 19, linear acceleration does the same measurement as accelerometer sensor, but the major difference is that it does not include the force of gravity. In the graph, Red line represents x-axis direction measures, Green line represents y-axis direction measures and Blue line represents z-axis direction measures of the mobile unit. When the mobile unit is moving around these lines change accordingly.

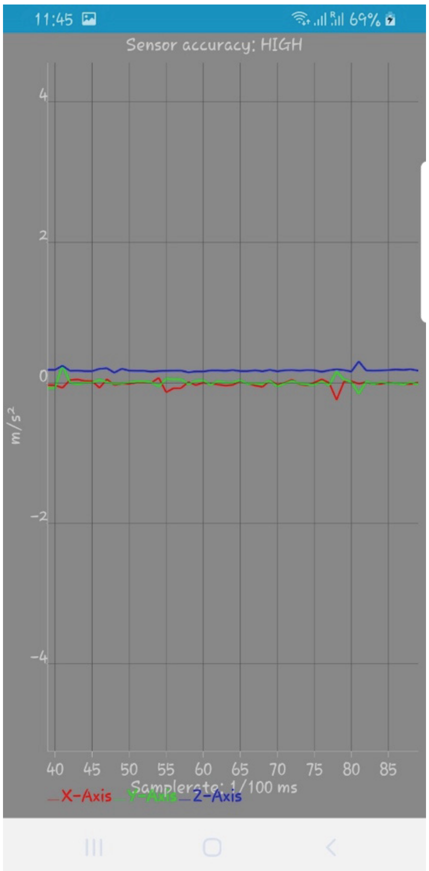

Figure 20 Linear acceleration sensor properties graph when mobile phone is stable

In the Figure 20, measurement is same as Figure 19 but these graphs are obtained when the mobile unit is stationary or stable i.e., it is lying flat on a desk.

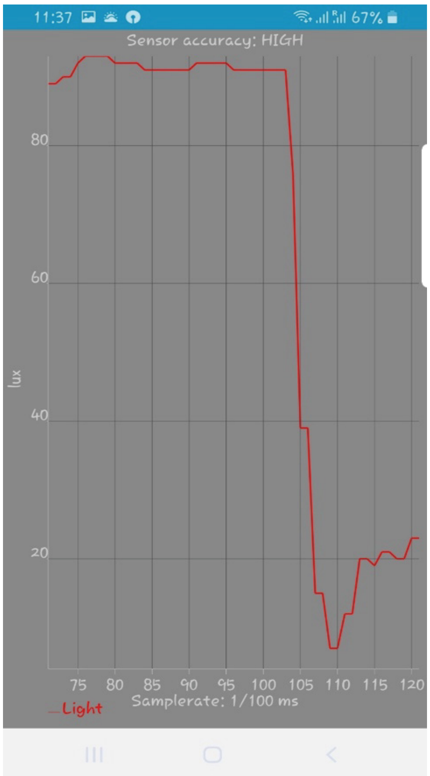

Figure 21 Light sensor properties graph when sensor is covered by hand

In the Figure 21, light sensor detects the ambient light of the mobile unit's surroundings. This data can be used to automatically control the screen brightness. In most cases, the sensor is located on the top of the mobile unit facing the user, but it may differ depending on the manufacturer of the mobile unit. The variation of the graph changes rapidly when the sensor is covered by hand because surrounding light becomes darker and lighter.

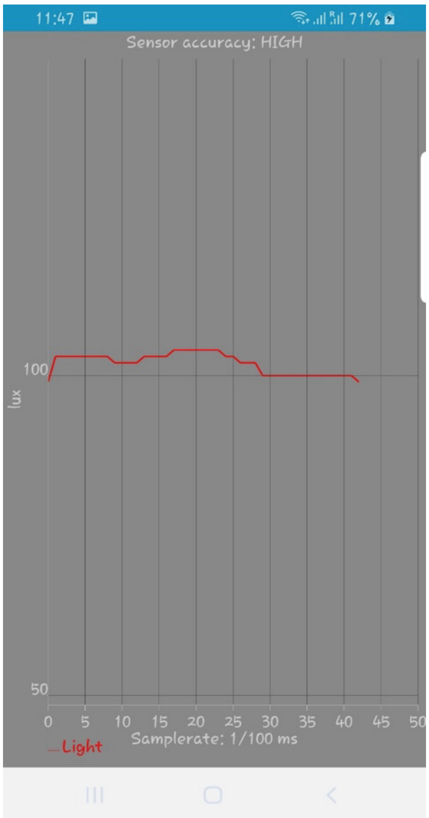

Figure 22 Light sensor properties graph when sensor is not covered by hand

In the Figure 22, the graphical data of light sensor when the light exposed on the sensor is natural ambient light and no hand is used to control the light exposed on the sensor.

