FRIENDLY HEALTHY PULSE HEART BEAT SENSOR

(FRIENDLY HePuHebSor)

ASMAWATI BINTI FARAKKASI

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LAMPIRAN B: BORANG PENGESAHAN STATUS TESIS

Contoh halaman pengesahan

BORANG PENGESAHAN STATUS TESIS*

JUDUL: FRIENDLY HEALTHY PULSE HEATBEAT SENSOR
(FRIENDLY HePuHebSor)

SESI PENGAJIAN: 2017/1028

Saya ASMAWATI BINTI FARAKKASI
(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

Tesis dan projek adalah hakmilik Universiti Teknikal Malaysia Melaka.
Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
** Sila tandakan (/)

_____	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
_____	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
___/___	TIDAK TERHAD	

(TANDATANGAN PENULIS)

Alamat tetap:

Pasir Putih Baru, Lorong 6A Mawar

91008, Tawau Sabah

Tarikh: 29 August 2017

(TANDATANGAN PENYELIA)

DR. NAZRULAZHAR BAHAMAN
Nama Penyelia

Tarikh: 29/08/2017

CATATAN: * Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM) ** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

FRIENDLY HEALTHY PULSE HEART BEAT SENSOR

(FRIENDLY HePuHebSor)

ASMAWATI BINTI FARAKKASI

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Computer Networking)

FACULTY OF INFORMATION AND COMMUNICATION

TECHNOLOGY UNIVERSITI TEKNIKAL MALAYSIA MELAKA 2017

DECLARATION

I hereby declare that this project report entitled

FRIENDLY HEALTHY PULSE HEART BEAT SENSOR

(FRIENDLY HePuHebSor)

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT: _____ Date : 29 August 2017

(ASMAWATI BINTI HARAKKASI)

SUPERVISOR: _____ Date: 29/08/2017

(DR. NAZRULAZHAR BAHAMAN)

DEDICATION

Thanks to Allah s.w.t

for giving me a chances to complete my PSM

To my beloved my parents,

Farakkasi Labundu and Isawami Latedu,

Thank you for your prayers and moral support.

To my supervisor,

Dr. Nazrulazhar Bahaman,

I would like to thank you a lot for keep encouraging me to complete this project,

and give non-stop guidance and advices throughout this journey.

To my friend,

I would like to thank you for your support.

ACKNOWLEDGEMENTS

All Praise to Allah SWT the Almighty for giving me the strength to finish my PSM. Thanks to Allah because I manage to complete my final year project without a major hiccup.

I am grateful to have my supervisor, Dr. Nazrulazhar Bahaman of his priceless effort in helping me whenever I find difficulties in completing my task and for swotting my report and comments for improvement for this report.

Special thanks to my family, especially my lovely parents, Farakkasi Labundu and Isawami Lattedu for the continuous support throughout this journey from day to day.

I would like to thank my friends for their time, concern, efforts and always encouraging me when preparing this report. A big thank to everybody that involve in my project directly or indirectly. Last but not least, thanks to Universiti Teknikal Malaysia Melaka (UTeM) for the opportunity given. I pray to Allah SWT may He send you His blessing.

ABSTRACT

Heart rate is a measured of counting beats over a set period of time influenced by physiological factors like the circadian cycle, posture, blood pressure, and physical activity. This project is to design and develop the prototype of a simple, concise and efficient heart rate monitoring using pulse sensor amped based on IoT. Pulse sensor (optical-based heart rate sensor) is specially created to detect pulse. Pulse sensor amped is an upgrade version of pulse sensor. Input signal from the pulse sensor amped is in the form of analog. It will be converted into digital signal using MCP3008. The data will be visualized in the form of real time graph using matplotlib and push notification will be sent if the heart rate is either Sinus Bradycardia or Sinus Tachycardia or normal using telegram-cli command-line client. Python language will be utilized for the interaction between the hardware on a linux based raspberry pi 3. The idea of developing this prototype is due to the existing heart rate monitoring is quite expensive, painful, risky and not really friendly to be used. In this project, a few simple steps need to be done as a project methodology to complete it. Firstly, literature review (previous research and theory) has to be done to gain information about the hardware, software, language, method that will be used for this project. Next, list all the components and do the project design for the hardware and software. At this point, the component and design must be suitable and meet the requirement for this project to solve the problem statement. Next is hardware development, start to work on the prototype by implementing all the software and the hardware. Finally is testing and verification where the prototype will be tested and the project will be documented for verification. This project are varies in contribution such as it is a better tool to measure heart rate, Smart step precaution for Sinus Bradicardya (slow heart rate) and Sinus Tachycardia(fast heart rate) and friendly IoT project.

ABSTRAK

Kadar jantung ialah diukur mengira denyutan dalam tempoh satu set masa dipengaruhi oleh faktor-faktor fisiologi seperti kitaran circadian, postur, tekanan darah, dan aktiviti fizikal. Projek ini adalah untuk mereka dan membangunkan prototaip kadar jantung yang mudah, ringkas dan berkesan pemantauan menggunakan „pulse sensor amped‟ berdasarkan IOT. „Pulse sensor‟ (sensor kadar jantung berdasarkan optik) dicipta khas untuk mengesan nadi. „Pulse sensor amped‟ ialah versi naik taraf „pulse sensor‟. isyarat input daripada „pulse sensor amped‟ adalah dalam bentuk analog. Ia akan ditukar ke isyarat digital menggunakan MCP3008. Data ini akan digambarkan dalam bentuk real Graf menggunakan matplotlib dan pemberitahuan menolak akan dihantar jika kadar jantung sama ada Sinus Bradycardia atau Sinus tachycardia atau biasa menggunakan telegram-cli Client. Python akan digunakan untuk interaksi antara perkakasan pada linux berdasarkan raspberry pi 3. Idea untuk membangunkan prototaip ini adalah disebabkan oleh pemantauan kadar jantung yang sedia ada agak mahal, menyakitkan, berisiko dan tidak benar-benar mesra yang akan digunakan. Dalam projek ini, beberapa langkah mudah yang perlu dilakukan sebagai metodologi projek untuk menyiapkannya. Pertama, kajian literatur (penyelidikan dan teori sebelumnya) yang perlu dilakukan untuk mendapatkan maklumat mengenai perkakasan, perisian, bahasa, kaedah yang akan digunakan untuk projek ini. Kemudian, senaraikan semua komponen dan melakukan reka bentuk projek bagi perkakasan dan perisian. Pada ketika ini, komponen dan reka bentuk mestilah sesuai dan memenuhi keperluan untuk projek ini untuk menyelesaikan pernyataan masalah. Seterusnya ialah pembangunan perkakasan, mula bekerja pada prototaip dengan melaksanakan semua perisian dan perkakasan. Akhir sekali adalah ujian dan pengesahan di mana prototaip akan diuji dan projek itu akan didokumentasikan untuk pengesahan. projek ini adalah berbeza dari segi sumbangan itu kerana ia adalah alat yang lebih baik untuk mengukur kadar jantung, Smart langkah langkah berjaga-jaga untuk Sinus Bradicardya (kadar jantung yang perlahan) dan Sinus tachycardia (kadar jantung cepat) dan projek IOT mesra.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

LED	-	Ling Emitting Diode
ADC	-	Analog to Digital Converter
HEPUHEBSOR	-	Healthy Pulse Heartbeat Sensor
PS	-	Project Statement
PQ	-	Project Question
PO	-	Project Objective
PC	-	Project Contribution
SA	-	Sinoatrial node
BPM	-	Beat Per Minute
ECG	-	Electrocardoagram
KSPS	-	Kilo Sample Per Second
PDIP	-	Plastic Dual In-Line Packages
SOIC	-	Small Outline Integrated Circuit
SPI	-	Serial Peripheral Interface
IOT	-	Internet Of Thing
GPIO	-	General-Purpose Input/Output
MHZ	-	Mega Hertz
RAM	-	Random Access Memory
CPU	-	Central Processing Unit
WIFI	-	Wireless Fidelity
GUI	-	Graphical User Interface
ME	-	Micro Edition
API	-	Application Program Interface

LCD	-	Liquid Crystal Display
MATLAB	-	Matrix Laboratory
CLI	-	Command Line Interface
GSM	-	Global System Mobile Communication
ABG	-	Arterial Blood Gas
MB	-	Monitoring Box
PO	-	Pulse Oximeter
GPU	-	Graphics Processing Unit
SDRAM	-	Synchronous Dynamic Random-Acces Memory
SD	-	Secure Digital
V	-	Voltage
CH	-	Channel
SCLK	-	Serial Clock
MISO	-	Master In Slave Out
MOSI	-	Master Out Slave In
CE	-	Channel Enable
CS	-	Channel Select
DGND	-	Digital Ground
AGND	-	Analog Ground
SHDN	-	Shutdown
DIN	-	Data In
DOUT	-	Data Out
VREF	-	Voltage Reference
VDD	-	Voltage Drain-to-Drain
VCC	-	Voltage Common Collector

CHAPTER I

INTRODUCTION

1.1 Introduction

Heart is one of the most important organs in human‟s body[1]. Heart rate is a measured of counting beats over a set period of time. It is often influenced by physiological factors like the circadian cycle, posture, blood pressure, and physical activity[1]. Now, human are trying to figure it out on how to measure heart rate easily and accurate as it is important for our health. Enhancement made from year to year by contributing in a project to make a prototype/product that can measure the heart rate. For the future health technology, people might have their own personal health care which can be used anytime and everywhere. In this project, we are focusing on heart rate monitoring prototype to understand how it works.

Previously, researcher used LED and photo-detector through transmittance/reflectance method. The light from LED is transmitted through tissue and the photo-detector will detect the pulse[2]. The idea of gaining the pulse through tissue will be used in this project as the arterial blood is located near the skin such asour finger[2]. This study is focusing on pulse sensor amped on gaining input via analog to digital converter (ADC) chip that will be converted into pulse[3]. It's noticeably faster and easier to get reliable pulse readings because it added amplification and noise cancellation circuitry to the hardware[3].

To make our imagination become reality, we are trying to design a friendly healthy pulse heart beat sensor (HePuHebSor) prototype to measure heart rate together with chart plotting and push notification for the result.

1.2 Problem statement

Nowadays, people are very busy with their life until they do not have much time to go to the hospital or any other health centre to check their heart beat rate. A normal heart beat rate is the thing that everybody wants but our heart beat is always changing due to the work that we do for daily basis[1]. Sometimes people who are stress do not notice that their heart beat is beating a little bit faster than the normal rate and it may be a sign of a serious heart condition[1]. Those who are caffeine lover or alcoholic often forget that those two substances can increase the heart beat rate and they forgot they need to reduce the consumption of the two substances[1]. For those who like to exercise, sometimes they like to push themselves to the limit without taking care of their heartbeat and when they become too fatigued they will tend to have an oxygen debt which will cause bleeding inside the muscle and heartbeat rate can determined whether our body is tired or not[1]. Table 1.1 below shows the summary of problem statement that this project will embark upon.

PS	Problem Statement
PS1	Heart rate monitoring tools are quite expensive, painful, risky and not really friendly to be used.

Table 1.1: Summary of Problem Statement

1.3 Project Question

According to previous project, health rate is always uncertain once we analysed it and do we know the health condition based on the uncertain heart rate? Are there any other heart rate monitoring tools that can analyse the heart rate better? Table 1.2 below shows the summary of project questions.

PS	PQ	Project Question
PS1	PQ1	How to measure heart rate through arterial pulse (arterial bloods near the skin)?
	PQ2	How to know health condition through heart rate?
	PQ3	How effective LED + Photo-detector transmit light through tissue?

Table 1.2: Summary of Project Question

1.4 Project Objective

Based on project questions, this project has three main objectives to make sure the prototype can be developed completely. The design of the prototype must be simple and finally it can be tested effectively. Table 1.3 below is the summary of project objectives.

PS	PQ	PO	Project Objective
PS1	PQ1	PO1	To design a heart rate analyser with simple, concise and efficient.
	PQ2	PO2	To develop the prototype of Friendly HePuHebSor.
	PQ3	PO3	To test the effectiveness of the prototype.

Table 1.3: Summary of Project Objective

1.5 Project Scope

For this project, the target will be focused on several scopes:

1. Heart rate needs to be analysed because it indicate the health condition.
2. Data capturing is based on the input from pulse sensor via analog to digital converter (ADC) chip.
3. A graph plotting to find the average of heart rate and send the result to mobile phone.

1.6 Project Contribution

This prototype is designed for adults and kids to ease them to carry out daily work because heart rate is often influenced by physiological factors like the circadian cycle, posture, blood pressure, and physical activity. It is a better solution to save time and will be alert about what action to be taken to prevent a worse thing to happen. This product can be used by everybody instantly and everywhere. Table 1.4 below shows the summary of project contribution.

PS	PQ	PO	PC	Project Contribution
PS1	PQ1	PO1	PC1	Measure heart rate in a friendly way (easy, inexpensive, effective)
	PQ2	PO2	PC2	Propose step precaution (notification) through the reading of heart rate using mobile phone.
	PQ3	PO3	PC3	Propose a better tool to get the heart beat measurement by using pulse sensor amped.

Table 1.4: Summary of project contribution

1.7 Thesis Organisation

This study consists of 7 chapters. Chapter one talk about the introduction of friendly healthy pulse heart beat sensor (HePuHebSor). This concern leads to study on the idea of developing the prototype based on the project background, problem statement, project question, project objective, project scope and project contribution. Next chapter is literature review. We need to review few journals that related to this project. Chapter three is project methodology. This chapter includes project milestones where action is needed for each stage of activities to develop the prototype of HePuHebSor. Chapter four is the analysis and design of the prototype. Chapter five is implementation. In this chapter, we need to have the expected output after we list all the things that should be implemented for this project. Chapter six is testing where we already have the prototype of HePuHebSor and we test it to get the result. The last chapter is conclusion where we conclude everything from the first chapter until the last chapter on what we have done for this project.

1.8 Conclusion

This chapter contains detail information of the proposed project, friendly healthy pulse heart beat sensor (HePuHebSor). Based on this chapter, the problem that occurred in previous project can be identified. Thus, the main objectives and the project scopes are clearly defined. Chapter 2 will discuss on the literature review that will be used in this project.

CHAPTER II

LITERATURE REVIEW

2.1 Introduction

A literature review is a writing of a scholarly paper, which includes the current knowledge, including functional findings, as well as theoretical and methodological helps to this project. These chapter reviews of articles, books and journals to understand the concept in order to complete this project.

2.2 Previous Work

Previously, IoT is widely used for healthcare application[13]. Heartbeat monitoring is one of them[9]. In this project, sensor is the major component. Therefore Pulse Oximeter is the best option to be referred as the reference compared to Arterial Blood Gas and Monitoring Box[4].

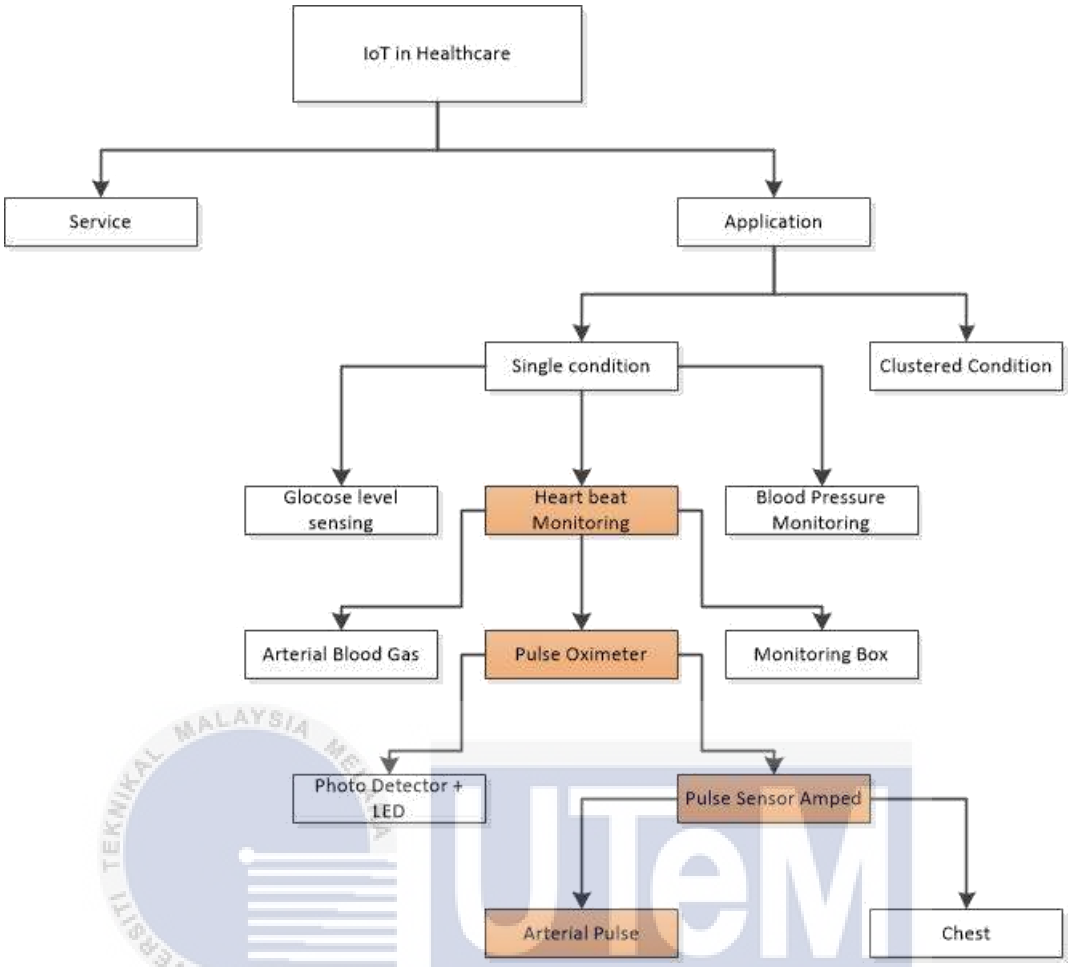

Figure 1.1 Taxonomy of IoT in healthcare system

2.3 Heart

Heart is one of the most important organs in our body[1]. The function of heart is to pump blood to the lung and all parts of the body via the circulatory system[1]. It supplies oxygen and nutrient to the body and eradicate carbon dioxide and other wastes[1].

The size of the heart can roughly be said the same like the size of a large fist and weighs between about 280 to 340 grams for men and 230 to 280 grams from women[23]. It has 4 chambers which are atria (upper chambers) and ventricles

(lower chambers). The heart contains an electrical cell which is "pacemaker" also known as Sinoatrial node (SA)[23]. It causes the heart to contract and produce a heartbeat and wave produced upon contraction[23]. Figue 2.2 shows the structure of heart. One complete cardiac cycle is 0.8 second[1].

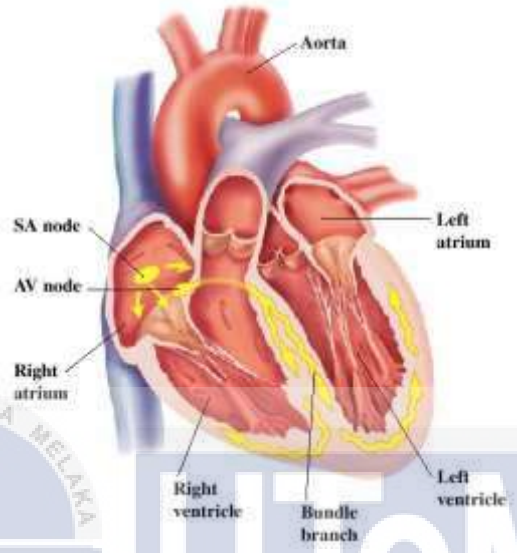

Figure 2.2: Structure of heart

2.3.1 Heart Rate

Heart rate or pulse is the number of heart beats per minute in a human body[24]. heart rate is often swayed by physiological factors like the circadian cycle, posture, blood pressure, and physical action[1]. There are three types of heart rate which are normal heart rate, lower heart rate and higher heart rate[23]. Each stages of age have different heart rate[1]. Scientifically, the children have higher normal heart rate compared to the adult and women have higher normal heart rate compared to men. The normal reference range in adults is between 60 beats per minute (bpm) to 100 bpm while below 60 bpm is considered as Sinus bradycardia (abnormally slow heart action) and higher than 100 bpm is termed Sinus tachycardia (heart rate that exceeds the normal resting rate). The infant rate of heart beat is around 130 - 150 bpm, the toddler‟s about 100 – 130 bpm, the older child‟s about 90 - 110 bpm [1].

2.3.2 Sinus Bradycardia

Sinus Bradycardia often related to slower heart rate[24]. Irony, a slow heart rate can be normal and healthy as it is common in young healthy adults, during sleep, and in elite athletes but the possibility for the heart may be at risk is quite high[1]. Slow heart rate can lessen coronary perfusion pressure (difference between the diastolic aortic pressure and the right atrial end diastolic pressure) especially for elderly with widened pulse pressure and firm vasculature[24]. Chronic lacking perfusion may induce or bother cardiovascular disease[24]. The symptoms of cardiovascular disease are fatigue, dizziness, syncope (temporary loss of consciousness caused by a fall in blood pressure) and dyspnoea (difficult or laboured breathing)[2]. Figure 2.3 shows ECG graph of Bradycardia.

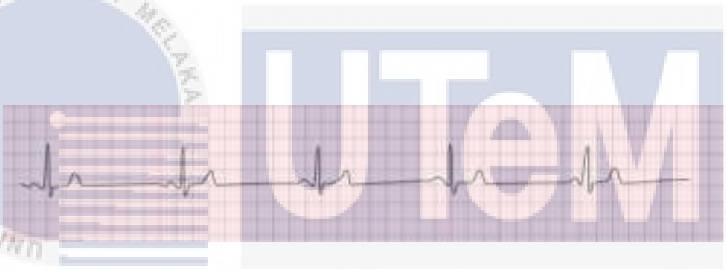

Figure 2.3: ECG graph of Bradycardia

2.3.3 Sinus Tachycardia

Sinus Tachycardia has the reverse meaning of Sinus Bradycardia which is higher heart rate[24]. Those who experience simple symptoms worsen in the upright position [3]. The symptoms may be intermittent and often include atypical chest discomfort, exercise intolerance, and orthostatic intolerance (disorder of the autonomic nervous system which is breathing occurring when an individual stands up)[3]. The symptom can cause high blood pressure (hypertension) Poor blood supply to the heart muscle due to coronary artery disease (atherosclerosis), heart valve disease, heart failure, heart muscle disease (cardiomyopathy), tumours, or infections[2]. Figure 2.4 shows ECG graph of Tachycardia.

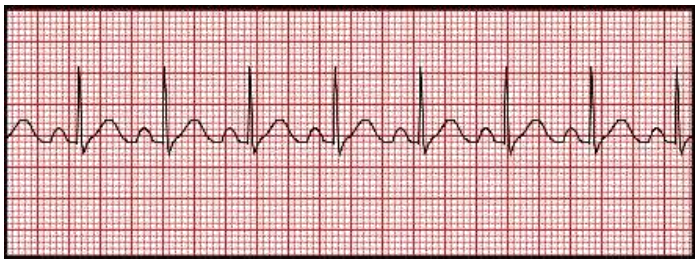

Figure 2.4: ECG graph of Tachycardia

2.3.4 Algorithm of Heart Rate Measurement

Heart beat is measured in beat per minute (bpm)[26]. Counting the number of pulses in given time will calculate an average while beat to beat calculation is done by measuring the time (T) in seconds, between two consecutive pulses, and converting the time into beats/min, using the formula beat/min. One complete cycle of heartbeat is 0.8 seconds at rest (normal)[26]. Karvonen formula can be used to get bpm in lesser than 1 minute [13]. The formula is simple, number of heart beat in „x" second should be time with „y" second to make sure the counting is 1 minute[13]. Figure 2.5 shows calculation of heartbeat over 4 minutes.

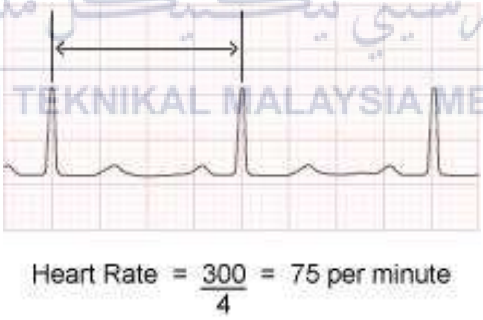

Figure 2.5: Calculation of heart beat of 300 beats over 4 minutes.

2.4 Pulse Detection

Pulse can be detected through arterial tissue[27]. Lately, there are many ways to detect pulse[27]. It can be done manually by using the fingertips gently in the

groove on the forearm, down from the fold of the wrist and do a manual beat per minute calculation[4]. It also can be done with the help of machine[4].

2.4.1 Pulse Sensor Amped

Pulse sensor (optical-based heart rate sensor) is specially created to detect pulse. It acts as an input provider in the form of analogue[4]. It works when the finger is placed on the sensor[5]. The pulse sensor can be worn on any other parts which is arterial bloods near the skin such as the earlobe and also on the wrist[6]. Pulse sensor amped is an upgrade version of pulse sensor. Pulse Sensor Amped adds amplification and noise cancellation circuitry to the hardware[6]. It's noticeably faster and easier to get reliable pulse readings[6]. Figure 2.6 shows pulse sensor amped.

Figure 2.6: Pulse Sensor Amped

2.4.2 Input via Analogue to Digital Converter

Analogue to Digital Converter (ADC) is a device to convert analogue signal to digital form[8]. The heart beat that measured by the sensor is processed by the ADC [7].

MCP3008 is a type of ADC chip. It is a serial peripheral interface (SPI) based analogue to digital converter (ADC)[8]. It has 8 analogue input channels that can be

arranged for single ended and differential ADC conversions[8]. It is a 10-bit ADC that can convert 200 kilo samples per second (ksps) and it comes in 28 Plastic Dual In-Line Packages (PDIP) and Small Outline Integrated Circuit (SOIC) packages[8]. The role of SPI is to allow the serial (one bit at a time) interchange of data between two devices where one called a master and the other called a slave[8]. It operates in full duplex mode. So, data can be shifted in both directions simultaneously[9].

Pulse sensor amped is originally built for arduino[5]. With the presence of ADC it can be connected to Raspberry pi with the installation of SPIdev package. Figure 2.7 shows MCP3008[9].

Figure 2.7: MCP3008

2.5 Computer to Mobile Interaction.

A computer is a device that can be instructed to carry out an arbitrary set of logical or arithmetic operations automatically[9]. It can be used to monitor and analyse the heart beat [9].

