Faculty of Electronics and Computer Technology and Engineering

DEVELOPMENT OF A HEALTH MONITORING SYSTEM USING ARDUINO FOR REMOTE PATIENT

NUR FARHAH ATIQ BINTI JAAFAR

Bachelor of Electronics Engineering Technology (Telecommunications) with Honours

2024

DEVELOPMENT OF A HEALTH MONITORING SYSTEM USING ARDUINO FOR REMOTE PATIENT

NUR FARHAH ATIQ BINTI JAAFAR

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electronics Engineering Technology (Telecommunications) with Honours

Faculty of Electronics and Computer Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI KEJUTERAAN ELEKTRIK DAN ELEKTRONIK

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : Development of a Health Monitoring System using Arduino for Remote Patient

Sesi Pengajian :2023/2024

Saya Nur Farhah Atiq binti Jaafar mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. 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**	

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: PT 333, Kampung Jelutong Kota, Lorong Paksin, Jalan Salor 15100, Kota Bharu

(COP DAN TANDATANGAN PENYELIA)

TS ZAHARIAH BINTI MANAP
Pensyarah Kanan
Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer (FTKEK)
Universiti Teknikal Malaysia Melaka (UTeM)

Tarikh: 25 / 1 / 2024

Tarikh: 22-02-2024

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled “Development of a Health Monitoring System using Arduino for Remote Patient” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Student Name : NUR FARHAH ATIQ BINTI JAAFAR

Date : 25 / 1 / 2024

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electronics Engineering Technology (Telecommunications) with Honours.

Signature :

Supervisor Name : TS ZAHARIAH BINTI MANAP

Date : 22-02-2024

DEDICATION

To my parents, whose boundless love, encouragement, and sacrifices have been my pillars of strength throughout this academic journey. Your unwavering support has fueled my pursuit of knowledge and shaped the person I am today.

ABSTRACT

In our daily lives, health is the most important aspect. To do the everyday tasks effectively, good health must be maintained. However, people nowadays do not have much time to prioritize their own health. A health monitoring system that automatically records health condition is necessary to provide users with information about their health status in real time. In conventional health monitoring systems, individuals need to meet their doctors regularly to evaluate their condition. However, the introduction of Internet of Things (IoT) technologies has facilitated the transition of healthcare from in-person consultations to telemedicine. Through IoT, machine-to-machine connection allow for immediate initiation of preprogrammed tasks that enables the health consultation to be made remotely in a more efficient way. The goal of this project is to develop an IoT-based health monitoring system that tracks a remote patient's vital signs in real time. This system uses two sensors: a pulse sensor and temperature sensor, to monitor body temperature and heart rate information. These sensors interface with a NodeMCU-32 controller board, enabling wireless data transfer to an IoT platform known as ThingSpeak. Additionally, Mit App Inventor is employed for data visualization, data display and storage. To test the developed system, measurements are compared with readings from commercially available devices. It was found that the accuracy of body temperature measurements is 1.21%, and heart rate measurements are 2.8%. The results affirm the suggested approach's reliability and accuracy across diverse individual health characteristics.

ABSTRAK

Dalam kehidupan harian kita, kesihatan merupakan aspek yang paling penting. Untuk menjalankan tugas-tugas harian dengan berkesan, kesihatan yang baik perlu dijaga. Namun, pada masa kini, orang ramai tidak mempunyai banyak masa untuk memberi keutamaan kepada kesihatan mereka sendiri. Sistem pemantauan kesihatan yang merekod keadaan kesihatan secara automatik adalah penting bagi menyediakan pengguna dengan maklumat tentang status kesihatan mereka dalam masa nyata. Dalam sistem pemantauan kesihatan konvensional, individu perlu berjumpa dengan doktor mereka secara berkala untuk menilai keadaan kesihatan mereka. Namun, dengan pengenalan teknologi Internet of Things (IoT), telah memudahkan peralihan penjagaan kesihatan daripada perundingan secara peribadi kepada teleperubatan. Melalui IoT, sambungan mesin ke mesin membolehkan permulaan segera tugas yang telah diprogramkan yang membolehkan perundingan kesihatan dibuat dari jauh dengan cara yang lebih cekap. Matlamat projek ini adalah untuk membangunkan sistem pemantauan kesihatan berasaskan IoT yang mengesan tanda-tanda vital pesakit jauh dalam masa nyata. Sistem ini menggunakan dua penderia: penderia nadi dan penderia suhu, untuk memantau suhu badan dan maklumat kadar jantung. Penderia ini berinteraksi dengan papan kawalan NodeMCU-32, membolehkan pemindahan data tanpa wayar ke platform IoT yang dikenali sebagai ThingSpeak. Selain itu, Mit App Inventor digunakan untuk visualisasi data, paparan data, dan penyimpanan. Untuk menguji sistem yang dibangunkan, pengukuran dibandingkan dengan bacaan daripada peranti komersial. Didapati bahawa ketepatan pengukuran suhu badan ialah 1.21%, dan pengukuran kadar denyutan jantung ialah 2.8%. Keputusan tersebut mengesahkan kebolehpercayaan dan ketepatan pendekatan yang dicadangkan pada pelbagai ciri kesihatan individu.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Ts. Zahariah binti Manap for her precious guidance, words of wisdom and patient throughout this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support which enables me to accomplish the project. Not forgetting my fellow colleague, thank you for the willingness of sharing her thoughts and ideas regarding the project.

My deepest appreciation goes to my parents, and family members for their love and prayer during the period of my studies. An honourable mention also goes to those who have always been with me for all the motivation and understanding. And thanks for always supporting me.

Finally, I would like to thank all the staff at the Faculty of Electronics and Computer Technology and Engineering (FTKEK), fellow colleagues and classmates, the faculty members, as well as other individuals who are not listed here for being co-operative and helpful.

TABLE OF CONTENTS

		PAGE
DECLARATION		
APPROVAL		
DEDICATIONS		
ABSTRACT		**i**
ABSTRAK		**ii**
ACKNOWLEDGEMENTS		**ii**
TABLE OF CONTENTS		**iv**
LIST OF FIGURES		**vii**
LIST OF ABBREVIATIONS		**viii**
LIST OF APPENDICES		**ix**
CHAPTER 1	**INTRODUCTION**	**10**
1.1	Background	10
1.2	Addressing Health Technology in Current Issues	10
1.3	Problem Statement	11
1.4	Project Objective	12
1.5	Scope of Project	13
CHAPTER 2	**LITERATURE REVIEW**	**14**
2.1	Introduction	14
2.2	Understanding Health Monitoring Systems during the pandemic in the literature	14
2.3	Overview of a Health Monitoring System	14
2.4	Types of Micro-controller	16
	2.4.1 Raspberry Pi	16
	2.4.2 Arduino Uno	17
	2.4.3 Node MCU	17
2.5	The Technology used in the Previous Project	17
	2.5.1 Wi-Fi Techonology	17
	2.5.2 LoRa Technology	17
	2.5.3 Zigbee	18
2.6	Applications and remote sensing	21
	2.6.1 Thingspeak	21
	2.6.2 Ubidots	21
2.7	Communication Protocol	21
	2.7.1 MQTT Protocol	22
	2.7.2 CoAP Protocols	23
	2.7.3 LAN Protocols	23

2.7.4 GSM Protocolos 23
2.8 Comparison of Previous Project 24
2.9 Summary 29

CHAPTER 3 METHODOLOGY 30
3.1 Introduction 30
3.2 Selecting and Evaluating Tools for a Sustainable Development 30
3.3 Project Planning and Flow Chart 30
3.3.1 Block Diagram of the Proposed System 33
3.3.2 Scematic diagram of the Proposed System 34
3.4 Tools 34
3.4.1 Software Specification 35
3.4.1.1 Thingspeak 35
3.4.1.2 Fritzing 36
3.4.1.3 Arduino IDE 37
3.4.1.4 MIT App Inventor 38
3.4.2 Hardware Specification 39
3.4.2.1 DB18S20 Temperature Sensor 39
3.4.2.2 Pulse Sensor (SEN-11574) 40
3.4.2.3 LCD I2C 42
3.4.2.4 NodeMCU32 43
3.4.2.5 Battery 44
3.4.2.6 Resistor 45
3.5 Limitation of proposed methodology 47
3.6 Summary 47

CHAPTER 4 RESULTS AND DISCUSSIONS 48
4.1 Introduction 48
4.2 Results and Analysis 48
4.2.1 Prototype System Design 48
4.2.2 Software Implementation 49
4.2.2.1 Data Storing 51
4.2.3 Comparing data with existing health monitoring system 52
4.2.3.1 Heart beat measurement by SEN-11574 pulse sensor and Smart Watch 52
4.2.3.2 Body temperature measurement by DS12B20 and thermometer 57
4.3 Discussion 61
4.4 Summary 62

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS 63
5.1 Conclusion 63
5.2 Potential for Commercialization 64
5.3 Future Works 64

REFERENCES 66

APPENDICES 71

LIST OF TABLES

TABLE	TITLE	PAGE
Table 1.1:	Comparison between different specifications of network technology	20
Table 2.1:	Comparison of among technologies used in related previous project	25
Table 3.1:	Pulse Sensor Pinout Descriptions	41
Table 3.2:	Specifications of NodeMCU 32	43
Table 3.3:	Specifications of Lithium ion 3.7V 5000mAh 18650 Li-ion Battery	45
Table 3.4:	Specifications of 4.7k Ohm resistor	46
Table 4.1:	Heart beat measurement by SEN-11574 pulse sensor	52
Table 4.2:	Heart beat measurement by Smart Watch	53
Table 4.3:	Average of error rate for heart beat measurement	54
Table 4.4:	Body temperatured measured by DS18B20	57
Table 4.5:	Body temperature measured by a thermometer	58
Table 4.6:	Average of error rate for temperature measurement	59

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1: Process of MQTT protocol		22
Figure 3.1: Flowchart of the proposed system		32
Figure 3.2: Block diagram of the Health Monitoring System		33
Figure 3.3: Schematic diagram of the Health Monitoring System		34
Figure 3.4: Logo of Thing Speak		35
Figure 3.5: Logo of Fritzing		36
Figure 3.6: Logo of proteus		37
Figure 3.7: Logo of MIT App Inventor		38
Figure 3.8: Pin configuration		40
Figure 3.9: SEN-11574 Pulse Sensor		41
Figure 3.10: I2C_LCD pinout		42
Figure 3.11: NodeMCU-32 pinout		43
Figure 3.12: Lithium ion 3.7V 5000mAh 18650 Li-ion Battery		44
Figure 3.13: Color code of 4.7k Ohm resistor		46
Figure 4.1: Complete hardware circuit communication		49
Figure 4.2: ThingSpeak platform created		50
Figure 4.3: MIT Apps Inventor created		50
Figure 4.4: Excel showing data collected from ThingSpeak		51
Figure 4.5: Comparison of heart rate mesurement		56
Figure 4.6: Comparison of body temperature		61

LIST OF ABBREVIATIONS

V	-	Voltage angle
IoT	-	Internet of Thinking
Wi-Fi	-	Wireless Fidelity
GPS	-	Global Positioning System
LM35	-	Linear Monolithic 35
DC	-	Direct Current
C	-	Celcius
IDE	-	Intergrated Development Environment
ECG	-	Electrocardiogram
WAN	-	Wide Area Network
LAN	-	Local Area Network
MQTT	-	Message Queuing Telemtry Transport
USB	-	Universal Serial Bus
Mb/s	-	Megabit per second
G	-	Giga
Hz	-	Hertz
MIT App Inventor	-	Massachusetts Institute of Technology App Inventor

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
Appendix A	Ghantt Chart	71
Appendix B	Code	72

CHAPTER 1

INTRODUCTION

1.1 Background

Some hospital patients require daily medical supervision from the facility. The majority of those who require medical monitoring are those who must have follow-up care and admitted to the hospital before being allowed to return home. An electronic device that measures vital indicators such as body temperature and heart rate are used nowadays to carry out monitoring responsibilities. Equipment (for measuring vital signs) is usually carried with the patient during monitoring, but in certain situations it may be fitted to each patient. Every patient must undergo periodic vital sign checks performed by nurses and doctors. Therefore, monitoring tasks require more concentration from nurses and doctors and may cause them fatigue.

