UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : Harvent Rau A/L Mogana Ravu

Identification Card No : 961005-10-5665

Title : ANDROID SMART SECURITY SYSTEMS USING INTERNET OF THINGS (IOT) AND FIREBASE

Academic Session : Semester 2 2018/2019

I declare that this thesis is classified as:

□	**CONFIDENTIAL**	(Contains confidential information under the Official Secret Act 1972)
□	**RESTRICTED**	(Contains restricted information as specified by the organization where research was done)*
☑	**OPEN ACCESS**	I agree that my thesis to be published as online open access (Full text)

I acknowledge that Universiti Malaysia Pahang reserve the right as follows:

1. The Thesis is the Property of University Malaysia Pahang.
2. The Library of University Malaysia Pahang has the right to make copies for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Author's Signature)

(Supervisor's Signature)

SYIFAK IZHAR HISHAM

Name of Supervisor

Date: 10/5/2019

Date: 10/5/2019

ANDROID BASED SMART SECURITY SYSTEM USING INTERNET OF THINGS(IoT) AND FIREBASE

HARVENT RAU A/L MOGANA RAVU

Bachelor of Computer Science (Computer Systems & Networking) with Honours

UNIVERSITI MALAYSIA PAHANG

Universiti
Malaysia
PAHANG
Engineering • Technology • Creativity

SUPERVISOR'S DECLARATION

I hereby declare that I have read this technical report and in my opinion this technical report is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Computer Systems & Networking) with Honours.

Signature : ....
Supervisor's Name : Dr. Syifak Binti Izhar Hisham
Date : 10/5/2019

Universiti
Malaysia
PAHANG

STUDENT'S DECLARATION

I hereby declare that this technical report entitles "Android Based Smart Security System using Internet of Things (IoT) and Firebase" is the result of my own revision except as cited in references. The technical report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :..

Name : Harvent Rau A/L Mogana Ravu

ID Number : CA16027

Date : 10/5/2019..

ANDROID BASED SMART SECURITY SYSTEM USING INTERNET OF THINGS (IoT) AND FIREBASE

HARVENT RAU A/L MOGANA RAVU

Thesis Submitted in fulfilment of the requirements for the award of the degree of Bachelor of Computer Science (Computer Systems & Networking) with honours

Faculty of Computer System and Software Engineering

UNIVERSITI MALAYSIA PAHANG

MAY 2019

ACKNOWLEDGEMENT

Firstly, I would like to thank to the God for being with me and helping me throughout this project. I would like to take this opportunity to express my grateful appreciation to all the wonderful peoples who have continuously giving me support, advices and knowledge towards the successful completion of this Final Year Project for my undergraduate course.

I also like to wish my sincere appreciation to my supervisor, Dr Syifak Binti Izhar Hisham for the encouragement, guidance, patience, advices, suggestion and motivation on developing this project.

Last but not least, I also like to thank and appreciate to my friends who had helped me in generating ideas and guide me throughout the whole year. Finally, I would like to express my sincerest thankfulness to my families for their love, support and encouragement that they had given to me to succeed my final year project.

ABSTRAK

Aplikasi ini diberi nama sebagai "Android based Smart Security System using IoT and Firebase" dimana ianya menggunakan konsep anti-pencuri melalui peranti mudah dan senang untuk didapatkan. Aplikasi ini bertindak sebagai penggera kepada pelanggan melalui penghantaran mesej dalam telefon pintar berkenaan pencerobohan. Aplikasi ini berfungsi dengan mudah kerana alatan yang senang didapati dan penyelarasan yang ringkas. Aplikasi ini amat untuk agar menyedarkan para pelanggan tentang kepentingan menjaga keselamatan rumah disamping dapat menjaga diri sendiri, harta benda dan juga ahli keluarga. Terdapat beberapa pernyataan masalah yang berkaitan dengan keperluan pelanggan. Pelanggan kurang kesedaran perlindungan peceroboh kerana ia adalah salah satu jenayah rumah yang paling kerap berlaku pada masa kini sekaligus peranti ini akan membantu pelanggan untuk mengukuh keselamatan untuk diri sendiri. Pelanggan akan menghadapi banyak kerugian harta berharga kerana tidak mempunyai mana-mana peranti keselamatan di dalam rumah/tempat kerja atau tempat lain. Kebanyakan pelanggan adalah daripada golongan orang yang sangat sibuk dengan perbezaan latar-belakang bekerja, mereka tidak mempunyai masa untuk memantau harta benda mereka di rumah kerana waktu sibuk kerja mereka. Objektif 7ystem ini adalah untuk merekabentuk 7ystem pemantauan inovasi menggunakan sensor ESP8266 Nodemcu dan Ultrasonik. Selain itu, 7ystem ini juga merancang 7ystem hantar mesej dalam telefon pintar menggunakan komunikasi API, Bahasa python, dan Bahasa Node.js. Metodologi yang telah dipilih adalah waterfall modal. Keseluruhan, sistem ini berjaya mengesan pergerakan manusia, menghantar isyarat ke firebase dengan menggunakan modul Wi-Fi (Nodemcu), dari firebase menghantar mesej melalui komunikasi API kepada pengguna android dan akhirnya membangun peranti pemantauan inovasi. Selepas pengguna mendapat pemberitahuan, pengguna boleh mengambil tindakan yang diperlukan.

ABSTRACT

This application is named as “Android based Smart Security System using IoT and Firebase” where it is an anti-burglar concept implementation by using an innovation device with less cost and easy to get. It acts by notifying the users as the device sends push up notifications to the user if any motion detected at the target location. This will aware the user how important is smart security for their life, property and family member. There are several problem statements related to customer requirements. Users are lack of awareness of thief protection as it is one of most frequent crimes nowadays thus this device will assist user to have security for themselves. Users will face many losses of his/her valuable properties due to not implementing any security device inside the home, workplace or other place. Finally, most of the users are very busy people with different working background and they do not have any time to monitor their self-belonging due to its busy working hours. The objectives for this system are to design an innovation monitoring system using ESP8266 Nodemcu and Ultrasonic sensor. Moreover, this system is also designing a push up notification system using an API communication, python language and Node.js language. The methodology that has been chosen is waterfall model. As a result, this system is successfully detect the human movement, send a signal to the firebase using Wi-Fi module (Nodemcu), from firebase send a push up notification via API communication to user android and lastly develop an innovation monitoring device. After user gets the notification, user can take the necessary actions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In this modern science century, there have million kinds of technologies running around the world. Apart of that technology, there were few systems monitor people either for their safety, attendance or physical monitoring. There are also has many different types of security systems that have been developed to monitor people activities. Besides of that, these systems are demanding more accuracy, speed and low cost. Especially, the sensors in the system are demanded more performance and lower cost. As an example, motion detection, automation control, remote access system, alarm system and so on.

According to the HR Grapevine Magazine (Garbutt, 2018.), they mentioned about how a company in Western Australia is planning to use drones to monitor employee activity. Not only at Western Australia but many other countries think differently to monitor people activities. Even at Malaysia, they use CCTV service to monitor people activities. In marketing there have many intelligent components and devices. Such as automatic detection, face detection, voice recognition and so many. This project is focusing on developing a new security system with innovative idea which working on the above-mentioned purpose, in the title of “Android based Smart Security System Using Internet of Things (IoT) and Firebase”.

In Malaysia, many security companies develop various kinds of security system. Some examples of security companies in Malaysia are ADT Company, Saxco company, Bluguard Company, Belco Company, Saferity Company and many more. All these companies are expert and have many years’ experience in security field. To control the unwanted activities by people, they invented many kinds of device or system to monitor them. The examples of security system invented by the security companies in Malaysia are alarm system with automation control, security control, remote access from anywhere and anytime, live and recorded video monitoring, wireless system monitoring, CCTV monitoring, motion sensor and so on.

Based on above mentioned, this project does not only monitor human detection at home, but it also monitoring all over the place such as office, school, hospital and other places where can monitor human movement. The specialty of this project is people are not able to realize that they are being monitored by a system. Usually, if there is any CCTV service, user will not involve any offense activity that can make them in trouble. It is good if people are not involving any offense or unwanted activities but, in some cases, we need the exact suspicious person who are involved. For example, if any person wants to pass any restricted area, this project will detect the movement and send an alert message to user without known by the user.

This project called HRSmartSec which is considered as a smart security system that can monitor and detect human movement without their knowledge. It consists of a software and a hardware. This system using few electronic components to design the hardware which can be communicate each other to inform to the user if there is any motion detected. ESP8266 nodemcu and Ultrasonic sensor are the electronic components that use to design the hardware. The hardware is easy to carry, easy to hang and can be kept anywhere user like. The appearance of the hardware is designed in innovative way and it uses limited resource of electric. The mini requirement to activate the device is user need to have internet connection to connect the device to receive notification.

To be more realistic, in Figure 1.1 there have a sample picture of a house layout and the suitable locations to keep the device to detect the human movement. The location of the device in Figure 1.1 are pointed by arrows. This project does not only work for housing area, but it can also function in any place that need to monitor human movement. However, this device cannot be placed on public area. It is suitable to work at restricted areas or any places where people should not enter to the target location. This is just an example of the layout for this project. User can keep the HRSmartSec anywhere in their place either hanging on wall, keep on table or anywhere. The physical appearance of the device is a modern light. The reason of the appearance is to avoid being identified by trespasser that a device is monitoring them.

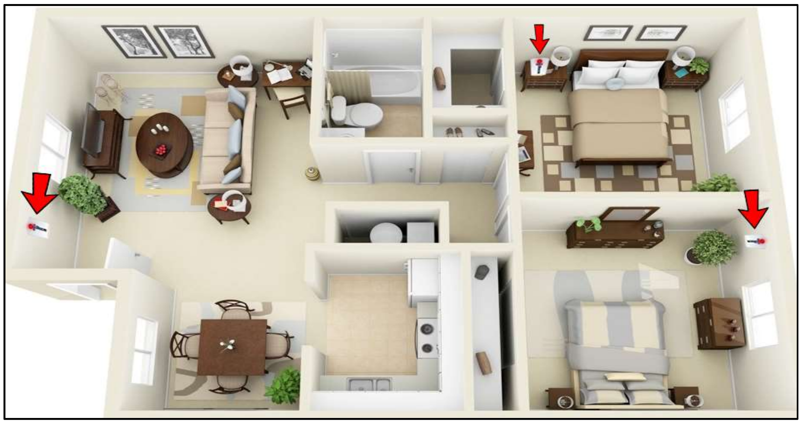


Figure 1.1: The sample house layout and position of the device in the house

1.2 PROBLEM STATEMENTS

There are several problems that arise when security of property is release. The first problem is users are lack of awareness of thief protection as it is one of most favourable home crimes nowadays. Thus, this device will assist as security for themselves and inform to user if any motion detected. So, this will increase the level of secure for them and may reduce the number of cases for intruders.

User will face many losses of his/her valuable properties due to not implementing any security device inside the home/workplace or other place. This is because there is no security protection that worked well when user not around the place. Moreover, the price of the security device is also one of the reasons that user cannot fulfil the security system.

Most of the clients are very busy people with difference working background, they do not have any time to monitor their self-belonging due to its busy working hours. So, by having this device, user can get notification alerts from devices and some evidence about the intruders.

To overcome all the issues, this project proposes HRSmartSec, an Android based smart security system using IoT and Firebase which will be designed as an innovation monitoring security system with low cost and low maintenance. It will be more secure to monitor human movement without their knowledge and easy to set up it.

1.3 OBJECTIVES

I. To study a smart way to monitor house security using IoT device that can detect human movement and send alert messages to user when motion detected.

II. To design and develop a low-cost innovative monitoring device using ESP8266 Nodemcu and Ultrasonic Sensor.

III. To evaluate the monitoring device functions and usability.

1.4 SCOPE

I. Civilians

For civilians, they can use HRSmartSec at their home to monitor their house when they were not around there. If any motion detected, the device will send an alert message to the users through push up notification.

II. Teacher or Lecture

For teacher or lecture, they can use HRSmartSec in their room to monitor their room from any students or staffs entering illegally without their knowledge.

III. Police or Security

For police or security, they can use HRSmartSec at any restricted area to monitor people who enter illegally.

IV. Office Administrator

For office administrator, they can use HRSmartSec at office room to monitor employee who enter illegally in office room.

1.5 THESIS ORGANIZATION

This thesis consists of five chapters.

Chapter 1 discusses about the Android based smart security system using IoT and Firebase. This chapter also explains about the reason on why we need to develop this security system by discovering the problem statement.

Chapter 2 discusses about the literature reviews of Android based smart security system using IoT and Firebase. This chapter also discusses about the comparison between the existing systems and HRSmartSec by showing the advantages and disadvantages of the existing systems.

Chapter 3 discusses about the usage of methodlogy in the project. This chapter showing the usage of the Ultrasonic sensor, servo motor, ESP8266 wi-fi module, 9V battery, jumper wire, led light, switch, regulator and zener diode.

Chapter 4 discusses about the implementation and result discussion of the system. This chapter explains about the development of monitor device.

Chapter 5 discusses about the conclusion. This chapter also discusses about limitations and the future of the motion monitor device if available.

1.6 CONCLUSION

This chapter has discussed about the introduction of the project, problem statements, objectives of the project which will be the main aim to develop this system, scope of the project that who can use this project and thesis organization of the project for overview of all the chapter. The next chapter is going to discuss about the literature reviews.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this literature review of the project are includes the theory, concept, perspective and the method of the project that is used to solve the problem occurs and any. hypothesis that are related to the research of methodology.

This chapter shows the function of existing systems, components, mechanism transmission data and comparison of the mechanism transmission data.

2.2 EXISTING SYSTEMS

There are a few existing systems that have been used, which are the Smart Security System using Raspberry Pi, Smart Security System using GSM Shield, Smart Security System using Bluetooth Module and Smart Security System using 1Sheeld.

2.2.1 Smart Security System Using Raspberry Pi

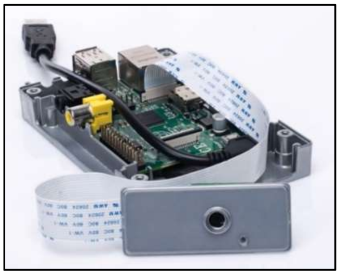


Figure 2.1: Motion Activated Security Camera Using Raspberry Pi

Figure 2.1 shows motion activated security camera system using Raspberry Pi module which is designed by Rada Zakaria (Rada, 2006.). In this system, the programming that use to executes the Raspberry Pi is a Python program that will begin starts when the Raspberry Pi is booted and wait for the motion to be detected by the PIR sensor. When PIR sensor detects any motion, it will transfer signal to the raspberry pi GPIO pin. Raspberry Pi read the signal and send the output signal temporarily. Next, the Raspberry Pi start to record video or take a photo and send that through notification on user's smartphone through an application via Internet. This System requires 650 mA and 5v power supplies, to capture and store image or video in Raspberry Pi SD card (Rada, 2006.).

2.2.2 Smart Security System Using GSM Shield

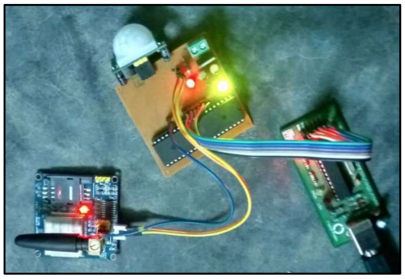
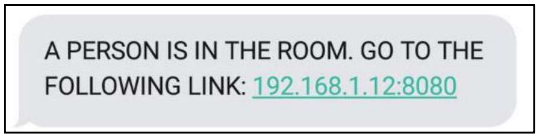


Figure 2.2: Implementation of GSM Based Security System

Figure 2.2 shows a Hardware of Implementation of GSM Based Security System with IOT Applications which is designed (Kumar & Shyam Akashe, 2017.). In this system, they used PIR sensor to track the human movement. When the PIR sensor detects any human movement, it will transfer a signal to the ATMega16 microcontroller and then it will send an alert notification to the user through GSM module. GSM module is an electronic that help to send messages or make a call using a SIM card from a network source. The GSM device can work with any GSM network operator SIM card like a mobile phone with its own unique mobile number. As an output, user will receive an alert message from their mobile as shown in Figure 2.3 (Kumar & Shyam Akashe, 2017.).



A PERSON IS IN THE ROOM. GO TO THE FOLLOWING LINK: 192.168.1.12:8080

Figure 2.3: Received Message from GSM module to User mobile

2.2.3 Smart Security System using Bluetooth Module

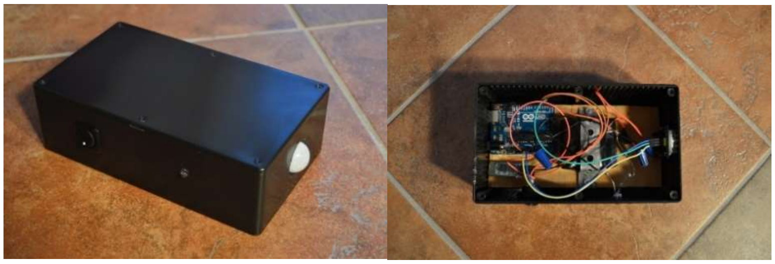


Figure 2.4: Motion Sensor Android Bluetooth Motion Sensor Using Arduino

Figure 2.4 shows a hardware that can monitor human detection using PIR sensor who designed by Shane Jansen, a software developer. This is an infrared motion sensor connected to an Arduino microcontroller. When the PIR sensor detect any motion, the microcontroller will communicate to the running Android application via Bluetooth to play a sound and display an alert notification. After the notification is dismissed, the microcontroller is ready to detect the next motion. The alarm can be disabled for 1 minute from the application and if the connection, a different sound will play (Jansen, 2017.).