2.5.1 Raspberry Pi

Raspberry Pi is a processor for used in many IoT applications[11]. Analogue signal cannot be processed by Raspberry Pi[12]. It can only read signal in a digital form [11]. It works on linux platform and the cost is very low. Raspberry Pi has GPIO pins which are used to connect the sensor to it. It will act as a server after connect to the internet [10]. Raspberry Pi is also a fully customizable and programmable small computer board [11]. The most popular raspberry pi used for IoT is raspberry pi 2 model B and raspberry pi 3 model B. Both provide 1GB RAM storage. The CPU speed of raspberry pi 3 (1,200MHz, quad core) is higher compared to raspberry pi 2 (900MHz, quad core)[12]. Raspberry pi 3 has 802.11n built in wifi and Bluetooth 4.1 which raspberry pi 2 does not have [12].

Raspberry Pi is widely used for health monitoring[13]. Raspberry Pi is act as a small clinic after connecting these (Temperature, Respiration, Accelerometer, Heartbeat) sensors. Raspberry Pi is works as small clinic in many places[13]. It can be connected to PC and send push notification to mobile phone. Figure 2.8 shows raspberry pi.

Figure 2.8: Raspberry Pi

2.5.2 Programming Environment

Python is used for the interaction between Raspberry pi and the interface[14]. Since the programming in Python is similar to the C language and also user friendly library is available [14]. Moreover, Python is a default programming environment offered by Raspberry pi [19]. Python come with many packages. It has some packages for GUI which is PyQT and Tkinter. To achieve large customization for PyQT, it utilizes styles as an alternative of theme while for Tkinter, it is not easy to utilizes theme or style for customization. PyQT has runtime (QDocWidget to allow user to freely move an reposition widget) while Tkinter does not have any runtime [22].

Java ME is also a popular programming used in raspberry pi[21]. It indicates itself in all its glory on platforms with weak computing power and small memory resources that work with network services[21]. It is also a free development tools and have a special API to access the periphery. The latest release of Java ME 8 includes a powerful API for controlling devices such as LEDs, relays, LCDs, sensors, motors, and switches [21].

2.6 Critical Review

Current trend, human being would like to have personal health care to increase early awareness on what‟s going on to their health[25]. Heart rate monitoring is one of the famous health care systems and personal belonging is better because it helps to save cost and time[25]. Previously, early models consisted of a monitoring box with a set of electrode leads that attached to the chest and then innovated to a Pulse Oximetry which is it determined blood saturation on measuring the absorption of red and infrared light that passes through a patient's finger or ear lobe by using light sensors[4]. But unfortunately, those method was expensive, painful and risky[15].

We do a lot of works in daily life. Heart is a major organ in the body[1]. Having a personal heart rate monitoring is a better solution to save time and will be alert about what action to be taken to prevent a worse thing to happen.

2.6.1 Reflectance Method

Reflectance method is used to detect the pulse from the arterial pulse[4]. This method is more useful where the vasculature is available close to the surface of skin such as finger, earlobe, forehead, wrist and forearm[15]. Pulse is a rhythmical throbbing of the arteries as blood is propelled through them or in an easy word heartbeat[15]. Pulse can be detected through chest, neck, wrist, finger, earlobe and etc[15]. Nowadays, people like to check their pulse through arterial pulse which is arterial bloods near the skin and they will choose the easiest and the most convenient way. In earlier days, the common method used to measure blood oxygen saturation for pulse detection was arterial blood gas measurement[4]. An Arterial Blood Gas as shown in Figure 2.11 is a blood test that involves puncturing an artery with a thin needle and syringe and drawing a small volume of blood[15]. This method was expensive, painful and risky[15].

Figure 2.11: Arterial Blood Gas

After that, people use a monitoring box. It is a set of electrode leads that attached to the chest[16]. A monitoring box as shown in Figure 2.12 in more likable as it is not painful but is quite troublesome because not everybody like to attach something on their chest especially women[16].

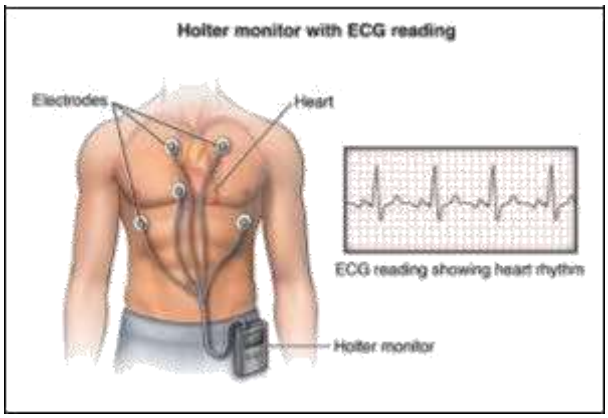

Figure 2.12: Monitoring Box

Oximeter system proposed the idea of using LED to transmit light through the tissue and detected on the other end using a photo-detector which is very interesting as it is one of the easiest way to detect pulse [4]. But the thing here, it is quite annoying to utilize two separate things the moment we want to measure the heart rate. Figure 2.13 shows pulse oximeter.

Figure 2.13: Pulse Oximeter

In this project, reflectance method that used in oxymeter system will be used but LED and photo-detector will be replaced by pulse sensor Amped[4]. Pulse sensor amped will ease the people to get the input of their heart beat with a single touch[4].

2.6.2 Real time chart monitoring.

Real time chart has been using for heart rate monitoring long time ago[12]. Previous work, chart plotting for heart rate used a website called thinkpseak.com used to get the heart rate reading[12]. The channel can be set as private or public as per the requirement. It is an open source API platform[12]. But, to use the private channel, we have to log in first and before we could log in, we need to create an account[12].

In a clinical environment, heart rate is measured under controlled conditions like blood measurement, heart beat measurement, listening to heartbeats using Stethoscope and Electrocardiogram (ECG), but these methods are expensive and cannot be carry by non-experience medical personnel[17]. Figure 2.14 shows Electrocardiogram (ECG).

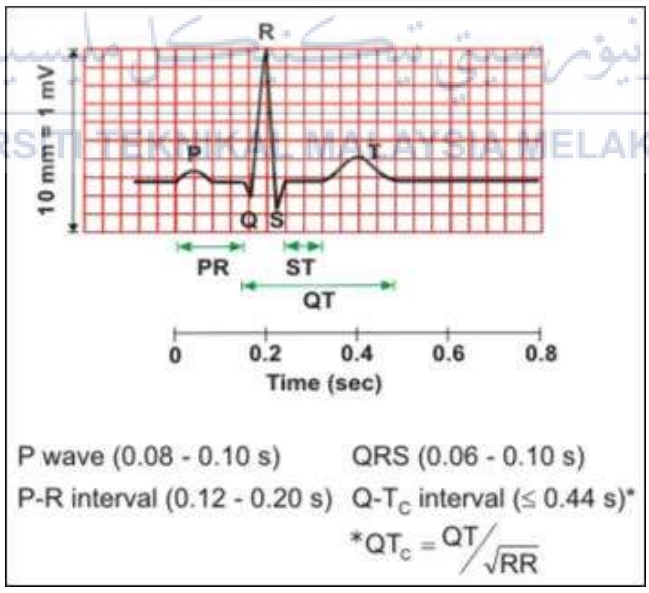

Figure 2.14: Electrocardiogram (ECG)

One of the software used for chart plotting is LabVIEW. It is software in which it is being used as a platform in heart rate monitoring system[6]. In which the

simulation can be done for the pulse sensor and the simulation will also give the value of the pulse obtained[6]. Previous project, NI myRIO is being used to get the signal from the sensor and to give it to the LabVIEW platform and to obtain the waveform[6]. Unfortunately, myRIO is quite expensive[6]. Figure 2.15 shows the hardware of NI myRIO.

Figure 2.15: NI myRIO

Matplotlib also can be utilized to plot graph[18]. It is also stepping with its Pythonic programming style. Matplotlib is a Python package for Plotting that creates production quality graphs[18]. It can be design using a few command in python. The best thing about matplotlib is, it is an open source software and has n licence server[18]. The interface of matplotlib is almost similar like MATLAB[18]. Figure 2.16 shows example of Matplotlib.

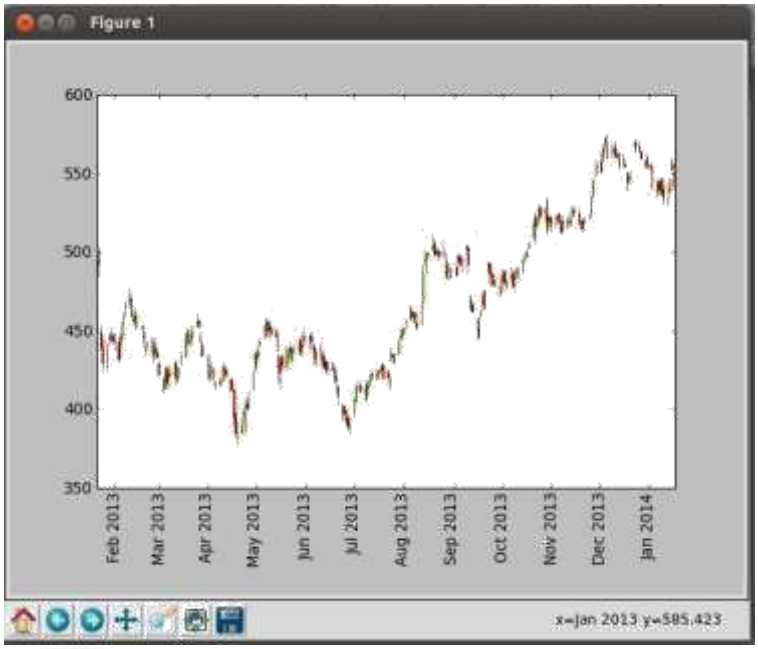

Figure 2.16: Example of Matplotlib Chart

2.6.3 Push notification

Push notification is a short, alert-style message that can be sent to users even when they are not actively using the app[18]. Varies of push notification available. One of them is Telegram[18]. It is a cloud based multi-platform messaging application[18]. The framework reacts to a specific message by setting off an arrangement of guidelines which may incorporate an answer back to the sender and a python program which can further be utilized to take control of the GPIO sticks on the Raspberry Pi[9]. The python program can be used to take control on the signal via GPIO [18]. Telegram is open source software and can be utilized for free. It will function together with Telegram-CLI and Telepot (for Bot) in python packages[21]. Bot usually used to make the telegram interact with raspberry pi[21]. Figure 2.17 shows Telegram application.

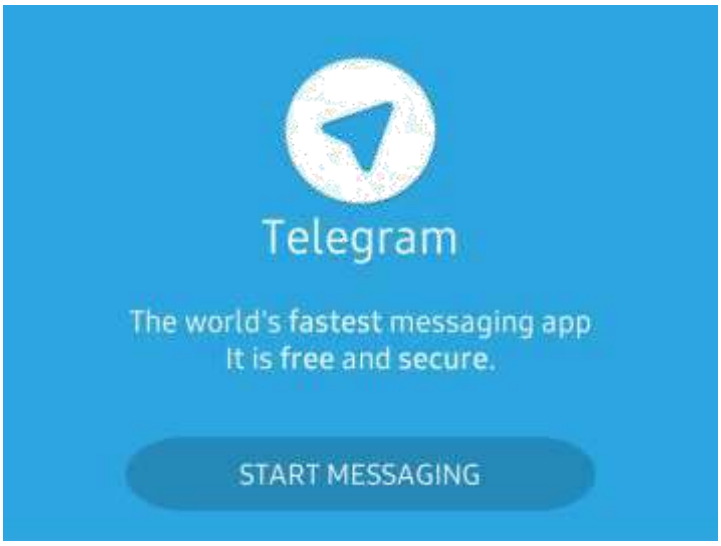

Figure 2.17: Telegram

Another one is Global System for Mobile Communication (GSM). GSM is the worldwide framework for Mobile Correspondence; it is advanced cell phone system. GSM module can send instant message to the required specialists as per the application. GSM is a remote framework which employments TDMA most broadly utilization of advanced remote innovation[20]. Figure 2.2 shows GSM PI SIM900.

Figure 2.18: GSM PI SIM900

2.7 Propose Solution

As proposed in Chapter 1, the best solution would be releasing the method on how to monitor heart rate with a friendly heart rate monitoring product. For this project, we will be focusing on the term „friendly‟ way to measure the heart rate whereas the person who wants to measure the heart rate won‟t feel hurt or uneasy, it can save time and cost and it can be used anytime and everywhere as a personal health care product. Table 2.1 is the review to existing platform:

Table 2.1: Existing Platform Review

Product/project Name	**Arterial Blood Gas (ABG)**	**Monitoring Box (MB)**	**Pulse Oximeter (PO)**
Advantage	Percentage of arterial blood saturated with oxygen helps to determine the effectiveness of a patients respiratory system[15].	Personal monitoring device that allows a subject to measure their heart rate in real time or record their heart rate for later study[16].	It is non-invasive and based on measuring the absorption of red and infrared light that passes through a patient's finger or ear lobe by using light sensors[13].
Disadvantage	This method was invasive, expensive, difficult, painful and potentially risky[15].	Electrode leads need to be attached to the chest[16].	Pulse oximeter device available in market are high pricing[13]. The prototype use two different element which is photo detector and LED to gain input.
Improvement can be made	• Use a pulse sensor instead of using syringe and needle/Electrode leads/Photo detector and LED. • Detect the pulse through other arterial pulse instead of chest and blood. • Push notification using Telegram		

In this project, heart rate monitoring is based on IoT where raspberry pi will be used with the default operating system which is Raspbian, a Linux-based OS and python programming will be used for the interaction of raspberry pi and the interface.

Below is the main overview of the whole project.

i. Pulse sensor is a hardware component that will be used to gain the input from the arterial tissue which is finger.
ii. Software is divided into two tasks which is data capturing and charting.
iii. The data is captured from the pulse sensor via ADC chip and will be converted into equivalent pulse.
iv. Whereas for charting, the captured data will be plotted to Matplotib chart.
v. Once the data have been captured and plotted, push notification using Telegram will be sent to the mobile phone for the user to check their heart rate, symptom and precaution steps.

New enhancement that will be made in this project is based on the improvement can be made which is stated in table 2.1. The proposed method is very friendly because it is easier, safer and inexpensive to be used as a personal health care.

2.5 Conclusion

Basically, this chapter is about the literature review of the whole research. The citations of this literature review are collected from various sources which are online journal, magazine, news and books. Prior from this research, the solution of this project can be proposed. The proposed solution methodology will be elaborated

more on the next chapter. The next chapter will be thriving on the methodology that is used in this project.

CHAPTER III

METHODOLOGY

3.1 Introduction

This chapter will discuss the detail explanation of methodology that used to make this project success and complete. Many methodologies from this field mainly have it reference for others to take advantage and improve as upcoming studies. The method is used to achieve the objective of the project that will accomplish a perfect result. In order to evaluate this project, the methodology based on this project generally about pulse sensor amped for friendly healthy pulse heart beat sensor.

3.2 Methodology

Figure 3.1 is the block diagram for project methodology which is used for this product to make sure the design and the development of a prototype is user friendly and the development of a prototype are able to send alert notification to user.

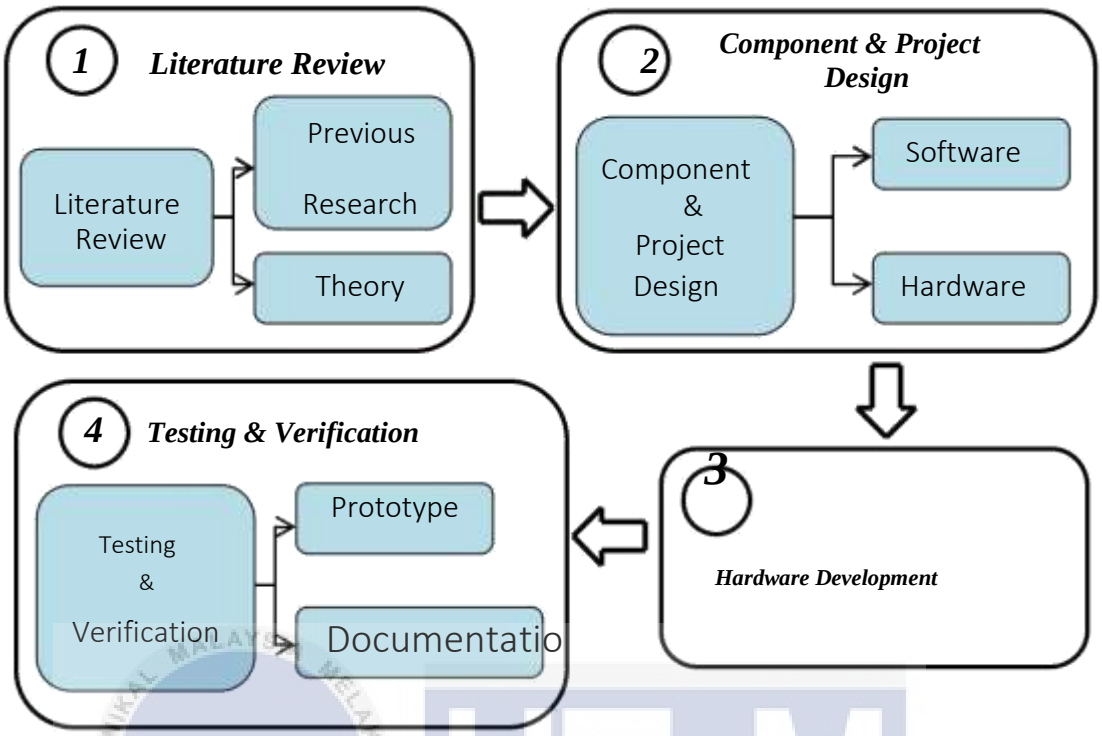

Figure 3.1: Blog diagram of project methodology

3.2.1 Phase I: Literature Review

A literature review is a text of a scholarly paper, which includes the current knowledge, including substantive findings, as well as theoretical and methodological contributions to this project. These chapter reviews of articles, books and journals to understand the concept that needs to know in order to complete this project. There are two category needed for the first method which are previous research and theory.

1. Previous Research

In the first category which is previous research, 26 journals read as reference to gain information about this project including hardware, software, programming environment, and method.

The journals are varies is field. Some of them are about health, IoT, electronic and so on. The purpose of referring to different field of journal is to make sure all the information is enough to start this project. Also, by correctly referencing, it allows the marker or reader to follow-up the references and to check the validity of the arguments for themselves by doing citation for each paper that has been read.

2. Theory

In the second category which is theory, all the ideas from the previous research used to justify what should be done for this project. Comparison between the ideas stated in the previous research has been made to make sure this project accomplish its objectives. Some enhancement also has been made after the comparison to make sure this project come out with a better result.

Research is usually being done to get requirement specification. It is vital and needed in order to understand the problem on how to get the heart beat measurement and how to implement a better solution with improvement from the previous product. In this project, the research was done by doing a literature review from the journals. A research method states that what is being built and in this project, Many journals from different field used as references.

3.2.2 Phase II: Component & Project Design

After the literature review is completed, the component will be chosen and design will be visualized based on what we will do to get the output. Inside the component, there are two categories which are hardware and Software. The important part is to make sure the design is based on the problem and the objectives.

It is also a guideline or key performance to make sure our progress is fulfilled. The design phase means establish the plan for how it should be built.

1. Component

The component (software & hardware) being elected based on the literature review by considering the pros and the cons which suited the objectives. In chapter II (Literature Review), there are many components revealed together with the description. Figure below shows the hardware and the software chosen for this project together with their features and advantages.

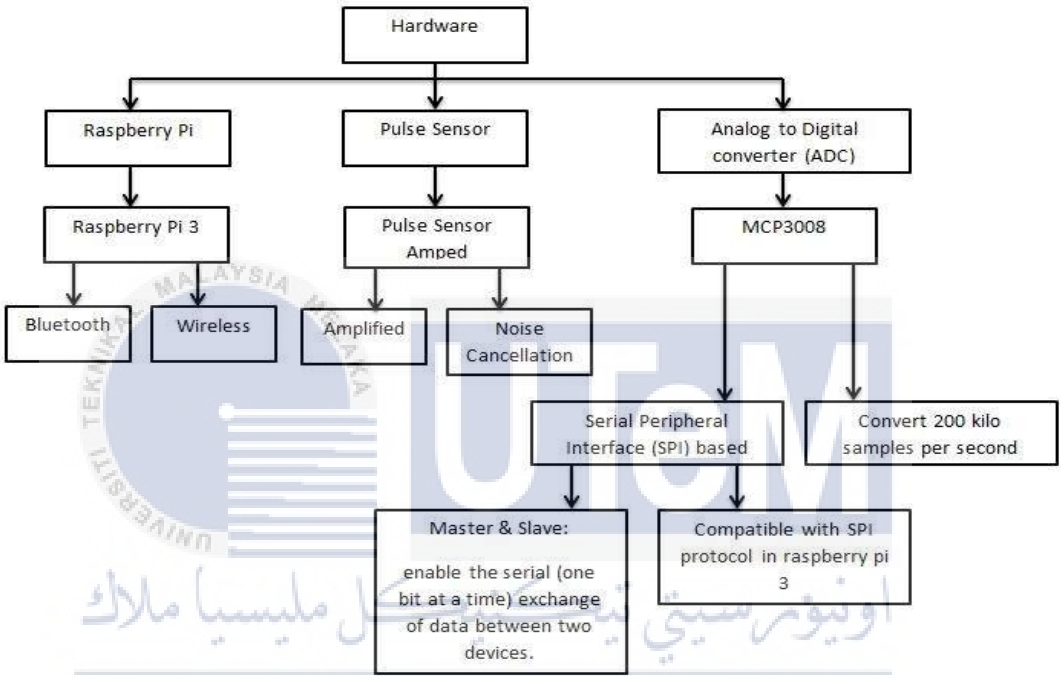

Figure 3.2: Feature & advantage of hardware

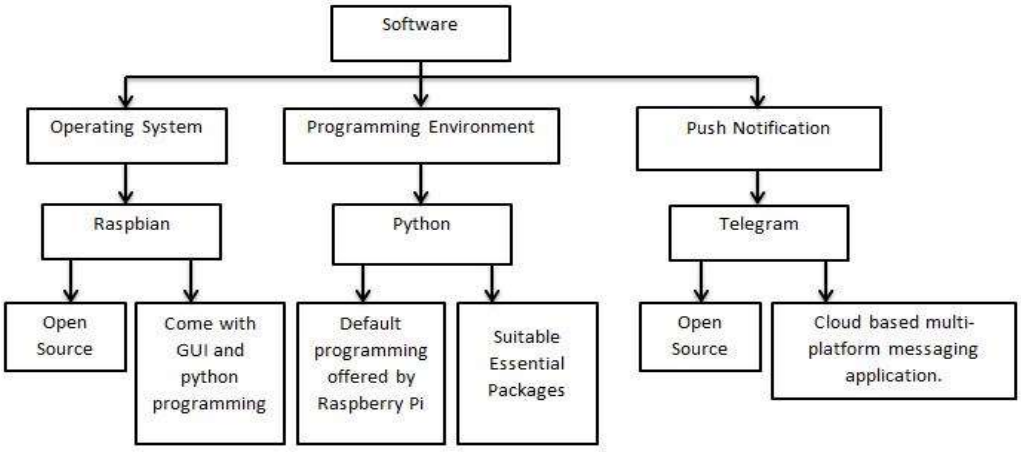

Figure 3.3: Feature & advantage of software

2. Project Design

Project design shows the overview of the prototype. It will be visualized in the form of fritzing schematic for the hardware. The fritzing schematic will be shown and discussed in details in the next chapter which is Chapter IV. For the software, it will be implemented after the prototype (hardware) is ready (exactly like the fritzing schematic) accept for the Raspbian because it is installed for the raspberry pi only. Figure 3.4 below shows the flow of project design. The design will be explained more in chapter IV (Analysis & Design).

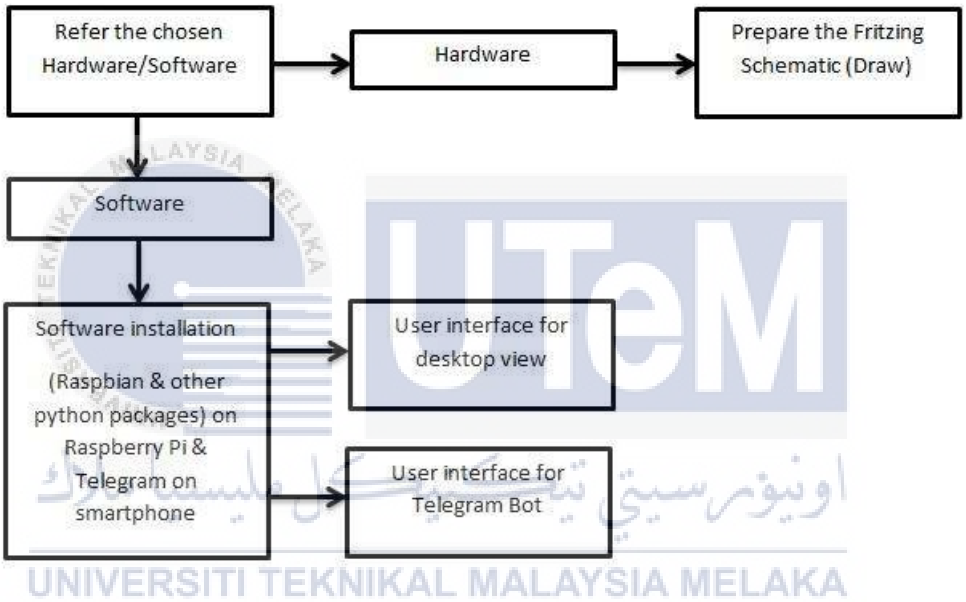

Figure 3.4: Flow of project design

3.2.3 Phase III: Hardware Development

In this phase, all the selected components and the design are ready. The prototype of Friendly Healthy Pulse Heart Beat Sensor (Friendly HePuHebSor) will start to be developed. Hardware will be developed according to the fritzing schematic. After that, the software will be implemented. Software divided into two tasks which are data capturing, charting and push notification. Figure below shows the flow of hardware development.