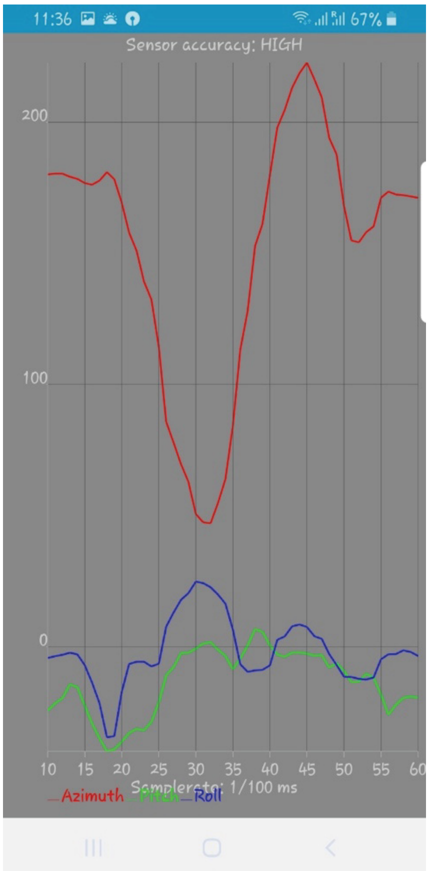

Figure 23 Orientation sensor properties graph when mobile phone is moving

In the Figure 23, orientation sensor measures mobile unit's degree of rotation along the three directional axes x, y and z. It can be used to determine the position of the mobile unit. In the graph, Red line represents Azimuth (angle around x-axis), Green line represents pitch (angle around y-axis) and Blue line represents roll (angle around z-axis) of the mobile unit. The graphical data in the figure shows the data collected during movement of mobile unit in different direction.

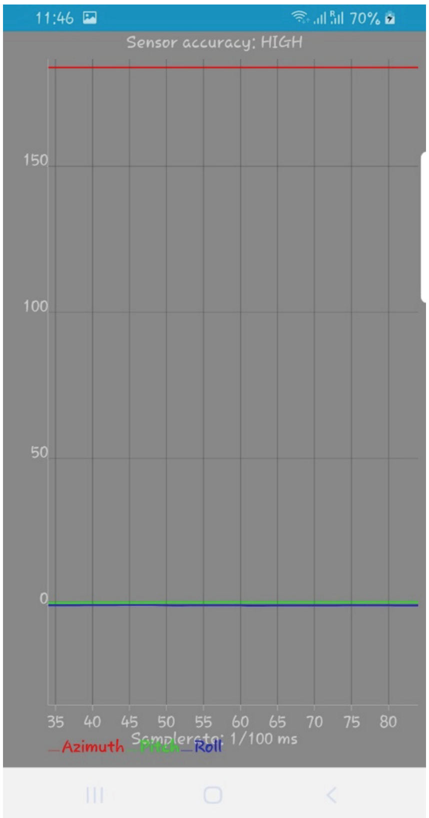

Figure 24 Orientation sensor properties graph when mobile phone is stable

In the Figure 24, the graphical data of orientation sensor obtained when the mobile unit is stationary i.e., on a desk. We can observe the difference of the graph from that of the previous figure.

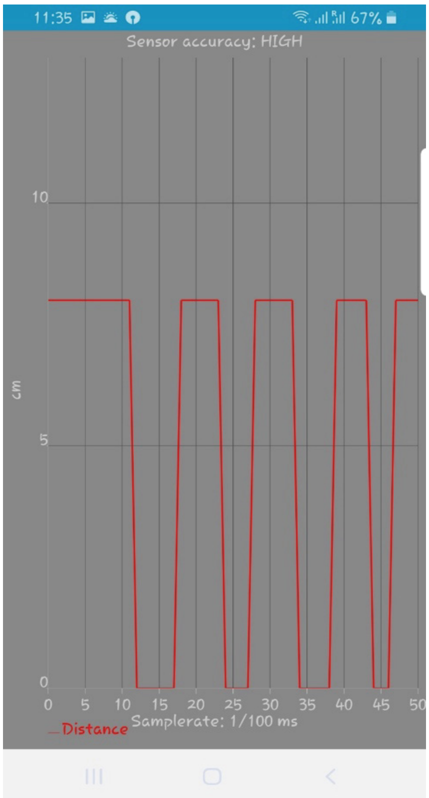

Figure 25 Proximity sensor properties graph

In the Figure 25, the graphical data of the proximity sensor properties is shown. This sensor is used to detect any foreign object that comes near the surface of the mobile unit. When the sensor detects an object in a certain proximity (generally the value is 5cm but it varies from one mobile unit to other) it shows high peak in the graph and if there is no object around the surface, the graph falls down to low peak.

