Faculty of Electronics & Computer Technology Engineering

Development of IoT Based Paralysis Patient Healthcare with Hand Gesture Recognition using ESP32

GAYATHIRI A/P SELVARAJU

Bachelor of Computer Engineering Technology (Computer Systems) with Honours

2024

Development of IoT Based Paralysis Patient Healthcare with Hand Gesture Recognition using ESP32

GAYATHIRI A/P SELVARAJU

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

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Faculty of Electronics & Computer Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI KEJUTERAAN ELEKTRONIK DAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek: Development of IoT Based Paralysis Patient Healthcare with Hand Gesture Recognition using ESP32

Sesi Pengajian: 2023/2024

Saya Gayathiri A/P Selvaraju 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**	

Yang Benar:

(GAYATHIRI A/P SELVARAJU)

NO.97, JALAN UTAMA 15,
TAMAN JAYA UTAMA, 42500
TELOK PANGLIMA GARANG,
SELANGOR

Tarikh: 8/1/2024

Disahkan oleh:

(TS. AHMAD FAIRUZ BIN MUHAMMAD AMIN)

Tarikh: 8/1/2024

DECLARATION

I declare that this project report entitled "Development of IoT Based Paralysis Patient Healthcare with Hand Gesture Recognition using ESP32" 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	:	GAYATHIRI A/P SELVARAJU
Date	:	8/1/2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPROVAL

I approve that this Bachelor Degree Project 2 (BPD 2) report entitled “Development of IoT Based Paralysis Patient Healthcare with Hand Gesture Recognition using ESP32” is sufficient for submission.

Signature :

Supervisor Name : AHMAD FAIRUZ BIN MUHAMMAD AMIN

Date : 8/1/2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

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 Computer Engineering Technology (Computer Systems) with Honours.

Signature	:	
Supervisor Name	:	TS. AHMAD FAIRUZ BIN MUHAMMAD AMIN
Date	:	8/1/2024
Signature	:	
Co-Supervisor Name (if any)	:	DR.MD ASHADI BIN MD JOHARI
Date	:	8/1/2024

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATION

I am immensely grateful to my God for His grace, which has guided me to successfully complete this report. It is with deep appreciation that I dedicate this thesis:

To my beloved mother, *Gunasunthary A/P Balakrishnan*, and father, *Selvaraju A/L Karupia*h, your unwavering love and support fueled my determination throughout this journey.

To my dearest siblings, *Tanaraj A/L Selvaraju* and *Harikanesh A/L Selvaraju*, your constant encouragement kept me motivated, and your belief in me meant the world.

To my supervisor, *Mr. Ahmad Fairuz bin Muhammad Amin* and *Mr. Nadzrie bin Mohamood* your guidance and encouragement were instrumental in shaping the success of this thesis.

To all my friends and classmates, your support was invaluable, and I'm grateful for your presence during the challenges and triumphs.

Thank you all for being part of this achievement.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

The IoT-based paralysis patient healthcare system aims to facilitate communication between patients and healthcare providers or loved ones by utilizing microcontroller-based circuitry. However, current devices in the market are often large and expensive, limiting their availability to hospitals and restricting their use within medical facilities. This creates a challenge for disabled individuals who require constant communication and monitoring. To address these limitations, our project focuses on developing a wearable device that is affordable, user-friendly, and allows patients to regain independence. This device incorporates motion detection capabilities, enabling patients to control it with their hand movements. By utilizing an accelerometer and gyro, the device detects and wirelessly transmits these movements to a receiver system using microcontroller. Temperature sensor value and pulse oximeter sensor value will read and transmit the value into ESP32 microcontroller. The receiver system processes the received commands and displays them on an OLED screen. Additionally, the data is transmitted online to a Blynk server, which is connected to an Mobile phone. The Mobile phone serves as a platform for displaying the patient's vital information and facilitating communication with healthcare providers or caregivers. By monitoring vital parameters such as body temperature, BPM and SpO2, this smart health monitoring system ensures better access to healthcare, improved quality of care, and provides peace of mind to patients and their caregivers. The system offers daily assurance and empowers patients to communicate effectively without the constant presence of a nurse or caregiver. In conclusion, the proposed IoT-based paralysis patient healthcare system addresses the need for affordable, user-friendly devices that enable communication and monitoring. The system's advantages include enhanced healthcare access, improved quality of care, and increased peace of mind for patients and their caregivers.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRAK

Sistem penjagaan kesihatan pesakit lumpuh berasaskan IoT ialah sistem yang direka untuk membantu pesakit menyampaikan pelbagai mesej kepada doktor, jururawat, atau orang tersayang yang duduk di rumah atau di pejabat melalui internet. Sistem ini menggunakan litar berasaskan mikropengawal untuk mengawal fungsi peranti ini. Matlamat sistem penjagaan kesihatan pesakit lumpuh ini adalah untuk membantu seseorang menyesuaikan diri dengan kehidupan lumpuh dengan menjadikan mereka berdikari seboleh mungkin. Peranti sebegini untuk pesakit lumpuh adalah sangat besar dan mahal. Hal ini dibuktikan bahawa peranti ini hanya terdapat di hospital dan tidak boleh digunakan di rumah pesakit atau di masa lapang mereka. Orang kurang upaya mendapati sukar untuk berkomunikasi dengan orang lain. Tiada mekanisme pemantauan khusus disediakan untuk memantau kesihatan orang kurang upaya. Matlamat kami adalah untuk mencipta peranti yang akan dapat melatih semula pergerakan pesakit serta murah untuk mereka bayar tanpa banyak hutang. Peranti ini mampu membantu menjaga kesihatan pesakit tanpa jururawat dan membolehkan pesakit berkomunikasi dengan orang lain. Sistem ini menyediakan pemantauan kesihatan yang menggunakan pengesan bioperubatan untuk memeriksa keadaan pesakit iaitu memantau suhu badan pesakit dan kadar degupan jantung serta menggunakan internet sebagai perhubungan untuk memaklumkan kepada pewaris. Peranti ini menggunakan litar pengecaman gerakan tangan dan litar penerima dan pemancar. Litar gerakan tangan digunakan untuk mengesan pergerakan tangan menggunakan pecutan dan giro dan kemudian menghantar maklumat ini secara tanpa wayar ke sistem aplikasi penerima. Sistem penerima bertujuan untuk menerima dan memproses arahan ini sebelum memaparkannya pada paparan OLED dan menghantar data melalui talian. Pelayan IoT Gecko kemudiannya memaparkan maklumat ini dalam talian untuk mencapai output yang dikehendaki. Kelebihan sistem ini adalah dapat membantu menyediakan akses yang lebih baik kepada penjagaan kesihatan, kualiti penjagaan yang lebih baik, ketenangan fikiran dan jamin kesihatan harian.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ACKNOWLEDGEMENTS

First and foremost, I extend my deepest gratitude to my supervisor, TS.Ahmad Fairuz bin Muhammad Amin, and co-supervisor, Dr. Md Ashadi bin Md Johari, for their invaluable guidance, unwavering support, and patience throughout this project. Their wisdom and mentorship have been instrumental in shaping the success of this endeavor.

I am profoundly grateful to Universiti Teknikal Malaysia Melaka (UTeM) for their financial support, which has played a pivotal role in the completion of this project. This assistance has been pivotal in accomplishing the goals set forth. My heartfelt thanks extend to my colleagues whose willingness to share their thoughts and ideas has contributed significantly to the depth and quality of this project.

To my beloved parents and family members, your love, prayers, and unwavering support have been my source of strength throughout my academic journey. Your motivation and understanding have been instrumental in my accomplishments. A special acknowledgment to my parents and siblings for their unending motivation and understanding. A heartfelt thank you to my classmates for their camaraderie and support throughout this journey. Your companionship has made this endeavor more enjoyable and rewarding.

Finally, my gratitude extends to all the staff members at FTKEK, fellow colleagues, classmates, and Faculty members who have been cooperative and helpful. Your collective support has been invaluable in this academic pursuit.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

TABLE OF CONTENTS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF TABLES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UTeM
اونيۈرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF SYMBOLS

℃	-	Degree Celcius
°F	-	Degree Fahrenheit

LIST OF ABBREVIATIONS

KNN	-	K-Nearest Neighbors
OLED	-	Liquid-Crystal Display
MCU	-	Microcontroller Uni
MPU	-	Memory Protection Unit
GSM	-	Global System for Mobile Communication
Wi-Fi	-	Wireless Fidelity
BP	-	Blood Pressure
ADC	-	Analog-to-Digital Converter
MEMs	-	Micro Electromechanical Systems
IoT	-	Internet of Things
IC	-	Integrated Circuit
MQTT	-	Message Queuing Telemetry Transport
ECG	-	Electrocardiogram
BPM	-	Beats Per Minute
BLE	-	Bluetooth Low Energy
SVM	-	Support Vector Machines
CNN	-	Convolutional Neural Network
CAMSHIFT	-	Continuously Adaptive Mean Shift
FCM	-	Fuzzy-C Means
Zigbee	-	Zonal Intercommunication Global-standard
IMU	-	Inertial Measuring Unit
OLED	-	Organic Light-Emitting Diode
SpO2	-	Saturation of Peripheral Oxygen

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF APPENDICES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 1

INTRODUCTION

1.1 Introduction

Paralysis is the inability to move and, in some cases, to feel in a portion or the entire body. This condition could be transitory or long-term. Recovery time varies from person to person. The population's aging process and the recent drop-in birth rates, as measured by rising life expectancy in wealthy countries. The lack of work, healthcare, and nursing care is thought to be a problem. Low birth rates and aging populations are projected to change in the future. It is well known that a stroke is a disease with a high mortality rate in the elderly, and as the population ages, the prevalence of stroke patients rises steadily.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 1.1: Paralysis[39]

Diplegia refers to a condition where paralysis occurs on both sides of the body, affecting areas like both arms, both legs, or both sides of the face. In contrast, hemiplegia is a type of paralysis that affects only one side of the body, typically involving an arm and a leg on the same side. Monoplegia describes the inability to move one limb, either an arm or a leg. Paraplegia involves paralysis of both legs and, in some cases, the lower torso. Quadriplegia is a condition where all four limbs are paralyzed, often resulting in limited or no movement from the neck down. It's important to understand that while these terms describe specific patterns of paralysis, the underlying causes and severity of paralysis can vary significantly among individuals.

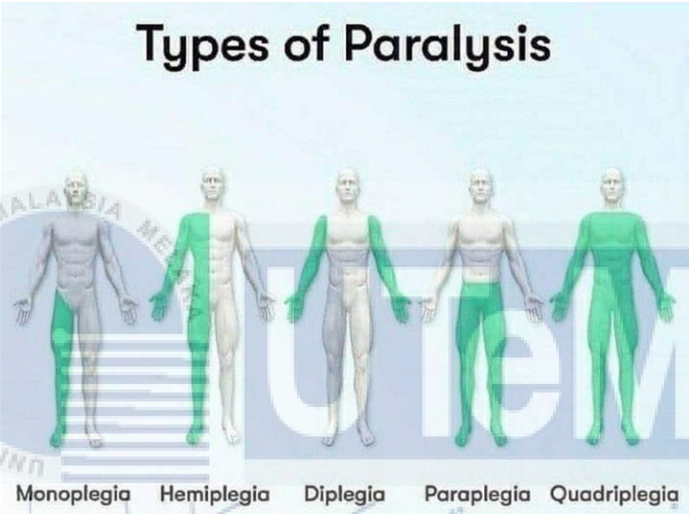


Figure 1.2: Types of Paralysis[40]

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Hemiparesis, which is the partial paralysis of one side of the body, is a common symptom experienced by stroke patients. Individuals with hemiplegia face numerous challenges in their daily lives. Effective rehabilitation and treatment are necessary to improve physical function, alleviate paralysis symptoms, and restore movement in the head/neck, arm, and leg affected by the condition. Initially, individuals with hemiparesis may be unable to actively move the paralyzed limbs, legs, or head/neck. Therefore, external assistance is required to facilitate movement in the affected areas. During the recovery period, patients will need support to move the impaired body part since they are unable to do so independently.

Since the introduction of Bluetooth technology in 2000, wearable electronics have gained popularity and witnessed a surge in demand. Nowadays, people rely on their smartphones to track various aspects of their lives, including their daily steps. This trend has been made possible through continuous technological advancements, which have reduced human effort and significantly improved the quality of life. One way to enhance the monitoring and care of patients is by leveraging GSM technology. This technology allows for the transmission of messages to a designated guardian whenever there are any changes in the patient's hand movements. As a result, the patient is not restricted to the constant presence of a doctor. Even when the patient is at home or elsewhere, their condition can be continuously monitored, ensuring timely assistance and care.

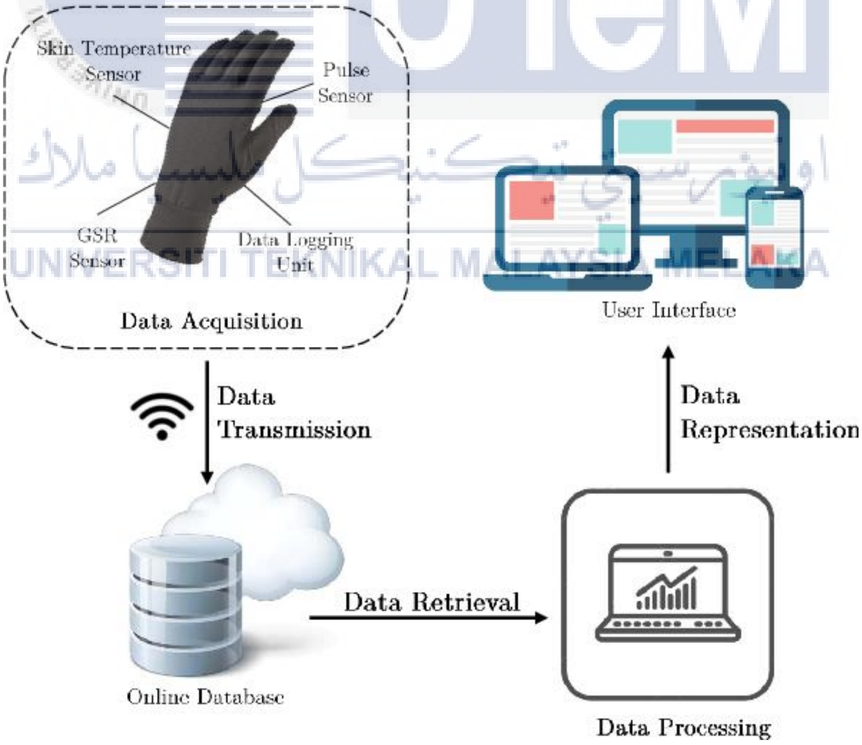


Figure 1.3: Glove-based paralysis patient monitoring system[41]

The proposed system focuses on regular monitoring of vital indicators such as temperature, BPM and SpO2, particularly for patients who may outwardly appear to be in good health but require frequent checks. By tilting the device at specific angles, the user can send different messages, as each orientation corresponds to a different message. The system utilizes an accelerometer to measure motion data, which is then transmitted to a microcontroller. The ESP32 microcontroller processes the data and displays the appropriate message on an OLED screen. The buzzer will sound when the body temperature goes beyond 37.5°C or drops below 35°C. Similarly, it will activate if the heart rate exceeds 100 bpm or falls below 50 bpm. Additionally, when a motion signal is received from the accelerometer, a buzzer is activated, providing an auditory indication along with the displayed message.

1.2 Theoretical Background

A disability is defined as a condition or impairment that significantly hinders an individual's functioning compared to the standard for that person or group. It encompasses various types of impairments, including physical, sensory, cognitive, intellectual, and mental health conditions, as well as chronic illnesses. Disabilities can have effects on organs or physical capabilities and impact a person's engagement in various aspects of life, such as vision, hearing, cognition, learning, movement, mental health, memory, speech, and social interactions.

Monitoring individuals in their homes and communities was initially suggested with the introduction of Holter monitoring in the late 1940s, and its clinical application began in the 1960s. However, the absence of suitable tools hindered comprehensive and continuous monitoring, resulting in intermittent and sometimes invasive monitoring methods being available for several decades.

In recent years, there have been significant advancements in wearable sensors and systems, enabling researchers in patient home monitoring to implement and deploy monitoring technologies effectively. These technological breakthroughs have opened up opportunities for early detection of diseases like congestive heart failure, prevention of chronic conditions such as diabetes, better management of neurodegenerative disorders like Parkinson's disease, and swift responses to emergencies like seizures in epilepsy patients or cardiac arrest in individuals undergoing cardiovascular monitoring. Current research endeavors are primarily focused on developing systems for therapeutic applications. The emphasis on creating and implementing specialized wearable technologies for therapeutic purposes holds the potential to facilitate their adoption in clinical settings within the next five to ten years.

Healthcare systems play a vital role in a country's economy and public health. In today's fast-paced world, it can be challenging for individuals to be constantly available to support their loved ones in times of need. To address these concerns, monitoring systems are being developed to track and cater to the needs of patients.

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.2.1 The elderly patient and Paralysis

The global population of elderly individuals began to rise significantly in the latter half of the twentieth century. Initially, only a small percentage (around 5%) of the population was aged 65 or above, and living past one's 70s was considered uncommon, with those reaching their 80s being revered as wise elders. However, in 1960, one-third of the world's population was over 65 years old. By 2013, the number of people over 65 surpassed the population under 5 years old. It is projected that by 2050, the population over 65 will be more than four times that of children under 5.

Paralysis commonly occurs when the neurological system, particularly the spinal cord, sustains damage. Other significant factors that can cause paralysis include stroke, nerve injury due to trauma, poliomyelitis, cerebral palsy, peripheral neuropathy, Parkinson's disease, ALS (Amyotrophic lateral sclerosis), botulism, spina bifida, multiple sclerosis, and Guillain-Barré syndrome.

During the REM (Rapid Eye Movement) stage of sleep, it is normal for temporary paralysis to take place as a physiological response. However, if there is a disruption in this system, individuals may experience episodes of waking paralysis, where they are temporarily unable to move despite being awake.

1.2.2 Human Activity Recognition

The senior care system plays a crucial role in supporting daily living through the use of Health Activity Recognition (HAR) technology, which offers valuable services. By continuously monitoring the activities of senior citizens, HAR enables the detection of abnormal circumstances and can help mitigate the impact of unexpected events, such as sudden fatigue. These features are essential for wearable technology to promote greater independence and safety among the elderly. Changes in the condition of objects or environmental factors resulting from human activity can also be tracked. Sensor-based recognition systems, utilizing on-body sensors like accelerometers and gyroscopes, are employed to monitor body movements.

Some seniors may initially feel uncomfortable using technology due to lack of trust or unfamiliarity. However, as individuals who grew up in the computer era reach their later years, the senior population is becoming increasingly tech-savvy and embracing technology to improve their health. It is crucial to understand the specific needs of the target user group, a frequently overlooked phase in developing digital health technology. Many connected health products, including wearables, are primarily designed for young people. However, wearables for seniors require thoughtful design considerations as their requirements

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

significantly differ from those of millennials or middle-aged individuals, especially in terms of hearing, vision, and mobility. The design of the device greatly influences a user's desire to interact with it. For example, a senior with arthritis and poor vision is unlikely to find value in a small device with a complex interface and tiny buttons that are difficult to operate.

Wearables offer two-way communication and the potential for integrating these devices into healthcare as seniors become more active participants in their care. Physicians who actively encourage their senior patients to utilize wearables can establish a better communication channel to monitor changes in their health or identify when intervention is necessary.

1.2.3 Communication problems

Paralyzed and elderly patients often experience communication difficulties, which can pose challenges in their interactions with doctors or caregivers. Stroke survivors, in particular, may face issues with communication and understanding, with approximately one-third of them experiencing such challenges. When areas of the brain responsible for language are damaged during a stroke, it can lead to communication difficulties. In most people, the left side of the brain controls language functions. As a result, many stroke survivors develop weakness or paralysis on the right side of their body since each hemisphere of the brain controls the opposite side of the body. Furthermore, if the muscles in the face, tongue, or throat are affected by a stroke, it can also impact communication abilities.

Due to these impairments, individuals may be unable to effectively communicate their needs as they struggle with speech or lack motor control for sign language. Research has shown that a significant number of people die each month due to neglecting their health management, often as a result of a high workload and insufficient time for self-care.

Communication difficulties are prevalent following a stroke, affecting speech, reading, writing, and comprehension. Aphasia, a condition that impairs the ability to understand and use language, is common among stroke survivors, affecting around one-third of them. It can also impact reading and writing skills. Dysarthria, on the other hand, occurs when there is a loss of control over the facial, mouth, and throat muscles, leading to difficulties in speaking clearly. Speech may become slurred, slow, or sound muffled. Apraxia of speech, another communication challenge, occurs when there is an inability to coordinate the facial, mouth, and throat muscles in the correct sequence during the speech, resulting in difficulties in being understood by others.

1.2.4 Societal and Global Issue

Addressing societal and global issues through the implementation of ESP32-based hand gesture healthcare for paralysis patients signifies a significant step toward inclusive healthcare and technological innovation. By leveraging ESP32's capabilities for hand gesture recognition and healthcare monitoring, this initiative tackles several critical societal and global challenges:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1. **Accessibility and Inclusivity**: The utilization of ESP32-based hand gesture technology enables accessibility for paralysis patients, empowering them to communicate their needs effectively. This addresses the societal issue of inclusivity, ensuring that individuals with physical limitations have a voice and means of interaction.
2. **Healthcare Disparities**: Paralysis patients often face healthcare disparities due to limited communication abilities. Implementing ESP32-based systems bridges this gap, providing comprehensive health monitoring and communication tools tailored

to their specific needs. This addresses a global issue of healthcare inequality and strives for more equitable care.

3. **Technological Innovation for Healthcare**: The integration of ESP32 technology showcases innovation in healthcare, demonstrating how advancements in IoT and gesture recognition can significantly impact patient care. This innovation not only aids paralysis patients but also sets a precedent for leveraging technology to address diverse healthcare challenges.
4. **Empowerment and Independence**: ESP32-based hand gesture healthcare fosters patient empowerment by enabling them to communicate and express their needs independently. This empowerment aligns with global initiatives promoting independence and autonomy for individuals with disabilities.
5. **Data-Driven Healthcare**: The monitoring capabilities provided by ESP32-based systems generate valuable health data. This data-driven approach not only aids individual patient care but also contributes to broader research efforts, potentially enhancing understanding and treatment methodologies for paralysis and related conditions globally.
6. **Reduction in Caregiver Burden:** This technology lessens the burden on caregivers by providing an efficient means of understanding and addressing patient needs. It promotes a more independent lifestyle for patients, easing the strain on healthcare providers and families.

In essence, the implementation of ESP32-based hand gesture healthcare for paralysis patients transcends technological innovation by addressing societal inclusivity, healthcare disparities, patient empowerment, and contributing to a more equitable and data-driven healthcare landscape on a global scale.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.2.5 Sustainable Development

Implementing ESP32-based hand gesture healthcare for paralysis patients addresses critical societal and global issues while aligning with Sustainable Development Goals (SDGs). This innovative approach contributes to sustainable healthcare solutions and societal inclusivity in several ways:

1. **SDG 3:** Good Health and Well-being: Leveraging ESP32-based technology for paralysis patient healthcare promotes good health by facilitating accessible and comprehensive healthcare solutions. It empowers patients to communicate effectively and ensures continuous monitoring of vital health metrics, contributing to improved well-being.
2. **SDG 9:** Industry, Innovation, and Infrastructure: The integration of ESP32-based hand gesture technology showcases innovation in healthcare infrastructure, fostering technological advancements and inclusive solutions. It promotes sustainable infrastructure development in the healthcare sector by emphasizing efficient and innovative methodologies.
3. **SDG 10:** Reduced Inequalities: This approach addresses inequalities faced by paralysis patients, enabling them to overcome communication barriers and access essential healthcare. By providing an inclusive platform for healthcare, it strives to reduce disparities and ensure equal opportunities for individuals with disabilities.
4. **SDG 11:** Sustainable Cities and Communities: By promoting sustainable healthcare within communities, ESP32-based solutions contribute to creating accessible and inclusive cities. These technologies facilitate community-based care, empowering local communities to provide better care for paralysis patients.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5. **SDG 17:** Partnerships for the Goals: The implementation of ESP32-based healthcare systems encourages collaboration among technology developers, healthcare providers, and communities. It fosters partnerships for innovative healthcare solutions that benefit society and align with global development objectives.

ESP32-based hand gesture healthcare for paralysis patients aligns with various Sustainable Development Goals by promoting inclusive healthcare, technological innovation, reducing inequalities, fostering community partnerships, and emphasizing environmental sustainability. This approach contributes to building a more equitable and sustainable healthcare landscape globally.

1.3 Problem Statement

a) In an emergency, the patient cannot be continuously watched for any help or assistance and is only available in hospitals which are very large and expensive machines.

b) Disabled persons find it difficult to communicate with others. There is no dedicated monitoring mechanism in place to keep track of the healthof paralyzed people.

c) There is no dedicated monitoring mechanism in place to keep track of the health of paralyzed people.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.4 Project Objective

The main aim of this project is to propose a systematic and effective methodology to create a paralytic healthcare system that is convenient for patients to use. Specifically, the objectives are as follows:

a) To design a wearable device for paralytic patients at an affordable price.

b) To develop a paralysis healthcare system based on IoT using ESP32 to allow them to communicate with others.

c) To develop a mobile application for monitoring patient body temperature, BPM, and SpO2.

1.5 Scope of Project

The scope of this project is as follows:

a) ESP32 is used as a main component for tracking a patient's health.

b) A model designed to recognize hand gestures and translate them into text messages.

c) Blynk App is used to develop mobile applications.

d) This system is designed to be utilized by patients in the comfort of their own homes.

1.6 Limitation

a) The Internet is necessary for individuals to communicate.

b) People who are completely paralyzed and have diplegia in both arms are unable to use it because they cannot produce hand gestures.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The Internet of Things (IoT) refers to a network of physical objects, known as "things," that are equipped with sensors, software, and other technologies to connect and exchange data with other devices and systems through the Internet. These objects can range from everyday household items to complex industrial machinery. These objects are given an Internet Protocol (IP) address in the IoT, allowing them to send data across a network.

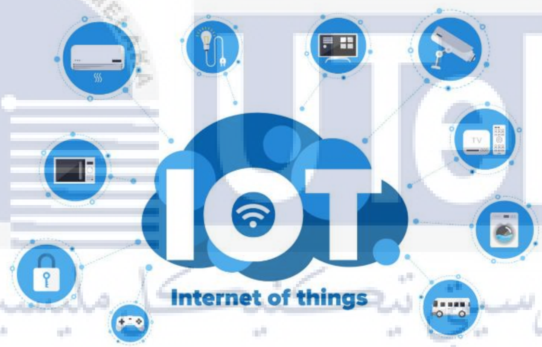
The IoT encompasses various applications, such as a person implanted with a heart monitor, a farm animal tagged with a biochip transponder, or a car equipped with sensors that notify the driver of low tire pressure. It enables these devices to collect and share data, facilitating communication and interaction between the physical and digital worlds.

It is estimated that the number of connected IoT devices will continue to rise significantly, reaching 10 billion by 2020 and 22 billion by 2025, surpassing the current count of over 7 billion devices. This growth presents new opportunities in healthcare, allowing healthcare providers to monitor patients remotely and enabling individuals to monitor their health.

Wearable IoT devices, in particular, have gained prominence in healthcare. These devices, such as smartwatches, fitness trackers, and medical sensors, offer benefits for both healthcare providers and patients. They provide real-time health monitoring, enabling healthcare professionals to track vital signs, detect anomalies, and intervene when necessary. Patients can also use wearable IoT devices to monitor their health, track fitness goals, and receive personalized insights and recommendations.

However, the adoption of wearable IoT devices also presents challenges. Data security and privacy concerns arise as personal health information is transmitted and stored. Interoperability between different devices and systems is another issue, as seamless integration and compatibility are crucial for effective data exchange and analysis. Additionally, ensuring the accuracy and reliability of the collected data is essential for making informed healthcare decisions.

Overall, the IoT and wearable IoT devices have the potential to revolutionize healthcare by improving remote patient monitoring, enabling personalized care, and empowering individuals to take control of their health. Addressing the associated challenges will be crucial in harnessing the full potential of IoT technologies in healthcare.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.1: Internet of Things[42]

2.2 Related Project Research

2.2.1 An IoT-based approach in paralysis patient healthcare system.

Jadhav et al. [1] introduced a design for a healthcare system catering to paralysis patients, utilizing an Internet of Things (IoT) approach. The system incorporates sensor data collection, an IoT-enabled microcontroller, and a web application. The microcontroller acts as the system's central processing unit and connects to the patient's Wi-Fi network and a web server. It transmits all gathered data to a website application, where both the patient and doctor can access and monitor the patient's health status. Furthermore, the microcontroller is equipped with the ability to control appliances in the patient's room in response to commands from the web application.

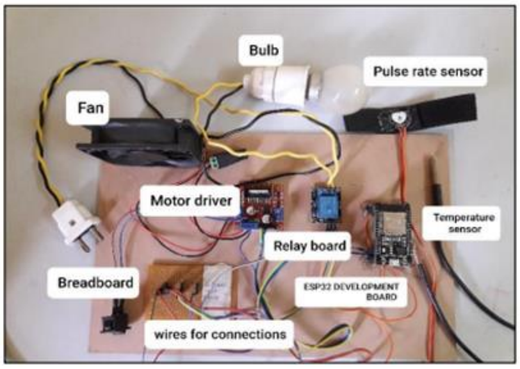


Figure 2.2: Model of IoT-based paralysis patient healthcare system[1]

2.2.2 Designing of automated paralysis patient healthcare system

Naveena et al. [2] developed an automated healthcare system specifically designed for paralysis patients. The system incorporates an accelerometer sensor that can detect gestures, taking into account the variations in the patient's position. The accelerometer sensor provides information about the direction and analog voltage changes on its x, y, and z pins in response to position changes. To convert this analog variation into a digital format, an Op-Amp is employed.

The microcontroller is responsible for receiving and analyzing the data from the accelerometer sensor. Additionally, it triggers a message and a siren based on the analysis results. Through the Internet of Things (IoT), an SMS is sent to the registered caregiver of the patient, displaying the specific message related to the detected gesture or position change. This feature aims to ensure that the caregiver promptly receives relevant information regarding the patient's condition.

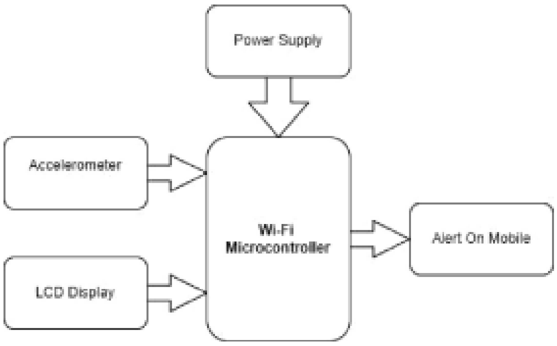


Figure 2.3: Block diagram of automated paralysis patient healthcare system[2]

2.2.3 Safety Wheel Chair for paralysis patient

Al-Sawaai et al. [3] developed a safety wheelchair system for paralysis patients in Oman. The system employs an acceleration sensor to detect falls and a buzzer to raise an audible alert. SMS messages are also sent to caregivers to promptly inform them of fall incidents. Additionally, forward bending detection is utilized using an ultrasonic sensor, triggering alerts and SMS notifications when patients bend their backs. The system incorporates SMS messaging to provide caregivers with specific information based on keypad inputs. This comprehensive approach enhances patient safety and enables caregivers to respond effectively to incidents and patient needs.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

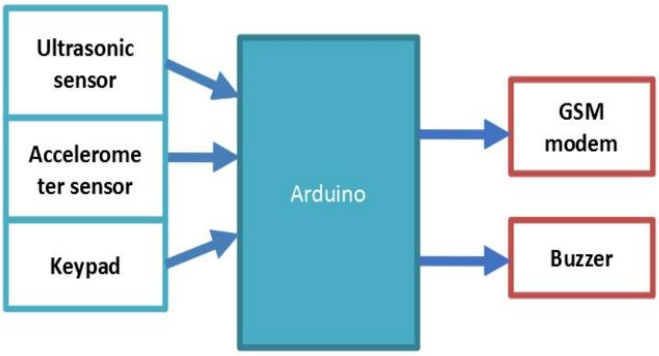


Figure 2.4: Block diagram of Oman's safety wheelchair for paralysis patients [3]

2.2.4 Paralyzed Patient Monitoring Equipment – IoT

Viancy et al. [4] developed a framework for monitoring paralyzed patients using IoT-based equipment. The framework includes a wearable gadget made of a rubberized material that tracks the patient's heart rate and hand motions, providing real-time alerts to the caretaker for any changes. The solution is accompanied by a smartphone app, both for the patient and the care provider. The patient's heart rate is periodically transmitted to the patient's application when the monitoring equipment is activated. The heart rate values are retrieved from Firebase, a real-time database, and sent along with the alert notifications and voice notifications through the Caretaker mobile application, which keeps the values updated. The use of Firebase allows for seamless data retrieval and delivery.

Figure 2.5: Model of paralyzed patient monitoring equipment[4]

2.2.5 GSM-based Paralysis Patient Monitoring System

Sindagi et al. [5] implemented a paralysis patient monitoring system that utilizes a temperature sensor to detect body temperature. The system is based on a GSM architecture, where a gyroscope is attached to the patient's finger. The gyroscope detects the angle at which the object is inclined with the ground by sensing static acceleration caused by gravity. Whenever the patient requires assistance, they adjust the gyroscope in different directions, which serves as input to the gyroscope. The output of the gyroscope, measured in volts, is connected to the controller board, acting as the processing unit. To determine the patient's state, a Wi-Fi transceiver is used to send and receive signals to the GSM module. The patient's heart rate and temperature are displayed using dedicated sensors for temperature and heart rate monitoring, respectively.

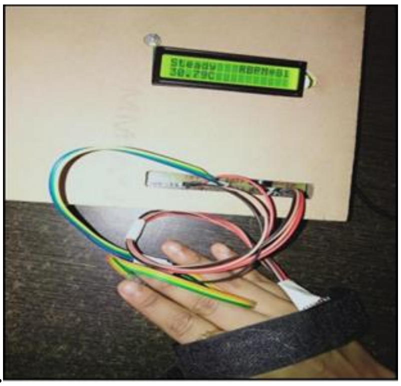


Figure 2.6: Model of GSM-based paralysis patient monitoring system[5]

2.2.6 GSM-based health monitoring system for paralysis patients

Kate et al. [6] developed a GSM-based paralysis health monitoring system that tracks motion using an accelerometer. The system includes a microcontroller that processes the motion data and displays relevant messages on an OLED panel. It also emits a buzzer for audible alerts. Additionally, the microcontroller can receive motion signals from the accelerometer. In cases where there is no immediate response to the displayed message, the patient can tilt the device, triggering the microcontroller to send a customized SMS to their registered caregiver using a GSM modem. This comprehensive system enables efficient communication between patients and caregivers, providing timely assistance and support. Overall, Kate et al.'s GSM-based paralysis health monitoring system incorporates motion tracking, microcontroller processing, message display, audible alerts, and SMS capabilities, providing patients with a means to communicate with their caregivers when immediate assistance is required.