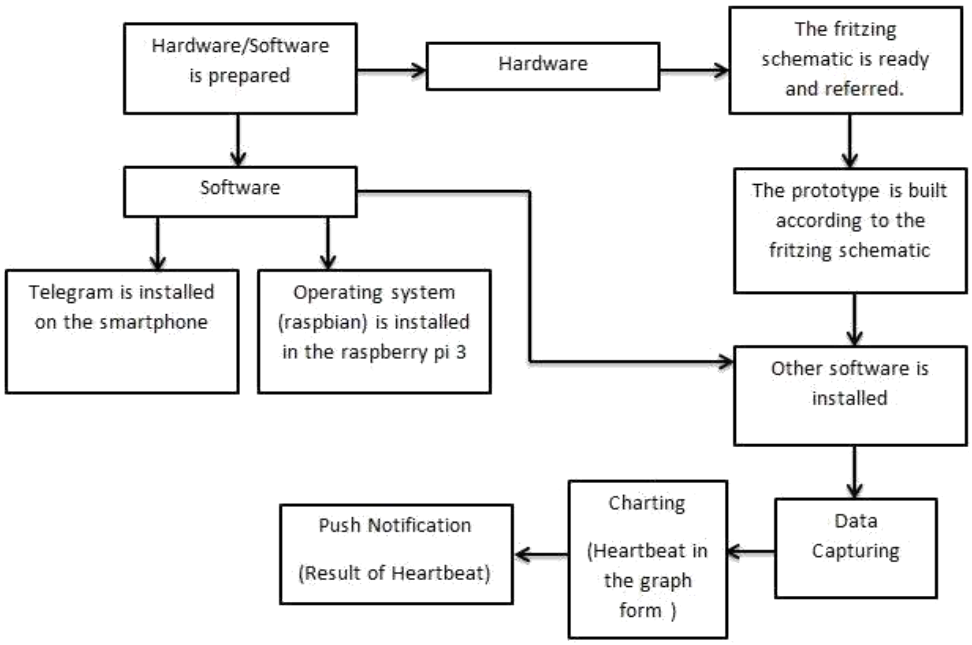

Figure 3.5: Flow of hardware development

3.2.4 Phase IV: Testing & Verification

There are two categories in this phase which is prototype and documentation. After the prototype is ready, the prototype will be tested to ensure that the project fulfils the requirement and objectives and it is working as is should be. To verify that the project is done it needs to be documented in the form of report.

3.3 Project Milestones

Extend Milestone is the exercises alongside the dates of different tasks that should be finished in a project. In this project the Gantt chart is the graphical portrayal of the project point of reference, it empowers the project developer to monitor the due date of undertakings and guaranteeing the project should be possible in the course of events that has been set. The motivation behind Gantt chart is to guarantee that the venture is produced from the earliest starting point to the end. Each of the procedure in a Gantt chart prompts another procedure.

Figure 3.6 below show the Gantt chart of the project and Table 3.1 show the milestone of the project.

Figure 3.6: Gantt chart of the project

	Activity	**Note / Measures**
1 13 - 17 Feb	Submit & Present PSM Proposal	Deliverable – **Proposal** Action – Student
		Deliverable – **Proposal Presentation** Action – Student
	Evaluation and verification of proposal Upload approved proposal to e-Repository System	Action – Supervisor, Evaluator
2 20 Feb - 24 Feb	Correction / Enhancement Proposal Chapter 1	Action – Student
	Supervisor list/PSM Title	Action – AJK PSM/PD

3 27 Feb - 3 Mar	Chapter 1 (System Development Begin)	Deliverable – **Chapter 1** Action – Student, Supervisor
4 6-10 Mar	Chapter 1 & 2	Action – Student
5 13 - 17 Mar	Chapter 2	Action – Student
6 20 Mar -24 Mar	Chapter 2 & 3	Deliverable – **Chapter 2** **Progress Presentation 1** Action – Student, Supervisor
	Determination of Status Students Students borrow equipment for development stage	Action – AJK PSM/PD, Supervisor
7 27 – 31 Mar	Demo Chapter 3, Chapter 4	Action – Student
8 3 - 7 Apr	**Mid Semester Break**	
9 10 - 14 April	Demo Chapter 4	Deliverable – **Chapter 3** Action – Student, Supervisor
10 17- 21 April	Demo Chapter 4	Deliverable – **Progress Presentation 2** Action – Student, Supervisor
	Student Status	Action – AJK PSM/PD, Supervisor Warning Letter 2
11 24 - 28 April	Demo PSM Report	Action-Student
	Determination of student status(Continue/ Withdraw)	Action –PSM/PD Committee, Supervisor(submit student status to AJK)
12 1 – 5 May	Demo PSM Report	Action – Student, Supervisor, Evaluator
13	Table Presentation	AJK PSM/PD
8 -12 May	Demo PSM Report	Deliverable PSM Report Action – Student, Supervisor

14 15 - 19 May	Project Demo & PSM Report	Deliverable – PSM Report Action – Student, Supervisor
15 22 -26 May	**Final Presentation**	Action – Student, Supervisor, Evaluator
16 29 May - 2 June	**STUDY WEEK** Correction draft report based on supervisor‟s and evaluator‟s comments during the final presentation session. Submission overall marks to PSM/PD committee.	Action – Student, Supervisor, Evaluator AJK PSM/PD
5-18 June	***FINAL SEMESTER EXAM***	

Table 3.1: Milestone of the project

3.4 Conclusion

This chapter explain more details about design and the method used for Healthy Pulse Heart Beat Sensor. In the next chapter, will be model development which is the further process for the system must be done well to achieve objective.

CHAPTER IV

ANALYSIS AND DESIGN

4.1 Introduction

In this chapter the design of the project that will be used in the next chapter which is implementation will be deliberated thoroughly. The methodology had been explained in the previous chapter and it will be adopted for the development of this project. The detail of project requirement regarding the software and hardware of this project will be described. In this chapter we also will explain about the overview of the prototype which is hardware design and the interface which is software design.

4.2 Problem Analysis

In this sub-chapter we will be deliberating about the Friendly HePuHebSor itself. It is important to know the process of how it will work. But before that, we will be discussing about the current heart rate monitoring tools. The problem statement described in the first chapter is heart rate monitoring tools are quite expensive, painful, risky and not really friendly to be used. Due to this problem, this prototype is designed for adults and kids to ease them to carry out daily work because heart rate is often influenced by physiological factors like the circadian cycle, posture, blood pressure, and physical activity. It is a better solution to save

time and will be alert about what action to be taken to prevent a worse thing to happen. This product can be used by everybody instantly and everywhere.

4.2.1 Architecture of heart rate monitoring tools

For this project, there are three different current heart rate monitoring tools selected as references to develop the prototype of Friendly HePuHebSor which is Arterial Blood Gas (ABG), Monitoring Box(MB) and Pulse Oximeter (PO). Figure below show the architecture of heart rate monitoring tools.

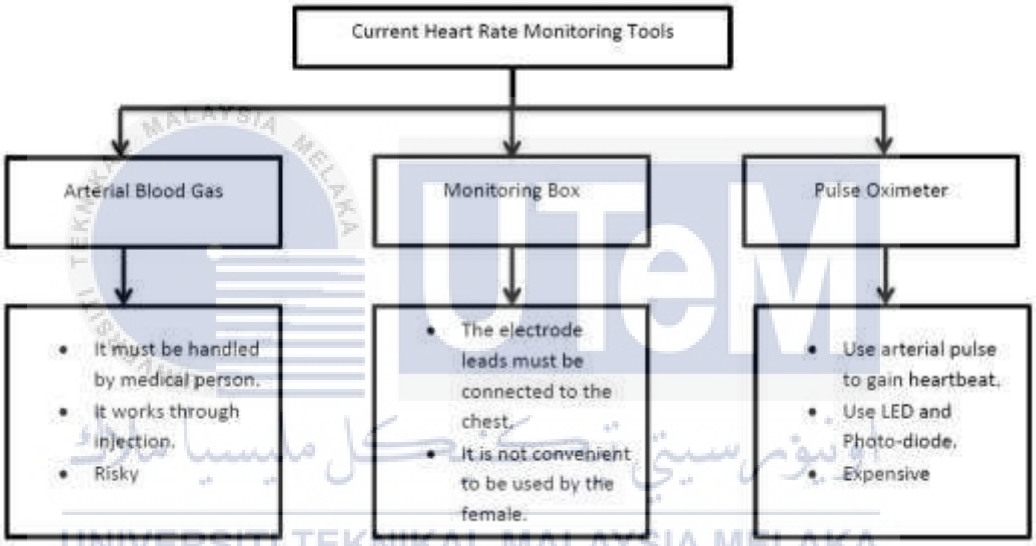

Figure 4.1: Architechture of heart rate monitoring tools.

4.2.2 Architecture of Friendly HePuHeBSor

In chapter I, the problem statement has been mentioned which is heart rate monitoring tools are quite expensive, painful, risky and not really friendly to be used. Acording to the architechture above, All the selected heart rate monitoring tools seem to have similar problem detailed in the problem statement. Pulse Oximeter (PO) give the impression to be the most likable tool to be used for heart rate monitoring. Based on the Pulse Oximeter tool, the idea of measuring the heart rate through arterial pulse is quite agreeable because it‟s easy and friendly. Therefore, the theory used for pulse oximeter in monitoring the heart rate will be utilized in this

project with some improvement. Figure 4.2 shows Architechture of Friendly HePuHebSor.

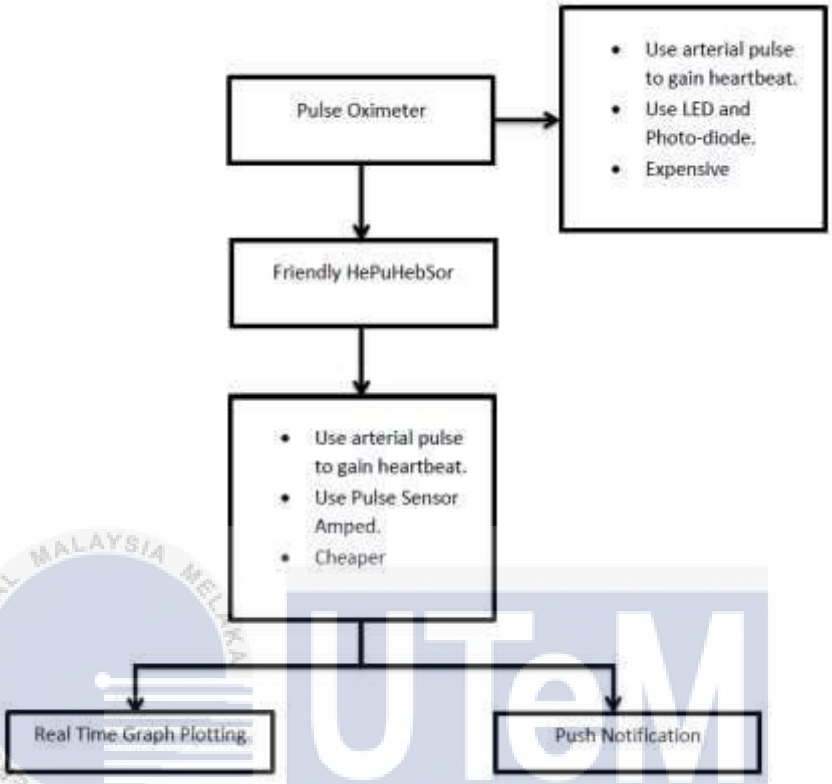

Figure 4.2: Architechture of Friendly HePuHebSor

4.2.3 Proses of Friendly HePuHeBSor

Flow chart or flow diagram is a series of symbols that are used to describe the step by step flow of a process, system, project or others. The following are the flow chart that is describing the flow of how Friendly HePuHebSor works. Since the prototype will be connected to the dekstop/computer and there will be interconnection between the computer and mobile phone therefore, there are two flowcharts used to show the whole process of Friendly HePuHebSor.

Figure 4.3 above shows the flowchart of Friendly HePuHebSor as the whole overview. In this flowchart, it will cover start from the user put the finger on the sensor until push notification text send to the user.

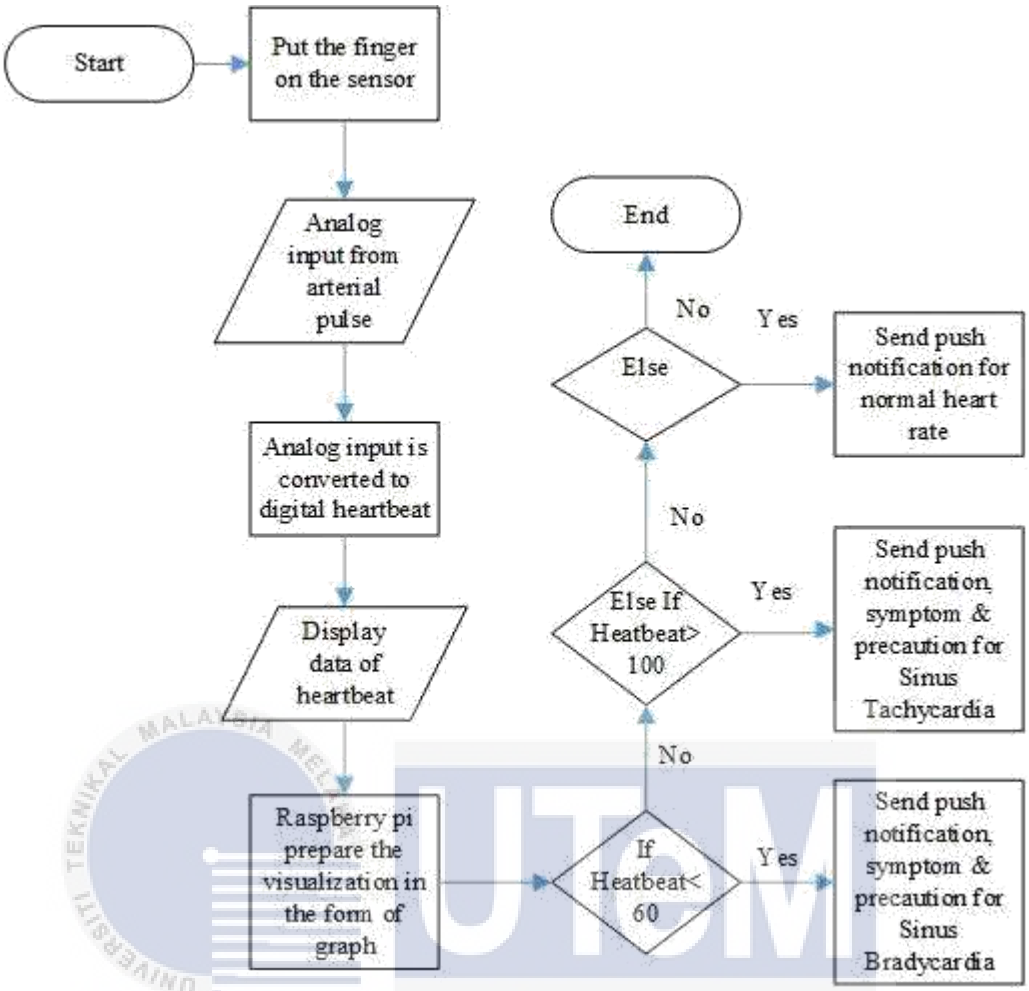

Figure 4.3: Flow chart of Friendly HePuHebSor

Figure 4.4 shows the sub-flowchart of the Telegram Bot for Hepuhebsor Result. This sub-flowchart will show how the telegram work for Hepuhebsor Result in mobile phone. This flowchart is the extended explaination of the prosess of send push notification from Figure 4.3.

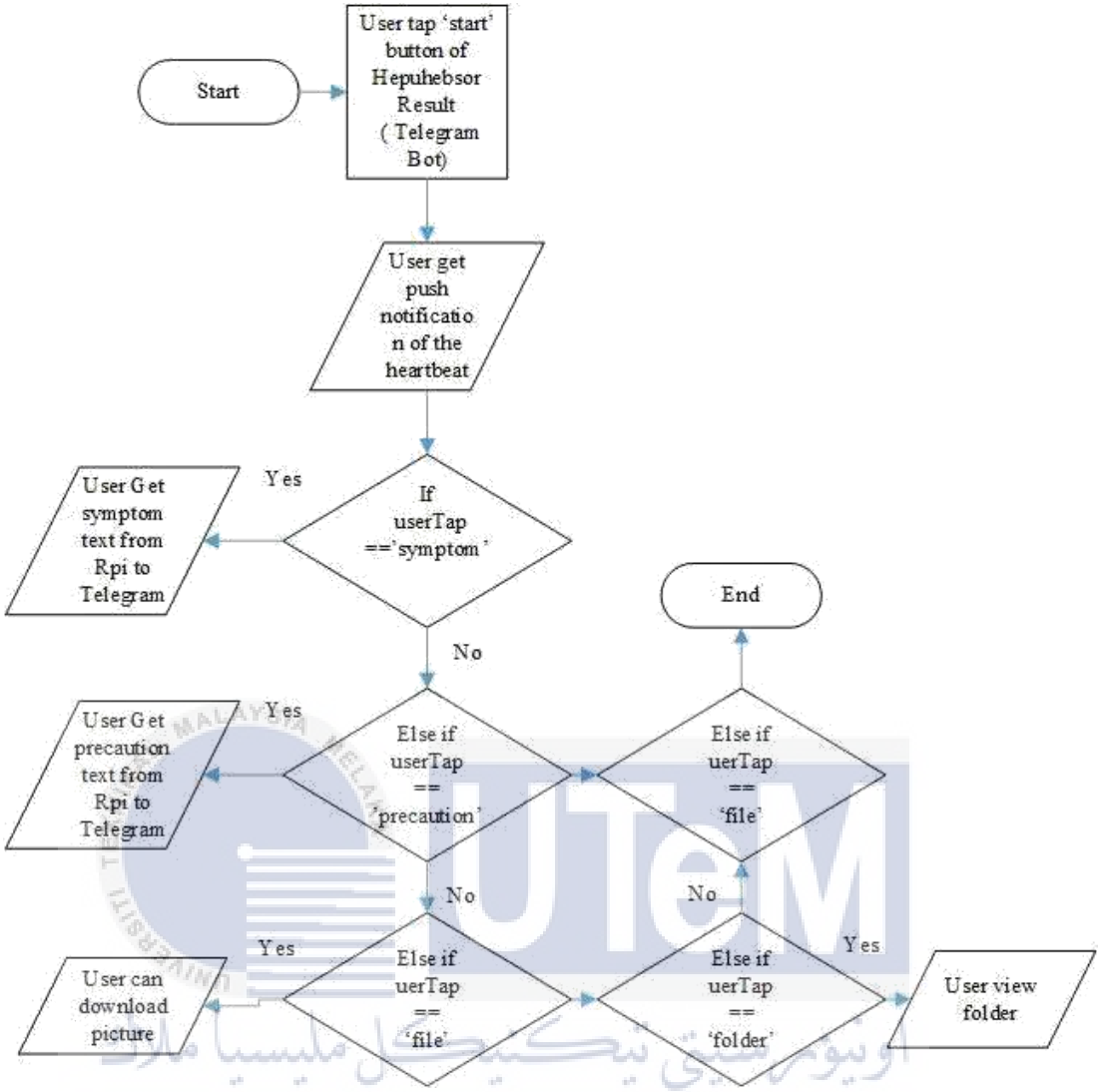

Figure 4.4: Sub-flowchart of Telegram Bot for Hepuhebsor Result

4.3 Project Requirement

Each project needs a project prerequisite to help the advancement of the project, and to guarantee that the project will run easily all through its improvement. With respect to this project, we have accumulated the most reasonable hardware and software prerequisites so we can guarantee this project will run easily.

4.3.1 Hardware requirement

To build the prototype of the the project, There are some main hardware needed. The hardware must fullfill the requirement to get this project done.

4.3.1.1 Raspberry pi 3 Model B

For this project we will be using raspberry pi 3 model B as the processor. It can be used as a cheap computer for some basic functions, especially for experiments and education and projects. Figure 4.5 shows the appearance of Raspberry Pi 3 together with the specifications stated in Table 4.1 .

Figure 4.5: Raspberry Pi 3 Model B

Table 4.1: Specification of Raspberry Pi

Specification	**Details**
CPU	Quad Cortex A53 @ 1.2GHz
GPU	400MHz VideoCore IV
RAM	1GB SDRAM
Storage	Micro SD
Ethernet	10/100
Wireless	802.11n/Bluetooth 4.0
GPIO	40 pins

Raspberry Pi 3 model B has 40 pins GPIO. In this project only several pins involved. Figure 4.6 shows the available GPIO pins and the used one. All the function of the GPIO pins that will be used for this project will be explained later in the proposed design.

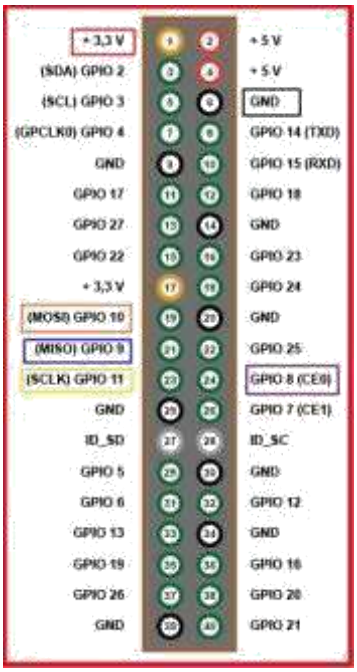

Figure 4.6: GPIO Pins

4.3.1.2 Pulse sensor Amped

Pulse sensor amped is a sensor which is used to gain the analog input from the arterial pulse. It is an optical-based heart rate sensor. It has LED, light sensor and flat colour coded ribbon cable (signal, 3v to 5v input/power and ground). Pulse Sensor Amped also made up of special feature such as amplified (stronger signal with the same waveform) and noise cancellation (reduce the unwanted sound) to make it fast and easy to get the pulse reading. Figure 4.7 shows the front and back side of pulse sensor amped together with the components. Table 4.2 shows the function of major component in pulse sensor amped which is very useful for this project.

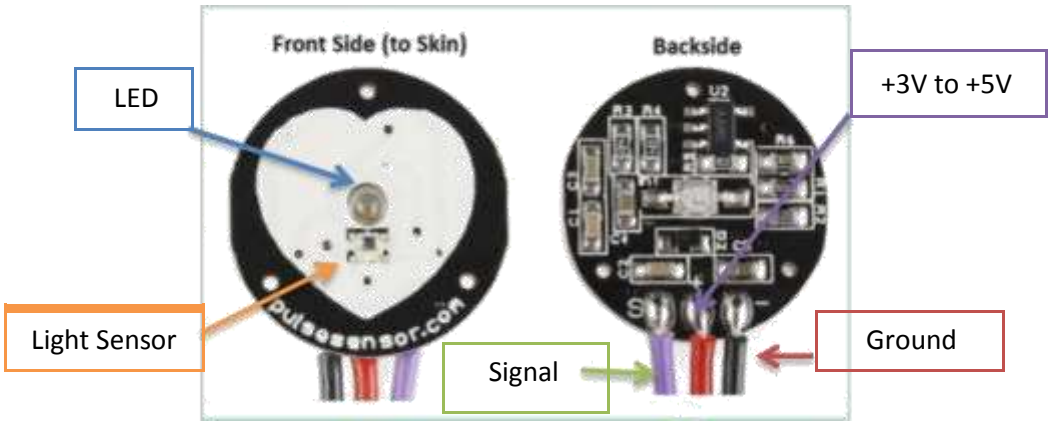

Figure 4.7: Pulse Sensor Amped front and back view

Table 4.2: Specification of Pulse Sensor

Front Side (to skin)		**Backside**	
	Function		**Function**
LED	Transmit infrared light to the finger. • Wave is produced upon contraction of the heart and pulse pressure will increase during systolic. When the infrared light is transmitted, the light will bounce back.	Signal	• To get the heartbeat. • It will be connected to channel 0 (CH0 from the MCP3008) to be processed from analog to digital signal.
Light Sensor	It read the amount of light that bounces back. • The intensity of reflected light depends upon the blood volume inside the fingertip. • This little change in the amplitude of the reflected light can be converted into a pulse (Analog).	+3V to +5V	• Input/Power.
		Ground	• Prevent shot circuit.

4.3.1.3 MCP3008 (ADC)

MCP3008 is a type of Analog to Digital Converter (ADC). It will be utilized to get the heartbeat data. In this project, the Serial Peripheral Interphase (SPI) in raspberry pi 3 will be enabled for the usage of MCP3008.

The Serial Peripheral Interface (SPI) is a communication protocol used to transfer data between the Raspberry Pi and peripheral device (MCP3008) as an analog to digital converter. SPI bus (from raspberry pi 3) has 4 separate connections to communicate with the device, Serial Clock (SCLK), Master in Slave Out (MISO), Master out Slave in (MOSI) and Chip Select (CE). The function of CE is to tell the slave to get ready to perform measurement. Once the CE pin goes low, it will start to count the fist bit. The fist bit is used to tell what kind of measurement should be done. There are two types which is single (bit 1) and differential (bit 0) after that followed by another 3 bits to which channel on the slave is going to be used. Measurement of the pulse uses the 10 bits of the slave. MOSI will send data to the slave while MISO send data to the master. MCP3008 is a full duplex so it can send data simultaneously. To do so SCLK is needed to synchronize the data transmission. The speed is determine by the frequency of the SCLK. Figure 4.8 below shows interaction between master and slave.

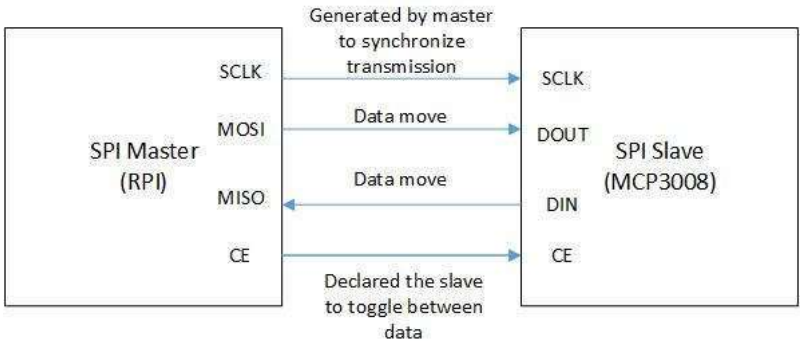

Figure 4.8: Interaction between raspberry pi and MCP3008

For this project, All the coloured PDIP, SOIC, shown in the Figure 4.9 will be utilized. The symbol and the description of each PDIP,SOIC of MCP3008 are shown in the Table 4.3. It will be explained further in the fritzing schematic (hardware design).

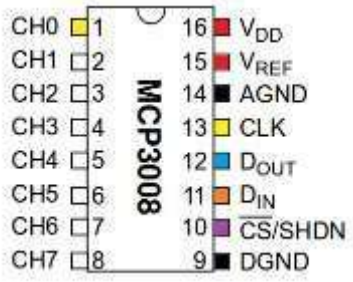

Figure 4.9: PDIP, SOIC of MCP3008

Table 4.3: Description of each PDIP and SOIC

PDIP and SOIC	Symbol	Description
1	CH0	Analog Input
2	CH1	Analog Input
3	CH2	Analog Input
4	CH3	Analog Input
5	CH4	Analog Input
6	CH5	Analog Input
7	CH6	Analog Input
8	CH7	Analog Input
9	DGND	Digital Ground
10	CS/SHDN/CE	Chip Select/Shutdown Input/Chip Enable
11	DIN	Serial Data In
12	DOUT	Serial Data Out
13	CLK	Serial Clock
14	AGND	Analog Ground
15	VREF	Reference Voltage Input
16	VDD	+2.7 V to 5.5 V Power Supply

4.3.2 Software requirement

Software that will be used for this project need to fullfill the requirement because only some software is compatible and needed for this project.