The health monitor project is an essential initiative aimed at improving patient care and outcomes while enhancing healthcare efficiency and reducing the cost of healthcare services. With the increasing demand for remote healthcare services, the health monitor project is well-positioned to play a crucial role in the future of healthcare delivery.

1.2 Addressing Health Technology in Current Issues

Health is always placed as the priority in any technical advancement the human race develops. Health care has become increasingly vital, as seen by the recent coronavirus pandemic that largely wrecked China's economy. Additionally, as Malaysia's temperature rises, the Malaysian Meteorological Department has issued a Level 1 alert, or yellow alert, for four districts in Peninsular Malaysia where the daily maximum temperature will be between 35°C and 37°C for at least 3 consecutive days. People must be aware of the risks that come with heat waves and be prepared to recognize the warning signs and symptoms of heat exhaustion and heat stroke. The symptoms included temperature, breathing, and heartbeat. Additionally, climate change makes already existent disparities in health and medical care worse. It is usually a better idea to use remote health monitoring technologies to keep an eye on these patients in these places where the disease is widespread. So, the current fix for it is an Internet of Things (IoT)-based health monitoring system.

1.3 Problem Statement

Health monitoring is essential for enhancing medical outcomes and assisting in the early diagnosis and treatment of disorders. However, developing nations frequently experience specific challenges that avoid successful health monitoring projects. In the traditional healthcare system, people are required to visit clinics or medical centers regularly for medical check-ups, which is less effective and time-consuming. The high expense of medical care and long waiting times may discourage people from scheduling routine medical check-ups. People will benefit from a health monitoring system that collects and tracks their health conditions in real time by not having to spend money or time traveling to clinics and medical facilities unless necessary.

In addition, the security of the health system is essential to protect the user's privacy. Concerns about the privacy of medical records may cause people to avoid seeking medical care in sensitive areas. The health information gathered is stored in the cloud by smart wearable devices like the Apple Watch and Samsung Galaxy Watch, by using this wearable devices, users can check their own health. Users can take advantage of excellent features with cloud storage without having to worry about storage periodic maintenance. However, compared to users of local storage, cloud users are more exposed to problems including theft, privacy, and information leakage to outside parties. As the information will only be accessed by authorized users, storing sensitive health data within the health system will increase the security and privacy of the system and lessen the likelihood of data leakage.

1.4 Project Objective

The project aims to develop an IOT-based health monitoring system that provides real time health insights for remote patients. The objectives are as follows:

a) To develop a sensor circuit that measures and collects real-time health parameters data.

b) To process the recorded data and display meaningful insights into the health condition on a timely basis.

c) To evaluate the performance of the developed health monitoring system like comprehensive examination of its technical, functional and user-related aspects to ensure its effectiveness in monitoring health.

1.5 Scope of Project

The goal of the project is to develop a hardware and software-based IoT-based health monitoring system. The prototype system includes data processing and sensors. The prototype enables authorized users to view the user's health information via the Internet. Additionally, by accessing the database that contains all of the user's health data, the authorized user can easily trace back to previous health data.

a) Data Collection and Transmission: The sensors would collect data at the project site and communicate it to a centralized monitoring system. Wi-Fi wireless communication methods can be used to do this.

b) Data Processing: All the information collected would be processed and saved in a central database or cloud storage for further analysis.

c) Real-time Monitoring: The monitoring system would give users access to real time information on the project's performance and status. It would continuously collect data from the sensors, process it, and display it in an automatic Thingspeak.

d) Analytics and Reporting: The monitoring system may include analytical tools to draw valuable conclusions from the information collected.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Technology innovation has completely changed the healthcare sector, resulting in the creation of innovative health monitoring tools. Wearable sensor integration with NodeMCU32 and the ThingSpeak platform to track vital signs like heart rate and temperature pressure is one such area of study. This review of the literature intends to investigate and evaluate the current research on the application of this comprehensive system for health monitoring. The storage stage and the data retrieval stage are the two main stages of how the system operates. Data is updated and stored at the storage step for later use. Retrieve data from the cloud during the data retrieval stage. With an authenticated user's request, the cloud server can share information. Based on the previous article that had related to this project.

2.2 Understanding Health Monitoring Systems during the pandemic in the literature

Through a review of the literature on health monitoring sensing technologies, this section will explore how health monitoring sensing projects can contribute to our understanding of remote health monitoring to support contactless health treatment and monitoring during the pandemic. Research has shown that health monitoring sensing technologies can provide valuable data on changes in temperature, heart rate, and blood pressure that are indicative of the impacts of current issues, such as Corona Virus and so on. For example, studies have used health monitoring sensing data to demonstrate that average temperatures and disease have been increasing over time, leading to more extreme temperature or blood pressure events. Additionally, health monitoring sensing technologies can be used to track patient health to measure for preventing diseases and detect potential risks early. By integrating health monitoring sensing data with other environmental and socio-economic datasets, researchers and policymakers can better understand the complex interactions between current issues, human activity, and ecosystem health. As such, health monitoring sensing projects have the potential to play a key role in addressing the global challenge with current issues.

2.3 Overview of a Health Monitoring System

A thorough analysis of the health monitoring system may be found in this article, the authors described the technologies, techniques, and algorithms used for a health monitoring systems. The most common and widely used technologies as mentioned in the articles are NodeMCU, Ubidots platform, Wi-Fi, and various component of health monitoring [1]. The author focused on the GPS tracker technology to locate the user [2]. From this article, the author offer an insightful examination of the current state of research on Health Monitoring System using IoT, drawing upon a wide range of scholarly works to provide a comprehensive understanding of the subject. The goal of this project is to create a system that uses LM35 and pulse sensors, Arduino, ESP8266, and ThingSpeak to monitor body temperature and heart rate. and employed IoT systems to gather relevant data, receiving data wirelessly over the internet [3].

For future development, the authors provide a platform of S-Health for reminding patients of their dosage timings through an alarm ringing system. This application used

JSON links for visualizing data patients on a smartphone [4]. There has been emphasis on the various wireless technologies and the advantages of using those technologies for faster communication. Some of article, suggests an Internet of Things-based system for real-time health monitoring that can assess, track, and report on people's health both online and offline [5].

2.4 Types of Micro-controller

2.4.1 Raspberry Pi

The author stated that the microcontroller Raspberry Pi is used for processing the data extracted from the wearable sensors. The Raspberry Pi acts as a server and sends the data on a specific URL, which can be monitored on any mobile device connected to the same network [6]. Some of article used Raspberry Pi to connect all other sensors and works at a 5V DC power supply. It is a small, low-cost, single-board computer that works as a sensor node and controller in the proposed healthcare monitoring system [7].

The Raspberry Pi 3 Model B microcontroller is used as a hardware platform to implement the Health Monitoring System using IoT. Its function is to communicate with the sensors and collect the medical information of the patient, such as heart rate and pulse rate. This author was mentioned use Raspberry Pi 3 to send an emergency alert to the patient if the sensor value exceeds the threshold data, which is to enable the collection and communication of patient data through the Health Monitoring System using IoT [8].

2.4.2 Arduino Uno

This article used the microcontroller Arduino Uno to interface with sensors that measure body temperature and heart rate. Arduino Uno based on ATMega 328 This device is used and features 14 digital I/O pins. With a built-in USB peripheral, the ATmega 16U2 transmits serial data to the main CPU. The simulation is done on Arduino IDE software [9].

2.4.3 Node MCU

NodeMCU is an open-source development board that integrates Wi-Fi module. Based on this article, they choose Node MCU as a microcontroller which is a developer kit and open-source firmware that help in the creation of IoT applications. It can be programmed using the Arduino IDE and is based on the ESP8266 Wi-Fi module. The Node MCU is used to collect real-time ECG signals, temperature, and values of oxygen saturation from human beings and send this data to the cloud server using Ubidots [10].

2.5 The Technology used in the Previous Project

2.5.1 Wi-Fi Techonology

The authors stated that Wi-Fi technology can effectively use to monitor patient health status. And the power consumption of the Wi-Fi module (ESP8266) can be reduced as much as possible [11]. This article used Wi-Fi technology for wireless data transmission on an IoT platform [3]. And other article used Wi-Fi technology to integrate the Raspberry Pi platform to the database[12]. In this studies provide wireless connectivity between the heart rate monitoring system and the MQTT broker running on a Raspberry Pi. The ESP8266 Wi-Fi module is used on the Arduino microcontroller to enable wireless communication between the heart ratesensor and the MQTT broker [13]. This article was stated that Wi-Fi technology allows doctors and medical professionals to monitor the health parameters of patients at regular intervals without the need for frequent visits to the hospital [14]. Besides that, this author used Wi-Fi to transmit ECG data from the Raspberry Pi to a CloudMQTT broker using IEEE 802.11 (WLAN) protocol[15]. This technology allows devices to connect to the internet or other devices without the need for physical cables. Based on this article, the author used Wi-Fi to enable wireless transmission of the patient's health data to the doctor's mobile device [16]. This article was stated that WiFi in the proposed patient health monitoring system is to send the data collected by the temperature and heartbeat sensors to a web server (wireless sensing node) using the internet. This allows the patient's health to be monitored remotely and alerts to be sent in case of any abrupt changes in the patient's heart rate or body temperature [17].

2.5.2 LoRa Technology

LoRa is a wireless communication technology designed for long-range, low-power communication between devices in the context of the Internet of Things (IoT). From this article stated in the world of the Internet of Things, LoRa technology is widely used to transport data over the air from sensors with short transmission range over a distance of a few tens of kilometers [18]. LoRaWAN offers good reliability in challenging communications environments and enables relatively long-distance communications with low-energy consumption and LoRaWAN is a low-cost, low-power, and secure communication network that enables the communication between e-health sensors at the patient's side and a dashboard at the doctor's side [19]. This author was mentioned that LoRaWAN consumes ten times less power than GPRS/3G/4G and has a maximum data rate of 50kbps (uplink) and 50kbps (downlink)[17]. The use of these technologies helps in the collection and transmission of data acquired from sensors for health monitoring purposes.

2.5.3 Zigbee

Zigbee is a wireless communication technology designed for short-range, low-power, and low-data-rate applications. It operates on the IEEE 802.15.4 standard and is commonly used in various wireless sensor and control network applications. On this article was stated that Zigbee technology is a low-power wireless network and is suitable for utilized in healthcare monitoring systems to keep an eye on senior residents in Ambient Assisted Living environments as well as patients in hospitals. It is an IEEE 802.15.4 standard created to alleviate network traffic problems in Personal Area Networks (PANs), like ZigBee [20]. Besides that, Zigbee had been employed in many other industries, but the home automation business has received a lot of attention since electrical and electronic items are used to automatically regulate them. Zigbee wireless communication technology is used to transmit data from the monitoring terminal to 12 the coordinator node. The WiFi module of the node then transmits the data to the host computer wirelessly. This allows for real-time display and storage of relevant data on the designed host computer software, based on a seminal review [21]. Furthermore, Zigbee is used in wireless body area networks (WBANs) for health telemonitoring systems due to its guaranteed delay requirement. However, the paper does not provide further details about the specific use of Zigbee in the proposed patient health monitoring system [17].

Table 1.1: Comparison between different specifications of network technology

Network technologies	Topology	Radio frequency	Range	Data rate	Energy consumption
Zigbee	Mesh	868.3MHz, 902-928MHz, 2.4GHz	100m	0.02 – 0.25 Mb/s	Low
Wi-Fi	Star	2.4GHz	Less than 1km	11Mb/s – 10Gb/s	High
LoRaWAN	Star or peer to peer	Between 860 and 1020MHz	Less than 30km	Up to 50Kb/s	Very low

2.6 Applications and remote sensing

Services are provided in a smart city to address problems with residential surroundings. Hospital services and health monitoring services make up the practical systems. With numerous studies and applications, the idea of remote sensing is frequently employed in healthcare monitoring services. These some applications and remote sensing that had done in previous project.

