Healthcare Monitoring System For Elders

BY

Kong Hong Tatt

(13ACB02977)

A Report

SUBMITTED TO

Universiti Tunku Abdul Rahman

In partial fulfillment of the requirements

For the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS) COMPUTER ENGINEERING

Faculty of Information and Communication Technology

Department of Computer and Communication Technology (Perak Campus)

January 2017
DECLARATION OF ORIGINALITY

I declare that this report entitled “Concept and Design of the Healthcare Monitoring System For Elders” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : __________________________

Name : __________________________

Date : __________________________
Healthcare Monitoring System For Elders

BY

Kong Hong Tatt

(13ACB02977)

A Report

SUBMITTED TO

Universiti Tunku Abdul Rahman

In partial fulfillment of the requirements

For the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS) COMPUTER ENGINEERING

Faculty of Information and Communication Technology

Department of Computer and Communication Technology (Perak Campus)

January 2017
ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Mr Teoh Shen Khang who has given me this bright opportunity to engage in an Embedded design project and also provide the guideline and tips along the way developing this project. A million thanks to you.
ABSTRACT

This is a project about building a Healthcare monitoring system for elders using Arduino based microcontroller. The main purpose of this project is helping the users or caretaker to tracking the health status or the situation of the elders when the users are working outside. By using this system, user can check the status of the elders via internet through the remote device includes smartphone, laptop. Besides that, the main function of this system includes enable the user to monitor the status of the elders through the server of the database and also when there are something bad happen such as the elders suddenly fall down, the body temperature abnormal, the pulse of the elders suddenly stops, then the system will send the notification such as e-mail to notify the user to take action. This may help the users to give the appropriate action or respond when the elders facing difficulties. Next, in order to continuous keep track the status of the elders, Arduino Micro had been chosen as the main microcontroller in this system due to the size is small and enable the elders can wear it using wristband on their hand. Arduino Mega board will work with Arduino Ethernet shield for purpose to store the data from the sensor attach on the Arduino Micro board to the database. On the other hand, to ensure the healthcare monitoring can fully function, some of the sensor is required such as Radio Frequency module (RF module), Pulse sensor, Accelerometer (MPU6050), Infrared Temperature sensor (TMP006) and so on.
# Table of Contents

Title ......................................................................................................................... 1  
Report Status Declaration Form .............................................................................. 2  
Title ......................................................................................................................... 3  
Acknowledgement .................................................................................................. 4  
Abstract .................................................................................................................. 5  
Table of Contents .................................................................................................... 6 - 7  
List of Figures ......................................................................................................... 8  
List of Abbreviations ................................................................................................ 9  
Chapter 1: Introduction ............................................................................................
  1.1 Background of Healthcare Monitoring System For Elders .............................. 10  
  1.2 Motivation ....................................................................................................... 11  
  1.3 Problem Statement .......................................................................................... 11  
  1.4 Project Scope ................................................................................................... 12  
  1.5 Project Objective ............................................................................................. 12  
  1.6 Impact, Significance and Contribution .............................................................. 13  
Chapter 2: Literature Review ....................................................................................
  2.1 Discussion of strength and weaknesses of existing product ......................... 14 - 16  
Chapter 3: Methodology / Technologies Involved ....................................................
  3.1 Introduction to Design Methodology ............................................................... 17  
  3.2 Waterfall Model ............................................................................................. 18  
  3.3 Spiral Model ................................................................................................... 19  
  3.4 Prototyping Model .......................................................................................... 20  
  3.5 Healthcare monitoring system design ............................................................... 21-22  
  3.6 Verification plan for Healthcare Monitoring System For Elders ..................... 23 - 32  
  3.7 Microcontroller Description ........................................................................... 33 - 37  
  3.8 Description and Functionality of each sensor ................................................. 38 - 37  
  3.9 Description of Wamp Server ........................................................................... 52 - 51  
  3.10 Webpage For Healthcare Monitoring System For Elders ................................ 56 - 63  
  3.11 Implementation Challenges and Issues ......................................................... 64
## List of Figures

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 3.2</td>
<td>Waterfall Model</td>
<td>18</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Spiral Model</td>
<td>19</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Prototyping Model</td>
<td>20</td>
</tr>
<tr>
<td>Figure 3.5a</td>
<td>Sensors use to implement the project</td>
<td>21</td>
</tr>
<tr>
<td>Figure 3.5b</td>
<td>System Block Diagram of Hardware Design</td>
<td>22</td>
</tr>
<tr>
<td>Figure 3.6a to Figure 3.6q</td>
<td>Verification plan For each sensor</td>
<td>23 - 32</td>
</tr>
<tr>
<td>Figure 3.7a</td>
<td>Front view of the Arduino Micro Boards</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.7b</td>
<td>Pin mapping of Arduino Micro Boards</td>
<td>34</td>
</tr>
<tr>
<td>Figure 3.7c</td>
<td>Front View of Arduino Mega Boards</td>
<td>35</td>
</tr>
<tr>
<td>Figure 3.7d</td>
<td>Pin Mapping of Arduino Mega</td>
<td>35</td>
</tr>
<tr>
<td>Figure 3.8a</td>
<td>Sample picture of Accelerometer (MPU6050)</td>
<td>38</td>
</tr>
<tr>
<td>Figure 3.8b</td>
<td>Approach of x, y, z axis measured through MPU6050</td>
<td>38</td>
</tr>
<tr>
<td>Figure 3.8c</td>
<td>Front and back view of Pulse sensor</td>
<td>42</td>
</tr>
<tr>
<td>Figure 3.8d</td>
<td>Sample picture of Infrared Temperature Sensor (TMP006)</td>
<td>45</td>
</tr>
<tr>
<td>Figure 3.8e</td>
<td>Sample picture of 433Mhz Transmitter and Receiver</td>
<td>49</td>
</tr>
<tr>
<td>Figure 3.10a to Figure 3.10i</td>
<td>Webpage For Healthcare Monitoring System For Elders</td>
<td>56 - 63</td>
</tr>
</tbody>
</table>
### List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IoT</td>
<td>Internet of thing</td>
</tr>
<tr>
<td>RF module</td>
<td>Radio Frequency module</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System For Mobile Communication</td>
</tr>
<tr>
<td>I²C</td>
<td>Inter-Integrated Circuit</td>
</tr>
<tr>
<td>BPM</td>
<td>Beat Per Minute</td>
</tr>
<tr>
<td>IMAP</td>
<td>Internet Message Access Protocol</td>
</tr>
</tbody>
</table>
Chapter 1: Introduction