4.3.2.1 Raspbian

Raspbian will act as the operating system to conduct the project because it has a faster booting and shutting down time. It is an open source debian linux based operating system. As we definitely know, operating system is a set of basic programs and utilities that keeps running on a predetermined hardware. Debian is extremely lightweight and settles on an extraordinary decision for the Pi. Raspbian also come with GUI and python programming which is the language that will be used for this project.

4.3.2.2 Python Programming

Python programming is a default programming environment offered by raspberry pi. For this project, python will be used as the language to interact between raspberry pi and the interface. Python also come with many essential packages but not all packages will be used for this project.

4.3.2.3 Essential Packages

Table 4.4 is the essential packages that must be installed with python to ensure the project run smoothly. These packages are installed by using Raspbian‟s repository which is by running sudo apt-get install.

Table 4.4: Essential Packages

Package	**Function**
Python 2.7 and above	For python environment
SPIdev module	A kernel module that can make use of the Raspberry Pi's built in SPI support
Matplotlib	It will be used to plot real-time graph for the user to analyse their heartbeat
Telegram-CLI Telegram-Bot	It is used to send text as a push notification to the user. Communicate with raspberry pi to perform task.
Telepot	It Helps to build application API for Telegram Bot
TKinter	Graphical User Interface (GUI)

4.3.2.4 Telegram

Telegram is open source software which is needed to be installed in the smartphone (Telegram Bot for Hepuhebsor Result) and in the raspberry pi by using the package described in the essential packages. Some programming knowledge needed to use the package which is Telegram-CLI on raspberry pi while for the mobile phone, we need to create a new Bot by searching for BotFather on the Telegram and perform some interaction without using any programming to register the new Bot. From the registration the new Bot will have a new Token. The same Token will be used in the raspberry pi to make the connection. Figure 4.10 shows the interaction between Telegram & Raspberry pi.

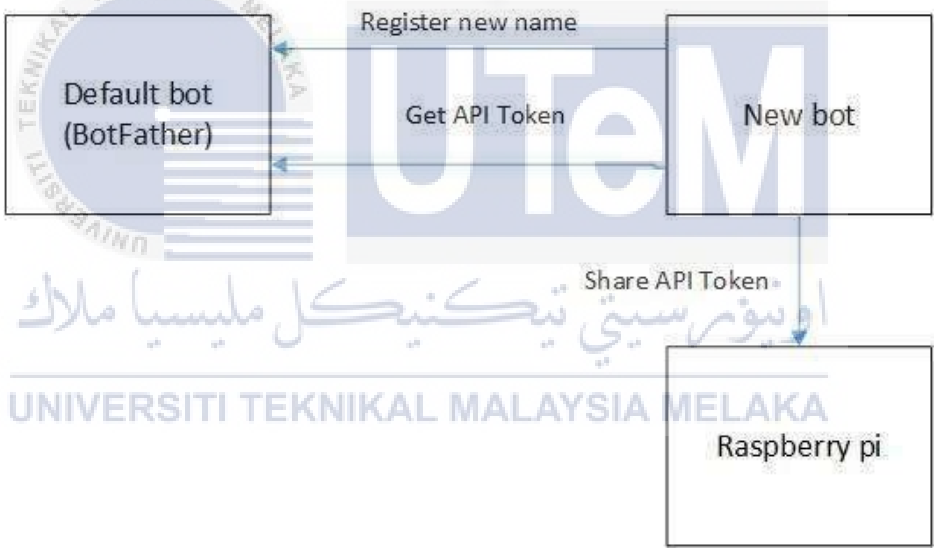

Figure 4.10: Interaction between Telegram & Raspberry pi

4.3.3 Functional Requirement

In this sub-chapter we will clarify about the functional requirement of Friendly HePuHebSor. Conferring to (Zhou 2004), functional requirement is describing the functionality of a project /system such as what function provided by the project. Below are the list of function requirements for this project.

i. Detect analog input that came from the arterial pulse.
ii. Convert the analog input to digital to get the heartbeat data.
iii. Plot the graph based on the heartbeat data to analyse the heartbeat.
iv. Send push notification of heartbeat reading to the user.
v. Give out reporting about symptom and step precaution based on the heartbeat reading.
vi. Send files and pictures from Raspberry pi to Telegram.

4.3.4 Non-functional Requirement

Non-functional requirement is referring to the requirement which is not really concern with the specific functionality of the project. It may relate to project properties. Below are the list of non-functional requirements for this project.

i. Must be able to plot understandable real-time graph.
ii. Must be able to convert analog input to digital signal.
iii. Must be installed with python 2.7 and above to be able to execute the program.
iv. Must be able to send files and pictures from Raspberry pi to Telegram.

4.4 Proposed design

For this chapter, the proposed design is provided. There are two categories for this sub-chapter which is hardware design and User interface design. For hardware design, it will show the overview of the prototype together with the function of each connection while the user interface design will show the design for the executed program of the prototype.

4.4.1 Hardware Design

In this project, fritzing schematic will be used to visualize the prototype. Fritzing schematic is the sketching of the prototype together with the connection. For the schematic, sub-components such as jumper wire (male to male and male to female) and breadboard will be utilized to complete the whole overview of the prototype. Figure 4.11 shows how all the components are connected. Figure below also shows the flow on how this prototype work start from user put the finger on the sensor until push notification send to mobile phone.

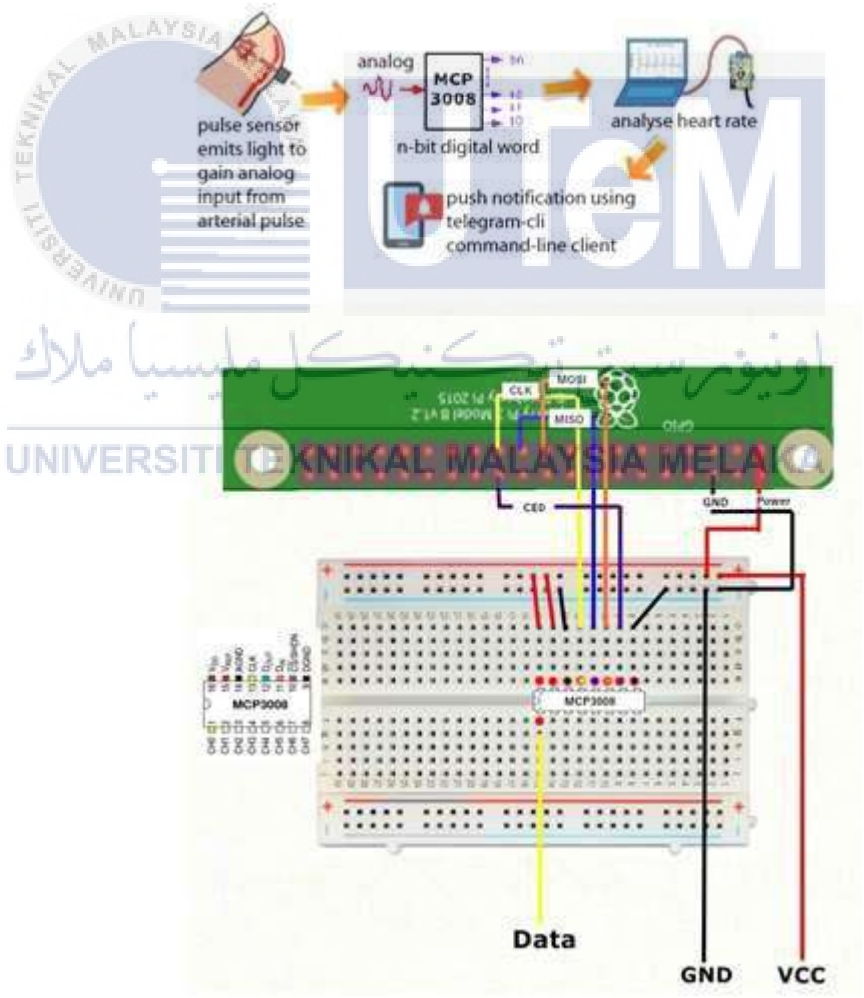

Figure 4.11: Fritzing Schematic of Friendly HePuHebSor prototype

Table 4.4 explain the connection between all the components. For table 4.4, the left column shows what part of Pulse Sensor Amped and MCP3008 should be connected to the GPIO pin (middle column) and the left column is the description /explanation about the connection.

Table 4.5: Explanation of connection between component and GPIO

Pulse Sensor Amped & MCP3008	GPIO pin Raspberry pi 3	Description/Function
• VDD (2.7 v to 5.5v) • V reference • VCC	3.3 v	• The input data from pulse sensor uses 3.3 v • It can be connected to 5.5 v output GPIO but it may damage the pulse sensor. • No resistor used to cut the 5.5 v output.
Analog Ground (AGND) Digital Ground (DGND) Ground (GND)	Ground (GND)	• Prevent shot circuit.
Data Out (DOUT)	SPI-Master in Slave out (MISO)	• Send data from slave to master.
Data In (DIN)	SPI-Master out Slave in (MOSI)	• Send data from master to slave.
Clock (CLK)	SPI-Clock (CLK)	• Act as synchronize bus. • Each bit of data that moving from MOSI and MISO correspond with the clock signal.
Chip Select/Chip Enable (CS/CE)	Chip Enable 0 (CE0)	• Need to be connected because it will toggle between words.

4.4.2 User Interface Design

In this sub-chapter, there are two different user interface design, Friendly HePuHebSor and Hepuhebsor Result. One is for desktop view and the other one is for smartphone. For the desktop view, it will be designed by using PyQT in python

environment while for the smartphone, the default interface for Telegram Bot will be utilized.

1. User Interface Design for Desktop (Friendly HePuHebSor)

User interface design for Friendly HePuHebSor created to ease the user to use it without any programming knowledge. Figure 4.12 shows the proposed design for Home interface of Friendly HePuHebSor. The purpose of the Home interface is to ensure the user know that it is a heart rate monitoring tool. As we can see the „Next‟ button is on the bottom right. It tells the user to go to the next page. The 3 circles tell the user how many page of interfaces and the black circle tells the user which page of interface they are at.

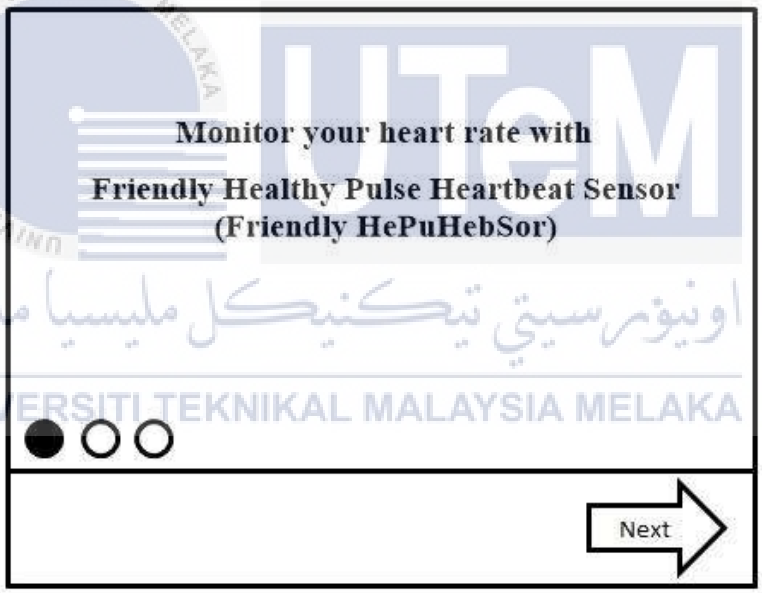

Figure 4.12: Home Interface of Friendly HePuHebSor

Figure 4.13 shows the proposed design for instruction interface. In this interface, there are 2 important instructions need to be followed by the user. On the bottom right, there is a “Next” button and on the bottom left is „Back‟ button for the user to visit the next and the previous page.

Figure 4.14 shows the monitoring interface. In this interface, the real-time graph will be visualized together with the info of beat per minute (bpm) in the oval

shape. The red horizontal line shows the sign of Sinus Tachycardia (fast heart rate). It starts from 100 bpm and above. The green horizontal line represents the normal heart rate. It shows 2 green horizontal lines because the range of normal heart rate is from 60 to 100 bpm. The blue horizontal line represents Sinus Bradycardia (slow heart rate). It starts from 40 bpm and below. The function of the graph is to ease the user to be alert of their heart rate.

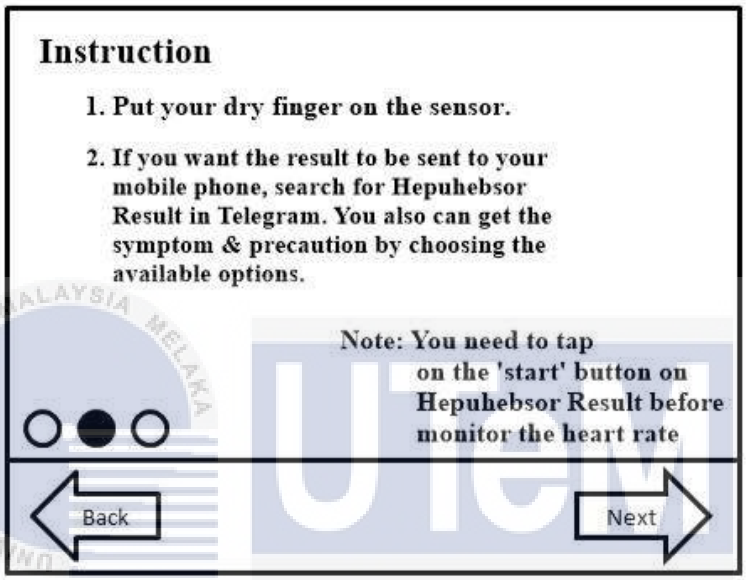

Figure 4.13: Instruction interface of Friendly HePuHebSor

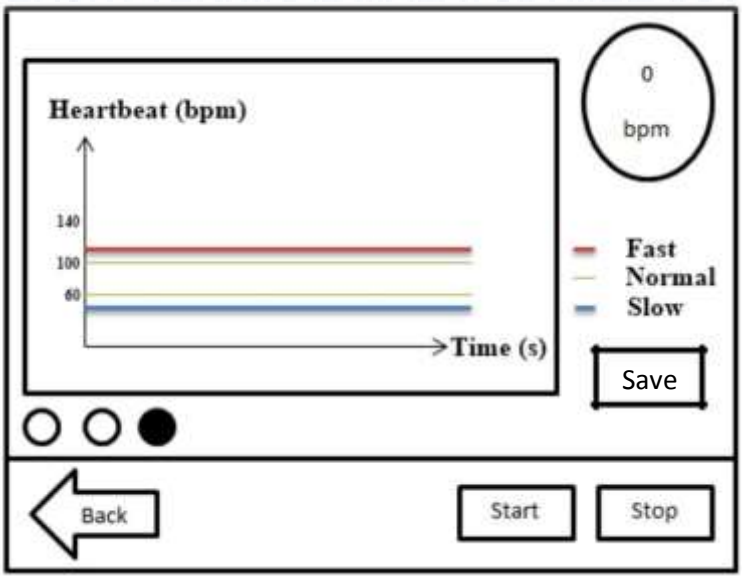

Figure 4.14: Monitoring interface of Friendly HePuHebSor

2. User Interface for Telegram Bot

User interface for Telegram Bot for Hepuhebsor Result on mobile phone is using the default interface of Telegram which means that we do not have to design it. The interfaces below will show how the interfaces will look like after some modification and coding applied to it. For this project, user will be able to download file or picture of their heart rate and get text message using Telegram.

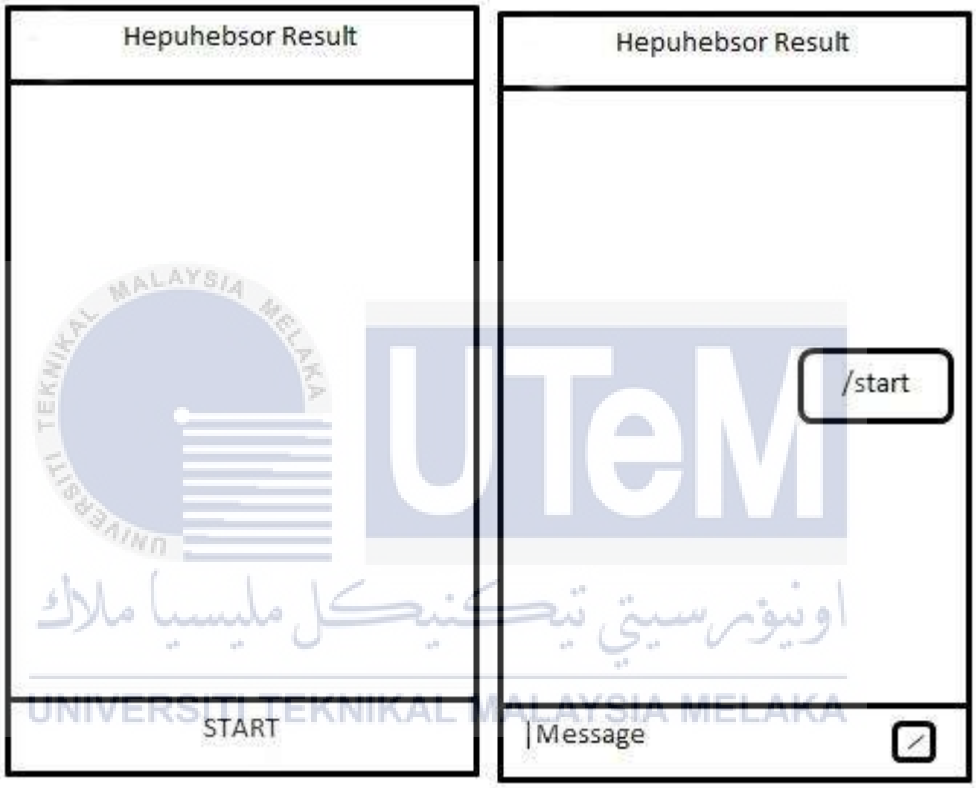

Figure 4.15: Telegram bot of Hepuhebsor Result interface

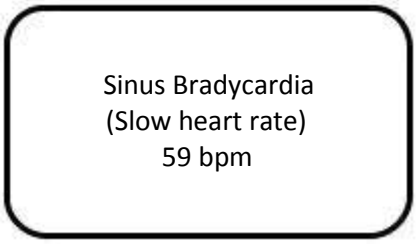

Figure 4.16: Example of push notification (Result of heartbeat)

Figure 4.15 shows the interface of Telegram Bot for Hepuhebsor Result. The left figure shows a “START” button which is needed to be tapped to receive push notification. Push notification from raspberry pi only can be sent after the “Stop” button from the monitoring page is clicked. The right figure is how it will look like after the button „start“ is tapped. Figure 4.16 shows the result for Sinus Bradycardia. It will work the same for Sinus Tachycardia and Normal heart rate.

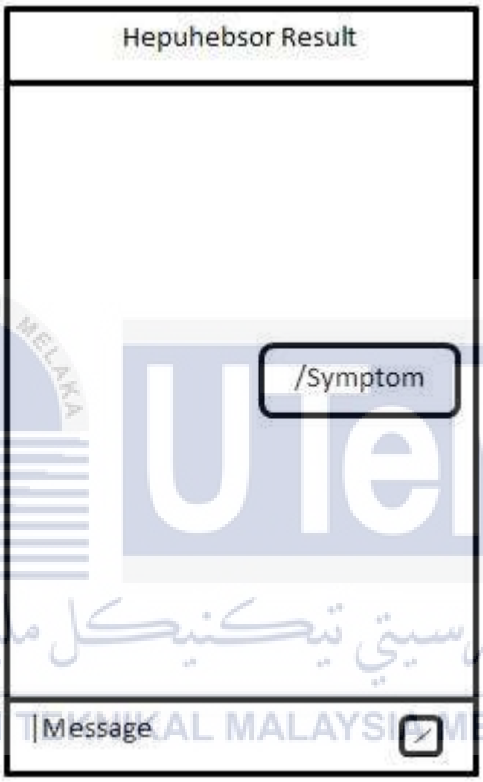

Figure 4.17: Telegram bot of Hepuhebsor Result interface (Symptom)

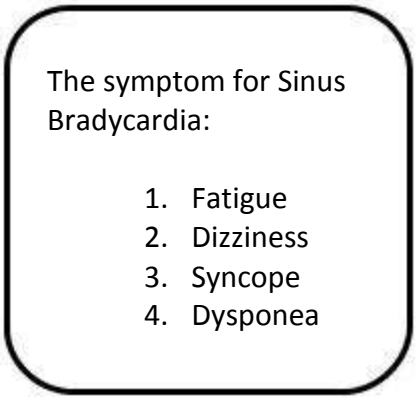

Figure 4.18: Example of text notification (symptom)

Figure 4.17 shows the interface of Hepuhebsor Result on how to get the symptom sent from the raspberry pi. The user tap on the „/" button on the right and choose "/Symptom". Figure 4.18 shows some example of the symptom of Sinus Bradycardia. The symptom is vital for the user to know what is happening and will alert the user to take action before something bad happen in the future. It will work the same for Sinus Tachycardia and Normal heart rate.

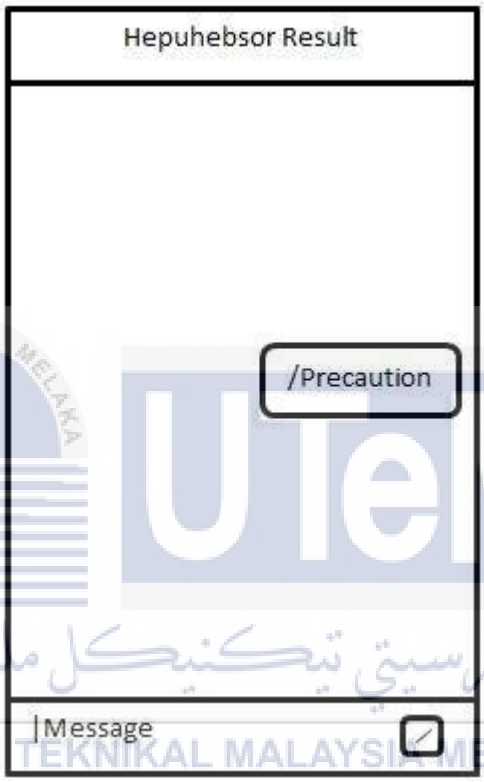

Figure 4.19: Telegram bot of Hepuhebsor Result interface (Precaution)

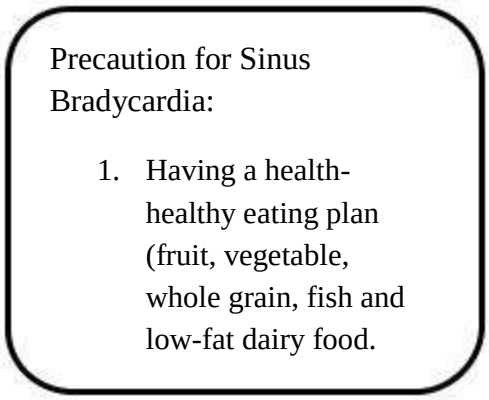

Figure 4.20: Example of text notification (symptom)

Figure 4.19 shows the interface of Hepuhebsor Result on how to get the precaution sent from the raspberry pi It will only be sent from raspberry pi to the smartphone when the user tap on the "/" button on the right and choose "/Precaution". Figure 4.20 shows the example of text notification of precaution. It will tell the user what they should do to prevent the worse thing to happen. Some of the precaution will tell the user the easy steps to be taken. It is a better solution to save time. It also will work the same for Sinus Tachycardia and Normal heart rate.

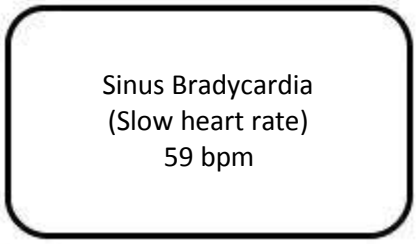

Figure 4.20: Text notification

4.5 Conclusion

In a nutshell, this chapter discuss more on the design of the project. Design is very vital to make sure the implementation can be done smoothly as it gives the overview of the project. It is also a blueprint for a project and the project cannot be carried on without it. The next chapter will be the implementation of the project. The full documentation will be provided including the setup of the prototype, installation, coding and so on.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

The previous chapter explained about the design, in this chapter, we are going to cover the implementation part of the project. All of the software will be implemented and hardware environment will be set up.

5.2 Software Development Environment setup.

In this part, we will briefly explain the environment for the project. This project requires the prototype of Friendly Hepuhebsor where the hardware are ready and the software are installed. The prototype will detect the heartbeat and display the result in the form of csv and graph.

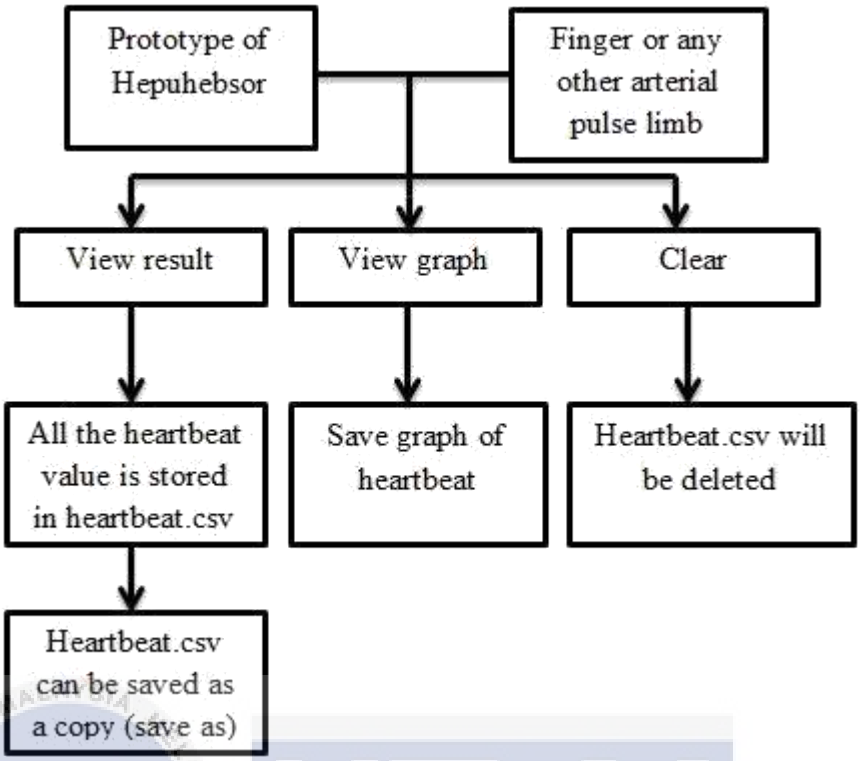

Figure 5.1: Architecture of Friendly Hepuhebsor Environment without internet connection.