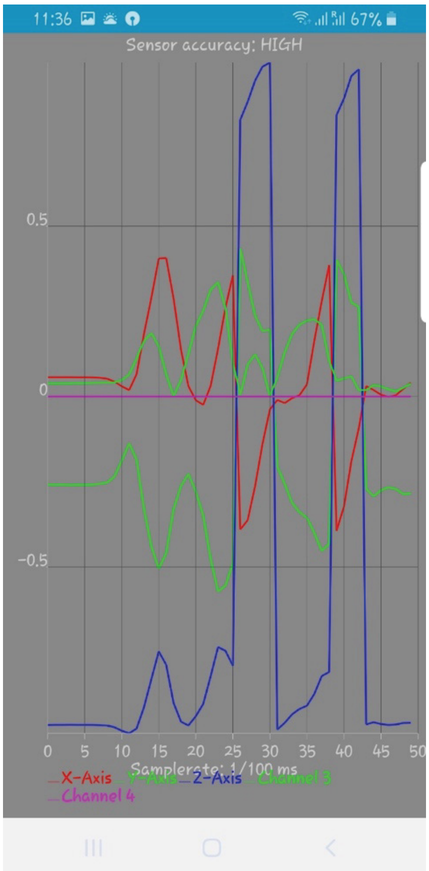

Figure 26 Rotation sensor properties graph

In the Figure 26, the orientation measurement of a mobile unit is shown by providing the three elements of the unit's rotation vector. The graphical data is displayed according to sensor's detection when a mobile unit is rotating.

The figures from Figure 12 to Figure 26 are the screenshots taken from an android mobile phone while testing the sensor detector application. These graphs displays the graphical representation of each sensor properties. These graphs can be viewed when clicking on the specific sensor in the list obtained as per the Figure 11. These help in knowing the performance of the mobile phone.

5.2 Lock Screen Application

This section discuss the results obtained for this application in both the testing cases. The first testing case is testing the application on emulator and the second testing case is testing the application on an android mobile phone.

5.2.1 Testing on Emulator

The following figures are the screenshots taken when this application is tested on emulator.

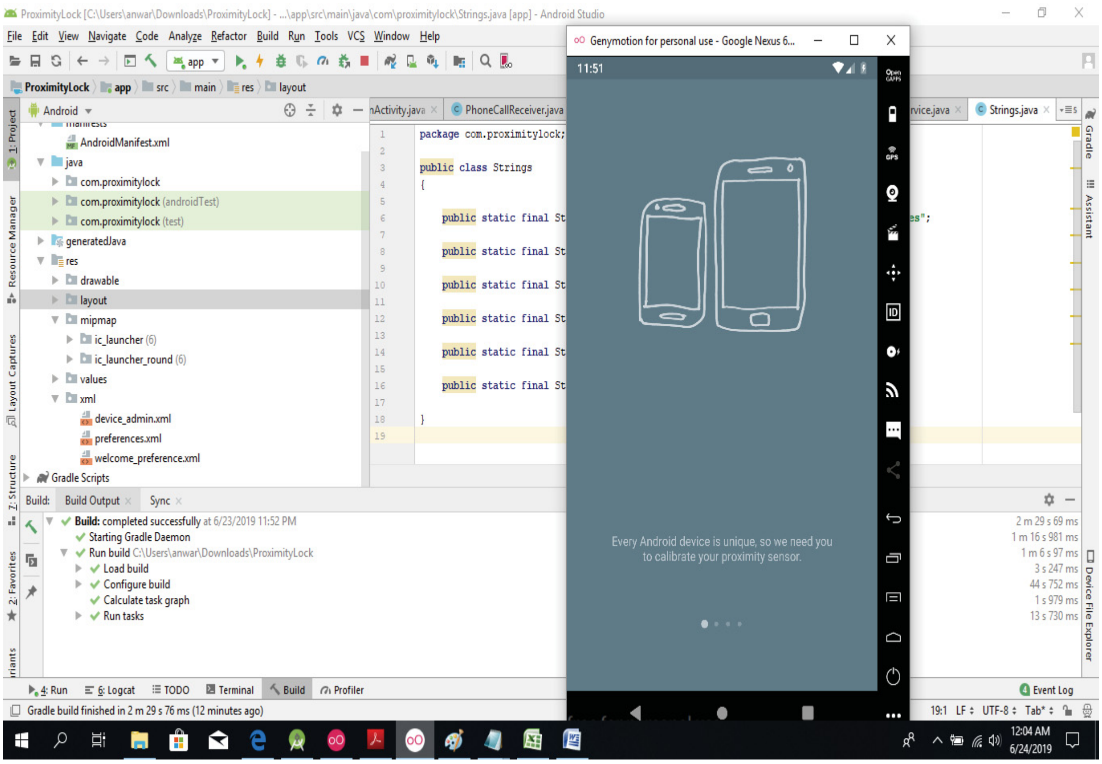

Figure 27 First instructions page of this application

The figures Figure 27 to Figure 30 are the results of instructions pages of the lock screen application. After clicking the calibrate button which is shown in the Figure 30, the user will be able to view the sensor value as shown in Figure 31. This value varies from one mobile device to another mobile device.