1.1 Background of Arduino based healthcare monitoring system for elders

Internet of Things (IoT) is the network to interconnected devices which are embedded with software, sensor, network connectivity and electronic device that enable them to exchange and collect the data to making them responsive (Rahul, 2015). Besides that, IoT is also an architectural framework which allows data exchange and integration between the actual world and computer system over existing network infrastructure. Throughout the year, there are many product built based on the IoT concept for example smartphone, laptop, healthcare device and others electronic devices. Healthcare is defined as improvement of health through the diagnosis, treatment and prevention of the diseases by the health professional such as doctor, nurse, dentist and others medical professional people (Merriam-Webster, 2015). Healthcare is very important for the elders people due to the immune system of the elders becomes weaker with the increasing of the age and the chances to affected by the diseases is higher. So, in order to take care those elders who faces healthy problem, healthcare monitoring system was introduced. Besides that, with the help of IoT devices such as healthcare monitoring system, the user can tracking the status of the elders anywhere through the internet and can give a proper respond when there is something happen on the elders.
1.2 Motivation

Nowadays, to overcome the increasing of the daily expenses, most of the young teenagers went out for work from early morning until late evening or night. Therefore, they have to leave their parent or elders that need to take care in the house while they are working. However, for some of the elders who are facing health problem such as, high blood pressure, heart attack, facing difficulties on movement or others diseases, if there is something happen on them, we may hard to notify since we are working outside. So, Arduino based healthcare monitoring system may help solve the problem with the feature below:

- The Arduino Healthcare monitoring System will transfer the status of the elders on the database and the users can monitor it through internet.
- The Arduino Healthcare monitoring System able to detect the body temperature, acceleration force (to detect is that the elders fall down) of the elders.
- The Arduino Healthcare monitoring System send the notification to the users if something bad happen on the elders through the e-mail.

1.3 Problem statement

Due to some reason, the user cannot take care the elders in the home daily, by introduce the healthcare monitoring system, the user can monitor the status of the elders through the server and take the action based on the data shown on the server. With the development of technology, there are many healthcare monitoring product exist in the market nowadays. However, the existing product has several function or feature that cannot provide which includes:

- Some healthcare monitoring system is not portable and wearable.
- Some healthcare monitoring system need more power to operate
1.4 Project Scope

The scope of this project is to develop a system that able to detect the status of the elders anytime and enable the user to monitor the status of the elders through log on to database server. So, to achieve the function above, Healthcare Monitoring System is specifically design out. The coverage of this project are limited to the elders at home and the area of the project will be within the range near the home due to the radio frequency module has limited received and transmit data range that send by the sensor. This project required some of the programming skills to handle the Arduino coding, some knowledge on database and architecture skill in order to complete this project.

1.5 Project Objective

Healthcare Monitoring System is not unique in the market, there are some of the similar product exist in the market, however there are some of the improvement can be done. The main objective of this project includes:

- To implement reliable and wearable healthcare monitoring device to the elders.
- Enable the user to monitor the status of the elders in the homes remotely through website.
- Able to send notification such as E-mail will to the user mobile phone when there are something bad happen on the elders.
1.6 Impact, Significance and Contribution

There are many existing healthcare monitoring for elders had been provided in the market, but there is still some limitation and improvement can be found on the existing product. With the development of technology, the power consumption of the device is very concern. Most of the user willing to choose those product with lower power consumption compare to the product that can perform the same function but consume more power. Therefore, when design the system, using an appropriate microcontroller to build the project is very important.

Besides that, the healthcare monitoring system for elders must be based on real time. This is because, the main purpose of this system is acknowledge the user the current status of the elders, if the system is not operate on real time, when the elder accidentally fall down or face others difficulties, it may causes the user cannot give an appropriate help during the elder need help from others. So, the healthcare monitoring system must be built based on real time.

Last but not least, the healthcare monitoring system for elder must build in small scale, this is to ensure the elders can wear the monitoring system on hand by using wristband. Therefore, the system must portable and can be easily wearing on the hand.

Next, when the elders facing the difficulties such as suddenly fall down or others, the healthcare monitoring system must acknowledge the user through the e-mail or trigger the alarm for seeking help from others people. Therefore, the system must have the function where it will send e-mail or trigger the alarm when the elders facing the difficulties.

Last, this project is to ensure the problem stated as above can be solved. By implement this project, the problem where the users need to leave the elders who facing health problem in the house while they are working can be solved and the elders can get the appropriate help from their family member and neighbor when they need help. As a conclusion, this system can bring many advantages to the user and also the elders.
Chapter 2: Literature Review

2.1 Discussion of strength and weakness of existing product

Based on the research, there are some of the similar products related to this current project. One of the products which known as “ATM Based Remote Healthcare Monitoring System” develop by (Shibu.J and Ramkumar.R, 2012). By comparing to the research product, there are some of the strength and weakness that can found in this product.

First, the microcontroller using in this system is ATmega 16. The microcontroller will interface with the basic sensor that needed in the healthcare monitoring system. A smart card will provide to each user who will use this system, the smart card contain the basic information of the user (Shibu.J and Ramkumar.R, 2012). When the user swiped the smartcard and using the ATM based remote healthcare monitoring system, the system will display the result out on the PC monitor and the health information will also transfer to the mobile phone through Global System for Mobile Communication (GSM). Moreover, the health information of the user will also store in the database which is own by the healthcare management for future references (Shibu.J and Ramkumar.R, 2012). However, this system is not suitable for those people who facing movement difficulties because the system is manage by the healthcare management, so each time the user want to use this system he or she may need went to the specific location that provide this services. The solution for this problem can be encountered by develops one system that completely belongs to the user who purchases this product. Moreover, this system is not portable since it need connect to the power supply in order to operate. The solution to solve the not portable issue is using a smaller microcontroller like Arduino Micro board to function as the main controller of the system and also replace the power supply with a battery to operate the system.