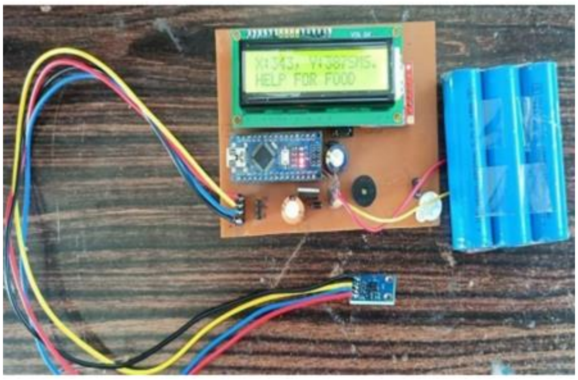


Figure 2.7: Model of GSM-based paralysis health monitoring system[6]

2.2.7 Smart Healthcare Monitoring using IoT

Banka et al. [7] developed a smart healthcare monitoring system using IoT technology to detect and predict diseases or disorders at an early stage. The system continuously monitors various physiological data, including blood pressure, body temperature, and heart rate. Sensors are positioned on the patient's body to collect the data, which is then transmitted to a Raspberry Pi, a credit card-sized single-board computer running the Linux operating system. The Raspberry Pi interfaces with a variety of sensors to capture the patient's vital signs. The collected data is transferred to a database through the Raspberry Pi and can be accessed online from anywhere in the world using a GSM module. This comprehensive system enables continuous monitoring of the patient's health and facilitates remote access to the collected data for analysis and forecasting.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

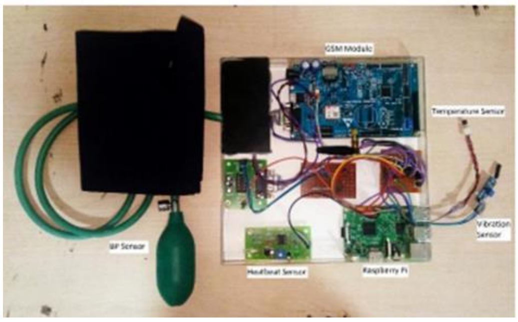
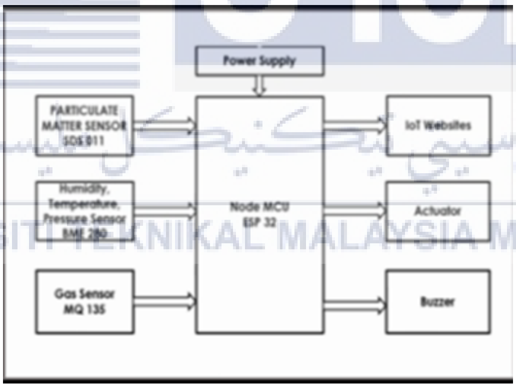


Figure 2.8: Model of smart healthcare monitoring using IoT[7]

2.2.8 Patient rescue and condition monitoring system using IoT

Ponlatha et al. [8] developed a remote patient health monitoring system that allows for the assessment of patients' health conditions. The study explores the use of various wearable health monitoring devices to track physiological data, including body temperature, heart rate, blood pressure, and pulse. A wireless sensor is utilized to collect the data, which is then processed, networked, and computed through the integration of the Internet of Things (IoT) for real-time monitoring. The system includes an ECG acquisition system that provides a clear and convenient display of the user's electrocardiogram (ECG). The pulse rates measured by the system for three subjects (78, 78, and 79 times per minute) are consistent with the results obtained from a medical pulse meter. Overall, Ponlatha et al.'s remote patient health monitoring system utilizes wearable devices, wireless sensors, and IoT integration to track and assess patients' physiological data.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.9: Block diagram of remote patient healthcare monitoring system[8]

2.2.9 Real-Time Health Monitoring System Using IoT

Sali and Dr. Parvathi [9] proposed a real-time health monitoring system based on IoT technology. The system involves processing signals from multiple sensors, which requires signal processing techniques. An Arduino Uno control board equipped with an ATmega328 microcontroller is utilized to process the signals from biomedical sensors. The processed parameters are then transmitted to a healthcare center through a Wi-Fi module. At the healthcare center, the received parameters are stored in the patient database. The system compares the patient's biological parameters with predefined threshold values to determine if the patient's health state is normal or abnormal. In case of an abnormal state, an emergency message is sent to the doctor and patient through the Wi-Fi module to ensure prompt medical attention. The system also features a Graphical User Interface (GUI) that provides detailed information about the patient and allows the doctor to view and monitor the patient's condition. Wireless body sensors are used to collect data from the patient's body, which is then processed by the Arduino Uno controller board connected to the sensors via a Wi-Fi module. The processed data is transmitted to the healthcare facility through the Wi-Fi module, enabling real-time health monitoring.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.10: GUI of patient's body temperature[9]

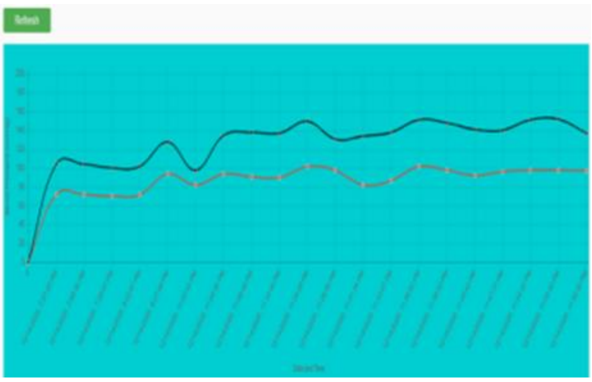


Figure 2.11: GUI of patient's heart rate[9]

2.2.10 Design and implementation of Real Time Health Care Monitoring System based on IoT

Abed and Hussein [10] proposed a system that integrates various medical sensors, including temperature sensors, pulse oximeters, and heart rate monitors, connected to a powerful microcontroller. The microcontroller is then linked to a computer, serving as a display device. The data transfer and display process occurs through a graphical interface designed using the 2013 Vision Basic program. Once the results are displayed, all the medical information is transmitted in real-time via the cloud to a specialist doctor. This allows for comprehensive diagnosis of the patient's condition. By connecting the sensors, microcontroller, display device, and cloud-based data transmission, the system enables efficient monitoring and timely communication of medical information to facilitate accurate diagnosis and treatment.

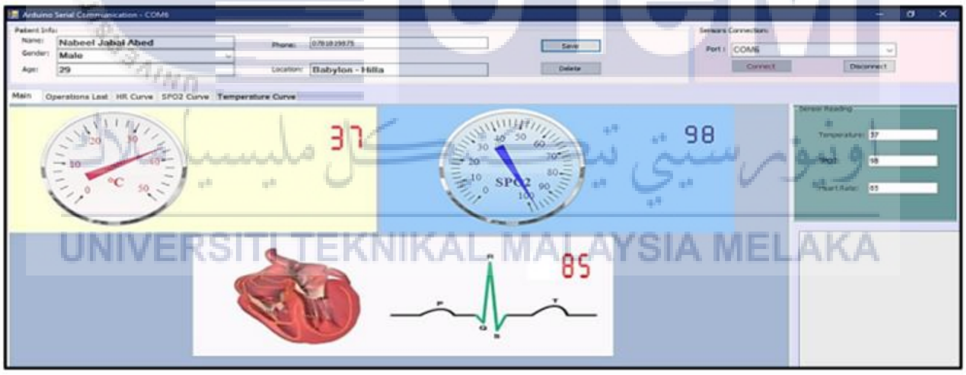


Figure 2.12: The result of the temperature, heart rate, and blood oxygen level of the patient[10]

2.2.11 Automated paralysis patient healthcare system

The approach suggested by Gaikar et al. [11] aims to assist individuals who are paralyzed in communicating using basic hand motions. The system utilizes accelerometers positioned on gloves, with each accelerometer connected to a specific finger. These accelerometers are connected to an Atmega8-powered Arduino UNO microcontroller using connecting wires.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

When the direction of the accelerometer is changed, the initial or stable value of the accelerometer is altered. This change in value is used to determine which pre-coded messages should be displayed. To notify the patients' caregivers, the system is equipped with a beep sound that is triggered when a message is displayed. Overall, this approach utilizes the motion of the fingers, detected by accelerometers, to enable paralyzed individuals to communicate pre-defined messages effectively.

Figure 2.13: Model of automated paralysis patient healthcare system[11]

2.2.12 Health Monitoring System using IoT and Raspberry Pi

Prof. Prasanna et al. [12] proposed a system that utilizes wireless sensors to gather data on various physiological parameters such as temperature, blood pressure, salinity level, and heart rate. These sensors are designed to be worn by individuals to continuously monitor their health. The system allows for the creation of a comprehensive database containing the patient's medical history. This information can be accessed by doctors, who can review and analyze the data as necessary. The data storage can be either permanent on a server or reset using software, depending on the requirements. In case of any unexpected activity or emergency, the system automatically generates alerts that are sent to both family members and physicians. Overall, this proposed system offers a wireless and IoT-based approach to monitor multiple physiological parameters. It provides real-time data collection, storage, and analysis, facilitating proactive healthcare monitoring and timely interventions when needed.

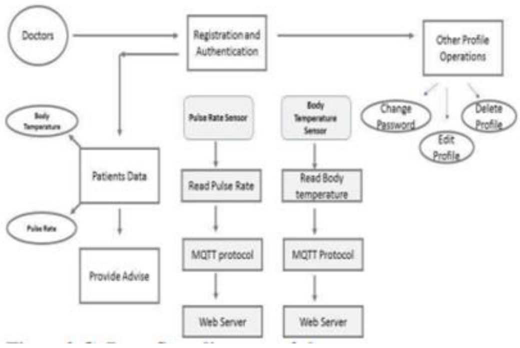


Figure 2.14: Flow diagram of health monitoring system using IoT and Raspberry Pi[12]

2.2.13 Smart Healthcare Monitoring System in IoT

Begum and Dharmarajan [13] proposed a system consisting of two separate circuits: a slave circuit and a master circuit. The slave circuit operates in both regular mode and ECG mode. The system collects data on temperature and humidity, which is then transmitted for data analysis. Additionally, an ECG and pulse sensor are used to detect cardiovascular diseases, taking into account the body posture of the patient. The collected data is uploaded and sent to doctors and caregivers. This allows healthcare professionals to monitor the patient's condition and provide timely assistance when needed, particularly in cases where the patient experiences breathing issues due to severe conditions. Furthermore, if any significant irregularities are detected in the collected data, the system issues a message notification. This alerts the relevant individuals to take immediate action or provide necessary medical attention. In summary, the proposed system operates on separate circuits, collecting and analyzing data on temperature, humidity, ECG, and heart rate.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

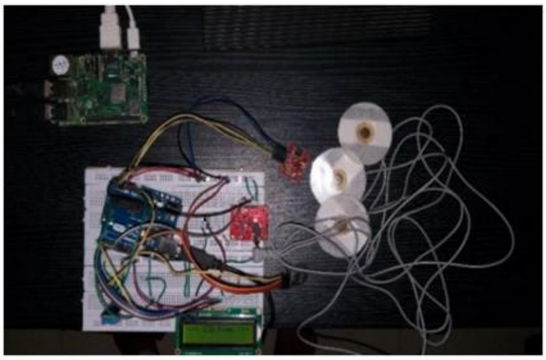


Figure 2.15: Model of smart healthcare monitoring system in IoT[13]

2.2.14 Automated paralysis patient healthcare system

Vidya et al. [14] developed a smart healthcare system designed to cater to the specific needs of paralyzed and mute individuals while also aiding in the diagnosis of heart attacks. The system incorporates various sensors that measure body temperature, heart rate, blood pressure, and oxygen saturation. An oscillator provides a clock signal to synchronize the operations of a microcontroller. A 5V regulated supply is used to provide a stable voltage to power the microcontroller. Upon turning on the device, the sensors are activated to gather the aforementioned physiological data. If the readings fall within the normal range, a green light illuminate to indicate a normal condition. Conversely, if any of the readings deviate from the normal range, a red-light illuminate, signaling an abnormality. In such cases, a message is automatically sent to the doctor, alerting them of the detected anomaly. Furthermore, the system is equipped with a microcontroller that detects hand movements in four different directions. When a patient performs a specific hand movement, such as a gesture, the microcontroller triggers the sending of an SMS to the caregiver. Additionally, the result of the detected hand movement is displayed on an OLED screen, providing immediate feedback to the user. Overall, this smart healthcare system effectively addresses the challenges faced by paralyzed and mute individuals while also assisting in the diagnosis of heart attacks.

Figure 2.16: Model of automated paralysis patient healthcare system[14]

2.2.15 An IoT Based Automated Communication System for Paralyzed Patients using Simple Hand Gestures

Mohana et al. [15] developed a technology aimed at enabling communication for paralyzed individuals using basic hand gestures. The system utilizes accelerometers mounted on gloves, with each accelerometer assigned to a specific finger. These accelerometers are connected to the Atmega 32B microcontroller of the Arduino UNO via connecting wires. When the direction of the accelerometer changes, it affects the initial or stable value of the accelerometer. This change in value is utilized to display pre-programmed messages. The system is equipped with a beep sound that is triggered when a message is displayed to alert the patient’s caretaker. In addition to the mentioned system components, your system incorporates the KNN algorithm. KNN is known for its efficiency, accuracy, and correctness, enabling improved mapping of messages within the system.

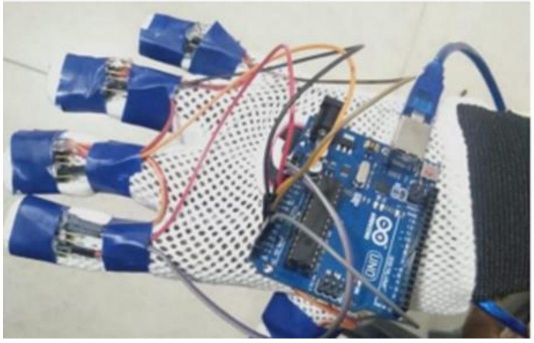


Figure 2.17: Model of IoT-based automated communication system for the paralyzed patient[15]

2.2.16 Hand Gesture Recognition and Voice Conversion System for Dumb and Deaf People

Krishna Rao et al. [16] conducted research on a system that enhances communication for individuals using sign language, enabling them to communicate more effectively with ordinary people. The project utilizes an Arduino controller that is interfaced with flex sensors and a speech playback circuit. For each detected motion, a corresponding speech track is programmed into the system. This allows ordinary people to understand the meaning behind sign language gestures more easily. In addition, the system incorporates a Bluetooth-enabled device. The audio instructions are converted to text using Bluetooth technology and an Mobile app. The text instructions are then displayed on an OLED screen, making them accessible to individuals who are hard of hearing. This feature ensures that the system caters to a wider range of users, including those with hearing impairments. Overall, Krishna Rao et al. [16] developed a system that combines flex sensors, speech playback, Bluetooth, and an Mobile app to facilitate communication between individuals using sign language and those who do not understand sign language. By converting sign language gestures into spoken words and displaying text instructions, this system improves communication and comprehension for both parties involved.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

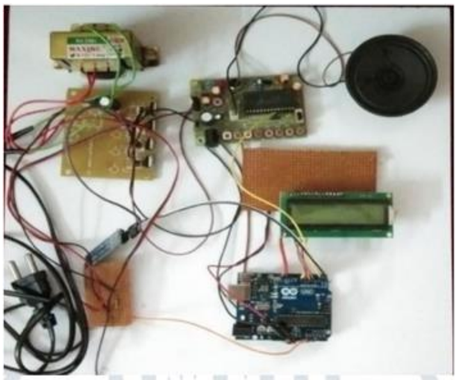


Figure 2.18: Model of hand gesture recognition and voice conversion system for dumb and deaf people[16]

2.2.17 Recognition of Hand Signs Based on Geometrical Features using Machine Learning and Deep Learning Approaches

The research conducted by Josephine Julina Josepha and Thangaswamy [17] focuses on American Sign Language (ASL) motions and the identification of specific action gestures using a single hand. The study involves recording hand signs and saving input videos that display various hand gestures. A customized dataset is created in real-time for further analysis. The collected video feed is processed by transforming it into frames. Pre-processing techniques are applied to eliminate noise from the frames, ensuring better accuracy in gesture recognition. In order to reduce computing costs, the RGB color frames are converted to grayscale. It is emphasized that careful attention must be given to the pre-processing of images as noise can significantly affect the performance of the system. By effectively addressing noise and optimizing image quality, the research aims to improve the overall performance and reliability of the system in recognizing and interpreting ASL hand gestures.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

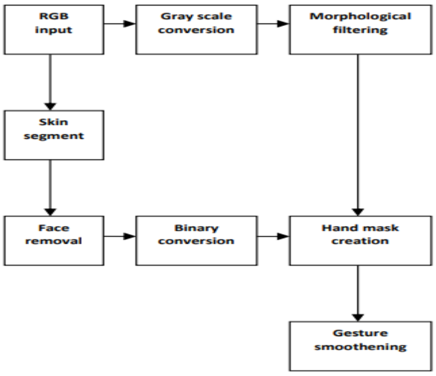


Figure 2.19: Block diagram recognition of hand signs based on geometrical features[17]

2.2.18 Real-Time Hand Gesture Interface for Medical Visualization Applications

Wachs et al. [18] proposed a framework for a tracking module that focuses on interpreting hand motion. The system initially captures an image and crops it to isolate the hand region, enabling a more precise segmentation. Symbolic characteristics, specifically of the Haar type, are extracted from a continuous image stream to identify hand postures. For the sequence of features, a supervised Fuzzy C-Means (FCM) algorithm is employed. The FCM is trained to differentiate between various hand positions, enabling accurate categorization of hand gestures. The recognized gestures are then utilized for various tasks such as displaying X-ray images, selecting patient records from a database, and manipulating items and windows on the screen. The proposed architecture consists of two layers. The lower-level layer focuses on tracking the hand and recognizing gestures, while the top-level layer is responsible for managing the user interface. In summary, Wachs et al. [18] presented a framework that includes a tracking module for interpreting hand motion. The system employs image processing techniques, symbolic feature extraction, and supervised FCM to recognize hand postures.

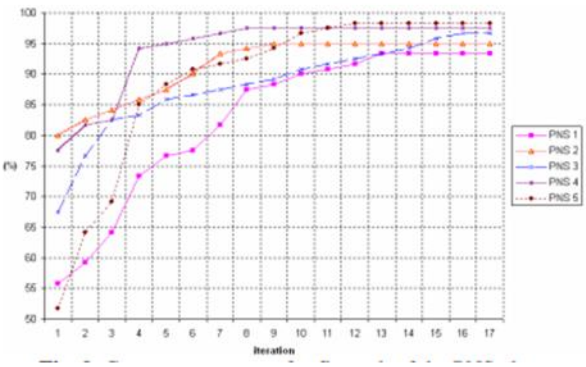


Figure 2.20: Graph of convergence behaviour of PNS algorithm[18]

2.2.19 Hand Gesture Recognition for Patients Monitoring

Muthumeena et al. [19] developed a tool that utilizes real-time hand gesture detection for monitoring hospital patients. The system involves capturing a scene using a webcam. Once the image is captured, the next step is to recognize and isolate the hand from the scene, as only hand gestures are relevant for correct classification. Image processing techniques are employed to create multiple frames. The captured image is initially displayed in the RGB color space and then converted to a new window. Skin-colored pixels are extracted from each frame, and the frames are transformed into binary images. The hand pixels are separated from the rest of the scene using the largest contour of the detected skin patches. The system then analyzes the information regarding hand movements and identifies the respective hand gestures. This information is transmitted to the main monitoring station, enabling healthcare providers to address the patient's needs or detect any emergencies.

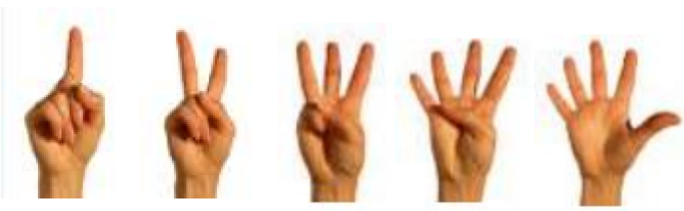


Figure 2.21: Bystanders alert, Nurse alert, Doctor alert, Emergency alert, and Food alert[19]

2.2.20 An M-health Application for Cerebral Stroke Detection and monitoring using Cloud Services

Garcia et al. [20] have developed a framework for a cerebral stroke detection system that leverages cloud technology for data storage and analysis. The system utilizes the Mobile Mobile Vision API and the Mobile API for voice recognition and synthesis. The project focuses on a mobile application designed for detecting cerebral strokes. The user is required to perform three tasks, each corresponding to a common symptom of a stroke. These tasks include smiling, repeating a simple sentence, and raising their arms. The application then analyzes the user's performance in these tasks. Based on the results, the application displays the outcome of the stroke detection process. If necessary, the application sends notifications to the user's family members and medical emergency services to ensure prompt assistance. The study validates the effectiveness of the cerebral stroke detection application developed in this project.

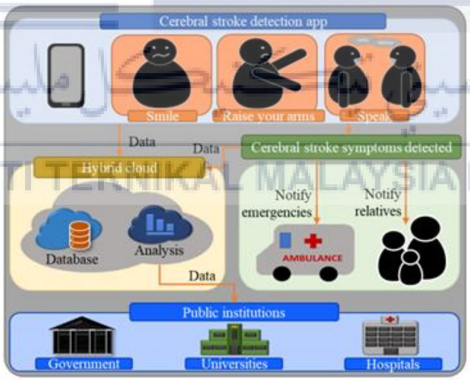


Figure 2.22: Framework of cerebral stroke detection and monitoring using cloud services[20]

2.2.21 Gesture-based Monitoring System for Partially Paralyzed Patients

Joshi Manisha et al. [21] proposed a gesture-based monitoring system specifically designed for partially paralyzed patients. The system utilizes an MPU6050 sensor, which combines an accelerometer and gyroscope, to recognize the hand movements of the patient. In this case, only the acceleration data is sensed to track the patient's movement. The microcontroller ESP32 is employed to receive the sensor data. It utilizes a common IC for communication purposes. The ESP32 then analyzes the received data and identifies left, right, up, and down hand movements based on the acceleration values. Once a gesture is recognized, the acceleration value and the specific gesture are communicated to the Firebase Realtime Database. The Firebase data is then accessed by a website, which requires the same login information provided to the ESP32. The website retrieves the data from Firebase and presents it to the user. Furthermore, the website has a feature that sends alerts to the caregiver or guardian's phone, enabling them to attend to the patient promptly.

Figure 2.23: Model of gesture based monitoring system for partially paralyzed patient[21]

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.2.22 IoT Paralysis Patient Health Care

Godavari et al. [22] developed a system aimed at taking care of paralysis patients and ensuring their safety. The system allows the patient to communicate their needs through small hand motions, enabling quick understanding by those around them. Additionally, the system provides monitoring capabilities for detecting the patient's standing position. When the patient is in bed and makes any movement in a specific direction, a pre-programmed message corresponding to that direction is displayed on the screen. Simultaneously, a notification is sent to the caregiver and the patient's family members' mobile phones. These notifications are also accessible on the cloud, allowing everyone involved to stay informed. Furthermore, if the patient attempts to stand up independently, the system will send a message indicating the patient's need for assistance, and the patient's needs are communicated to the caregiver and family members, and in case of emergencies, appropriate notifications and calls are triggered to ensure prompt assistance.

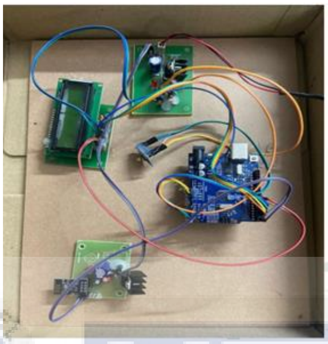


Figure 2.24: Model of IoT paralysis patient healthcare[22]

2.2.23 An IoT-based Wearable Smart Glove for Remote Monitoring of Rheumatoid Arthritis Patients

Raad et al. [23] presented a project that aims to create a Smart Glove system to support older individuals with joint movement disabilities while they are at home. The system utilizes a Smart Glove with E-textile sensors and a microprocessor to measure the range of motion (ROM) of the fingers. The gathered rehabilitation data can be transmitted to physiotherapists for analysis. The integrated microcontroller allows for easy and efficient control of the Smart Glove's functions. Furthermore, the Smart Glove is connected to a Bluetooth module, enabling continuous monitoring of the patient's palm status. In case of any errors or abnormalities, the system notifies the physiotherapist. The Smart Glove also enables monitoring the patient's response to medication and recommended movements. Overall, this system provides valuable insights for physiotherapists, enabling them to track progress and customize treatment plans accordingly.

Figure 2.25: Model of an IoT-based wearable smart glove for rheumatoid arthritis patients[23]

2.2.24 An IoT-based smart glove

Urshlila et al. [24] developed a smart glove called the Infinity Glove, which utilizes the GY-61 3-axis accelerometer, Arduino microcontroller, and HC-05 Bluetooth module. The glove captures gesture data through the accelerometer sensors and converts it into gesture inputs. The Arduino microcontroller receives these inputs and compares them to previously saved gestures. When a gesture input matches a previously saved gesture, the microcontroller triggers the associated action. Additionally, the microcontroller transmits the gesture data to an Mobile application installed on a smartphone via the HC-05 Bluetooth module. The data is also displayed on a 16x2 OLED screen attached to the glove.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.26: Model of an IoT based smart glove[24]

2.2.25 IoT based remote medical diagnosis system using NodeMCU

Faisal & Hossain [25] proposed a remote medical diagnosis system that utilizes a combination of hardware components and web technologies. The system begins by establishing serial connectivity between NodeMCU and Arduino Nano. NodeMCU automatically collects data from the Arduino Nano, including BPM (heart rate) and temperature readings. Next, the system utilizes the Wi-Fi network generated by a Wi-Fi router to transfer the collected data to a remote web server. The data, stored in separate variables, is used to create a URL string that can be accessed using the HTTP "GET" method. To ensure secure and reliable data transmission, Transmission Control Protocol (TCP) is employed. A PHP script receives the real-time data from NodeMCU and stores it in a MySQL database. The stored data is accompanied by a timestamp and identification number, facilitating easy retrieval and organization. Another PHP script is responsible for retrieving the data from the database and displaying it on the system's homepage when the URL of the remote medical diagnosis system (RMDS) is accessed.

Figure 2.27: Model of IoT-based remote medical diagnosis system using node MCU[25]

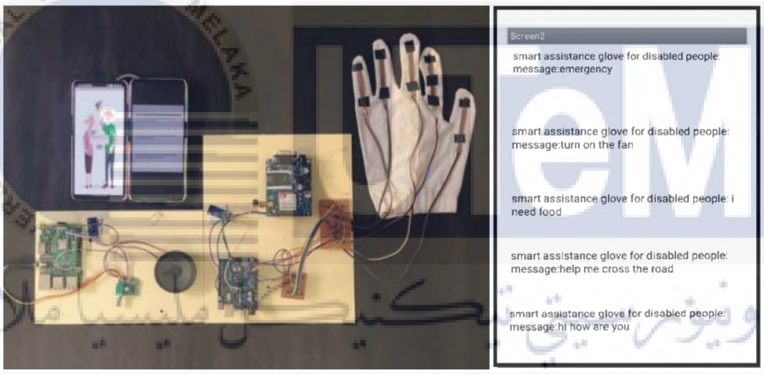
2.2.26 IoT Based Patient Health Monitoring System Using LabVIEW and Wireless Sensor Network

Julius & Jian-Min [26] developed a health monitoring system utilizing LabVIEW, Arduino Mega, and Xbee wireless communication modules. The system incorporates multiple sensors to collect health-related signals from the patient. These signals are processed by the Arduino Mega and wirelessly transmitted using Xbee modules. At the receiving end, the signals are received by another Xbee module and processed in LabVIEW, a graphical programming environment. LabVIEW enables signal analysis and can detect normal and abnormal conditions, such as arrhythmia. The system utilizes the LabVIEW online publishing tool to inspect the signals and send them to a web server. By leveraging the web server, the system allows for remote access and monitoring of the patient's health data. Healthcare professionals can access the data from any location and at any time. If any abnormal health condition is detected, the system can trigger a text message notification to alert doctors or caregivers.

Figure 2.28: Model of IoT-based patient health monitoring system using LABVIEW[26]

2.2.27 Implementation of IoT-based Smart Assistance Gloves for Disabled People

Senthil Kumar et al. [27] developed an IoT-based smart assistance glove designed to assist disabled individuals. The glove incorporates flex sensors that detect movements and translate them into corresponding commands. These commands are then transmitted to a Raspberry Pi using a Lora transceiver for further processing. The results and instructions derived from the glove's movements are presented on a webpage in audio format, ensuring accessibility for users. Additionally, a mobile app is utilized to display the output, providing convenience and ease of use. One of the key features of the system is the ability to notify attendants or caregivers even when they are physically distant from the disabled person. When the user moves or requires assistance, an alert is triggered to inform the attendant. In case of emergencies, such as a fall or sudden health issue, a message and email are sent to the person's emergency contact and the attendant, ensuring prompt attention and support.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.29: Model of IoT-based smart assistance gloves for disabled people[27]

2.2.28 IoT-based Paralyzed Patient Health and Body Movement Monitoring System

Kumar R and Padmaja [28] proposed an IoT-based system for monitoring the health and body movements of paralyzed patients. The system incorporates various sensors to measure key parameters such as heart rate and body temperature. Threshold values were set at 80 beats per minute for heart rate and 34°C for temperature. When the patient's paralysis, temperature, or heart rate exceeds the defined threshold, the data is transmitted to a Raspberry Pi. The Raspberry Pi serves as the central processing unit and receives information from sensors implanted in the patient's head, arms, and legs, specifically the ADXL345 accelerometer sensor. The Raspberry Pi then communicates the collected data to an HC-05 Bluetooth module. The module facilitates wireless transmission of the data to a dedicated mobile application. The mobile application is designed to deliver voice alerts to caregivers or nurses, notifying them of the patient's condition and the need for assistance.

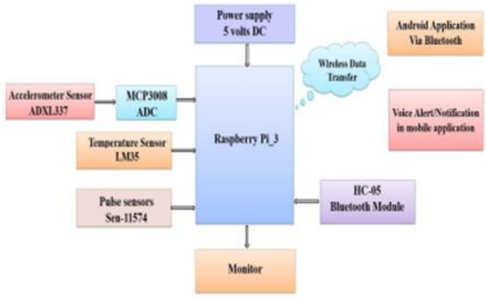


Figure 2.30: Framework of the paralysis patient monitoring system[28]

2.2.29 IoT-based Paralysis Patient Healthcare

Kad et al. [29] developed an IoT-based paralytic healthcare system aimed at transmitting patient data and alerts to doctors, caregivers, and loved ones through the internet. The system utilizes various sensors such as temperature detectors, heartbeat detectors, and blood pressure detectors to continuously monitor and collect patient readings.

The collected data from these sensors is then displayed on TV screens, allowing healthcare professionals to have real-time access to the patient's vital signs. Additionally, an Mobile operating system is developed to enable doctors to remotely monitor patients' conditions and provide necessary interventions. By implementing this system, the number of physical visits by doctors and caregivers to the patient's ward can be reduced while still ensuring continuous monitoring of the patient's health. This not only improves patient care but also allows for early detection of any abnormalities or emergencies. The temperature detector accurately

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

measures the patient's body temperature, the blood pressure detector provides accurate blood pressure readings, and the heartbeat detector monitors the patient's heart rate.

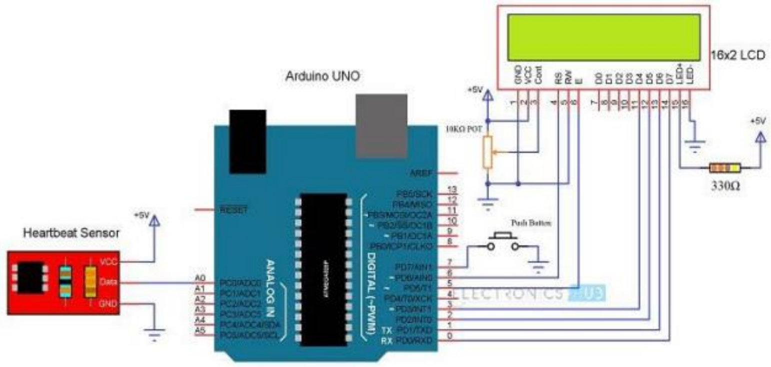


Figure 2.31: Model of IoT-based paralysis patient heathcare[29]

2.2.30 Smart Wearable Hand Device for Sign Language Interpretation System with Sensors Fusion

Lee.B.G & Lee.S.M [30] developed a Smart wearable hand device for a sign language interpretation system using sensor fusion. The system consists of an application module running on an Mobile-based mobile smartphone and a smart wearable hand device that integrates sensor and processing modules. The data required for sign language interpretation is collected from the flex sensor and IMU (Inertial Measurement Unit). These sensors are connected to an Arduino Pro Mini 328, which is equipped with an external resonator and an ATmega328 CPU operating at 16 MHz with a 0.5% tolerance at 5V. The collected sensor data is processed using a built-in Support Vector Machine (SVM) classifier. The SVM classifier uses the extracted features from the sensor data as inputs to recognize and interpret the letters of the sign language alphabet.By combining sensor fusion techniques and machine learning algorithms, the smart wearable hand device is capable of accurately interpreting sign language gestures and translating them into corresponding letters. This enables effective communication between sign language users and non-sign language users.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

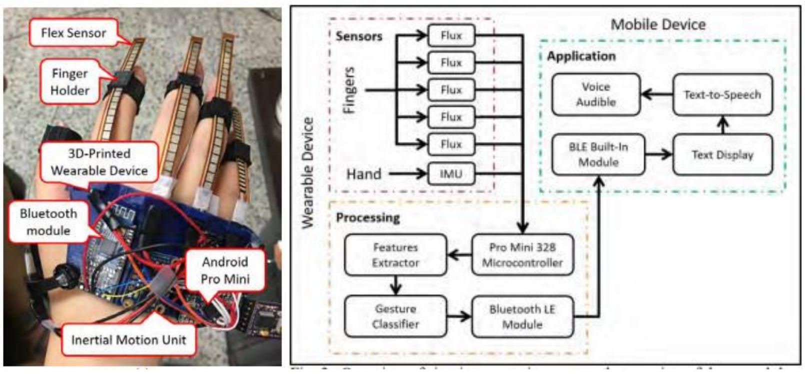


Figure 2.32: Model of smart wearable hand device for sign language interpretation system[30]

2.2.31 Smart glove for hand gesture recognition

Bhavana et al. [31] proposed a framework for smart gloves designed for hand gesture recognition. The framework utilizes Arduino as the microcontroller, which acts as the processing unit for all the connected devices. The smart gloves are equipped with flex sensors and an IMU (Inertial Measurement Unit) to detect hand motions and finger positions in a three-dimensional space. When a gesture is performed, a set of values is obtained from these sensors. These values are then processed to represent or identify the specific gesture. Once the gesture is recognized, the framework communicates the gesture information, along with the corresponding text format, to an external device. In this case, the external device is a mobile phone. Communication between the smart gloves and the mobile phone is established through Bluetooth. On the mobile phone, a Text-to-Speech (TTS) app is installed, which converts the text representation of the gesture into speech output. This enables the smart gloves to provide voice output for the recognized gestures.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 2.1: Range of values for each finger[31]

Finger	Values
Thumb finger	>=970 &&<=985
Index finger	>=710 &&<=735
Middle finger	>=986 &&<=995
Ring finger	>=900 &&<=920
Little finger	>=760 &&>=780

2.2.32 Speaking System for Speech-Impaired People

Abinayaa et al. [32] developed a speaking system specifically designed for individuals with speech impairments. The system consists of a device that connects to a mobile app for communication purposes. Upon turning on the system, it first checks whether the panic switch is activated. If the panic switch is triggered, indicating an emergency, the system immediately traces the user's location and sends an emergency message to the designated caregiver or emergency contact. If the panic switch is not activated, the system registers the user's gestures using input sensors such as flex sensors and accelerometers. These sensors capture and record the user's hand movements and gestures. The recorded gesture data is then compared to a predefined set of gestures stored in the system. If a match is found, the corresponding speech output is generated and transmitted through the mobile application.