2.2.4 Smart Security System using 1Sheeld

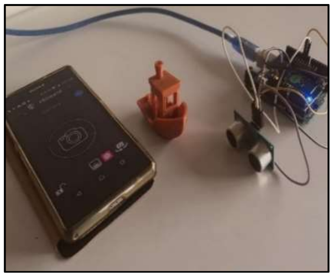


Figure 2.5: Arduino Bluetooth Camera using 1Sheeld

Based on Figure 2.5 shows a hardware that can snap a picture when a motion detected from mobile phone using 1Sheeld module. According to the 1Sheeld official website says that Arduino Bluetooth Camera is aimed to use the Camera shield with an ultrasonic to detect any human movement when enter illegally into our house. It will snap a photo of the human automatically once the human gets into the area of the ultrasonic wave. Later, the pictures will be uploaded into the twitter or Gmail and delivered an alert message as mentioned that something happens, so that user can check their account. 1Sheeld have its own application which design with some parameters like Accelerometer, Colour detector, Email, Face Detection, Notification, SMS, Skype and many. With a microcontroller and a Bluetooth module 1Sheeld can transmit data between Arduino and user's smartphone (1Sheeld, 2018.).

2.2.5 Smart Security Using Arduino

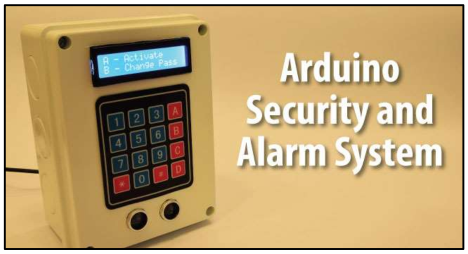


Figure 2.6: Arduino Security and Alarm System Project

Figure 2.6 shows a hardware that can detect object and for the alert they included alarm system. This project is designed by Dejan who run an official page name by How to Mechatronics. Once user press the 'A' button, the alarm will be activated within 10 seconds. They used ultrasonic sensor to detect the motion. If the sensor detects any motion, it automatically starts to alarm sound. To stop the alarming sound, user need to enter the password. User are able to change their password as they needed. To change the password, user need to press 'B' button and then they need to enter the current password to continue to enter new 4-digit password. Once we had changed the new password, next time to deactivate the alarming sound, user need to enter the new password to stop the alarming sound. If any user or anyone enter an invalid password, the system will display an alert message which shows an invalid password so that the user can re-enter the correct password again. (Dejan Nedelkovski, 2017.).

2.3 Components

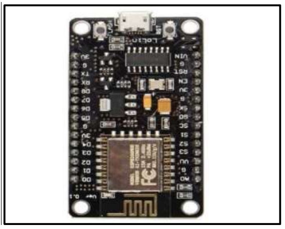


Figure 2.7: ESP8266 Nodemcu

ESP8266 Nodemcu is an open-source electronics platform or board to use hardware and software. Nodemcu are able to read many inputs after programming such as read the light sensor, a finger sensor on a button, or even get read message like twitter and email. In addition, it also able to read the output after programming such as it can control a motor to spin speed, can control the LED light by turning on or off, or even can publish a message by itself in online. The ESP8266 Wi-Fi Module is a self-contained of system on chip with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. It provides capabilities for 2.4 GHz Wi-Fi. The ESP8266 is have strength of either host an application or offload all Wi-Fi networking functions from another application processor (SparkFun, 2018.).

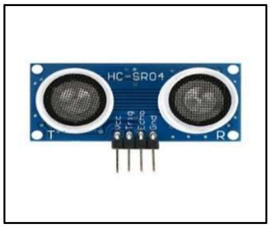


Figure 2.8: Ultrasonic sensor

Ultrasonic sensor will measure the distance to an object by measuring the time taken by the ultrasonic sound waves to reflect from that object. The frequency of the sound is somewhere in the range of ultrasound, this ensures more concentrated direction of the sound wave because sound at higher frequency dissipates less in the environment. They have two types of ultrasonic. One is produces sound and the another will catch the reflected echo. Second is read the arrival of sound impulse and stops the timer. From the timer it is possible to calculate the distance travelled by the sound. The distance to the object is half of the distance travelled by the sound wave (Heikopikner, 2013.)

2.4 Mechanism Transmit Data

Data transmission is a process of transferring data from one digital devices to another in analog or digital format. It either transfer one to one or one to many digital devices. Normally, data transmission is work in a devices or components within the devices to communicate each other to transfer data. There have two methods used to transmit data between digital devices which are serial transmission and parallel transmission. Serial data transmission sends data bits one after another over a single channel. Parallel data transmission sends multiple data bits at the same time to the multiple channels(Mellon, 2017).

There are also has different kind of device or module to transmit data from one device to others. All this device is made for communicate each other to transfer the data. It might be serial transmission or parallel transmission. In this project, ESP8266 Nodemcu Wi-Fi module used as data transmission from the device (HRSmartSec) to the user android. Example of transmit device are Bluetooth module, GSM Module, Wi-Fi Module and NRF24L01 transceiver module. Table 2.1 shows the comparison of the transmission module.

2.4.1 Comparison of Transmission Module

Table 2.1: Comparison of the transmission module

Type of Module & Specification	**Wi-Fi Module (Chosen)**	**Bluetooth Module**	**GSM Module**	**NRF24L01 Module**
Cost (RM) (Cytron, 2007)	13.50	15.90	60.00	5.00
Secure Connection	Medium	Low	high	Low
Data Rate	54Mbits/s	3mbps	9.6kbps	2Mbps
Distance Range	400m	10-100m	900MHz	2.4Ghz
Power Consumption	High	Low	Average	Low
User Connection	Multi user connection	Two-way user connection	Multi user connection	Multi user connection

Based on Table 2.1 indicates the comparison of the transmission module. Each module has its own up and down specification. Example, the Wi-Fi module is expensive compare with NRF24L01 module but from power consumption view NRF24L01 module is using low energy than Wi-Fi module. For this project, ESP8266 Wi-Fi module is chosen for the transmit data from hardware to firebase and to user. This is because ESP8266 Wi-Fi module has the ability to transfer the data from device to user using Internet. It is also a multi user connection. The speed of data rate is higher than other transmission module. The range of receive and send data is quite enough for the system.

2.5 CONCLUSION

This chapter has discussed about the different type existing system that related with the project. In this chapter, there have a few electronic components included and their details. Moreover, the comparison of transmission module is also included. This will explain about the specification of the transmission module.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

To ensure the software and hardware that will be designed are able to be developed within the budget and the time constraint. To make a software, we need a good software development methodology. There are various kind of software methodology and each of it has its own advantages and disadvantages. In this chapter, several methods used to develop the application to monitor the prototype, along with the requirements of the system and how it was used on the development of this system.

To be specific, methodology is a system of method that used to overcome a project step by step in an area of study. It is a kind of documentations that covers the procedures, a schematic representation of tools and material to be used. Each methodology is different from the others due to what the system needed and its intricacy. For the Android based Smart Security System using IoT and Firebase, System Development Life Cycle (SDLC) is used to approach this project to successful.

SDLC is a concept where models are used in the management of a project to describe the levels of stage that involve within the system development project starting from the initial feasibility study through the last stage in which is the maintenance of the completed system. In SDLC, there have various methodologies developed to guide the process of involved such as Waterfall Model, Joint Application Development (JAD), Rapid Application Development (RAD) and Spiral Model. Hence, the methodology that chosen to design the system and approach the project to success is Waterfall Model. This chapter also shows the requirement of project, design, implementation, verification and maintenance. Further explanation will be done on this chapter later.

3.2 SOFTWARE DEVELOPMENT METHODOLOGY

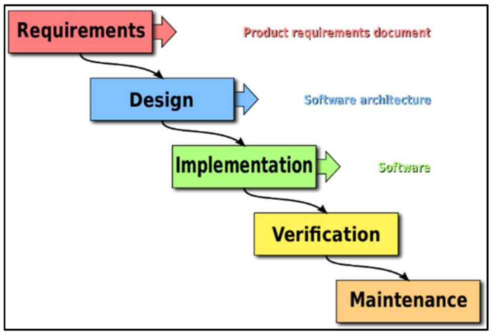


Figure 3.1: Waterfall Model

A. Requirements

During this starting phase, all the possible requirements for the Hardware for this project and the application use for this project are analysis and written down in a detailed document. This document will act as a basis for the future development.

B. Design

In this phase, the requirements analysis will serve as a guide to design the security system and the application. Moreover, the security system and the application design assists in specifying the software and the hardware requirement that are important in defining the whole system infrastructure.

3.2.1 Flowchart

This section shows the flowchart of Android based Smart Security System using IoT and Firebase. Based on Figure 3.2 indicates the flow of the process of how the signal transfer from device to firebase and to user. When the device detects any human movement at target location, a positive signal will transfer from Nodemcu to Firebase via Internet. From Firebase Cloud Message (FCM), user will receive a push up notification via API communication using Python language and Node.js. While, user also receive a pop message “Motion Detected! Do you want me to call an

Emergency Number?" on android app. If user press yes button, user smartphone immediately displays call function to call anyone user want it to check their place.

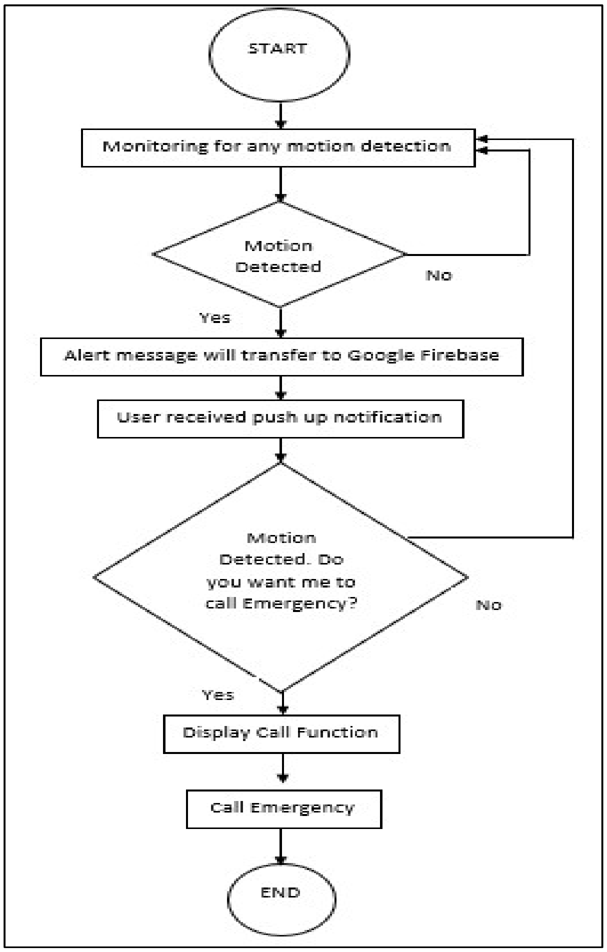

START
Monitoring for any motion detection
Motion Detected
No
Yes
Alert message will transfer to Google Firebase
User received push up notification
Motion Detected. Do you want me to call Emergency?
No
Yes
Display Call Function
Call Emergency
END

Figure 3.2: Flowchart for Android based Smart Security System using IoT and Firebase

3.2.2 Context Diagram

In Figure 3.3 is about the context diagram of Android based Smart Security System using IoT and Firebase. The context diagram contents all the input and the output based on the interaction between the security system, mobile application and the user. The smart security system interacts with itself using Nodemcu by its own process that only involves monitoring motion detection and send a push up notification to user if a human movement detected.

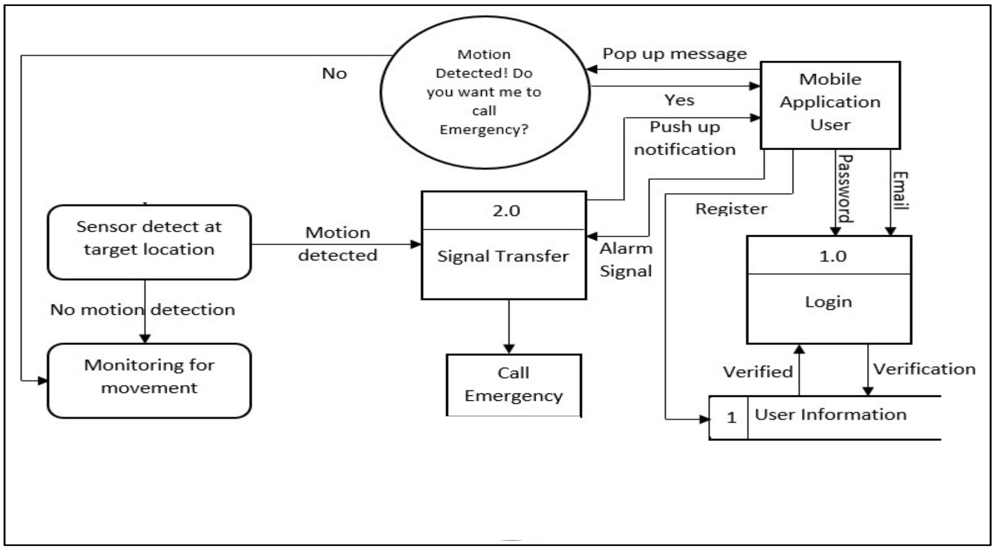


Figure 3.3: Context Diagram design for Android based Smart Security System using IoT and Firebase

3.2.3 Use case Diagram

This section displays the use case diagram for the Android based Smart Security System using IoT and Firebase. The use case is as follows at Figure 3.4 which is the design of the use case diagram that displays the interaction between the system, mobile application user and the unknown user. All other functions and interaction could be more clearly understand by Figure 3.3 as show on top. The differences between each actor could be clearly seen.

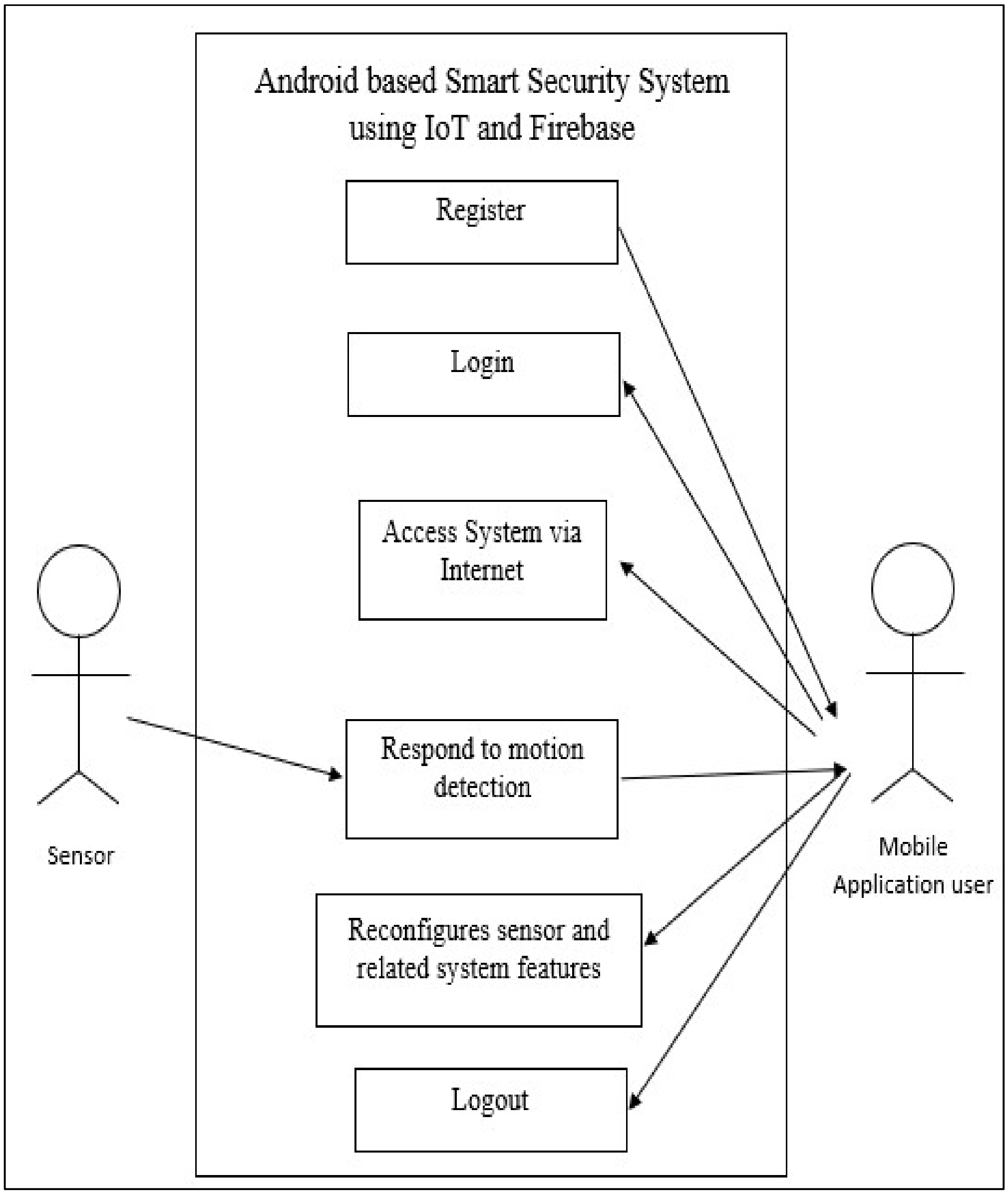

Android based Smart Security System using IoT and Firebase
Register
Login
Access System via Internet
Respond to motion detection
Reconfigures sensor and related system features
Logout
Sensor
Mobile Application user