Another product that related to this project is “Smartphone Based Continuous Monitoring System for Home-bounds Elders and Patients” by (Rajesh Kannan Megalingam, Goutham Pocklassery, Vivek Jayakrishnan, Galla Mourya &
Athul Asokan Thulasi, 2014). By comparing to the research project, there are some strength and weakness that can found from this project.

The strength of this research project include they can use smartphone to communicate with a wearable health monitoring system and also provide a simple user interface for the elder to operate (Rajesh Kannan Megalingam, Goutham Pocklassery, Vivek Jayakrishnan, Galla Mourya & Athul Asokan Thulasi, 2014). This product concern in the fall tilt detection, it will alert the caretaker if the elder lays in bed in the same position for more time or fall down unconscious and it also provide a mechanism to request for assistance (Rajesh Kannan Megalingam, Goutham Pocklassery, Vivek Jayakrishnan, Galla Mourya & Athul Asokan Thulasi, 2014). However, this product transmit the data receive from the sensor to the smartphone via the Bluetooth module. Although Bluetooth is easy to use and low power but the data transmission speed is depend on the distance between Bluetooth transmitter and the device, the closer the distance the faster the receiving rate. Imagine if the elders stay far from the Bluetooth module, the data may not successful transmitted and this may lead to the user may not receive the actual data that should transmitted from the sensor. This can be solve by apply Radio Frequency Module (RF module), the data transmission can be done within the data transmission range of the RF module and also the distance problem that exists on Bluetooth can be solved with increase the power of transmitter of RF module to increase the data transmission range.

Besides that, there are one more research project that related to this project which known as “Healthcare monitoring system using WIFI as a communication medium on ARM7” by (G.Vijaya Lakshmi, B.Suresh Ram & T.Ramakrishna, 2013). By comparing to this product, some strength and weakness can be point out from this project for further discussion.

In this research project, the strength of this project include the healthcare monitoring system is wearable, check the vital sign of the patient or the elders, store the data receive from the sensor to the Access database and using the wireless sensor WI-FI as a real-time
monitoring system to monitor the health of the patient in the hospital or the elders in the home (G. Vijaya Lakshmi, B. Suresh Ram & T. Ramakrishna, 2013). The main function of the WI-FI in this project is transmitted the data receive from the sensor that wear on the elders or patient to the user or doctor PC for monitoring purpose (G. Vijaya Lakshmi, B. Suresh Ram & T. Ramakrishna, 2013). However, the power consumption, data transmission range and the price of the WI-FI module is higher compared to the RF module (Charalampos, 2012). In order to develop cheap and reliable healthcare monitoring system for elders, choosing the right module to construct the device is very important.

At last, a better healthcare monitoring system should be developing based on the energy efficiency, portable, reliability, ease to use and low cost with high functionalities.
Chapter 3: Methods / Technologies Involved

3.1 Introduction to Design Methodology

In order to produce a high functionalities system, select the appropriate design methodology is very important. System development methodology is a framework that is used to plan, structure and control the process of developing an information system (Software development methodologies, 2016). There are three basic System Development Methodology that common apply when designer wish to choose which model to design the system which includes Waterfall model, Spiral Model and Prototyping Model. So, to choose the model that best fit to this project, we may discuss the three model on subtopic which includes 3.2 Waterfall model, 3.3 Spiral model and 3.4 Prototyping model. The importance of Design Methodology in designing the system includes the following criteria:

- Can estimate the design cost for a system.
- Reliable, precise in term of functionality and able to provide a product in term of quality.
- Function correctly without having major Error.
- Able to avoid deliver the buggy system to the customer.

Next, regarding the basic hardware and software that needed in this project will be discussed on subtopic 3.5. Besides that, prototyping model will be chosen as a model to implement in this project.
3.2 Waterfall Model

Waterfall model is one of the popular version of system development life cycle model for software engineering. Waterfall model describe a method of development that is inflexible and linear. Besides that, waterfall development has distinct goal for each development phase where it need to wait the phase before it to completed before the next phase is started and once started there is no way to reverse back to the previous phase (Software development methodologies, 2016). So, this model is not suitable to be used on this project because as the development future develops, it cannot loop back to the previous phase to retrieve the error and changing the content. Figure 3.2 shows the waterfall model.

![Figure 3.2 Waterfall Model](image-url)
3.3 Spiral Model

Spiral Model also known as spiral lifecycle model which concentrate on early reduction and identification of project risks. Spiral project normally starts on small scale, then follow by risk explores, make a plan to handle the risks and then make decision to proceed to next step of the project or not (Software development methodologies, 2016). This spiral model is not suitable to healthcare monitoring project because spiral model is focus on expensive, large and complicated project while this project is targeting simple, cheaper and reliable design. Figure 3.3 show the spiral model.

![Spiral Model Diagram](image-url)
3.4 Prototyping Model

Prototyping model is a model that developed the system based on the currently known requirement. Prototyping is suitable for the system where there is no existing system or manual process for determining the requirement. Prototype is referring to early approximation of final system. First, prototype is develop, follow by tested it, reworked until an acceptable prototype is finally achieved (ISTQB exam certification, 2016). Prototyping model is suitable in this healthcare monitoring system design because not all requirements are known in detail. This model can loop back to the previous model and find out the bug or error when the error exists. Therefore, prototyping will be used to implement this project. Figure 3.4 shows the prototyping model.