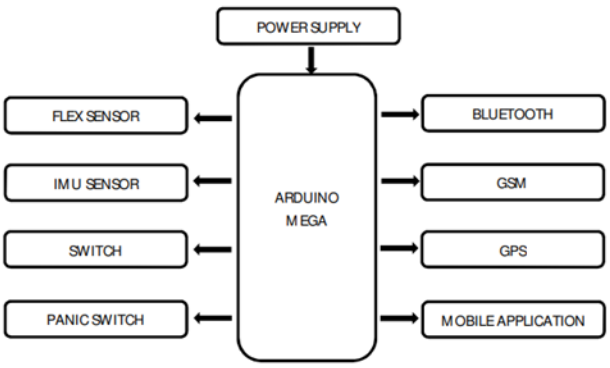


Figure 2.33: Block diagram of speaking system for speech-impared people[32]

2.3 Summarization of all the related research

Table 2.2: Summarization and Comparison of all related research

No	Title of paper	Year / Author	Objective of study	Components used	Disadvantages/lack of project	Advantages of project
1.	An IoT-based approach in paralysis patient healthcare system	(Jadhav et al., 2021)	The purpose of the method is to assist the patient in conveying various messages to other individuals.	DS18B20 temperature sensor, pulse sensor, ESP WROOM32 WIFI + Bluetooth module, L298N 2A-based motor driver, relay boards.	When the pulse sensor becomes unusable, there are no replacement parts available. Before calibration, the DS18B20 waterproof sensors had an average inaccuracy of 3.00%.	The method was developed to enable patients to communicate their needs or requests.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.	Designing of automated paralysis patient healthcare system	(Naveena et al., 2023)	The objective is to design a device that can facilitate the retraining of a patient's movement while enabling independent use and maintaining an affordable price point that allows them to cover the cost personally.	Node MCU ESP8266, SIM900A GSM module, LCD, buzzer.	Liquid crystals serve as the primary technology utilized in LCD (liquid crystal display) flat panel displays. In LCD technology, the blocking of light is essential for its operation. Unlike direct light emission, liquid crystals rely on a backlight or reflector to generate images that can be displayed in color or black and white.	This project aims to assist individuals who are deaf, mute, or have speech impairments to communicate effectively with regular people. It also benefits those who struggle to speak due to paralysis or other conditions.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.	Oman safety wheelchair for paralysis patient	(Al-Sawaai et al., 2020)	The study aims to achieve several goals, including fall detection, forward bending detection, and implementing an SMS service.	Ultrasonic sensor, ADXL335 accelerometer sensor, GSM modem, buzzer, keypad.	The high cost of the prototype is due to the use of ultrasonic technology, which has limitations such as limited testing distance, inaccurate readings, and inflexible scanning methods.	The technology enables patients to move without assistance and provides alerts to both the patient and caregiver by detecting falls and forward bending, allowing them to take timely action to prevent potential accidents.
4.	Paralyzed patient monitoring equipment-IoT	(Viancy et al., 2020)	The main objective of this project is to develop wearable technology that enables patients to establish immediate communication with their caregivers and monitor their	ATmega32U4 microcontroller, pulse sensor Amped (SEN 11574), MPU 6050, HC-05 Bluetooth module.	For certain patients, rubberized materials can lead to skin irritation caused by an allergic reaction to nickel or specific polymers.	The proposed apparatus is designed to be cost-effective and incorporates a system that includes a heart rate sensor and movement sensor. These sensors are capable of

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

			health in real-time.			detecting both the patient's movement and pulse rate.
5.	GSM-based paralysis patient monitoring system	(Sindagi et al., 2020)	The system aims to enhance the autonomy of individuals with paralysis, facilitating their adaptation to daily life.	Microcontroller, temperature sensor, heartbeat sensor, gyroscope, GSM module, Wi-Fi transceiver circuit.	The system uses a temperature sensor to monitor patient body temperature, but it has a higher initial cost and requires a more complex measurement circuit.	Temperature sensors are known for their high precision in measurement. This technology can be conveniently used by patients in the comfort of their own homes.
6.	GSM-based paralysis health monitoring system	(Kate et al., 2022)	A monitoring system is established and employed to oversee the patients' health and detect their vital signs or physiological parameters.	Arduino Uno ATmega328p microprocessor, accelerometer, SIM900A GSM module, LCD.	SIM900A is indeed an outdated module that does not support Bluetooth connectivity..	This facility is designed for family members of non-life-threatening patients who require regular health checks for monitoring purposes.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

7.	Smart healthcare monitoring using IoT	(Banka et al., 2018)	IoT is utilized to provide healthcare solutions that are accessible and available anytime, anywhere. This technology enables the use of interconnected devices and data to improve patient care.	Raspberry Pi microcontroller, LM35 temperature sensor, heartbeat sensor, vibration sensor, BP sensor, MCP3008 ADC, SIM900A GSM module.	The BP sensor is a fragile and inaccurate device, especially without a metal ring to secure the cuff. The system is expensive and utilizes various components such as temperature sensors, vibration sensors, BP sensors, ADC, Raspberry Pi, and GSM modules.	The automated system continuously monitors health indicators, predicts illnesses or disorders, and reduces the need for frequent hospital visits, providing convenience and comfort to patients.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

8.	Patient rescue and condition monitoring system using IoT	(Ponlatha et al., 2023)	The main objective is to develop a remote patient health monitoring system that enables the assessment of patients' health conditions from a distance.	Node MCU ESP32, particulate matter sensor SDS 011, humidity; temperature; pressure sensor BME280, gas sensor MQ135, GSM/GPRS module, actuator, buzzer.	Patients using these systems can authorize the home health monitoring system to access their information exclusively. However, ensuring data security and privacy protection remains a major challenge in the health IoT field.	Patients using these systems can authorize the home health monitoring system to access their information exclusively. However, ensuring data security and privacy protection remains a major challenge in the health IoT field.
9.	Real-time health monitoring system using IoT	(Sali & Dr.Parvathi, 2021)	The system is designed to incorporate a real-time, continuous, and long-term health monitoring system that tracks multiple biomedical parameters	Arduino Uno ATmega328P, LM35 temperature sensor, pulse oximeter(SP02) sensor, heart-rate sensor, blood pressure sensor, Node MCU	This technology allows devices to collect and exchange data using an IoT server, but there is a risk of data breaches and potential limitations in the built-in	The suggested system integrates biomedical sensors to address the concerns of the elderly by offering precise, noninvasive,

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

			including heart rate, blood pressure, ECG, body temperature, and pulse oximetry.	ESP8266, IoT server, LCD.	protection's processing capacity.	consistent, comfortable, low-cost, and adaptable monitoring. It aims to meet all the necessary medical requirements while alleviating worries.
10.	Design and implementation of a real-time healthcare monitoring system based on IoT	(Abed & Hussein, 2021)	This scheme aims to create a monitoring system that provides improved medical care to patients, including those in distant locations.	Arduino Uno ATmega328, LM35 temperature sensor, pulse oximeter, and heart-rate sensor MAX30100, 2013 Vision Basic Program.	Improper finger placement and ambient light can lead to incorrect readings from pulse oximeters and heart rate sensors, which may result in undetected low oxygen saturation levels.	This research has resulted in a low-cost system, and the collected data has been securely stored in the cloud for easy retrieval.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

11.	Automated Paralysis patient healthcare system	(Gaikar et al.,2021)	The system aims to create an affordable GSM-based device that helps patients retrain their mobility and enables independent use.	Arduino Uno ATmega8, accelerometer, SIM900A GSM module, LCD, buzzer, LM324 quad op-amp IC.	The LM324 quad op-amp IC is employed in the system, but its slow response time or slew rate makes it unsuitable for quickly switching output voltage in high-frequency or rapidly changing signal scenarios.	This approach addresses the communication gap, facilitates stress release and expression for paralyzed patients, and provides practical and affordable solutions without excessive debt.
12.	Health monitoring system using IoT raspberry pi	(Prof.Prasanna et al., 2018)	The suggested health monitoring system can measure multiple body indicators and transmit the data to an IoT server using 2G, 3G, or 4G GSM technology.	DHT22 body temperature sensor, Raspberry PI microcontroller, pulse sensor, 2G/3G/4G GSM.	MQTT, using TCP, consumes more memory and computing power. The handshake protocol of TCP requires regular timeouts and wakes, leading to increased battery usage and potential drain.	The suggested method offers a precise, inexpensive, and low-power solution for remote health monitoring. It enhances self-monitoring through its wearable design.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

13.	The smart healthcare monitoring system in IoT	(Begum & Dharmarajan, 2020)	The main goal of this framework is to provide cardiac patients with instant services for self-monitoring their body temperature, heart rate, and body position. It also prioritizes maintaining a hygienic atmosphere.	Arduino Uno ATmega328P, AD8232 ECG sensor, LM35 body temperature sensor, DHT11 humidity and temperature sensor, ADXL335 body position sensor, MAX 30105 heart rate monitor sensor, Bluetooth 4 BLE module, Raspberry Pi, LCD.	The cost of this framework is high due to the utilization of specific components like the LM35, MAX30105, and DHT11 sensors.	This framework helps cardiac patients monitor their heart function, even if their ECG on a single lead graph is not normal. It enables continuous and comprehensive monitoring of their cardiac health.
14.	Automated paralysis patient health care system	(Vidya et al., 2021)	The goal of this project is to create a smart healthcare system that supports paralyzed and mute individuals and aids in the diagnosis of heart attacks.	RF Rx module, 4MHz Oscillator, GSM modem, voice processor, LCD, audio AMP, microcontroller, accelerometer.	This device is not wearable and comes with a higher price tag due to its advanced features and equipment.	The device, equipped with an oxygen saturation monitor, is useful for heart attack patients and those with blood oxygen level concerns. It provides continuous

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

						monitoring and alerts for dangerously low oxygen levels.
15.	An IoT-based automated communication system for paralyzed patients using simple hand gestures	(Mohana et al., 2020)	The project aims to assist paralyzed patients in communication by using finger movements and the KNN algorithm to display relevant messages.	Arduino Uno Atmega 32B, accelerometer motion detection sensor, USB serial connector	The KNN algorithm has limitations with large datasets due to high computational costs. Feature scaling is required to improve its performance, as it involves computing distances between points.	The system can be customized for communication based on the severity of the condition and accessed from any convenient location. It offers improved efficiency, accuracy, and message mapping capabilities.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

16.	Hand gesture recognition and voice conversion system for dumb and deaf people	(Krishna Rao et al., 2019)	This research focuses on providing sign language capabilities to enable effective communication between ordinary people and individuals who are deaf or mute.	Arduino Uno ATmega328, Voice Play back Module base on ISD1820, flex sensor, HC-05 Bluetooth, LCD display	The flex sensor used for gesture movement is sensitive to external pressure, ambient temperature fluctuations, and friction temperature changes.	This technology enables people with speech impairments to communicate with others in real-world situations through hand gesture recognition and voice conversion.
17.	Recognition of hand signs based on geometrical features using machine learning and deep learning approaches	(Josephine Julina Josepha & Thangaswamy, 2021)	This research focuses on the use of American Sign Language (ASL) motions as well as the identification of a few action gestures with a single hand.	Recognition models namely SVM and CNN.	A limitation in this experiment is the inability to effectively handle obstacles caused by variable lighting conditions.	This initiative aims to facilitate communication for individuals who are deaf and mute by recognizing hand signs.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

18.	A real-time hand gesture interface for medical visualization applications	(Wachs et al., 2020)	The objective of a vision-based system capable of interpreting a user's movements in real-time and manipulating objects in a medical data display environment is to enhance the interaction and manipulation of medical data for healthcare professionals.	CAMSHIFT algorithm, FCM algorithm, fuzzy0c classification algorithm, camera, Haar type.	This technique reduces the Haar rectangular positions to a selected set of rectangles for improved efficiency. However, we still need to address false triggers caused by fast-moving objects passing between the hand and the camera.	A vision-based system capable of real-time has the potential to enhance communication and decision-making among healthcare professionals.
19.	Hand gesture recognition for patients monitoring	(Muthumeena et al., 2020)	The primary objective of this project is to develop a tool that utilizes real-time hand gesture detection to monitor hospital patients.	The camera module, a detection module, an interface module, contour extraction, convex hull and convexity defects, Haar-cascade classifier.	The proposed method's performance in this project heavily relies on the accuracy and reliability of hand detection.	This hand gesture recognition application operates on a computer with a webcam, requiring no gloves. Its simple design allows intuitive interaction, making it

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

						accessible. It uses computer vision to interpret hand gestures, facilitating seamless communication.
20.	An M-health application for cerebral stroke detection and monitoring using cloud services	(Garcia et al., 2018)	The objective is to develop a cloud-based framework for our cerebral stroke detection system to securely store and analyze data, enabling efficient storage, advanced analysis, and real-time monitoring of stroke cases.	Hybrid cloud, cerebral stroke detection app, Mobile API 26 (v8.0 Oreo.	The results of cerebral stroke detection tests can be uncertain and vary for each person. The accuracy depends on the proper execution of the test and the skill of the person administering it, as errors can lead to incorrect results.	This system uses advanced technologies to detect symptoms of various diseases and aid in their early detection, leading to prompt and improved healthcare intervention.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

21.	Gesture-based monitoring system for partially paralyzed patients	(Joshi Manisha et al., 2022)	The objective is to utilize a gyroscope to detect even the subtlest body movements in partially paralyzed patients.	ESP32, MPU6050, Firebase, TP4056 battery charging circuit, and push button.	The charging circuit for the TP4056 battery can become hot during use due to heat generated during the charging process.	The gyroscope sensor MPU6050 operates based on the conservation of angular momentum, allowing it to maintain its orientation regardless of the patient's spinning or movements.
22.	IoT paralysis patient healthcare	(Godavari et al., 2022)	The goal is to develop a flexible and cost-effective system that utilizes the Internet of Things (IoT) to track a patient's health state.	Temperature sensor, heart rate sensor, accelerometer, GSM, Wi-Fi module, buzzer, LCD	Heart rate monitoring faces challenges of poor accuracy and chafing issues. Addressing these problems requires advancements in sensor technology, signal processing, and considering user	The efficient use of a reliable heart rate monitoring system reduces the healthcare burden, improves the quality of life, and secures prosperity through technology-driven health

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

					comfort in device design.	protection for humanity.
23.	An IoT-based wearable smart glove for remote monitoring of rheumatoid arthritis patients	(Raad et al., 2019)	Proposing a unique Smart Glove design that provides reliable readings and transmits information to physiotherapists, facilitating remote patient monitoring and informed medication decisions for effective therapy.	LilyPad Arduino microcontroller	The Arduino microcontroller's limited number of input/output pins imposes constraints on the number of sensors that can be used to detect illness signs or immobility, requiring careful selection and alternative approaches.	The proposed solution boasts simplicity, cost-efficiency, and scalability with home-based IoT devices. It was built for less than $100, ensuring affordability, and has low power requirements for sustainability.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

24.	An IoT-based smart glove	(Urshlila et al., 2019)	The goal is to create a wearable IoT-based smart glove with Gesture Recognition technology, allowing users to control automated machinery and appliances through hand gestures for improved convenience and efficiency.	GY-61 3-axis accelerometer, Arduino microcontroller, and Bluetooth module HC-05	The implementation cost of the proposed wearable IoT-based smart glove with Gesture Recognition technology can be relatively high.	The Infinity Glove distinguishes itself with its wireless design and two-axis accelerometer, providing enhanced convenience, faster response, and improved accuracy compared to other gloves for gesture recognition applications.
25.	IoT-based remote medical diagnosis system using node MCU	(Faisal & Hossain, 2019)	To determine the heart rate and body temperature of a person over the internet.	Microcontroller Atmel ATMega328P Arduino Nano, heartbeat sensor, temperature sensor LM35, Wi-Fi module ESP8266 Node MCU	The ESP8266 microcontroller module has limitations in memory capacity, processing power, and overall performance compared to microcontroller.	Remote diagnosis offers several advantages, including cost reduction in therapy, time-saving for doctors and patients, reasonable accuracy, and easing the

						burden of manual data collection through automated processes.
26.	IoT-based patient health monitoring system using LabVIEW and wireless sensor network	(Julius & Jian-Min, 2019)	The goal is to develop a monitoring system that can provide real-time measurement of a patient's temperature, detect specific cardiac function issues, and continuously track their heart rate.	Arduino Mega 2560, ZigBee or Xbee modules, GSM module SIM900A, temperature sensor LM35, ECG sensor, and heartbeat rate sensor.	The LabVIEW Web Publishing Tool has limitations in sharing retrieved data with remote users, requiring the provision of a specific link. Alternative solutions may be needed for seamless and convenient remote data access.	Improved health and quality of life, especially for the aging population, can be achieved through real-time assistance and interventions. Continuing care at home after discharge can prevent emergencies and readmissions, promoting better long-

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

						term health outcomes.
27.	Implementation of IoT-based smart assistance gloves for disabled people	(Senthil Kumar et al., 2021)	The proposal suggests utilizing Arduino Uno and Raspberry Pi to create smart assistance gloves designed for individuals with disabilities.	Flex sensor, Arduino Uno, GSM module, wireless serial port module, Raspberry Pi 3B	The wireless serial port module has the drawback of using fewer lines for transmission between devices, which can result in slower transmission speeds. The reduced number of lines limits the data bandwidth and can affect the overall performance of the wireless communication.	The suggested model benefits paralyzed, deaf, and mute patients by presenting output commands through an Mobile app, facilitated by wireless serial port modules that ensure quick and secure data transfer.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

28.	IoT-based paralyzed patient health and body movement monitoring system	(Kumar. R & Padmaja, 2021)	The study aims to create a portable, sensor-based prototype for monitoring body movements and spinal cord damage, providing accurate assessment and facilitating medical evaluation and rehabilitation.	Accelerometer sensor ADXL337, MCP3008 ADC, Temperature sensor LM35, Pulse sensor Sen-11574, Raspberry pi_3, HC-05 Bluetooth module	The mentioned equipment may face limitations in usability for people with disabilities and can be prohibitively expensive, hindering its accessibility and affordability for widespread use.	This method aids in improving the overall bustle of the body movement preservation apparatus, allowing resources to be grouped easily. This prototype model enables paralyzed patients to complete their tasks with the assistance of a caregiver.
29.	IoT-based paralysis patient healthcare	(Kad et al., 2022)	The suggested system keeps track of the body's physiological data, including heart rate.	Arduino Uno, LCD, Wifi module, blood pressure sensor, heartbeat sensor, temperature sensor, buzzer, GSM, accelerometer.	People with disabilities cannot wear this equipment.	This system's benefits include better healthcare access, higher standards of treatment, mental tranquility, and

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

						daily assurance.
30.	Smart wearable hand device for sign language interpretation system with sensors fusion	(Lee.B.G & Lee.S.M, 2018)	The idea is to create a custom-made wearable device using a 3D printer to hold the hardware components.	Atmega328 processor, flex sensor, finger holder, 3D-printed wearable device, Bluetooth module, mobile pro mini, inertial motion unit.	This system is very expensive. The main drawback of an IMU is that they are susceptible to mistake that builds up over time, often known as "drift." As a result of constantly evaluating changes to itself.	A text-to-speech feature that rapidly turns received texts into sound outputs is also offered in the mobile application.
31.	Smart glove for hand gesture recognition	(Bhavana et al., 2019)	The major objective of the suggested system is to create a smart glove for Internet of Things-based real-time gesture recognition.	Flex sensor, IMU (inertial measuring unit), HC-05 Bluetooth module, Arduino Uno	The main drawback of an IMU is that they are susceptible to mistake that builds up over time, often known as "drift." As a result of constantly evaluating changes to itself.	The less expensive glove uses sensors and eliminated any risks the sensors would have posed to users. It is done so that communication between them is not restricted

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

						by the use of sign language, which is converted into text and voice.
32.	Speaking system for speech-impaired people	(Abinayaa et al., 2021)	The project's purpose is to bridge the communication gap between the disabled and the general public.	Atmega 2560, IMU sensor (MPU-6050), flex sensor, HC-05 Bluetooth module, SIM800 GSM module, NEO-6M GPS module	NEO-6M GPS module may cause loss of power.	Because the device is portable, hearing and speech-impaired individuals can communicate more easily. The output of the motions will be conveyed to traditional individuals in the form of speech.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.4 Summary

Remote patient monitoring is a prominent application of IoT devices in the healthcare industry. It involves the collection of health indicators, such as heart rate, blood pressure, and temperature, from patients who are not physically present at a healthcare facility. This eliminates the need for patients to travel or manually gather data themselves. When IoT devices capture patient data, it is transmitted to a software application where healthcare professionals and/or patients can access it. Algorithms can be applied to analyze the data, enabling the generation of recommendations or alarms. For instance, an IoT sensor detecting a significantly low heart rate can trigger an alarm to alert healthcare practitioners for timely intervention.

While remote patient monitoring offers numerous benefits, ensuring the security and privacy of the highly personal data collected by IoT devices remains a significant challenge. Safeguarding measures, such as robust data encryption, authentication, and access control, are necessary to protect sensitive patient information from unauthorized access or breaches. Overall, remote patient monitoring with IoT devices has the potential to transform healthcare delivery by providing convenience, cost-effectiveness, and improved access to healthcare services. It enables real-time monitoring, early detection of health issues, personalized treatment plans, and proactive healthcare management. However, ensuring the privacy and security of patient data should be a top priority in implementing such systems.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 3

METHODOLOGY

3.1 Introduction

Body vitals, including body temperature, BPM and SpO2 are crucial for assessing a person's well-being and developing an effective treatment plan. However, the traditional approach of manually gathering vital statistics from a large population can be both challenging and prone to errors. To address this issue, we propose a digitally calibrated and real-time vital measurement tool that can accurately capture and transmit data for expert consultation. Our solution enables real-time monitoring of vital signs, such as body temperature, BPM, and SpO2 providing valuable information to enhance health tracking records and enable better healthcare management. The system utilizes wearable devices, such as affordable gloves, which are easily accessible for patients to purchase and use. These gloves are equipped with accelerometers that detect changes in direction, resulting in corresponding variations in measured values.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

By analyzing the accelerometer data, pre-coded messages like "I need water" or "Emergency" can be displayed on OLED and indicating the need for immediate attention. In addition, a buzzer emits a sound to alert caregivers or other individuals, ensuring effective communication and prompt response to critical situations. An LED will blink when heart rate detected. Additionally, the sensor data, including the body temperature, SpO2 and BPM readings, will be displayed on OLED and sent over the internet to inform the caregivers. The mobile application Blynk will display the relevant information. The Blynk database retains recorded body temperature, SpO2, and BPM values, allowing access to this data for a period of up to three months. Overall, our digitally calibrated and real-time vital dimension tool enhances patient care, enables early detection of abnormalities, and promotes effective communication between patients, caregivers, and healthcare professionals.

3.2 Project Planning

The Gantt chart is a vital tool for project planning and management. It visually displays all the project tasks and steps, ensuring effective coordination and resource allocation. It helps keep the project on track and ensures timely completion by identifying critical paths and monitoring progress. Overall, the Gantt chart is essential for conveying information and ensuring the project is finished successfully within the planned timeline.

Table 3.1: Gantt Chart for PSM1

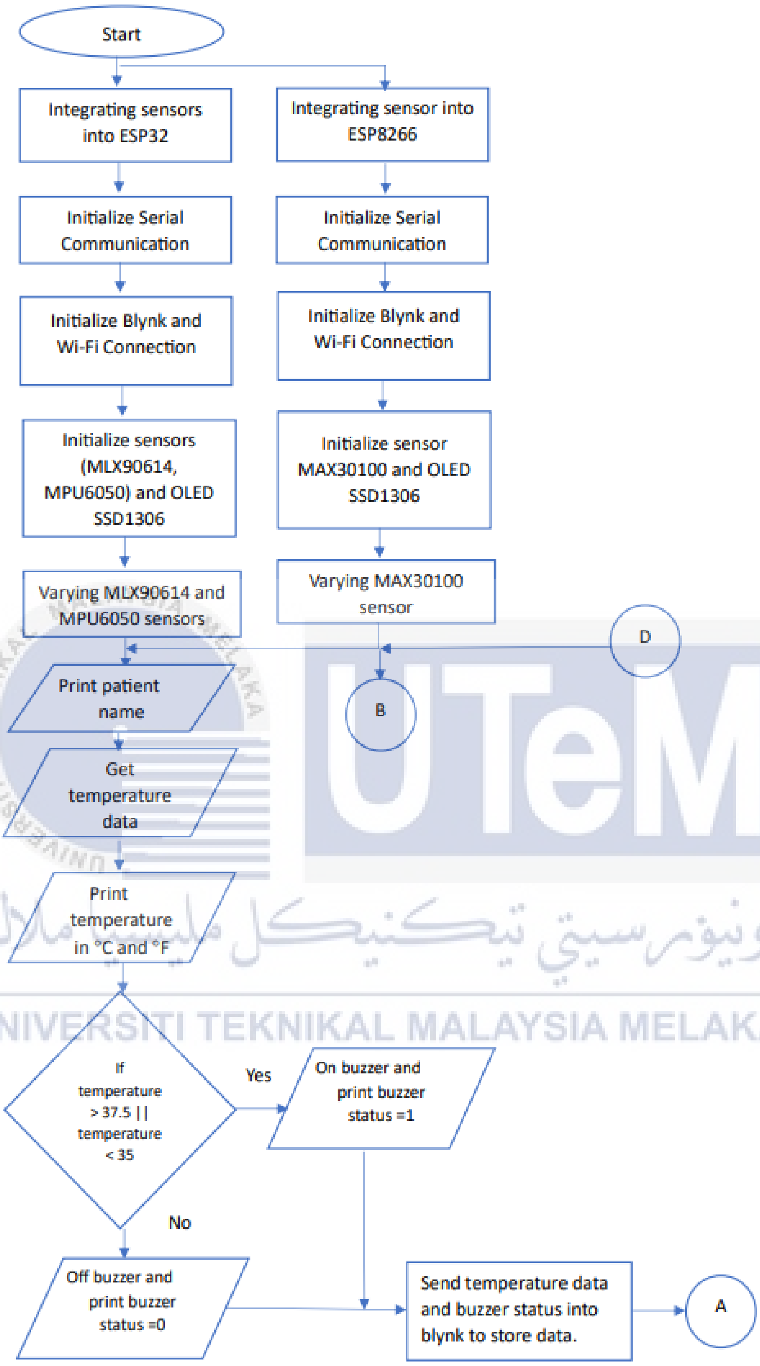
	PROJECT PLANNING FOR PSM 1													
Project Activity	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
BDP 1 briefing by JK PSM	■													
Title Submission	■													
PSM module and format briefing		■												
Research for Chapter 2 Literature review		■	■	■										
Writing Literature review		■	■	■										
Research for Chapter 1 Introduction					■									
Writing introduction, theoretical background, problem statement, objective, scope and summary					■									
Research for Chapter 3 Methodology						■	■	■	■					
Writing hardware and software description, project planning, flowchart and project flow						■	■	■	■					
Hardware purchase (preliminary result)										■				
Preliminary result integration of microcontroller and sensors										■	■	■		
Writing expected result and conclusion for Chapter 4 and 5										■	■	■		
Check and correction of Chapter 1, 2, 3, 4 and 5												■		
First draft submission													■	
Submission for Turnitin check													■	
Submission of report													■	
Prepare for presentation slide													■	■
BDP 1 presentation														■

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

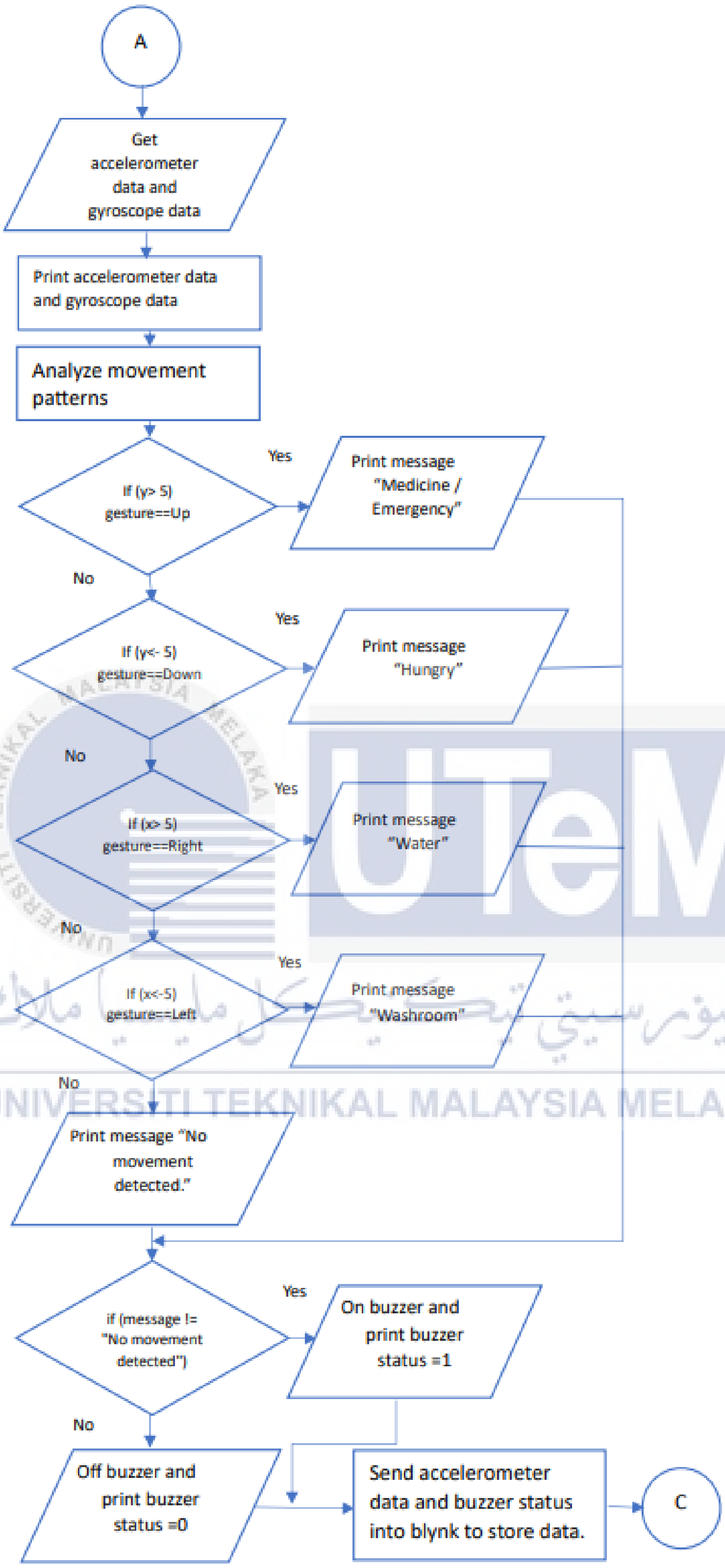
Table 3.2: Gantt Chart for PSM2

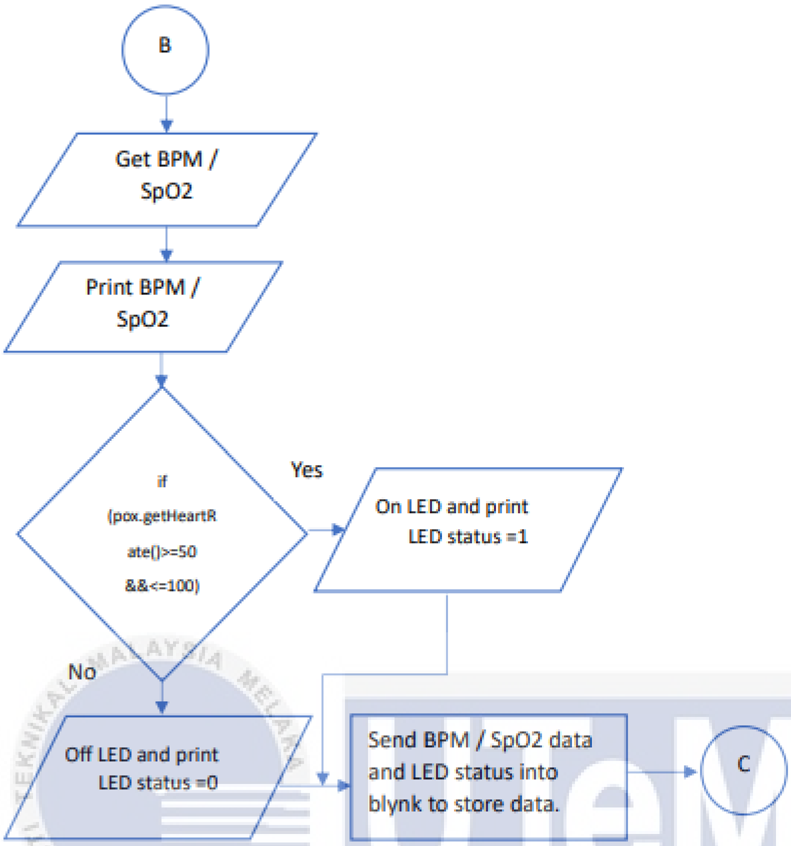
	PROJECT PLANNING FOR PSM 2													
Project Activity	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
BDP 2 briefing by JK PSM	■													
Hardware purchase	■	■												
Development of paralysis healthcare system		■	■	■	■	■	■	■						
Mobile Application development using Blnyk App		■	■	■	■									
Integrate mobile app with hardware system		■	■	■	■									
Whole project testing						■	■	■						
Test, run and troubleshoot							■	■						
Correction of Abstract, Chapter 1, 2 and 3									■					
Writing result and conclusion for Chapter 4 and 5								■	■	■				
First draft submission											■			
Second draft submission												■		
Submission for Turnitin check												■		
Submission of final report													■	
Prepare for presentation slide													■	■
BDP 2 presentation														■

3.3 Flowchart of IoT based Paralysis Patient Healthcare using ESP32



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

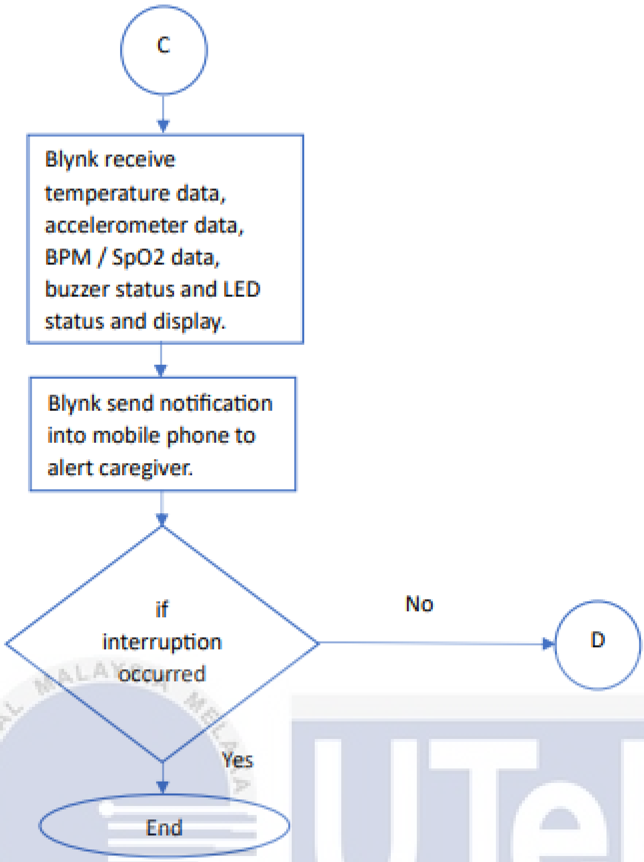

A
Get accelerometer data and gyroscope data
Print accelerometer data and gyroscope data
Analyze movement patterns
If (y> 5) gesture==Up
Yes
Print message "Medicine / Emergency"
No
If (y<- 5) gesture==Down
Yes
Print message "Hungry"
No
If (x> 5) gesture==Right
Yes
Print message "Water"
No
If (x<-5) gesture==Left
Yes
Print message "Washroom"
No
Print message "No movement detected."
if (message != "No movement detected")
Yes
On buzzer and print buzzer status =1
No
Off buzzer and print buzzer status =0
Send accelerometer data and buzzer status into blynk to store data.
C



The circuit will be worn by the patient.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.1: **Project flow for a patient using an IoT-based healthcare system**



The mobile phone using the Blynk server will be used by the caregiver.