Figure 5.1 above shows the architecture of Friendly Hepuhebsor environment without internet connection where the user can save the result of several minutes of monitoring the heartbeat in various forms such as png, jpeg and pdf. The whole result of heartbeat will be stored in a file (heartbeat.cvs), which is the result of heartbeat. The system will clear the result (heartbeat.csv) once the user finished using it but it can be saved as a copy by using save as function . To set up this kind of environment, the existence of the prototype and the finger or any other arterial pulse is a must. To monitor the heartbeat internet connection is not needed but id the internet connection is available, it will still function well.

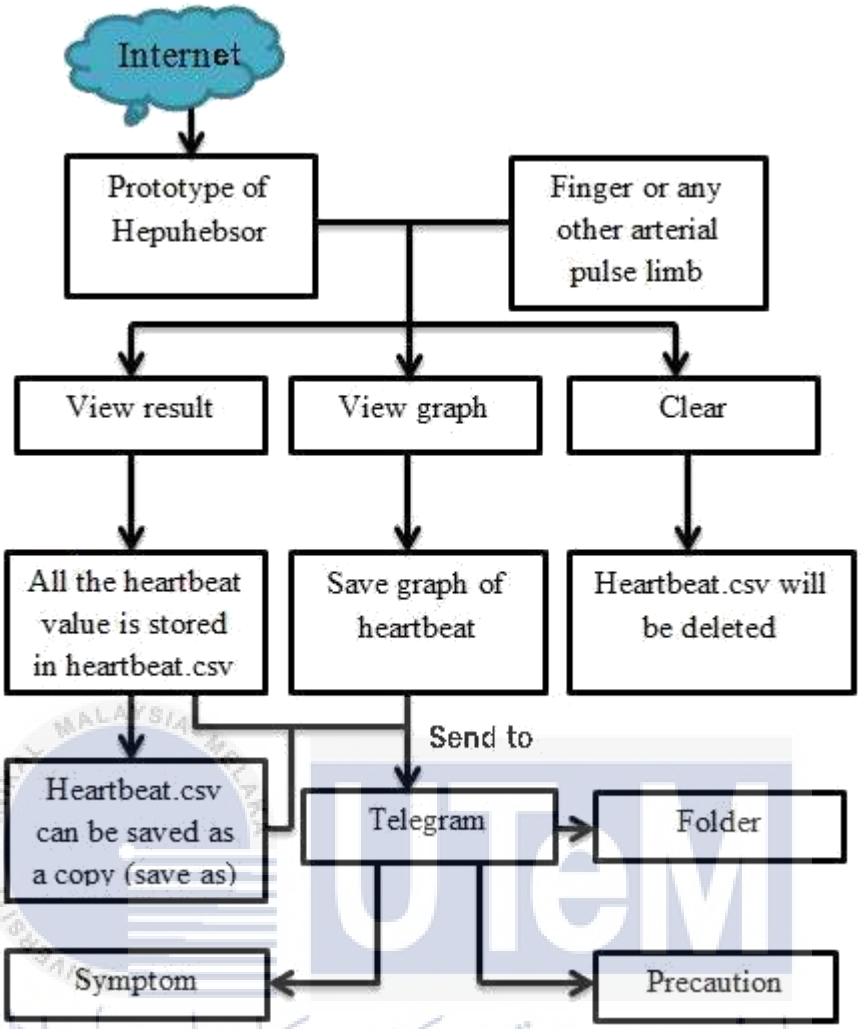

Figure 5.2: Architecture of Friendly Hepuhebsor Environment with internet connection.

Figure 5.2 above shows architecture of Friendly Hepuhebsor Environment with internet connection. The prototype will still do the same like in Figure 5.1 but it will be connected to Telegram. The function of Telegram is to download the files and view the folder, symptom and precaution. To do all of them, the existence of internet connection and the Telegram is a must.

5.3 Software Configuration Management

In this part, we will thoroughly explain about the design and the setup of the configuration management of this project. We will also be explaining the software tools that are used to support our configuration control.

5.3.1 Configuration Environment Setup.

Friendly Hepuhebsor environment setup is done step by steps. All the flowcharts below will explain more about the setup together with the configuration starting from how to make the MCP3008 work to convert analog signal to digital signal for the pulse sensor until how the Telegram able to do the tasks mentioned above on figure 5.2.

1. Development and Configuration (Raspberry Pi)

	Software Implemented
Raspberry Pi	• Noobs (Linux) • SSH (MobaXterm) • IP Scanner (RPi IP Address)

Table5.1 :Software implementation for Raspberry Pi

Table 5.1 above shows the software implemented on Raspberry Pi. Noobs is used as the operating system, MobaXterm as the virtual machiene to run the linux based operating system and IP Scanner is used to get the IP Address (ms.pi.home) of Raspberry pi to be connected as SSH.

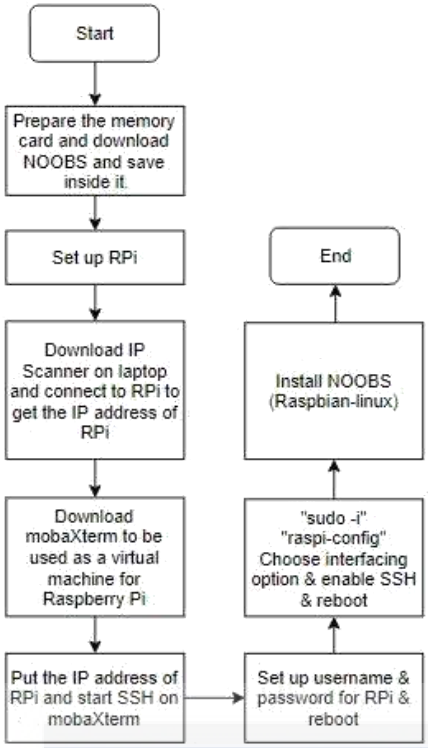

Figure 5.3: Installation of Raspbian on Raspberry Pi.

Figure 5.3 and table 5.1 above show the installation of Raspbian on Raspberry Pi and the software implemented on Raspberry Pi. For this project Raspberry Pi is connected to laptop therefore, SSH is needed. MobaXterm is used to enable the SSH and IP Scanner is used to get the IP address of Raspberry Pi. Raspbian is installed as followed to be used as the server for this project.

2. Development and Configuration (MCP3008).

	Software Implemented
MCP3008	• Spidev

Table 5.2: Software implementation of MCP3008

Table 5.2 above shows the software implementation of MCP3008. Only Spidev is implemented as it is the most compatible with MCP3008 analog to digital converter.

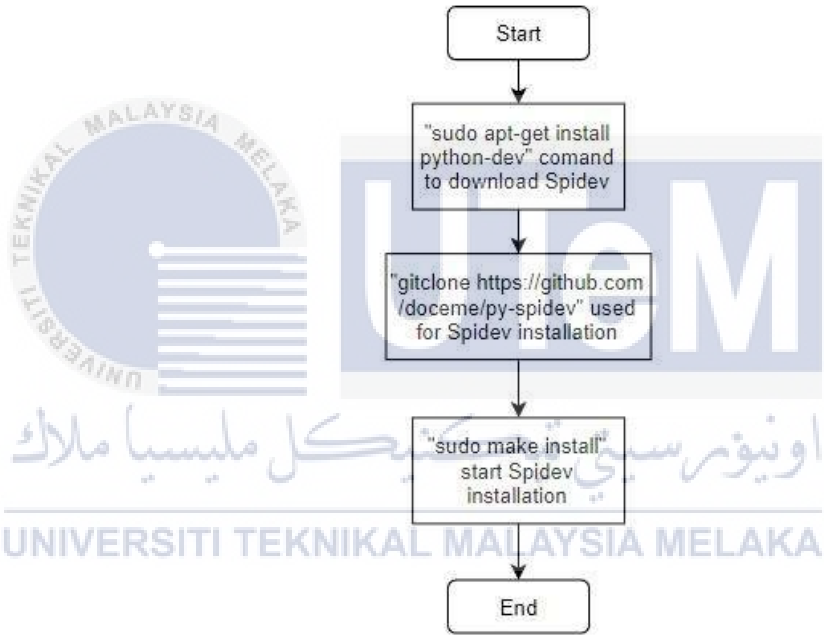

Figure 5.4: Spidev installation for SPI device (MCP3008)

Figure 5.4 above shows Spidev installation for SPI device (MCP3008). Serial Peripheral Interface (SPI) is disable by default and Spidev is SPI driver that need to be installed. Spidev module needs to be loaded before using SPI device. For this configuration, MCP3008 will work as mentioned in chapter IV (Figure 4.8 & Table 4.3). Analog to digital conversion can only work for the pulse sensor after finished the installation of Spidev module.

3. Development and Configuration (Pulse Sensor).

	Software Implemented
hb.py	Python language
	CSV File
	pulsesensor.py
	MCP3008.py

Table 5.3: Software implementation of hb.py

Table 5.3 above shows the software implementation of hb.py. Pyhton is used as the language, pulsesensor.py and MCP3008.py is imported inside hb.py to make a complete function for the sensor. CSV file is used to store the heartbeat data.

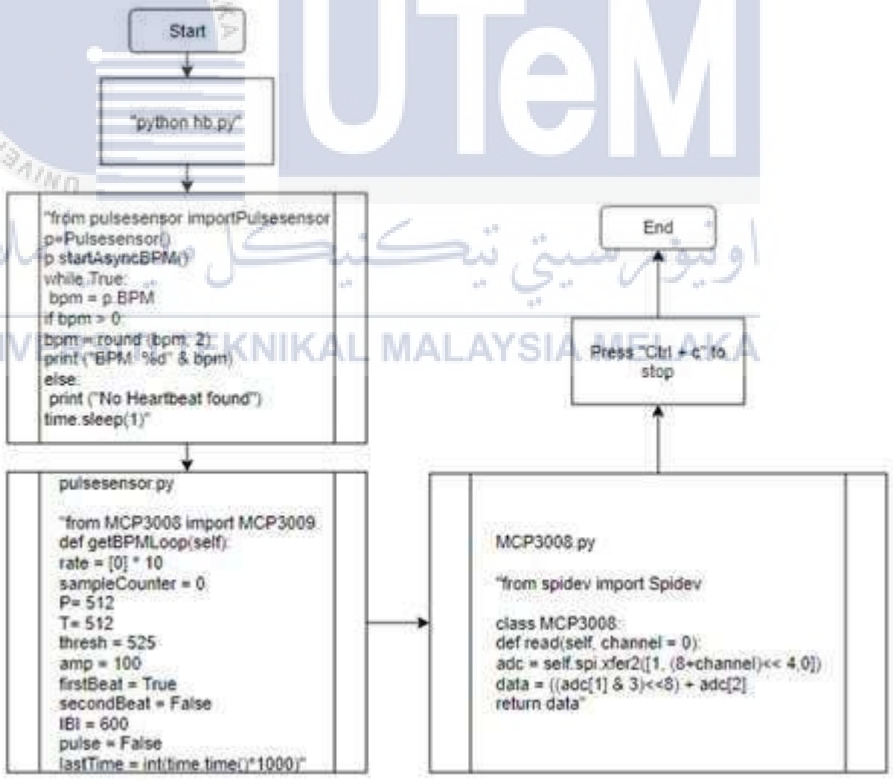

Figure 5.5: Configuration of Pulse Sensor in hb.py

Figure 5.5 above shows the configuration of Pulse Sensor in hb.py. For this part, there are 3 python script needed. Among the 3 python script only one "hb.py"

will be executed. The first python script is MCP3008.py. In this script Spidev that has been installed will be imported and we will tell the Spidev kernel which channel need to be used and it will function exactly like stated in Chapter IV (Figure 4.8). Pulsesensor.py is where the configuration of the sensor stored.

4. Development and Configuration (heartbeat.csv).

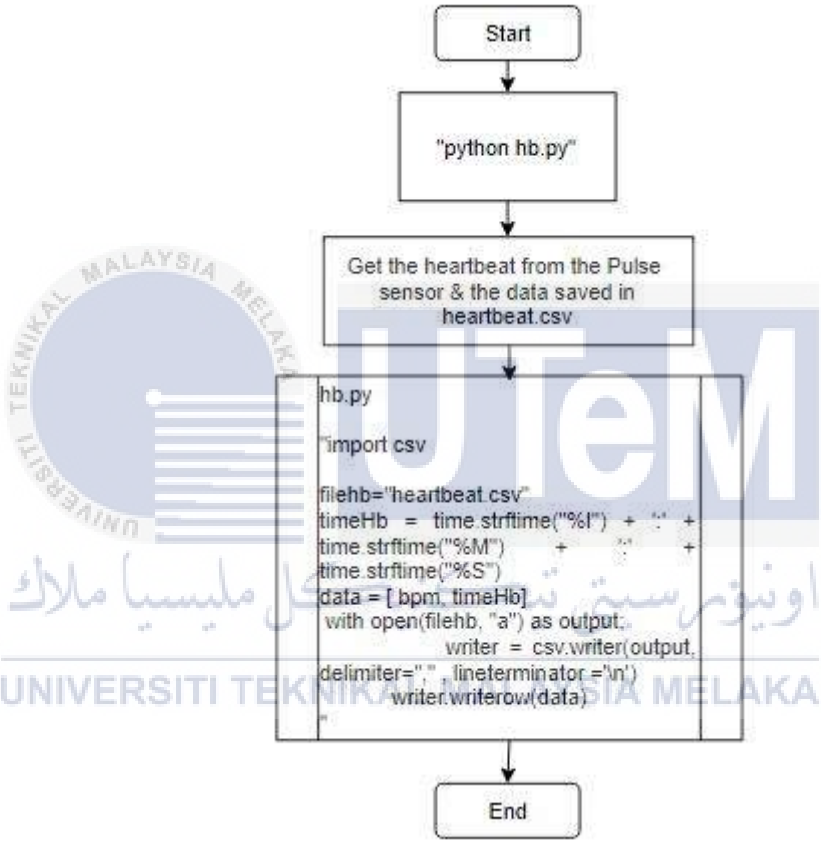

Figure 5.6: heartbeat.csv

Figure 5.6 shows how the heartbeat data is saved in heartbeat.csv. In heatbeat.csv, only the value of heartbeat and the time of heartbeat are stored in it. That information will be used to plot the graph of heartbeat to ease the user to monitor their heart rate.

5. Development and Configuration (Matplotlib)

	Software Implemented
hbgraph.py	Python language
	Matplotlib (graph)
	Numpy (Calculation of average)
	Pyplot (Animation)

Table 5.4: Software implementation of hbgraph.py.

Table 5.4 above shows the implementation of hbgraph.py. Python is used as the language. For the graph, matplotlib is imported while for the average calculation of heartbeat and the animation of the graph, numpy and pyplot are imported as well in hbgraph.py.

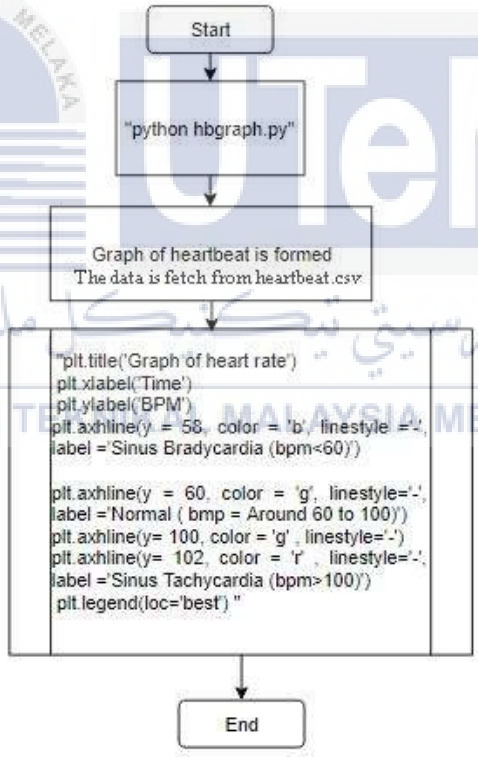

Figure 5.7: Graph of heartbeat

Figure 5.7 shows the flowchart of how the prototype produces the graph of heartbeat. The data is first fetched from hearbeat.csv to plot the graph. In the graph, there will 4 horizontal lines. The two lines in green colour indicate the normal heartbeat. Two lines are used to show the range which is 60 to 100 bpm for a normal

heartbeat while the other two horizontal line which is in blue and red colour indicate Sinus Bradycardia (less than 60 bpm) and Sinus Tachycardia (higher than 100 bpm). Labels also provided for this graph and it uses colour to differentiate between normal, slow and fast heart rate.

6. Development and Configuration (Telegram)

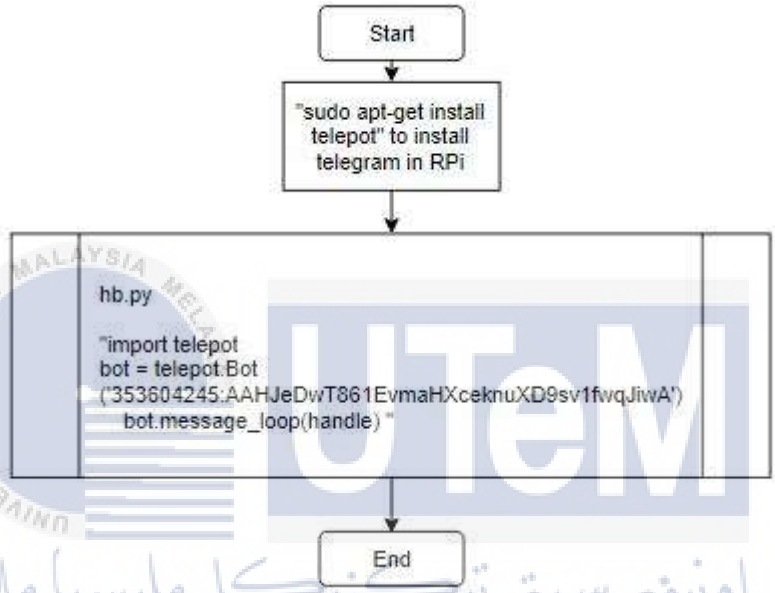

Figure 5.8: Install Telegram on Raspberry Pi

Figure 5.8 above shows the installation of Telegram on Raspberry Pi. Telepot helps to build application API for Telegram Bot. After finish installing, Telegram Bot (on smartphone) can be created by communicating with Bot Father which means that, another new Bot is created with the name of Hepuhebsor. After that, Bot Father will give a token to be shared with Raspberry Pi.

7. Development and Configuration (Telegram-send file)

	Software Implemented
telegram.py	Telepot
	Token API
	Python Language

Table 5.5: Software implementation of telegram.py.

Table 5.5 above shows the implementation of telegram.py. Python is used as the language. Token API is used to enable Raspberry Pi and telegram to share data. Telepot is imported inside telegram.py to activate the telegram in Noobs (operating system).

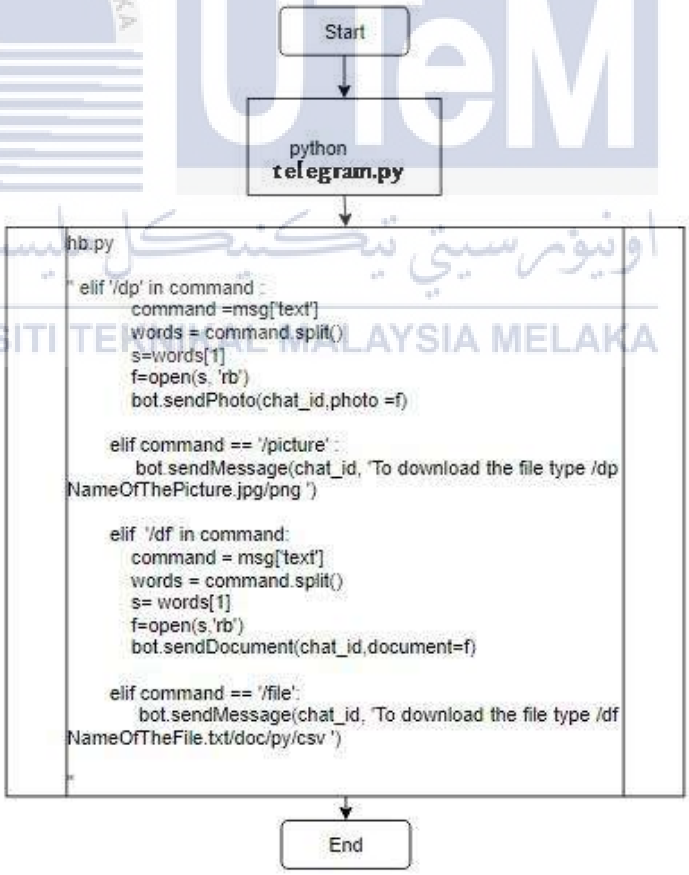

Figure 5.9: Telegram Bot Download function.

Figure 5.9 above shows the download function of Telegram. Once the data of heartbeat stored in heartbeat.csv, it can be downloaded (after raspberry pi send it). Same goes to the graph. The graph of heartbeat for several minutes can be saved whether in the form of picture or others and it can be downloaded as well.

8. Development and Configuration (Telegram-view symptom)

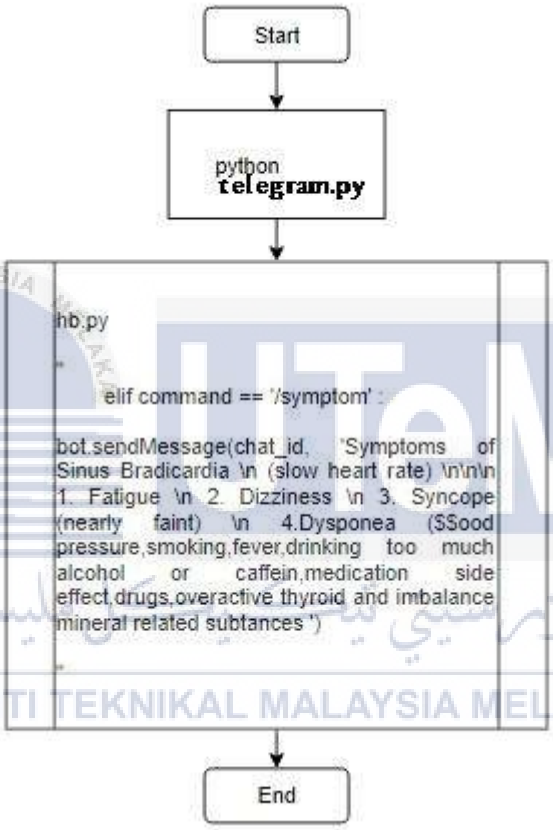

Figure 5.10: Telegram Bot Symptom function

Figure 5.10 above shows the symptom function that will be sent to Telegram. The symptom will be sent in the form of simple text. The user need to send request for the symptom text then Raspberry Pi will send it. The function of symptom text is to tell the user about the symptom they may face according to the result of their heartbeat. It will give symptom of both Sinus Bradycardia (slow) and Sinus Tachycardia (fast). It is good for the user to know which symptom stated match with their condition at that moment

9. Development and Configuration (Telegram-view precaution)

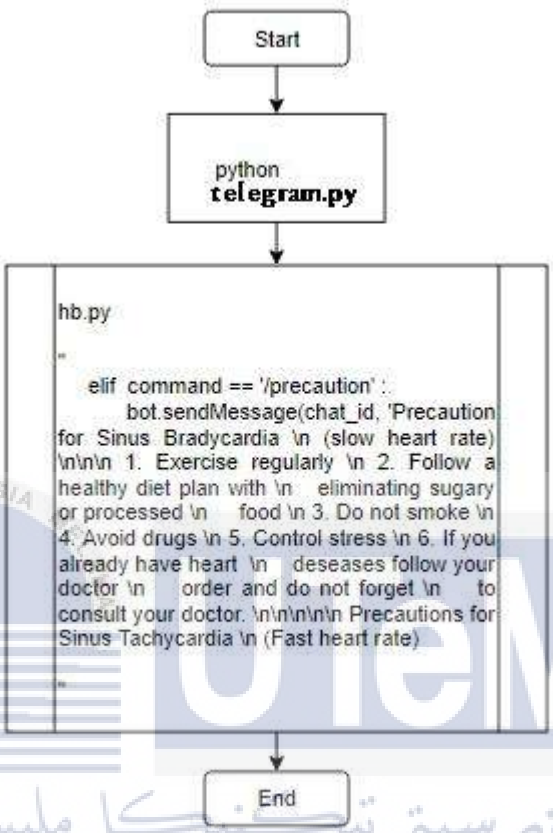

Figure 5.11: Telegram Bot Precaution function

Figure 5.11 above shows the precaution function that will be sent to Telegram. The precaution will be sent in the form of simple text. The user needs to send request for the precaution text then Raspberry Pi will send it. It will give precaution of both Sinus Bradycardia (slow) and Sinus Tachycardia (fast). It is good for the user to take ealy precaution before it gets worse.

10. Development and Configuration (Telegram-view folder)

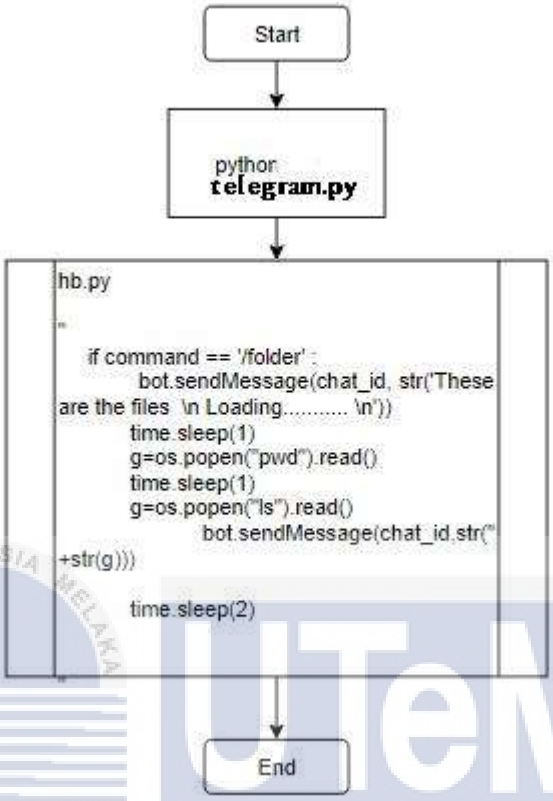

Figure 5.12: Telegram Bot view folder function

Figure 5.12 above shows the view folder function. User can view all the files in Raspberry Pi"s folder. It is to ease the user to see and choose which one needs to be downloaded especially the graph that they have saved.