Figure 3.4: Use case diagram of Android based Smart Security System using IoT and Firebase

Table 3.1: Use Case Diagram

Use Case ID	Android_based_Smart_Security_System_using_IoT_and_Firebase, HRSmartSec
Brief Description	This use case is initiated by the sensor and mobile application user. The system process shows how the human movement motion detected and how the alert notification received by user.
Actor	Sensor and Mobile Application User

Pre-condition	- Mobile application user request to connect the system with Wi-Fi to access the internet. - User must register and login the mobile application to receive push up notification.
Basic Flow	1. The use case starts when user register to the mobile application. If new user means the user should enter user information in the mobile application. Or else user can login by enter their email id and password. 2. If the email id and the password are verified, user's able to access the mobile app or else user should keep try. **[A-1 Verification User Email ID and Password] [E-1 Invalid Email ID and Password]** 3. When any human enters to the target location, the device detects the motion and send the signal to the Google Firebase via Internet. 4. From Google Firebase, the signal transfer to the user as push up notification via API communication. 5. While, users also receive a pop message "Motion Detected! Do you want me to call an Emergency? If user enter "No Button", the system will ignore and continue to monitor human movement. Else if user enter "Yes Button", the app will open to call apps function to call a person who register as Emergency number during registration. **[A-2 Authentication Person]** 6. The use case end.
Alternative Flow	**A-1: Verification User Email ID and Password** 1. User enter their email ID and Password. 2. System will check for email Id and password for the authentication. **A-2: Authentication Person** 1. System will detect the human movement and send the signal to user 2. After user receive the pop message, user can click no button and the system will ignore the person or else the system will start to display call app function to call Emergency number.

Exception Flow	**E-1 Invalid Email ID and Password** 1. User have to login the mobile app by entering their valid email id and password. 2. If the email id and password are not authentication, user are not able to access the mobile application and user will not receive any alert notification.
Post-Condition	- User will receive a push up notification - User can call their neighbour or security to check their place.

3.2.4 Propose User Interface

Figure 3.5: Icon of my application

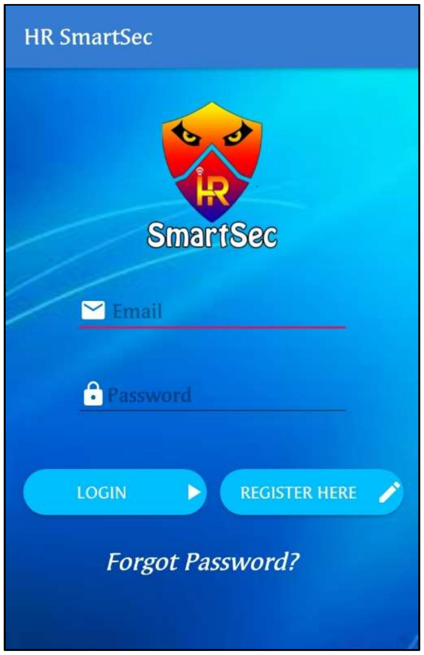


Figure 3.6: Login Interface

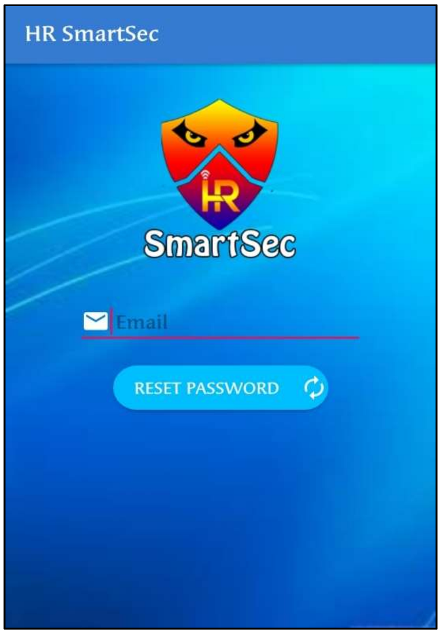


Figure 3.7: Reset Password Interface

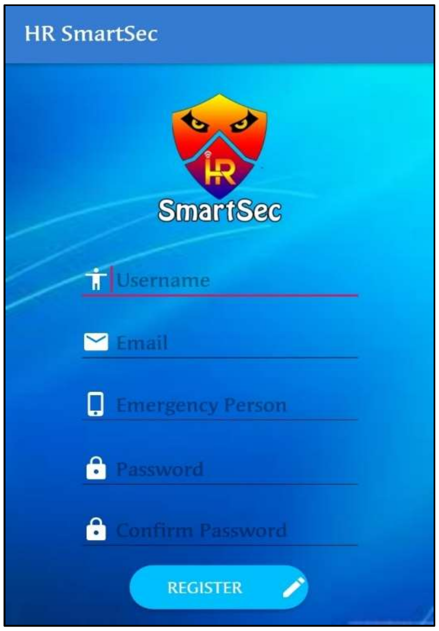
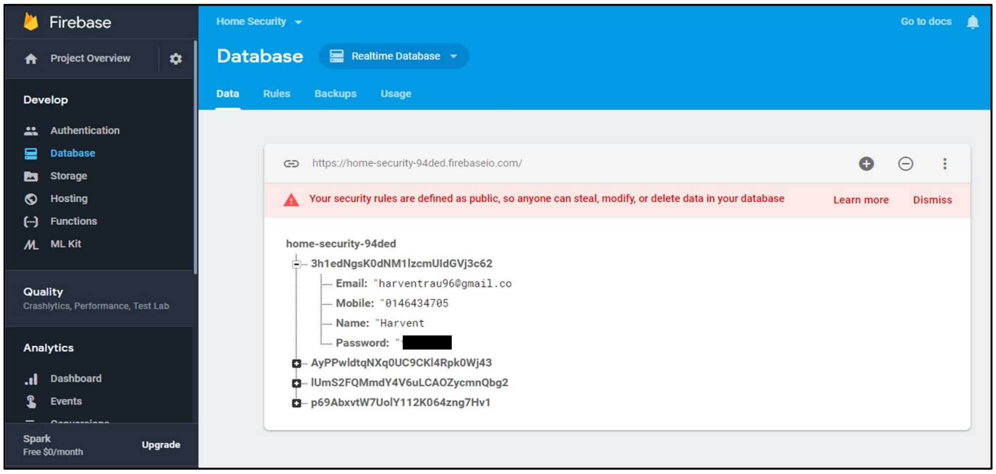


Figure 3.8: Register Interface



https://home-security-94ded.firebaseio.com/

Your security rules are defined as public, so anyone can steal, modify, or delete data in your database

```
home-security-94ded
  3h1edNgsK0dNM1lzcmUldGVj3c62
      Email: "harventrau96@gmail.co
      Mobile: "0146434705
      Name: "Harvent
      Password: "[redacted]
  AyPPwldtqNXq0UC9CKl4Rpk0Wj43
  lUmS2FQMmdY4V6uLCAOZycmnQbg2
  p69AbxvtW7UolY112K064zng7Hv1
```

Figure 3.9: Database Interface (Google Firebase)

Based on Figures show the interface of the application. Figure 3.5 shows the icon of the application which is “HR Smart Security”. Figure 3.6 shows the login interface which is user’s need to login to access the main interface and get push up notification. Figure 3.7 shows the reset password interface which is if user forget his or her password, they can reset their password by entering their email id as they registered. In Figure 3.8 shows register interface which is for new user who first time access this application. Finally, in Figure 3.9 shows the database of this system. All the user’s information that registered by user’s and the motion detection from the device will be store in the google firebase.

3.2.5 Package Module

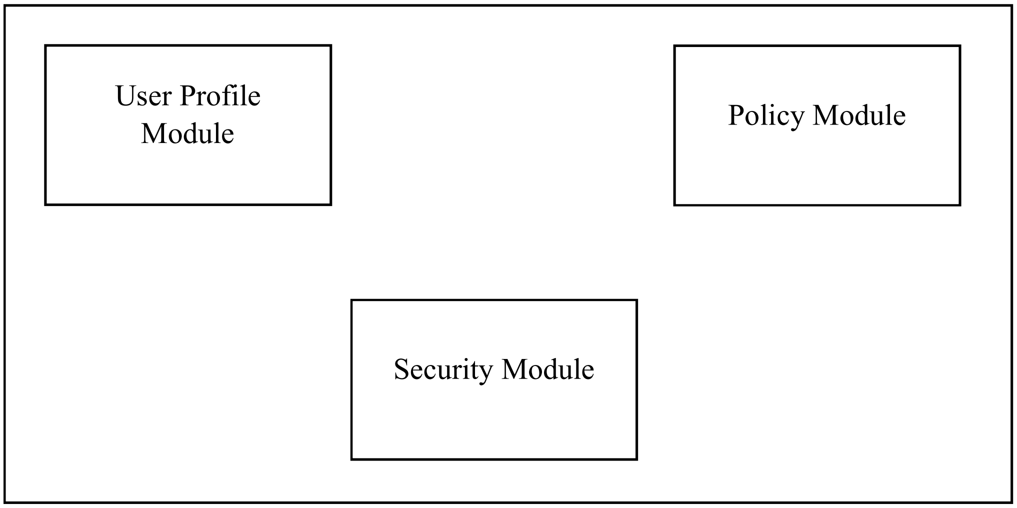


Figure 3.10: Package Module

A. User Profile Module
  - This package consists of all user's information that registered by the user in the register interface.

B. Policy Module
  - This package is used to manage the terms and condition of the prototype

C. Security Module
  - This package consists of security key that will provided to user such as if user forget her or his password, they can reset password by entering their email id, and then they will give a link to reset their password. The link is the security key of the user.

C. Implementation

The system is built to monitor unknown person when user is not around the place. The system is more secure and innovation smart security system. The small program will later be integrated with the next phase meaning that the actual source code will be programmed in this stage. To generate the codes, the programming tools such as compilers, debuggers as well as interpreter are used. Later, implementation phase occurs as all models will be implemented along with business logic and service integrations that had been specified through the previous phases.

D. Verification

After the implementation, need to verify the program to make sure there is no any fault before deliver the project to the client. To do that, need to verify the program. Example, when the system detects the human movement at target location, it should transfer the signal to the user. Make sure user received a proper alert notification.

E. Maintenance

For the last phase of the cycle, if there are any problems that were received via the smart security system, the patches will be released automatically to fix the problems. Hence, the maintenance will be fixed by delivering the changes in the user environment.

3.3 HARDWARE AND SOFTWARE REQUIREMENT

A. Software Requirement

Table 3.2: Software Requirement

SOFTWARE	PURPOSE
Microsoft Words 365	- To edit and prepare my proposal - To design the use case diagram, context diagram, flowchart, data flow diagram and others.
PDF	- To read articles that relate to this project
Google Chrome	- To search and find an information of my project that related to the project
Microsoft Window 10 OS	- To run the software and save all the project in folder
Arduino	- Program the Arduino to control the security system - Program the NRF24L01 for the signal transfer - Monitor the system input and output
Android Studio	- Design the mobile application to control the system and receive alert notification through push up notification
Microsoft Visual Studio	- To receive the mobile app token from firebase - To send alert notification from cloud to android

B. Hardware Requirement

i. ESP8266 Nodemcu

ESP8266 Nodemcu is an open-source electronics platform or board to use hardware and software. Nodemcu can read many inputs after programming such as read the light sensor, a finger sensor on a button, or even get read message like twitter and email. In addition, it also able to read the output after programming such as it can control a motor to spin speed, can control the LED light by turning on or off, or even can publish a message by itself in online. The ESP8266 Wi-Fi Module is a self-contained of system on chip with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. It provides capabilities for 2.4 GHz Wi-Fi. The ESP8266 is have strength of either host an application or offload all Wi-Fi networking functions from another application processor (SparkFun, 2018.).

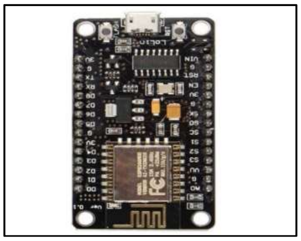


Figure 3.11: ESP8266 Nodemcu

ii. Breadboard

A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. It designed for the testing purpose of a circuit. Most electronic components in electronic circuits can be interconnected by inserting their terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connects the holes on the top of the board (Barragan, n.d.).

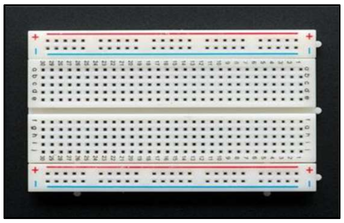


Figure 3.12: Breadboard

iii. Jumpers

Jumper wires is a wire that have a connector pins at both end and allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. In fact, it doesn't get much more basic than jumper wires. The design of the jumper wires is in variety of colours. There is no any technical meaning for the colours. It is simply designed. Example, the usage of the red jumper wire is technically same as the usage of the black wire. The only things that the colour can be used to identify the types of connections like ground or power (HEMMINGS, n.d.).

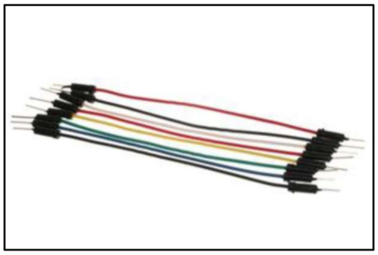


Figure 3.13: Jumper Wires

iv. Soldering Iron

Soldering irons are hand tools like soldering stations but more compact and light-weight. It needs an electrical supply to soldering electronic components. These are available with variable temperature control and thermal balancing. Unlike a soldering station, a soldering iron is lightweight, compact, and portable. Soldering irons supply heat to melt solder. This provides an electrically conductive connection (SpecialistsSign, n.d.).

Figure 3.14: Soldering Iron

V. Ultrasonic Sensor

Ultrasonic sensor will measure the distance to an object by measuring the time taken by the ultrasonic sound waves to reflect from that object. The frequency of the sound is somewhere in the range of ultrasound, this ensures more concentrated direction of the sound wave because sound at higher frequency dissipates less in the environment. They have two types of ultrasonic. One is produces sound and the another will catch the reflected echo. Second is read the arrival of sound impulse and stops the timer. From the timer it is possible to calculate the distance travelled by the sound. The distance to the object is half of the distance travelled by the sound wave (Heikopikner, 2013).

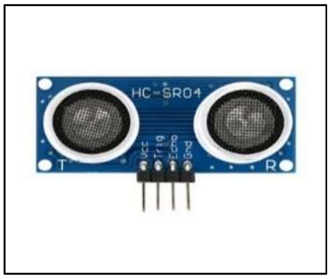


Figure 3.15: Ultrasonic Sensor

VI. Zener Diode

A Zener diode is a silicon semiconductor device that permits current to flow in either a forward or reverse direction. The diode consists of a special, heavily doped p-n junction, designed to conduct in the reverse direction when a certain specified voltage is reached. It also used to drop voltage and maintain the voltage between 3v to 7v(Khan, 2015).

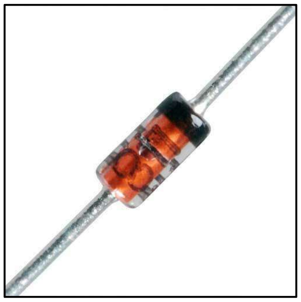


Figure 3.16: Zener Diode

VII. 9V Battery

The nine-volt battery, or 9-volt battery, is a common size of battery that was introduced for the early transistor radios. It has a rectangular prism shape with rounded edges and a polarized snap connector at the top. This type is commonly used in walkie-talkies, clocks and smoke detectors.