![Prototyping Model Diagram](image-url)
3.5 Healthcare Monitoring For Elders System Design

The hardware that will be used in this project include Arduino Micro board, Arduino Mega board, Arduino Ethernet Shield, some sensor such as DHT11 sensor (temperature and humidity sensor), Accelerometer, Radio Frequency module, Pulse sensor and battery is using to supply the power to the Arduino Micro board. The Arduino Micro board will be wear by the elder and the sensor will collect the information of the elders and transfer it to the database through Radio frequency module. To interface with the hardware, Arduino IDE – A Software which can interact with Arduino board was needed. Besides that, Arduino Mega board will operate with Arduino Ethernet Shield to build the database for storing the data obtain from the sensors. The user may connect the server of the database to monitor the status of the elders by using remote devices such as smartphone, laptop and so on. Figure 3.5a shows the sensors that will use to implement the healthcare monitoring system and Figure 3.5b show the simple system block diagram of hardware design on this project.

![Sensors Diagram]

Figure 3.5a, from left TMP006 (infrared temperature sensor), pulse sensor, Radio Frequency module (433Mhz RF module) and MPU6050 (accelerometer)
Function of the sensor:

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Function of the sensor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature sensor</td>
<td>Measure the temperature of the elders.</td>
</tr>
<tr>
<td>Pulse sensor</td>
<td>Measure the heart beat rate of the elders</td>
</tr>
<tr>
<td>RF module</td>
<td>Consist of transmitter and receiver to transfer and receive the data</td>
</tr>
<tr>
<td>Accelerometer</td>
<td>To detect the acceleration force of the elders, for purpose of detect is that the elders is fall down.</td>
</tr>
</tbody>
</table>

Block Diagram and Description of Healthcare Monitoring System

Figure 3.5b, Healthcare monitoring system block diagram.

- Figure 3.5b shows the overall system block diagram of the healthcare monitoring system for elders.
- Arduino Micro and Radio Frequency transmitter will collect data from each sensor then pass it to Arduino Mega with Radio Frequency receiver which function as base station to receive the data.
- Then the Ethernet shield will attach with the Arduino Mega, then connect it to the modem using Ethernet cable
- After that, connect the laptop to the modem using Ethernet cable for purpose of host the wamp server.
- At last, all the data will upload to the database and user will able to view those data on the website.
3.6 Verification plan for Healthcare Monitoring System

- In order to make sure the healthcare monitoring system can operate correctly some of the test plan was created to justify it.

Test plan 1 : Testing Temperature sensor

- Verify the correctness of the TMP006 sensor by using the cases as below:
- Case 1 :
  - Input : Measure the temperature of cold water
  - Expected Output : The temperature of the cold water should be less than normal surrounding temperature.

- Experimental Result :

  - From the figure 3.6a above, it shows the surrounding temperature measured from the room without air conditioner.
  - The temperature obtained is in between 30.00°C to 32.00 °C
- Then continue test the temperature sensor by giving some input to the sensor. Below are one of the example:

![Figure 3.6b](image)

- From the figure 3.6b above, in order to test whether the reading read from the sensor is accurate. Some cases is design to verify it.
- First of all, put the cold water on the top of the temperature sensor, then observe the result of the temperature through open the serial monitor of the Arduino IDE.

![Figure 3.6c](image)

- From the figure 3.6c above, the cold water temperature is in between 20.00°C to 22.00°C.
- So, the assumption that set by the case 1 where the temperature of the cold water should less than the normal surrounding temperature is justify.
Case 2:

- **Input**: Measure the temperature of the human body part
- **Expected Output**: The temperature of the human should be more than normal surrounding temperature.

**Experimental Result**:

- From the figure 3.6d above, in order to measure the temperature of the hand, place your hand on top of the TMP006, then the sensor will measure the skin temperature and pass it to the Arduino Micro.
- Next, observe the result through the serial monitor of the Arduino IDE.
- From the figure 3.6e above, the temperature of my hand is around 34.00°C to 35.00°C. Although the temperature obtained is not 36.00°C (normal temperature of human) but the temperature obtained is more than the normal surrounding temperature.

- By going through the cases 1 and 2 above, it can verify that the TMP006 functional correctly.
Test plan 2: Testing pulse sensor

- Input: Measure the heartbeat rate of the elder through the pulse sensor and compare to the heartbeat rate measure through mobile application.
- Expected Output: The heartbeat rate measured from 2 platform should almost same.

Experimental Result:

- From the figure 3.6f above, using the mobile application (S Health) provide by Samsung Note 5, it can measure the heartbeat rate by putting the finger at the back next to the camera lens.
- Then, the sensor at the back of the mobile phone will start to measure the heartbeat rate (Figure 3.6g) by detecting the pulse of the finger.
Figure 3.6h

- From the figure 3.6h above. It show the way to measure the heartbeat rate using the pulse sensor

Figure 3.6i

Figure 3.6j

- The Figure 3.6i show the result of the heartbeat rate and Figure 3.6j show the result measured through the mobile application of Mobile phone. Both of the result shows the heartbeat rate is approximate 80bpm.
- So, as conclude the pulse sensor able to measure the heartbeat rate correctly.
Test plan 3: Testing Accelerometer MPU6050

- **Input**: Lift and lower down the accelerometer and observe the z-axis value
- **Expected Output**: When the accelerometer is lift up, the z-axis value should be in positive (increase) while when lower down the z-axis value should in negative (decrease).

- **Experimental Result**:

  - From the figure 3.6k above, when lifting the accelerometer up the z-axis value increase from 15544 to 16156.
From the figure 3.6m above, when lower down the accelerometer, the z-axis value decrease from 19024 to 13668.

By gone through this test plan, we can conclude that the accelerometer provide the correct result and function correctly.
Test plan 4: Testing Radio Frequency Module

- Input: Increase the distance between the transmitter and receivers
- Expected Output: The data will hard to receive as the distance increases.
- Experimental Result:

- From the Figure 3.6o, as the distance between the Radio Frequency Transmitter and Receiver is very near, the data easily received by the receiver. The sample result shown in the Figure 3.6p below.
- From the Figure 3.6q, as the distance between the Radio Frequency Transmitter and Receiver increase, the receiver hard to receive the data from the transmitter.