2.6.1 Thingspeak

ThingSpeak ia an Internet of Things (IoT) platform that allows users to collect, analyse, and visualize data from various connected devices. It provides a cloud-based infrastructure to store and retrieve data from sensors and other IoT devices in real-time. Based on this article, they used a technology called Thing Speak is being developed to save and access data from a patient's body[22]. Besides that, this author was using Think Speak Platform to configure activities based on real-time data and submit info to the cloud[3]. In this article, stated that Think Speak provides an API for data analysis and visualization. The API allows users to access the data stored on the cloud platform and perform various operations such as data analysis, visualization, and sharing [14].

2.6.2 Ubidots

The author offers Ubidots platform for developers that enables them to quickly collect sensor data from any internet-capable device, transform it into useful information, and transfer data to the cloud [1]. Ubidots was choosen to received message and get notification send through text message [23]. Based on this a stated that Ubidots is a cloud-based platform that provides a simple and powerful way to store, analyze, and visualize data from connected devices [23].

2.7 Communication Protocol

A communication protocol is a collection of guidelines that control how information is sent between devices. In order to ensure successful communication, it specifies the format, time, and sequencing of data delivery. Protocols allow for the connection, data sharing, and teamwork of devices, fostering interoperability and compatibility. Ethernet, TCP/IP, Wi-Fi, and Bluetooth are some instances of commonly used protocols. They are necessary for creating effective communication across a range of sectors and fields.

2.7.1 MQTT Protocol

The author preferred use MQTT protocol to transmitted data to Node-Red and ThingSpeak for monitoring [12]. MQTT protocol use in the proposed real-time remote patient monitoring system based on the Internet of Things (IoT) to ensure the accuracy of the vital real-time signal [24]. It is used to transmit the vital real-time signal from the proposed method to the website. It is used to condense the subject matter and enables transparent data sharing and quick identification.

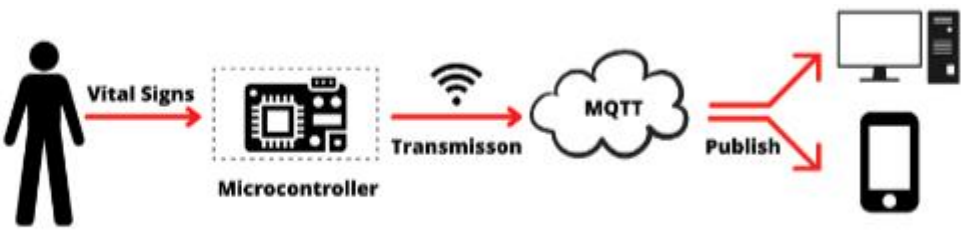


Figure 2.1: Process of MQTT protocol

The author used to evaluate its performance based on its capability, efficiency, communication method, message delay, and Quality of Service (QoS) [25]. Based on this journal, they were chosen MQTT protocol because it requires a small code footprint and is easy to scale the sensor nodes, making it very suited for the condition where the network bandwidth is limited [13]. In this article used MQTT protocol to transmit ECG data from the sensor to a Cloud MQTT broker, which can be accessed remotely by healthcare centers, hospitals, or treating doctors [15].

2.7.2 CoAP Protocols

Constrained Application Protocol (CoAP) is a lightweight and efficient communication protocol designed for the Internet of Thing (IoT) and constrained networks. The author preferred used CoAP used in IoT as an alternative to http in hardware-restricted devices to make it easy to utilise with these devices and it compare it with MQTT protocol to choose the most suitable messaging protocol for e-health systems [25].

2.7.3 LAN Protocols

Local Area Network (LAN) is used to manage the monitoring terminal's parameter synchronisation requires. The hardware component of the lower computer integrates the detection of ECG and pulse to develop a multi-parameter monitoring system of vital signs and realises the parameter synchronisation management needs of the monitoring terminal based on the LAN [21].

2.7.4 GSM Protocolos

Global System for Mobile Communications, or GSM. It is a widely used digital cellular communication standard that enables wireless networks to link mobile devices to the internet and allow them to interact with one another. This author stated that the GSM protocol is use in the SMS-based patient health monitoring system. The health parameters of the patient are sent using GSM via SMS. This system is useful in areas where medical facilities are not easily accessible. The health parameters are sent to the authorized personnel using GSM via SMS, and the personnel can then diagnose the patient's condition and take appropriate measures [26].

2.8 Comparison of Previous Project

Table 2.1: Comparison of among technologies used in related previous project

References	Component/ technology used	Advantages	Disadvantages
Patient Monitoring System using Thinkspeak Platform[27]	Component: -Pulse sensor -temperature sensor -Arduino -ESP8266 Wi-Fi modem Technology: -Thinkspeak platform	-Low cost. -Low power consumption. -Open source community and resource.	-Less focus on real-time control. -Limited mobile app development.
IoT Based Real-Time Remote Patient Monitoring System[28]	Component: -ECG sensor -Arduino Technology: -MQTT	-Low power consumption. -Allow for easy integration with other system and devices. -Low cost. -ECG sensor can collect data from the cardiac patient and amplify the ECG signal baased on the	-ECG sensor is not suitable for all patient. -MQTT no suitable for all IoT application.

		electrical activity of heart muscle.	
Designing ECG Monitoring Healthcare System Based Internet of Things Blynk Application [29]	Component: -ECG sensor -Arduino Uno -ESP8266 Technology: -Blynk	-Blynk is an pen-source platform that can handle thousands of devices. -Blynk provides a user-friendly interface. -Low cost microcontroller board -Arduino has a wide range of input and output pins.	-ECG sensor is not suitable for all patient
IoT Based Health Monitoring System [30]	Component: -temperature sensor -pulse sensor -room temperature and humidity sensor -GSM and Wi-Fi module -Arduino	-Allow for the remote monitoring of a patient’s health parameters using SMS. -Display the patient’s health in real-time on mobile device. -Easy to use. -Low cost.	-It’s challenging to evaluate the algorithm's effectiveness, accuracy, and applicability to different health conditions.

	Technology: -Matlab simulation -Blynk platform	-MATLAB simulation enables the testing and validation of the system's diagnosis algorithm.	-GSM technology has limited bandwidth, which means that it can only transmit a limited amount of data at a time. This can be a problem when transmitting large amounts of health data, as it may take a long time to transmit all the data.
A Zigbee Based Patient Health Monitoring System [31]	Component: -heart rate -temperature sensor -fall monitoring system -IR LED -IR sensor Technology: -Zigbee	-Allow for early detection of any abnormalities or emergencies -Low power consumption of zigbee technology -Automatic logging of vital parameter of patients for easy access.	-Zigbee has a limited data transfer rate and limited range.

IoT-Based Smart Health Monitoring System for Covid-19 Patients [4]	Component: -Arduino Uno -Node MCU ESP8266 -Temperature sensor -Pulse rate -Oxygen saturation -LCD Technology: -MIT App Inventor -Firebase	-Provides real time monitoring. -Low cost and widely available microcontroller board. -Potential to be valuable during Covid 19 pandemic	- Data Security of firebase, although Firebase provides security precautions, it is ultimately the app developer's duty to ensure the security of data. Data may be at danger due to improper setup or weak security procedures.
Smart Health Care [32]	Component: -Temperature sensor -Heartbeat sensor -Blood pressure sensor -Glucose sensor -GSM modem Technology: -Bluetooth -Wi-Fi -RFID	-Automatic and contactless identification and tracking of things or people are made possible by the wireless operation of RFID.	-Bluetooth has a limited range, typically up to 10 meters or 33 feets. -Low power consumption.

2.9 Summary

The literature review presents a comprehensive of the integration of technology into health monitoring systems, with a specific emphasis on the utilization of wearable sensors. The transformation of the healthcare sector is evident through the advent of innovative tools, enabling the tracking of vital signs like heart rate and temperature. The review examines the significance of health monitoring sensing technologies, particularly in the context of the ongoing pandemic. These technologies offer valuable insights into changes in temperature and heart rate, serbing as crucial indicators of issues such as the Coronavirus. Additionally, health monitoring systems are known to play role in tracking patient health, preventing disease and detecting potential risks early on. The literature highlights various microcontrollers, including Raspberry Pi, Arduino Uno, and NodeMCU, with distinct applications in health monitoring. Besides that, Wi-Fi, LoRa and Zigbee technologies are extensively discussed for their roles in wireless data transmission, long-distance communication, and wireless body area networks, respectively. The review also underscires the importance of platforms like ThingSpeak and Ubidots for data storage, analysis and visualization in health monitoring. And communication potocols such as MQTT, CoAP, LAN, and GSM are examined for their roles in data transmission and parameter synchronization in health monitoring systems. Overall, the literature review provides a comprehensive understanding of the multifaceted technologies and mothodolgies applied in health monitoring, addressing both current issues and broader challenges in the healthcare landscape.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter covers the project's overview, setups, and design, outlines the methods used in this project. Additionally, a list and explanation of the hardware and software utilised in this project will be provided. The proposed system's flowchart, block diagram, and communication model.

3.2 Selecting and Evaluating Tools for a Sustainable Development

In the process of selecting and evaluating tools for sustainable development, careful consideration is given to both software and hardware specifications. To ensure a clear understanding of project objectives, tools are employed to define the requirements, features, and specifications of the products. Regarding software, ThingSpeak serves as a crucial tool for seamless data collection and visualization, offering a user-friendly platform for monitoring project metrics. Additionally, MIT App Inventor contributes to creating an intuitive interface for displaying data from ThingSpeak on mobile devices so that, easy for doctors or who cares for patient to track the patient. These software tools collectively enhance efficiency and user interaction within the project, fostering a smooth and accessible experience for data monitoring and visualization. On the hardware front, specifications systematically outline essential components, including the main processing unit sensors for capturing temperature and pulse data, Wi-Fi connectivity modules, and interfaces for seamless communication.

3.3 Project Planning and Flow Chart

The data visualisation and storage layer makes up this Internet of Things based health monitoring system. The SEN-11574 pulse sensor, NodeMCU32, and DS18B20 temperature sensor make up the gateway layer. The NodeMCU32 are immediately connected to the temperature sensor and pulse sensor, which measures the user's body temperature and transmits the heart rate and temperature data to it through serial connection. The sensors will gather the user's heart rate and body temperature and transmit those readings to the cloud Wi-Fi gateway. The health data will then be sent through data base communication protocol via the cloud Wi-Fi gateway.

The user's health information will be shown in Thingspeak and updated in real-time. The data will only be accessible to authorised users. The health information will also be kept on file database for tracking reasons. Figure 3.1 below displays the suggested system's flowchart. The Wi-Fi gateway is where the system begins by gathering health information. The microcontroller NodeMCU-32 will capture data from sensors and send to the Internet on a specific Thingspeak Status Channel and also receive data from the Internet, by reading the data from specific Thingspeak Channel. The data will then be processed and received in accordance with the system's programme. For real-time data visualisation and storage, the health data will be supplied to Thingspeak and Android application which is Mit Inventor Apps. The application also read the data from Thingspeak Status Channel and display the data for the user. The health information gathered will be examined. The data will display the alert notifications to the designated users if there is a sudden change in pulse rate, or body temperature not more or less than 36.1 to 37.2 celcius as well as pulse rate not more than or less than 60 to 100 beat per minute (BPM). The procedure will then begin over from scratch.