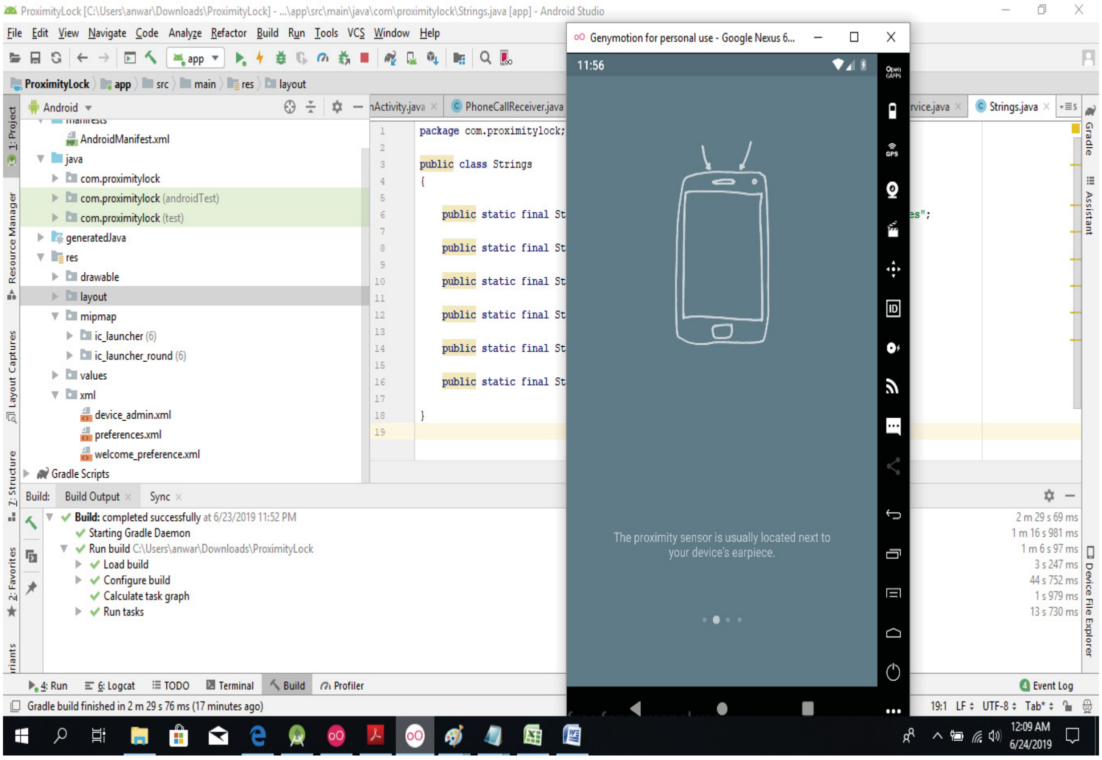

Figure 28 Second instructions page of this application

The Figure 28 is the second page of instructions for using the application which specifies the location of proximity sensor on a mobile device.

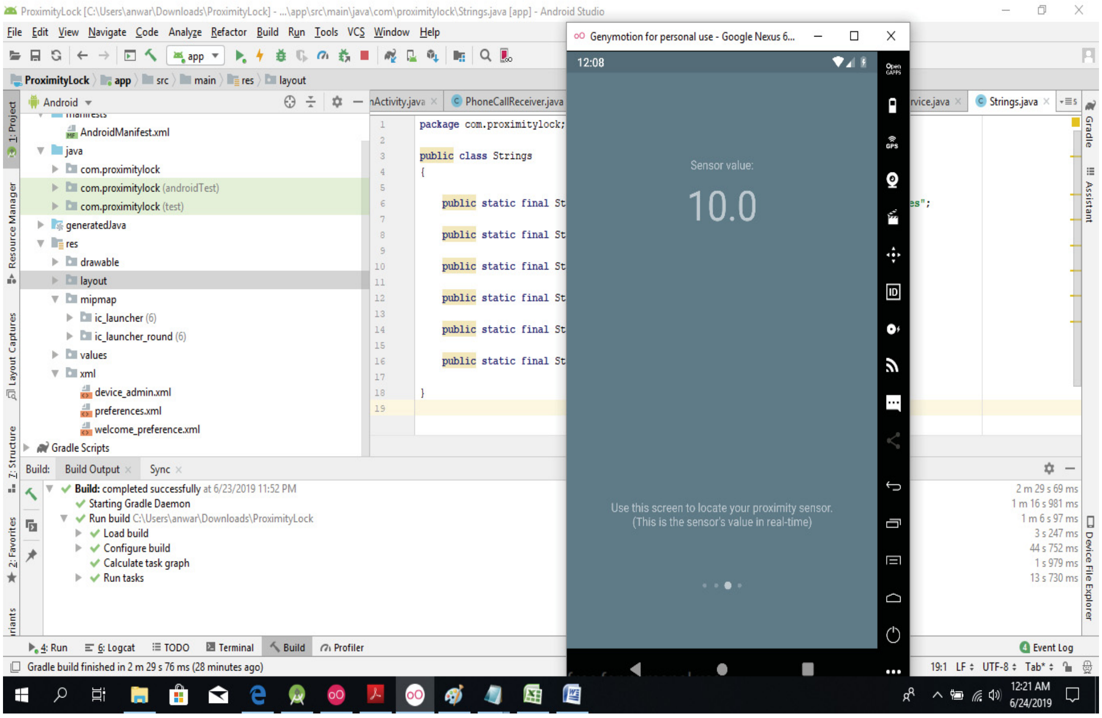

Figure 29 Third instructions page of this application

The Figure 29 is the third page of instructions for using the application which gives the real-time example value of the proximity sensor and using the screen of the mobile unit to detect the sensor.