- As shown in the Figure 3.6q, the receiver didn’t receive any data from the transmitter.
- This maybe because the less voltage is supply to the transmitter due to too many sensor share the same 5V voltage from the Arduino Micro Vcc and causes the transmission length decrease.
- Therefore, this limited the transmission range of the transmitter in this project to around less than 10 cm.
3.7 Microcontroller Description

In this project, Arduino Micro and Arduino Mega 2560 is using to develop the healthcare monitoring system for elders. Arduino Micro will act as a transmitter station to transmit the data gather from the sensors to the Arduino Mega 2560 with Ethernet shield which act as a receiver station that will upload the data received to the database for purpose of able the user to view the data through the webpage created.

**Arduino Micro**

The main reason to choose Arduino Micro is due to the size of the board is small hence it can easily wear on the body the elders such as wear the system on the hand and the operating voltage of the Arduino board is 5V where is suitable for system aim for low power consumption. The microcontroller of the Arduino Micro board is based on ATmega32U4(Arduino, 2016). ATmega32U4 is a 8-bit microcontroller with 32K bytes of In-system Programming(ISP) Flash and USB controller. Figure 3.7a below shows the front view of the Arduino Micro Board while Figure 3.7b below shows the overall pin mapping for the Arduino Micro Board.

![Arduino Micro Board](image)

Figure 3.7a shows the front view of the Arduino Micro Boards
Figure 3.7b shows the pin mapping of the Arduino Micro Boards
**Arduino Mega 2560**

The main reason to choose the Arduino Mega for the base station to receive data in this project is due to the mysql library using a lot of memory when compile the code, so Arduino Mega with bigger memory: 128 KB flash memory is suitable using in this project. Furthermore, the Arduino Mega also can power up using 5V which consider as low power also is suitable using in this project. Figure 3.7c below show the front view of the Arduino Mega 2560 boards and Figure 3.7 d show the overall pin mapping for Arduino Mega 2560 boards.

![Arduino Mega 2560 Image](image1)

Figure 3.7c shows the front view of the Arduino Mega 2560 Boards

![Arduino Mega 2560 Pin Mapping Image](image2)

Figure 3.7d shows the pin mappingss of the Arduino Mega 2560 Boards
Chapter 3: Circuit Approach of Healthcare Monitoring System

Circuit Approach of Healthcare Monitoring System (Transmitter Side)

- Pulse Sensor
- Accelerometer (MPU6050)
- Temperature sensor (TMP006)
- RF Transmitter

Diagram showing connections between Pulse Sensor, Accelerometer, Temperature sensor, RF Transmitter, and Arduino Microcontroller.
Circuit Approach of Healthcare Monitoring System (Receiver Side)

- Arduino Mega attach with Ethernet Shield
- RF Receivers

Connections:
- Vcc
- Ground
- Data
3.8 Description and Functionality of each sensor

i) Accelerometer (MPU6050)

![Accelerometer (MPU6050)](image)

Figure 3.8a shows the sample picture of the Accelerometer (MPU6050)

- MPU6050 sensor contains the accelerometer and gyroscope in a single chip.
- MPU6050 accelerometer able to capture the x, y, z channel at the same times.
- In order to able interface with the Arduino micro, **I²C** bus is using to communicate between sensor and Arduino board.
- **I²C** is referring to inter-integrated circuit. It commonly use in intra-board communication between processors and microcontroller in short distance.
- The main function of the accelerometer is using to detect the elder fall down.

![Orientation of Axes of Sensitivity and Polarity of Rotation](image)

Figure 3.8b shows the approach of x, y, z axis measured through the accelerometer

- In order able to detect fall down condition, we need to look at the z-axis value, where the z-axis is refer to the height of the object.
- If the differences between the z-axis are too high, it may indicate the elder is fall down due to the differences between the height is higher.

**MPU6050 code description**

```c
#include<Wire.h>
const int MPU_addr=0x68; // I2C address of the MPU-6050
int16_t AcX, AcY, AcZ;
void setup(){
  Wire.begin();
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x68); // PWR_MGMT_1 register
  Wire.write(0); // set to zero (wakes up the MPU-6050)
  Wire.endTransmission(true);
  Serial.begin(9600);
}
```

- In order to get the raw value of the x,y,z axis through the accelerometer, PCanaddress of the MPU6050 is needed for communication between sensor and Arduino board.

- 0x68 is the PC address of the MPU6050, it can be found on the MPU6050 datasheet.

- PWR_MGMT_1 is a register to allow user configure the device into power mode ad clock source. Besides that, this register also can resetting the entire device.

- During setup, the PWR_MGMT_1 register is in sleep mode, in order to wake it up, assign 0 the register.

- After the PWR_MGMT_1 wake up, now the MPU6050 able to take in the sample data through the accelerometer.
Then by using loop(), it enable the accelerometer continuously to take in the raw data.

In order to start the accelerometer measurements, configure the code to able it request the $x$, $y$, $z$-axis value through the register below:

1) ACCEL_XOUT – 16-bit two complement value. Function as stores the Latest X axis accelerometer measurement.

2) ACCEL_YOUT – 16-bit two complement value. Function as stores the Latest Y axis accelerometer measurement.

3) ACCEL_ZOUT – 16-bit two complement value. Function as stores the Latest Z axis accelerometer measurement.

After grab the data of the $x$, $y$, $z$ axis value, pass those value to the respective variable where include AcX for store X-axis value, AcY for store Y-axis value, AcZ for store Z-axis value. Last display the result of the $x$, $y$, $z$ axis result out respectively.

- In order able to detect the condition of elder fall down. One way to do it is, store the previous $z$-axis value into a variable then using the current $z$-axis value minus the previous $z$-axis value.
- The sample formula is like below:
  \[
  \text{Detect Fall down} = \text{latest z-axis value} - \text{previous z-axis value}
  \]

- If the elder is fall down, the fall down variable will obtained a negative value, so using this negative value obtained to compare with the threshold of the value(-2000). 