Figure 3.17: 9-Volt Battery

3.4 GANTT CHART

Task Name	Duartion	Start	Finish
Planning and Analysis	**91**		
Thinking To Do What	1	05/09/2018	05/09/2018
Come up with a Security System Idea	1	06/09/2018	06/09/2018
Collect all the details about the system	6	07/09/2018	12/09/2018
Collect the current issues to related my project	6	07/09/2018	12/09/2018
Start to do Chapter 1	14	13/09/2018	26/09/2018
Start to do Chapter 2	7	04/10/2018	10/10/2018
Reviewing the existing system	3	06/10/2018	08/10/2018
Submission Chapter 1 and 2	1	12/10/2018	12/10/2018
Start to do Chapter 3	22	18/10/2018	08/11/2019
Choose suitable Methodology	1	19/10/2018	19/10/2018
Use case diagram, context diagram, DFD	2	26/10/2018	27/10/2018
Hardware and Software requirement	2	03/11/2018	04/11/2018
Submission Chapter 3	1	16/11/2018	16/11/2018
Correction of chapter 1,2,3	14	19/11/2018	02/12/2018
Submission final report	1	04/12/2018	04/12/2018
Submission turnitin report and Logbook	1	04/12/2018	04/12/2018
Submission Presentation Approval Formal	1	04/12/2018	04/12/2018
Preparation for Presentation	6	06/12/2018	11/12/2018
Presentation	1	12/12/2018	12/12/2018
Design	**14**		
Design the mobile app	7	07/01/2019	13/01/2019
Design the system	7	14/01/2019	20/01/2019
Implementation	**51**		
Software Installation	1	25/01/2019	25/01/2019
Software and Database Implementation	50	07/01/2019	24/02/2019
Testing	**10**		
Unit testing and integration testing	10	01/04/2019	10/04/2019

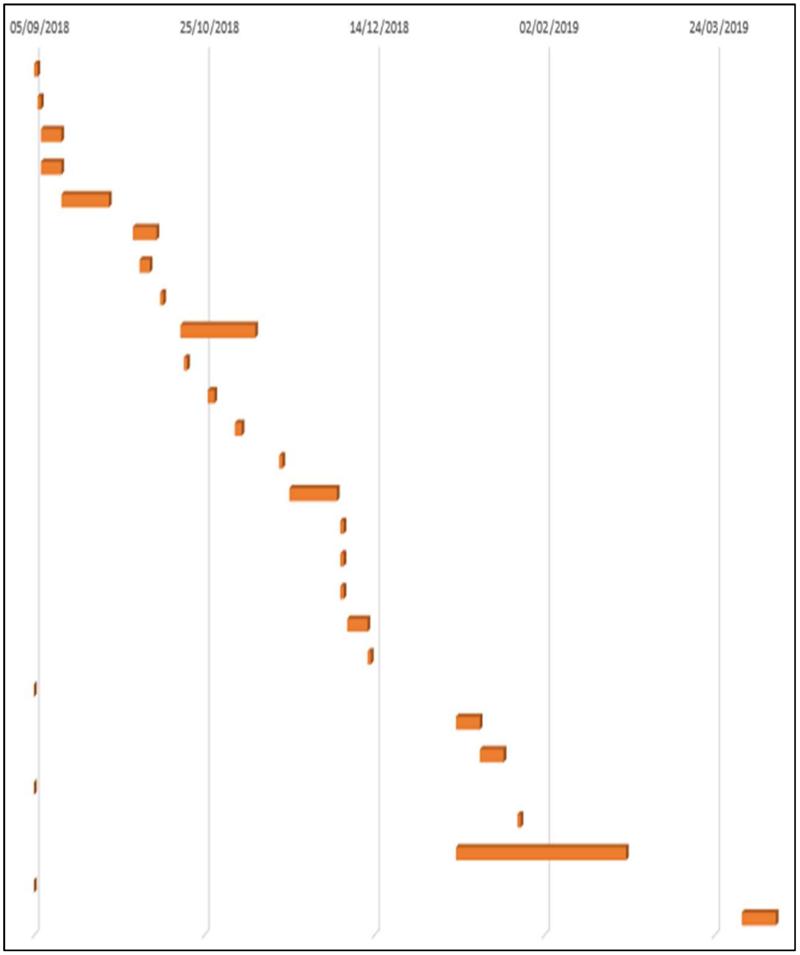


Figure 3.18: Gantt Chart of Table Form

3.5 CONCLUSION

Chapter 3 has discussed about the methodology that we used to overcome this project. The methodology that choose for this project is waterfall model. Each phase of the waterfall model is explained clearly in this chapter. Next, we had designed the context diagram, use case diagram, package module and general architecture of the project. This diagram will show how the process are work in step by step. Moreover, in this chapter has discussed the hardware and software that we used to complete this thesis. Lastly, this chapter are also explained and included the testing part for the system.

CHAPTER 4

IMPLEMENTATION, TESTING AND RESULT DISCUSSION

4.1 INTRODUCTION

The purpose of this chapter is to discuss the implementation of the Android based Smart Security System using IoT and Firebase. The process and data gathering for research and proposes are will discuss on this chapter. Moreover, the sketching of the workflow and the model that using a special tool such as Microsoft Visual Code, Arduino IDE and Android Studio will be discuss on this chapter. The coding apply will be justified and reasoning for each module of the Android based Smart Security System. Testing method will be stated to be used in testing process for Smart Security System and result discussion will be analysed. All the method of getting result will be mentioned.

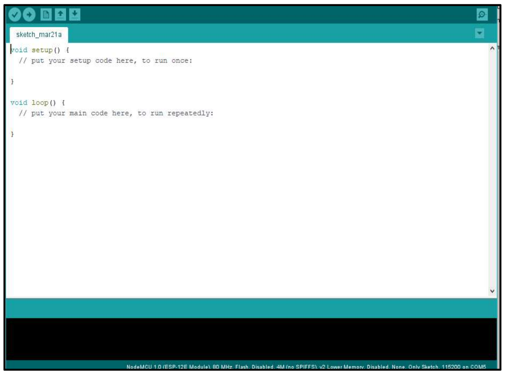
Android based Smart Security System using IoT and Firebase is developed using a few language programming in different tools such as C programming in Arduino IDE, Java Programming in Android Studio and Python programming in Microsoft Visual Code. Additionally, this system used Google Firebase as a cloud storage to store or receive all the data from the device and android.

4.2 IMPLEMENTATION ON DATA/MODEL/PROCESS/HARDWARE

The main purpose of this stage is to document all process that involved in the development of the system. Basically, this chapter is explaining about the Android based Smart Security System using IoT and Firebase development that have been discussed from previous stages. The content in implementation is depending on the project that has been developed.

4.2.1 Arduino IDE platform (Software)

The Arduino Integrated Development Environment or the Arduino IDE is one of open source software that used to interact with all type of Arduino board. The open-source Arduino IDE makes it easy to write code and sketch and upload it to the selected user board. It runs on Window, Mac OS X and Linux. The environment is written in Java and based on processing and other open-source software. There are many types of Arduino board but not all Arduino Software can be used. For this project, I choose ESP8266 Nodemcu board which are not from Arduino family, but it is a Wi-Fi module that can use in Arduino IDE to configure the code. For Arduino software, I used a version of Arduino 1.8.9 as it can be used in any platform of Arduino board type. If other than Arduino board type like ESP8266 Nodemcu board, user need to download some library and need to install the library in Arduino IDE for further process. This software will detect the current date and version of the software at the header of the Arduino IDE.

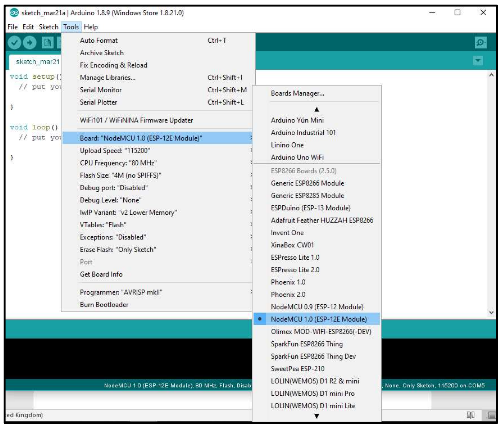


```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Disabled, 4M (no SPIFFS), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM5

Figure 4.1: Arduino IDE interface

To make Arduino IDE connect to the ESP8266 Nodemcu, user must change their Board at tools bar. From tools bar, user must select their respective board. Thus, for this system, NodeMCU 1.0 (ESP-12E Module) is chosen as main board. Then, port must be change based on the IP address for the ESP8266 Nodemcu board. Port can be change at the tools bar. So, for this system, the port number is COM5 and the IP address is 192.168.0.1.



sketch_mar21a | Arduino 1.8.9 (Windows Store 1.8.21.0)

```
void setup()
  // put yo

}

void loop()
  // put yo

}
```

Board: "NodeMCU 1.0 (ESP-12E Module)"
Upload Speed: "115200"
CPU Frequency: "80 MHz"
Flash Size: "4M (no SPIFFS)"
Debug port: "Disabled"
Debug Level: "None"
lwIP Variant: "v2 Lower Memory"
VTables: "Flash"
Exceptions: "Disabled"
Erase Flash: "Only Sketch"
Programmer: "AVRISP mkII"

NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Disab … , None, Only Sketch, 115200 on COM5

Figure 4.2: Arduino IDE interface selecting board type and port

4.2.2 Continuation of component hardware

As mentioned in hardware scope for Android based Smart Security System using IoT and Firebase from Chapter 1 and Chapter 3 above, the hardware stated will be connecting to each other as a complete structured system. HRSmartSec hardware consists of an ESP8266 Nodemcu board, Ultrasonic sensor, jumper wires, 9v battery, switch, zener diode, regulator and Led lights.

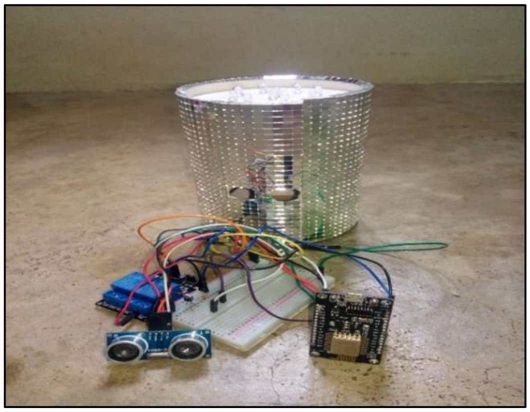


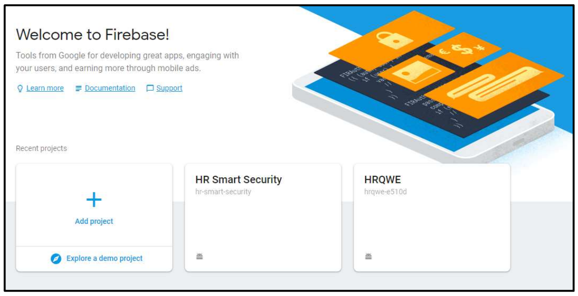
Figure 4.3: Continuation of component hardware

4.2.3 Firebase Registration

Firebase is a mobile and web application development platform or in other words it is a cloud that can store all the data and retrieve the data from user mobile or web. For this project, firebase is playing an important role to store all data from user android. Especially the motion detection data will store in Realtime database and send the data to Firebase Cloud Message (FCM) for the push up notification. To do that, we need to setup the firebase account on Google Firebase. There are several web services that will be used for Android based Smart Security System such as Google Firebase and Gmail.

A) Firebase account registration

The first one is Firebase account. Firebase software provides pre-built process that handle common aspect of application including Authentication, Database, Storage and FCM. Usually, Firebase act as a server or intermediate between two host or software like Arduino IDE (Arduino board) and Android Studio (Mobile Apps). Arduino board cannot communicate or transfer data directly to mobile app. In that case, developer use google firebase as an Intermediate for both software or host to communicate. Figure 4.4 shows the interface of the Google Firebase.



Welcome to Firebase!

Tools from Google for developing great apps, engaging with your users, and earning more through mobile ads.

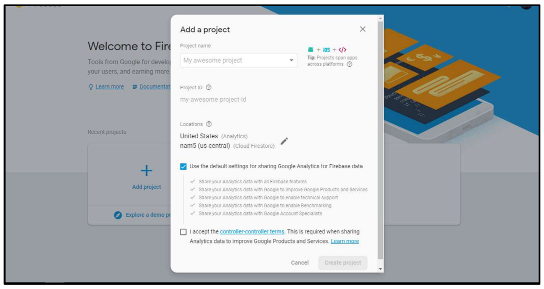
Recent projects

HR Smart Security
hr-smart-security

HRQWE
hrqwe-e510d

Figure 4.4: Google Firebase Home Interface

Figure 4.5 shows the add project interface. After all the registration, developer needs to click the add project on Google Firebase Home interface for further process. Next developer needs to add a name for their project. After add name, developer needs to check the accept box and need to click the create project button. After clicked the create project button, Google Firebase will start to verify and create the project as shown in Figure 4.6.



Add a project

Project name: My awesome project

Project ID: my-awesome-project-id

Locations: United States (Analytics) nam5 (us-central) (Cloud Firestore)

☑ Use the default settings for sharing Google Analytics for Firebase data

- ✓ Share your Analytics data with all Firebase features
- ✓ Share your Analytics data with Google to improve Google Products and Services
- ✓ Share your Analytics data with Google to enable technical support
- ✓ Share your Analytics data with Google to enable Benchmarking
- ✓ Share your Analytics data with Google Account Specialists

□ I accept the controller-controller terms. This is required when sharing Analytics data to improve Google Products and Services. Learn more

Figure 4.5: Add project Interface

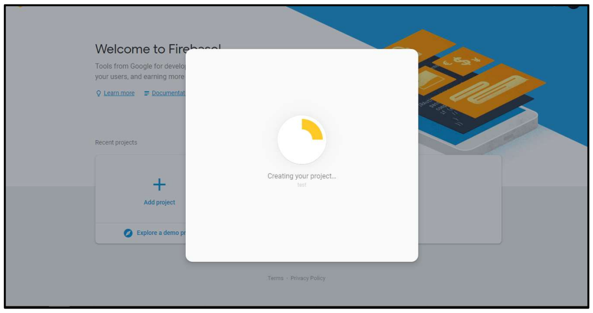
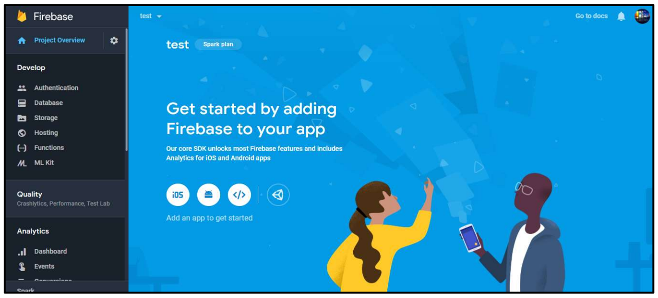


Figure 4.6: Creating Project Interface

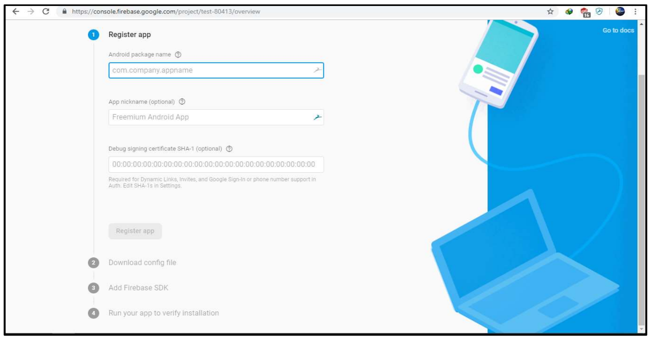
After Google firebase create the project, it shows the main page of the firebase as shown in Figure 4.7. Next, developer need to click the android icon for further process. In Figure 4.8 shows the register app interface which will be show after clicked the android icon. In register app interface, developer needs to add android package and android nickname which can get from the Android Studio as shown in Figure 4.9. Lastly, developer needs to click next button for the end process. Now, developer can use the firebase as Intermediate for both Arduino board and Android App.



Get started by adding Firebase to your app

Our core SDK unlocks most Firebase features and includes Analytics for iOS and Android apps

Figure 4.7: Google Firebase Main Interface



https://console.firebase.google.com/project/test-80413/overview

1 Register app

Android package name

com.company.appname

App nickname (optional)

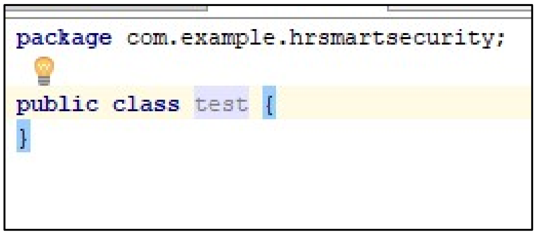
Freemium Android App

Debug signing certificate SHA-1 (optional)

00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

Required for Dynamic Links, Invites, and Google Sign-In or phone number support in Auth. Edit SHA-1s in Settings.

Figure 4.8: Register app Interface

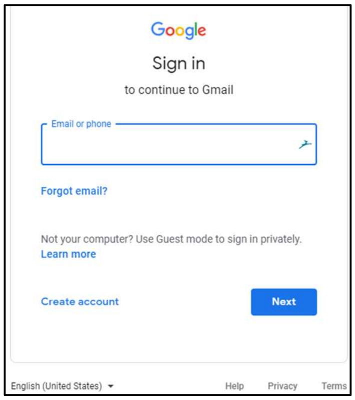


```
package com.example.hrsmartsecurity;

public class test {
}
```

Figure 4.9: Android Studio

B) Google email account registration

As mentioned above, developer should register a Google Email account firstly. This is because Google Firebase requires a google email account before proceed to next step. Thus, this is very important for the development of Android based Smart Security System. After the Gmail registration, developer should click allow button to allow the Gmail to access the firebase.



Google

Sign in

to continue to Gmail

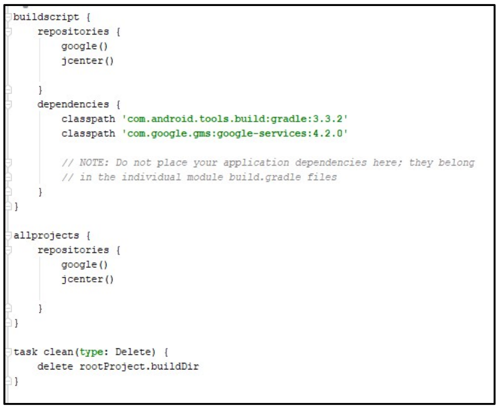
Email or phone

Not your computer? Use Guest mode to sign in privately.

Figure 4.10 Gmail login interface

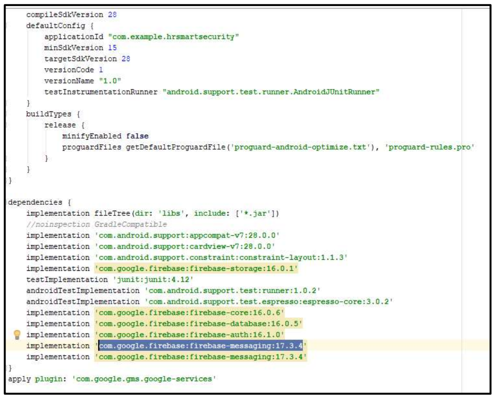
4.2.4 Implementation of Build Gradle in Android Studio

First of all, Android Studio needs to communicate with Google Firebase. To do that, there are several steps. Step 1 is the developer needs to add Firebase configuration file in Android Studio. The configuration file should download during register app as shown in Figure 4.8. After download the ‘google-service.json’ (configuration file), the developer needs to move the file into module (app-level) directory of the app. Next, to enable firebase product in app, the developer should add the ‘google-service plugin’ inside Grade File in Android Studio as shown in Figure 4.11. Additionally, the developer should add Google play service Gradle plugin at the bottom of the line into module (app-level) Gradle file as shown in Figure 4.12.