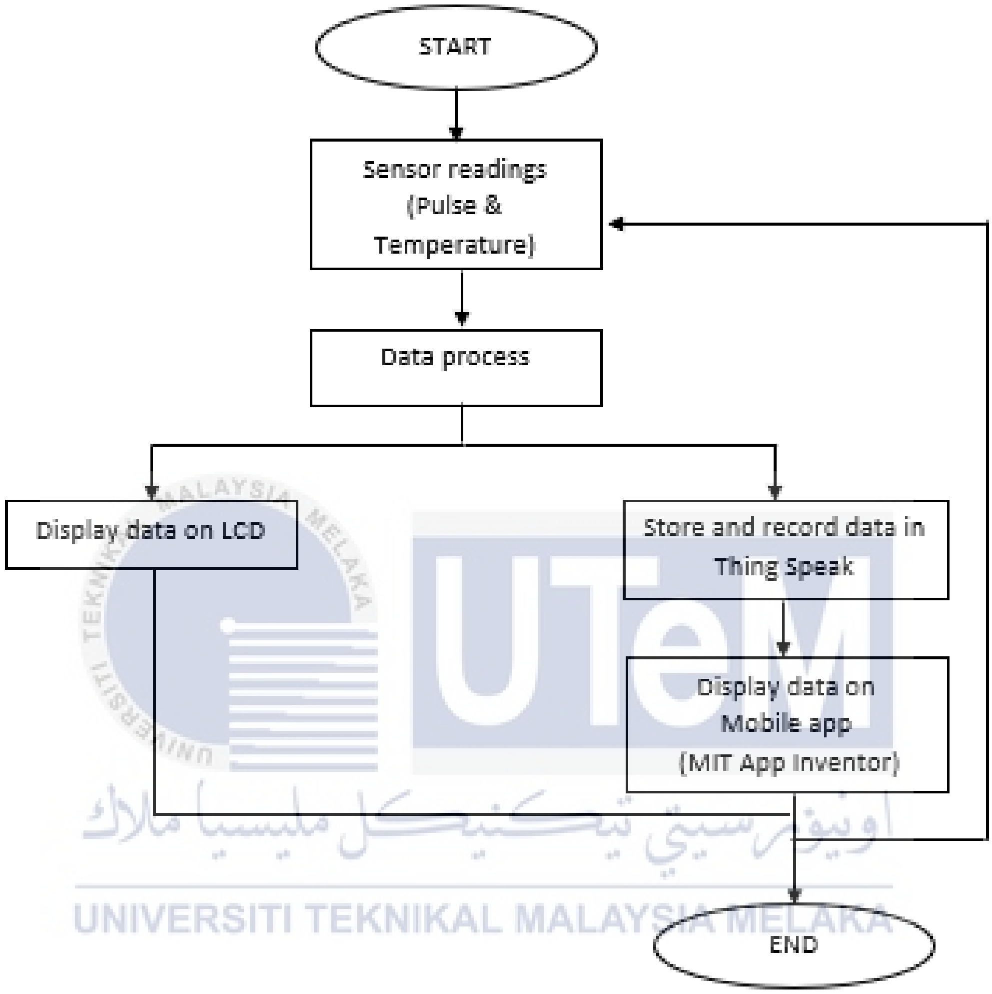


Figure 3.1: Flowchart of the proposed system

3.3.1 Block Diagram of the Proposed System

The block diagram for the suggested system health monitoring system is shown in Figure 3.2. The suggested solution is based on the integration of medical sensors and the controller in charge of gathering patient physical characteristics. User input was taken in the form of reading patient data from a serial monitor. It is possible to save sensor readings together with the date and time they were obtained using ThingSpeak platforms and data will recorded in excel format. The patient's condition will displayed abnormal or nomal on LCD and on the mobile application will be displayed data collection from ThingSpeak. ThingSpeak can be used by doctors or patient caregivers to track patient data.

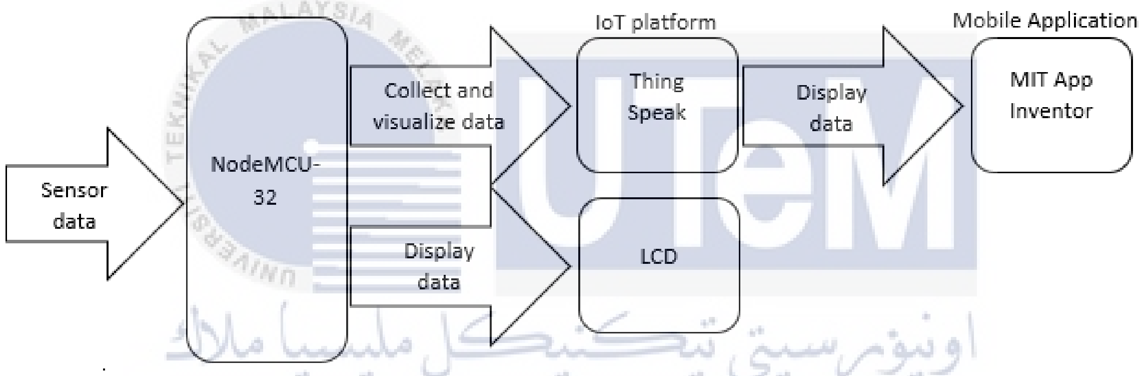


Figure 3.2: Block diagram of the Health Monitoring System

3.3.2 Scematic diagram of the Proposed System

Figure 3.3 shows the schematic diagram of proposed health monitoring system. The figure is shown the connection of each component using microcontroller NodeMCU-32.

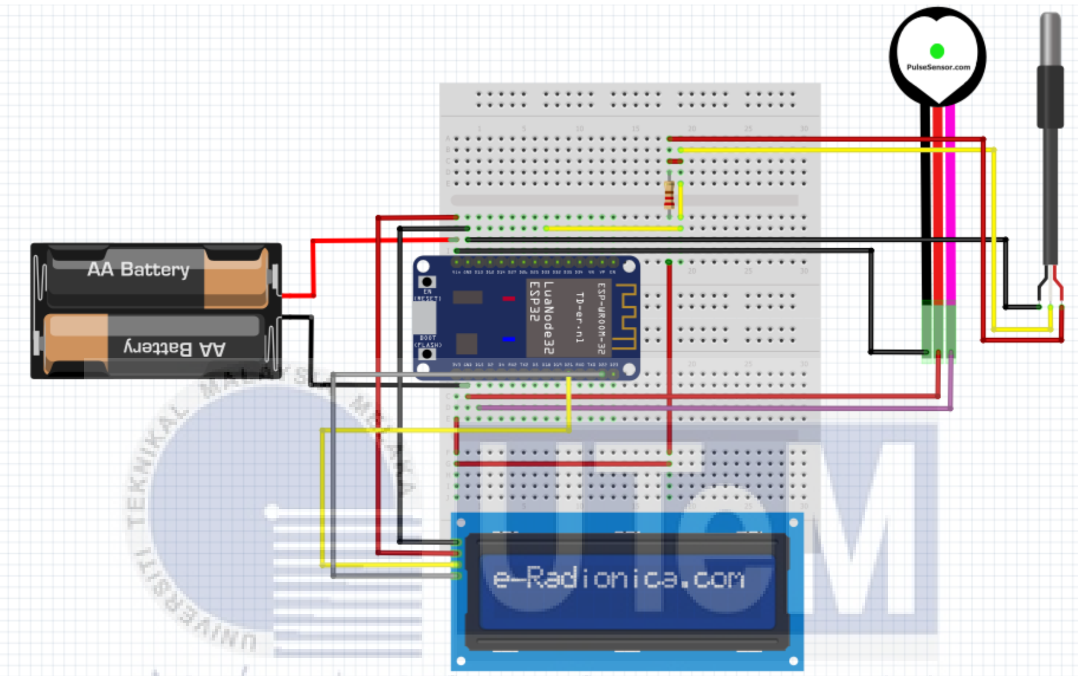


Figure 3.3: Schematic diagram of the Health Monitoring System

3.4 Tools

For the development and documentation of software and hardware systems, software and hardware specification tools are crucial. In order to ensure a clear understanding of the project goals, these tools assist in defining the requirements, features, and specifications of a product.

3.4.1 Software Specification

The software specifications for this project involve the use of several tools to seamlessly collect, visualize, and display data. ThingSpeak is employed for data collection and visualization, providing a user-friendly platform for monitoring project metrics. Fritzing is utilized to skecth the project design, offering a visual representation of the hardware components and their connections. Furthermore, MIT App Inventor is incorporated to create an intuitive interface for display data from ThingSpeak on mobile devices. These software components collectively contribute to the efficient functioning and user interaction within the project, ensuring a smooth and accessible experience for data monitoing and vizualization.

3.4.1.1 Thingspeak

ThingSpeak is an IoT analytics platform service that enables the collection, visualisation, and cloud-based analysis of real-time data streams. This project leverages ThingSpeak as an IoT analytics platform to analyse and visualize patient data. This platform enables the collection, real-time visualization, and cloud-based analysis of data stream. Through ThingSpeak, the project allows for the creation instant visualization of live data and uploading of patient data from devices. The data is securely stored in private channels by default, but the option to share it in public channels is available. Once the patient data is on ThingSpeak, the project analysis and visualization data interaction with other devices.

Figure 3.4: Logo of Thing Speak

3.4.1.2 Fritzing

The current project employs Fritzing as an integral tool for sketching IoT circuits. Fritzing is an open-source hardware project that enables everyone to use electronics as a tools for a design which helps artists, designers and DIY-tinkerers to prepare their hardware ideas for manufacturing. Fritzing has demonstrated its capacity to offer helpful support in the stages that follow the creation of an interactive product through the use of an interface concept based on the normal process of the target group. Fritzing is a useful method for recording these interactive documents as well. A common and understandable form of documentation is required, and Fritzing can offer it, since knowledge sharing has been the inspiration behind this new DIY movement. Fritzing has been shown to be an effective training tool for those without any previous technical experience when it comes to electronics.

Within this project, Fritzing provides three distinct perspectives on the circuit: a traditional schematic diagram view, a PCB design view, and a breadboard view that is similar to the real world. The standard 'breadboard view' shows the typical electrical components in an abstracted manner, mainly from a top view. Fritzing's schematic view is useful for handling more complex circuits as well as for instructional purposes. It may also act as a point of entry for consumers with a background in more conventional devices. The designer may transform the sketch into an expert circuit board by using the PCB view.

Figure 3.5: Logo of Fritzing

3.4.1.3 Arduino IDE

The current project is executed using the Arduino Software (IDE), aslo referred to as the Arduino Integrated Development Environment (IDE). This development environment offers a comprehensive set of tools, including a text editor for code writing, a message box, a text terminal, a toolbar with essential buttons, and various menus. This project leverage the IDE to input Arduino code through a text editor, allowing the execution of programs written in the C/C++ like Arduino programming language.

The code editor within the IDE is equipped with features such as syntax highlighting, auto-indentation, and code recommendations, faciliting the creation of error-free code. The enhance project functionality, the IDE incorporates libraries; pre-written pieces of codes that can easily search for, install, and manage through the Library Manager.

Figure 3.6: Logo of proteus

3.4.1.4 MIT App Inventor

The current project utilizes MIT App Inventor as a powerful tool for displaying patient data retrieved from ThingSpeak. MIT App Inventor, known for its user-friendly and visual programming approach, empowers individuals, including beginners and children, to create fully functional apps for smartphones and tablets. The visual, blocks-based programming tool accelerates app development, enabling even beginners to create a basic app within 30 minutes.

In alignment with the project's goals, MIT App Inventor contributes to the democratization of software development, encouraging individuals to transition from technology consumers to creators. In the context of the project, MIT App Inventor serves as a platform for creating applications that provide an easy method for users to browse and interact with patient data from ThingSpeak. This approach enhances the display of information, offering a more effective means for users to understand the healthcare-related data compared to traditional websites.

Figure 3.7: Logo of MIT App Inventor

3.4.2 Hardware Specification

The hardware specification for this project outline the essential components that form the core of the system. These include the main processing unit, sensors capturing data like temperature and pulse, connectivity modules such as Wi-Fi, the power supply mechanism, and interfaces for seamless communication. Additionally, other components, like display units, contribute to the overall functionally. These specifications provide a clear understanding of the technological aspects guiding the project, ensuring a solid foundation for its successful implementation.

3.4.2.1 DB18S20 Temperature Sensor

The DS18B20 digital thermometer features an alert function with nonvolatile user-programmable upper and lower trigger points, and it measures temperatures in Celsius between 9 and 12 bits. The DS18B20 connects to a central CPU using a 1-Wire bus, which by definition only needs one data line (and ground). Furthermore, the DS18B20 does not require an external power supply as it can obtain power straight from the data line (it may take known as "parasite power").

The described device features a unique 1-Wire interface, a communication protocol that requires only a single port pin, thereby reducing the overall component count. This integrated device combines a temperature sensor and EEPROM, capable of measuring temperatures in a wide range from -55°C to +125°C with a remarkable ±0.5°C accuracy between -10°C and +85°C. The programmable resolution, ranging from 9 to 12 bits, offers flexibility in temperature sensing. Notably, the device operates in a parasitic power mode, requiring only two pins (DQ and GND) for its operation. This simplifies distributed temperature-sensing applications with multidrop capability, as each device is uniquely identified by a 64-bit serial code stored in onboard ROM. Furthermore, the device supports user-definable nonvolatile alarm settings, enhancing its adaptability by allowing users to set temperature limits. This versatile component is available in 8-Pin SO (150 mils), 8-Pin µSOP, and 3-Pin TO-92 packages, providing options to suit different application requirements.