Figure 3.2: Project flow of mobile phone using Blynk server

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.4 Proposed methodology

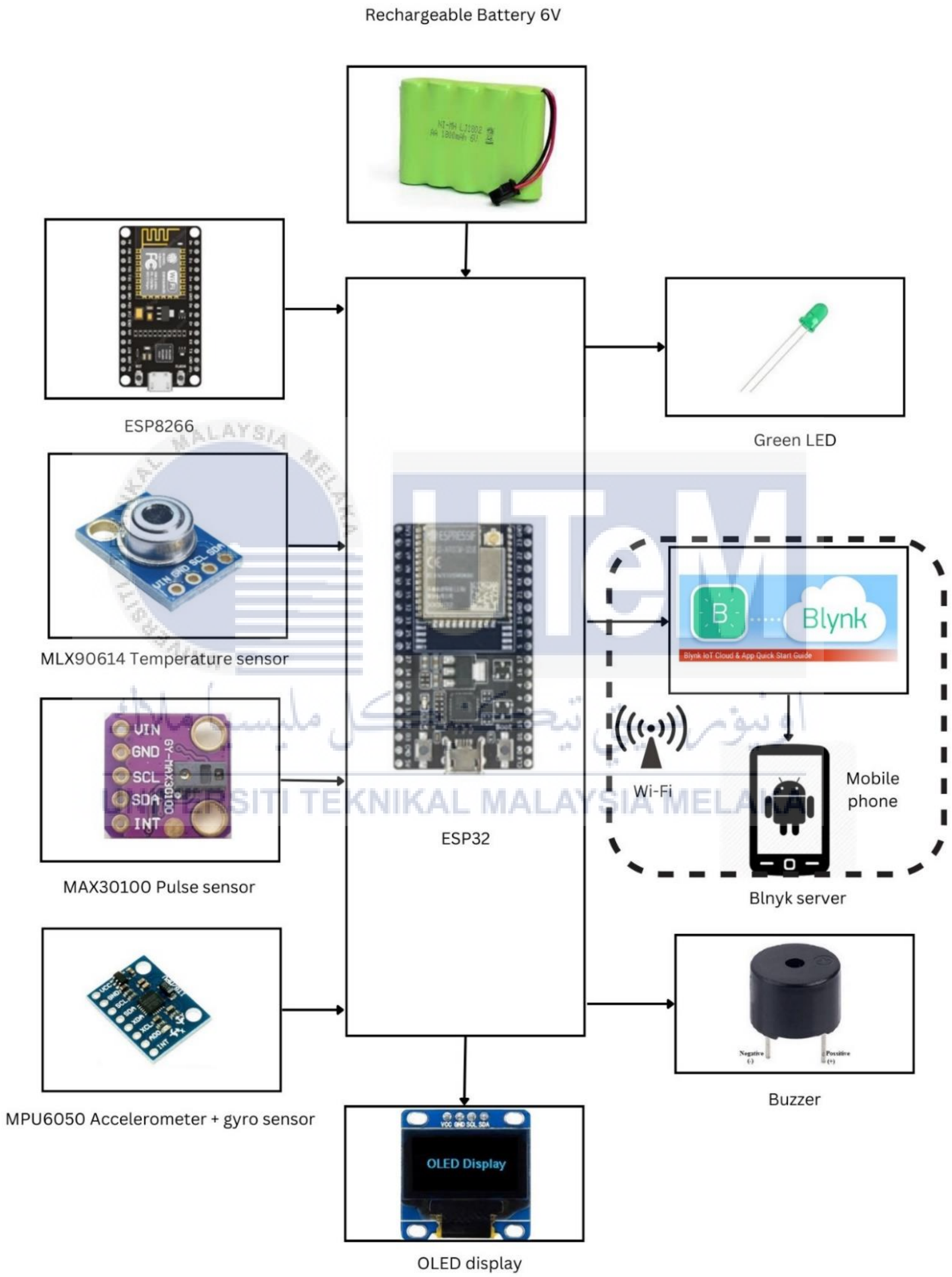


Figure 3.3: Proposed Framework

3.5 System configuration process

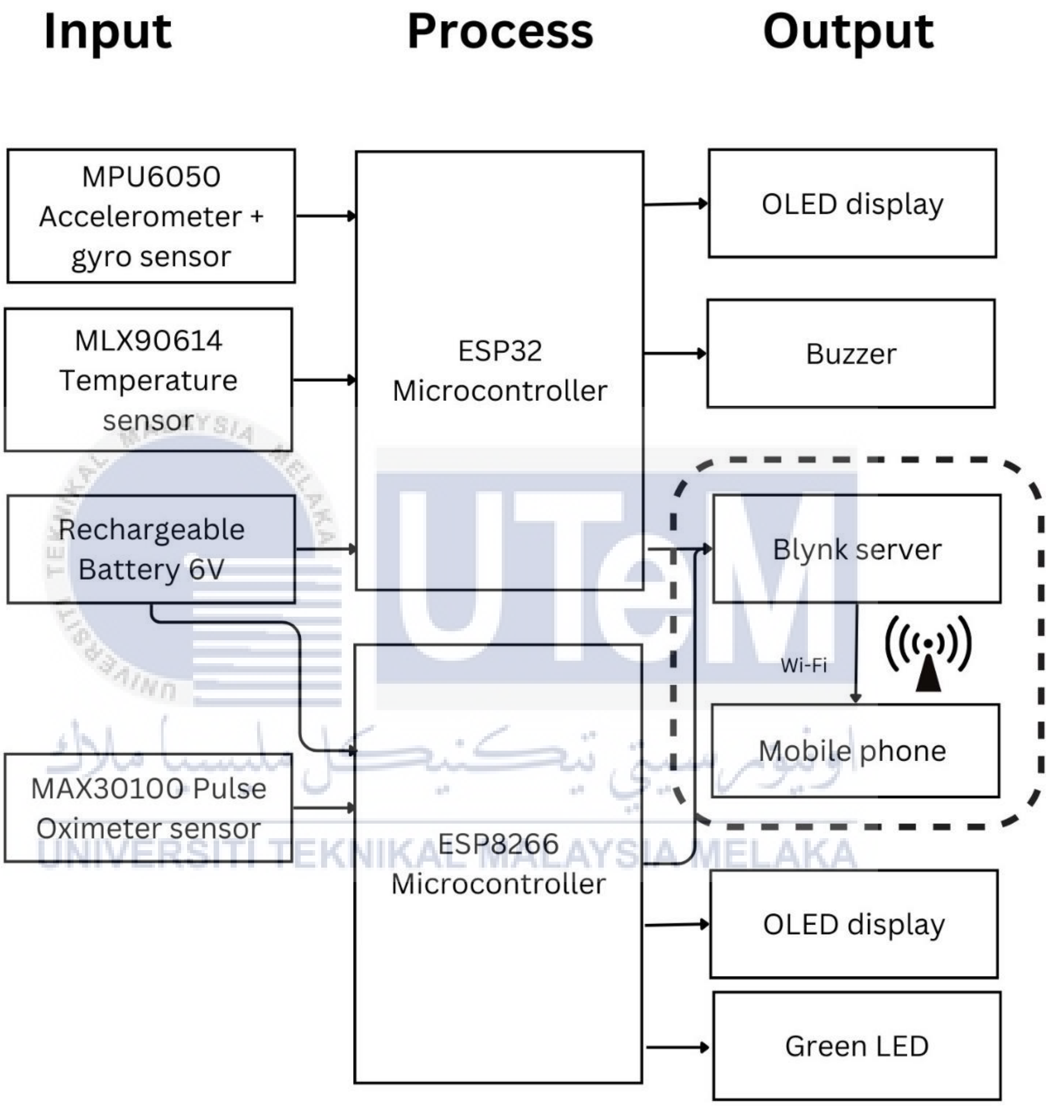


Figure 3.4: Block diagram of IoT based paralysis patient healthcare

3.5.1 Block diagram explanation

The main focus of this project is to utilize the ESP32 microcontroller as the core hardware element for a healthcare system dedicated to monitoring the well-being of paralyzed patients. The ESP32 with ESP8266 function simultaneously carries out mathematical and logical operations and manages all connected peripheral devices based on the provided code. The primary objective of the system is to keep track of the health status of paralyzed patients. To achieve this, the system incorporates a temperature sensor to monitor body temperature and a pulse oximeter sensor to measure the patient's heart rate in beats per minute (BPM) and SpO2. The temperature sensor connected to ESP32 utilizes two distinct functions for temperature retrieval: `mlx.readObjectTempC()` retrieves the temperature in Celsius (°C), while `mlx.readObjectTempF()` retrieves it in Fahrenheit (°F) from the MLX sensor. The MLX90614 infrared temperature sensor directly outputs temperature readings without the need for mathematical conversions. The functions mlx.readObjectTempC() and mlx.readObjectTempF() retrieve temperature values in Celsius and Fahrenheit, respectively from the sensor. The MLX90614 provides the temperature directly through mlx.readObjectTempC() function, which fetches the temperature in Celsius without requiring manual mathematical calculations. To convert Celsius to Fahrenheit, Temperature in Fahrenheit (°F)=Temperature in Celsius (°C)$\times \frac{9}{5}$+32. This formula converts the Celsius temperature reading obtained from mlx.readObjectTempC() to Fahrenheit. 9/5 represents the conversion factor from Celsius to Fahrenheit. It's 1.8 in decimal form, indicating the temperature change per degree between the two scales. 32 is the adjustment or offset value added to align the zero points of Celsius (0°C) and Fahrenheit (32°F) scales. Additionally, an accelerometer and a gyroscope are connected to the ESP32 to detect hand gestures and relay that information to the microcontroller. Using an MPU6050 sensor, it's

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

possible to interpret hand gestures made by a patient, such as up, down, left, and right movements. By capturing these gestures' motions and patterns, the system can translate them into predefined messages displayed on an OLED screen. This setup enables patients, especially those with limited mobility or communication abilities, to convey their needs or messages without vocal communication. The MPU6050 sensor can detect accelerations and gyroscopic movements, allowing it to sense changes in hand positioning or gestures. By implementing specific algorithms or gesture recognition techniques, the system can recognize distinct hand movements, associate them with predefined codes, and display corresponding messages on the OLED screen. This technology aids patients in expressing their requirements or messages efficiently, offering them a means of communication and interaction despite physical limitations.

The pulse oximeter sensor connected to ESP8266 employs photoplethysmography to detect changes in blood volume within any organ of the body. These changes in blood volume correspond to variations in light intensity. The code utilizes the function `pox.getHeartRate()` to retrieve the heart rate information from the pulse oximeter sensor and acquire the BPM (beats per minute) value. Additionally, it employs `pox.getSpO2()` to obtain the SpO2 (blood oxygen saturation) level from the sensor. The MAX30100 sensor estimates blood oxygen saturation (SpO2) by assessing the absorption of light at distinct wavelengths. It employs red and infrared LEDs to measure the difference in light absorbed by oxygenated and deoxygenated hemoglobin, estimating the ratio of oxygenated hemoglobin to total hemoglobin in the blood. This ratio directly translates into a SpO2 value. By detecting changes in light absorption due to blood volume variations with each heartbeat, the MAX30100 computes heart rate. It evaluates the time intervals between peaks in the detected signal to determine the beats per minute (BPM). Both calculations rely on complex internal algorithms within the sensor and library functions to process raw data. These

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

functions pox.getSpO2() and pox.getHeartRate() provide straightforward access to vital physiological parameters without necessitating an understanding of the intricate mathematical computations or the underlying sensor processes.

These sensors such as temperature sensor MLX90614, pulse oximeter sensor MAX30100 and accelerometer sensor MPU6050 are integrated into a glove, which will be worn by the paralyzed patient. The gathered data from the sensors, as well as the detected gestures, body temperature, BPM and SpO2 will be presented on an OLED display. Upon displaying the data, a buzzer will emit a sound as an alert to caretaker if the value exceeds threshold value. The buzzer will sound when the body temperature goes beyond 37.5°C or drops below 35°C. Similarly, it will activate if the heart rate exceeds 100 bpm or falls below 50 bpm. A green LED will be used to blink when the heart rate is detected.

Moreover, these microcontrollers then process the received data and generate the appropriate message based on the input. This message is subsequently displayed on the screen. The buzzer is triggered when hand motion is detected by the accelerometer by providing additional alert. When a motion signal is received from the accelerometer, the buzzer emits a sound, and the ESP32 microcontroller displays a corresponding message on the OLED screen. Furthermore, the system is designed to send the collected data to a Blynk server, which allows the patient's family members and other loved ones to stay informed about the patient's condition. The Blynk application will send notification into caregiver’s mobile phone if hand gesture detected, temperature value and bpm value reached above threshold value. The temperature data, BPM/SpO2 data, accelerometer data will be transmitted to mobile application Blynk and will be stored into Blynk database. A Blynk server is utilized to facilitate this functionality, enabling extended family members and doctors to easily access and monitor the patient's health status on their smartphones from anywhere in the world.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6 Hardware Description

3.6.1 ESP32

Figure 3.5: ESP32[38]

Table 3.3: ESP32 Description

Specification	**Description**
Microcontroller	ESP32
Features	Built-in Wi-Fi and Bluetooth capabilities
Size (LxWxH)	6 cm x 3 cm x 1 cm
Usage	Control center for system operation
Advantages over ESP8266	Enhanced functionality and features
Integrated functionalities	Wi-Fi and Bluetooth combined, no additional modules required
Components	Integrated antenna, RF balun, power amplifier, low-noise amplifiers, filters, power management module
Design efficiency	Components designed for minimal PCB space
Technology	TSMC 40nm low-power technology
Wireless capabilities	2.4 GHz dual-mode Wi-Fi and Bluetooth
Memory	448KB ROM, 520KB SRAM, two 16KB RTC memories
Data handling	Efficient storage and processing
IoT applications	Enables direct device-to-cloud communication
Communication flexibility	Supports peer-to-peer networks for device-to-device data exchange

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.2 ESP8266

Figure 3.6: ESP8266[51]

Table 3.4: ESP8266 Description

Specification	**Description**
Microcontroller	ESP8266
Features	Integrated Wi-Fi
Size (LxWxH)	5.5 cm x 2.5 cm x 0.9 cm
Usage	Embedded Wi-Fi module for IoT applications
Advantages over others	Low-cost, power-efficient Wi-Fi connectivity
Integrated functionalities	Wi-Fi connectivity, no additional modules required
Components	Wi-Fi connectivity, no additional modules required
Design efficiency	Compact design for space-constrained applications
Technology	TSMC 55nm low-power technology
Wireless capabilities	2.4 GHz single-mode Wi-Fi
Memory	512KB ROM, 64KB instruction RAM, 96KB data RAM
Data handling	Limited but suitable for basic IoT tasks
IoT applications	Enables basic device-to-cloud communication
Communication flexibility	Limited peer-to-peer capability

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.3 Temperature sensor MLX90614

Figure 3.7: MLX90614[37]

Table 3.5: MLX90614 Description

Specification	Description
Sensor type	MLX90614 infrared temperature sensor
Measurement range	Accurately measures object temperature from -70°C to 380°C
Output temperature units	Provides temperature readings in Celsius (°C) and Fahrenheit (°F)
Communication interface	I2C (Inter-Integrated Circuit) protocol
Features	Non-contact temperature sensing, high accuracy, factory calibration
Size (LxWxH)	10 mm x 10 mm x 5.2 mm
Technology	Infrared thermopile sensor technology
Application areas	Medical devices, industrial temperature monitoring, automotive
Compact design	Small form factor for integration into various systems
Power consumption	Low power consumption for energy efficiency

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.4 Pulse oximeter sensor MAX30100

Figure 3.8: MAX30100[43]

Table 3.6: MAX30100 Description

Specification	Description
Sensor type	MAX30100 pulse oximeter and heart-rate sensor
Measurement capabilities	Measures heart rate (BPM) and blood oxygen saturation (SpO2) levels
Operating voltage and input current	1.8V to 3.3V and 20 mA
Size (LxW)	3mm x 3mm
LED types	Integrates red and infrared LEDs for photoplethysmography (PPG)
Sampling rate	Configurable sampling rates up to 320 Hz
Communication interface	I2C (Inter-Integrated Circuit) protocol
Low power consumption	Optimized for low-power operation
Motion artifact resilience	Includes built-in motion artifact detection and reduction
Integrated ambient light rejection	Minimizes interference from ambient light for accurate readings
High sample rate capability	Ability to support high sampling rates for data acquisition
Fast data output capability	Capable of providing rapid data output

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.5 Gyroscope and Accelerometer sensor MPU6050

Figure 3.9: GY-521 MPU6050[36]

Table 3.7: MPU6050 Description

Sensor Module	MPU6050
Functionality	Enables gesture detection by integrating a gyroscope and accelerometer
Gyroscope	Measures rotational speed or angular rotation (expressed in degrees per second - deg/s)
Accelerometer	Detects proper acceleration, indicating the rate of change of velocity in an object's rest frame; distinct from coordinate acceleration in a fixed coordinate system
Module Specifications	GY-521 module designed for MPU-6050 MEMS sensor; incorporates a 3-axis gyroscope, a 3-axis accelerometer, a digital motion processor (DMP), and a temperature sensor
Size (LxW)	20mm x 16mm
Communication Protocol	I2C (Inter-Integrated Circuit) protocol
Communication Interface	Two-wire interface: SCL (clock) and SDA (data) for efficient communication
Application	Integration of MPU6050 module enables accurate gesture detection, allowing the system to interpret recognized gestures and provide relevant feedback

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.6 OLED Display

Figure 3.10: OLED[35]

Table 3.8: OLED Description

Specification	Description
Display Type	Monochrome graphical OLED
Size	0.96 inch / 24 mm
Resolution	128 x 32 pixels
Board Size	28 mm x 28 mm
Effective Display Surface	11 mm x 23 mm
Visual Angle	> 160°
Input Voltage	3.3 V ~ 5 V
Drive IC	SSD1306
Operating Temperature	-30°C to 80°C
Communication Protocol	I2C (Inter-Integrated Circuit) protocol
Communication Interface	Two-wire interface: SCL (clock) and SDA (data) for efficient communication
I2C Address	0x78 (or 0x3c, default)
Advantages	Smaller volume: Ultra-low power consumption: High Contrast: Display dot self-luminous; Wide voltage support

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.7 Piezo Buzzer

Figure 3.11: Piezo Buzzer[45]

Table 3.9: Piezo Buzzer Description

Component	**Description**
Device	Buzzer / Beeper
Type	Auditory signaling device capable of producing sound
Variants	Electromechanical, Piezoelectric, Mechanical
Function	Converts electrical audio signal into audible sound
Power Source	Typically powered by DC voltage
Applications	Timers, alarm systems, printers, computers, various electronic devices
Sound Types	Alerts, music, bells, sirens, depending on design and configuration
Operating Principle	Utilizes piezoelectric materials: Piezo crystals, when subjected to electric potential, undergo pressure variation, causing mechanical vibrations; resulting in sound wave production
Structure	Piezo buzzer contains piezo crystals between two conductors; application of potential difference causes crystals to compress and expand, generating sound waves

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.6.8 LED

Figure 3.12 Green LED[49]

Table 3.10: LED Description

Specification	Description
Type	Light-Emitting Diode (LED)
Color	Green
Size (LxW)	8.7mm and 5.0mm
Forward Voltage	1.8 - 2.2 volts
Forward Current	20 - 30 milliamps (mA)
Luminous Intensity	Measured in millicandela (mcd) or lumens (lm)
Wavelength	Approximately 620 - 750 nanometers (nm)
Viewing Angle	Various angles available, often around 120 - 160 degrees
Operating Temperature	-40°C to 85°C
Lifespan	Typically rated for tens of thousands of hours of use
Function	LED lighting products produce light up to 90% more efficiently than incandescent light bulbs.

3.7 Software description

3.7.1 Arduino IDE

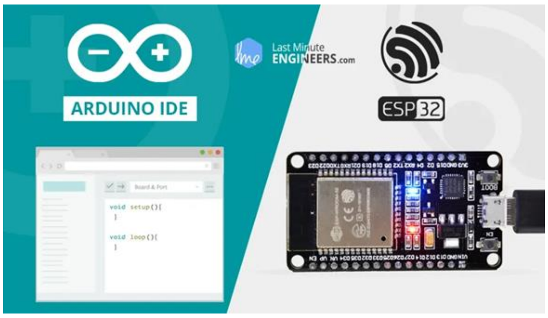


Figure 3.13: Arduino IDE[47]

Table 3.11: Arduino IDE Description

Specification	Description
Name	Arduino IDE
Type	Open-source Integrated Development Environment (IDE)
Purpose	Development and compilation of code for Arduino boards
Official Support	Provided by Arduino
Operating Systems	macOS, Windows, Linux
Platform	Java
Interface	User-friendly with built-in functions and commands
Features	Editing, debugging, and code compilation within the IDE environment
Code Components	Users write code (sketch) in the editor; IDE's compiler translates code into machine-readable instructions for the microcontroller
Programming Languages	C, C++
Code Compilation Process	Code is compiled into a hex file for the microcontroller
Code Deployment	Hex file is uploaded onto the Arduino board's microcontroller to execute the desired functionality
Libraries and Examples	Extensive library support and examples to aid project development

3.7.2 Blynk

Figure 3.14: Blynk[46]

Table 3.12: Blynk Description

Specification	**Description**
Name	Blynk
Platform	Comprehensive platform for remotely controlling IoT devices over the internet
Supported Apps	iOS and Mobile apps available for remote device control
Compatible Devices	Arduino, Raspberry Pi, ESP boards, and various hardware connected via Wi-Fi, Ethernet, or other means
Functionality	Allows remote control, sensor data monitoring, data storage, visualization, and various IoT-related tasks
Flexibility	Not limited to specific boards or shields; adaptable to various hardware choices
Interface Design	Blynk App offers an intuitive interface creation by drag-and-drop of widgets
Components	- Blynk App: User-friendly interface for widget-based project design - Blynk Server: Handles communication between smartphone and connected hardware - Blynk Libraries: Available for major hardware platforms for seamless server communication and command management
Server Options	Blynk Cloud or the option to set up a private Blynk server
Open Source	Open-source platform with the capability to manage thousands of devices

3.8 Bill of Materials (BOM) Table

ITEM NO.	COMPONENT	PRICE	QTY.	COST
1	ESP32-CP2102	RM17.99	1	RM17.99
2	ESP8266 V3 LOLIN(CH340G)	RM10.70	1	RM10.70
3	MLX90614	RM55.99	1	RM55.99
4	MAX30100	RM9.40	1	RM9.40
5	MPU6050-GY521	RM9.90	1	RM9.90
6	PIEZO BUZZER	RM2.40	1	RM2.40
7	GREEN LED	RM0.30	1	RM0.30
8	OLED	RM14.90	2	RM28.80
9	GLOVE	RM4.00	1	RM4.00
10	BREADBOARD	RM3.90	2	RM7.80
11	JUMPER WIRE	RM3.70	2	RM7.40
12	RECHARGEABLE BATTERY 6V + USB CHARGER	RM31.20	1	RM31.20
13	MICRO USB 5V	RM3.00	1	RM3.00
14	ELECTRONIC PROJECT BOX (8x6x4)	RM12.00	1	RM12.00
		TOTAL	COST	RM 200.88

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.9 Power Consumption (WATT)

Components	Operating Voltage (V)	Operating Current(mA)	Power Consumption(W)
ESP32	3.6V	180mA	0.648
ESP8266	3.6V	120mA	0.432
MLX90614	3.6V	2mA	0.0072
MAX30100	3.3V	140mA	0.462
MPU6050	3.46V	0.5mA	0.00173
OLED	3.3V	20mA	0.066
TOTAL		462.5mA	1.61693

Battery Voltage = $\frac{1.61693}{462.5mA}$ = 3.496 ≈3.5V

6V(2000mAh) rechargeable battery was chosen because 6V rechargeable batteries are commonly available and can provide a higher voltage than the required 3.5V for the circuit. This extra voltage could allow for extended operating time before needing a recharge or replacement. This battery provides longer lifespan compared to smaller ones, providing better value in the long run.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Battery Life Calculation

Battery Life = Battery Capacity (mAh) / Load Current (mA)

= 2000mAh / 462.5 mA

= 4 hours 19 minutes

After a duration of 4 hours and 19 minutes, the battery requires recharging and should be placed into the charger for replenishment.

3.10 Hardware Development

3.10.1 ESP32 Wroom 32 Pin Declaration

The proposed circuit for the ESP32 involves utilizing pins V5, G13, GND, 3V3, G22, and G21.

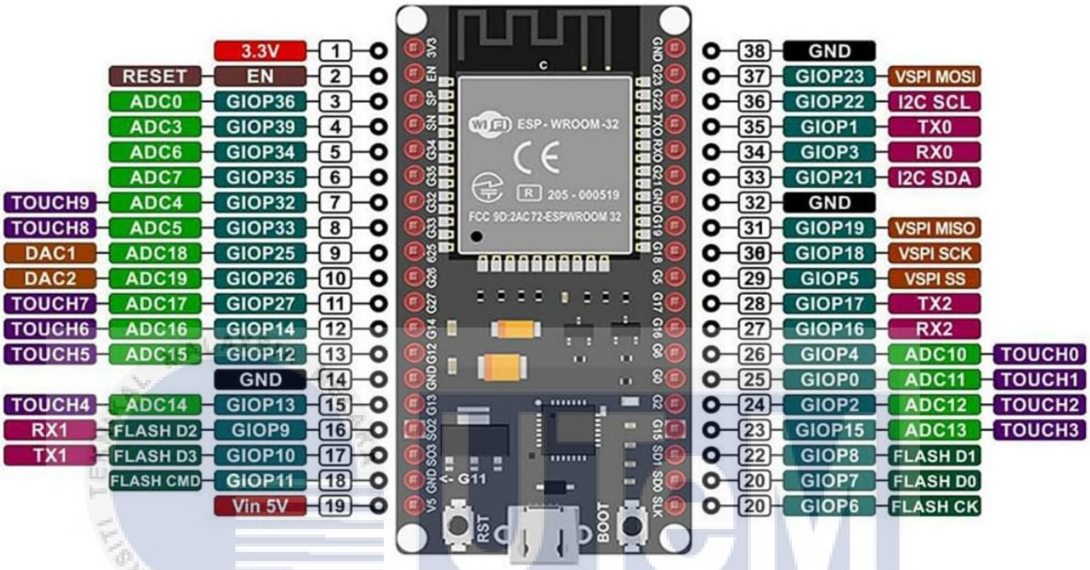


Figure 3.15: ESP32 Wroom 32 pinout[38]

This Table 3.3 organizes the ESP32 different pins and their respective functionalities for easy reference in a project or circuit design.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 3.13: ESP32 Wroom 32 pinout

Pin Name	Description
V5	Dedicated pin for a 5V power input
G13	GPIO pin used for digital input/output
GND	Ground pin providing the circuit's reference voltage
3V3	Supplies a 3.3V output as the power supply pin
G22	GPIO pin used for I2C communication (SCL)
G21	GPIO pin used for I2C communication (SDA)

3.10.2 ESP8266 Pin Declaration

For the ESP8266, the circuit involves utilizing pins VV, G, D0, D1, D2, and 3V.

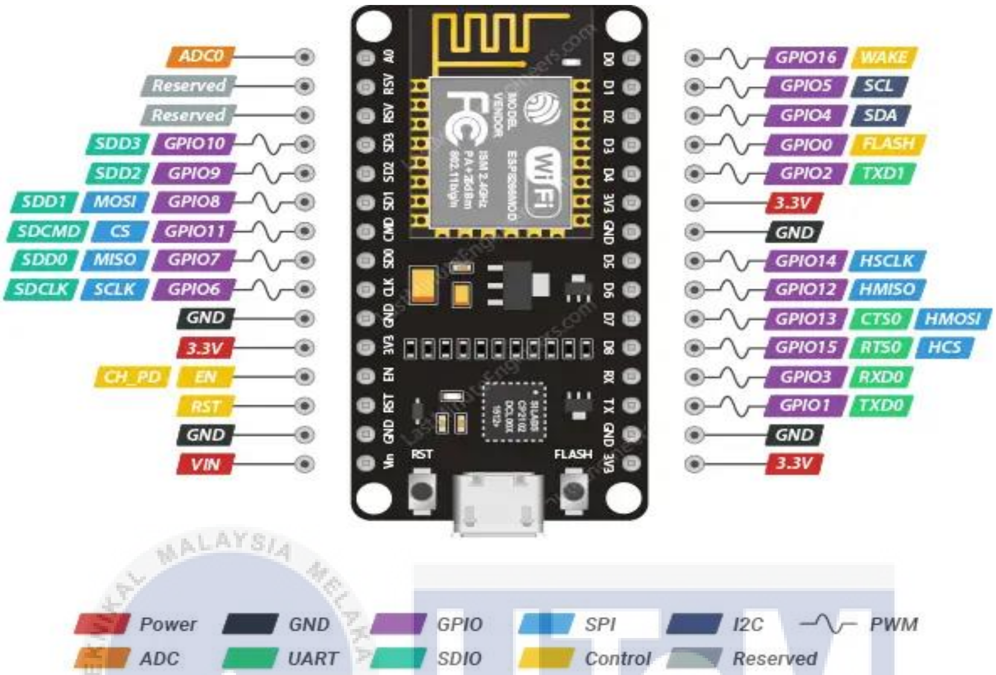


Figure 3.16: ESP8266 pinout[51]

This Table 3.4 highlights the specific pins along with their respective functionalities and characteristics for ease of reference in your projects or circuit designs involving the ESP8266 board.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 3.14: ESP8266 Pinout

Pin Name	Description
VV	The power supply pin
G	Ground pin offering the circuit's reference voltage
D0	GPIO pins used for digital input/output operations
3.3V	Provides a 3.3V output as the power supply pin
D1	GPIO pin used for I2C communication (SCL)
D2	GPIO pin used for I2C communication (SDA)

3.11 Project View

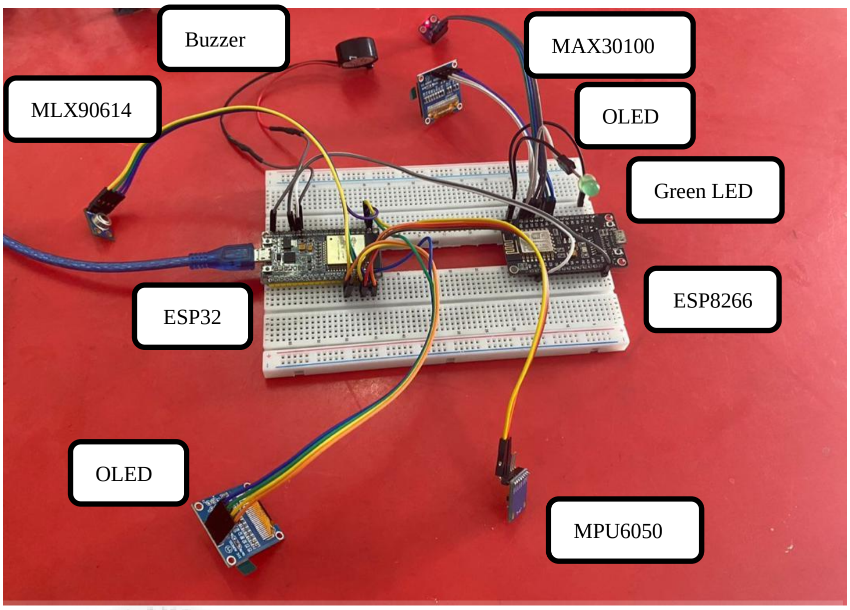


Figure 3.17: Project View Circuit

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.17 shows the circuit connections. The ESP32 and ESP8266 establish their connection by linking their power sources through the VCC and GND connections known as power bridging. Power bridging is a connection or bridging of power supply lines, such as connecting the VCC (power) and GND (ground) lines of different devices or components. In this circuit by linking the 5V output from the ESP32 to the VV input of the ESP8266 and connecting the GND from the ESP32 to the G input of the ESP8266, a power bridge is created between the two microcontrollers. This arrangement allows both ESP devices to read sensors and exchange data concurrently, enabling simultaneous functionality. This parallel operation enables the devices to work together simultaneously, enhancing the overall efficiency and capability of the system. The power supply for both the ESP32 and ESP8266 can be enabled by connecting the 6V rechargeable battery to the 5V input on the ESP32.