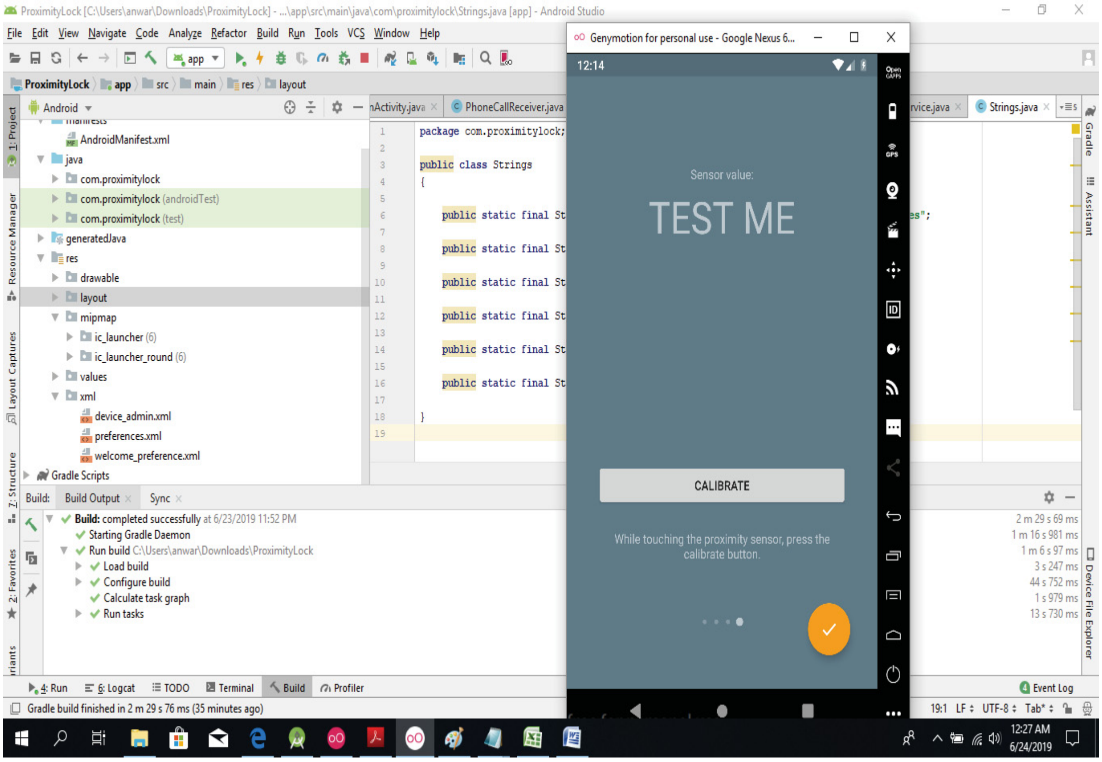

Figure 30 Last instructions page of this application

The Figure 30 is the fourth page instructions for using the application and it is also the page where the sensor calibration is done. In this page, there is a button named calibrate. When the user clicks this button, the app calibrates the sensor and displays the sensor value in real-time.