- The threshold of the fall down value set in this system is -2000, the reason to choose the condition less than -2000 is for testing purpose because the system able to detect the condition of elder fall down more easily.
ii) **Pulse Sensor**

![Pulse Sensor Image]

*Figure 3.8c shows the front and back view of Pulse sensor*

- The operating voltage of pulse sensor is between 3.3V to 5V.
- Figure 3.8c shows the front and back view of the pulse sensor.
  - Purple wire : Signal wire, connect to pin A0 of Arduino micro
  - Red wire : Vcc
  - Black wire : Ground
- The main function of the pulse sensor is measure the heart beat rate of the elders by detecting the pulse rate on the fingertip.
- The normal heart beat rate for the elders is around 60 – 100 BPM which same with the normal adult people.
- Some of the variable that include to calculate the heartbeat rate include :
  i) Signal – Receive the Analog Signal from the sensor every 2ms
  ii) IBI – Interbeat Interval. It holds the time interval between beat with 2ms Resolution
  iii) BPM – Refer to Beats per Minutes. It holds heart beat rate value, get the heart beat rate value by taking the average of previous 10 IBI value.
Pulse sensor Code Description

ISR(TIMER3_COMPA_vect){
  cli();
  Signal = analogRead(pulsePin); // read the Pulse Sensor
  sampleCounter += 2; // keep track of the time in mS with this variable
  int N = sampleCounter - lastBeatTime; // monitor the time since the last beat to avoid noise

  - The code above is for purpose of read the analog value getting from the
    pulse sensor then pass it to Signal variable for calculate Beats per Minute
    (BPM). Every 2 milliseconds, the analog value read from the sensor will
    pass in to the Signal variable.

  if (N > 250){ // avoid high frequency noise
    if ((Signal > thresh) && (Pulse == false) && (N > (IBI-3)+3) ){
      Pulse = true; // set the Pulse flag when think there is a pulse
      digitalWrite(blinkPin, HIGH); // turn on pin 13 LED
      IBI = sampleCounter - lastBeatTime; // measure time between beats in mS
      lastBeatTime = sampleCounter;
      if(firstBeat){ // if it's the first time we found a beat, if firstBeat == TRUE
        firstBeat = false;
        return;
      }
      if(secondBeat){ // if this is the second beat, if secondBeat == TRUE
        secondBeat = false;
        for(int i=0; i<9; i++)
        {
          rate[i] = IBI;
        }
      }
    }
  }

  - The code above is to look for the heartbeat. During start up, the firstBeat
    variable is initialized as true and secondBeat is initialized as false.

  - When the firstBeat is detected, the code above will throw the IBI value away
    due to maybe it is noise value and may affect the result of the BPM.

  - During the second time detected the heartbeat, if the secondBeat variable
    detected the heartbeat, it will change the secondBeat flag to true and pass the
    value to the rate array for calculate BPM.
The loop will loop 9 times for purpose of sum up the oldest IBI value stored in the rate array and then store the result in runningTotal variable.

Then, pass in the latest IBI value into the position 9 of the array and sum runningTotal value with the latest IBI value in position 9.

Follow by find the average IBI value by divide the runningTotal with 10.

Lastly, the BPM value is calculated by using how many beats can fit into one minutes (60000ms) divide by the average value of runningTotal.
iii) **Infrared Temperature Sensor (TMP006)**

![Infrared Temperature Sensor (TMP006)](image)

**Figure 3.8d shows the sample picture of the TMP006**

- The operating voltage of TMP006 is about 3.3V to 5V and also can be used with battery application so this sensor is suitable to use in this project due to low power consumption.
- TMP006 communicate with Arduino Micro via I²C protocol.
- TMP006 able to detect the object temperature without contact the sensor by measure the infrared energy being emitted from the surface of the object.
- In order for TMP006 to function with the Arduino Micro, SDA and SCL pin of the TMP006 need to connect to the SDA and SCL pin of Arduino Micro respectively.
- The SI unit for measure the temperature is in degree Celsius(°C).
TMP006 sensor code description

```c
#include <stdint.h>
#include <math.h>
#include <Wire.h>
#include "I2C_16.h"
#include "TMPO06.h"

uint8_t sensor1 = 0x40; // I2C address of TMP006, can be 0x40-0x47
uint16_t samples = TMPO06_EXAMPLE; // # of samples per reading, can be 1/2/4/8/16

void setup()
{
  Serial.begin(115200);
  Serial.println("TMP006 Example");
  config_TMP006(sensor1, samples);
}

void loop()
{
  float object_temp = readObjTemp(sensor1);
  Serial.print("Object Temperature: ");
  Serial.print(object_temp); Serial.println("°C");
}
```

- The code above is for purpose of take the reading of the object temperature through the TMP006 sensor.
- First configure the PC address of the TMP006 which by default is 0x40 if only using one TMP006 sensor.
- Then set the amount of sample per reading to either 1/2/4/8/16 to get the object temperature.
- After that calculate the target object temperature by going through the equations below:

**Equations for calculating target object temperature**

- To calculate the target object temperature, some calculation that consists a series of equation is required in order to calculate the target object temperature \( (T_{OB}) \) in Kelvins.
- Equation1:
  \[
  S = S_0 \left[ 1 + a_1 (T_{DIE} - T_{REF}) + a_2 (T_{DIE} - T_{REF})^2 \right]
  \]
- The equation 1 is to represent the thermopile sensor sensitivity and how it changes over temperature.

- \( S_0 \) = Primary calibration factor (typical values between \( 5 \times 10^{-14} \) and \( 7 \times 10^{-14} \))

- \( T_{DIE} \) = Raw Die Temperature value read from the TMP006 sensor

- \( T_{REF} \) = Constant value (298.15)

- \( a_1 \) = Constant value \((1.75 \times 10^{-3})\) and \( a_2 = (-1.678 \times 10^{-3})\)

- **Equation 2**:

\[
V_{OS} = b_0 + b_1(T_{DIE} - T_{REF}) + b_2(T_{DIE} - T_{REF})^2
\]

- Equation 2 shows the offset voltage arises due to the TMP006 slight self-heating, caused by the package non-zero thermal resistance and the small operational power dissipation (1mW) in the device.