The ESP32 is linked to various components: the MLX90614 temperature sensor, the MPU6050 accelerometer and gyroscope, an OLED display, 6V rechargeable battery and a buzzer. This Table 3.5 outlines the connections for each device with the specific ESP32 pins to which they should be connected.

Table 3.15: ESP32 connection with other components

Components	Connection Details
MLX90614	VIN to ESP32's 3V3, GND to ESP32 GND, SCL to ESP32 G22, and SDA to ESP32 G21.
MPU6050	VCC to ESP32 3V3, GND to ESP32 GND, SCL to ESP32 G22, and SDA to ESP32 G21.
OLED	VCC to ESP32 3V3, GND to ESP32 GND, SCL to ESP32 G22, and SDA to ESP32 G21.
Buzzer	Connects to ESP32 G13 and ESP32 GND.
6V rechargeable battery	Connects to ESP32 5V and GND.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

This setup allows the ESP32 to interact with and receive data from these individual components, facilitating the functionality and integration of the temperature sensor, accelerometer, gyroscope, display, and buzzer with the ESP32 microcontroller.

The ESP8266 is interfaced with three components: the MAX30100 pulse oximeter sensor, a green LED, and an OLED display. This Table 3.6 lays out the connections for each device with the specific ESP8266 pins to which they should be connected for proper functionality.

Table 3.16: ESP8266 connection with other components

Components	Connection Details
MAX30100	VIN to ESP8266 3V, GND to ESP8266 G, SCL to ESP8266 D1, and SDA to ESP8266 D2.
Green LED	Connects to ESP8266 D0 and G.
OLED	VCC to ESP8266 3V, GND to ESP8266 G, SCL to ESP8266 D1, and SDA to ESP8266 D2.

This setup enables the ESP8266 to communicate with and gather data from the MAX30100 sensor, control the green LED, and display information on the OLED screen.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.11.1 Interfacing Sensors with ESP32/ESP8266 and Other Components

MLX90614 Sensor Communication with ESP32:

The MLX90614 sensor exclusively utilizes the I2C (Inter-Integrated Circuit) protocol for communication with the ESP32 microcontroller. Operating as an I2C slave, it requires the ESP32 as the master device to initiate communication and retrieve data. The sensor has a fixed I2C address of 0x5A, ensuring proper communication. Analog pins are not involved in sensor readings, with digital pins dedicated exclusively to I2C communication lines (SCL and SDA). SCL carries the clock signal, and SDA carries the data, facilitating bi-directional communication between the ESP32 and MLX90614. The sensor performs serial communication through I2C, avoiding conventional UART or SPI methods. The I2C protocol enables the exchange of data, crucial for retrieving information about the object temperature measured by the sensor.

MPU6050 Sensor Communication with ESP32:

The MPU6050 sensor communicates with the ESP32 using the digital I2C protocol. Connected to the ESP32's GPIO pins configured as SDA and SCL, the sensor's SDA and SCL pins facilitate digital communication. Analog pins are not utilized for this sensor. The MPU6050 combines a gyroscope and accelerometer, and the I2C protocol enables the exchange of data between the sensor and the ESP32. This digital communication method is vital for obtaining accurate measurements of motion and orientation from the MPU6050.

MAX30100 Sensor Communication with ESP8266:

The MAX30100 sensor employs the I2C protocol for digital communication with the ESP8266. The SDA and SCL pins, connected to the ESP8266's GPIO pins, facilitate bi-directional communication. Analog pins are not used for communication with the

MAX30100. As a pulse oximeter and heart-rate sensor, the MAX30100 relies on I2C for transmitting data to the ESP8266. This digital communication method is essential for obtaining pulse and blood oxygen level measurements from the MAX30100 sensor.

OLED Communication with ESP32 and ESP8266:

In I2C communication, the OLED display is connected to the microcontroller using two lines: SDA (Serial Data Line) and SCL (Serial Clock Line). The microcontroller sends commands and data to the OLED display using the I2C protocol to initialize and control the display.

Buzzer Communication with ESP32:

The communication with a buzzer, in the context of an ESP32 microcontroller, typically involves using a digital signal to control the buzzer. Buzzer operation is generally controlled using a digital signal, meaning it's either ON or OFF. The ESP32 has digital GPIO (General Purpose Input/Output) pins that can be used to provide a high (ON) or low (OFF) signal to control the buzzer.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LED Communication with ESP8266:

The communication with an LED in the context of an ESP8266 microcontroller usually involves using a digital signal. LEDs are typically controlled using digital signals, where the microcontroller's GPIO (General Purpose Input/Output) pins provide either a high voltage (ON) or a low voltage (OFF) to control the LED. The ESP8266 has digital GPIO pins that are suitable for controlling LEDs.

3.11.2 Interfacing ESP32/ESP8266 with Blynk: Communication Overview

The interaction between both the ESP32 and ESP8266 microcontrollers and the Blynk platform adheres to a client-server architecture. In this model, the ESP32 or ESP8266 devices operate as clients, establishing connections with the Blynk server using the Blynk library. These microcontrollers transmit data, such as sensor readings or control signals, to the Blynk server. The Blynk server, acting as the central hub, manages multiple connections from various clients, including both ESP32 and ESP8266 devices. It receives data from these devices and facilitates its relay to the Blynk mobile app, providing users with a graphical interface for interaction. In this seamless exchange, the app can send commands to the Blynk server, which, in turn, directs them to the appropriate microcontroller. This client-server architecture simplifies the development of Internet of Things (IoT) projects, enabling remote control and monitoring through the Blynk platform for both ESP32 and ESP8266 devices.
Blynk primarily uses the TCP (Transmission Control Protocol) for communication between the Blynk server and the microcontroller devices (such as ESP32 and ESP8266). TCP provides a reliable and ordered delivery of data packets. It establishes a connection between the client (microcontroller) and the server (Blynk server) before data exchange begins. TCP ensures that data is received without errors and in the correct order. Blynk uses the TCP protocol to maintain a stable and reliable connection between the IoT devices and the Blynk server, ensuring a consistent flow of data for monitoring.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.12 Software Development

3.12.1 ESP32 Code

This code is designed for an IoT-based system using ESP32 and ESP8266 microcontrollers along with various sensors (MLX90614, MPU6050) and an OLED display. It starts by initializing the modules and connecting to Blynk for IoT interactions. Figure 3.18 shows code initialization.

```
#include <Adafruit_MLX90614.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_SSD1306.h>
#include <Wire.h>
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL6JEVBCWO4"
#define BLYNK_TEMPLATE_NAME "PROJECT PSM1"
#define BLYNK_AUTH_TOKEN "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn"
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
BlynkTimer timer;

char auth[] = "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn";
char ssid[] = "HUAWEI nova 3i";
char pass[] = "sggayathiri";

Adafruit_MLX90614 mlx = Adafruit_MLX90614();
Adafruit_MPU6050 mpu;
Adafruit_SSD1306 OLED(128, 32, &Wire);

const int BUZZER_PIN = 13;
bool isBuzzerActive = false; // Declare isBuzzerActive in the global scope
// Define variables to track buzzer states
int buzzerStateTemperature = 0;
int buzzerStateAccelerometer = 0;

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1
```

Figure 3.18: ESP Code Initialization

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The setup() function handles the initialization of sensors, OLED display, and Blynk connection. The loop() function will print patient name and calls two specific functions in sequence - temperatureLoop() and accelerometerGyroscopeLoop().

```
void setup()
{
  Serial.begin(115200);
  while (!Serial);

  pinMode(BUZZER_PIN, OUTPUT);

  if (!mlx.begin())
  {
    Serial.println("Error connecting to MLX sensor. Check wiring.");
    while (1);
  }

  if (!mpu.begin())
  {
    Serial.println("Sensor init failed");
    while (1)
    yield();
  }

  if (!OLED.begin(SSD1306_SWITCHCAPVCC, 0x3C))
  {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;)
    ;
  }

  OLED.display();
  delay(500);
  OLED.setTextSize(1);
  OLED.setTextColor(SSD1306_WHITE);


  OLED.clearDisplay();
  OLED.display();

  Blynk.begin("gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn", ssid,pass);
  Serial.println("");
  delay(100);
}

void loop()
{
  String patientName = "John";
  Serial.print("Patient name: ");
  Serial.println(patientName);
  OLED.clearDisplay();
  OLED.setCursor(0, 0);
  OLED.print("Patient name: ");
  OLED.println(patientName); // Print the patient name on the next line
  OLED.display(); // Update the OLED display
  delay(1000);
  Serial.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
  Blynk.run();
  temperatureLoop();
  accelerometerGyroscopeLoop();
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.19: ESP32 Code Setup() function and Loop() function

The temperatureLoop() function reads body temperature using the MLX90614 sensor, triggers a buzzer based on predefined temperature ranges, updates the OLED display with temperature readings, and sends data to Blynk. Figure 3.20 shows the temperatureLoop () function code. The threshold value used are 35°C and 37.5°C. The threshold values were determined through research and are documented in the included Appendix D. When the temperature between the threshold, it considered as optimal temperature. When the temperature less than 35°C or more than 37.5°C, it considered as low or high temperature.

```
void temperatureLoop()
{
  String tempState = "Optimal"; // Default state is Optimal
  float objectTemperature = 0.0;
  Serial.print("Body temperature = ");
  float objectTempC = mlx.readObjectTempC();
  Serial.print(objectTempC);
  Serial.println("°C");

  Serial.print("Body temperature = ");
  float objectTempF = mlx.readObjectTempF();
  Serial.print(objectTempF);
  Serial.println("°F");

  if (objectTempC >= 35 && objectTempC <= 37.5) {
    noTone(BUZZER_PIN);
    isBuzzerActive = false;
    buzzerStateTemperature = 0;
    Serial.println("Optimal Temperature");
    tempState = "Optimal Temperature";
  }
  else if (objectTempC < 35)  {
    playBuzzerSound();
    buzzerStateTemperature = 1; // Update buzzer state for temperature
    Serial.println("Low Temperature");
    tempState = "Low Temperature";
  }
  else if (objectTempC > 37.5)  {
    playBuzzerSound();
    buzzerStateTemperature = 1; // Update buzzer state for temperature
    Serial.println("High Temperature");
    tempState = "High Temperature";
  }
```

```
  // Print the buzzer state
  Serial.print("Buzzer State (Temperature): ");
  Serial.println(buzzerStateTemperature);

  //blynk notification
  if (objectTempC > 37.5) {
    Blynk.logEvent("high_temperature_alert","High Body Temperature");
  }
  else if (objectTempC < 35) {
    Blynk.logEvent("low_temperature_alert","Low Body Temperature");
  }
  objectTemperature = objectTempC;
  Blynk.virtualWrite(V0, objectTemperature);
  Blynk.virtualWrite(V6, buzzerStateTemperature); // Send buzzer state for temperature to V6

  OLED.clearDisplay();
  // Set the cursor position for temperature
  OLED.setCursor(0, 0);
  OLED.print("Body Temp: ");
  OLED.print(objectTempC, 2);
  OLED.print((char)247);   // Degree symbol
  OLED.print("C");
  OLED.println(" ");
  OLED.print("Body Temp: ");
  OLED.print(objectTempF, 2);
  OLED.print((char)247);   // Degree symbol
  OLED.print("F");
  OLED.println(" ");
  OLED.println(tempState);
  OLED.display();
  Serial.println("------------------------------------------------");
  delay(1000);
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.20: temperatureLoop() function

The accelerometerGyroscopeLoop() function collects accelerometer data from the MPU6050 sensor, identifies gestures based on specific movements, triggers a buzzer accordingly, and updates the OLED display with gesture information. It also sends data to Blynk. Figure 3.21 below shows the accelerometerGyroscopeLoop() function.

```
void accelerometerGyroscopeLoop()
{
  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  float accelerationX = a.acceleration.x;
  float accelerationY = a.acceleration.y;
  OLED.clearDisplay();

  OLED.setCursor(0, 0);

  Serial.print("Accelerometer ");
  Serial.print("X: ");
  Serial.print(a.acceleration.x, 1);
  Serial.print(" m/s^2, ");
  //oled x
  Serial.print("Y: ");
  Serial.print(a.acceleration.y, 1);
  Serial.print(" m/s^2, ");
  //oled y
  Serial.print("Z: ");
  Serial.print(a.acceleration.z, 1);
  Serial.println(" m/s^2");

  String message;
  if (a.acceleration.y > 5)
  {
    message = "Medicine/Emergency";
    Blynk.logEvent("gesture_alert_1","Medicine/Emergency");
  } else if (a.acceleration.y < -5)
  {
    message = "Hungry";
    Blynk.logEvent("gesture_alert_2","Hungry");
  } else if (a.acceleration.x > 5)
  {
    message = "Water";
    Blynk.logEvent("gesture_alert_3","Water");
  } else if (a.acceleration.x < -5)
  {
    message = "Washroom";
    Blynk.logEvent("gesture_alert_4","Washroom");
  } else
  {
    message = "No movement detected";
  }

  Serial.println("Gesture:" + message);

  OLED.clearDisplay();
  OLED.setCursor(0, 0);
  OLED.println("Accelerometer ");
  OLED.print("X: ");
  OLED.print(a.acceleration.x, 1);
  OLED.print("  ");
  OLED.print("Y: ");
  OLED.println(a.acceleration.y, 1);
  OLED.display();
  delay(500);

  OLED.clearDisplay();
  OLED.setCursor(0, 0); // Move cursor to the top-left corner
  OLED.println("Gesture: " + message); // Print Gesture message
  OLED.display();


  if (message != "No movement detected") {
    playBuzzerSound();
    buzzerStateAccelerometer = 1; // Update buzzer state for accelerometer
  } else {
    noTone(BUZZER_PIN);
    isBuzzerActive = false;
    buzzerStateAccelerometer = 0; // Update buzzer state for accelerometer
  }

  // Print the buzzer state
  Serial.print("Buzzer State (Accelerometer): ");
  Serial.println(buzzerStateAccelerometer);

  Serial.print("Gyroscope ");
  Serial.print("X: ");
  Serial.print(g.gyro.x, 1);
  Serial.print(" rps, ");
  Serial.print("Y: ");
  Serial.print(g.gyro.y, 1);
  Serial.print(" rps, ");
  Serial.print("Z: ");
  Serial.print(g.gyro.z, 1);
  Serial.println(" rps");
  Serial.println("************************************************************ ");

  Blynk.virtualWrite(V2,accelerationX);
  Blynk.virtualWrite(V3,accelerationY);
  Blynk.virtualWrite(V7, buzzerStateAccelerometer); // Send buzzer state for accelerometer to V7

  OLED.display();
  delay(1000);
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.21: accelerometerGyroscopeLoop() function

The playBuzzerSound() function generates a sound using the buzzer and manages the buzzer's state. Figure 3.22 shows the playBuzzerSound() function.

```

void playBuzzerSound()
{
  tone(BUZZER_PIN, 1000, 500);
  delay(500);
  noTone(BUZZER_PIN);
  isBuzzerActive = true; // Update the buzzer state variable
}
```

Figure 3.22: playBuzzerSound() function

The code employs various sensor data and interactions to monitor patient vitals and gestures, updating both local displays and sending data to the Blynk IoT platform for remote monitoring and alerts.

3.12.2 ESP8266 Code

The provided code initializes essential components and establishes connections for data acquisition and communication within an ESP8266-based setup. It begins by setting up serial communication and configuring pins, enabling communication with the Serial Monitor and preparing an LED pin (GREEN) for output. The Blynk IoT platform is initialized using predefined authentication credentials (auth, ssid, pass). Additionally, the code attempts to initialize the MAX30100 pulse oximeter sensor (pox) and configures the OLED display (OLED) for subsequent use, setting its text size, color, and clearing the display.

Within the loop function, the code ensures continuous operations. It executes routines to facilitate Blynk communication (Blynk.run()) and updates data from the pulse oximeter sensor (pox.update()). It periodically reports data, displaying heart rate (rate) and blood oxygen levels (sp) on the Serial Monitor and OLED display when the elapsed time since the last report surpasses a set duration (REPORTING_PERIOD_MS).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Additionally, it manages the green LED state based on heart rate conditions: illuminating for optimal heart rates (50-100 bpm) and turning off for rates below 50 or above 100 bpm. Therefore, it can be inferred that the threshold values for high and low heart rates are set at 100 bpm (upper limit) and 50 bpm (lower limit), respectively. If the heart rate goes below 50 bpm or above 100 bpm, the LED state would be adjusted accordingly. The threshold values were determined through research and are documented in the included Appendix C. The code updates Blynk widgets with heart rate and blood oxygen level data (V1 and V5, respectively), and logs events on the Blynk platform for high or low BPM alerts based on predefined thresholds. This loop ensures continuous data retrieval, display, LED control, and remote communication, crucial for real-time monitoring and alerting in this IoT setup. Figure 2.23 below shows the ESP8266 code.

```
#include <Wire.h>
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL6JEVBCWO4"
#define BLYNK_TEMPLATE_NAME "PROJECT PSM1"
#define BLYNK_AUTH_TOKEN "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn"
#include <SPI.h>
#include <Ethernet.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "MAX30100_PulseOximeter.h"
#include <Adafruit_SSD1306.h>
BlynkTimer timer;
Adafruit_SSD1306 OLED(128, 32, &Wire);
#define REPORTING_PERIOD_MS 1000
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1
char auth[] = "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn";
char ssid[] = "HUAWEI nova 3i";
char pass[] = "sggayathiri";
const int GREEN = 16;
PulseOximeter pox;
uint32_t tsLastReport = 0;
void onBeatDetected()
{
Serial.println("Beat!");
}
void setup()
{
Serial.begin(9600);
pinMode(GREEN, OUTPUT);
Blynk.begin(auth, ssid, pass);
Serial.print("Initializing pulse oximeter..");
if (!pox.begin())
{
Serial.println("FAILED");
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
for (;;)
;
}
else
{
Serial.println("SUCCESS");
}
pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
pox.setOnBeatDetectedCallback(onBeatDetected);
if (!OLED.begin(SSD1306_SWITCHCAPVCC, 0x3C))
{
Serial.println(F("SSD1306 allocation failed"));
for (;;)
;
}
OLED.display();
OLED.setTextSize(1);
OLED.setTextColor(SSD1306_WHITE);
OLED.clearDisplay();
OLED.display();
}
void loop()
{
Blynk.run();
pox.update();
if (millis() - tsLastReport > REPORTING_PERIOD_MS)
{
Serial.print("Heart rate:");
Serial.print(pox.getHeartRate());
Serial.print("bpm / SpO2:");
Serial.print(pox.getSpO2());
Serial.println("%");
OLED.clearDisplay();
OLED.setCursor(0, 0);
OLED.print("Heart rate:");
OLED.print(pox.getHeartRate());
```

```
OLED.print("bpm / SpO2:");
OLED.print(pox.getSpO2());
OLED.println("%");
OLED.display();
int rate = pox.getHeartRate();
int sp = pox.getSpO2();
tsLastReport = millis();
if (rate >= 50 && rate <= 100 )
{
digitalWrite(GREEN, HIGH);
Serial.println("Optimal heart rate");
OLED.println("Optimal heart rate");
OLED.display();
Serial.println("LED STATUS: 1");
}
else if (rate <50)
{
digitalWrite(GREEN, LOW);
Serial.println("Low heart rate");
OLED.println("Low heart rate");
OLED.display();
Serial.println("LED STATUS: 0");
}
else if (rate >100)
{
digitalWrite(GREEN, LOW);
Serial.println("High heart rate");
OLED.println("High heart rate");
OLED.display();
Serial.println("LED STATUS: 0");
}
Blynk.virtualWrite(V1, rate);
Blynk.virtualWrite(V5, sp);

if (rate > 100)
{
```

```
Blynk.logEvent("high_bpm_alert", "High BPM");
}
else if (rate <50)
{
Blynk.logEvent("low_bpm_alert", "Low BPM");
}
}
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.23: ESP8266 Code

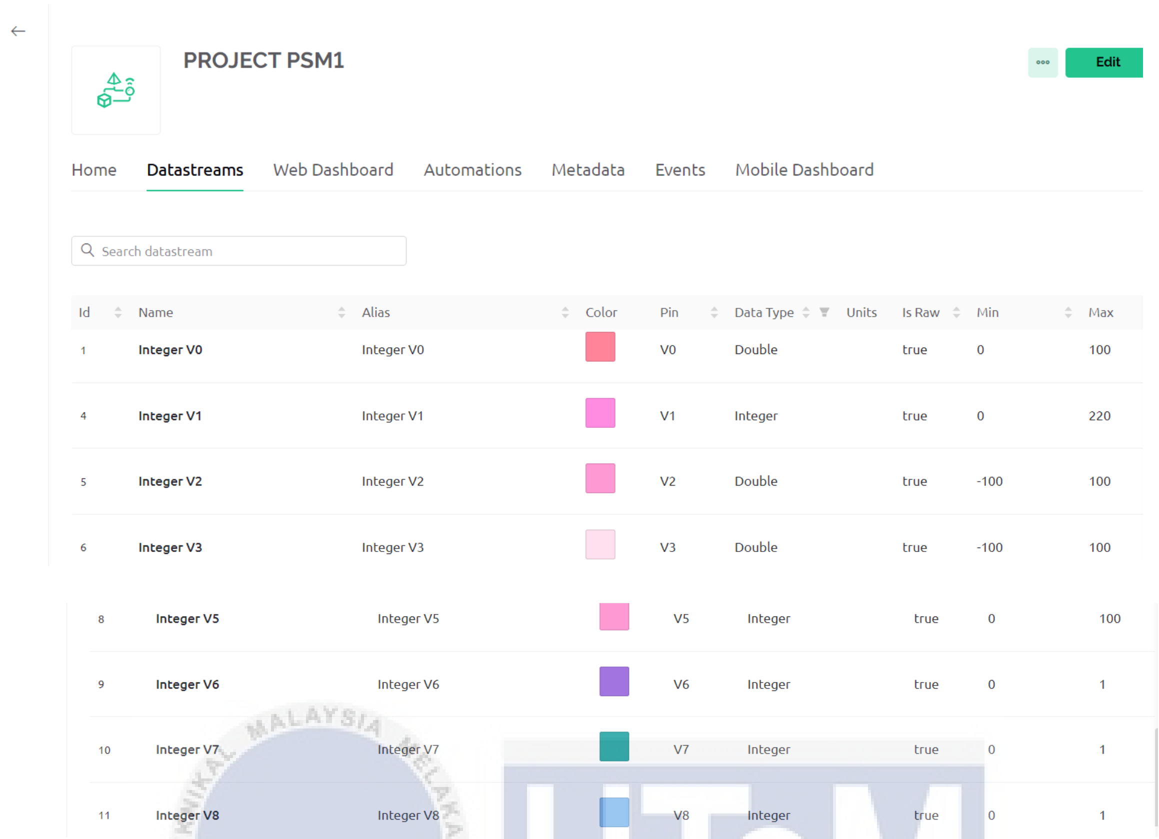
3.13 Mobile App Integration

Mobile App Widgets:

When designing mobile applications, integrating various widgets becomes crucial to present information effectively. Text Widgets serve to relay critical information like sensor readings or alerts, ensuring concise communication. Gauge Widgets offer a visual representation of data, ideal for showcasing metrics such as temperature or heart rate. Label Widgets play a vital role in annotating or describing the displayed information. Superchart Widgets provide a comprehensive view of historical data trends, aiding in analyzing patterns over time. Level V Widgets, on the other hand, cater to discrete levels or states, facilitating the display of gesture alerts or binary statuses within the application's interface. These diverse widgets collectively enhance user experience by presenting data in meaningful and accessible formats.

Datastreams and Device Integration:

In this integration setup, the widgets are directly linked to designated datastreams (V1-V8), acting as the visual interpreters of information received from both ESP32 and ESP8266 devices. This connection allows for real-time visualization and monitoring of the data collected from these devices. For instance, temperature readings, heart rate data, and gesture information can be displayed in gauge widgets for immediate visualization, offering a quick glance at these metrics. Simultaneously, a superchart can track these parameters over time, providing a historical perspective on trends and patterns. Moreover, these parameters can be configured to trigger notifications when they surpass predefined thresholds or specific conditions, adding proactive monitoring capabilities and ensuring timely alerts for efficient management and response to critical situations. Figure 3.24 shows the V1-V8 datastreams.



PROJECT PSM1

Home | Datastreams | Web Dashboard | Automations | Metadata | Events | Mobile Dashboard

Id	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min	Max
1	Integer V0	Integer V0		V0	Double		true	0	100
4	Integer V1	Integer V1		V1	Integer		true	0	220
5	Integer V2	Integer V2		V2	Double		true	-100	100
6	Integer V3	Integer V3		V3	Double		true	-100	100
8	Integer V5	Integer V5		V5	Integer		true	0	100
9	Integer V6	Integer V6		V6	Integer		true	0	1
10	Integer V7	Integer V7		V7	Integer		true	0	1
11	Integer V8	Integer V8		V8	Integer		true	0	1

Figure 3.24: V1-V8 Datastreams

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Web Dashboard and Blynk Console:

The integration of a web dashboard and the Blynk Console provides a versatile framework for arranging and managing widgets along with data sources, enabling tailored user experiences. Within these platforms, users can establish specific conditions or thresholds for parameters such as temperature, BPM (heart rate), and gesture alerts. This customization enables the system to generate notifications or alerts when sensor readings exceed predefined limits. Leveraging this functionality, the platforms seamlessly facilitate the delivery of immediate alerts to the user via the associated mobile app, ensuring timely awareness and response to critical changes or events detected by the connected devices.

3.14 Summary

In the hardware development section, the ESP32 and ESP8266 microcontrollers are detailed, outlining their pin configurations and connections with various components. The ESP32 interacts with sensors like the MLX90614 and MPU6050, an OLED display, and a buzzer, while the ESP8266 interfaces with the MAX30100 sensor, a green LED, a 6V battery, and an OLED display. By employing power bridging, both microcontrollers enable simultaneous functionality, facilitating data exchange and sensor readings.

ESP32 and ESP8266 codes are tailored for IoT functionality, sensor data acquisition, and Blynk platform communication. They initialize modules, continuously collect sensor data, update displays, and communicate with Blynk for remote monitoring. The ESP32 code reads temperature and gestures, triggers buzzers, and sends data to Blynk, while the ESP8266 monitors heart rate, controls LEDs, and communicates sensor data to Blynk. Specific functions manage sensor data, display updates, and buzzer control, pivotal for real-time monitoring.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The integration incorporates diverse widgets (Text, Gauge, Label, Superchart, and Level V) to effectively present real-time data from the microcontrollers. Datastreams (V1-V8) interpret sensor readings, showcasing real-time and historical data trends, along with alerts based on predefined thresholds. Web dashboards and the Blynk Console allow user-specific configurations for parameters and notifications, ensuring timely alerts and efficient management in response to critical sensor data.

The integrated system presents an advanced IoT setup leveraging ESP32 and ESP8266 microcontrollers, enabling real-time monitoring, data visualization, and alerting mechanisms. The synergy between these components paves the way for robust systems applicable in healthcare, remote monitoring, and IoT solutions.

The proposed methodology introduces a glove-based healthcare system for paralysis patients. It aims for simplicity, efficiency, and effectiveness catering to diplegia, hemiplegia, and monoplegia individuals in home settings.

This methodology enhances healthcare access by minimizing hospital visits, improves monitoring and management of patients' health conditions, and fosters mental peace for patients and caregivers. Offering continuous support and supervision ensures safety and peace of mind for patients and their families.

In summary, this methodology seeks to address paralysis patients' unique challenges by providing a straightforward yet effective healthcare system. It aims to enhance healthcare access, improve medical care, and offer mental peace and continuous support to individuals with paralysis.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

The conducted research and experimentation yielded comprehensive results in the successful implementation of the glove-based paralysis patient healthcare system. This chapter encapsulates the conclusive outcomes obtained through the integration and testing of hardware and software components proposed for this system. The ESP32 and ESP8266 microcontrollers, in conjunction with various sensors including the pulse oximeter sensor MAX30100, temperature sensor MLX90614, gyroscope and accelerometer sensor MPU6050, showcased seamless integration within the glove-based system. These components functioned cohesively to capture and transmit relevant data accurately. The accelerometers embedded within the gloves effectively detected directional changes, correlating with significant variations in measured values. The analysis of accelerometer data facilitated the triggering of pre-coded messages displayed on the OLED screen and simultaneous notifications sent to mobile phone using Blynk application. These notifications indicated situations requiring immediate attention, enhancing the system's responsiveness. The software interfacing with the hardware components demonstrated robust functionality. Sensor data encompassing body temperature and heart rate readings were efficiently transmitted over the Internet to inform designated caregivers. This communication protocol was established through a Mobile phone acting as the receiving device. The dedicated Blynk application displayed critical information, encompassing the patient's body temperature, BPM, and SpO2 corresponding to the recorded readings and will be stored into Blynk cloud.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2 Expected Result

Upon detection using the MLX90614 temperature sensor, MAX30100 pulse oximeter sensor, and MPU6050 accelerometer sensor, the OLED screen will sequentially display the patient's name, body temperature, gesture message, BPM, and SpO2 values, allowing convenient real-time vital sign monitoring. Furthermore, the collected sensor data, encompassing body temperature, BPM/SpO2, and gesture messages, will be transmitted over the Internet to notify caregivers. A mobile phone equipped with the Blynk application will serve as the receiver, displaying relevant information, sending notifications, and storing data in the Blynk database for easy access.

4.2.1 Expected result for MLX90614

Body temperature will be display in Celcius (°C) and Farenheit(°F). Table 4.1 below shows the expected result for MLX90614.

Table 4.1: Expected result for MLX90614

<table>
<tr><th>Condition</th><th>Buzzer Status</th></tr>
<tr><td>if (objectTempC > 37.5 || objectTempC < 35)</td><td>The buzzer will turn on.
The buzzer state will be 1.
A notification will be send to mobile phone as High or Low Body Temperature.</td></tr>
<tr><td>else</td><td>The buzzer will turn off.
The buzzer state will be 0.</td></tr>
</table>

4.2.2 Expected result for MAX30100

BPM/SpO2 will be displayed. The Table 4.2 below shows the expected result using MAX30100.

Table 4.2: Expected result for MAX30100

Condition	LED Status
if (pox.getHeartRate() >= 50 && < =100)	A message "Optimal heart rate" will be display. The LED will turn on when a pulse is detected. The LED state will be 1.
else if (pox.getHeartRate() < 50)	A message "Low heart rate" will be displayed. The LED will turn off when no pulse is low. The LED state will be 0. A notification will be send to mobile phone as Low BPM.
else if (pox.getHeartRate() > 100)	A message "High heart rate" will be displayed. The LED will turn off when no pulse is high. The LED state will be 0. A notification will be send to mobile phone as High BPM.

4.2.3 Expected result for MPU6050

Accelerometer value and gyroscope value for x,y,z axis will be displayed. A gesture message will be displayed based on condition. Table 4.3 below shows the condition for each hand gesture detection.

Table 4.3: Hand Gesture Condition

Condition	Message
if (a.acceleration.y > 5)	message = "Medicine/Emergency"
else if (a.acceleration.y < -5)	message = "Hungry"
else if (a.acceleration.x > 5)	message = "Water";
else if (a.acceleration.x < -5)	message = "Washroom"
Else	message = "No movement detected"

Table 4.4 below shows the expected result using MPU6050.

Table 4.4: Expected result for MPU6050

Condition	Buzzer Status
if (message != "No movement detected")	The buzzer will turn on. The buzzer state will be 1. A notification will be send to mobile phone as gesture detected.
else	The buzzer will turn off. The buzzer state will be 0.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2.4 Expected result for OLED

The OLED display will showcase the patient's name along with their body temperature in both Celsius (°C) and Fahrenheit (°F). Additionally, it will present the accelerometer's x-axis and y-axis values along with the corresponding gesture message, and also display the BPM/SpO2 message.

4.3 Project Analysis

4.3.1 MLX90614 Temperature Sensor

The room temperature typically ranges between 28 to 30 degrees Celsius (82 to 86 degrees Fahrenheit). Within this temperature range, the ceiling fan is set to run at a medium or low speed. The below figure shows serial monitor result for temperature sensor at room temperature. The recorded temperatures of 29.59 and 29.57 fall within the range of 28 to 30 degrees.

```
16:53:33.150 -> Patient name: John
16:53:34.138 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
16:53:34.138 -> Body temperature = 29.59°C
16:53:34.138 -> Body temperature = 85.26°F
16:53:34.651 -> Low Temperature
16:53:34.651 -> Buzzer State (Temperature): 1
16:53:34.792 -> ---------------------------------------------------------
16:53:35.808 -> Accelerometer X: -2.9 m/s^2, Y: 0.4 m/s^2, Z: -7.8 m/s^2
16:53:35.808 -> Gesture:No movement detected
16:53:36.316 -> Buzzer State (Accelerometer): 0
16:53:36.316 -> Gyroscope X: -0.0 rps, Y: -0.0 rps, Z: -0.0 rps
16:53:36.316 -> *********************************************************
16:53:37.476 -> Patient name: John
16:53:38.518 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
16:53:38.518 -> Body temperature = 29.59°C
16:53:38.555 -> Body temperature = 85.26°F
16:53:39.018 -> Low Temperature
16:53:39.018 -> Buzzer State (Temperature): 1
16:53:39.163 -> ---------------------------------------------------------
16:53:40.140 -> Accelerometer X: -2.9 m/s^2, Y: 0.3 m/s^2, Z: -7.8 m/s^2
16:53:40.140 -> Gesture:No movement detected
16:53:40.664 -> Buzzer State (Accelerometer): 0
16:53:40.664 -> Gyroscope X: -0.0 rps, Y: -0.0 rps, Z: 0.0 rps
16:53:40.701 -> *********************************************************
16:53:41.829 -> Patient name: John
16:53:42.846 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
16:53:42.846 -> Body temperature = 29.57°C
16:53:42.846 -> Body temperature = 85.23°F
16:53:43.337 -> Low Temperature
16:53:43.337 -> Buzzer State (Temperature): 1
16:53:43.507 -> ---------------------------------------------------------
```

Figure 4.1: Result for temperature sensor at room temperature

The Table 4.5 shows fluctuating body temperatures in both Celsius and Fahrenheit, along with corresponding buzzer states. The buzzer state appears to switch between 1 and 0 based on certain conditions, possibly related to temperature thresholds, possibly indicating a buzzer to alert for certain temperature ranges. When the temperature falls below 29.97 or rises above 38.09, the buzzer is activated and set to 1. Conversely, when the temperature reaches the optimal value of 35.89, the buzzer is deactivated and set to 0. These temperature-based conditions serve as triggers for toggling the buzzer state between on and off states, potentially indicating temperature-related alerts.