```
buildscript {
    repositories {
        google()
        jcenter()

    }
    dependencies {
        classpath 'com.android.tools.build:gradle:3.3.2'
        classpath 'com.google.gms:google-services:4.2.0'

        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

allprojects {
    repositories {
        google()
        jcenter()

    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

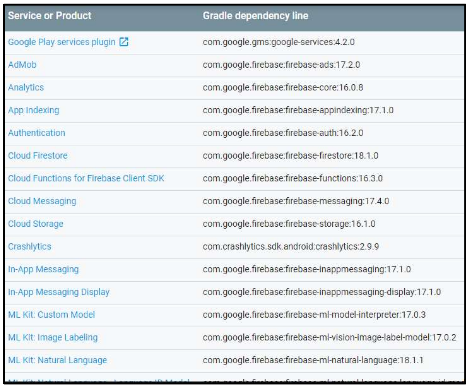
Figure 4.11: Build Gradle of project in Android Studio



```
    compileSdkVersion 28
    defaultConfig {
        applicationId "com.example.hrsmartsecurity"
        minSdkVersion 15
        targetSdkVersion 28
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    //noinspection GradleCompatible
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support:cardview-v7:28.0.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.google.firebase:firebase-storage:16.0.1'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
    implementation 'com.google.firebase:firebase-core:16.0.6'
    implementation 'com.google.firebase:firebase-database:16.0.5'
    implementation 'com.google.firebase:firebase-auth:16.1.0'
    implementation 'com.google.firebase:firebase-messaging:17.3.4'
    implementation 'com.google.firebase:firebase-messaging:17.3.4'
}
apply plugin: 'com.google.gms.google-services'
```

Figure 4.12: Build Gradle of module in Android Studio

The next step is the developer needs to add Firebase SDK to the app. There are many supported firebase products as shown in Figure 4.13. Developer can choose which product there need. In this case, we need authentication, database, analysis and cloud messaging product. Implement all the dependency line in module directory as shown in Figure 4.12. After that, the developer needs to click the sync button to ensure all the dependencies have necessary versions. Lastly, run the app to send verification to firebase that developer successfully integrated Firebase. The device logs will display the Firebase verification that initialization is complete. If developer run his/her app on an emulator that has network access, the Firebase console notifies that the app connection is complete.



Service or Product	Gradle dependency line
Google Play services plugin	com.google.gms:google-services:4.2.0
AdMob	com.google.firebase:firebase-ads:17.2.0
Analytics	com.google.firebase:firebase-core:16.0.8
App Indexing	com.google.firebase:firebase-appindexing:17.1.0
Authentication	com.google.firebase:firebase-auth:16.2.0
Cloud Firestore	com.google.firebase:firebase-firestore:18.1.0
Cloud Functions for Firebase Client SDK	com.google.firebase:firebase-functions:16.3.0
Cloud Messaging	com.google.firebase:firebase-messaging:17.4.0
Cloud Storage	com.google.firebase:firebase-storage:16.1.0
Crashlytics	com.crashlytics.sdk.android:crashlytics:2.9.9
In-App Messaging	com.google.firebase:firebase-inappmessaging:17.1.0
In-App Messaging Display	com.google.firebase:firebase-inappmessaging-display:17.1.0
ML Kit: Custom Model	com.google.firebase:firebase-ml-model-interpreter:17.0.3
ML Kit: Image Labeling	com.google.firebase:firebase-ml-vision-image-label-model:17.0.2
ML Kit: Natural Language	com.google.firebase:firebase-ml-natural-language:18.1.1

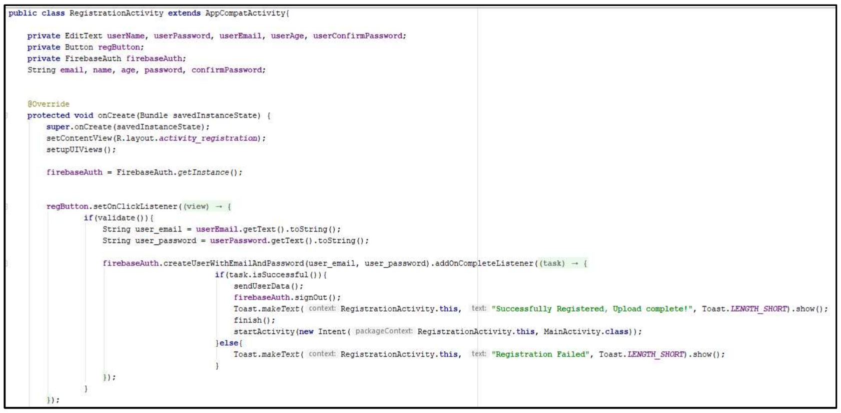
Figure 4.13: Supported Firebase Product

4.2.5 Implementation of Android Studio code

The first coding that implementing in Android Studio is Java code which make an interaction between the Google Firebase and Android App. In Android Studio, developer can design their own interface and the layout. In this system, there are three main layouts had designed which are Login Interface, Registration Interface and Main Page Interface. In Login Interface user can login by entering their email id and password. In Registration Interface user can register their information such as username, emergency person number, email and etc.

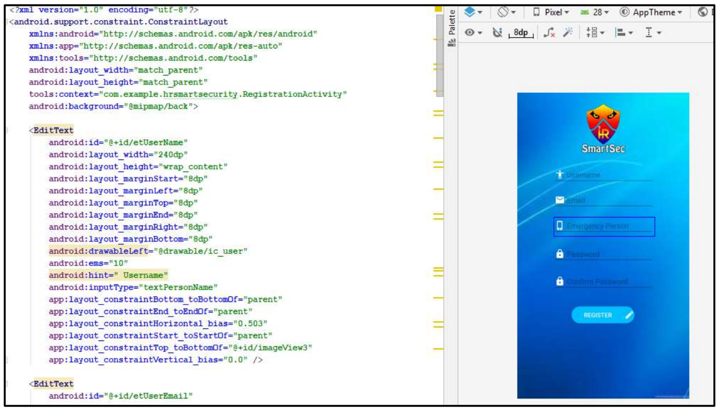
All the information that enter by user in registration interface will be store inside the Google Firebase Realtime Database as shown in Figure 4.16. In Figure 4.14 and 4.15 show the code for the registration interface and the layout respectively.

In Figure 4.17 shows the main page of the app. User can control the device by press the on/off button and can view the battery status and motion status from this interface. Beside of that, this interface is also will show the register username which retrieve the data from the firebase. All the retrieve codes are included in this chapter.



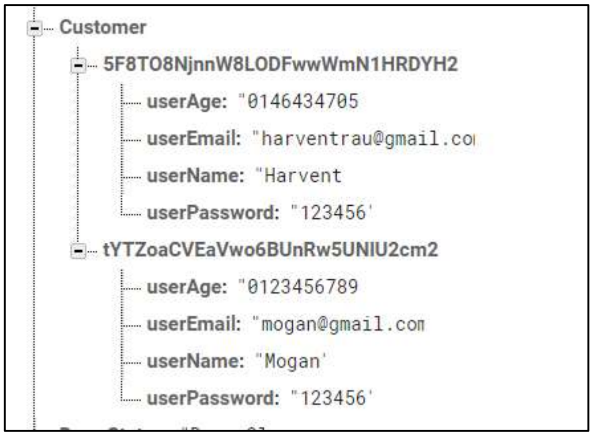
```java
public class RegistrationActivity extends AppCompatActivity{

    private EditText userName, userPassword, userEmail, userAge, userConfirmPassword;
    private Button regButton;
    private FirebaseAuth firebaseAuth;
    String email, name, age, password, confirmPassword;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registration);
        setupUIViews();

        firebaseAuth = FirebaseAuth.getInstance();


        regButton.setOnClickListener((view) → {
                if(validate()){
                    String user_email = userEmail.getText().toString();
                    String user_password = userPassword.getText().toString();

                    firebaseAuth.createUserWithEmailAndPassword(user_email, user_password).addOnCompleteListener((task) → {
                                        if(task.isSuccessful()){
                                            sendUserData();
                                            firebaseAuth.signOut();
                                            Toast.makeText( context: RegistrationActivity.this,  text: "Successfully Registered, Upload complete!", Toast.LENGTH_SHORT).show();
                                            finish();
                                            startActivity(new Intent( packageContext: RegistrationActivity.this, MainActivity.class));
                                        }else{
                                            Toast.makeText( context: RegistrationActivity.this,  text: "Registration Failed", Toast.LENGTH_SHORT).show();
                                        }
                    });
                }
        });
```

Figure 4.14: Registration Interface Code



```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.hrsmartsecurity.RegistrationActivity"
    android:background="@mipmap/back">

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="240dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginBottom="8dp"
        android:drawableLeft="@drawable/ic_user"
        android:ems="10"
        android:hint=" Username"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.503"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/imageView3"
        app:layout_constraintVertical_bias="0.0" />

    <EditText
        android:id="@+id/etUserEmail"
```

Figure 4.15: Registration Layout Code



```
⊟ Customer
   ⊟ 5F8TO8NjnnW8LODFwwWmN1HRDYH2
        userAge: "0146434705
        userEmail: "harventrau@gmail.co
        userName: "Harvent
        userPassword: "123456'
   ⊟ tYTZoaCVEaVwo6BUnRw5UNIU2cm2
        userAge: "0123456789
        userEmail: "mogan@gmail.com
        userName: "Mogan'
        userPassword: "123456'
```

Figure 4.16: Realtime Database

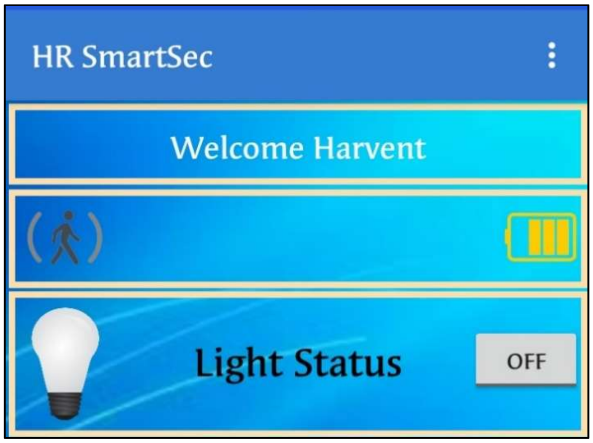
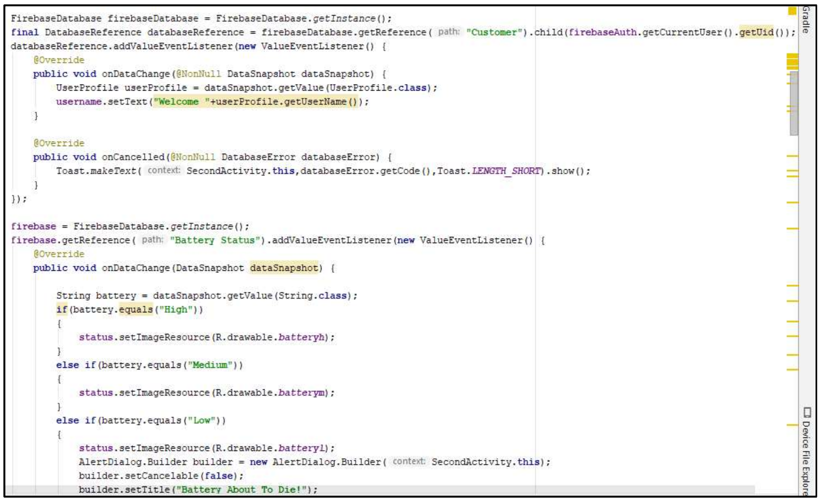
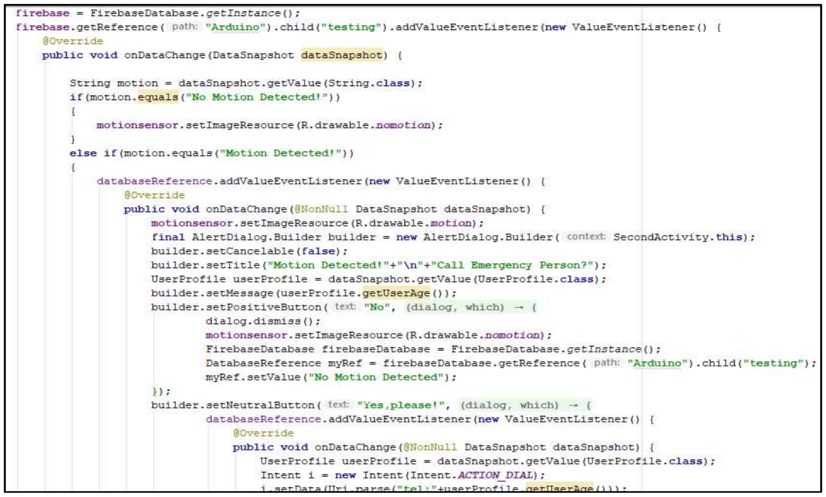


Figure 4.17: The Main Page of the App



```
FirebaseDatabase firebaseDatabase = FirebaseDatabase.getInstance();
final DatabaseReference databaseReference = firebaseDatabase.getReference( path: "Customer").child(firebaseAuth.getCurrentUser().getUid());
databaseReference.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
        UserProfile userProfile = dataSnapshot.getValue(UserProfile.class);
        username.setText("Welcome "+userProfile.getUserName());
    }

    @Override
    public void onCancelled(@NonNull DatabaseError databaseError) {
        Toast.makeText( context: SecondActivity.this,databaseError.getCode(),Toast.LENGTH_SHORT).show();
    }
});

firebase = FirebaseDatabase.getInstance();
firebase.getReference( path: "Battery Status").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {

        String battery = dataSnapshot.getValue(String.class);
        if(battery.equals("High"))
        {
            status.setImageResource(R.drawable.batteryh);
        }
        else if(battery.equals("Medium"))
        {
            status.setImageResource(R.drawable.batterym);
        }
        else if(battery.equals("Low"))
        {
            status.setImageResource(R.drawable.batteryl);
            AlertDialog.Builder builder = new AlertDialog.Builder( context: SecondActivity.this);
            builder.setCancelable(false);
            builder.setTitle("Battery About To Die!");
```