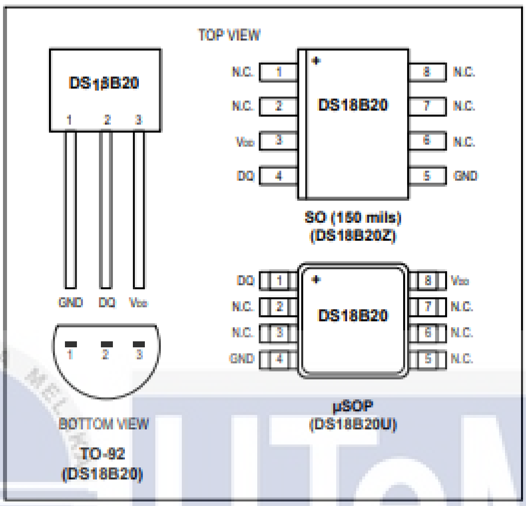


Figure 3.8: Pin configuration

3.4.2.2 Pulse Sensor (SEN-11574)

The pulse sensor is used to determine the user's heart rate in beats per minute (BPM). The pinout is represented by the positive terminal, negative terminal, and digital output pin in Figure 3.9 and Table 3.1. A photosensor and a green-emitting LED make up the pulse sensor. For the purpose of detecting changes in blood flow brought on by the heartbeat, the photosensor and green light are used. While the LED on the pulse sensor emits green light, the photosensor detects the green light that is reflected. The amount of light reflected depends on the size of the blood vessel in the artery.

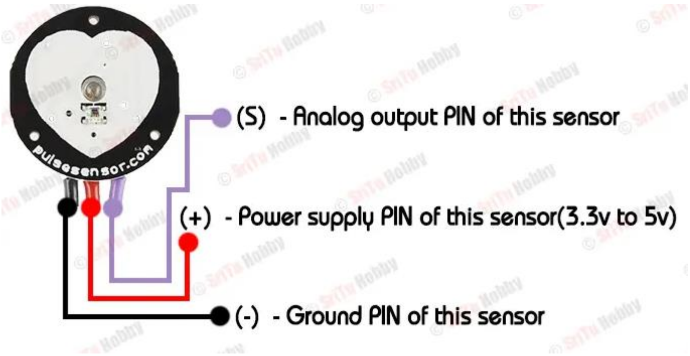


Figure 3.9: SEN-11574 Pulse Sensor

Table 3.1: Pulse Sensor Pinout Descriptions

Pin Colour	Descriptions
Black	Power supply
Red	Connected to ground
Purple	Output

3.4.2.3 LCD I2C

An easy-to-use display module that can simplify display is I2C_LCD. Making it simpler to use enables producers to focus on the current work. A typical I2C LCD display consists of an I2C LCD adapter and a character LCD display constructed on the HD44780 platform. These LCDs function best when displaying simply characters, as their name suggests. On a 16 by 2 character LCD, for instance, 32 ASCII characters can be shown over two rows. In comparison to an I2C LCD, a normal LCD requires significantly more effort to connect. Four pins are all that must be connected. Start by connecting the VCC pin to the Arduino's 5V output and the GND pin to the ground, as illustrated in Figure 3.10. I2C pins vary amongst Arduino boards and must be appropriately connected. On Arduino boards with the R3 layout, the SDA (data line) and SCL (clock line) are situated next to the AREF pin on the pin headers. They also go by the names A5 (SCL) and A4 (SDA).

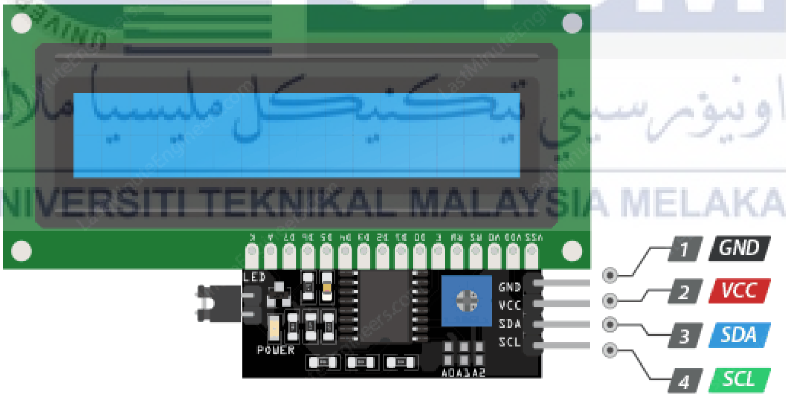


Figure 3.10: I2C_LCD pinout

3.4.2.4 NodeMCU32

The ESP32 WiFi SOC from Espressif are supported by the open-source, Lua-based NodeMCU software, which employs an on-module flash-based SPIFFS file system. The C-implemented NodeMCU is built around the Espressif ESP-IDF. ESP32 A feature-rich MCU for a variety of applications with built-in Bluetooth and Wi-Fi connection functionality through its SPI / SDIO or I2C / UART interfaces. The pinouts and specifications of the NodeMCU 32 are shown in Figure 3.11 and Table 3.2 below.

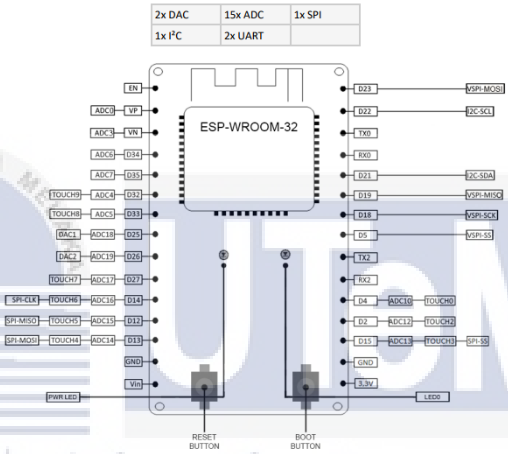


Figure 3.11: NodeMCU-32 pinout

Table 3.2: Specifications of NodeMCU 32

Specifications	
Operating voltage	3.3V
Operating current	10µA – 170mA
Input voltage	7V – 12V
Processor	Tensilica L106 32-bit
GPIO	80 – 160MHz
RAM	17
ADC Pin	One input with 1024 resolution

3.4.2.5 Battery

One type of lithium ion battery which is a chemical battery is the lithium polymer battery, frequently referred as the polymer lithium battery. Its features include high energy, miniaturisation, and small weight as compared to other batteries. Due to its ultra-thin nature, batteries with a variety of capacities and forms may be created to suit the requirements of certain devices. The smallest thickness that is possible in theory is 0.4 mm. Typically, a lithium polymer battery is made up of many battery packs connected in series to increase the available voltage or multiple identical secondary cells arranged in parallel to enhance the discharge current.

The most widely used 3.7V lithium polymer battery is the Lithium ion LP18650A+, which has a capacity of 5000mAh. And this project are using a nodemcu shield casing of nodeMCU-32 that support 6.5 to 16Volt. So, its must use two pieces of 3.7V lithium polymer battery to support the input voltage. Table 3.3 below shown a specifications of Lithium ion 3.7V 5000mAh 18650 Li-ion Battery.

Figure 3.12: Lithium ion 3.7V 5000mAh 18650 Li-ion Battery

Table 3.3: Specifications of Lithium ion 3.7V 5000mAh 18650 Li-ion Battery

Specifications	
Rated Capacity	5000mAh
Nominal Voltage	3.7V
Max Charge Voltage	4.2V
Discharge Cut Off Voltage	2.75V
Charging Current	2A
Max. Continuous Discharging Current	2A
Cycle Life	500times

3.4.2.6 Resistor

The 4.7kΩ resistor is characterized by a resistance value of 4.7 kilo Ohms, with a standard tolerance of 5%. It possesses a power rating of 1/4W, indicating its capability to dissipate heat without damage. Typically featuring a temperature coefficient of around 100 parts per million per degree Celsius (ppm/°C), this resistor exhibits a moderate sensitivity to temperature variations. It is crucial to ensure that the voltage across the resistor remains within its operational limits, as specified by its voltage rating. These specifications collectively define the key parameters necessary for the proper selection and usage of the 4.7kΩ resistor in electronic circuits.

The DS18B20 is a digital temperature sensor that uses the Dallas Semiconductor 1-Wire communication protocol. The 4.7kΩ (ohm) resistor is not directly connected to the DS18B20 sensor but is typically used in a pull-up configuration on the data line (DQ) of the 1-Wire bus. The 4.7kΩ pull-up resistor is an integral part of the 1-Wire bus configuration for DS18B20 temperature sensors. It ensures reliable communication, stable bus conditions, and supports the parasitic power mode of the DS18B20.

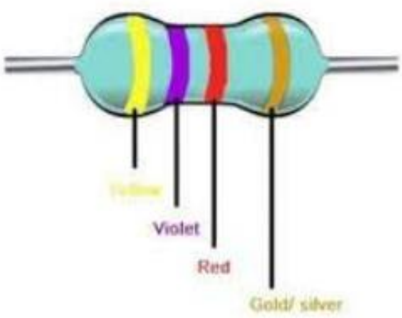


Figure 3.13: Color code of 4.7k Ohm resistor

Table 3.4: Specifications of 4.7k Ohm resistor

Specifications	
Resistance	4.7kΩ
Tolerance	5%
Color Code	Yellow / Violet / Red / Gold
Type	Carbon Film
Voltage (max operating)	350V
Operating temp	-55C to +155C
Package	Conformal Coated, Axial

3.5 Limitation of proposed methodology

The proposed methodology's reliance on a single data gathering technique poses a problem because it may limit the scope of understanding and fail to capture aspects of the research topic.

3.6 Summary

In order to maintain track of vital signs and health-related information and construct a user-friendly health monitoring system, the method that has been developed is presented in this chapter. The suggested methodology's main goal is to visualize and collect data and develop a system that employs database management to store patient data in a way that does not materially reduce the accuracy of the findings.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents the results and analysis on the development of health monitoring system. The focus of the analysis section is on uncover pattern, trends, and implications within the context of the health mnitoring technology. Specifically, it aims to interpret and comprehend the results obtained from the prototype system designed for health monitoring. The analysis involves connecting temperature sensor and pulse sensor using NodeMCU-32. The software implementation strategically places sensors to collect patient data, which is data will uploaded to ThingSpeak and transmitted to MIT App Inventor for mobile display. The information collected by the health monitoring system is displayed in the results section. According on the particular parameters being measured, it contains both initial data and generated metrics. The findings can be shown as tables, graphs, charts, or visualisations for simple comparison and understanding.

4.2 Results and Analysis

Result and analysis are to make understanding of findings data, interpret their and analyze the implications they carry within the context of the study. This segment provides a detailed account of the results engages in analysis, shedding light on patterns, trends, and any noteworthy observations. Through this examination, a deeper understanding of the research outcomes is encouraged, facilitating informed discussions and conclusions.

4.2.1 Prototype System Design

It is essential to test the hardware and software once development is finished to see if the intended outcomes were achieved. As shown in Figure 4.1, this is accomplished by connecting a DS18B20 temperature sensor and a pulse sensor using NodeMCU-32. In this particular case, the circuit and system to support the Nodemcu shield shell are started by a 7.4V battery, however the NodeMCU microcontroller may operate at 5V or more. Serial communication, specifically universal asynchronous receiver/transmitter (UART) communication, permits connection between electronic devices.

Figure 4.1: Complete hardware circuit communication

4.2.2 Software Implementation

Health monitoring system collect data from the patient by positioned the sensors in a way that aligns with the patient's position. The pulse sensor is placed on the fingertip while the DS18B20 temperature sensor is grasp or placed on the patient's armpit. The sensors are placed in these locations due to their compatibility with the patient's environment and the specifications required for data collection. In Figure 4.2, the collected data from the Nodemcu is uploaded to ThingSpeak, with the temperature value on field 2 and the patient's heartbeat value on field 1. Then data will transmit to the application mobile MIT App Inventor to display the graph data patient in a day, as shown in Figure 4.3. Some data is not accurate with the real data because of the various factor, such as accuracy of the sensor, ambient temperature and so on.