Table 4.5: Body Temperature Sample Readings

Figure	Body Temperature in °C	Body Temperature in °F	Buzzer State
Figure 4.2	29.97	85.95	1
Figure 4.3	34.71	94.48	1
Figure 4.4	34.77	94.59	1
Figure 4.5	38.01	100.42	1
Figure 4.6	36.65	97.97	0
Figure 4.7	35.91	96.64	0
Figure 4.8	33.19	91.74	1
Figure 4.9	38.37	101.07	1
Figure 4.10	37.91	100.24	1
Figure 4.11	38.09	100.56	1
Figure 4.12	37.29	99.12	0
Figure 4.13	30.31	86.56	1
Figure 4.14	35.89	96.60	0
Figure 4.15	35.99	96.78	0
Figure 4.16	36.89	98.40	0
Figure 4.17	36.79	98.22	0
Figure 4.18	29.93	85.87	1

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The provided information comprises figures illustrating the output from the serial monitor and its correlated display on the Blynk application.

Figure 4.2: Body Temperature Sample 1

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

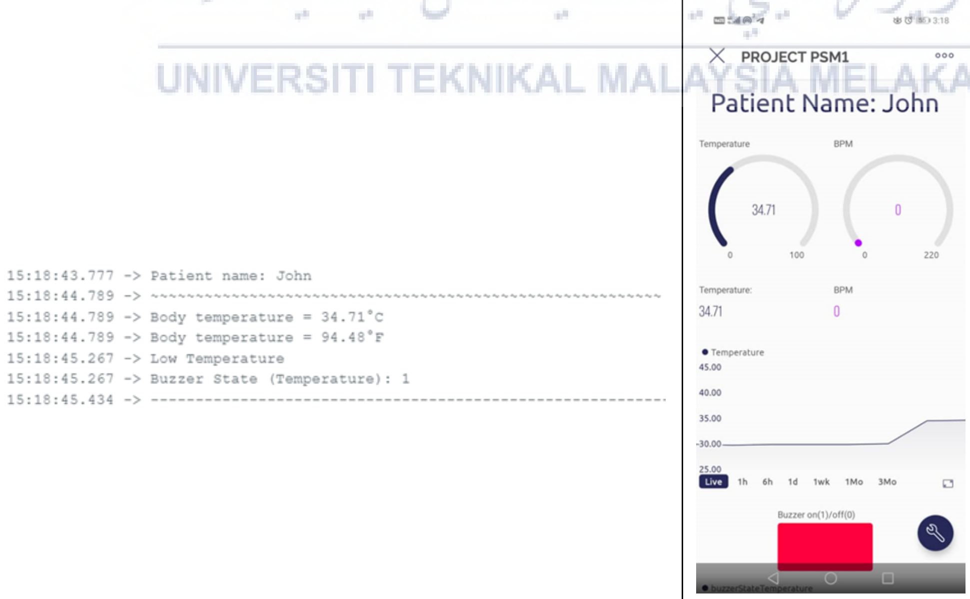


Figure 4.3: Body Temperature Sample 2

```
15:18:52.466 -> Patient name: John
15:18:53.485 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:18:53.485 -> Body temperature = 34.77°C
15:18:53.485 -> Body temperature = 94.59°F
15:18:53.973 -> Low Temperature
15:18:53.973 -> Buzzer State (Temperature): 1
15:18:54.130 -> ------------------------------------------------------------
```

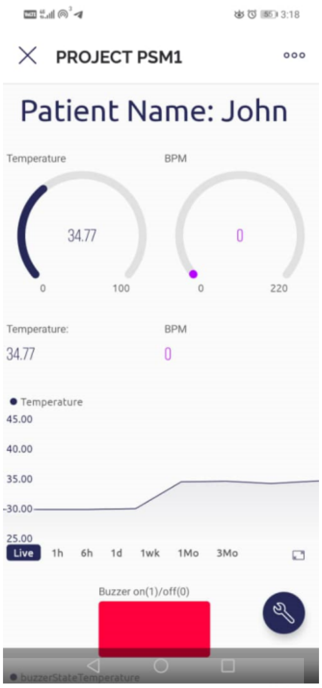


Figure 4.4: Body Temperature Sample 3

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
15:19:09.872 -> Patient name: John
15:19:10.883 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:19:10.883 -> Body temperature = 38.01°C
15:19:10.883 -> Body temperature = 100.42°F
15:19:11.405 -> High Temperature
15:19:11.405 -> Buzzer State (Temperature): 1
15:19:11.554 -> ------------------------------------------------------------
```

Figure 4.5: Body Temperature Sample 4

Figure 4.6: Body Temperature Sample 5

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

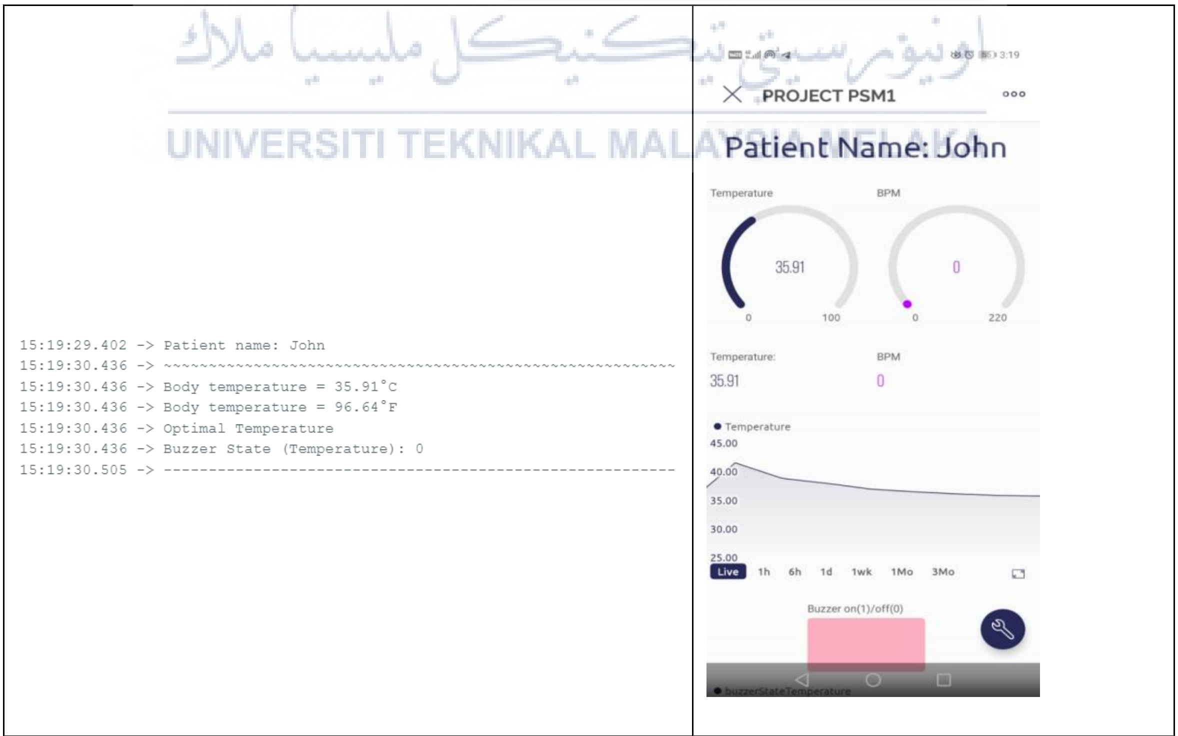


Figure 4.7: Body Temperature Sample 6

```
15:19:48.400 -> Patient name: John
15:19:49.393 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:19:49.393 -> Body temperature = 33.19°C
15:19:49.393 -> Body temperature = 91.74°F
15:19:49.909 -> Low Temperature
15:19:49.909 -> Buzzer State (Temperature): 1
15:19:50.048 -> ----------------------------------------------------------------
```

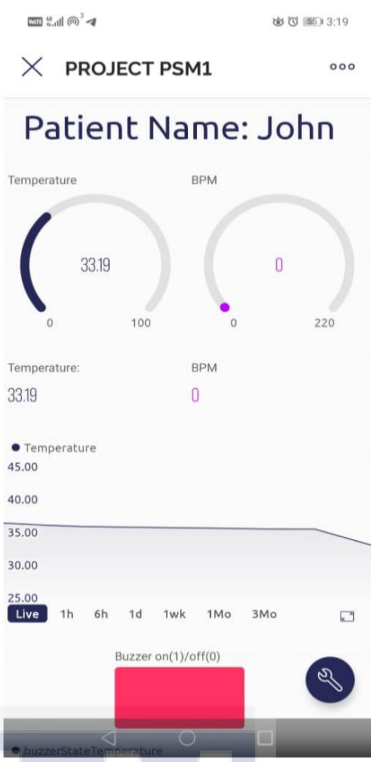


Figure 4.8: Body Temperature Sample 7

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
15:20:36.299 -> Patient name: John
15:20:37.302 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:20:37.302 -> Body temperature = 38.37°C
15:20:37.302 -> Body temperature = 101.07°F
15:20:37.799 -> High Temperature
15:20:37.799 -> Buzzer State (Temperature): 1
15:20:37.942 -> ----------------------------------------------------------------
```

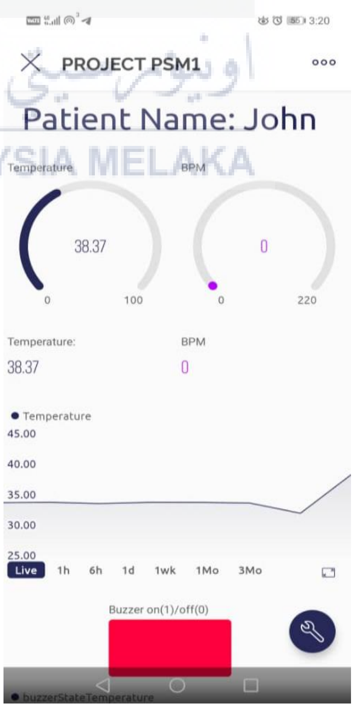


Figure 4.9: Body Temperature Sample 8

```
15:20:53.147 -> Patient name: John
15:20:54.160 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:20:54.160 -> Body temperature = 37.91°C
15:20:54.160 -> Body temperature = 100.24°F
15:20:54.646 -> High Temperature
15:20:54.646 -> Buzzer State (Temperature): 1
15:20:54.786 -> ----------------------------------------------------------------
```

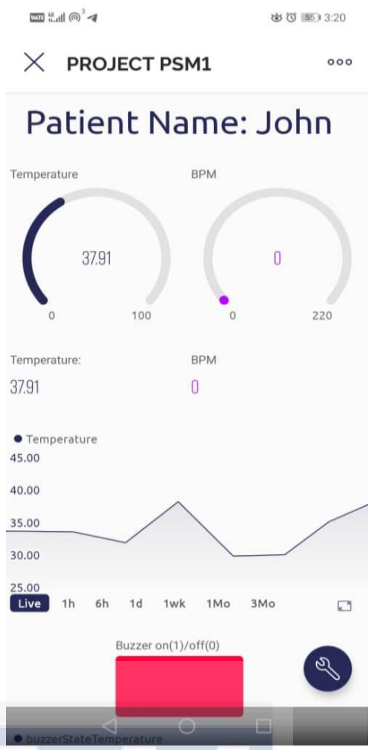


Figure 4.10: Body Temperature Sample 9

```
15:20:57.481 -> Patient name: John
15:20:58.477 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:20:58.510 -> Body temperature = 38.09°C
15:20:58.510 -> Body temperature = 100.56°F
15:20:59.000 -> High Temperature
15:20:59.000 -> Buzzer State (Temperature): 1
15:20:59.131 -> ----------------------------------------------------------------
```

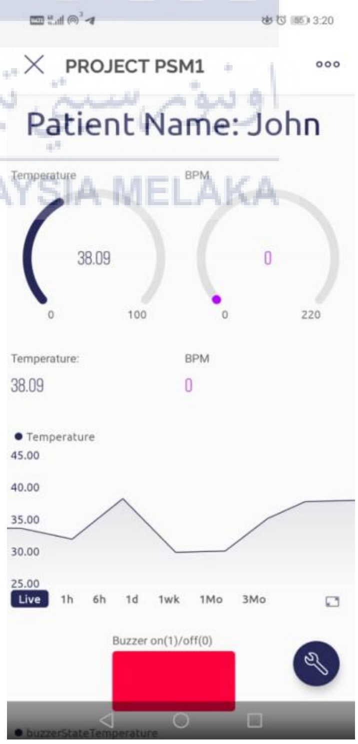


Figure 4.11: Body Temperature Sample 10

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
15:21:01.836 -> Patient name: John
15:21:02.857 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:21:02.857 -> Body temperature = 37.29°C
15:21:02.857 -> Body temperature = 99.12°F
15:21:02.857 -> Optimal Temperature
15:21:02.857 -> Buzzer State (Temperature): 0
15:21:02.955 -> -----------------------------------------------------------
```

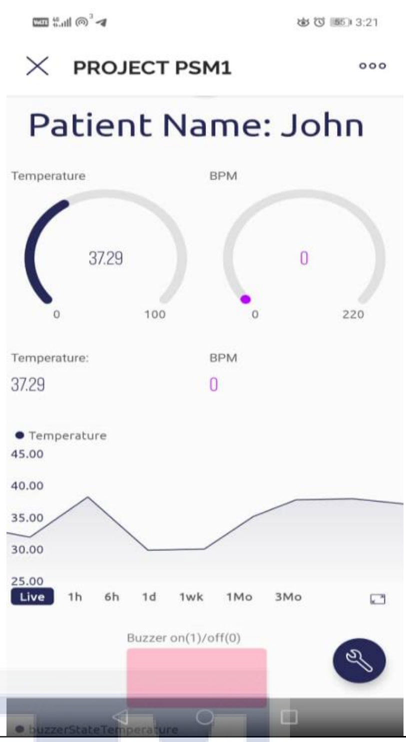


Figure 4.12: Body Temperature Sample 11

```
15:21:26.839 -> Patient name: John
15:21:27.877 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:21:27.877 -> Body temperature = 30.31°C
15:21:27.877 -> Body temperature = 86.56°F
15:21:28.345 -> Low Temperature
15:21:28.345 -> Buzzer State (Temperature): 1
15:21:28.497 -> -----------------------------------------------------------
```

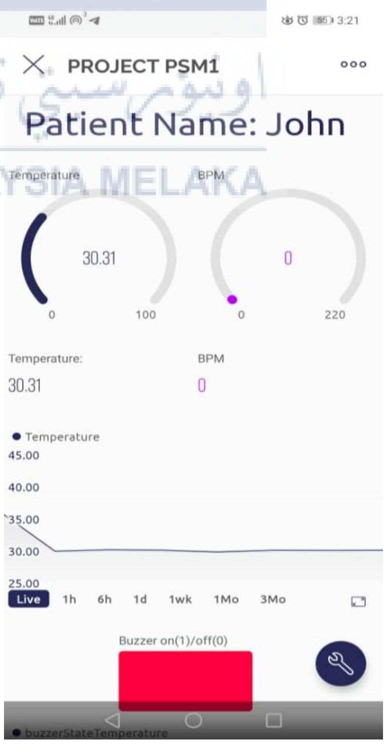


Figure 4.13: Body Temperature Sample 12

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

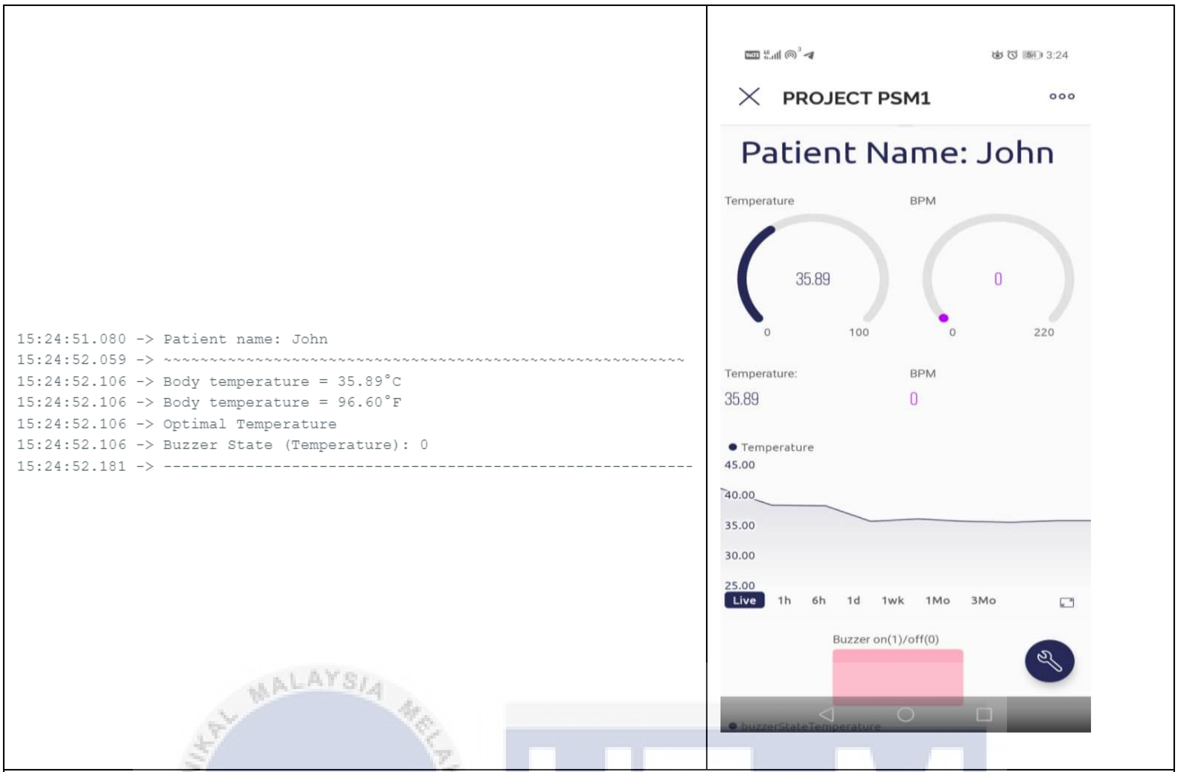


Figure 4.14: Body Temperature Sample 13

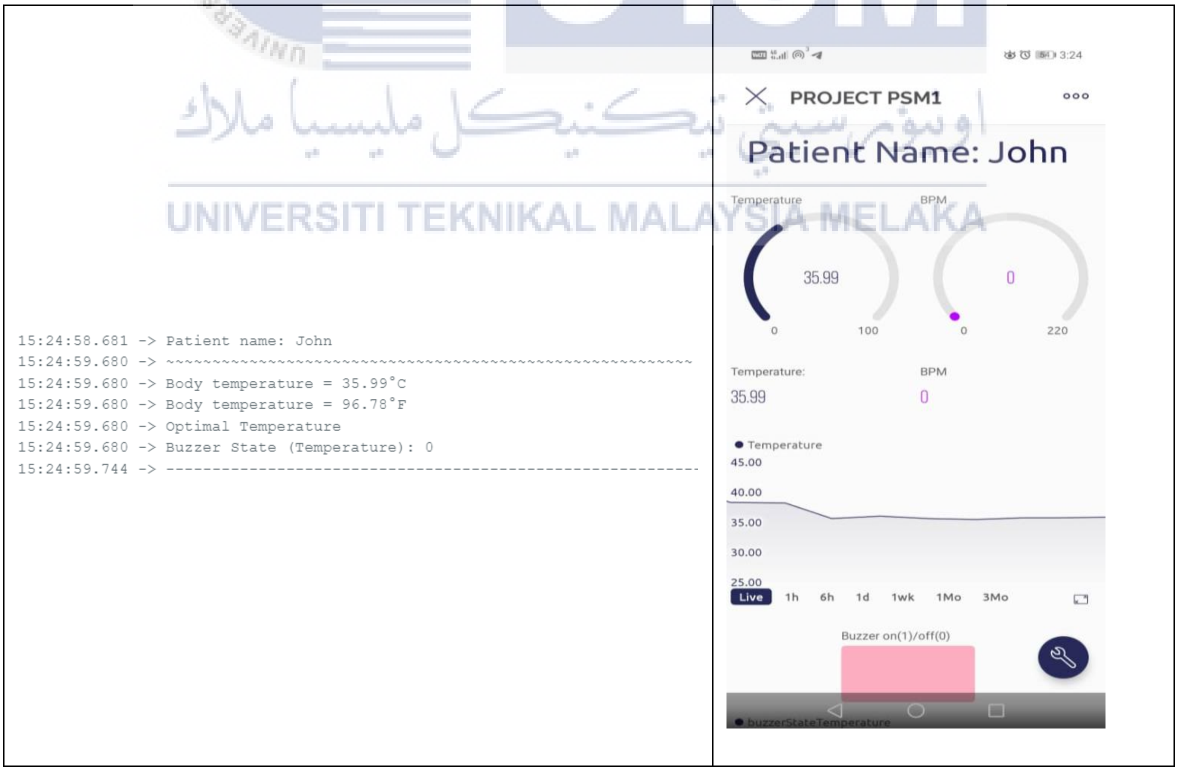


Figure 4.15: Body Temperature Sample 14

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
15:25:40.413 -> Patient name: John
15:25:41.440 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:25:41.440 -> Body temperature = 36.89°C
15:25:41.440 -> Body temperature = 98.40°F
15:25:41.440 -> Optimal Temperature
15:25:41.440 -> Buzzer State (Temperature): 0
15:25:41.517 -> ----------------------------------------------------------------
```

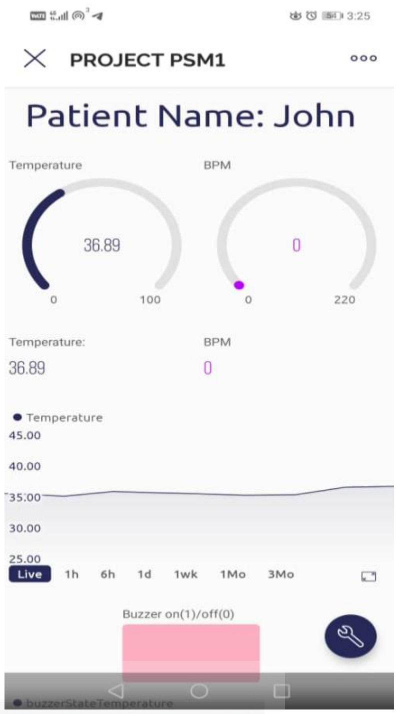


Figure 4.16: Body Temperature Sample 15

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
15:25:44.216 -> Patient name: John
15:25:45.210 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
15:25:45.211 -> Body temperature = 36.79°C
15:25:45.248 -> Body temperature = 98.22°F
15:25:45.248 -> Optimal Temperature
15:25:45.248 -> Buzzer State (Temperature): 0
15:25:45.329 -> ----------------------------------------------------------------
```

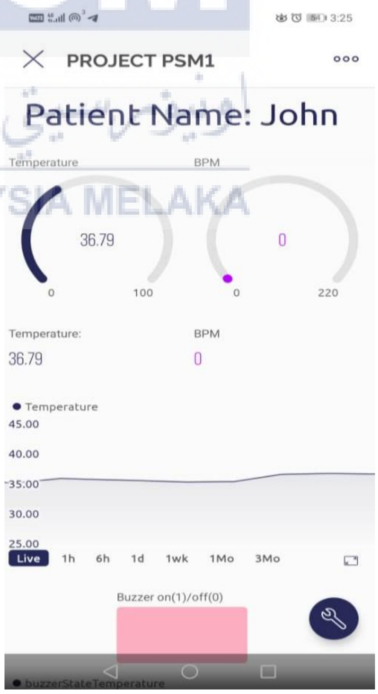
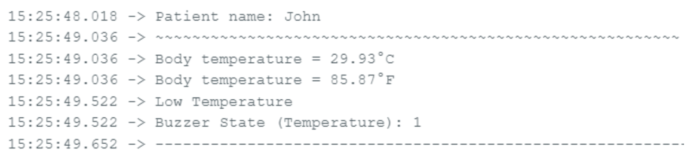


Figure 4.17: Body Temperature Sample 16



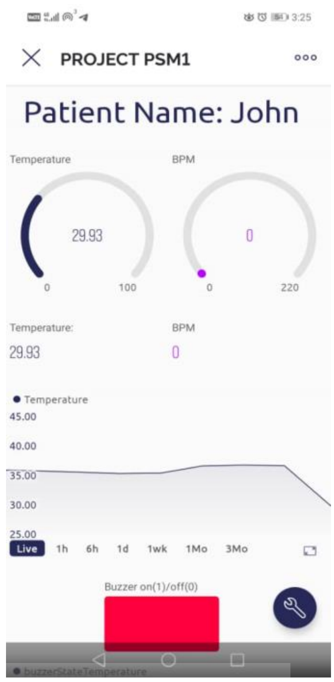


Figure 4.18: Body Temperature Sample 17

4.3.1.1 Blynk Notification for MLX90614 Sensor

Table 4.6 shows instances of temperature-related events triggering notifications on a Blynk application. At various timestamps on December 31st, 2023, specific temperature readings prompted warnings. For instance, temperatures of 29.93, 29.99, and 33.19 initiated "Low_Temperature_Alert" notifications, indicating low body temperature conditions. Conversely, temperatures of 41.31, 38.37, and 41.71 led to "High_Temperature_Alert" notifications, signaling elevated body temperature levels. These notifications serve to highlight fluctuations in body temperature, raising awareness of potential health concerns based on predefined thresholds for high and low temperatures.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.6: Blynk Notification for MLX90614

Time	Temperature	Event Type	Name	Description
12/31/23 03:25:49 PM	29.93	WARNING	Low_Temperature_Alert	Low Body Temperature
12/31/23 03:24:24 PM	41.31	WARNING	High_Temperature_Alert	High Body Temperature
12/31/23 03:24:15 PM	29.99	WARNING	Low_Temperature_Alert	Low Body Temperature
12/31/23 03:21:11 PM	30.17	WARNING	Low_Temperature_Alert	Low Body Temperature
12/31/23 03:20:37 PM	38.37	WARNING	High_Temperature_Alert	High Body Temperature
12/31/23 03:19:50 PM	33.19	WARNING	Low_Temperature_Alert	Low Body Temperature
12/31/23 03:19:02 PM	41.71	WARNING	High_Temperature_Alert	High Body Temperature
12/31/23 03:18:19 PM	29.83	WARNING	Low_Temperature_Alert	Low Body Temperature

The data below illustrates notifications as they are presented within the Blynk application, highlighting occurrences of "Low_Temperature_Alert" or "High_Temperature_Alert" alongside their corresponding dates and times in Figure 4.19.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

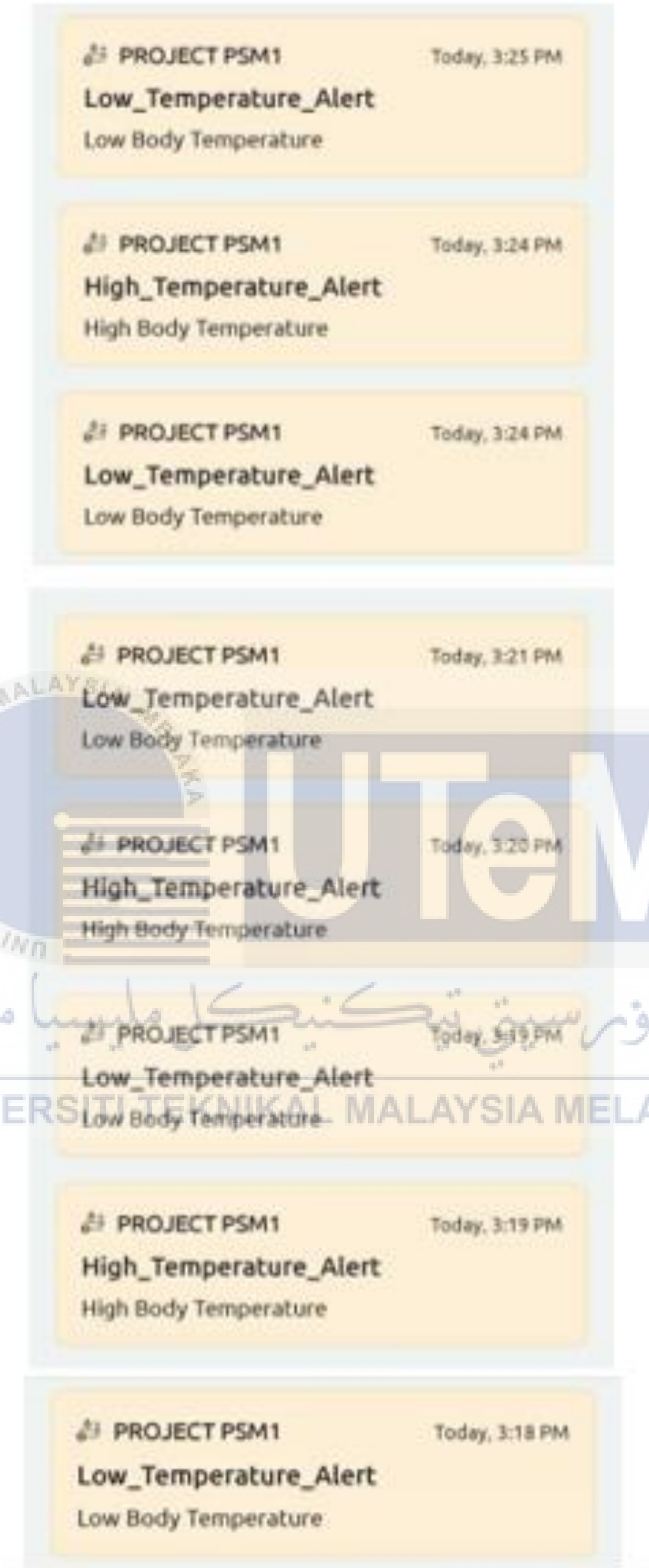


Figure 4.19: Sample MLX90614 Notification

UTeM
اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

This graph in Figure 4.20 illustrates the correlation between Temperature and the corresponding Buzzer State over time, providing a comprehensive overview of their relationship. The Buzzer State is represented by binary values, 0 or 1, based on detected temperature levels. When the temperature ranges between 50 and 57.5 (Point 2), the Buzzer State remains at 0. However, when the temperature exceeds 57.5 (Point 3) or falls below 50 (Point 1), the Buzzer State switches to 1, indicating activation or deactivation of the buzzer. In the graph, the blue points denote the recorded temperature values, while the orange points serve as visual markers indicating moments when the buzzer is triggered or deactivated based on temperature changes. Each temperature point corresponds to a distinct line indicating the state of the buzzer on or off.

This visual representation facilitates quick identification of temperature shifts influencing the buzzer's behavior. By observing this graph, patterns emerge, highlighting instances when temperature aligns with the predefined conditions, triggering the activation or deactivation of the buzzer. Such a clear visualization streamlines monitoring and allows for prompt interventions based on temperature variations.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

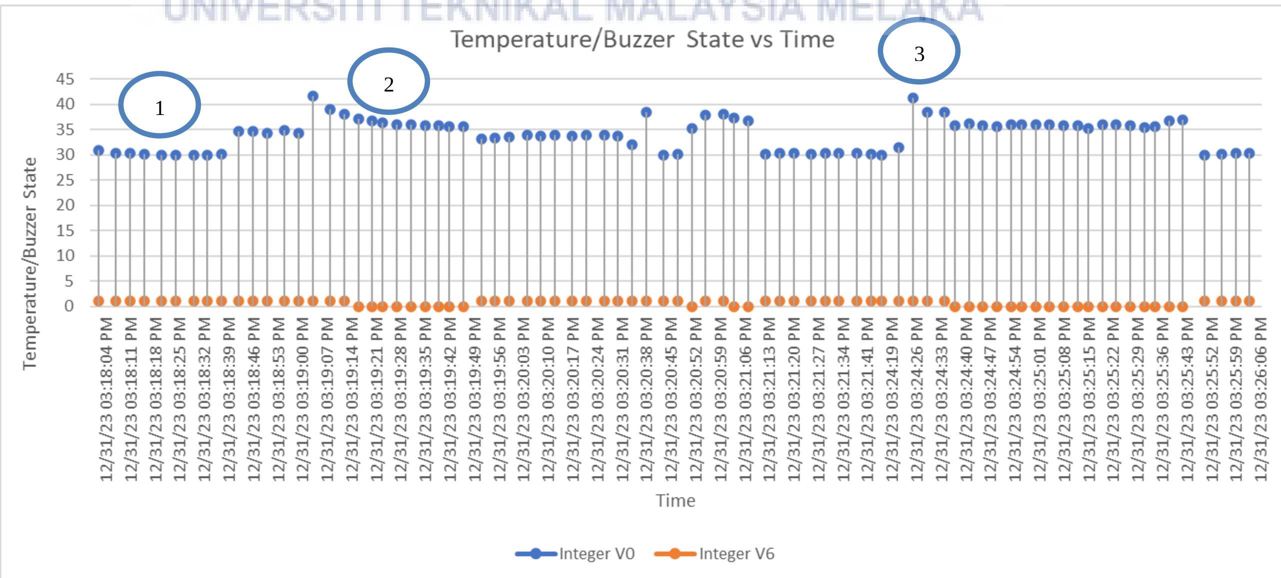


Figure 4.20: Temperature/Buzzer State vs Time Analysis Graph

4.3.2 MPU6050 Accelerometer sensor

Table 4.7 presents distinct gestures recorded by an accelerometer sensor, delineating specific movements along the x and y axes. For instance, readings like x=10.0, y=-0.9 correspond to the "Water" gesture, initiating a buzzer alert (buzzer state: 1). Similarly, values such as x=-9.3, y=-0.1 signify the "Washroom" gesture, prompting a specific response (buzzer state: 1). Movements indicated by x=-4.4, y=-10.0 represent a "Hungry" gesture, while x=2.4, y=9.5 denotes a "Medicine/Emergency" scenario, both resulting in buzzer activations (buzzer state: 1). Furthermore, instances with minimal activity, such as x=-0.9, y=-0.4, indicate "No movement detected," maintaining the buzzer at rest (buzzer state: 0). These data instances underscore the diverse gestures detected by the accelerometer, each linked to distinct movements and resulting in specific buzzer states or alerts. Figure 4.21 shows the gesture message based on the direction of accelerometer x-axis and y-axis value.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

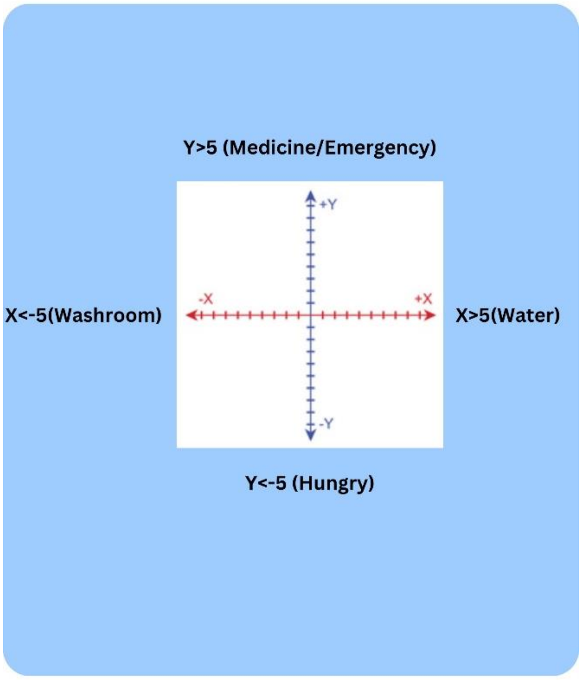


Figure 4.21: Accelerometer a-axis and y-axis gesture message

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.7: Accelerometer Sample Readings

Figure	Accelerometer X (m/s^2)	Accelerometer Y (m/s^2)	Gesture	Buzzer State
Figure 4.22	-1.0	1.3	No movement detected	0
Figure 4.23	0.1	10.1	Medicine/Emergency	1
Figure 4.24	2.4	9.5	Medicine/Emergency	1
Figure 4.25	2.3	-2.1	No movement detected	0
Figure 4.26	10.0	-0.9	Water	1
Figure 4.27	10.4	-0.6	Water	1
Figure 4.28	-0.9	-0.4	No movement detected	0
Figure 4.29	1.3	-10.6	Hungry	1
Figure 4.30	-0.5	-9.8	Hungry	1

Figure 4.31	0.6	0.5	No movement detected	0
Figure 4.32	-9.3	-0.1	Washroom	1
Figure 4.33	-9.2	-1.0	Washroom	1
Figure 4.34	0.2	0.8	No movement detected	0
Figure 4.35	2.0	9.4	Medicine/Emergency	1
Figure 4.36	0.2	2.4	No movement detected	0
Figure 4.37	-4.4	-10.0	Hungry	1

The provided information comprises figures illustrating the output from the serial monitor and its correlated display on the Blynk application.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:15:57.252 -> ------------------------------------------------------------
18:15:58.252 -> Accelerometer X: -1.0 m/s^2, Y: 1.3 m/s^2, Z: -8.1 m/s^2
18:15:58.252 -> Gesture:No movement detected
18:15:58.736 -> Buzzer State (Accelerometer): 0
18:15:58.736 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: 0.0 rps
18:15:58.736 -> ************************************************************
```