Figure 4.18: Retrieve code for Username and Battery Status

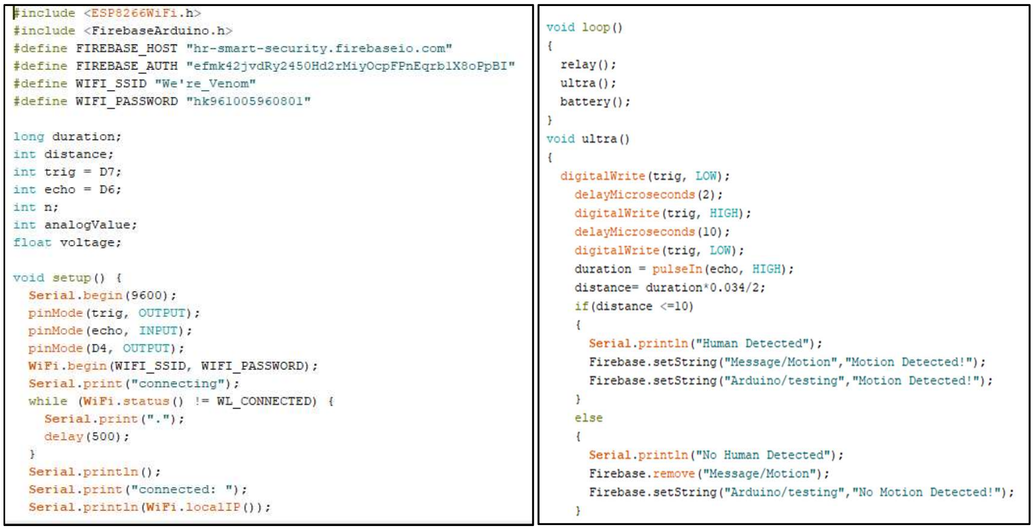


```
firebase = FirebaseDatabase.getInstance();
firebase.getReference( path: "Arduino").child("testing").addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {

        String motion = dataSnapshot.getValue(String.class);
        if(motion.equals("No Motion Detected!"))
        {
            motionsensor.setImageResource(R.drawable.nomotion);
        }
        else if(motion.equals("Motion Detected!"))
        {
            databaseReference.addValueEventListener(new ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                    motionsensor.setImageResource(R.drawable.motion);
                    final AlertDialog.Builder builder = new AlertDialog.Builder( context: SecondActivity.this);
                    builder.setCancelable(false);
                    builder.setTitle("Motion Detected!"+"\n"+"Call Emergency Person?");
                    UserProfile userProfile = dataSnapshot.getValue(UserProfile.class);
                    builder.setMessage(userProfile.getUserAge());
                    builder.setPositiveButton( text: "No", (dialog, which) → {
                            dialog.dismiss();
                            motionsensor.setImageResource(R.drawable.nomotion);
                            FirebaseDatabase firebaseDatabase = FirebaseDatabase.getInstance();
                            DatabaseReference myRef = firebaseDatabase.getReference( path: "Arduino").child("testing");
                            myRef.setValue("No Motion Detected");
                    });
                    builder.setNeutralButton( text: "Yes,please!", (dialog, which) → {
                            databaseReference.addValueEventListener(new ValueEventListener() {
                                @Override
                                public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                                    UserProfile userProfile = dataSnapshot.getValue(UserProfile.class);
                                    Intent i = new Intent(Intent.ACTION_DIAL);
                                    i.setData(Uri.parse("tel:"+userProfile.getUserAge()));
```

Figure 4.19: Retrieve code for Motion Status

4.2.6 Implementation of Arduino Sketch

Arduino sketch functionally used to recall the Android App and Firebase that already created. Basically, it will read the motion and send the motion to the firebase. Moreover, it also read the battery voltage and send the voltage to the firebase. Later, from firebase it will retrieve into the android app so that user can view all the status. Two libraries needed which are ESP8266WiFi.h and FirebaseArduino.h. ESP8266WiFi.h library is to allow the Arduino IDE sends the code into the Nodemcu board. Additionally, there are two more variable which must need for the Nodemcu board which are WIFI_SSID and WIFI_PASSWORD. This both variables are used to connect the Wi-Fi to send the data to firebase via Internet. FirebaseArduino.h library is to allow the Arduino IDE to access the firebase. For this library, there have two variable which are FIREBASE_HOST and FIREBASE_AUTH. This both variables are for the authentication. Based on Figure 4.20 shows the source code that will be implemented into Smart Security System to allow the features to be responded by motion sensor. This code will trigger the ultrasonic sensor to read the motion.



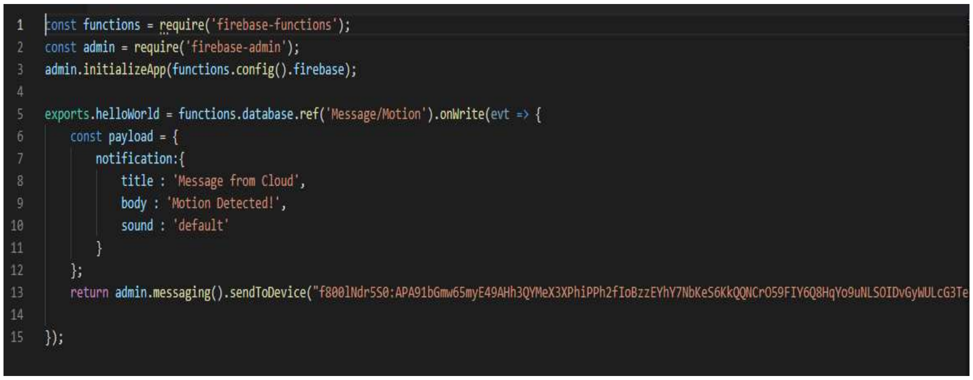
```
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
#define FIREBASE_HOST "hr-smart-security.firebaseio.com"
#define FIREBASE_AUTH "efmk42jvdRy2450Hd2rMiyOcpFPnEqrb1X8oPpBI"
#define WIFI_SSID "We're_Venom"
#define WIFI_PASSWORD "hk961005960801"

long duration;
int distance;
int trig = D7;
int echo = D6;
int n;
int analogValue;
float voltage;

void setup() {
  Serial.begin(9600);
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(D4, OUTPUT);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());
```

```
void loop()
{
  relay();
  ultra();
  battery();
}
void ultra()
{
  digitalWrite(trig, LOW);
    delayMicroseconds(2);
    digitalWrite(trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig, LOW);
    duration = pulseIn(echo, HIGH);
    distance= duration*0.034/2;
    if(distance <=10)
    {
      Serial.println("Human Detected");
      Firebase.setString("Message/Motion","Motion Detected!");
      Firebase.setString("Arduino/testing","Motion Detected!");
    }
    else
    {
      Serial.println("No Human Detected");
      Firebase.remove("Message/Motion");
      Firebase.setString("Arduino/testing","No Motion Detected!");
    }
```

Figure 4.20: Arduino Source Code

4.2.7 Implementation of Visual Code

The code that use in Visual Code is python. This code will look for the specific android app to send the push up notification. There has a unique id for each app which help find the specific android app. By tracking this unique id, python will send the push up notification to the user. If any data the change in database, this code will read the file and send the alert message to the user. Figure 4.21 shows the source code of the visual code.



```
const functions = require('firebase-functions');
const admin = require('firebase-admin');
admin.initializeApp(functions.config().firebase);

exports.helloWorld = functions.database.ref('Message/Motion').onWrite(evt => {
    const payload = {
        notification:{
            title : 'Message from Cloud',
            body : 'Motion Detected!',
            sound : 'default'
        }
    };
    return admin.messaging().sendToDevice("f800lNdr5S0:APA91bGmw65myE49AHh3QYMeX3XPhiPPh2fIoBzzEYhY7NbKeS6KkQQNCrO59FIY6Q8HqYo9uNLSOIDvGyWULcG3Te

});
```

Figure 4.21: Microsoft Visual Source Code

4.3 TESTING PHASE

The main purpose of this stage is to test each of the component from HRSmartSec device to android app before integrated between each other. Since waterfall model methodology is use for the software development, thus integrating test should be conducted.

4.3.1 Integration Testing

Integration testing is associated with the architectural design phase. Integration tests are performed to find out the coexistence as well communication from the internal modules inside the system.

For the integrated testing, each module will be test from the starting point where the device detects the motion until user receive a push up notification in Android. The test may be differ to support certain constraint that will be happen.

Table 4.1: Integration testing result and discussion

Test Data	Expected Result	Actual Result	Pass/Fail	Comment
Ultrasonic sensor to ESP8266 Nodemcu	- Detect for human movement	- Motion Detected - Send the motion detection to firebase	Pass	All work fine
Firebase Cloud Message to Android App	- FCM receive motion detection signal - Android App should receive the push up notification	- FCM received motion detection signal - Android App is never receive the push up notification	Fail	The Token ID is not same as in Firebase and Android App.
Firebase Cloud Message to Android App	- FCM receive motion detection signal - Android App should receive the push up notification	- FCM received motion detection signal - Android App is receive the push up notification	Pass	After configured the token id in visual code, the android app can receive the notification.
Android App (Motion Status)	- Receive push up notification - Motion detection icon will change and pop up alert message	- Receive push up notification - Motion detection icon will change and pop up alert message - Display call function with the registered emergency number	Pass	All work fine

Android App (Light Status)	- Change light icon after press on button - Send the signal to the firebase	- Change light icon after press on button - Send the signal to the firebase	Pass	All work fine
Device Light	- Light on after user press on button on Android app	- Light on after user press on button on Android app	Pass	All work fine
Android App (Battery Status)	- Change battery icon according to the voltage level	- Change battery icon according to the voltage level - If battery status low means android displays a pop message for user	Pass	All work fine

Based on Table 4.1 shows the result and discussion of the system. All part is work well. The ultrasonic sensor can able to read the human movement. After it the motion, it send a positive signal to the ESP8266. ESP8266 receive the signal and send the signal to firebase via Internet. Firebase database receive the data and update on its database. Once firebase update the motion data, it sends another signal to the FCM function. FCM send signal to the python file where it is an API communication to the Android App. Next, Android app receive a push up notification while the icon of the motion changed. Moreover, the android app is also receiving a pop message which mentioned “Motion Alert! Do you want me to call your emergency number?”. If user press no button means, the system will ignore the motion and start monitor again. Instead of that, if user press yes button, the system will display the call function to call your emergency person to check the place. All this process is work well.

4.3.2 Implementation the testing process

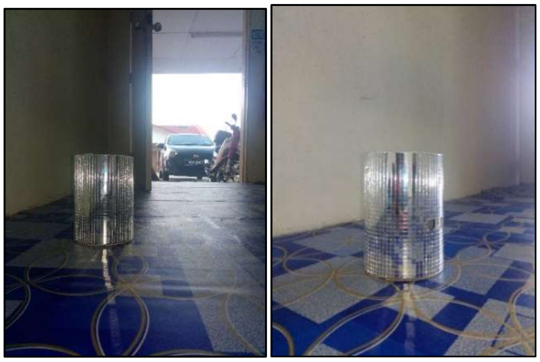
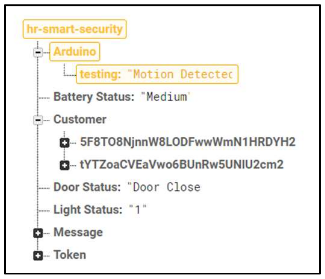


Figure 4.22: The Monitoring Device (HRSmartSec)

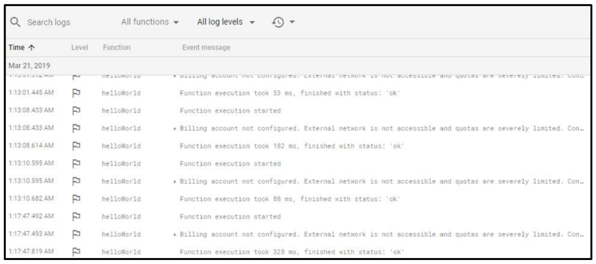
Figure 4.22 shows the monitoring device where it is monitoring for a motion detection. The appearance of the device is wrap with a mirror and on top of the device insert a group of led bulb. It looks like a modern light so that any stranger who illegally enter into a target location, are not able to identify that this device is monitoring them.



```
hr-smart-security
⊟– Arduino
   └testing: "Motion Detectec
 – Battery Status: "Medium"
⊟– Customer
   ⊞– 5F8TO8NjnnW8LODFwwWmN1HRDYH2
   ⊞– tYTZoaCVEaVwo6BUnRw5UNlU2cm2
 – Door Status: "Door Close
 – Light Status: "1"
⊞– Message
⊞– Token
```

Figure 4.23: Motion Detected in Firebase

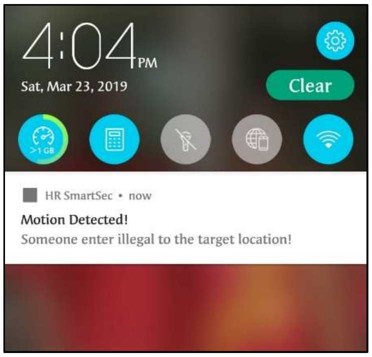
Figure 4.23 shows the signal that receive from the ESP8266 when the device detects a motion on target location. This signal immediately will send to the FCM.



```
Time ↑          Level  Function     Event message
Mar 21, 2019
                       helloWorld  ▸ Billing account not configured. External network is not accessible and quotas are severely limited. Con…
1:13:01.445 AM         helloWorld    Function execution took 53 ms, finished with status: 'ok'
1:13:08.433 AM         helloWorld    Function execution started
1:13:08.433 AM         helloWorld  ▸ Billing account not configured. External network is not accessible and quotas are severely limited. Con…
1:13:08.614 AM         helloWorld    Function execution took 182 ms, finished with status: 'ok'
1:13:10.595 AM         helloWorld    Function execution started
1:13:10.595 AM         helloWorld  ▸ Billing account not configured. External network is not accessible and quotas are severely limited. Con…
1:13:10.682 AM         helloWorld    Function execution took 88 ms, finished with status: 'ok'
1:17:47.492 AM         helloWorld    Function execution started
1:17:47.493 AM         helloWorld  ▸ Billing account not configured. External network is not accessible and quotas are severely limited. Con…
1:17:47.819 AM         helloWorld    Function execution took 328 ms, finished with status: 'ok'
```

Figure 4.24: FCM Receive the signal

Figure 4.24 shows the motion signal that send from Firebase Database. After FCM receive the signal, it will transfer the signal to the API Communication. According to the specific token id, the message will send to the user android.

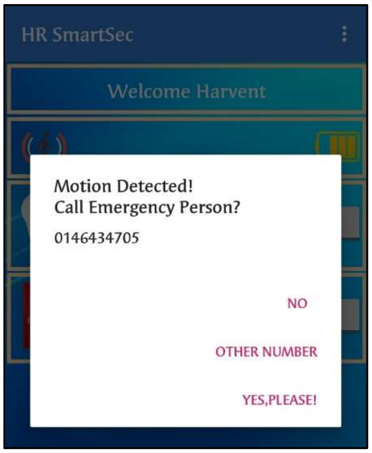


HR SmartSec • now

Motion Detected!

Someone enter illegal to the target location!

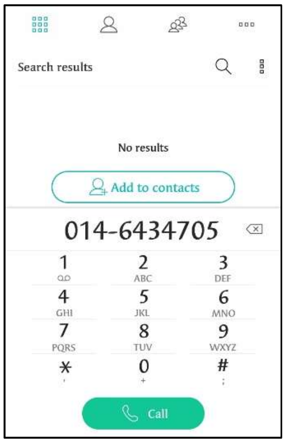
Figure 4.25: Push up notification



Motion Detected!
Call Emergency Person?

0146434705

Figure 4.26: Pop up message on android app


Search results

No results

014-6434705

Figure 4.27: Call Function

Figure 4.25 shows the push up notification that send from the FCM. Figure 4.26 indicates the pop alert message. On the pop alert message is also displaying the register emergency number. User can call the emergency number by pressing yes button as shown in Figure 4.27 or else if other number means user can press other number button. If user want to ignore the message means user can press no button. This whole system is design as user friendly. It is easy to access the system.

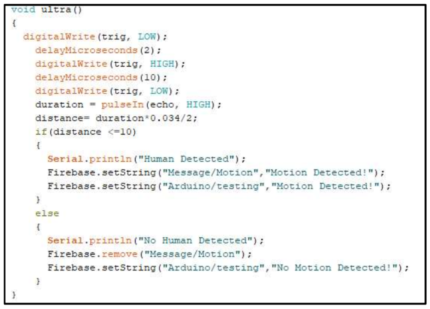
4.3.3 User Acceptance Testing

Result of User Acceptance Test can be referred in Appendices.

4.4 RESULT AND DISCUSSION

Based on Table 4.1 shows the integration testing result. Apart from the observation, there have 7 results (6 pass and 1 fail). Each test data has its own reason of the results. This subtopic will explain and discuss the reason the results and solution for the fail results.

In first test data, the ultrasonic sensor can read the motion and can transfer the signal to the Firebase. Figure 4.28 shows the code for the motion detection. This line of the code will help to transfer the signal to Firebase via ESP8266 Nodemcu using Internet. Instead of using ultrasonic sensor, developer is also can use PIR sensor and IR sensor to read the motion.

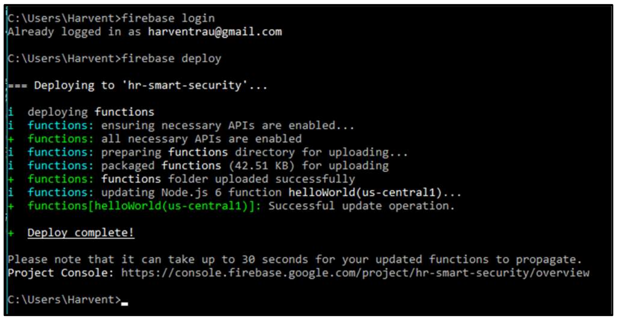


```
void ultra()
{
  digitalWrite(trig, LOW);
    delayMicroseconds(2);
    digitalWrite(trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig, LOW);
    duration = pulseIn(echo, HIGH);
    distance= duration*0.034/2;
    if(distance <=10)
    {
      Serial.println("Human Detected");
      Firebase.setString("Message/Motion","Motion Detected!");
      Firebase.setString("Arduino/testing","Motion Detected!");
    }
    else
    {
      Serial.println("No Human Detected");
      Firebase.remove("Message/Motion");
      Firebase.setString("Arduino/testing","No Motion Detected!");
    }
}
```

Figure 4.28: The code for motion detection

Next test data is Firebase Cloud Message (FCM) is fail to send notification to Android App. This is because the token id in firebase is not similar will the Android App. The token id is the unique id for both firebase and android app. So that it is important to configure similar the token id to transfer the notification from FCM to Android App.