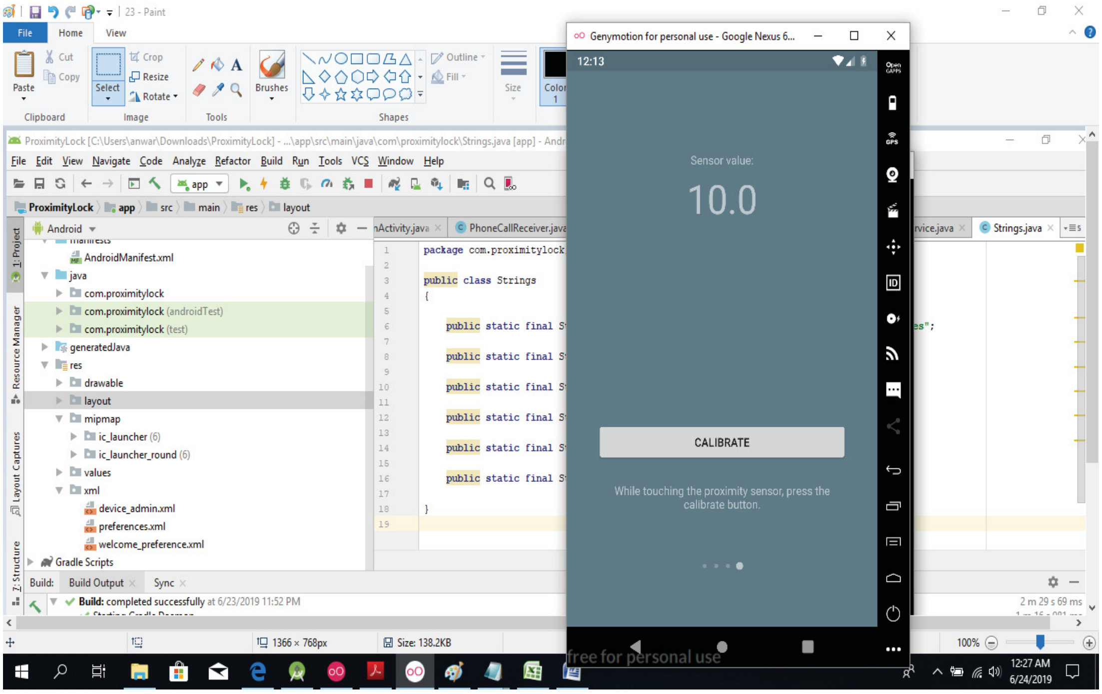

Figure 31 Sensor value after calibration

After calibrating the sensor in the previous page, the user clicks the tick button which brings you to the page in the Figure 31. After clicking on the calibrate button in this page it takes you to the next page where the user needs to give some extensive permissions.

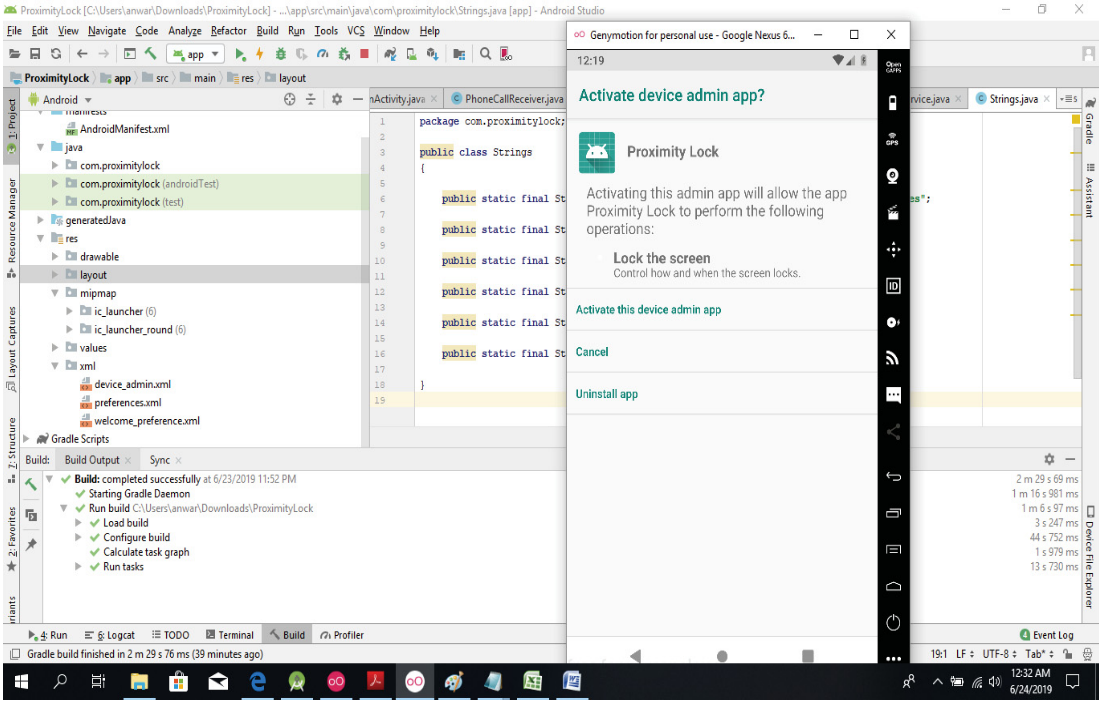

Figure 32 Proximity sensor operation

In the Figure 32, after clicking the 'CALIBRATE' button for the first time, the user is asked to give permission as an admin of mobile phone for activating the app. This permission gives access the app to lock the mobile unit.