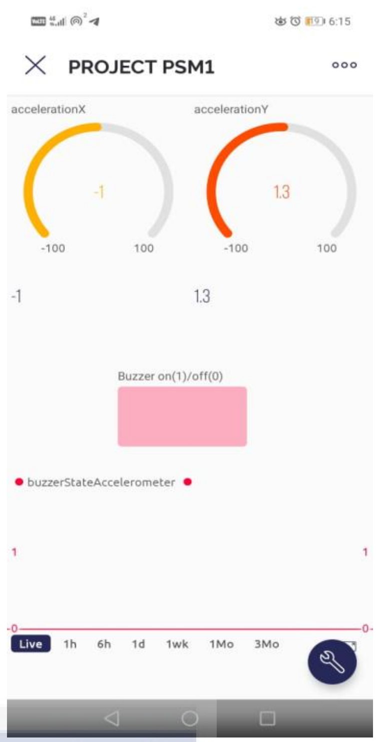


Figure 4.22: Accelerometer Reading Sample 1

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:16:01.012 -> ------------------------------------------------------------
18:16:02.008 -> Accelerometer X: 0.1 m/s^2, Y: 10.1 m/s^2, Z: 3.8 m/s^2
18:16:02.008 -> Gesture:Medicine/Emergency
18:16:03.020 -> Buzzer State (Accelerometer): 1
18:16:03.020 -> Gyroscope X: 0.2 rps, Y: 0.2 rps, Z: 0.3 rps
18:16:03.020 -> ************************************************************
```

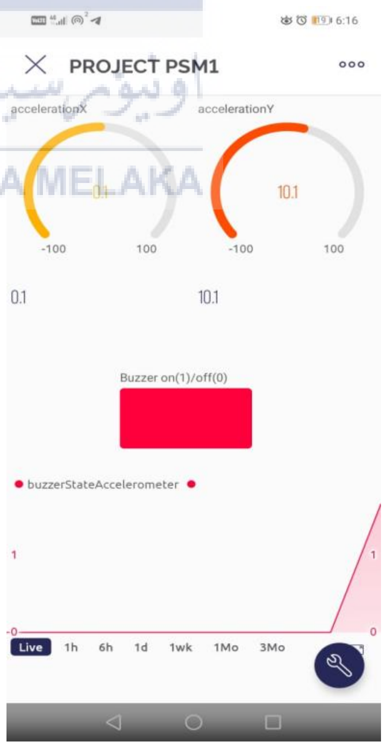


Figure 4.23: Accelerometer Reading Sample 2

```
18:16:21.768 -> ---------------------------------------------------------
18:16:22.773 -> Accelerometer X: 2.4 m/s^2, Y: 9.5 m/s^2, Z: 0.3 m/s^2
18:16:22.773 -> Gesture:Medicine/Emergency
18:16:23.780 -> Buzzer State (Accelerometer): 1
18:16:23.780 -> Gyroscope X: -0.0 rps, Y: -0.0 rps, Z: 0.0 rps
18:16:23.780 -> *********************************************************
```

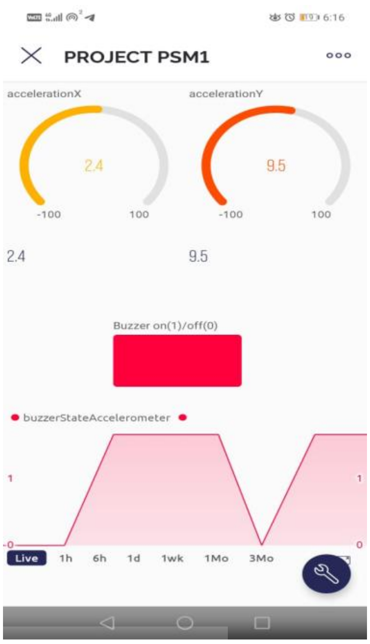


Figure 4.24: Accelerometer Reading Sample 3

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:16:26.017 -> ---------------------------------------------------------
18:16:27.029 -> Accelerometer X: 2.3 m/s^2, Y: -2.1 m/s^2, Z: -10.0 m/s^2
18:16:27.029 -> Gesture:No movement detected
18:16:27.522 -> Buzzer State (Accelerometer): 0
18:16:27.522 -> Gyroscope X: -0.5 rps, Y: -0.3 rps, Z: 0.2 rps
18:16:27.522 -> *********************************************************
```

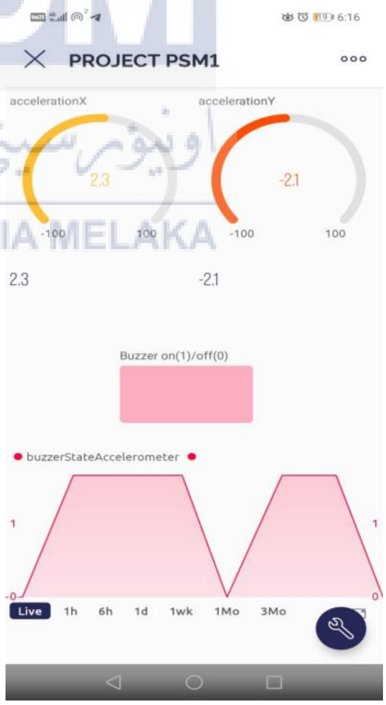


Figure 4.25: Acceleromter Reading Sample 4

```
18:16:33.516 -> ------------------------------------------------------------
18:16:34.511 -> Accelerometer X: 10.0 m/s^2, Y: -0.9 m/s^2, Z: 2.2 m/s^2
18:16:34.511 -> Gesture:Water
18:16:35.541 -> Buzzer State (Accelerometer): 1
18:16:35.541 -> Gyroscope X: -0.0 rps, Y: 0.0 rps, Z: 0.0 rps
18:16:35.541 -> ************************************************************
```

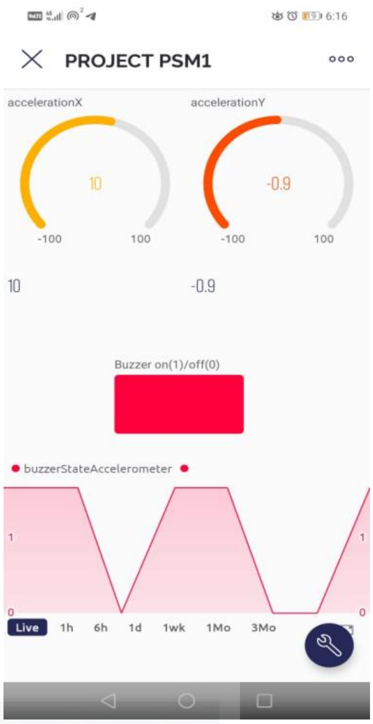


Figure 4.26: Acceleromter Reading Sample 5

```
18:16:37.738 -> ------------------------------------------------------------
18:16:38.765 -> Accelerometer X: 10.4 m/s^2, Y: -0.6 m/s^2, Z: 1.7 m/s^2
18:16:38.765 -> Gesture:Water
18:16:39.793 -> Buzzer State (Accelerometer): 1
18:16:39.793 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: 0.0 rps
18:16:39.793 -> ************************************************************
```

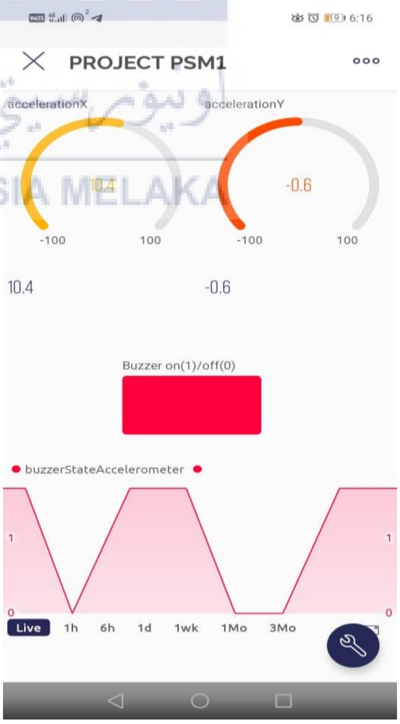


Figure 4.27: Acceleromter Reading Sample 6

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:16:49.528 -> ------------------------------------------------------------
18:16:50.494 -> Accelerometer X: -0.9 m/s^2, Y: -0.4 m/s^2, Z: -8.1 m/s^2
18:16:50.539 -> Gesture:No movement detected
18:16:51.036 -> Buzzer State (Accelerometer): 0
18:16:51.036 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: -0.0 rps
18:16:51.036 -> ************************************************************
```

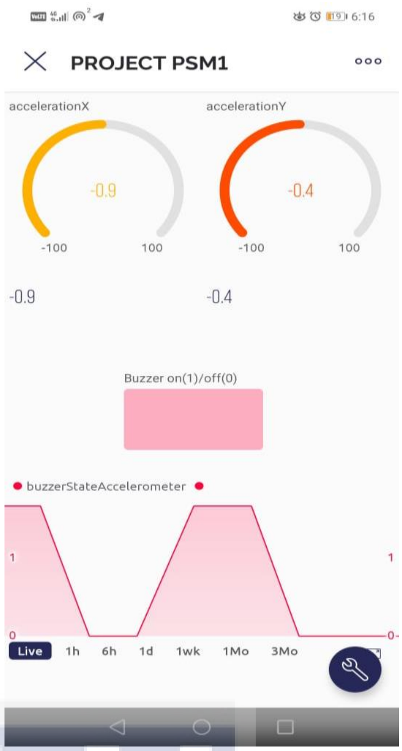


Figure 4.28: Acceleromter Reading Sample 7

اونيۆرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:16:53.277 -> ------------------------------------------------------------
18:16:54.275 -> Accelerometer X: 1.3 m/s^2, Y: -10.6 m/s^2, Z: 1.4 m/s^2
18:16:54.275 -> Gesture:Hungry
18:16:55.284 -> Buzzer State (Accelerometer): 1
18:16:55.284 -> Gyroscope X: -0.6 rps, Y: 0.3 rps, Z: 1.0 rps
18:16:55.284 -> ************************************************************
```

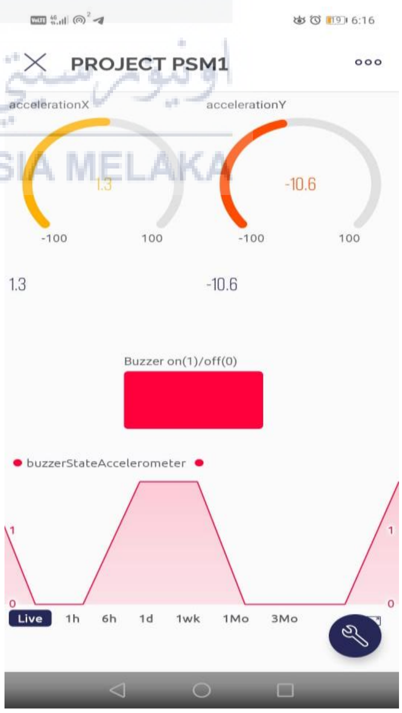


Figure 4.29: Acceleromter Reading Sample 8

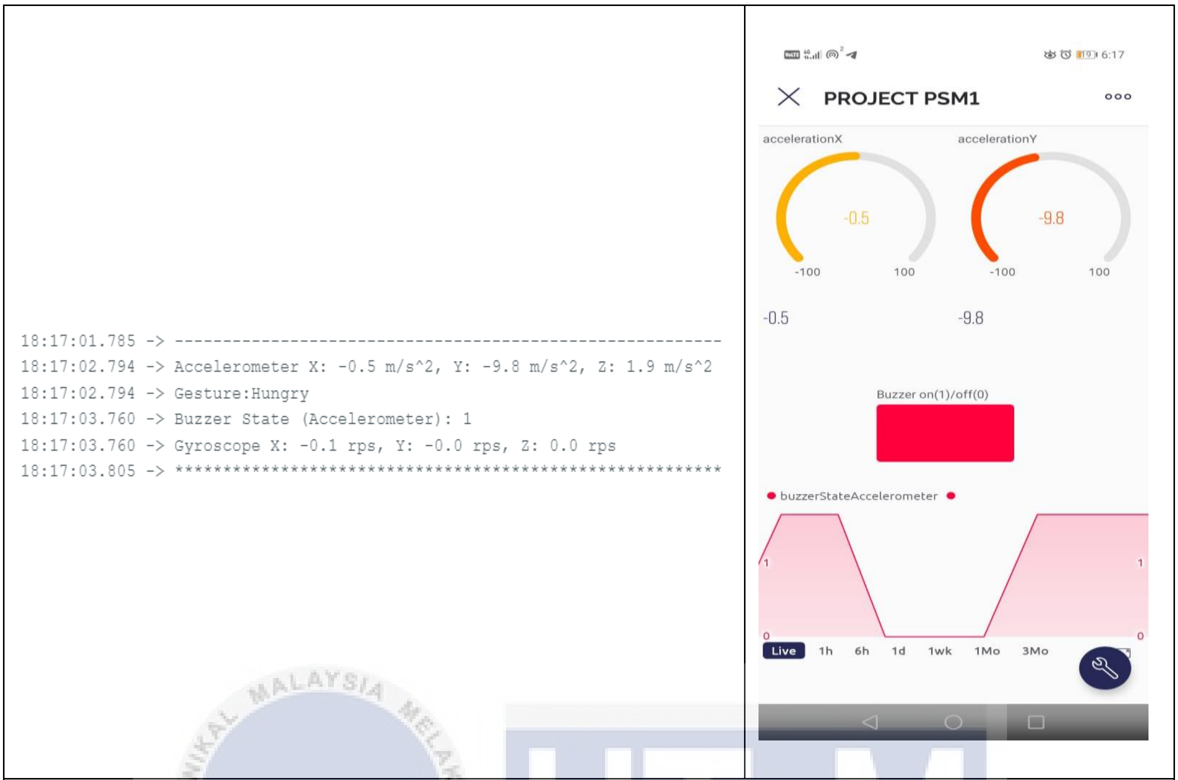


Figure 4.30: Acceleromter Reading Sample 9

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

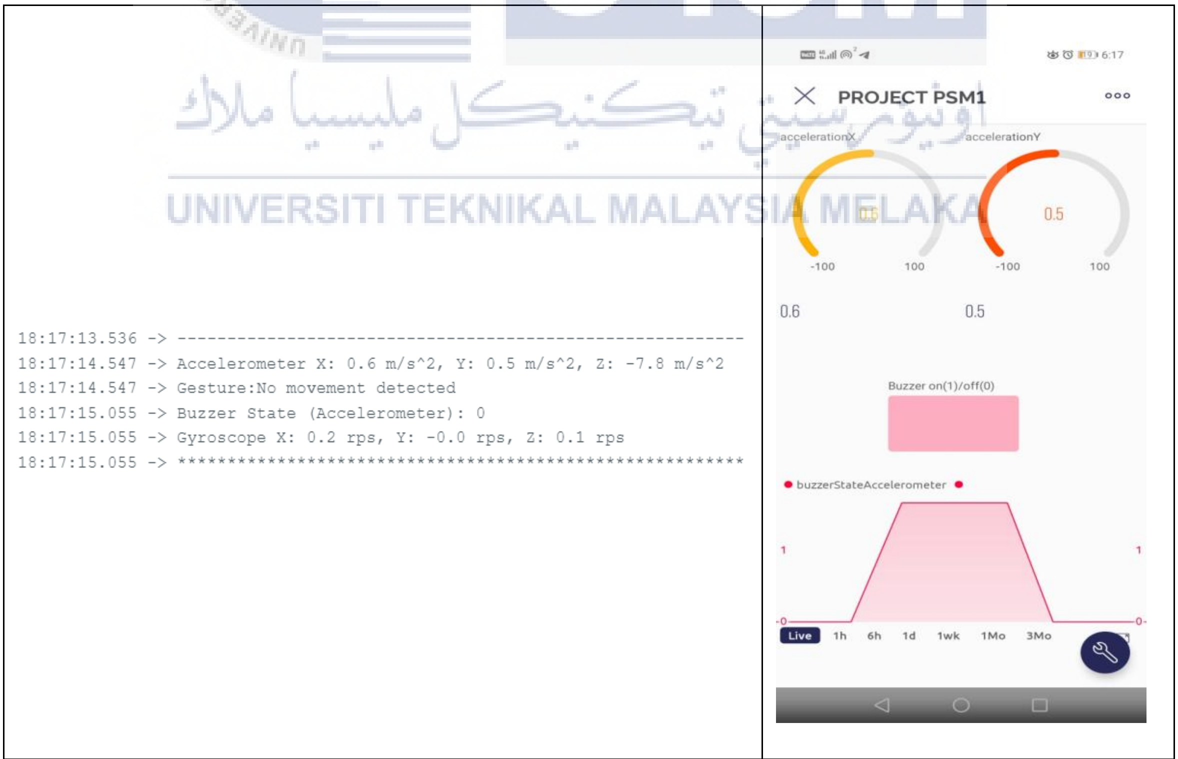


Figure 4.31: Acceleromter Reading Sample 10

```
18:17:21.044 -> ------------------------------------------------------------
18:17:22.025 -> Accelerometer X: -9.3 m/s^2, Y: -0.1 m/s^2, Z: 0.8 m/s^2
18:17:22.025 -> Gesture:Washroom
18:17:23.052 -> Buzzer State (Accelerometer): 1
18:17:23.052 -> Gyroscope X: -0.1 rps, Y: 0.1 rps, Z: -0.0 rps
18:17:23.052 -> ************************************************************
```

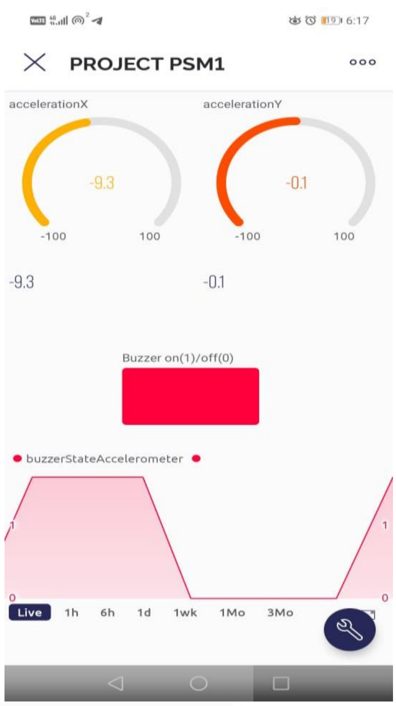


Figure 4.32: Acceleromter Reading Sample 11

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:17:38.010 -> ------------------------------------------------------------
18:17:39.055 -> Accelerometer X: -9.2 m/s^2, Y: -1.0 m/s^2, Z: 3.3 m/s^2
18:17:39.055 -> Gesture:Washroom
18:17:40.060 -> Buzzer State (Accelerometer): 1
18:17:40.060 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: 0.0 rps
18:17:40.060 -> ************************************************************
```

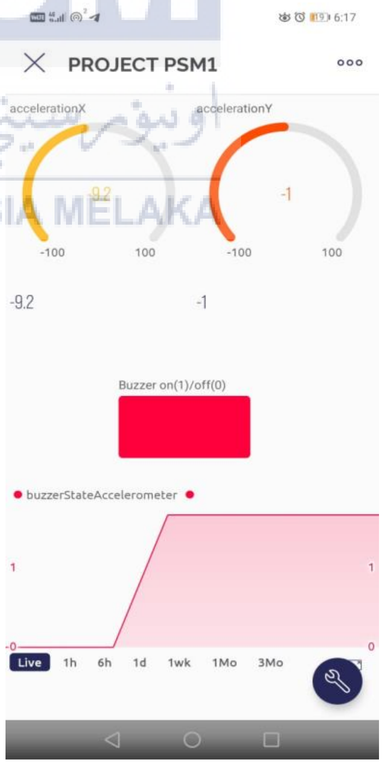


Figure 4.33: Accelerомter Reading Sample 12

```
18:17:42.296 -> ----------------------------------------------------------
18:17:43.290 -> Accelerometer X: 0.2 m/s^2, Y: 0.8 m/s^2, Z: 11.6 m/s^2
18:17:43.290 -> Gesture:No movement detected
18:17:43.803 -> Buzzer State (Accelerometer): 0
18:17:43.803 -> Gyroscope X: -0.2 rps, Y: 0.1 rps, Z: -0.1 rps
18:17:43.803 -> **********************************************************
```

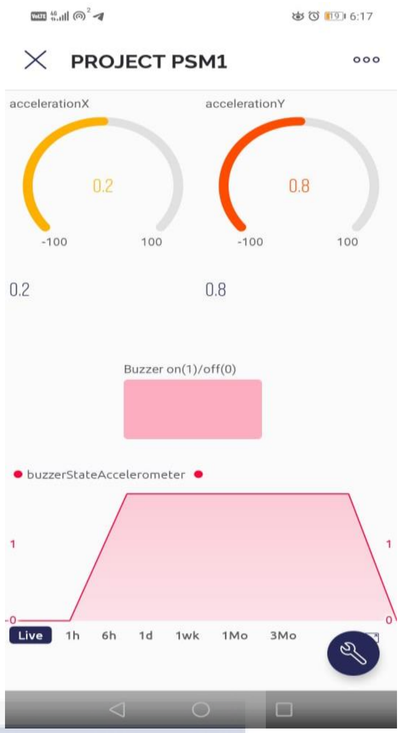


Figure 4.34: Acceleromter Reading Sample 13

```
18:17:49.801 -> ----------------------------------------------------------
18:17:50.768 -> Accelerometer X: 2.0 m/s^2, Y: 9.4 m/s^2, Z: 0.7 m/s^2
18:17:50.813 -> Gesture:Medicine/Emergency
18:17:51.779 -> Buzzer State (Accelerometer): 1
18:17:51.825 -> Gyroscope X: 0.1 rps, Y: -0.0 rps, Z: -0.2 rps
18:17:51.825 -> **********************************************************
```

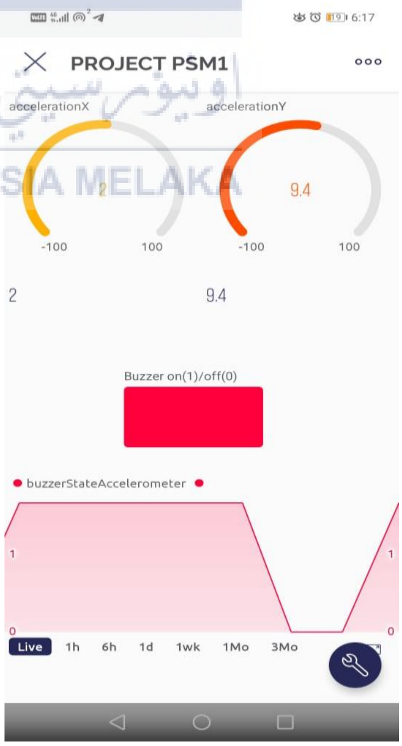


Figure 4.35: Acceleromter Reading Sample 14

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:17:58.271 -> ---------------------------------------------------------
18:17:59.314 -> Accelerometer X: 0.2 m/s^2, Y: 2.4 m/s^2, Z: 11.2 m/s^2
18:17:59.314 -> Gesture:No movement detected
18:17:59.827 -> Buzzer State (Accelerometer): 0
18:17:59.827 -> Gyroscope X: -0.1 rps, Y: -0.1 rps, Z: -0.0 rps
18:17:59.827 -> *********************************************************
```

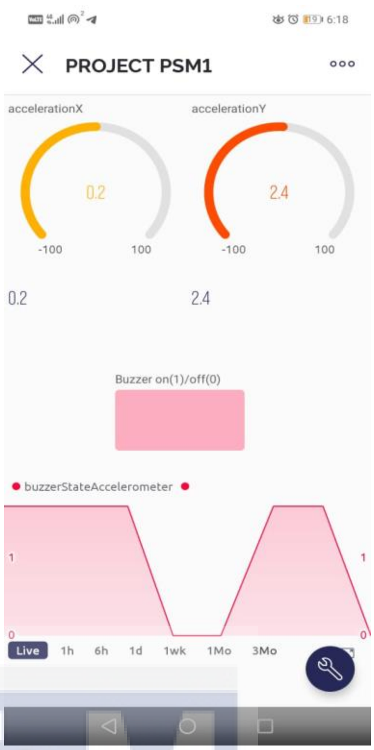


Figure 4.36: Accelerometer Reading Sample 15

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
18:18:02.022 -> ---------------------------------------------------------
18:18:03.048 -> Accelerometer X: -4.4 m/s^2, Y: -10.0 m/s^2, Z: -0.8 m/s^2
18:18:03.048 -> Gesture:Hungry
18:18:04.063 -> Buzzer State (Accelerometer): 1
18:18:04.063 -> Gyroscope X: -2.5 rps, Y: 2.1 rps, Z: -1.8 rps
18:18:04.063 -> *********************************************************
```

Figure 4.37: Accelerometer Reading Sample 16

4.3.2.1 Blynk Notification for MPU6050 Accelerometer Sensor

The Table 4.8 below shows accelerometer data log from various time points on December 31, 2023, reflects distinct movements triggering specific gestures and corresponding messages. At 06:21:03 PM, an X-axis value of 10.1 and a Y-axis value of 0.1 led to the detection of a "Water" gesture, prompting Gesture_Alert_3. Prior to this, at 06:20:47 PM, the X-axis at 2.3 and Y-axis at 9.3 initiated an "Emergency/Medicine" message through Gesture_Alert_1. A different movement, noted at 06:20:31 PM with an X-axis of 0.7 and Y-axis of -10.5, triggered a "Hungry" alert marked by Gesture_Alert_2. Preceding this, at 06:20:19 PM, an X-axis of -9.3 and Y-axis of 0.4 prompted a "Washroom" message (Gesture_Alert_4). Lastly, at 06:19:50 PM, the accelerometer recorded an X-axis of 10.2 and Y-axis of -0.7, resulting in another "Water" gesture and Gesture_Alert_3. Earlier, at 06:19:32 PM, an X-axis value of 0.8 and Y-axis value of 9.7 corresponded to an "Emergency/Medicine" message via Gesture_Alert_1. These distinct accelerometer readings, indicative of specific movements, triggered alerts that corresponded to different gestures as programmed within the system.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.8: Blynk Notification for MPU6050

Time	Integer V2 (x-axis)	Integer V3 (y-axis)	Event Type	Name	Description
12/31/23 06:21:03 PM	10.1	0.1	WARNING	Gesture_Alert_3	Water
12/31/23 06:20:47 PM	2.3	9.3	WARNING	Gesture_Alert_1	Medicine/Emergency
12/31/23 06:20:31 PM	0.7	-10.5	WARNING	Gesture_Alert_2	Hungry
12/31/23 06:20:19 PM	-9.3	0.4	WARNING	Gesture_Alert_4	Washroom
12/31/23 06:19:50 PM	10.2	-0.7	WARNING	Gesture_Alert_3	Water
12/31/23 06:19:32 PM	0.8	9.7	WARNING	Gesture_Alert_1	Medicine/Emergency
12/31/23 06:18:04 PM	-4.4	-10.0	WARNING	Gesture_Alert_2	Hungry
12/31/23 06:17:51 PM	2.0	9.4	WARNING	Gesture_Alert_1	Medicine/Emergency
12/31/23 06:17:23 PM	-9.3	-0.1	WARNING	Gesture_Alert_4	Washroom
12/31/23 06:16:55 PM	1.3	-10.6	WARNING	Gesture_Alert_2	Hungry
12/31/23 06:16:35 PM	10.0	-0.9	WARNING	Gesture_Alert_3	Water
12/31/23 06:16:03 PM	0.1	10.1	WARNING	Gesture_Alert_1	Medicine/Emergency

The provided data showcases notifications displayed in the Blynk application, indicating instances of "Gesture_Alert_1," "Gesture_Alert_2," "Gesture_Alert_3," or "Gesture_Alert_4." Each notification includes a description indicating whether the patient requires water, medicine/emergency assistance, is hungry, or needs to use the washroom. These notifications are accompanied by specific dates and times, presented in numerical form.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.38: Sample MPU6050 Notification

This graph in Figure 4.39 showcases two key elements: the Accelerometer's x-axis and y-axis readings alongside the Buzzer State over time, offering a detailed view of their correlation.

The Buzzer State is depicted as either 0 or 1, representing the absence or presence of a detected gesture, respectively. When no significant gesture is detected, the Buzzer State remains at 0. However, when the accelerometer detects a gesture that meets predefined conditions, the Buzzer State switches to 1, indicating the activation of the buzzer. For instance, Pattern 1 signifies the 'Water' gesture when the V2(X) value exceeds 5, prompting the buzzer to activate. Pattern 2 indicates the 'Hungry' gesture when the V3(Y) value falls below -5, activating the buzzer accordingly. Similarly, Pattern 3 triggers the 'Washroom' gesture if the V2(X) value drops below -5, prompting the buzzer. Lastly, Pattern 4 signifies the 'Medicine/Emergency' gesture when the V3(Y) value surpasses 5. Each pattern and value range corresponds to a specific action, signalling different needs or emergencies.

The introduction of a red line in the graph serves as a visual indicator pinpointing moments when the buzzer is triggered due to the detection of a gesture. This line highlights instances where the accelerometer's readings match the conditions for gesture recognition, leading to the buzzer turning on. This visual setup aids in swiftly recognizing when specific gestures align with changes in the buzzer's behaviour. By observing this representation, it becomes easier to identify patterns: instances when gestures meet the conditions and result in the activation of the buzzer. This clear visualization helps monitor and intervene promptly based on distinct gesture values, ensuring efficient tracking and response to detected movements.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.39: Accelerometer/ Buzzer State vs Time Analysis Graph

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3.3 MAX30100 Pulse Oximeter

The table shows output of MAX30100 in serial monitor and Blynk that the LED light on the sensor behaves in response to the heartbeat rate (BPM). When the heart rate is too slow (less than 50 beats per minute) or too fast (more than 100 beats per minute), the LED turns off. However, when the heart rate is just right (between 50 and 100 beats per minute), the LED stays on. For example, when the heart rate is 48.81and 102.42, the LED turns off, but when it's at 89.24 or 95.09, within the ideal range, the LED stays on. This means the LED reacts to how fast or slow the heart is beating, showing this information visually by being on or off.

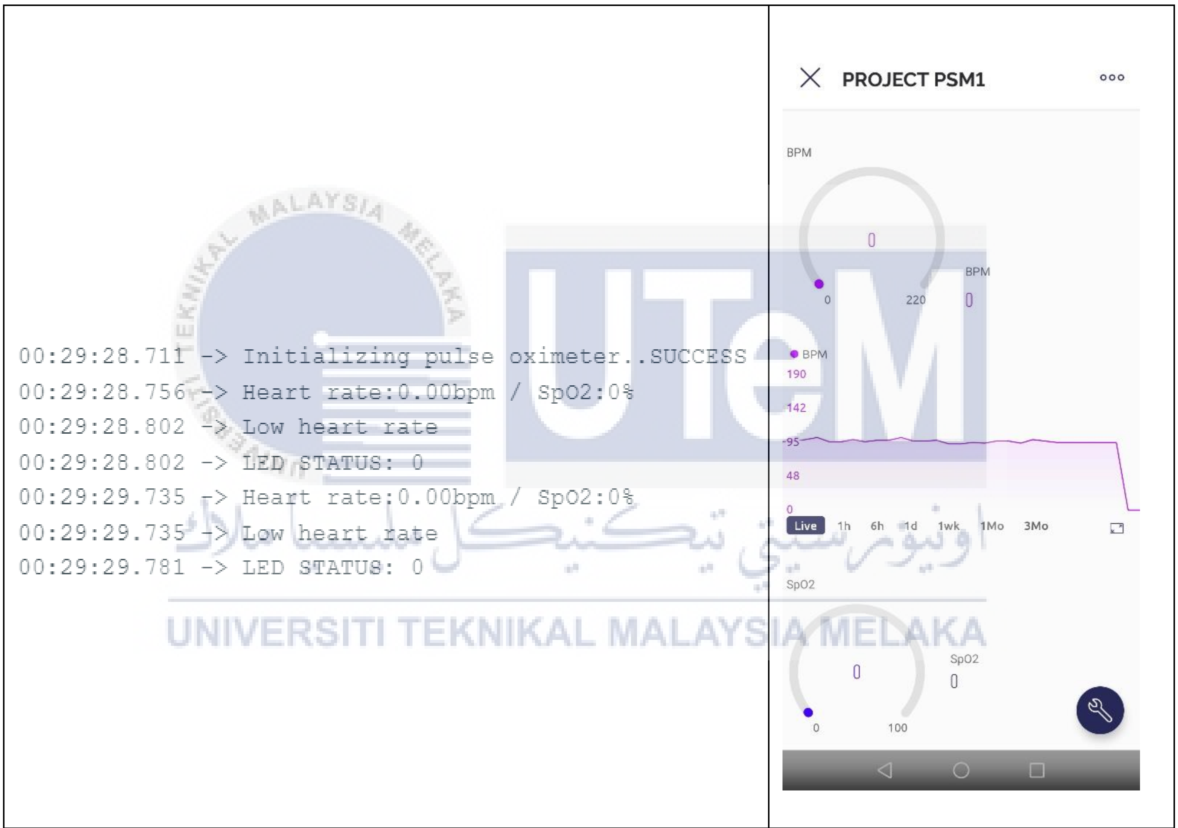
Table 4.9: BPM/SpO2 Sample Readings

Figure	Heart rate (bpm)	SpO2 (%)	LED State
Figure 4.40	0.00	0	0
Figure 4.41	13.33	0	0
Figure 4.42	48.81	93	0
Figure 4.43	40.19	97	0
Figure 4.44	37.89	93	0
Figure 4.45	89.24	99	1
Figure 4.46	58.84	98	1
Figure 4.47	98.19	98	1
Figure 4.48	102.42	100	0
Figure 4.49	110.51	99	0
Figure 4.50	109.32	100	0
Figure 4.51	95.09	100	1
Figure 4.52	101.72	100	0
Figure 4.53	82.02	96	1
Figure 4.54	98.75	96	1
Figure 4.55	97.95	99	1

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The provided information comprises figures illustrating the output from the serial monitor and its correlated display on the Blynk application.

When the hand is not placed on the MAX30100 pulse oximeter sensor, it typically returns a value of 0 as in Figure 4.40. This reading indicates that there's no signal or detection of the pulse or oxygen levels, commonly because there's no contact or interaction between the sensor and the hand.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.40: BPM/SpO2 Reading Sample 1