There have a several step to configure the token id. First of all, developer need to identify the token id in firebase and the same time need to identify the token id in Android App. And then user need to add the token id in visual code as shown in Figure 4.21. After that developer need to save the Visual code file while need to open the Node.js file by opening Commend Prompt (CMD). In CMD, developer need to add a few commands to activate the API communication to allow the notification transfer from FCM to Android App. Figure 4.29 shows the line of the code. Once the deploy is complete, user can receive the push up notification if the device detect and human movement.



```
C:\Users\Harvent>firebase login
Already logged in as harventrau@gmail.com

C:\Users\Harvent>firebase deploy

=== Deploying to 'hr-smart-security'...

i  deploying functions
i  functions: ensuring necessary APIs are enabled...
+  functions: all necessary APIs are enabled
i  functions: preparing functions directory for uploading...
i  functions: packaged functions (42.51 KB) for uploading
+  functions: functions folder uploaded successfully
i  functions: updating Node.js 6 function helloWorld(us-central1)...
+  functions[helloWorld(us-central1)]: Successful update operation.

+  Deploy complete!

Please note that it can take up to 30 seconds for your updated functions to propagate.
Project Console: https://console.firebase.google.com/project/hr-smart-security/overview

C:\Users\Harvent>_
```

Figure 4.29: The Node.js code

Besides of that, the fourth test data is about light status where if user click on button from android app, the device will start to light up. The intermediate device that help to transfer the signal from android app to device is called Relay module as shown in Figure 4.30. This relay module will activate the light according to the instruction code. The code line is shown in Figure 4.31.

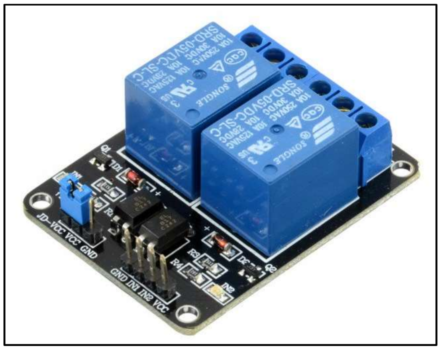
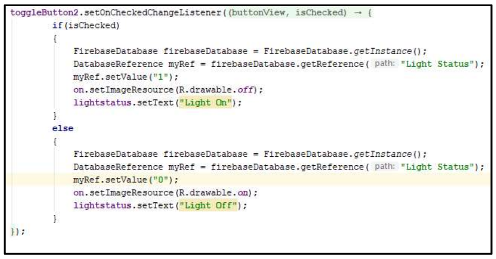


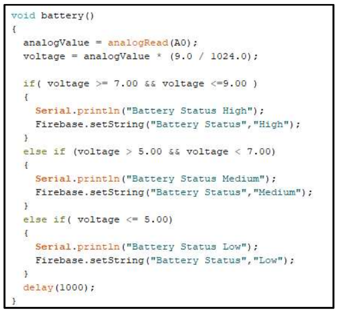
Figure 4.30: Relay module



```
toggleButton2.setOnCheckedChangeListener((buttonView, isChecked) → {
        if(isChecked)
        {
            FirebaseDatabase firebaseDatabase = FirebaseDatabase.getInstance();
            DatabaseReference myRef = firebaseDatabase.getReference( path: "Light Status");
            myRef.setValue("1");
            on.setImageResource(R.drawable.off);
            lightstatus.setText("Light On");
        }
        else
        {
            FirebaseDatabase firebaseDatabase = FirebaseDatabase.getInstance();
            DatabaseReference myRef = firebaseDatabase.getReference( path: "Light Status");
            myRef.setValue("0");
            on.setImageResource(R.drawable.on);
            lightstatus.setText("Light Off");
        }
});
```

Figure 4.31: Light status code

The last test data is the battery status which work fine. It can able to read the voltage from the device and can show the correct icon in the android app. The code line to display the battery status is shown in Figure 4.32.



```
void battery()
{
  analogValue = analogRead(A0);
  voltage = analogValue * (9.0 / 1024.0);

  if( voltage >= 7.00 && voltage <=9.00 )
  {
    Serial.println("Battery Status High");
    Firebase.setString("Battery Status","High");
  }
  else if (voltage > 5.00 && voltage < 7.00)
  {
    Serial.println("Battery Status Medium");
    Firebase.setString("Battery Status","Medium");
  }
  else if( voltage <= 5.00)
  {
    Serial.println("Battery Status Low");
    Firebase.setString("Battery Status","Low");
  }
  delay(1000);
}
```

Figure 4.32: Battery Status Code

During this phase, all the output from the Smart Security System was discussed and analysed whether it has followed requirement and criteria needed.

CHAPTER 5

CONCLUSION

5.1 INTRODUCTION

This chapter is mainly about the overall conclusion of the Android Based Smart Security System using Internet of Things (IoT) and Firebase, drawback as well as future enhancement that can be made in the future development.

5.2 OBJECTIVE ACHIEVEMENT

The aim of these objectives is to develop the anti-burglar system by applying the minimalism factor into the system with varies of features. In this system, the development is based on the objectives stated as below:

I. To study a smart way to monitor house security using IoT device that can detect human movement and send alert messages to user when motion detected.

II. To design and develop a low-cost innovative monitoring device using ESP8266 Nodemcu and Ultrasonic Sensor.

III. To evaluate the monitoring device functions and usability.

5.2.1 First Objective

The first objective is to study a smart way to monitor house security using IoT device that can detect human movement and send alert messages to user when motion detected. Whenever a motion is cross on target location, the device is able to read the motion and able to send the data to user via Internet using Ultrasonic sensor and ESP8266 Nodemcu. The first objective is success.

5.2.2 Second Objective

The second objective is to design and develop a low-cost innovative monitoring device using ESP8266 Nodemcu and Ultrasonic Sensor. The device is successfully developed in innovative ways with low cost. My device can read human movement and can send alert message to user without knowledge of user.

5.2.3 Third Objective

The third objective is to evaluate the monitoring device functions and usability. Last but not least, after all the testing made it, my device successfully detects human in range of target location and send the push up notification to user. This device was approved by clients where the result of User Acceptance Test can be referred in Appendices.

Thus, by reviewing the document and the system behaviour, it can be concluded that this system has met the objectives that have been stated.

5.3 CONTRIBUTION

The main contribution of this system is to produce monitoring system by using ESP8266 Nodemcu and Ultrasonic Sensor with execution of other web services. Android based smart security system using Internet of Things and Firebase gives advantages to user as it can be used easily without having any misunderstanding of system operation and procedures. User manual have been provided for user to know the Smart Security System behaviour, operation, procedures and other related information.

In terms of budget, Smart Security System has low cost with low maintenance for development. This is because this system apply minimalism concept with varies of features provided. This concept inspires from people who are unable to have their own security despite of knowing the expensive price in the market.

Furthermore, the most important core for this system is to protect the user belongings by tracking the motion activities. For sure, somehow the user feels uneasy when they go somewhere. This is because user might have insecure feeling about their place. For that case, this system will overcome all the worries and user will have the secure feeling when user is not around the place. The system will detect for human movement and if the system found any motion, it will send a signal to the user through a push up notification. From that notification, user can be alert and can call either their neighbour, security or any emergency person to check their place.

5.4 DRAWBACK AND FUTURE ENHANCEMENT

During development for any systems, they might have its ups and down problem. Problems may appear before the process, during the process or after the completion of the process. There are several factors that hold up the system development and its performances. The system drawback factors can be classified into three categories: -

a) Time constraints

Time range give for the development is too short. The development by applying waterfall model consume a a lot of time due to many factors such as limited skill in programming, problem skill, elicited requirement from the client and more.

b) Development tools constraint

Smart Security System requires Raspberry pi board and raspberry camera module before undergo the development. Apart from that, the source is very expensive, and the configuration part is too complicated. Since the source is very expensive, it will not suite for my objective of the project.

c) Skill constraints

Limited skills available on one developer as the Python language is not familiar as the Arduino language. Plus, there is no exposed from learning and teaching plan from course structure. As a solution, they need to search, learn and apply the language by themselves to the system. When they undergo for researching and learning the language, it might take time to fully understand before applied to the system.

In addition, some development hardware and software are required specific knowledge skill to use them. For an example, Raspberry pi board and raspberry camera module is used to sketch the python language and as a medium to upload the sketch to the raspberry pi board. Another example, Putty software and raspberry pi command software are needs to configure the raspberry pi. Unfortunately, Putty software and raspberry pi command software is not familiar with the developer. This is because the both software only used to give command via SSH to Raspberry pi. Same situation as Arduino IDE, developer must need to know how the Putty software and raspberry pi command software work thorough the SSH.

The application seem can be improved by build other interesting applications using the system such as build a camera that take snapshots at regular intervals. By having this method, camera can move at certain distant and take snapshot for each motion detection at target location. This system also can be extending by adding raspberry pi zero wireless board and camera module to have a complete video monitoring system. This will enhance the security level for the user. Furthermore, Smart Security System can focus on one scope of very important items for user like jewellery, documents and so many.

5.5 SUMMARY

As a conclusion, Android based smart security system using Internet of Things and Firebase system has met the criteria and objectives based on the literature report. The device is a user-friendly and ease to be adapted when interacts with it. By using waterfall model methodology, there are some advantages have been observed such as requirement are well defined, clearly documented and fixed values. Besides of that, it works well for smaller projects where requirement is very well understood and easy to manage. Although the device only consist simple hardware and software integration, but it still can worked properly and enable to reduce the security cases.

REFERENCE

1Sheeld. (n.d.). Security System using Arduino Bluetooth Camera. Retrieved from https://1sheeld.com/tutorials/arduino-bluetooth-camera-security-system/

Ada, Lady. (n.d.). PIR. Retrieved from https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor?view=all

Barragan, H. (n.d.). Breadboard. Retrieved from http://wiring.org.co/learning/tutorials/breadboard/

CCTV Camera Set. (n.d.). Retrieved from https://www.lazada.com.my/products/plug-and-play-4ch-1080p-hd-wireless-nvr-kit-p2p-720p-waterproof-outdoor-ir-security-ip-camera-wifi-home-cctv-system-set-i195664597-s240294895.html?spm=a2o4k.searchlist.list.1.71e837a8zYnz5P&search=1

Cytron. (2007). Components Price. Retrieved from https://www.cytron.io

Dejan Nedelkovski. (n.d.). Arduino Security and Alarm System Project. *How To Mechatrones*. Retrieved from https://howtomechatronics.com/projects/arduino-security-alarm-system-project/

Garbutt, G. (n.d.). How far are you willing to go to monitor employee activity? Retrieved from https://www.peoplehr.com/blog/index.php/2016/12/14/how-far-are-you-willing-to-go-to-monitor-employee-activity/

Heikopikner. (n.d.). Ultrasonic Sensor. Retrieved from http://home.roboticlab.eu/en/examples/sensor/ultrasonic_distance

HEMMINGS, M. (n.d.). Jumpers. Retrieved from http://blog.sparkfuneducation.com/what-is-jumper-wire

Jansen, S. (n.d.). Android Bluetooth Motion Sensor Using Arduino. Retrieved from http://shanejansen.com/android-bluetooth-motion-sensor-using-arduino/

KHAN, R. (2015). Zener Diode Basic Operation and Applications. Retrieved from https://www.digikey.com/en/maker/blogs/zener-diode-basic-operation-and-applications

Kumar, S. V. S. V. P. D., & Shyam Akashe. (n.d.). No Implementation of GSM

Based Security System with IOT ApplicationsTitle. *International Journal of Computer Network and Information Security*. Retrieved from https://www.researchgate.net/publication/317413875

Mellon, L. (2017). Data Transmission – Parallel vs Serial.

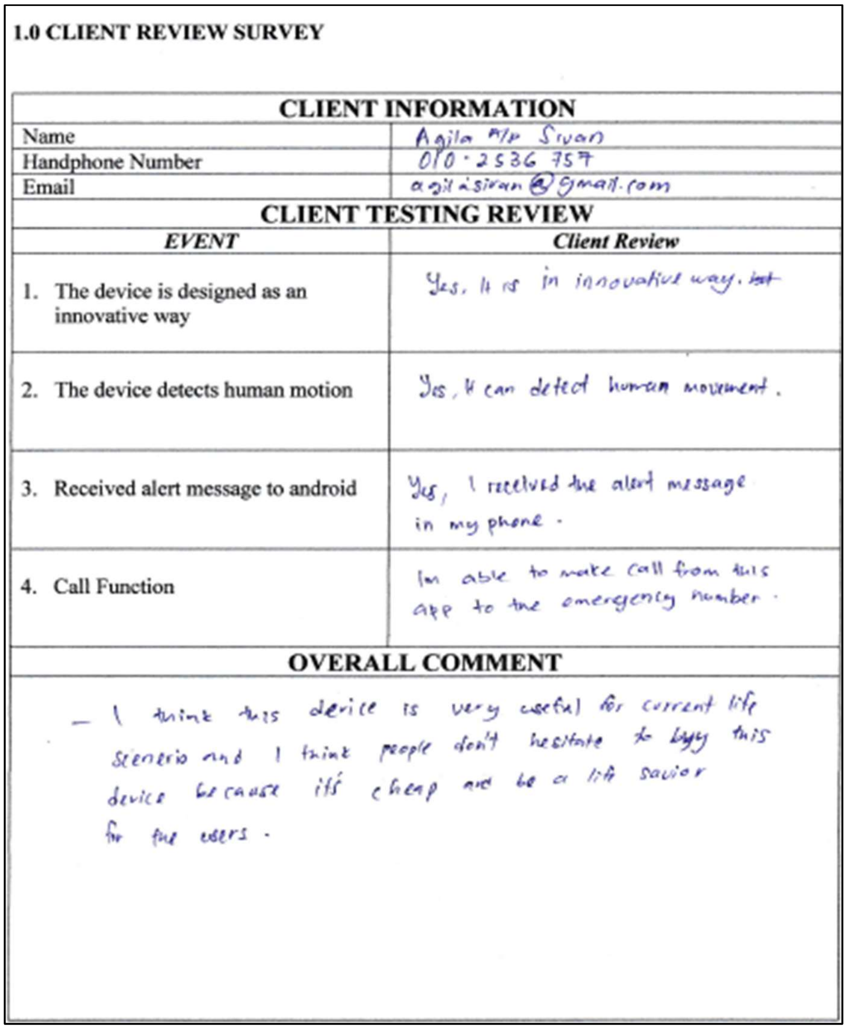
Rada, Z. (2016). Smart Motion Detection: Security System Using Raspberry Pi. *International Journal of Computer Network and Information Security*. Retrieved from https://www.researchgate.net/publication/312218161

SparkFun. (n.d.). WiFi Module - ESP8266. Retrieved from https://www.sparkfun.com/products/13678

SpecialistsSign, C. (n.d.). Solderinf Iron. Retrieved from https://www.circuitspecialists.com/soldering-irons

Techmirtz. (2017). Visual Capturing with OV7670 on Arduino. Retrieved from https://www.hackster.io/techmirtz/visual-capturing-with-ov7670-on-arduino-069ebb

APPENDICES



1.0 CLIENT REVIEW SURVEY

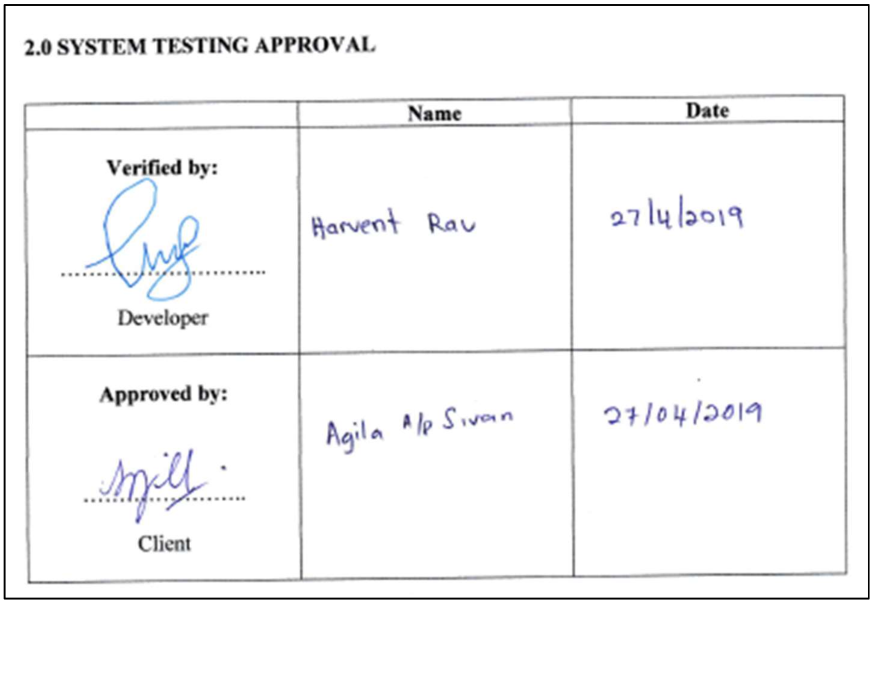
CLIENT INFORMATION	
Name	Agila A/p Sivan
Handphone Number	010-2536 757
Email	agilasivan@gmail.com

CLIENT TESTING REVIEW

EVENT	*Client Review*
1. The device is designed as an innovative way	Yes. It is in innovative way. ~~but~~
2. The device detects human motion	Yes, it can detect human movement.
3. Received alert message to android	Yes, I received the alert message in my phone.
4. Call Function	Im able to make call from this app to the emergency number.