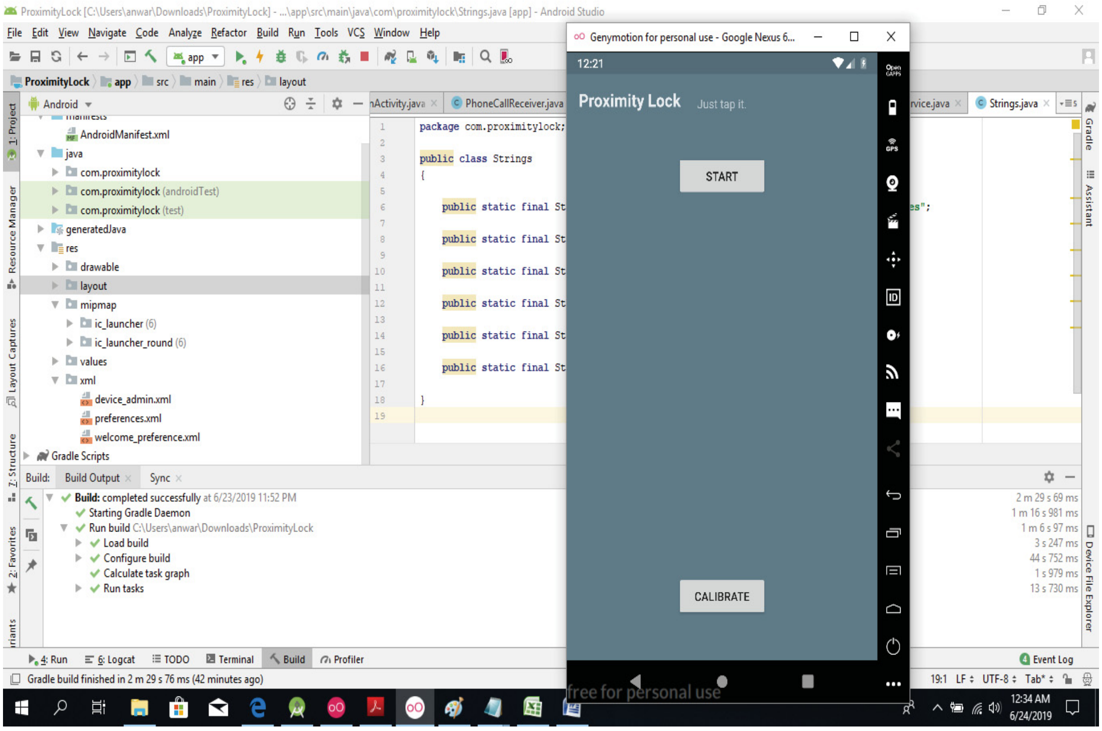

Figure 33 Screen with lock operation

The Figure 33 is the emulator result of the application screen. This screen contains a 'START' button which starts the screen locking application. The application screen with locking process is shown in the following Figure 34.

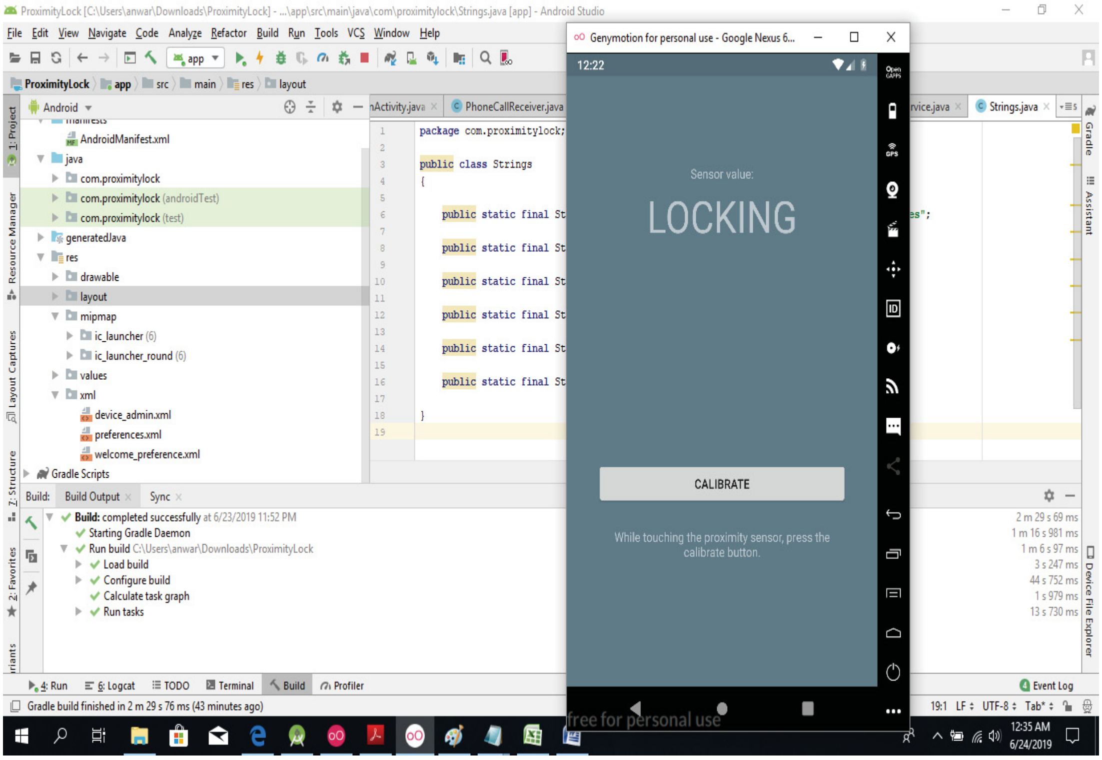

Figure 34 Screen with locking process

If the user wants to check if the app is locking the device correctly, the user can check by hitting calibrate button displayed as in the Figure 34. Here the user can see when the device is locking i.e., if there is any foreign object in the range of the proximity sensor the app is locking mobile device otherwise not.

5.2.2 Testing on Mobile Phone

The following images are the screenshots taken during the testing process of this application on an android mobile phone.

Figure 35 Application home page

The Figure 35 is the initial page displayed to the user after launching the application.