- \( B0 = \) constant value \((-2.94 \times 10^{-5})\) and \( B1= (-5.7 \times 10^{-7})\),

- \( B2 = \) constant value \((4.63 \times 10^{-9})\)

- **Equation 3**:

\[
f(V_{OBJ}) = (V_{OBJ} - V_{OS}) + c_2(V_{OBJ} - V_{OS})^2
\]

- Equation 3 is using to calculate thermopile sensor Seebeck coefficients and how these coefficients change over temperature.

- Seebeck coefficients is refer to heat conversion into electricity at the junction of different types of wire

- \( C2 \) is a constant value \( C2=13.4\)

- **Equation 4**

\[
T_{OBJ} = \sqrt[4]{T_{DIE} + \frac{f(V_{OBJ})}{S}}
\]

- Equation 4 relates the radiant transfer of IR energy between the target object and TMP006 conducted heat in the thermopile in the TMP006.
• Solve the equation and finally get the temperature of the object in the kelvins.

• After that convert back the object temperature to celcius by use the object temperature = object temperature - 273.15

```cpp
void takeReading()
{
    object_temp = readObjTemp(sensor1);
}

void displayResult()
{
    Serial.print("Current heartbeat rate : "); Serial.print(BPM); Serial.println(" BPM");
    if (BPM > 100)
        Serial.println("heart rate abnormal");
    Serial.print("Object Temperature: ");
    Serial.print(object_temp); Serial.println("°C");
    if (object_temp > 35)
        Serial.println("Patient In Fever");
}
```

- In order for detect is that the elder is in fever, test the object temperature is that exceed 35°C (35°C is just for testing purpose, when fever, temperature should exceed 36°C).
- If exceed 35°C, notify the user about elder is in fever
iv) Radio Frequency Module (RF Module 433Mhz)

- The working Voltage of the RF Module is between 3V to 12 V. In this project, 5V is using to power up the sensor.
- The transmission range of the 433 RF Module is between 20 to 200 meters.
- Besides that, the transmission range also depend on the Voltage supply, the higher the voltage supply, the higher the transmission range.
- RF transmitter responsible to transmit the data gather from other sensor such as pulse sensor, accelerometer, temperature sensor that attach on Arduino Micro.
- RF receiver responsible to receive the data from the Arduino Micro which act as a base station that gather the data.

Figure 3.8e shows the sample picture of the 433 Mhz RF Module sensor
433Mhz Radio Frequency Module code description

Transmitter Site

```c
void transmitdata()
{
    char str["dbafafaafaf0""] = "dbafafaafaf0";  // a string with 12 byte
    sprintf(str, "%d_%d_%d", BPM, (int)object_temp, flag); // combine the bpm, temperature and the fall_down flag to string
    // Serial.println(str);
    driver.send((uint8_t*)str, strlen(str));  // send the string to receiver
    driver.waitForPacketSent();
}
```

- The code above is for purpose transmit wirelessly the data gathered from the pulse sensor, temperature sensor and accelerometer to the Arduino Mega board.
- First, append the BPM, temperature, the fall down flag to the string that created just now.
- Then send the string through the driver.send() function and driver.waitForPacketSent() function to make sure the previous data successfully transmit.
Receiver Site

```c
void loop()
{
  /t.update();
  uint8_t buf[12];
  uint8_t buflen = sizeof(buf);
  String v1, v2, v3;
  int pos1 = 0, pos2 = 0, cnt = 0;
  String str;
  if (driver.recv(buf, &buflen)) // Non-blocking
  {
    Serial.write(buf, buflen);
    Serial.println();
    str = (char*)buf;
    for (int i = 0; i < buflen; i++)
    {
      if (str.substring(i, i + 1) == "_")
      {
        if (cnt == 0) pos1 = i;
        else pos2 = i;
        cnt++;
      }
    }
    v1 = str.substring(0, pos1);
    v2 = str.substring(pos1 + 1, pos2);
    v3 = str.substring(pos2 + 1);
    BPM_r = v1.toInt();
    Temp_r = v2.toInt();
    fall_down_ind = v3.toInt();
    Serial.print("BPM: "); Serial.println(BPM_r);
    Serial.print("Temp: "); Serial.println(Temp_r);
    Serial.print("Flag: "); Serial.println(fall_down_ind);
  }
}
```

- In receiver site, the `driver.rec()` will receive the data from the transmitter.
- Then split the string that receive from the transmitter.
- The sample string will be `78_30_0`, the 78 refer to BPM, 30 refer to temperature and 0 indicate the fall down flag.
- Then use a loop to split the string out and assign to the substring to respectively variable.
3.9 Description of WAMP server

Wamp server is a software that using for web development. Wamp server support web application such as PHP, Apache and a MySQL database. Besides that, the Wamp server software also provide the PhpMyAdmin that able to manage the database easily.

Code description of connect the Ethernet shield to Wamp Server

```c
char user[] = "root";
char password[] = "root";
byte mac[] = { 0x00, 0x0c, 0x10, 0x00, 0x00, 0x00 };
byte ip[] = { 192, 168, 1, 110 }; // ip In lan (that's what you need to use in your browser. ("192.168.1.178")
byte gateway[] = { 192, 168, 1, 1 }; // internet access via router
byte subnet[] = { 255, 255, 255, 0 };
IPAddress server_addr(192, 168, 1, 107); //server ip, pc ip

void setup()
{
    Serial.begin(115200); // Debugging only
    Ethernet.begin(mac, ip, gateway, subnet);
    delay(1000);
    Serial.print("Ethernet Shield IP is: ");
    Serial.println(Ethernet.localIP()); //library to set ip of arduino
    Serial.println("Connecting...");
    if (my_conn.mysql_connect(server_addr, 3306, user, password)) //code to connect to database
        Serial.println("Connection Success!");
    else
        Serial.println("Connection Failed.");
    analogReference(DEFAULT); //for analog
}
```

- First, setup the user and password to login the database.
- In order to connect to the wamp server, an ip address with the same subnet (red pointer) was assign to the Ethernet shield.
- Then, assign the pc ip address to the server_addr (blue pointer)
Steps to connect Ethernet shield to Wamp server

- After that, launch the wamp server and put it online.