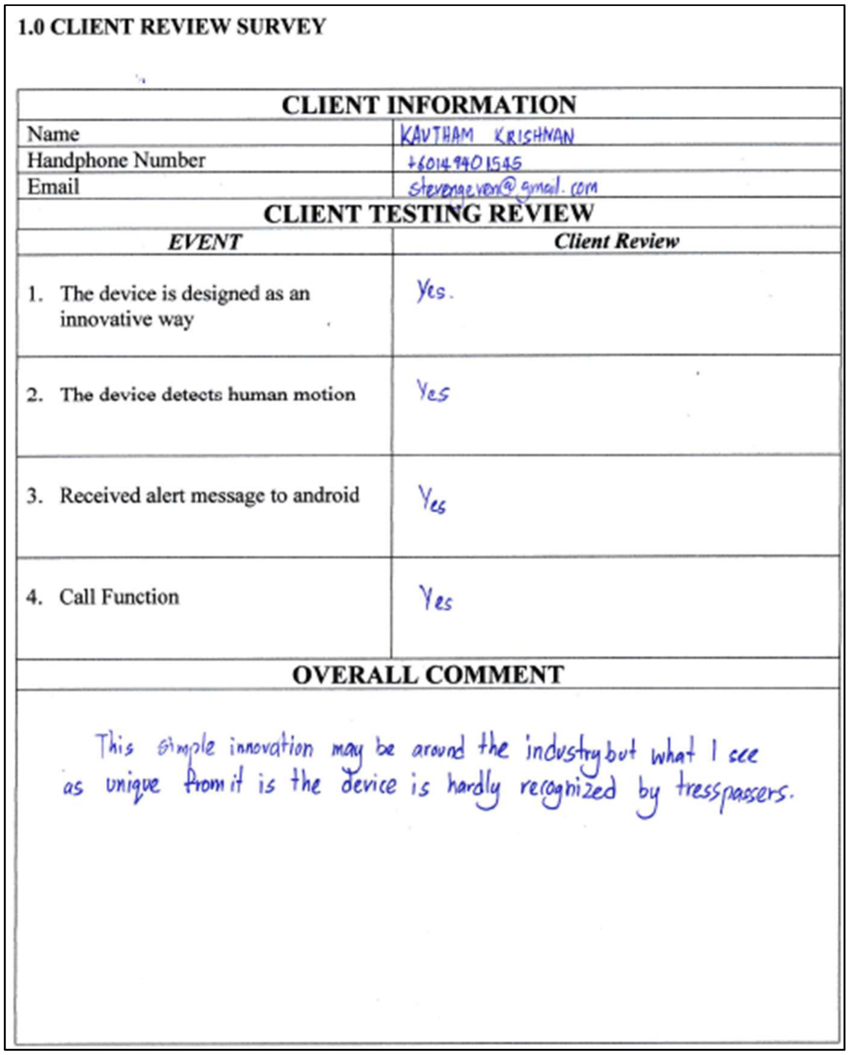
OVERALL COMMENT

- I think this device is very useful for current life scenerio and I think people don't hesitate to buy this device because it's cheap and be a life savior for the users.



2.0 SYSTEM TESTING APPROVAL

	Name	Date
Verified by: …………………… Developer	Harvent Rau	27/4/2019
Approved by: …………………… Client	Agila A/p Sivan	27/04/2019



1.0 CLIENT REVIEW SURVEY

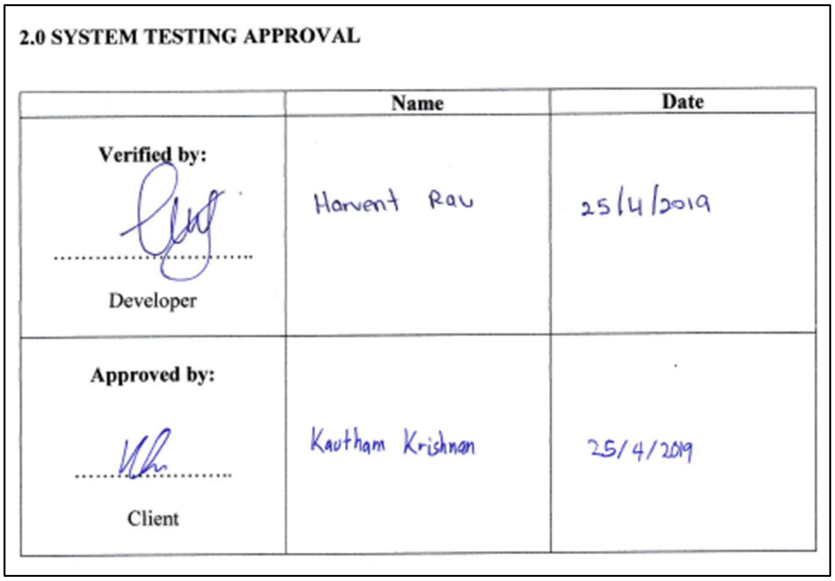
CLIENT INFORMATION	
Name	KAVTHAM KRISHNAN
Handphone Number	+60149401545
Email	stevengeven@gmail.com

CLIENT TESTING REVIEW

EVENT	*Client Review*
1. The device is designed as an innovative way	Yes.
2. The device detects human motion	Yes
3. Received alert message to android	Yes
4. Call Function	Yes

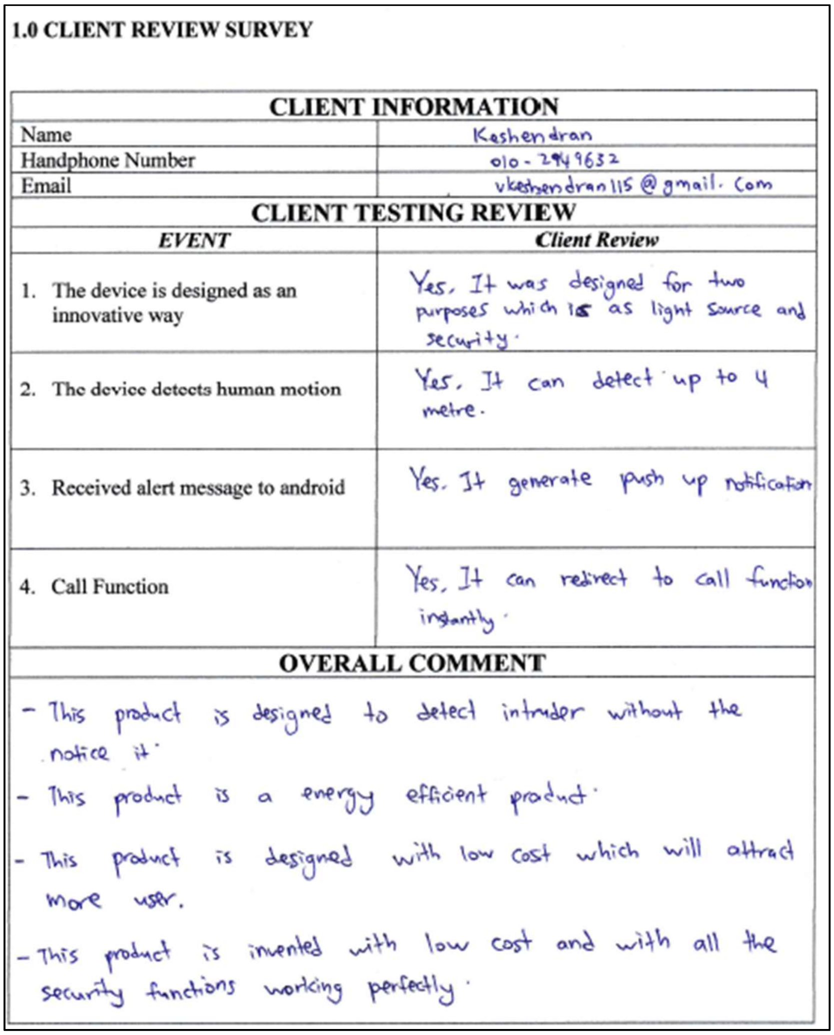
OVERALL COMMENT

This simple innovation may be around the industry but what I see as unique from it is the device is hardly recognized by tresspassers.



2.0 SYSTEM TESTING APPROVAL

	Name	Date
Verified by: …………………………….. Developer	Harvent Rau	25/4/2019
Approved by: …………………………….. Client	Kautham Krishnan	25/4/2019



1.0 CLIENT REVIEW SURVEY

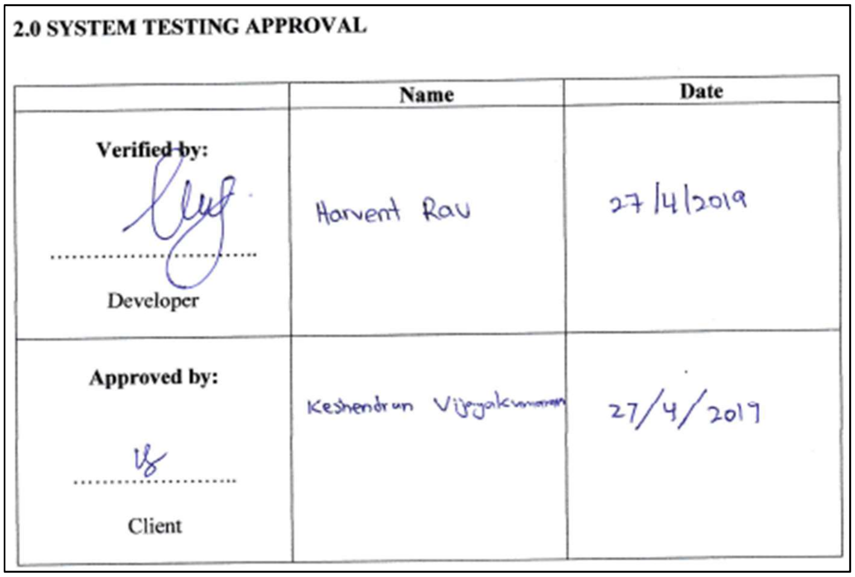
CLIENT INFORMATION	
Name	Keshendran
Handphone Number	010-2949632
Email	vkeshendran115@gmail.com

CLIENT TESTING REVIEW

EVENT	*Client Review*
1. The device is designed as an innovative way	Yes, It was designed for two purposes which is as light source and security.
2. The device detects human motion	Yes, It can detect up to 4 metre.
3. Received alert message to android	Yes. It generate push up notification
4. Call Function	Yes, It can redirect to call function instantly.

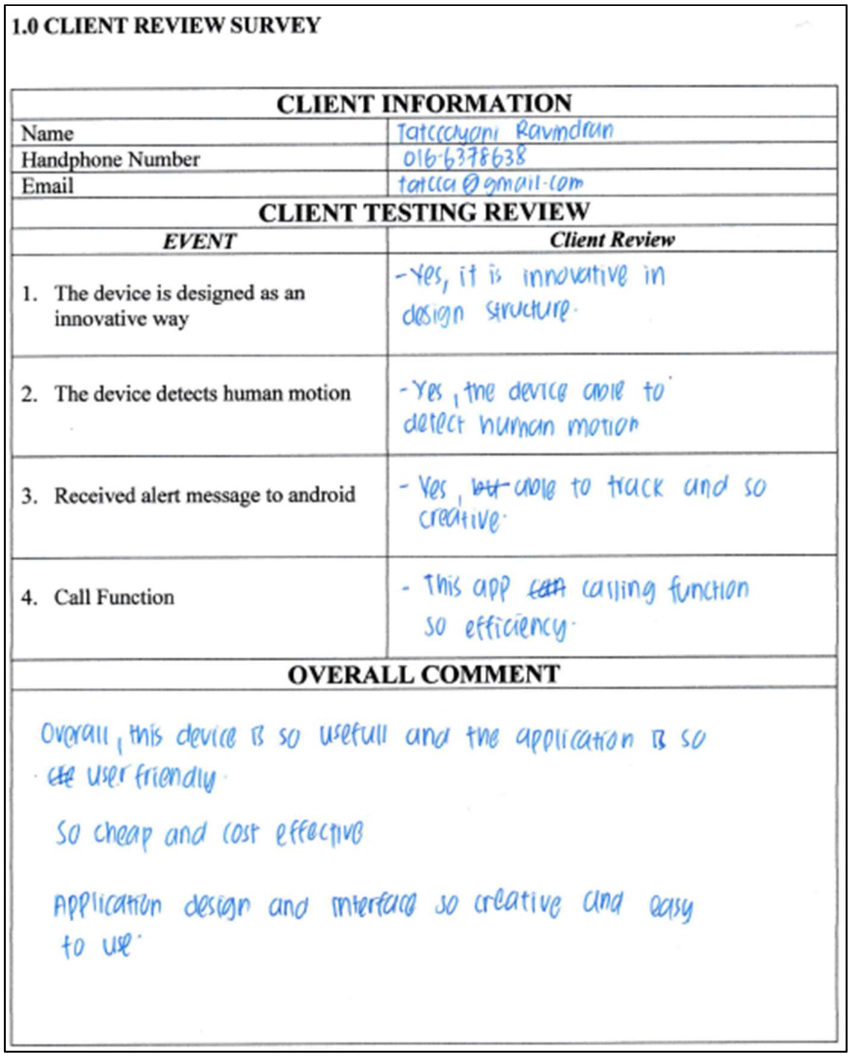
OVERALL COMMENT

- This product is designed to detect intruder without the notice it.
- This product is a energy efficient product.
- This product is designed with low cost which will attract more user.
- This product is invented with low cost and with all the security functions working perfectly.



2.0 SYSTEM TESTING APPROVAL

	Name	Date
Verified by: .... Developer	Harvent Rau	27/4/2019
Approved by: .... Client	Keshendran Vijayakumaran	27/4/2019



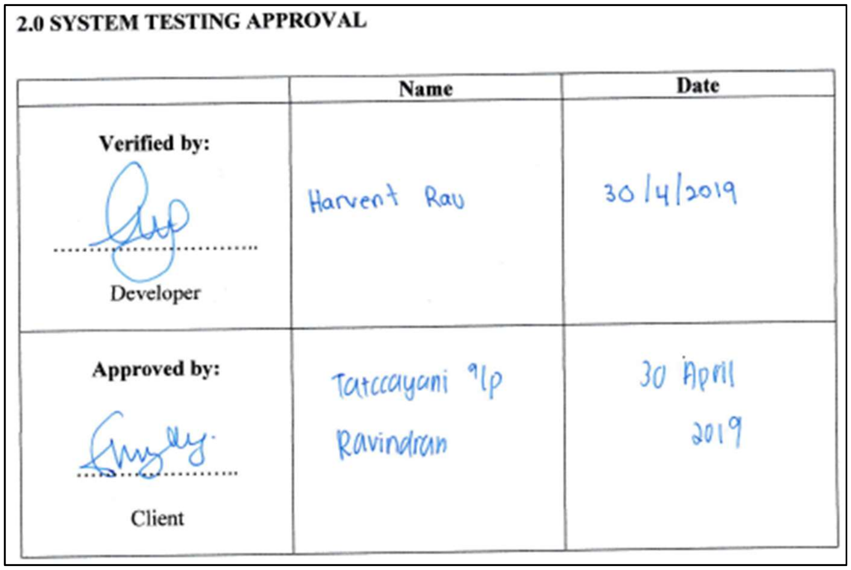
1.0 CLIENT REVIEW SURVEY

CLIENT INFORMATION	
Name	Tatccayani Ravindran
Handphone Number	016-6378638
Email	tatcca@gmail.com
CLIENT TESTING REVIEW	
EVENT	***Client Review***
1. The device is designed as an innovative way	-Yes, it is innovative in design structure.
2. The device detects human motion	-Yes, the device able to detect human motion
3. Received alert message to android	- Yes, ~~but~~ able to track and so creative.
4. Call Function	- This app ~~can~~ calling function so efficiency.
OVERALL COMMENT	

Overall, this device is so usefull and the application is so ~~ete~~ user friendly.

So cheap and cost effective

Application design and interface so creative and easy to use.



2.0 SYSTEM TESTING APPROVAL

	Name	Date
Verified by: Developer	Harvent Rau	30/4/2019
Approved by: Client	Tatccayani a/p Ravindran	30 April 2019