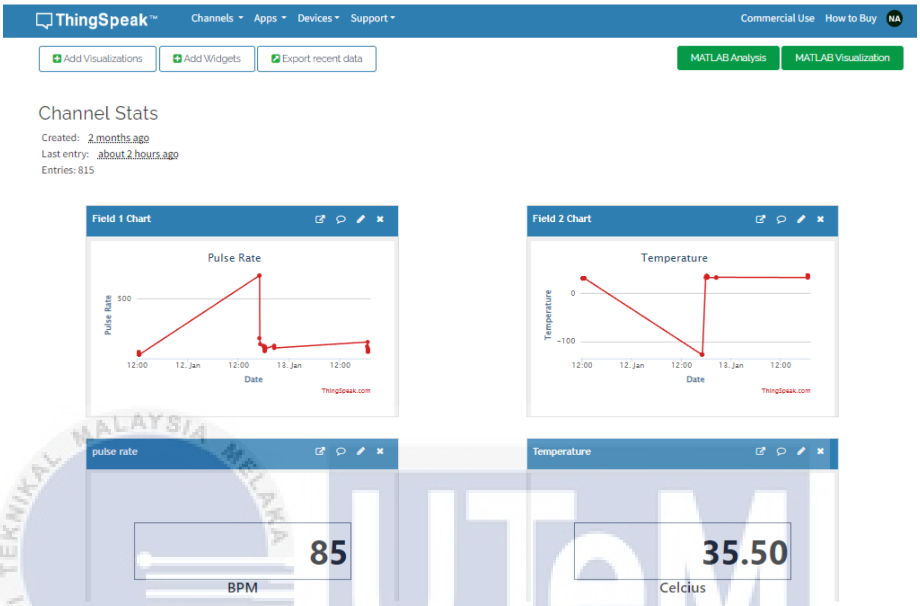


Figure 4.2: ThingSpeak platform created

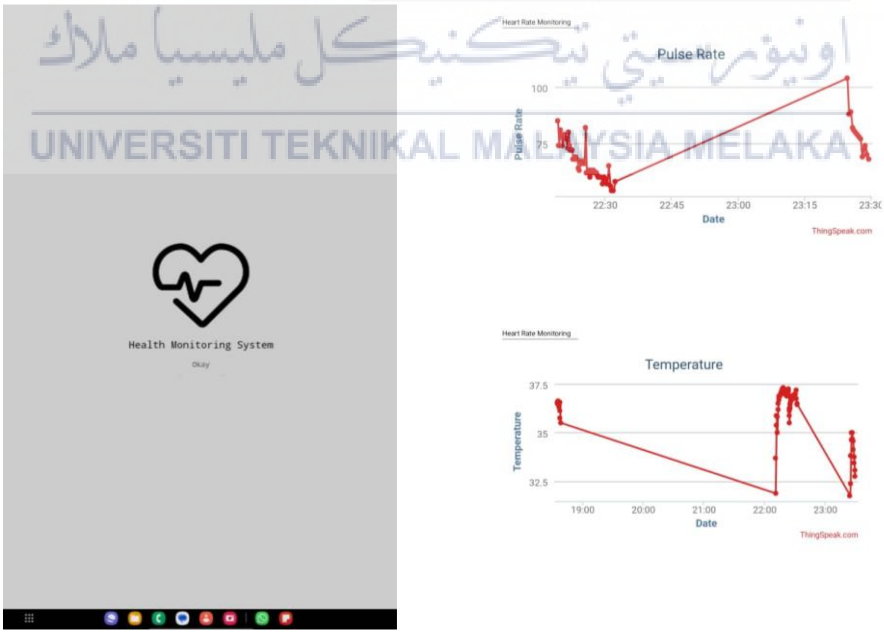


Figure 4.3: MIT Apps Inventor created

The health data received by the system is continuously updated in real-time on ThingSpeak, as illustrated in the figure above. The displayed information includes the individual's body temperature and pulse rate (BPM).

4.2.2.1 Data Storing

The data was collected from various patients, capturing their temperature and pulse rate under normal conditions and subsequently after engaging in activities such as sports, brisk walking, and other physical exertions. This comprehensive dataset enables a thorough analysis of how these activities impact physiological parameters, providing valuable insights into the dynamic changes in temperature and pulse rate across different individuals. Such data holds significance in understanding the immediate physiological responses to physical activities, contributing to the broader field of health monitoring and personalized healthcare.

Date / Time	entry_id	Pulse Rate(BPM)	Temperature (°C)
2023-12-25 15:01:08 UTC	660	96	35.375
2023-12-25 15:01:26 UTC	661	95	35.5
2023-12-25 15:01:43 UTC	662	92	35.5
2023-12-25 15:02:00 UTC	663	95	35.375
2023-12-25 15:02:18 UTC	664	94	35.5
2023-12-25 15:02:35 UTC	665	93	35.5
2023-12-25 15:02:52 UTC	666	92	35.5
2023-12-25 15:03:09 UTC	667	92	35.4375
2023-12-25 15:03:27 UTC	668	92	35.125
2023-12-25 15:03:44 UTC	669	85	34.5
2023-12-25 15:04:01 UTC	670	91	34
2023-12-25 15:04:18 UTC	671	90	33.5625
2023-12-25 15:05:41 UTC	673	93	32.3125
2023-12-25 15:05:56 UTC	674	91	32.1875
2023-12-25 15:06:12 UTC	675	90	32
2023-12-25 15:06:27 UTC	676	87	33.375
2023-12-25 15:06:42 UTC	677	89	34.375
2023-12-25 15:06:58 UTC	678	86	34.8125
2023-12-25 15:07:13 UTC	679	86	35.1875
2023-12-25 15:07:29 UTC	680	83	35.375
2023-12-25 15:07:44 UTC	681	84	35.5625
2023-12-25 15:08:01 UTC	682	84	35.4375
2023-12-25 15:08:17 UTC	683	94	35.375

Figure 4.4: Excel showing data collected from ThingSpeak

4.2.3 Comparing data with existing health monitoring system

Comparing the data with existing health monitoring system provides valuable insights. The comparative analysis aims to assess the performance and characteristics of implemented health monitoring components, particularly focusing on the SEN-11574 heart rate sensor and the DS18B20 temperature sensor. By examining results in relation to established systems, the goal is to draw insights into the efficacy of the approach, shedding light on potential strengths and areas for improvement within the realm of health monitoring technology.

4.2.3.1 Heart beat measurement by SEN-11574 pulse sensor and Smart Watch

The comparison to measured heart rate is conducted using a smartwatch and a pulse sensor. To measure the heart rate, the smartwatch is worn on the wrist, and the pulse sensor is placed on the fingertip or another suitable location. This experiment were recorded in three times to see the accuracy of the measurement. The data collected from twenty readings were then analyzed and presented in Table 4.1 and Table 4.2 to illustrate the study findings.

Table 4.1: Heart beat measurement by SEN-11574 pulse sensor

entry_id	Heart rate(BPM) by pulse sensor			Average
1	86	85	68	80
2	83	88	69	80
3	84	87	68	80
4	84	85	64	78
5	94	84	63	80
6	80	82	67	76

7	80	85	66	77
8	80	74	67	74
9	81	81	66	76
10	79	78	82	80
11	89	74	62	75
12	96	79	62	79
13	95	78	63	79
14	92	77	60	76
15	95	73	63	77
16	94	80	62	79
17	93	74	63	77
18	92	72	80	81
19	92	72	74	79
20	92	68	72	77

The measuring processes were repeated on the same people by using Smart Watch in 3 times. The obtained results are shown in Table 4.2 below.

Table 4.2: Heart beat measurement by Smart Watch

entry_id	Heart rate(BPM) by Smart Watch			Average
1	80	81	81	81
2	83	78	77	79
3	84	74	74	77
4	81	79	75	78
5	79	78	78	78
6	79	77	77	78

7	80	73	71	75
8	79	80	78	79
9	78	74	71	74
10	76	72	70	73
11	76	72	70	73
12	91	79	75	82
13	90	74	70	78
14	95	85	83	88
15	94	88	85	89
16	93	87	85	88
17	92	85	82	86
18	92	84	80	85
19	92	82	80	85
20	85	85	85	85

The data obtained indicates that the heart rate sensors measured values ranging from 60 to 96 beats per minute (BPM). The recorded heart rates fall within the normal range, which is defined as 60 to 100 bpm. The SEN-11574 pulse sensor utilized demonstrates reliable perfomance, exhibiting an average accuracy with a minimal error rate of 2.8%.

Table 4.3: Average of error rate for heart beat measurement

Smart Watch	Pulse rate sensor	Error (%)
81	80	1.23
79	80	-1.27
77	80	-3.90

78	78	0.00
78	80	-2.56
78	76	2.56
75	77	-2.67
79	74	6.33
74	76	-2.70
73	80	-9.59
73	75	-2.74
82	79	3.66
78	79	-1.28
88	76	13.64
89	77	13.48
88	79	10.23
86	77	10.47
85	81	4.71
85	79	7.06
85	77	9.41
Average		2.80

The findings presented in Figure 4.5 indicate the effectiveness of the SEN-11574 heart rate sensor in detecting increases in heart rate. Minor fluctuations in the obtained BPM readings can be attributed to the pressure exerted by the fingertip on the pulse sensor. The pulse sensor utilizes LED light and a photosensor to discern changes in blood flow within the fingertip for BPM measurement. However, it's important to note that the applied fingertip pressure can impact blood flow, leading to variations in the BPM measurements.

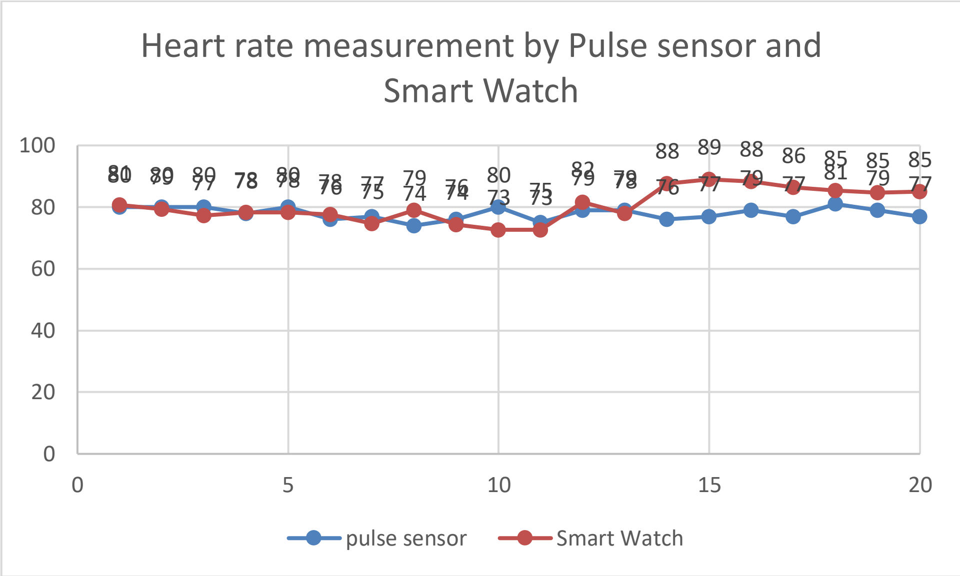


Figure 4.5: Comparison of heart rate mesurement

4.2.3.2 Body temperature measurement by DS12B20 and thermometer

The comparison to measured temperature body is by using the DS18B20 temperature sensor and a thermometer. To measure temperature, insert the thermometer on the forehead patient and the DS18B20 temperature sensor use under the armpit or grasp. A group of female participants aged up to 23 years, engaged in various activities, was selected for this experiment. The subject's body temperature were recorded, and the measurement was repeated in three time. The data collected from twenty readings were then analyzed and presented in Table 4.4 and Table 4.5 to illustrate the experiment findings.