11. Overview of Development and Configuration GUI of Friendly Hepuhebsor.

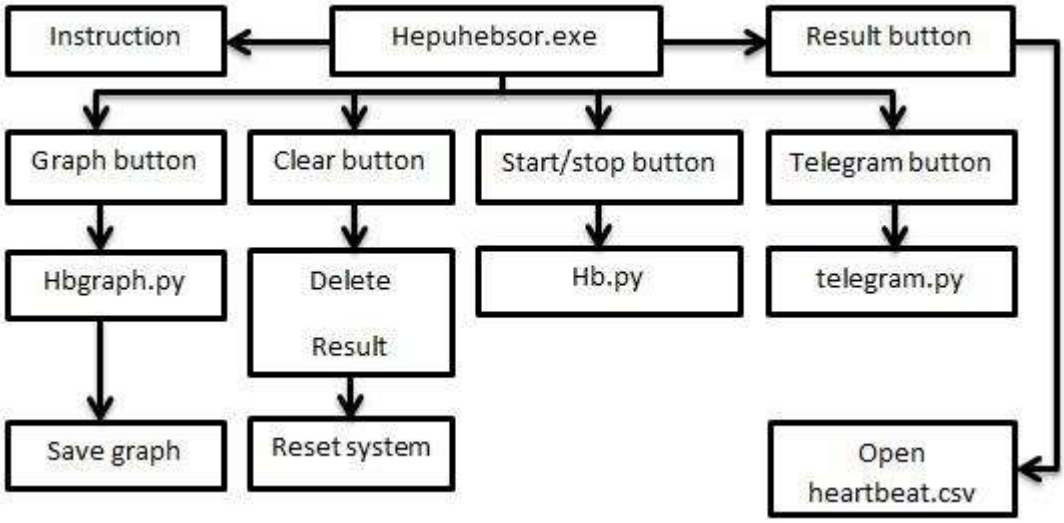

Figure 5.13: Overview of Friendly Hepuhebsor GUI.

Figure 5.13 above shows the overview of the development of GUI used for this project. Overall it has 5 functions and 8 buttons. There are 3 functions (hb.py, hbgraph.py and telegram.py) use 2 buttons which are graph, start/stop and telegram button. Clear buton is used to reset the system and Result button is used to open heartbeat.csv which is the result of heartbeat.

12. Development and Configuration (GUI-Instruction)

	Software Implemented
hepuhebsor.py	• Python language • TKinter • Subproses • Os.system • Threading • webbrowser

Table 5.6: Software implementation for hepuhebsor.py

Table 5.6 above shows the software implementation for hepuhebsor.py. Python is used as the language. TKinter is imported for the interface and all the buttons. Subproses and Threading are also imported to be used for every function that indicate the buttons (without freezing the other python scripts). Os is imported inside hepuhebsor.py to execute another python script. Webbrowser is used to execute the csv file with a button click.

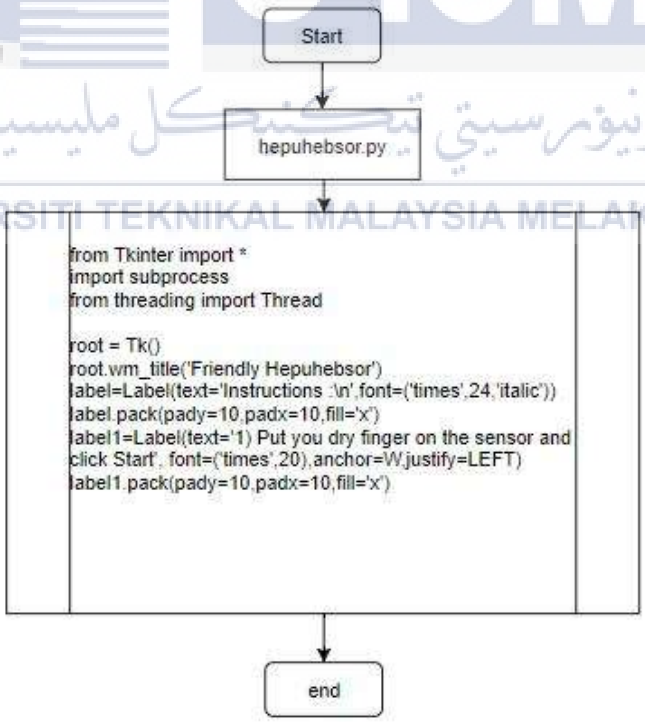

Figure 5.14: Flowchart of the instruction.

Figure 5.14 above shows the instruction of how the prototype work. For this system the GUI only use single frame. There are 5 instructions need to be followed by the user. Tkinter is used as the GUI builder. It is also installed in the raspberry pi and imported to be used.

13. Development and Configuration (GUI- Start & Stop button for hb.py)

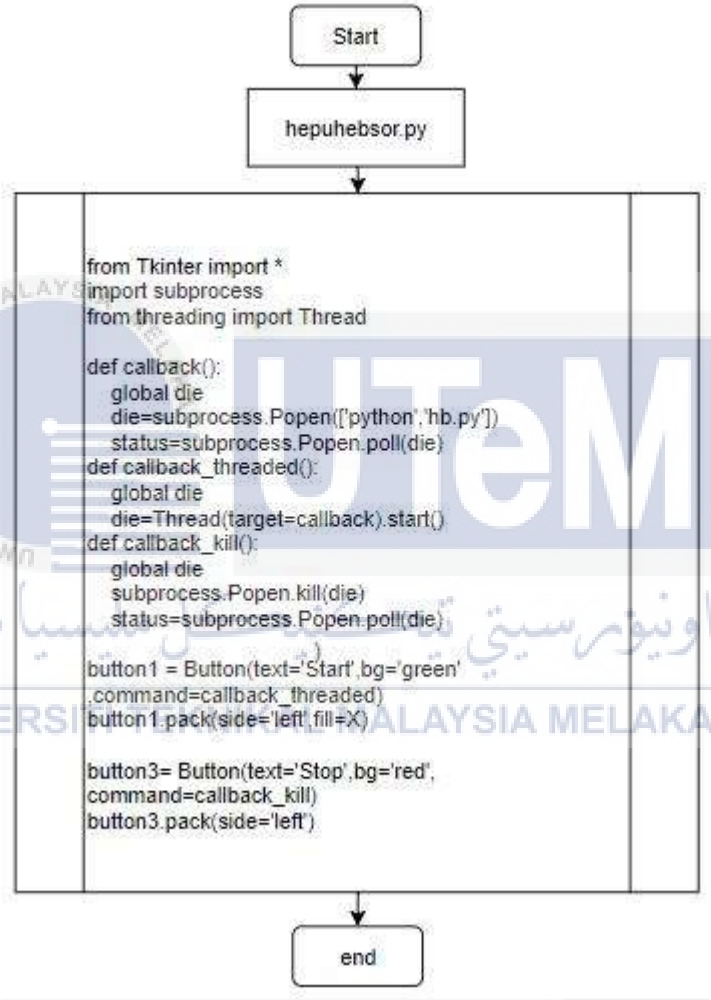

Figure 5.15: Flowchart of Start/Stop button for hb.py

Figure 5.15 above, shows the flowchart of button used for hb.py (Start/Stop button). Subprocess and threading is used for this function. Start button is used to start the measurement of heartbeat and if user want to stop monitoring they can click on the Stop button.

14. Development and Configuration (GUI- Start & Stop button for hbgraph.py)

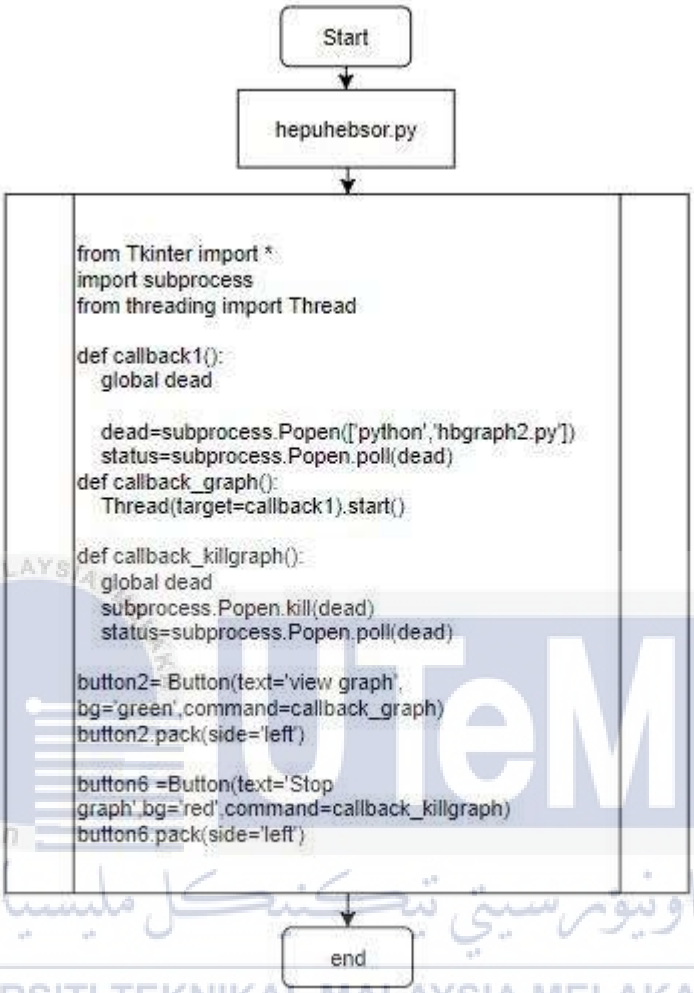

Figure 5.16: Flowchart of Start/Stop button for hbgraph.py

Figure 5.16 above shows the button to control the graph. The graph will be executed outside the frame. It is depends on the user whether they want to use it or not. The function of a graph is to ease the user to see their heart rate. Using this graph user also can save their heart rate graph for several minutes. Subprocess and threading also used for this function.

15. Development and Configuration (GUI- Start & Stop button for telegram.py)

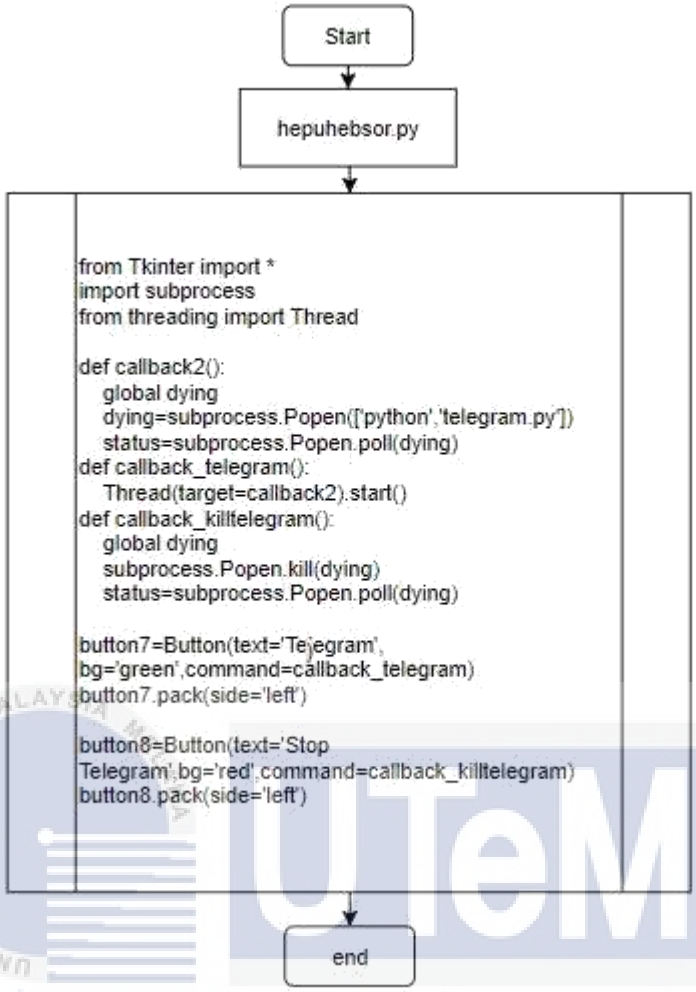

Figure 5.17: Flowchart of Start/Stop button for telegram.py

Figure 5.17 above shows the button to control the telegram. Once the user click Start button raspberry pi will be connected to telegram. It is depends on the user whether they want to use it or not. The function of telegram is to ease the user to get the result if they want to keep it. Using telegram user also can get download their heart rate graph for several minutes and the result (heartbeat.csv) as well. Subprocess and threading also used for this function.

16. Development and Configuration (GUI- view result button)

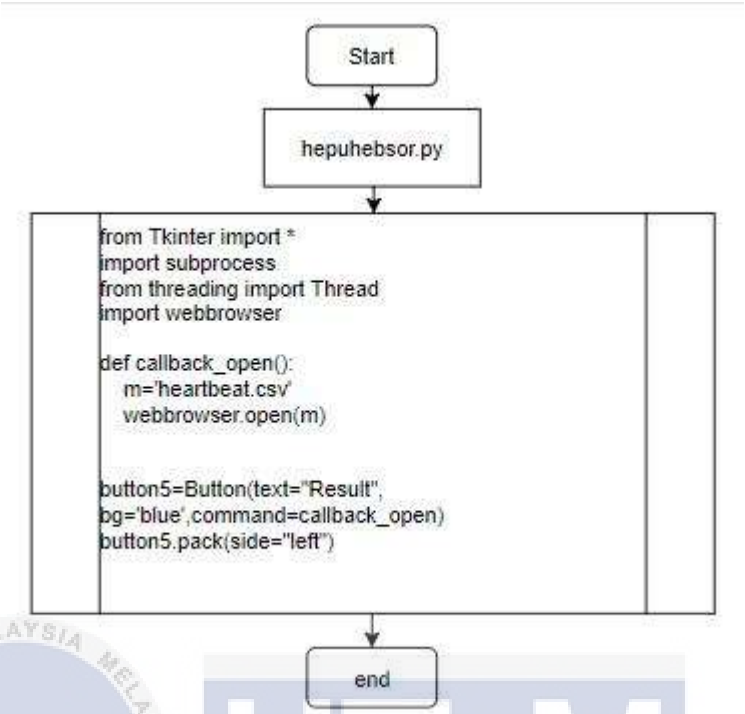

Figure 5.18: Flowchart of button view result (heartbeat.csv)

Figure 5.18 above shows the button to control heatbeat.csv. The heartbeat value is stored inside heartbeat.csv and it will show the time as well. So the user will know how many minutes of monitoring. The result will be executed outside the frame. The value of heartbeat and the time stored inside heartbeat.csv is used to plot the graph. For this function webbrowser is used to open the file.

17. Development and Configuration (GUI- clear system button)

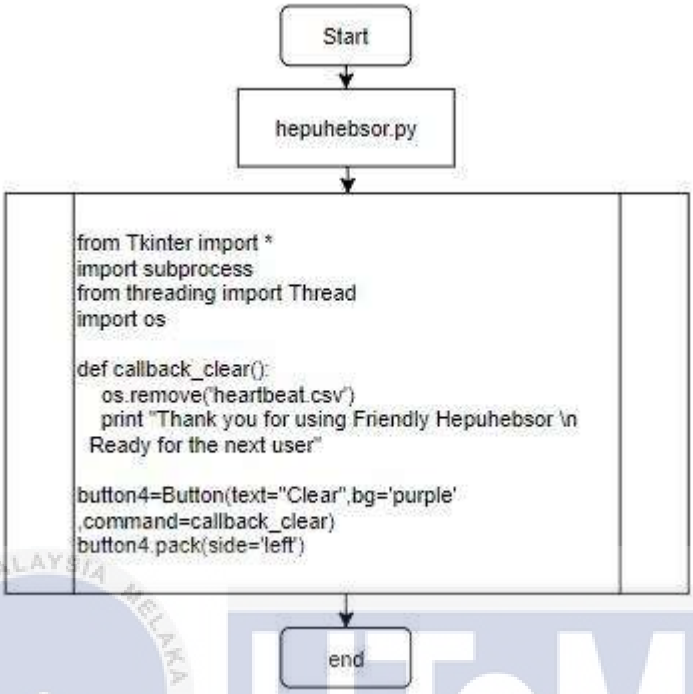

Figure 5.19: Flowchart of Clear system button

Figure 5.19 above shows the flowchart to clear the system so another user can use it back. Os is used to control this function. Once Clear button is clicked, the result will be deleted.

18. Development and Configuration (Convert hepuhebsor.py to hepuhebsor.exe)

	Software Implemented
hepuhebsor.exe	• Setuptools • Pyinstaller

Table 5.7: Software implementation of hepuhebsor.exe

Figure 5.7 above shows the software implementation of hepuhebsor.exe. hepuhebsor.py is changed to hepuhebsor.exe to enable „python hepuhebsor.py" to be executed with a single click. Setuptools is installed on the Raspberry Pi to enable the the installation of pyinstaller. Pyinstaller is used to convert python script to exe file to enable user to click the icon to execute the python script.

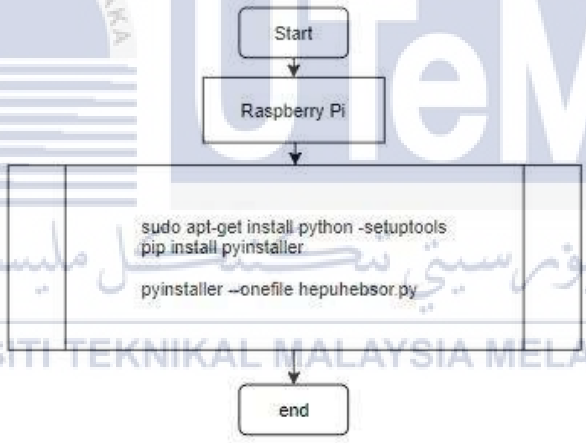

Figure 5.20: Flowchart of converting hepuhebsor.py to hepuhebsor.exe

Figure 5.20 above shows the flowchart of converting hepuhebsor.py to hepuhebsor.exe The function is to ease the user to click the icon to execute Friendly hepuhebsor.

5.4 Implementation Status

In this part, we will explain about the status of the development of each component or module. In this explanation, it will comprise of module name,

description, duration to complete and date completed. The progress is shown in Table 5.1 below.

No	Module Name	Description	Duration	Date to completed
1	Set up server	Setting up Raspbian server on Raspberry Pi	3 Days	4 Months starting from 10 March 2017 until 12 June 2017
2	Set up the prototype	Make sure all the components connected to breadboard and Raspberry Pi	5 Days	
3	Install Spidev module on Raspberry Pi	The installation of Spidev is a must because it will be used for SPI	1 Day	
4	Implement the code for MCP3008	For this code, we are going to set which channel is going to be used for analog to digital conversion.	3 Days	
5	Implement the code for pulse sensor.	2 python scripts are used for pulse sensor. One is pulsesensor.py and another one is hb.py and only hb.py will be executed.	7 Days	
6	Implement the code for the graph of heartbeat.	Code for the graph is saved in hbgrah.py (another python script). For this script the GUI is created using Tkinter.	15 Days	
7	Install Telepot & Implement the Token for Telegram Bot.	Connect the telepot from Raspberry pi with Telegram Bot on smartphone.	4 Days	

8	Implement download function for Telegram Bot	The code for download function is saved in hb.py	6 Days	15 June 2017 until 30 June 2017
9	Implement view function for Telegram Bot	The code for view function is saved in hb.py	5 Days	
10	Complete the system.	Create GUI for the system.	15 Days	18 July 2017 – 11 August 2017
11	Test the prototype	Test the prototype wether it‟s working or not.	2 Days	12 August until 13 August

Table 5.8: Implementation status

5.5 Conclusion

This chapter explained about the whole implementation for the prototype. Implementation refers to the process of completing the plans in order to achieve the objective of the project. In Chapter 6, we will conduct the testing of the system to determine if the prototype is working properly.

CHAPTER VI

TESTING

6.1 Introduction

In the previous chapter, we have discussed briefly about the implementation of the project. After finished the implementation, we will now continue to discuss and review the testing of this project. Testing is vital to make sure the project is completed and it works as mentioned in the objectives which are simple heartbeat analyser, friendly to be used and effective.

6.2 Test Plan

This part will be explaining about the basis of each of the system testing. Testing scope and the activities that will be carried out throughout the testing phase will also be covered in this part.

6.2.1Test Organization

Friendly Hepuhebsor prototype is designed as a personal healthcare. Therefore it can be used by anybody who wants to monitor the heartbeat. The result can be viewed by the person who monitored his heartbeat and other people related to it like doctor or relatives can view the result of heartbeat via Telegram Bot.

6.2.2 Test Environment

The testing can be carried out everywhere but the best place is the environment that has less electronic gadgets. That is because pulse sensor uses infrared light and infrared is well known as wave detector. In order to prevent pulse sensor to detect any waves other that arterial pulse wave, the prototype need to be tested in a place that have less electronic gadget (highly recommended).

6.2.3 Test Schedule

The cycle of the testing plan will be as follows. After the prototype has been set up, the user needs to run the hepuhebsor.exe and to use it, user only need to click available buttons.

1. Start the system.

To use the system, user need to click „hepuhebsor.exe" icon and open it with terminal. The terminal is important to let user know whats going on with the system. Besides, user need to read the instructions carefully to understand how the system work. Figure 6.1 below shows

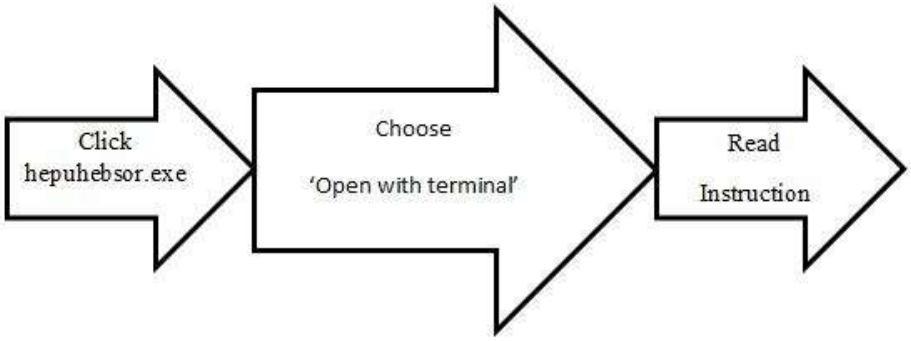

Figure 6.1: Function of hepuhebsor.py

2. Get the heartbeat reading.

The system has start and stop button. The function of these button is to let the user to control when they want to use it and when they want to stop using it. Figure 6.2 below shows the

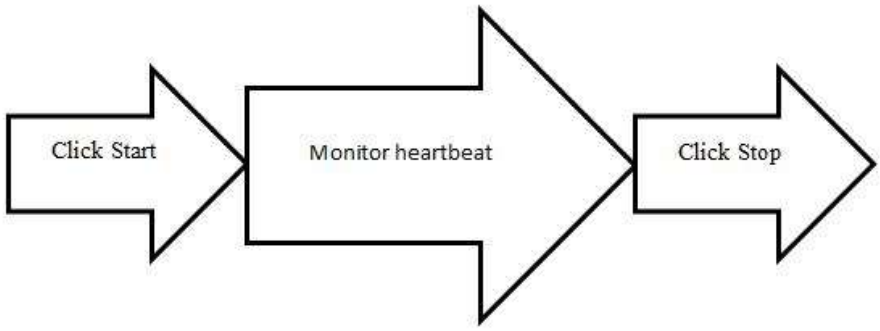

Figure 6.2: Function of hb.py

3. View the result of heartbeat.

The result will be stored in heartbeat.csv. User can only click „Result button" once they click „start button". Figure 6.3 below shows

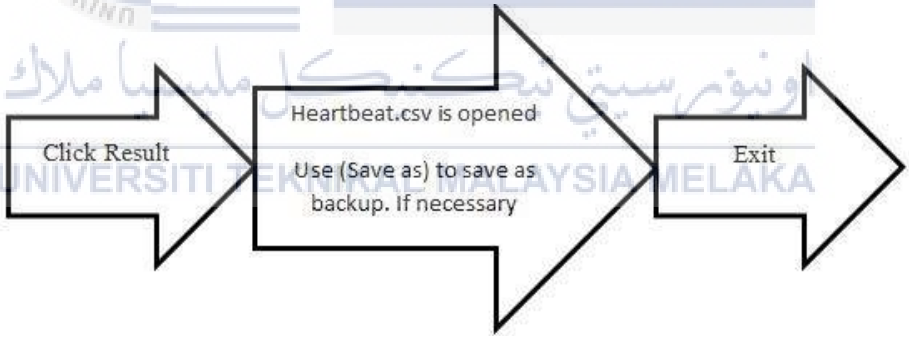

Figure 6.3:Function of webbrowser (heartbeat.csv)

4. View result in a graph.

Another way to view the result is by transforming it into a graph. The graph view is to make the user to view their heartbeat easier because it is more interactive . User can choose whether they want to save it the graph form or not (for several minute). Figure 6.4 below shows

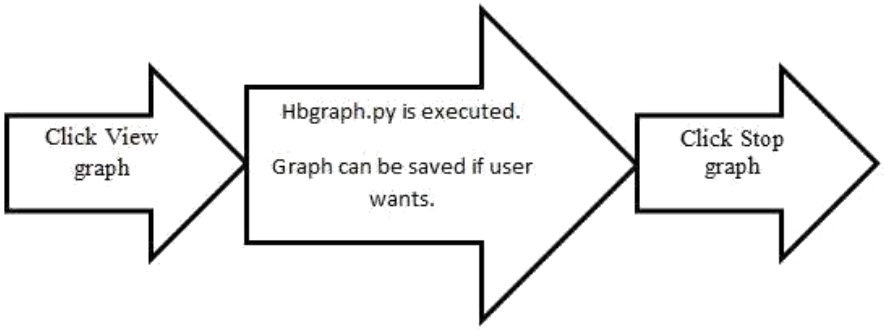

Figure 6.4: Function of hbgraph.py

5. Connect to telegram.

The system also has buttons for telegram. The function is to connect the raspberry pi tu Telegram. Raspberry pi can send the result, graph image, symptom and precaution to telegram. Telegram also can view all the files inside the hepuhebsor folder (this function is only used if the user want to see file/picture name that have been saved). For heartbeat.csv, it‟s send automatically but once it is save as another name(as back up purpose if user want), user need to type the name to get it from raspberry pi. Figure 6.5 below shows.