- Then logon to the 127.0.0.1, the wamp server homepage and click on the phpmyadmin to enter the database.
After that click on the mysql.

Then, search for the user table.
In order to enable permission of the Ethernet shield able to connect to wamp server. Insert the ip address that assign to the Ethernet of the user table.

Then now, the Ethernet shield able to connect to the wamp server.
3.10 Webpage For Healthcare Monitoring System For Elders

- Figure 3.10a above show the sample webpage for the Healthcare Monitoring System For Elders.
- User able to view all the data received by the receiver on this webpage.
- By clicking on the submit button, all the data in that store in database will show out.
- Figure 3.10b above shown the sample output of the data after click on the submit button.
- After that, all the data that store in database will show out on the table above.
- The first column on the table above is refer to the Beat Per Minute (Heartbeat of the Elders), second column refer to the Temperature of the Elders, third column is an indicator to check when the Elder Fall Down and the last column shown the timestamp that the data insert to database.
- If the Fall_Down_Indicator is 1, an alert message will pop out in order to notify the user.

- Besides that, if the BPM > 100, the row color of the BPM will change to red color to notify the user the BPM of the Elders is too high.

- Follow by, if the Temperature of the elder is exceed 30 degree Celsius, the row color of the temperature will change to blue color to notify the user abnormal temperature of the elder.

- Last, if the Fall_Down_Indicator is 1, the row color will also change to green color to notify the user elder is fall down.
Moreover, the Healthcare Monitoring System Webpage also provide a send email function to the user when the elder is fall down, BPM abnormal and Temperature abnormal.

- The email will send to the user G-mail account to notify the user there are something bad happen on the elder.

**Send email php code description**

```php
require 'Mailer/PHPMailerAutoload.php';

$mail = new PHPMailer;

$mail->isSMTP(); // Set mailer to use SMTP
$mail->Host = 'smtp.gmail.com'; // Specify main and backup SMTP servers
$mail->SMTPAuth = true; // Enable SMTP authentication
$mail->Username = 'kivensai@gmail.com'; // SMTP username
$mail->Password = '********'; // SMTP password
$mail->SMTPSecure = 'tls'; // Enable TLS encryption, 'ssl' also accepted
$mail->Port = 587; // TCP port to connect to
$mail->SetFrom('kivensai@gmail.com', 'kiven');
$mail->addAddress('kivenkong0719@gmail.com'); // Add a recipient
$mail->Subject = 'Message From Healthcare Monitoring System!!';
$mail->Body = 'Caution, something bad happen on Elders!!!';

if(!$mail->send()) { echo 'Message could not be sent.'; } else { echo 'Message has been sent!'; }
```

- In order to use the send email function, phpmailer function is needed.
- First, set up the Username and the Password for the Gmail account that responsible to send out the email.
Then, add the recipient address to receive the email.

After that, go to the setting of the sender email and enable IMAP (Internet Message Access Protocol) to make sure the php code able to send mail using this sender email.
- As we can observe from the Figure 3.11d, the email will send from the kivenzai@gmail.com account to the kivenkong0717@gmail.com.

- From the Figure 3.12d, as we open the email from the smartphone, we can notice that the email is already send to recipient email to notify the user when something bad happen on elder.
Recheck the data with selected date range

- Besides that, if the user wish to recheck the data within the date range, user may key in the range of date and time he would like to check on the Start and End column that shown on the Figure 3.10e above.
- The input format for the date range is (YYYY-MM_DD) (HH-MM-SS). The sample is 2017-04-08 14:00:00.
- Due to the webpage is still under developing, the only format that accepted is the format that describe above.
- For example, if user wish to check the available data from 2017-04-08 14:00:00 to 2017-04-08 15:00:00, he/she just key in the Start and End column and check on the submit button.
After click on the submit button, user able to view all the result that fall on the range between the 2017-04-08 14:00:00 to 2017-04-08 15:00:00.
3.11 Implementation Issue and Challenges

Along the way to develop the Healthcare Monitoring System For Elders, there are some issues and challenges exist that increase the difficulties of implementation of this project. First, sometimes the pulse sensor unable to detect the correct heartbeat rate by measure the pulse from the finger. Because of this problem, it make the verification part where to justify the pulse sensor is function correctly become harder. Next is the timing conflict between the sensor, this issues causes the reading taken from each sensor incorrect and in order to solve this issues, timer is introduce to display the result collected. Third, the data transmission between the Arduino Micro and Arduino Mega are facing some issues where the data collected in Arduino Micro not able to transmit real time to the Arduino Mega due to some hardware or software issue, so in order to solve the issue more time was needed to spent in solve the problem.
Chapter 4: Conclusion

This proposal delivers the full idea, implementation and design of the healthcare monitoring system for elders. In this project, the main objective is able to deliver a low power, wearable healthcare monitoring system for elder and ensure the user can keep track the status of the elders through the webpage even though they are not around in home to take care the elders. Next, the motivation to develop this project is because of when something bad happen on the elder and none of his family member is around, this project able to notify and seek help from the elder family member through the notification send by healthcare monitoring system. Although there are many existing healthcare Monitoring system exist in the market, however there are some limitation for those system which include some are high power consumption and not wearable. So in order to solve the problem stated just now, choosing the low power microcontroller and sensor is one of the ways to solve the problem.

Last but not at least, the main objective of FYP 2 is more concentrate on the data transmission part between the transmitter and receiver and the implementation of webpage in order to enable the user to view the result remotely. However, one of the function that provided by this project which is able to send email to notify the user is still developing. For the future, this project will more concentrate to develop the website, some extra function will provided by the webpage that able the user to monitor the status of the elder remotely through the website more effectively.

As a conclusion, this project will be complete into more functionality and reliability system in the future.
References


References