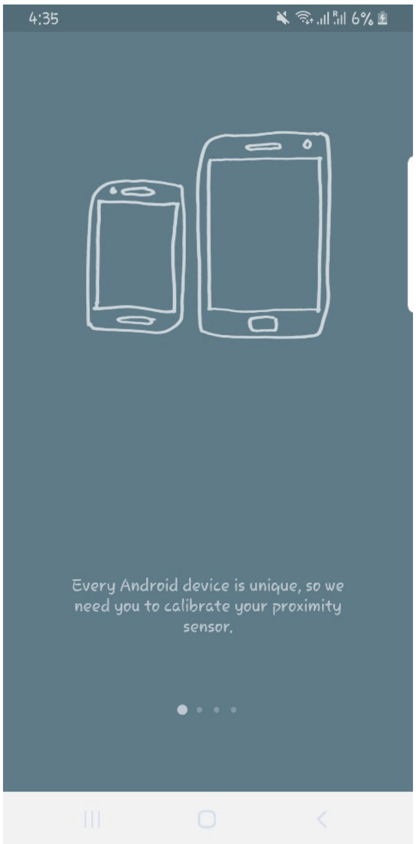

Figure 36 Application instructions first page

The figures from Figure 36 to Figure 38 are the application screens with instructions to use this application. These screens gives a detailed explanation about the usage and operation of designed application. These pages appear only when the application was installed for the first time in mobile unit.

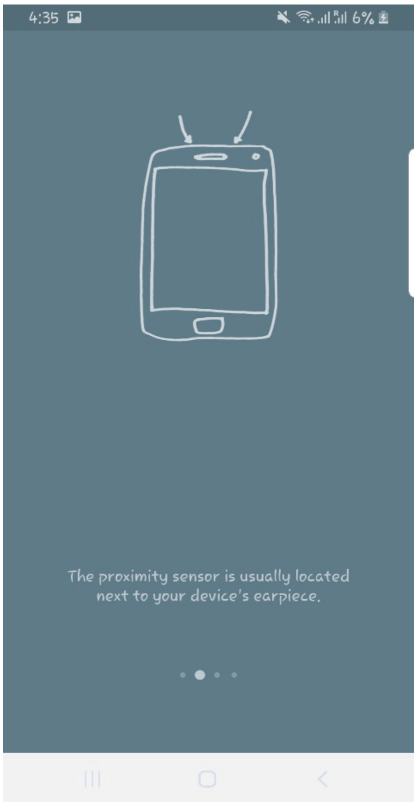

Figure 37 Application instructions second page

Figure 38 Application instructions third page

Figure 39 Calibration of sensor

Figure 40 Application screen where lock screen operation starts

Figure 41 Application screen with locking process

The figures from Figure 39 to Figure 41 are the images showing calibration of sensor operation in this application. These images include the screenshots showing information regarding mobile unit locking process using this app.

6 CONCLUSION AND FUTURE WORK

6.1 Conclusions

For the design and the development of the android application, we developed the code using java and XML files. It is developed on android studio IDE. The applications are successfully developed and implemented in an android phone.

Taking various parameters in to consideration, we tested the applications in two cases. One is with the Genymotion emulator and the other is on an android phone. In both the cases, we are able to achieve the desired outputs and graphs. The considered objectives of thesis are accomplished.

6.2 Future work

In this thesis, during the coding the major drawback we faced is the most lengthy and complicated java files. The most challenging is to maintain and develop a interaction between java and XML files. As these applications would help in knowing the performance status of a mobile phone, it would be interesting to overcome this drawback in future.

Regarding Lock screen application, we even had an idea to design in an another way using different combination of sensors. Due to time factor, we couldn't implement this and it can be developed in future.

7 BIBLIOGRAPHY

[1] Á. López Revuelta, "Orientation estimation and movement recognition using low cost sensors," Master Thesis, Blekinge Institute of Technology, Karlskrona, 2017.

[2] P. Munihanumaiah and H. Sarojadevi, "Design and Development of Network-Based Consumer Applications on Android," in *IEEE International Conference on Computing for Sustainable Global Development*, Bangalore, 2014.

[3] S. Guo-Hong, "Application Development Research Based on Android Platform," in *IEEE 7th International Conference on Intelligent Computation Technology and Automation*, Beijing, 2014.

[4] A. Ahl and K. Andin, "Mobile application for speech quality measurement," Bachelor Thesis, Linköping University, Linköping, 2012.

[5] P. Boberg and A. Lagerström, "Fall detection using smartphone application," Student Thesis, KTH Institute of Technology, Stockholm, 2018.

[6] F. Bernspång, "Smart Parking using Magnetometers and Mobile Applications," Master Thesis, Luleå University of Technology, Luleå, 2011.

[7] A. Singhai, R. S. Ramanujam, J. Bose and V. Kumari, "Implementation and Analysis of Pluggable Android Applications," in *IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, Kerala, 2015.

[8] P. Bhide, "Design and Evaluation of Accelerometer Based Mobile Authentication Techniques," Master Thesis, Linköping University, Linköping, 2017.