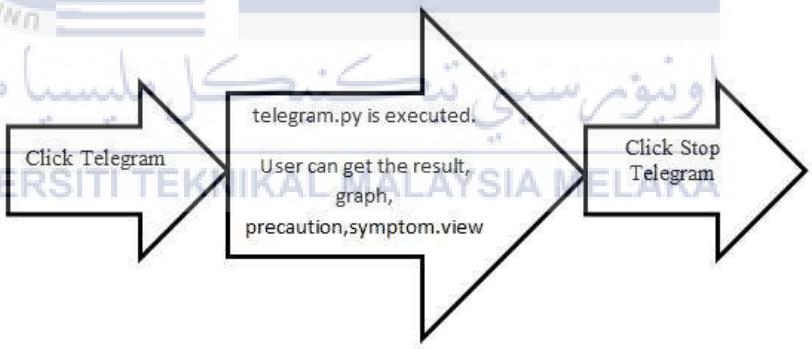

Figure 6.5: Function of telegram.py

6. Clear the system.

The system need to be cleared to ensure other user can use the system. System clear means the result(heartbeat.csv) is deleted. This is important for graph plotting. Figure 6.6 below shows

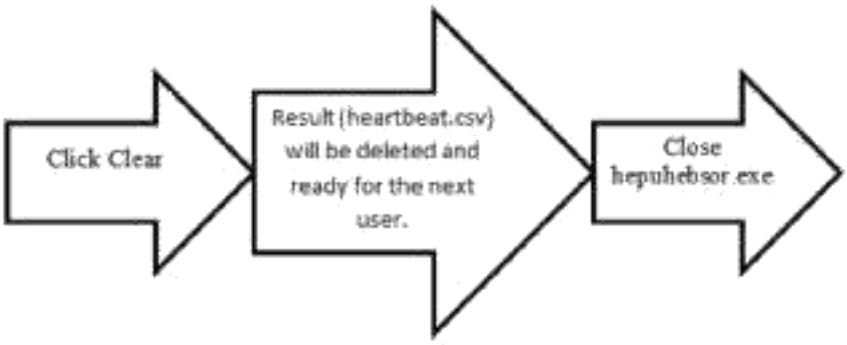

Figure 6.6: Function of clear system.

6.3 Testing Strategy

Various strategies used for the testing. The testing most preferably carried out in room temperature . The testing involving several people doing some tasks and then they can start to measure their heartbeat. There are 3 categories need to be tested which are normal, slow and fast heartbeat. Before the testing carried out, as interview session with a medical doctor was held in UTeM Clinic. The function of the interview is to know how to test the prototype and who is suitable to test the prototype.

1. Interview Session with Doctor Suria Bt Abu Kassim (Medical Doctor)

For the interview session, there were several questions asked to Dr. Suria which are :

Question 1: How pulse generated?

Answer: Pulse generated when there is a pulse preasure (systole and diastole process of heart and blood capillaries).

Question 2: Which group of people has normal, slow and fast heart rate?

Answer: People who has normal heart rate is usually when sleeping or resting. Fast heart rate, happened to those who are exercising, thyroid patient and alcoholic while for slow hear rate always happened to those who take some medicine, old people or athele at resting time.

Question 3: How to get heartbeat reading from the pulse using machine(pulse sensor) ?

Answer : There are 3 ways to measure heartbeat manually which are from the neck, arm and wrist.If using a machine (pulse sensor), pulse can be detected from finger using carotid artery which is the main artery came from the wrist.

Question 4: Is it possible for human to have no pulse for several minutes?

Answer: It is not possible. No pulse means the human is dead. If using a machine (sensor), if there is a 0 value for several seconds, that‟s mean there is something wrong with the machine(pulse sensor) or it‟s still reading (looking for pulse).

2. Testing strategy for normal heartbeat.

For normal heartbeat, there are two options for testing. For this project „Resting‟ is chosen as the method to test normal heartbeat as shown in the figure 6.7 below.The monitoring of heartbeat will be carried out in several seconds/minutes.

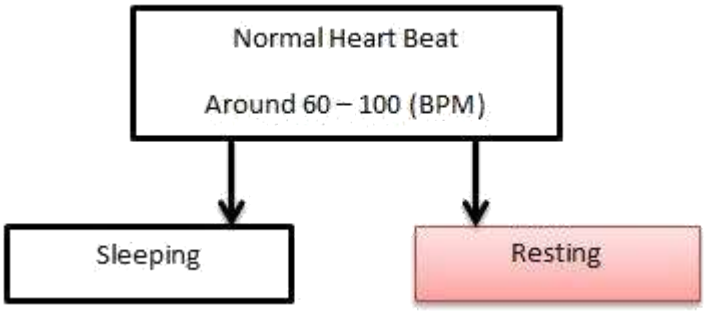

Figure 6.7: Strategy used for normal heartbeat 3. Testing strategy for slow heartbeat.

There are some options to test the slow heartbeat . For this project drug/medication and old people are chosen as shown in figure 6.8 below. The monitoring of heartbeat will be carried out in several seconds/minutes.

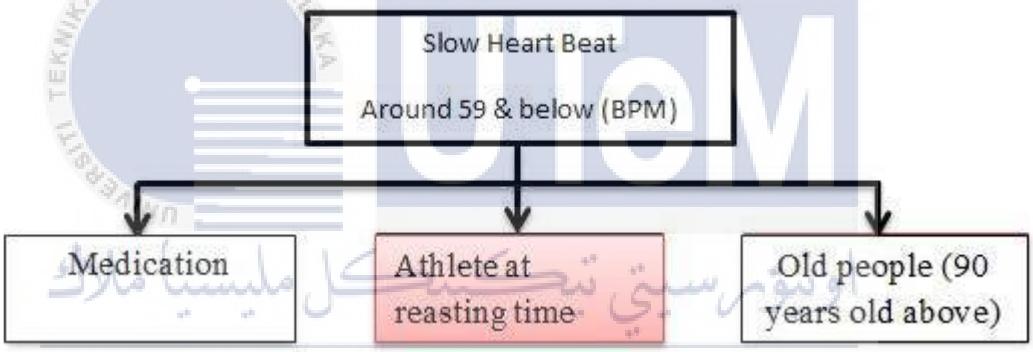

Figure 6.8: Strategy used for slow heartbeat

4. Testing strategy for fast heartbeat.

For fast heartbeat, there are three options for testing. For this project „Exercising‟ is chosen as the method to test normal heartbeat as shown in the figure 6.9 below.The monitoring of heartbeat will be carried out in several seconds/minutes.

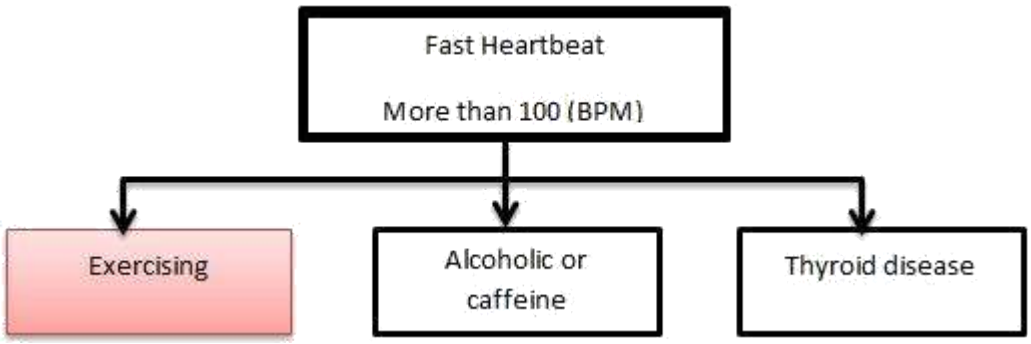

Figure 6.9: Strategy used for fast heartbeat

6.4 Test Design

Test design is the act of creating and writing test suites for testing software. In this project, the testing will be conducted as planned.

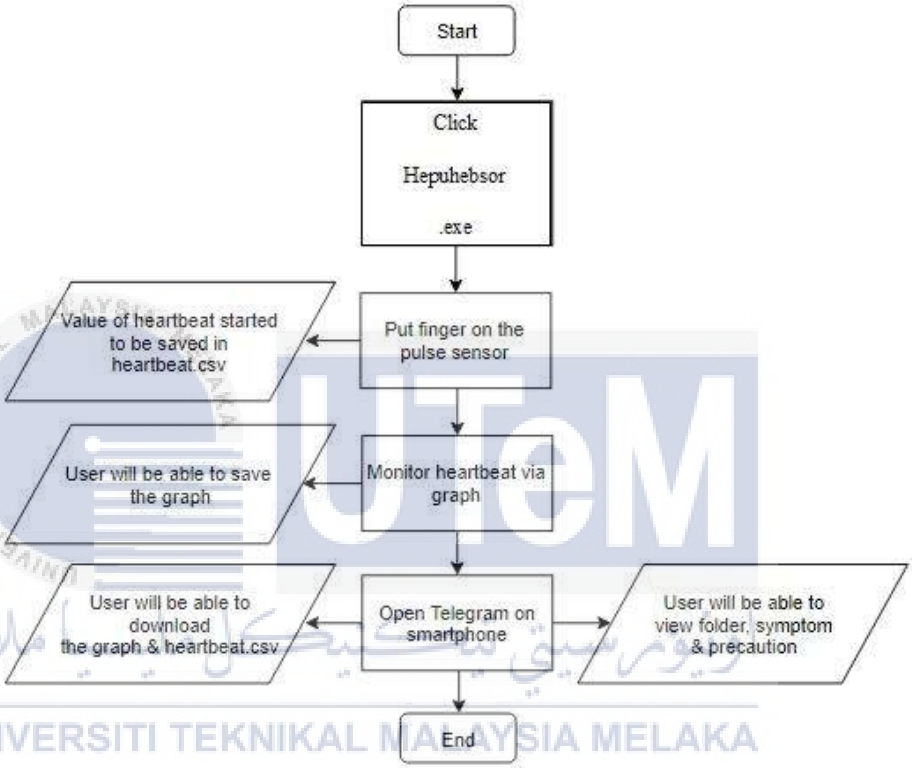

Figure 6.10: Flowchart of system testing.

Figure 6.10 above shows the flowchart of system testing. Hepuhebsor.exe has been created to replace Hepuhebsor.py. User can simply click to start the system. After that user can start the monitoring session and do other related things on the system such as, view result, transform the result in the form of graph, connect to telegram and clear. Hepuhebsor.exe need to be executed together with the terminal because the terminal will show what is happening the moment click on the available buttons.

6.5 Test Result.

After carried out the testing strategy and testing plan explained earlier, the project was successfully developed as mentioned in chapter V. Test result is divided into two segment which is on Raspberry Pi and Telegram.

6.5.1 Test Result Raspberry Pi & Raspbian Server.

Figure 6.11: Prototype of Friendly Hepuhebsor.

Figure 6.11 above shows the prototype of Friendly Hepuhebsor. The prototype has been set up and connected to the laptop. The sensor is function well with the infrared light. All the python code also already being implemented.

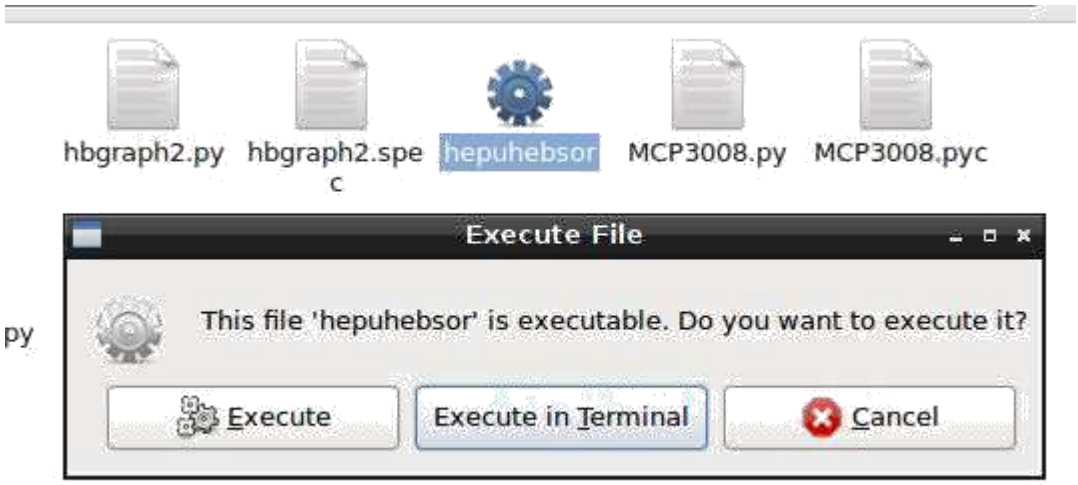

Figure 6.12: Execution of hepuhebsor.exe

To use this system, user need to click on the „hepuhebsor.exe" as shown in the figure 6.12 above. User also need to chose „Execute in terminal because the terminal will tell the user the proses happened after they click all the available buttons.

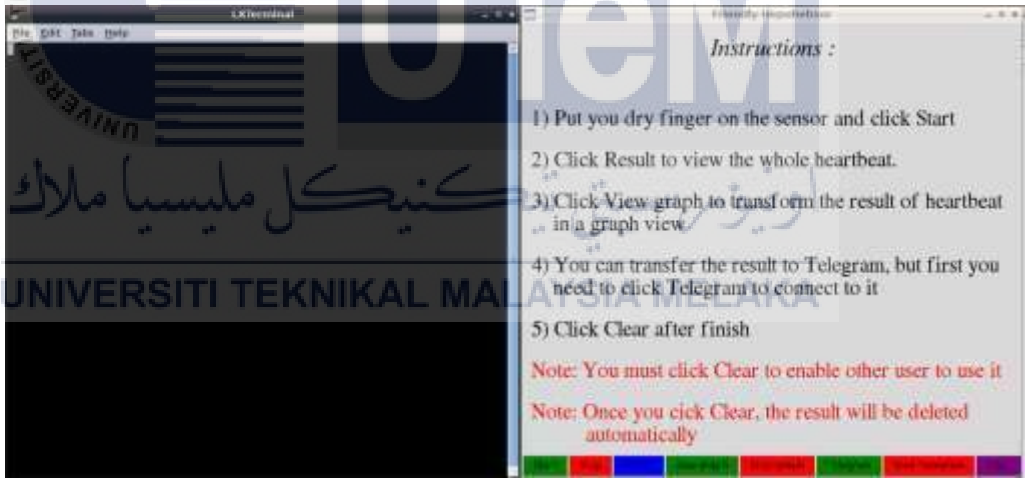

Figure 6.13: Interface of Friendly Hepuhebsor.

The interface consist of simple insruction and several buttons as shown in figure 6.13 above. It is a must for user to read the instruction to understand how the system work. Once the button click, it will show something(output/process/text) on the terminal.

Figure 6.14: Normal heartbeat (normal people at rest).

Figure 6.14 above shows normal heartbeat. Normal heartbeat is between 60 bpm to 100 bpm. The value of heartbeat is shown on the terminal. User can control the monitoring session by using the „start" and „stop" button. Friendly Hepuhebsor is tested on people on their resting time. The heartbeat stated in figure above is the heartbeat of a resting woman.

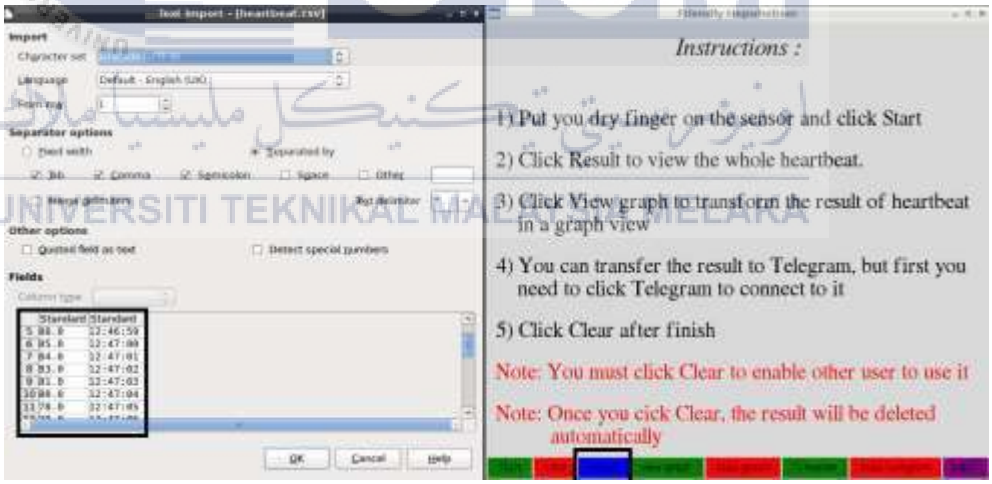

Figure 6.15: Normal heartbeat data

All the heartbeat data is stored in heartbeat.csv. User can click „Result" as shown in figure 6.15 above to view the data. The heartbeat data above is for normal bpm. Heartbeat.csv has heartbeat value and time (h:m:s). Heartbeat.csv is important for graph plotting.

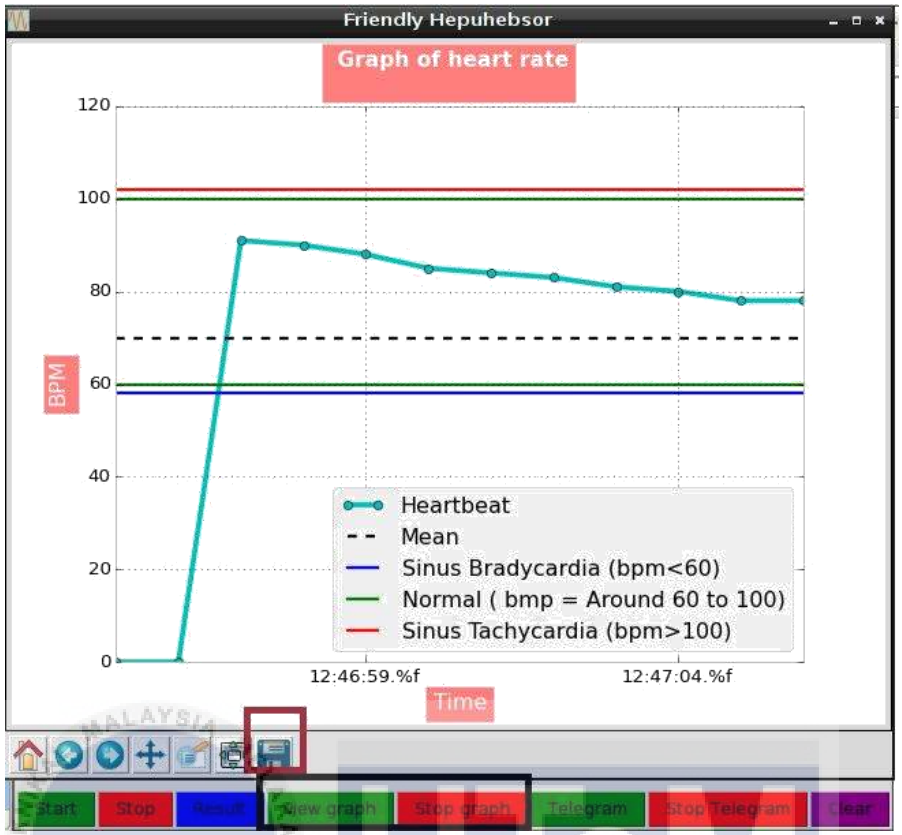

Figure 6.16: Graph of normal heartbeat.

Figure 6.16 above shows the graph of normal heartbeat The graph will plot graph of heartbeat vs time. It will show 4 horizontal lines. The two lines in green colour represent the range of normal heart rate which is from 60 bpm to 100 bpm. The blue horizontal line represents the starting point of slow heart rate and the value is 59 and below. The red horizontal line represents the starting point of fast heart rate and the value is 100 and above.This is the graph result for nomal heartbeat tested on normal people at resting time. It plots every beat stored in the result and it will also calculate the mean of heartbeat (the mean shows bpm is around 70).

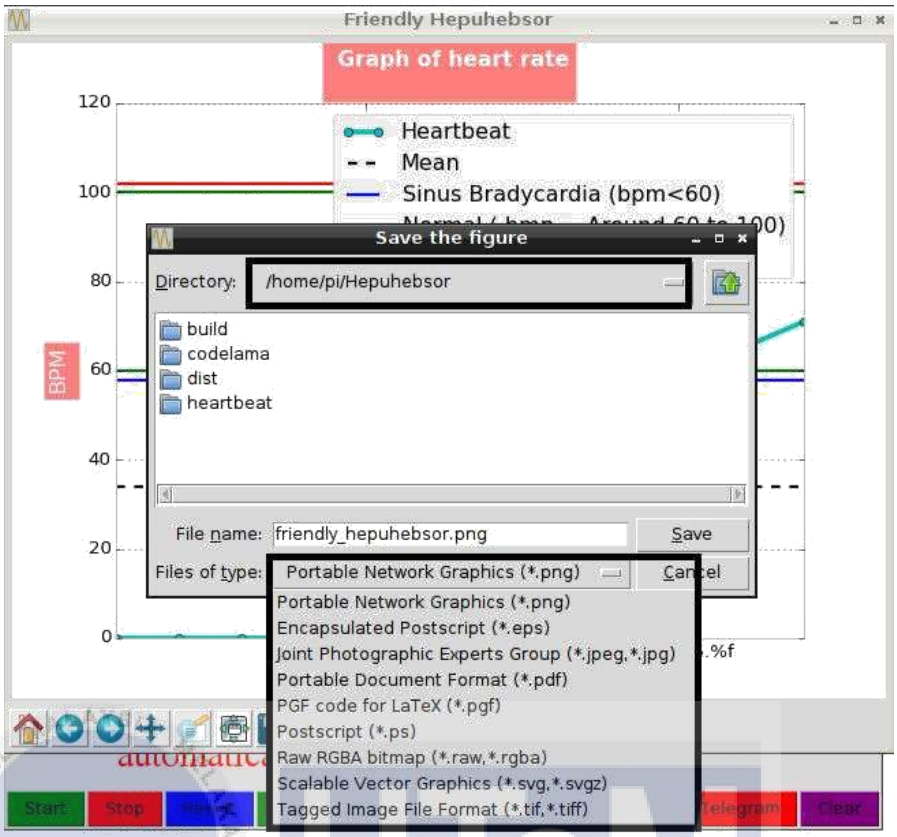

Figure 6.17: Path to save the graph result.

The graph must be saved in hepuhebsor folder if user want to transfer the graph to the telegram because the path of the telegram is in /home/pi/Hpuhebsor. User can save in various forms as shown in figure 6.17 above.

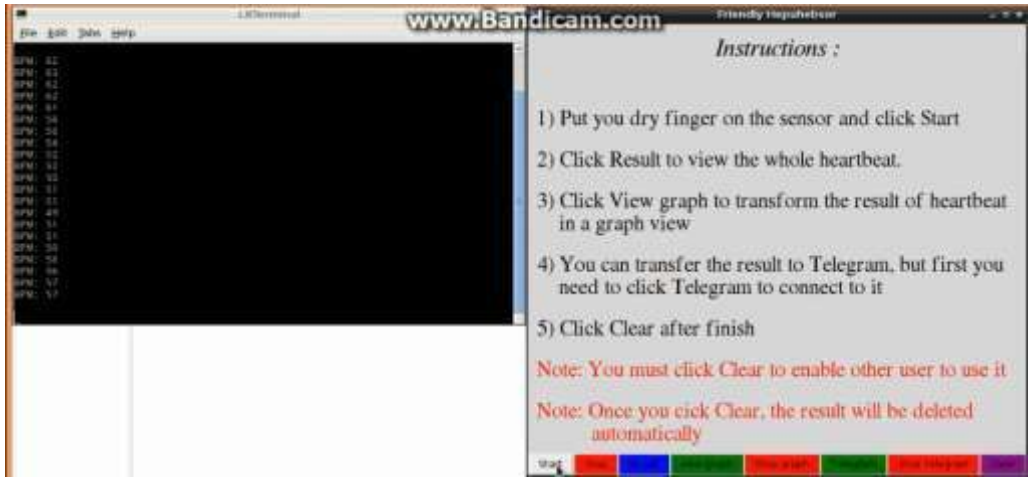

Figure 6.18: Slow heartbeat (athelete at rest).

Figure 6.14 above shows slow heartbeat. Slow heartbeat is 59 bpm and below. This is the heartbeat of a young atheletic man. Athelete at resting time tend to have slower heartbeat compared to the non-atheletic people.

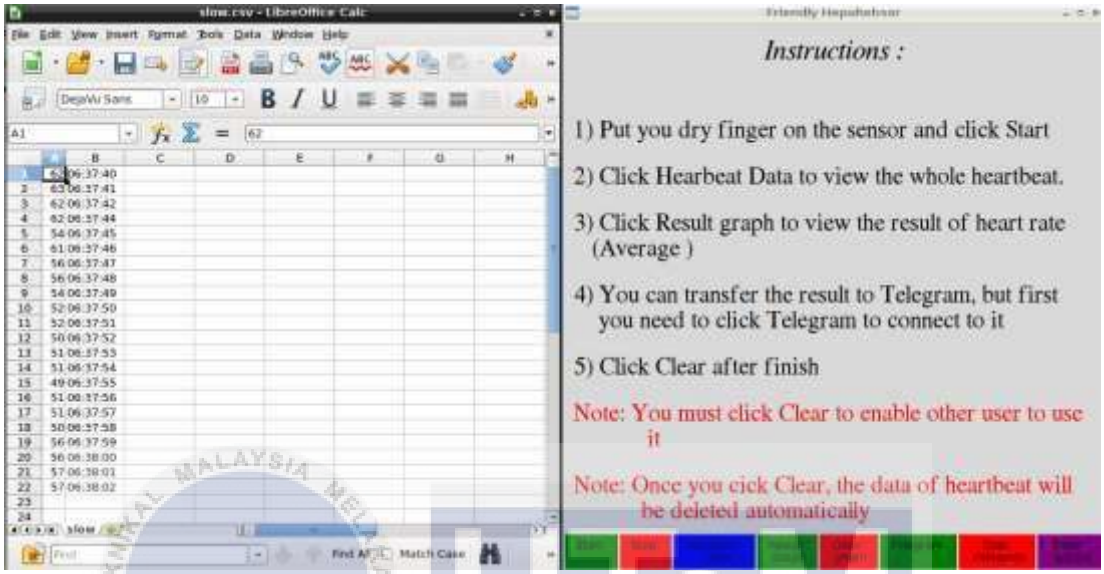

Figure 6.19: Heartbeat data of athelete at rest.

All the heartbeat data is stored in heartbeat.csv. Figure 6.15 above shows the heartbeat data of an athelete. Most of value of bpm is lesser than 60 bpm. Heartbeat.csv has heartbeat value and time (h:m:s).