Table 4.4: Body temperatured measured by DS18B20

entry_id	Temperature Reading (°C) by temperature sensor			Average
1	35.19	37.19	37.25	36.54
2	35.38	37.25	37.25	36.63
3	35.56	37.13	37.31	36.67
4	35.44	36.94	37.13	36.50
5	35.38	36.25	37.13	36.25
6	35.38	35.50	37.19	36.02
7	35.38	35.88	37.19	36.15
8	35.50	36.13	37.25	36.29
9	35.31	36.25	37.06	36.21
10	35.00	36.44	37.00	36.15
11	35.00	36.50	37.13	36.21
12	35.38	36.63	37.13	36.38

13	35.50	36.69	37.00	36.40
14	35.50	36.75	36.94	36.40
15	35.38	36.88	37.06	36.44
16	35.50	36.81	37.13	36.48
17	35.50	36.81	37.13	36.48
18	35.50	37.13	36.88	36.50
19	35.44	37.13	36.94	36.50
20	35.13	36.88	37.00	36.33

The measuring processes were repeated on the same people by using a thermometer in 3 times. The obtained results are shown in Table 4.5, below.

Table 4.5: Body temperature measured by a thermometer

entry_id	Temperature Reading (°C) by thermometer			Average
1	35.80	37.19	37.00	36.66
2	36.20	37.10	37.00	36.77
3	36.20	37.13	37.00	36.78
4	36.90	36.90	37.10	36.97
5	36.70	36.50	37.10	36.77
6	36.70	36.50	37.00	36.73
7	36.70	36.80	37.00	36.83
8	36.50	36.50	37.10	36.70
9	36.80	36.50	37.00	36.77
10	36.80	36.40	37.00	36.73
11	37.00	36.50	37.10	36.87

12	36.90	36.60	37.10	36.87
13	37.00	36.70	37.00	36.90
14	37.10	36.80	36.90	36.93
15	36.38	36.80	37.10	36.76
16	36.50	36.80	37.10	36.80
17	36.80	36.80	37.00	36.87
18	36.50	37.10	37.00	36.87
19	36.80	37.10	36.90	36.93
20	36.30	37.00	37.10	36.80

The data obtained indicates that the temperature sensor measured values ranging from 35 to 37.25 ºC. The recorded temperature fall within the normal range, which is defined as 36.1 to 37.2ºC. The DS18B20 temperature sensor utilized demonstrates reliable perfomance, exhibiting an average accuracy with a minimal error rate of 1.21%.

Table 4.6: Average of error rate for temperature measurement

Thermometer	Temperature Sensor	Error (%)
36.66	36.54	0.33
36.77	36.63	0.38
36.78	36.67	0.30
36.97	36.5	1.29
36.77	36.25	1.43
36.73	36.02	1.97
36.83	36.15	1.88
36.7	36.29	1.13

36.77	36.21	1.55
36.73	36.15	1.60
36.87	36.21	1.82
36.87	36.38	1.35
36.9	36.4	1.37
36.93	36.4	1.46
36.76	36.44	0.88
36.8	36.48	0.88
36.87	36.48	1.07
36.87	36.5	1.01
36.93	36.5	1.18
36.8	36.33	1.29
Average		1.21

The stability of body temperature readings from a thermometer contrasts with the fluctuations observed in the graph depicting body temperature measured by the DS18B20, as illustrated in Figure 4.6. The DS18B20's inaccuracy introduces fluctuations in the recorded body temperature, as the readings are directly tied to the voltage output of the DS18B20. Fluctuations in the DS18B20's output voltage lead to corresponding variations in temperature readings, potentially triggering false alarms intermittently or maybe because of the various factor, such as accuracy of the sensor, ambient temperature. To enhance system reliability, it is suggested that future iterations incorporate a more accurate temperature sensor with Internet of Things (IoT) capabilities.

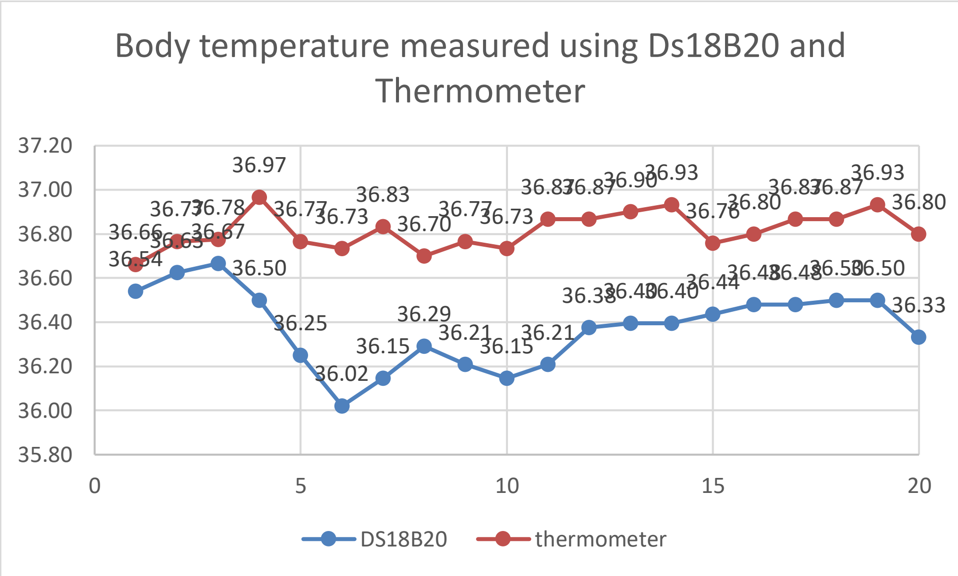


Figure 4.6: Comparison of body temperature

4.3 Discussion

This chapter present about a system that keeps track of people's health. It explains how the system analyzes the gathered data to find important information about a person's well-being. To make sense of the data, the system uses methods like statistics and visual representations, like graphs and charts. The results are displayed, showing both the original data and created new information by the system.

The discussion part of the chapter looks deeper into the results. It talks about testing the hardware and software of the system to make sure everything works as intended. The system uses sensors to collect information from patients, and the data is sent to a platform called Thing Speak and then to a mobile app for display. The chapter also discusses how the system compares to existing health monitoring methods, looking at how well it measures heart rate and body temperature.

In the end of the chapter discusses the findings, pointing out what worked well and what could be improved. It suggests using better sensors for more accurate readings and highlights the importance of the system for quickly assessing a person's health.

4.4 Summary

This chapter presented case studies to demonstrate applicability of the proposed system health monitoring method. The case study is based on two real utility measurement from the existing measurement of health monitoring. The two existing measurement selected for the case study serving in different people and some of them engaged in various activities. This experiment measurement is estimated using the proposed approach 3 times to get accuracy of the measurement and was selected in 20 people. Using standard spreadsheet software from the Excel that collected data at ThingSpeak, the documentation of software and hardware systems was (presented in Chapter 3) are programmed to calculate repeated 3 times for 20 people choosen. In addition, the health monitoring results obtained through the suggested approach demonstrate consistency with variations in individual health characteristics, yielding reliably accurate outcomes.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This report presents a method for develop an IoT based health monitoring system that able to provide real-time health data monitoring of the user. The proposed methodology is effective and robust in order to obtain good results utilizing only fairly accurate information and with a minimum network measurement information. The proposed analytical approach to track the health patient, send the health data to the visualization platform in real-time and send the data to application mobile. The MIT App Inventor and LCD display allow authorised users to examine the user's health data. When there are sudden changes in heart rate and abnormal body temperature, the warning will appear on LCD display. The user may monitor their health status by recording their health data, which are saved in both local and cloud databases. To demonstrate the sensors' and the system's functioning, experiments are carried out.

Overall, the research presented in this thesis has succeeded in making a contribution to understanding the importance of effective and practical system wide estimation methodology of health monitoring systems. The presented method utilizes reasonable type and amount of data input, using simple mathematical manipulations, yet, capable to produce quick, credible, representative and reasonably accurate results. Moreover, the undertaken research has entailed the formulation of methodologies facilitating the quantification and assessment of a distribution health monitoring system, alongside the potential cost savings associated with introducing new infrastructure or enhancing existing elements. This analysis serves as a basis for justifying and prioritizing investments aimed at reducing health-related issues in distribution networks. Consequently, it establishes a foundational framework upon which future research in the realm of health monitoring systems can be constructed.

5.2 Potential for Commercialization

The outlined health monitoring project possesses considerable potential for commercialization across various avenues. One avenue involves the sale of the comprehensive hardware and software components comprising the IoT-based health monitoring system, targeting healthcare institutions and individual consumers alike. Subscription-based services offer another commercialization opportunity, providing continuous access to real-time health data, analytics, and reporting features with recurring revenue models. The analytical tools and insights derived from the health data can be packaged and commercialized, catering to the needs of healthcare professionals and research institutions seeking in-depth analyses. Cloud storage and database services represent yet another avenue, offering secure data storage solutions for healthcare projects or institutions. Customization and integration services can be commercialized to tailor the health monitoring system for specific healthcare providers, fostering adaptability. Exploring partnerships and collaborations with healthcare organizations, insurance companies, and research institutions opens doors for joint ventures, co-development, or data-sharing agreements. Additionally, wearable devices equipped with the health monitoring system can be developed and commercialized for consumer markets interested in personal health tracking. Data licensing agreements and positioning the system as a solution for remote patient monitoring further expand the potential for commercial success. Finally, opportunities for international expansion can be explored by aligning the health monitoring system with global healthcare standards and regulations. Overall, these diverse commercialization strategies leverage the project's multifaceted capabilities, ensuring economic viability and scalability.

5.3 Future Works

For future improvements, accuracy of the health monitoring system estimation results could be enhanced as follows:

i) Integrate solar panels into the health monitoring system to harness solar energy storage units, providing a sustainable and renewable source of power to use in long term and safe to use when in rural area.

ii) Design the systems to withstand and adapt to real-world environmental conditions and reliability in diverse settings where users may require continuous health monitoring.

iii) Used the accuracy sensor, to get the exact value in real-time.

iv) Integrate machine learning models that can track from patient data over time, allow the system make predictions and forecasts based on patterns and correlations identified within the dataset.

v) Enhancing the mobile application development experience on MIT App Inventor could involve integrating advanced features for smoother collaboration, expanding compatibility with diverse platforms, and implementing additional tools that cater to evolving technological needs

REFERENCES

[1] K. T. Wai, N. P. Aung, and L. L. Htay, "Internet of Things (IoT) Based Healthcare Monitoring System using NodeMCU and Arduino UNO the Creative Commons Attribution License (CC BY 4.0)," *International Journal of Trend in Scientific Research and Development (IJTSRD) International Journal of Trend in Scientific Research and Development*, no. 5, pp. 755–759, 2019, doi: 10.31142/ijtsrd26482.

[2] P. Valsalan, T. A. B. Baomar, and A. H. O. Baabood, "IoT based health monitoring system," *Journal of Critical Reviews*, vol. 7, no. 4, pp. 739–743, 2020, doi: 10.31838/jcr.07.04.137.

[3] S. Patil and S. Pardeshi, "Health Monitoring system using IoT," *International Research Journal of Engineering and Technology*, 2018, [Online]. Available: www.irjet.net

[4] M. M. Khan, S. Mehnaz, A. Shaha, M. Nayem, and S. Bourouis, "IoT-Based Smart Health Monitoring System for COVID-19 Patients," *Comput Math Methods Med*, vol. 2021, 2021, doi: 10.1155/2021/8591036.

[5] P. Suryanarayana, R. Kalyankumar, S. Bhaskarakrishna, and R. Chinnu, "DESIGN AND IMPLEMENTATION OF HEALTH CAREMONITORING SYSTEM USING CLOUD AND MOBILEAPPLICATION."

[6] S. Nandyal, R. U. Kulkarni, and R. P. Metre, "Old Age People Health Monitoring System using IoT and ML," 2019. [Online]. Available: www.ijisrt.com160

[7] S. N. Korra, E. Sudarshan, and K. S. Naik, "The user has requested enhancement of the downloaded file," vol. 14, no. 4, 2020, [Online]. Available: www.arpnjournals.com

[8] E. N. GANESH, “Health Monitoring System using Raspberry Pi and IOT,” *Oriental journal of computer science and technology*, vol. 12, no. 1, pp. 08–13, Apr. 2019, doi: 10.13005/ojcst12.01.03.