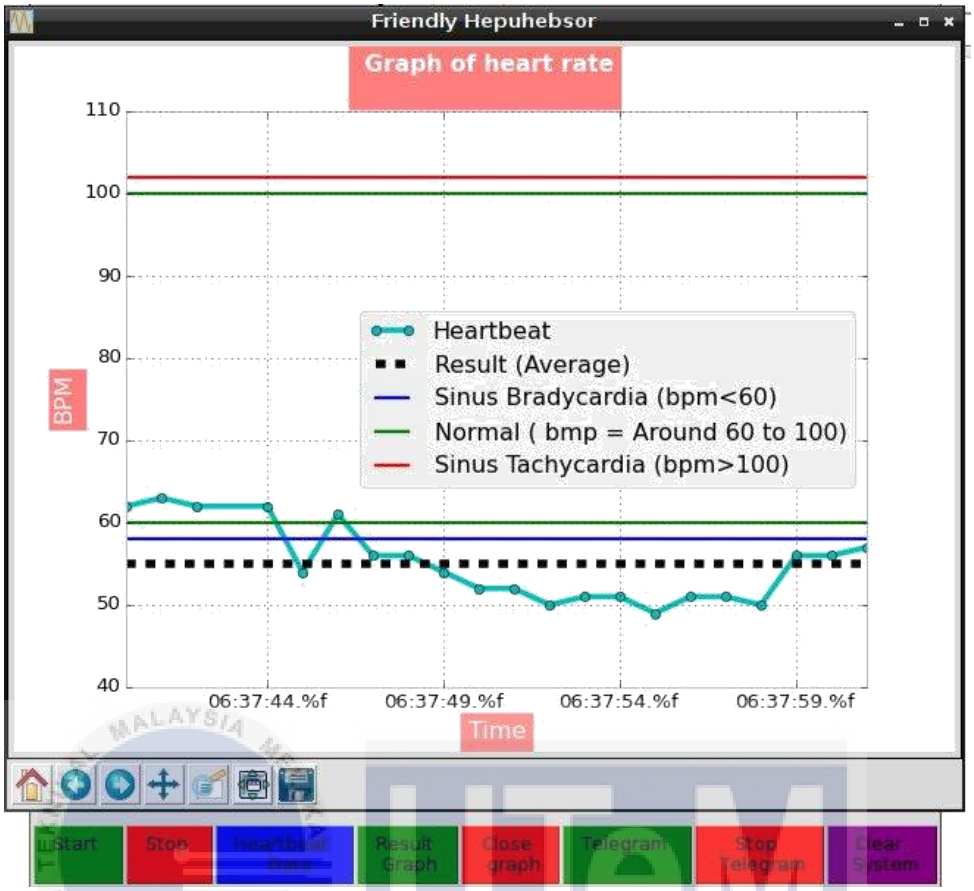

Figure 6.20: Graph of slower heartbeat of an athele at rest.

Figure 6.20 above shows the graph of slow heartbeat of an athelete at his resting time. The graph shows the average of heartbeat (black horizontal line) is lesser than 60 bpm.

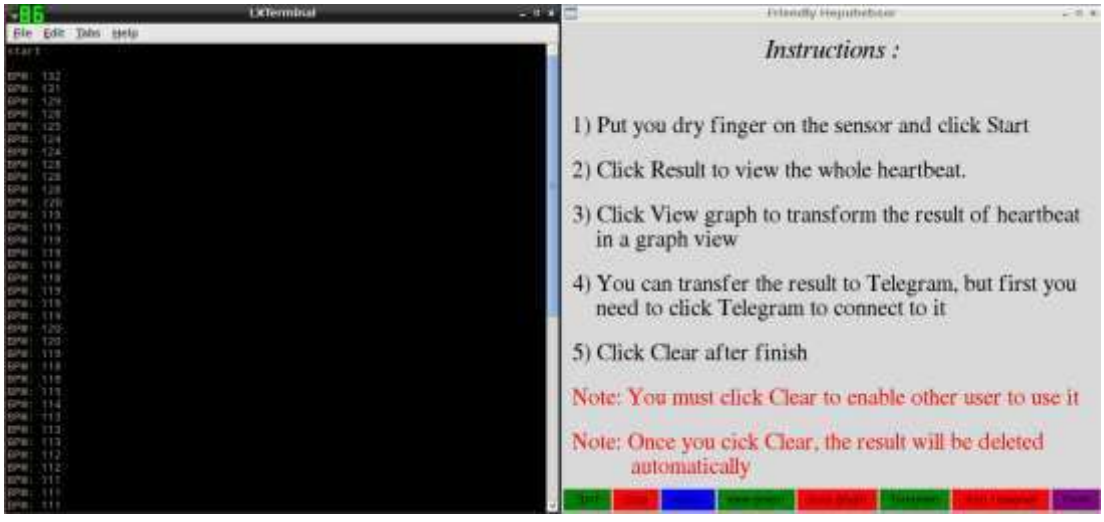

Figure 6.21: Fast heartbeat of exercising man.

Figure 6.21 above shows fast heartbeatof exercising man. Fast heartbeat is greater than 100 bpm. The man had been exercising for more than 5 minutes. The reading of heartbeat is taken the moment she started to breath vigorously.

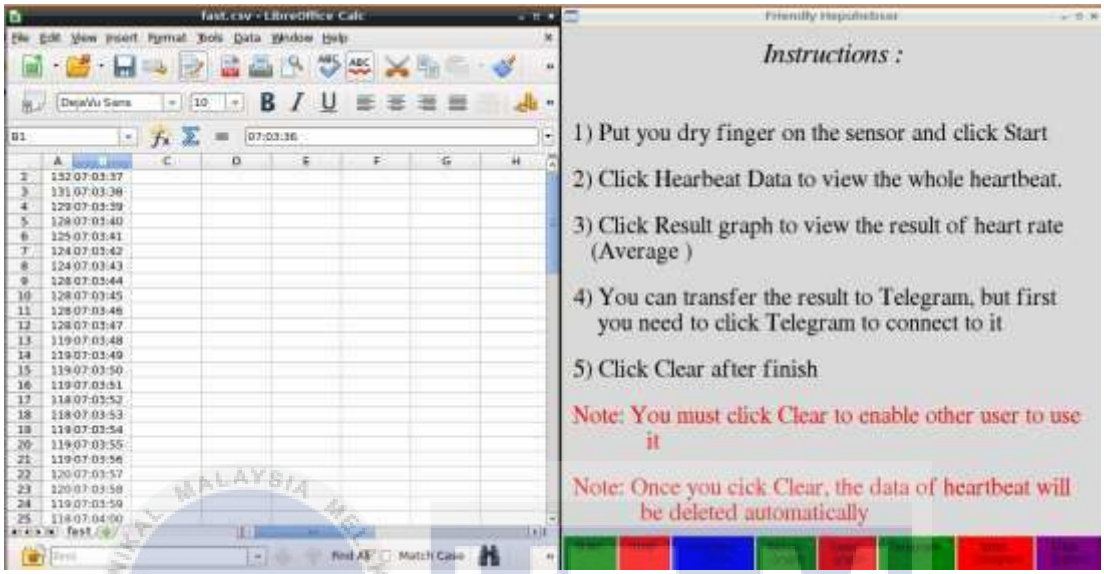

Figure 6.22: Heartbeat data of fast heartbeat.

Figure 6.22 above shows the heartbeat data of the a woman who undergo 5 minutes exercise. Most of value of bpm is lesser more than 100 bpm. Heartbeat which is higher than 100 bpm is known as Sinus Tachycardya (fast heart rate).

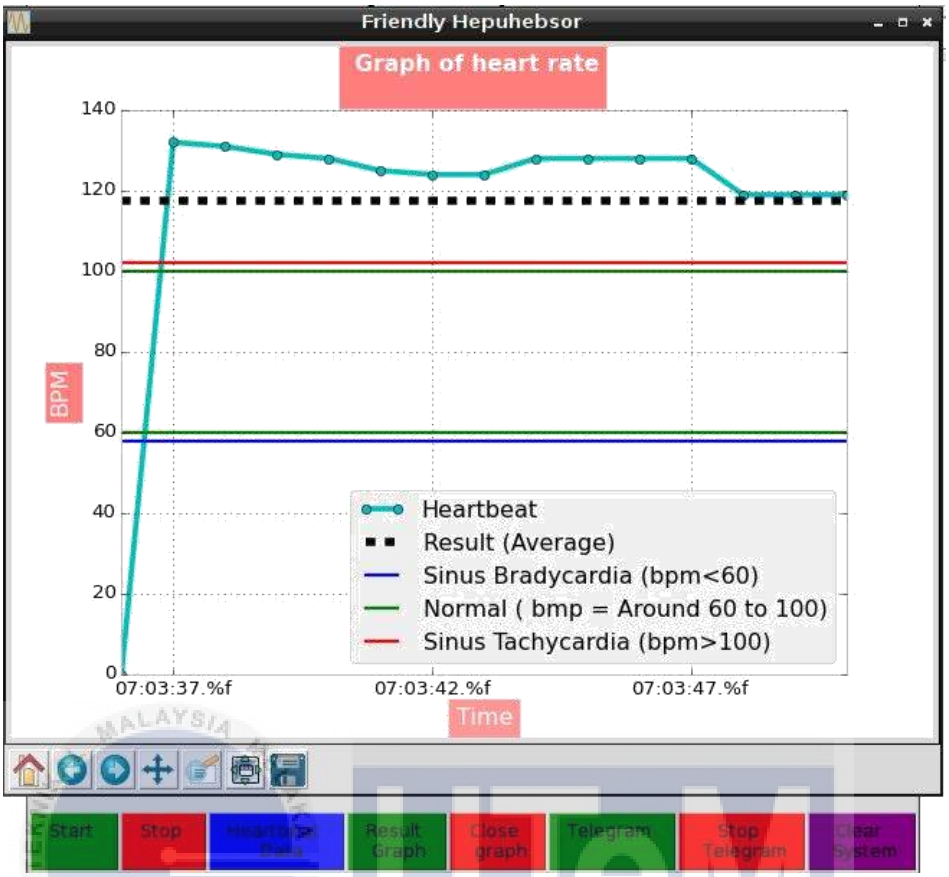

Figure 6.23: Graph of heartbeat of exercising woman.

Figure 6.23 above shows the graph of fast heartbeat of a woman who undergo 5 minutes exercise. The graph shows the average of heartbeat (black horizontal line) is greater than 100 bpm.

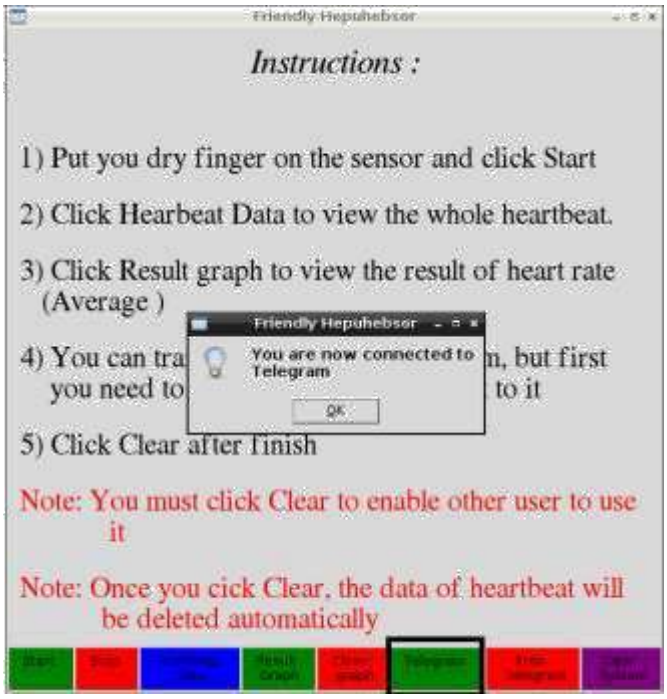

Figure 6.24: Connection to Telegram.

User can send the result and the graph that they have saved to telegram. To do so, first raspberry pi need to be connected to telegram. Once user click "Telegram", it will be connected automatically ."Connected to telegram…" will appear in the terminal as shown is figure 6.24 above. The system also has "Stop Telegram" to stop the connection.

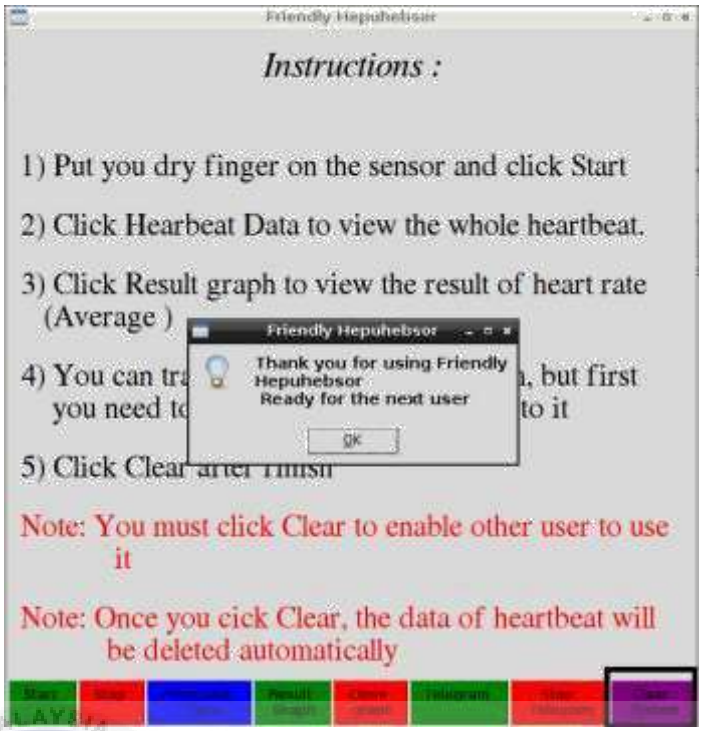

Figure 6.25: Clear the system.

The system need to be cleared. This is to ensure the graph will plot the new data. Once user click "Clear" button, the result (heartbeat.csv) will be deleted. After the clear button is clicked the terminal will tell the user that the system is ready for the next user as shown in the figure 6.25 above.

Figure 6.26: Error Handling.

Figure 6.26 above shows error handling used in the system. Error handling aso implemented on Stop, Close graph and Close telegram buttons. Once the user click the buttons, it will pop up an error.

6.5.2 Test Result on Smartphone (Telegram).

Figure 6.27: Hepuhebsor Bot

To use the Hepuhebsor Bot shown in figure 6.27 above, user need to search the bot first and start using it.Using this bot user will be able to get the saved graph, the result, symptom and precaution.

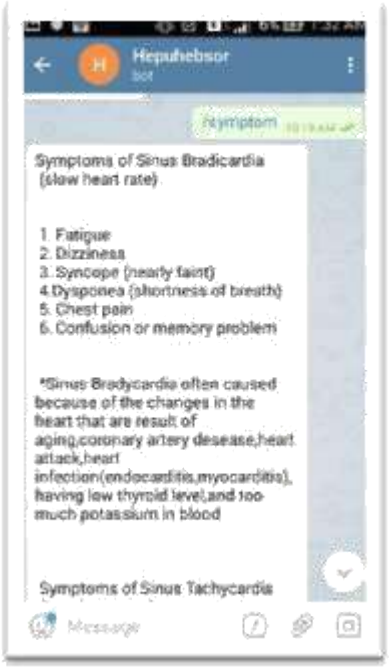

Figure 6.28: Symptom

Figure 6.28 above shows the symptom that user might have for slow and fast heart rate. User can refer to the symptom provided. The function of the symptom text is to create awareness of the user. To get the symptom text, user need to tap „/symptom" option.

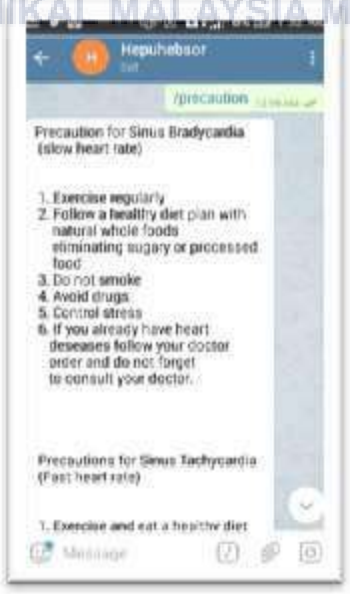

Figure 6.29: Precaution

Figure 6.29 above shows the precaution provided for user. User can refer to the precaution text as the first step to prevent something worse for health. To get the precaution text, user need to tap „/precaution‟ option.

Figure 6.30: Send heartbeat.csv.

User can transfer the heartbeat data (heartbeat.csv) from raspberry pi to telegram. This is optional, depends on the user. Heartbeat.csv contain all the heartbeat value and the time it is taken. After the file is sent, user need to download it as shown in figure 6.30 above. To get the heartbeat data, user need to tap „/heartbeat_data‟ option.

Figure 6.31: Send graph.

Figure 6.31 above shows the graph is sent to telegram from the raspberry pi. User can transfer the graph from raspberry pi to telegram. This is optional, depends on the user. The graph contain all the visualization of heartbeat.csv including the average heartbeat. To get the graph, user need to tap „/picture" or „/file" option depends on what form user use to save the graph.

Figure 6.32: View folder

Figure 6.32 above shows user can view the contain inside the folder of Hepuhebsor file. This is important if the user save the graph with different names. To view the folder, user need to tap „/folder option.

6.5.3 User Feedback.

After the prototype has been tested on 3 different people for 3 chategories which are normal, slow and fast heart rate, all of the people who tested it said it is user friendly and easy to be understood.

6.6 Conclusion

In a nutshell, all of the objectives had been achieved which are to design a heart rate analyser with simple, concise and efficient, to develop the prototype of Friendly HePuHepSor and to test the effectiveness of the prototype. In the next chapter, we will cover about the conclusion of the whole project development.

CHAPTER VII

PROJECT CONCLUSION

7.1 Introduction

In this chapter we will discuss about the summarization of the project, project contribution, project limitation and future works of the project.

7.2 Project Summarization

In this project. There are 3 objectives need to be achieved which are to design a heart rate analyser with simple, concise and efficient, to develop the prototype of Friendly HePuHepSor and to test the effectiveness of the prototype. The fist objective has been achieved in chapter 4 where user can use it by just clicking at all the buttons. Second objective also has been achieved in chapter 5 where the prototype has been developed and the software has been implemented as well. The last objective also has been achieved in chapter 6 which is testing. The prototype has been tested for 3 categories which are normal, slow and fast heart rate.

7.3 Observation on Strengths.

The strengths are, it is user friendly as stated in chapter VI which is user testing. User only need to click buttons. Besides, it has a graph view, user can monitor the graph of heartbeat using the interactive graph and user also can transfer the result to telegram.

7.4 Project Contribution

This prototype is designed for human to ease them to carry out daily work because heart rate is often influenced by physiological factors like the circadian cycle, posture, blood pressure, and physical activity. It is a better solution to save time and will be alert about what action to be taken to prevent a worse thing to happen. This product can be used by everybody instantly and everywhere. People can measure their heart rate in a friendly way (easy, inexpensive, effective).

7.5 Project Limitation

There are several limitation for this project, First is the backend of matplotlib used to plot the graph is not supported by Tkinter makes it hard to combine between the GUI and the graph plotting because Tkinter has no package to plot graph. The graph also has delay for several seconds due to Matplotlib is is not optimized for visualizations that require real-time, interactive performance, or that incorporate large data volumes[28]. To make both function well, both interface must be separated to avoid each of them freeze one another. Second is, the documentation of telegram is still lack since it is still new. The function of tap on download could not be implemented due to lack of sources. Lastly, the pulse sensor is very sensitive. It has reading when the finger is not on the sensor and not covered by anything else.

7.5 Future Works

Future work need to be done to evercome all the limitations stated in project limitation above to make the project better in terms of function and layout. The first improvement is to use monitor or high spec (RAM) laptop. Second is, to build the GUI and to plot the graph using PyQtGraph because it use the same packages to build GUI and fast real time graph plotting[28]. From the real time plotting, it can detect other than heartbeat like sugar in blood, blood pressure, and so on.

7.6 Conclusion

Finally we can conclude that all the objectives of the project has been achieved. The prototype is developed well and all the system can function as stated in the previous chapters.

REFERENCES

[1] T. K. Mishra and P. K. Rath, "Pivotal role of heart rate in health and disease," Journal, Indian Acad. Clin. Med., vol. 12, no. 4, pp. 297–302, 2011.

[2] E. F. Liu, L. Chen, and B. X. Gao, "Sinus Bradycardia : Normal Phenomenon or Risk Factor ? Evaluation Based on Recent Evidence," pp. 102–111, 2012.

[3] A. D. Narichania, J. W. Schleifer, and W.-K. Shen, "Managing the patient with episodic sinus tachycardia and orthostatic intolerance," Cardiol. J., vol. 21, no. 6, pp. 665–673, 2014.

[4] L. C. Keat, A. B. Jambek, and U. Hashim, "Heart-rate Monitoring System Design and Analysis Using a Nios II Soft-core Processor," Int. J. Electron. Telecommun., vol. 62, no. 3, pp. 283–288, 2016.

[5] P. K. Nisha and Y. Vinita, "Heart Rate Monitoring and Data Transmission via Bluetooth," Int. J. Innov. Emerg. Res. Eng., vol. 2, no. 2, pp. 99–105, 2015.

[6] S. A. R. Devi, K. R, M. P, H. R, and S. R. Prabha, "PC based Monitoring of Human Pulse Signal using LabVIEW," Ijireeice, vol. 3, no. 3, pp. 186–187, 2015.

[7] B. H. Latha, "A Tele-medicine System for Measuring Heart Rate , Blood pressure , And Drug Level Detection," vol. 2, no. 1, pp. 23–29, 2014.

[8] P. V. Janakray, "Controlling Analog Sensors using SPI (Serial Peripheral Protocol) with Raspberry Pi and MCP 3008," Ijsrd, vol. 2, no. 3, pp. 432–434, 2014.

[9] I. Engineering, "Raspberry-Pi Based Health Monitoring," pp. 7147–7154, 2015.

[10] C. Engineering and C. Engineering, "Status Observing Method Using Internet of Things," pp. 101–104, 2016.

[11] P. Navdeti, S. Parte, and P. Talashilkar, “Patient Parameter Monitoring System using Raspberry Pi,” Int. J. Eng. Comput. Sci., vol. 5, no. 16018, pp. 16018–16021, 2016.

[12] M. Bhaumik, P. Verma, R. Kant, P. Kumar, and L. Bhaskar, “Real-Time Heartbeat Rate Monitoring System using Raspberry Pi,” vol. 9, no. 41, pp. 381–389, 2016.

[13] R. Kumar and M. P. Rajasekaran, “An IoT based patient monitoring system using raspberry Pi,” 2016 Int. Conf. Comput. Technol. Intell. Data Eng., pp. 1–4, 2016.

[14] A. Ingole, S. Ambatkar, S. Kakde, and S. M. Ieee, “Implementation of Health-care Monitoring System using Raspberry Pi,” Commun. Signal Process. (ICCSP), 2015 Int. Conf. on, 2-4 April 2015. Melmaruvathur, India, pp. 1083–1086, 2015.

[15] E. Jahan, T. Barua, and U. Salma, “an Overview on Heart Rate Monitoring,” Int. J. latest Res. Sci. Technol., vol. 3, no. 5, pp. 148–152, 2014.

[16] B. Mallick and A. K. Patro, “International Journal of Science, Engineering and Technology Research (IJSETR),” Hear. Rate Monit. Syst. Using Finger Tip Through Arduino Process. Softw., vol. 5, no. 1, pp. 84–89, 2016.

[17] E. M. Dogo, F. Sado, and S. M. Adah, “Design of a Simple and Low-Cost Microcontroller Based Medicare Device for Heart Beat Monitoring,” vol. 6, no. 5, 2013.

[18] N. Ari and M. Ustazhanov, "Matplotlib in python," 2014 11th International Conference on Electronics, Computer and Computation (ICECCO), Abuja, 2014, pp. 1-6.

[19] V. R. Vatsa and G. Singh, “Raspberry Pi based Implementation of Internet of Things using Mobile Messaging Application - „ Telegram ,”” vol. 145, no. 14, pp. 17–21, 2016.

[20] Shrikrushna Khedkar, “There‟s an App for That: International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) - 2016, vol. 7, no. 2, p. e218, 2015.

[21] K. Rostyslav, S. Tkatchenko and R. Golovatsyy, "Features home automation system development based Raspberry Pi using Java ME SDK," The Experience of Designing and Application of CAD Systems in Microelectronics, Lviv, 2015, pp. 484-486.

[22] Polo, Guilherme, “PyGTK, PyQT, Tkinter and wxPython comparison”, vol. 3, no. 1,2008.

[23] Atefeh Valipour1, Reza Abbasi-Kesbi2,, “A Heartbeat and Respiration Rate Sensor Based on Phonocardiogram for Healthcare Applications” ICEE 20l7.

[24] Raluca Marinescu, Anişor Nedelcu, “Smartphone Application for Heart Rate Monitoring”, The 6th IEEE International Conference on E-Health and Bioengineering - EHB 2017.

[25] Maria Romano, Luigi Iuppariello, “Computerised Simulation of Fetal Heart Rate Signals”, The 6th IEEE International Conference on E-Health and Bioengineering - EHB 2017

[26] Roxana Al., “Stress Influence on Drivers Identified by Monitoring Galvanic Skin Resistance and Heart Rate Variability”, The 6th IEEE International Conference on E-Health and Bioengineering - EHB 2017.

[27] Hariton Costin, “Short-Term Heart Rate Variability using Wrist-Worn

Pulse Wave Monitor Compared to a Holter ECG”, The 6th IEEE International Conference on E-Health and Bioengineering - EHB 2017.

[28] Luke Campagnola, “Harnessing The GPU For Fast, High-Level

Visualization”, Proc. of the 14th Python In Science Conf. (SCIPY 2015)

Appendix A: Testing heartbeat at rest

Appendix B: Testing heartbeat after running.

Appendix C: Testing heartbeat of an athelete at rest.

Testing Strategy Questions (Medical Doctor)

Question 1: How pulse generated?

Question 2: Which group of people has normal, slow and fast heart rate?

Question 3: How to get heartbeat reading from the pulse using machine(pulse sensor) ?

Question 4: Is it possible for human to have no pulse for several minutes?

Appendix D: Testing Strategy Questions (Medical Doctor)

User Feedback

1. Do you find it's easy to use the prototype?

   Yes ☐ No ☐

2. Do you understand the graph of heart rate with all the labels?

   Yes ☐ No ☐

Appendix E: User Feedback After Using The Prototype.