[9] S. Patil and S. Pardeshi, “Health Monitoring system using IoT,” *International Research Journal of Engineering and Technology*, vol. 5, no. 04, 2018, [Online]. Available: www.irjet.net

[10] A. Ubale, N. Mumabai, I. O. Boritkar, I. V. Ladge, and I. R. More, “SMART HEALTHCARE MONITORING SYSTEM,” 2022.

[11] A. Abdulkadir Ibrahim and W. Zhuopeng, “IOT Patient Health Monitoring System,” *Journal of Engineering Research and Application www.ijera.com*, vol. 8, pp. 77–80, 2018, doi: 10.9790/9622-0801037780.

[12] N. Bin Kamarozaman and A. H. Awang, “IOT COVID-19 Portable Health Monitoring System Using Raspberry Pi, Node-Red and ThingSpeak,” in *IEEE Symposium on Wireless Technology and Applications, ISWTA*, IEEE Computer Society, 2021, pp. 107–112. doi: 10.1109/ISWTA52208.2021.9587444.

[13] K. Chooruang and P. Mangkalakeeree, “Wireless Heart Rate Monitoring System Using MQTT,” in *Procedia Computer Science*, Elsevier B.V., 2016, pp. 160–163. doi: 10.1016/j.procs.2016.05.045.

[14] V. Bhardwaj, R. Joshi, and A. M. Gaur, “IoT-Based Smart Health Monitoring System for COVID-19,” *SN Comput Sci*, vol. 3, no. 2, Mar. 2022, doi: 10.1007/s42979-022-01015-1.

[15] A. Mishra, A. Kumari, P. Sajit, and P. Pandey, “Remote web based ECG Monitoring using MQTT Protocol for IoT in Healthcare Internet of Things View project Smart city waste management system based on IoT View project REMOTE WEB BASED ECG MONITORING USING MQTT PROTOCOL FOR IOT IN HEALTHCARE,”

International Journal of Advance Engineering and Research Development, vol. 5, 2018, doi: 10.21090/IJAERD.71092.

[16] C. Senthamilarasi, J. J. Rani, B. Vidhya, and H. Aritha, "A SMART PATIENT HEALTH MONITORING SYSTEM USING IOT," 2018. [Online]. Available: http://www.acadpubl.eu/hub/

[17] Dr. A. Gangopadhyay, "IOT based Patient Health Monitoring using Sensor Technology," *Int J Res Appl Sci Eng Technol*, vol. 9, no. 5, pp. 1554–1562, May 2021, doi: 10.22214/ijraset.2021.34597.

[18] I. Bouazzi, M. Zaidi, M. Usman, M. Z. M. Shamim, V. K. Gunjan, and N. Singh, "Future Trends for Healthcare Monitoring System in Smart Cities Using LoRaWAN-Based WBAN," *Mobile Information Systems*, vol. 2022, 2022, doi: 10.1155/2022/1526021.

[19] C. Vitalis Amadi and S. Kim, "Remote Monitoring System for Smart Health Data Collection Analysis and Supervision Based on Lorawan Technology." [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128610001568

[20] B. E. Buthelezi, M. Mphahlele, D. Duplessis, S. Maswikaneng, and T. Mathonsi, "ZigBee Healthcare Monitoring System for Ambient Assisted Living Environments," 2019.

[21] X. Zhu and Y. Ye, "Monitoring of clinical signs of intravenous infusion patients with ZigBee wireless technology," *Int J Distrib Sens Netw*, vol. 18, no. 4, Apr. 2022, doi: 10.1177/15501329221091505.

[22] S. Krishna and V. Rongala, "PATIENT MONITORING SYSTEM USING THINGSPEAK PLATFORM", [Online]. Available: www.irjmets.com

[23] A. Abu Bakar, S. S. A. Rahim, A. R. Razali, E. Noorsal, R. Radzali, and A. F. Abd Rahim, "Wearable Heart Rate and Body Temperature Monitoring Device for

Healthcare," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1742-6596/1535/1/012002.

[24] H. H. Alshammari, "The internet of things healthcare monitoring system based on MQTT protocol," *Alexandria Engineering Journal*, vol. 69, pp. 275–287, Apr. 2023, doi: 10.1016/j.aej.2023.01.065.

[25] M. Zorkany, K. Fahmy, and A. Yahya, "Performance Evaluation of IoT Messaging Protocol Implementation for E-Health Systems," 2019. [Online]. Available: www.ijacsa.thesai.org

[26] P. Valsalan, T. A. B. Baomar, and A. H. O. Baabood, "IoT based health monitoring system," *Journal of Critical Reviews*, vol. 7, no. 4, pp. 739–743, 2020, doi: 10.31838/jcr.07.04.137.

[27] P. Kakria, N. K. Tripathi, and P. Kitipawang, "A real-time health monitoring system for remote cardiac patients using smartphone and wearable sensors," *Int J Telemed Appl*, vol. 2015, 2015, doi: 10.1155/2015/373474.

[28] M. M. Khan, S. Mehnaz, A. Shaha, M. Nayem, and S. Bourouis, "IoT-Based Smart Health Monitoring System for COVID-19 Patients," *Comput Math Methods Med*, vol. 2021, 2021, doi: 10.1155/2021/8591036.

[29] D. Hasan and A. Ismaeel, "Designing ECG Monitoring Healthcare System Based on Internet of Things Blynk Application," *Journal of Applied Science and Technology Trends*, vol. 1, no. 3, pp. 106–111, Jul. 2020, doi: 10.38094/jastt1336.

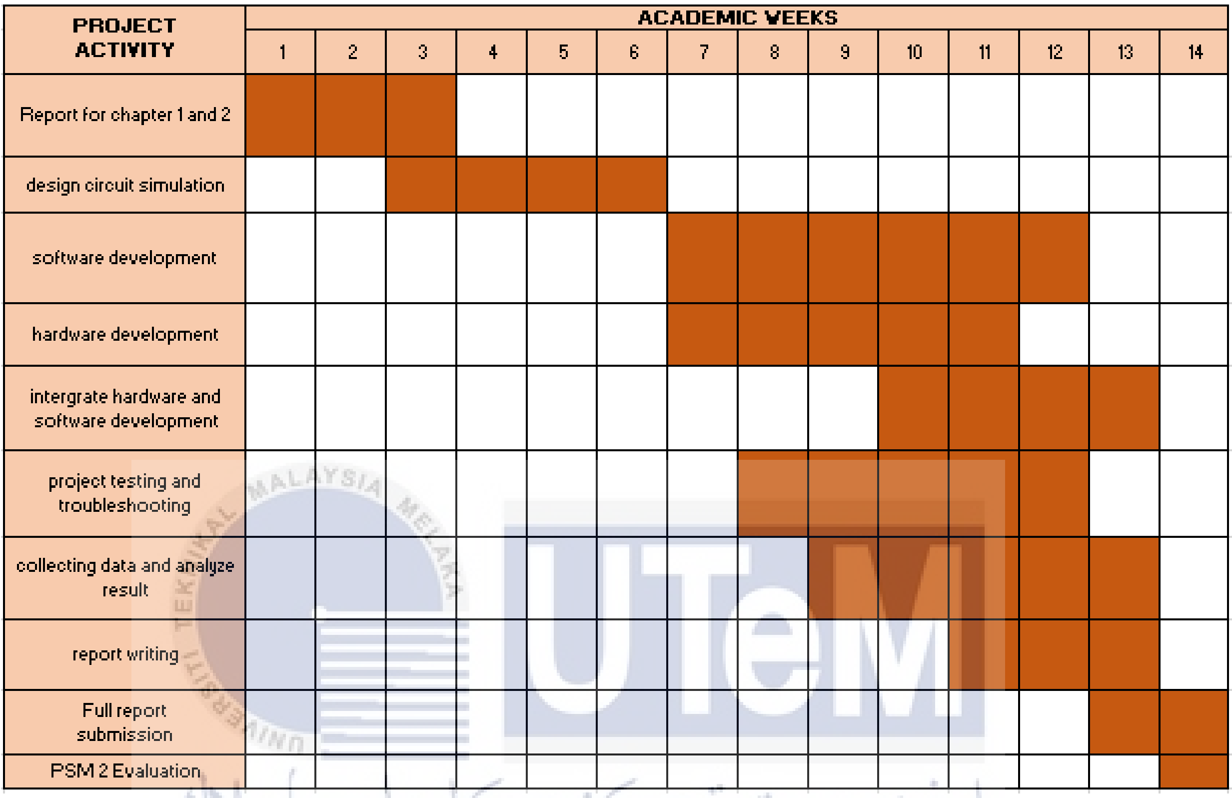
[30] P. Valsalan, T. A. B. Baomar, and A. H. O. Baabood, "IoT based health monitoring system," *Journal of Critical Reviews*, vol. 7, no. 4, pp. 739–743, 2020, doi: 10.31838/jcr.07.04.137.

[31] K. Navya and M. B. R. Murthy, "A Zigbee Based Patient Health Monitoring System," 2013. [Online]. Available: www.ijera.com

[32] Kavitha Rangasamy, "2018smarthealthcareIJIET," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 2S, 2018.

APPENDICES

Appendix A



Appendix A

```
#include <Wire.h>

#include <WiFi.h>

#include <ThingSpeak.h>

#include <HTTPClient.h>

#include <OneWire.h>

#include <DallasTemperature.h>

#include <LiquidCrystal_I2C.h>


#define HEART_RATE_SENSOR_PIN A0


WiFiClient client;


long myChannelNumber = 2326323;

const char myWriteAPIKey[] = "CRTCBIMREYYGS26U";

const int oneWireBus = 15; // D5 on NodeMCU-32

LiquidCrystal_I2C lcd (0x27, 16, 2);


OneWire oneWire(oneWireBus);

DallasTemperature sensors(&oneWire);


// Define threshold values for heart rate and temperature

const int normalHeartRateMin = 60;
```

```
const int normalHeartRateMax = 100;
const float normalTemperatureMin = 36.1;
const float normalTemperatureMax = 37.2;

void setup() {
  Serial.begin(115200); // Start Serial Communication @ 115200
  WiFi.begin("MuhdNur", "mnr0137089943");
  lcd.init();
  lcd.backlight();

  while (WiFi.status() != WL_CONNECTED) {
    delay(200);
    Serial.print("..");
  }

  Serial.println();
  Serial.println("NodeMCU is connected!");
  Serial.println(WiFi.localIP());
  ThingSpeak.begin(client);
}

void loop() {
  sensors.requestTemperatures();
  float temperatureCelsius = sensors.getTempCByIndex(0);
  int heartRateValue = analogRead(HEART_RATE_SENSOR_PIN);
```

```
  // Convert analog value to BPM (adjust this formula based on your sensor
characteristics)
  int heartRateBPM = map(heartRateValue, 0, 1023, 40, 200);


  // Print the BPM value to the Serial Monitor
  Serial.print("Heart Rate (BPM): ");
  Serial.println(heartRateBPM);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("BPM: ");
  lcd.print(heartRateBPM);
  delay(3000);


  {
  Serial.print("Temperature: ");
  Serial.print(temperatureCelsius);
  Serial.println(" °C");
  lcd.clear();
  lcd.setCursor(0, 1);
  lcd.print("Temp: ");
  lcd.print(temperatureCelsius);
  lcd.print(" C");
  delay(3000);
  }
```

```
// Check for high heart rate
if (heartRateBPM < normalHeartRateMin || heartRateBPM > normalHeartRateMax)
{
  Serial.println("Abnormal Heart Rate!");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Abnormal BPM!");
  delay(3000);
}

  // Check for high temperature
  if (temperatureCelsius < normalTemperatureMin || temperatureCelsius >
normalTemperatureMax)
  {
    Serial.println("Abnormal Temperature!");
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Abnormal Temp!");
    delay(3000);
    // You can add additional actions or alerts here
  }

ThingSpeak.setField(1, heartRateBPM);
ThingSpeak.setField(2, temperatureCelsius);
```

```
  ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

  delay(500);

}
```