```
00:29:31.746 -> Heart rate:13.33bpm / SpO2:0%
00:29:31.746 -> Low heart rate
00:29:31.791 -> LED STATUS: 0
```

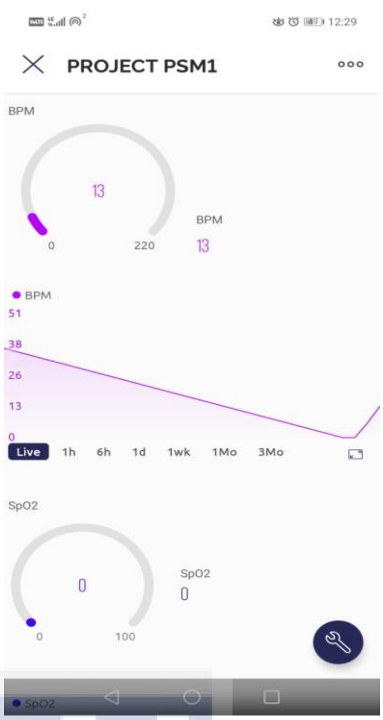


Figure 4.41: BPM/SpO2 Reading Sample 2

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
00:29:37.414 -> Beat!
00:29:37.767 -> Heart rate:48.81bpm / SpO2:93%
00:29:37.802 -> Low heart rate
00:29:37.802 -> LED STATUS: 0
```

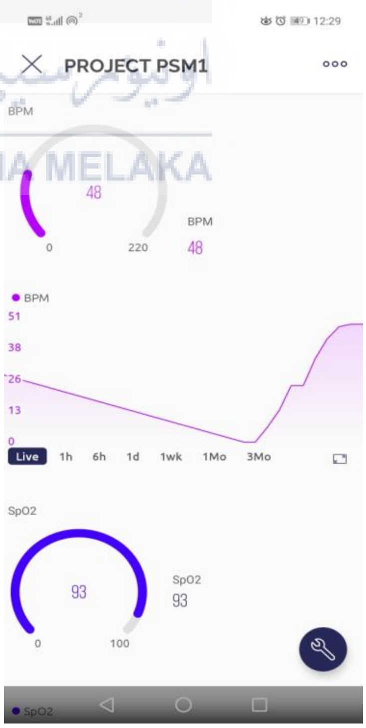


Figure 4.42: BPM/SpO2 Reading Sample 3

```
00:31:27.882 -> Beat!
00:31:28.163 -> Heart rate:40.19bpm / SpO2:97%
00:31:28.209 -> Low heart rate
00:31:28.209 -> LED STATUS: 0
```

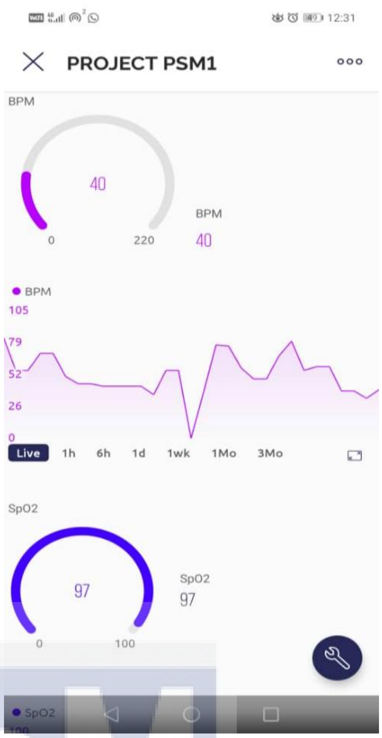


Figure 4.43: BPM/SpO2 Reading Sample 4

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
00:35:21.092 -> Beat!
00:35:21.589 -> Heart rate:37.89bpm / SpO2:93%
00:35:21.635 -> Low heart rate
00:35:21.635 -> LED STATUS: 0
```

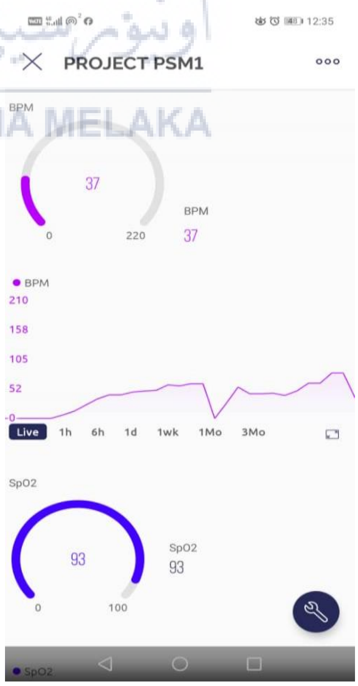


Figure 4.44: BPM/SpO2 Reading Sample 5

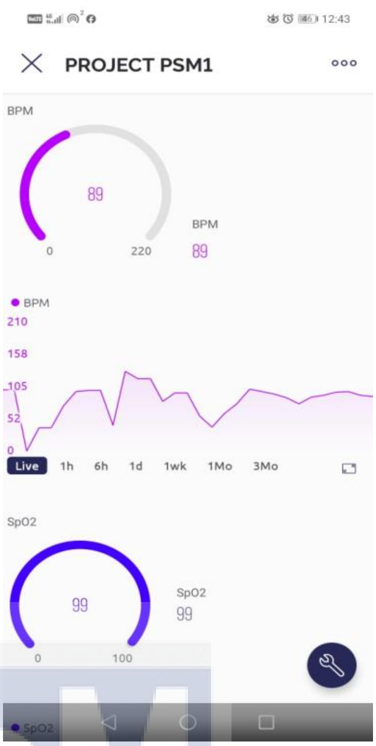
00:43:05.469 -> Beat! 00:43:05.925 -> Heart rate:89.24bpm / SpO2:99% 00:43:05.968 -> Optimal heart rate 00:43:05.968 -> LED STATUS: 1	

Figure 4.45: BPM/SpO2 Reading Sample 6

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

00:46:17.329 -> Beat! 00:46:18.152 -> Heart rate:58.84bpm / SpO2:98% 00:46:18.187 -> Optimal heart rate 00:46:18.187 -> LED STATUS: 1	

Figure 4.46: BPM/SpO2 Reading Sample 7

```
00:47:59.988 -> Beat!
00:48:00.524 -> Heart rate:98.19bpm / SpO2:98%
00:48:00.561 -> Optimal heart rate
00:48:00.561 -> LED STATUS: 1
```

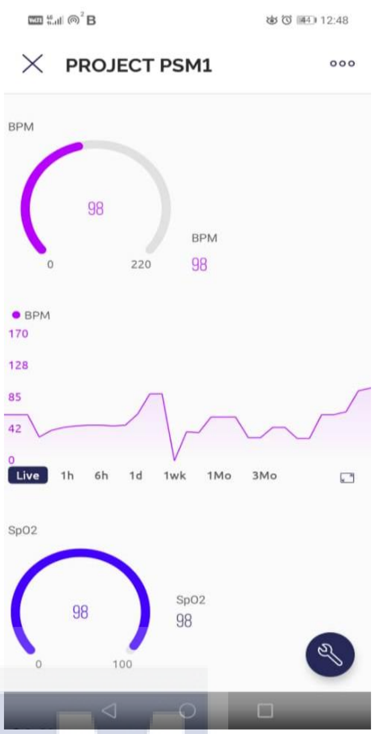


Figure 4.47: BPM/SpO2 Reading Sample 8

```
00:48:02.530 -> Heart rate:102.42bpm / SpO2:100%
00:48:02.567 -> High heart rate
00:48:02.567 -> LED STATUS: 0
```

Figure 4.48: BPM/SpO2 Reading Sample 9

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

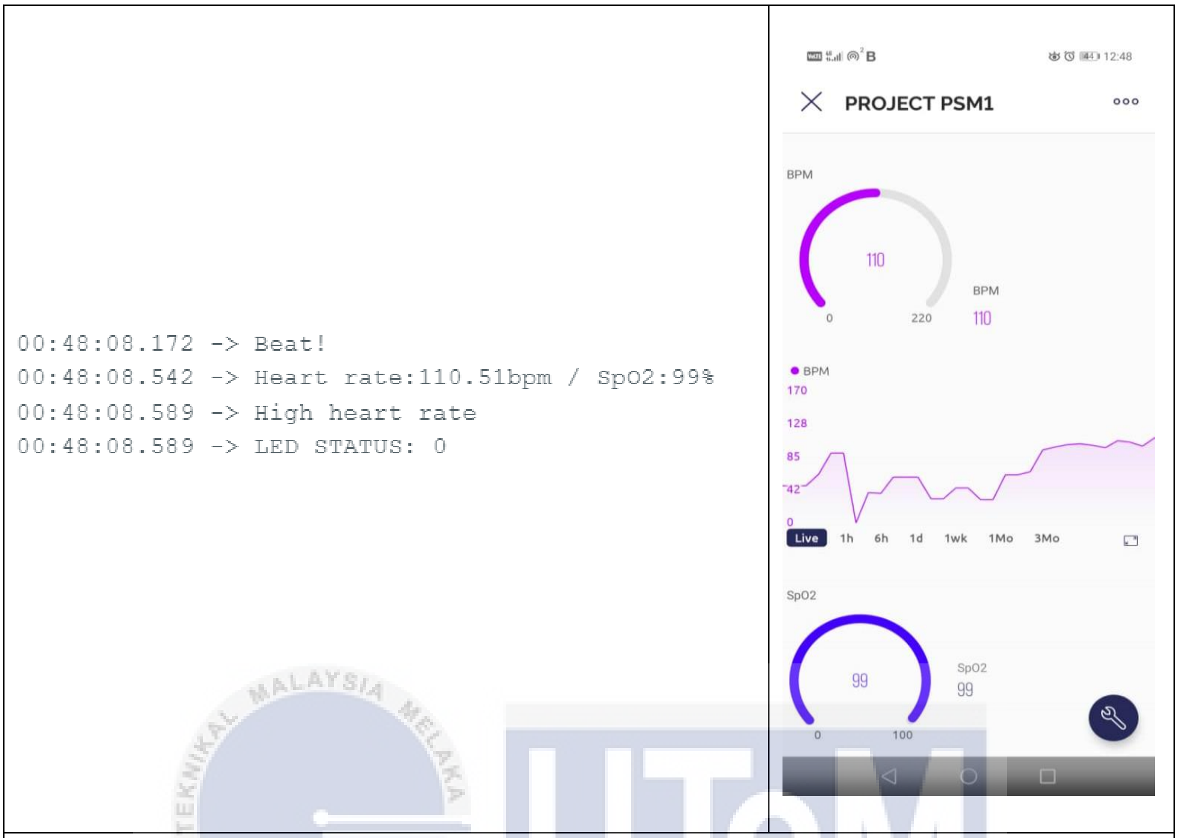


Figure 4.49: BPM/SpO2 Reading Sample 10

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

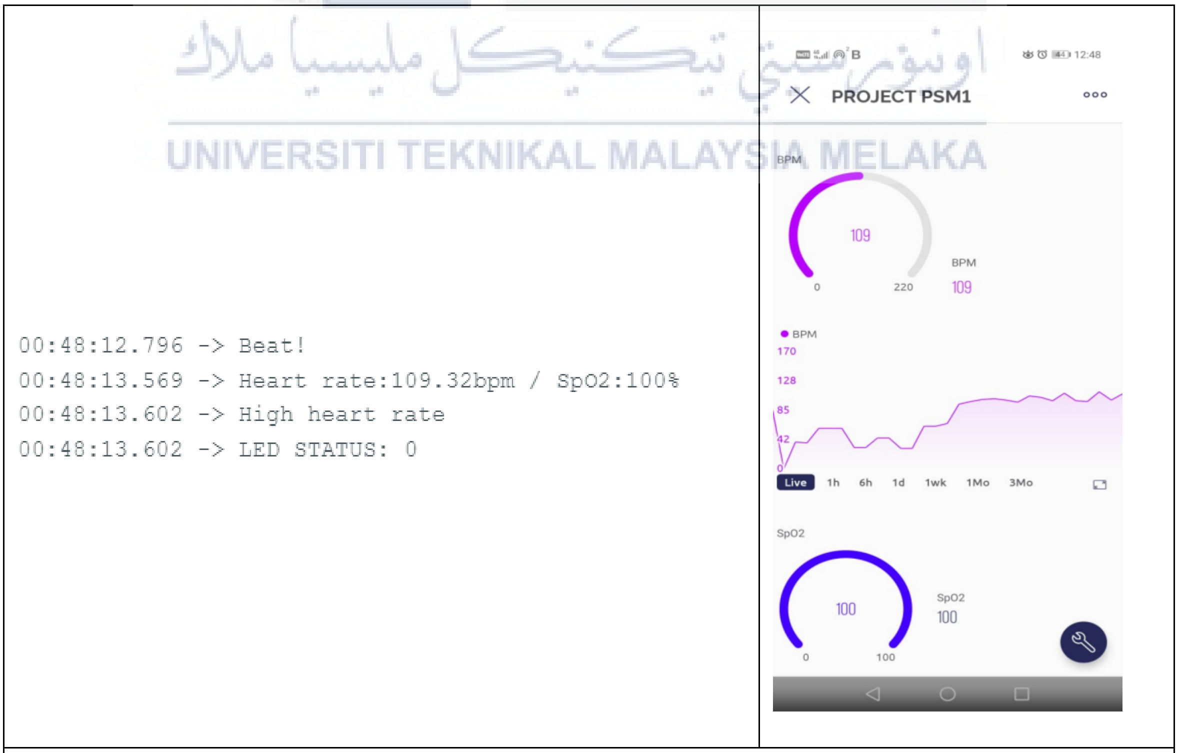


Figure 4.50: BPM/SpO2 Reading Sample 11

```
00:48:15.171 -> Beat!
00:48:15.575 -> Heart rate:95.09bpm / SpO2:100%
00:48:15.621 -> Optimal heart rate
00:48:15.621 -> LED STATUS: 1
```

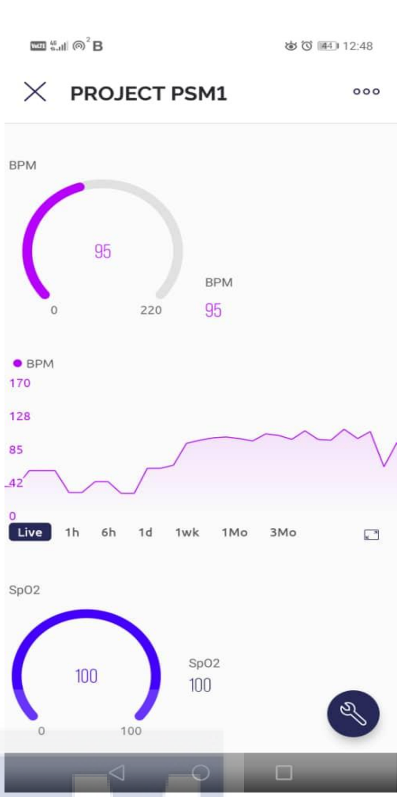


Figure 4.51: BPM/SpO2 Reading Sample 12

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
00:48:17.548 -> Beat!
00:48:17.581 -> Heart rate:101.72bpm / SpO2:100%
00:48:17.626 -> High heart rate
00:48:17.626 -> LED STATUS: 0
```

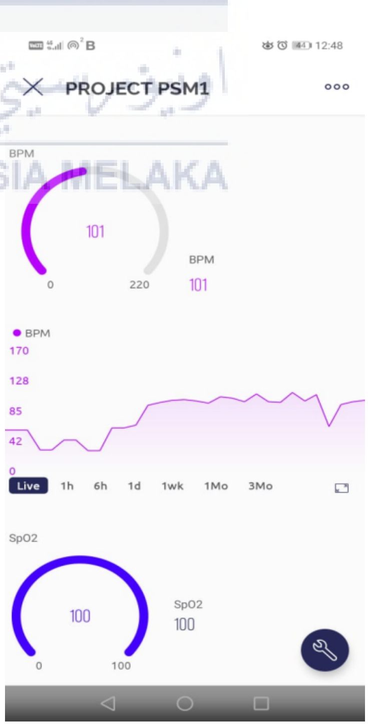


Figure 4.52: BPM/SpO2 Reading Sample 13

```
00:48:20.556 -> Beat!
00:48:20.593 -> Heart rate:82.02bpm / SpO2:96%
00:48:20.634 -> Optimal heart rate
00:48:20.634 -> LED STATUS: 1
```

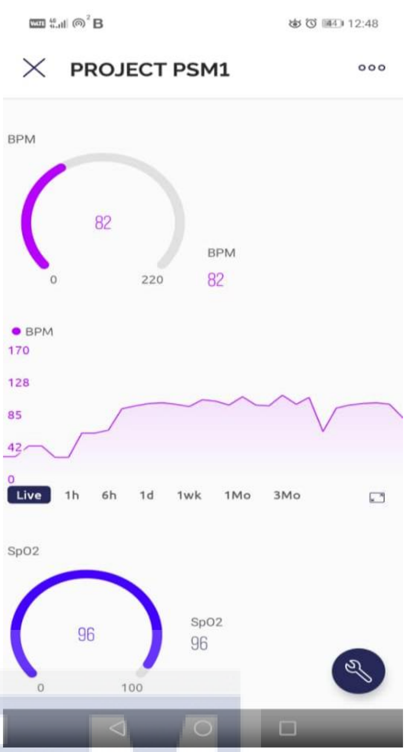


Figure 4.53: BPM/SpO2 Reading Sample 14

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
00:48:21.082 -> Beat!
00:48:21.597 -> Heart rate:98.75bpm / SpO2:96%
00:48:21.638 -> Optimal heart rate
00:48:21.638 -> LED STATUS: 1
```

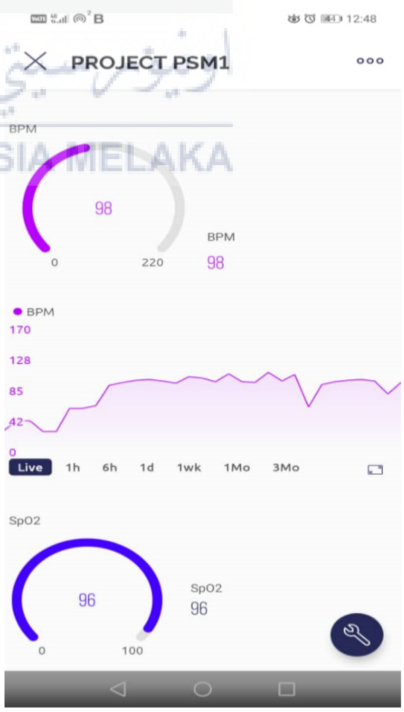


Figure 4.54: BPM/SpO2 Reading Sample 15

```
00:49:10.613 -> Beat!
00:49:10.738 -> Heart rate:97.95bpm / SpO2:99%
00:49:10.814 -> Optimal heart rate
00:49:10.814 -> LED STATUS: 1
```

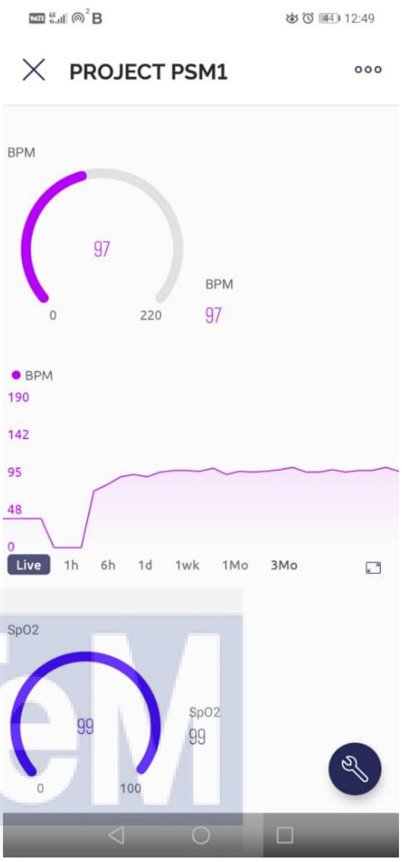


Figure 4.55: BPM/SpO2 Reading Sample 16

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3.3.1 Blynk Notification for MAX30100 Pulse Oximeter Sensor

The recorded Blynk notification samples that show in Table 4.10 present instances triggering alerts for both low and high BPM (beats per minute) readings, offering insight into potential health concerns. Several occurrences triggered low BPM alerts (BPM < 50), indicating notably slow heart rates. For instance, at 1/1/2024 0:40:22, the BPM was recorded as 17, signifying a significant drop-in heart rate. Similarly, readings of 37 and 32 at 1/1/2024 0:48:34 and 1/1/2024 0:47:33 respectively, continued to prompt low BPM alerts, suggesting persistently low heart rates during those times. Conversely, instances such as 1/1/2024 0:49:02 and 1/1/2024 0:46:47 registered BPM values of 102 and 111 respectively, triggering high BPM alerts. These readings indicated elevated heart rates, potentially signaling health concerns. This data highlights the system's monitoring capability, promptly notifying users or caregivers when heart rates deviate from predefined thresholds for low and high BPM. Such timely alerts enable appropriate attention and care to potential health issues as they arise.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.10: Blynk Notification for MAX30100

Time	Integer V1	Integer V5	Event Type	Name	Description
1/1/2024 0:49:35	0	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:49:02	102	99	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:48:34	37	95	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:48:01	101	98	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:47:33	32	95	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:46:47	111	100	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:46:30	42	98	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:45:36	105	40	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:45:30	0	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:43:26	126	95	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:43:21	0	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:42:15	103	97	WARNING	High_BPM_Alert	High BPM
1/1/2024 0:42:01	8	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:40:22	17	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:39:08	56	0	WARNING	Low_BPM_Alert	Low BPM
1/1/2024 0:38:28	109	105	WARNING	High_BPM_Alert	High BPM

The following figure 4.56 show the notifications for Low_BPM_Alert and High_BPM_Alert are generated based on specific conditions: Low_BPM_Alert triggers when a BPM reading falls below 50, indicating a very low heart rate, while High_BPM_Alert activates when the BPM exceeds 100, signaling a high heart rate. Each notification includes the date and time when these conditions are met. These alerts serve as timely indicators, providing the date and time of instances where the heart rate deviates significantly from the normal range, prompting attention to potential health concerns.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.56: Sample MAX30100 Notification

This Figure 4.57 shows a BPM/SpO2 vs Time graph that displays the LED state corresponding to the BPM values. Specifically, the LED state is represented as 0 when the BPM is either greater than 100 (Point 2) or less than 50 (Point 3). Conversely, when the BPM falls within the normal range of 50 to 100 (Point1), the LED state is indicated as 1, and there's a red line added to indicate when the LED turns on during this normal pulse range.

This configuration likely offers a clear visual representation of the relationship between BPM values and the LED state. The red line's presence signifies instances when the LED activates, specifically coinciding with the normal pulse range of 50 to 100 BPM. This visual cue helps easily identify when the BPM readings are within the desired normal limits, aligning with the LED state's behavior.

This visual representation allows for quick identification of when the BPM readings trigger the LED state changes, emphasizing both abnormal and normal BPM ranges with distinct LED behaviors for efficient monitoring and intervention.

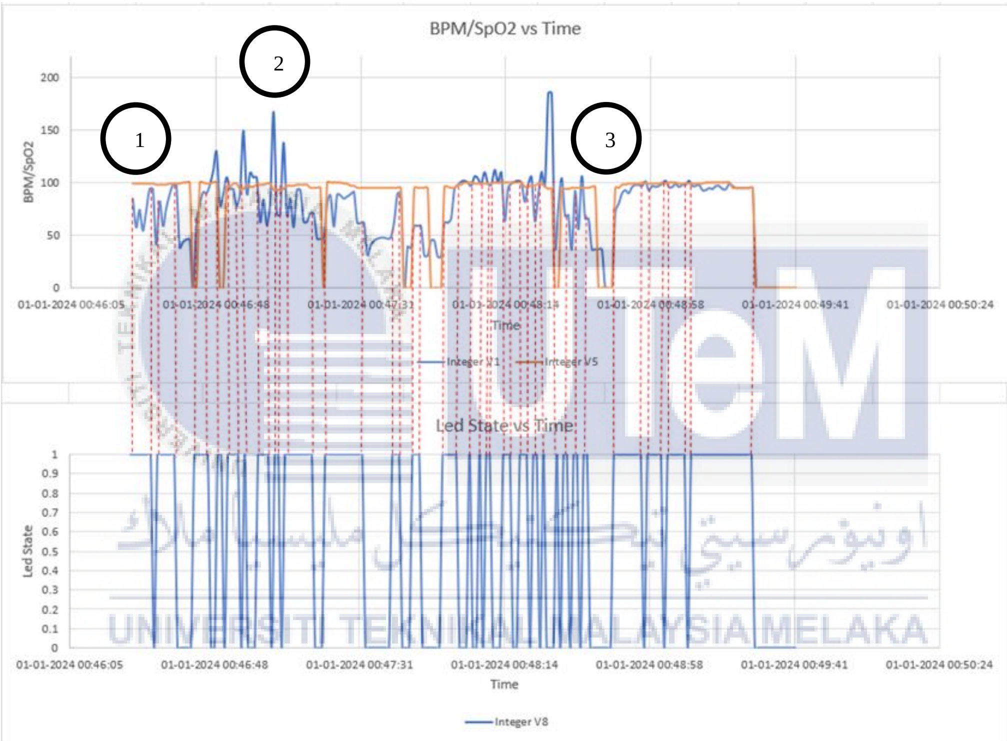


Figure 4.57: BPM/SpO2 vs Time & Led State vs Time Analysis Graph

4.4 Serial Monitor Output

4.4.1 Serial Output for ESP8266

```
00:48:54.039 -> Beat!
00:48:54.601 -> Beat!
00:48:54.715 -> Heart rate:98.71bpm / SpO2:99%
00:48:54.756 -> Optimal heart rate
00:48:54.756 -> LED STATUS: 1
00:48:55.207 -> Beat!
00:48:55.718 -> Heart rate:97.07bpm / SpO2:100%
00:48:55.751 -> Optimal heart rate
00:48:55.751 -> LED STATUS: 1
00:48:55.876 -> Beat!
00:48:56.439 -> Beat!
00:48:56.704 -> Heart rate:101.03bpm / SpO2:100%
00:48:56.751 -> High heart rate
00:48:56.751 -> LED STATUS: 0
00:48:47.635 -> Beat!
00:48:47.681 -> Heart rate:72.52bpm / SpO2:95%
00:48:47.728 -> Optimal heart rate
00:48:47.728 -> LED STATUS: 1
00:48:48.332 -> Beat!
00:48:48.692 -> Heart rate:80.16bpm / SpO2:95%
00:48:48.736 -> Optimal heart rate
00:48:48.736 -> LED STATUS: 1
00:48:48.922 -> Beat!
00:48:49.608 -> Beat!
00:48:49.695 -> Heart rate:90.81bpm / SpO2:98%
00:48:49.731 -> Optimal heart rate
00:48:49.731 -> LED STATUS: 1
00:48:50.229 -> Beat!
00:48:50.699 -> Heart rate:93.07bpm / SpO2:98%
00:48:50.743 -> Optimal heart rate
00:48:50.743 -> LED STATUS: 1
00:48:50.835 -> Beat!
00:48:51.538 -> Beat!
00:48:51.703 -> Heart rate:90.15bpm / SpO2:99%
00:48:51.740 -> Optimal heart rate
00:48:51.740 -> LED STATUS: 1
00:48:52.128 -> Beat!
00:48:52.702 -> Heart rate:96.27bpm / SpO2:99%
00:48:52.736 -> Optimal heart rate
00:48:52.736 -> LED STATUS: 1
00:48:52.780 -> Beat!
00:48:53.371 -> Beat!
00:48:53.695 -> Heart rate:98.12bpm / SpO2:99%
00:48:53.742 -> Optimal heart rate
00:48:53.742 -> LED STATUS: 1
00:48:42.579 -> Beat!
00:48:42.669 -> Heart rate:37.31bpm / SpO2:0%
00:48:42.669 -> Low heart rate
00:48:42.703 -> LED STATUS: 0
00:48:43.672 -> Heart rate:37.31bpm / SpO2:0%
00:48:43.672 -> Low heart rate
00:48:43.714 -> LED STATUS: 0
00:48:44.676 -> Heart rate:0.00bpm / SpO2:0%
00:48:44.676 -> Low heart rate
00:48:44.712 -> LED STATUS: 0
00:48:45.675 -> Heart rate:0.00bpm / SpO2:0%
00:48:45.675 -> Low heart rate
00:48:45.722 -> LED STATUS: 0
00:48:46.686 -> Heart rate:0.00bpm / SpO2:0%
00:48:46.686 -> Low heart rate
00:48:46.686 -> LED STATUS: 0
00:48:46.732 -> Beat!
00:48:47.041 -> Beat!
```

Figure 4.58: Serial Monitor Output of ESP8266

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.4.2 Serial Output for ESP32

```
04:57:30.809 -> **********************************************************
04:57:31.963 -> Patient name: John
04:57:32.974 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:32.974 -> Body temperature = 32.31°C
04:57:32.974 -> Body temperature = 90.16°F
04:57:33.445 -> Low Temperature
04:57:33.445 -> Buzzer State (Temperature): 1
04:57:33.627 -> ----------------------------------------------------------
04:57:34.593 -> Accelerometer X: -1.6 m/s^2, Y: 3.9 m/s^2, Z: -7.4 m/s^2
04:57:34.640 -> Gesture:No movement detected
04:57:35.152 -> Buzzer State (Accelerometer): 0
04:57:35.152 -> Gyroscope X: -0.5 rps, Y: -0.3 rps, Z: -0.3 rps
04:57:35.152 -> **********************************************************
04:57:36.320 -> Patient name: John
04:57:37.321 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:37.321 -> Body temperature = 34.97°C
04:57:37.321 -> Body temperature = 94.95°F
04:57:37.833 -> Low Temperature
04:57:37.833 -> Buzzer State (Temperature): 1
04:57:37.974 -> ----------------------------------------------------------
04:57:38.975 -> Accelerometer X: 1.2 m/s^2, Y: 3.9 m/s^2, Z: 7.4 m/s^2
04:57:38.975 -> Gesture:No movement detected
04:57:39.506 -> Buzzer State (Accelerometer): 0
04:57:39.506 -> Gyroscope X: -1.0 rps, Y: -0.1 rps, Z: -1.3 rps
04:57:39.506 -> **********************************************************
04:57:40.675 -> Patient name: John
04:57:41.686 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:41.686 -> Body temperature = 33.85°C
04:57:41.686 -> Body temperature = 92.93°F
04:57:42.182 -> Low Temperature
04:57:42.182 -> Buzzer State (Temperature): 1
04:57:42.307 -> ----------------------------------------------------------
04:57:43.328 -> Accelerometer X: 3.8 m/s^2, Y: 8.8 m/s^2, Z: 0.2 m/s^2
04:57:43.328 -> Gesture:Medicine/Emergency
04:57:44.355 -> Buzzer State (Accelerometer): 1
04:57:44.355 -> Gyroscope X: -0.2 rps, Y: 0.2 rps, Z: 0.4 rps
04:57:44.355 -> **********************************************************
04:57:45.525 -> Patient name: John
04:57:46.539 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:46.539 -> Body temperature = 33.79°C
04:57:46.539 -> Body temperature = 92.82°F
04:57:47.036 -> Low Temperature
04:57:47.036 -> Buzzer State (Temperature): 1
04:57:47.175 -> ----------------------------------------------------------
04:57:48.190 -> Accelerometer X: 10.1 m/s^2, Y: -0.8 m/s^2, Z: 1.3 m/s^2
04:57:48.190 -> Gesture:Water
04:57:49.188 -> Buzzer State (Accelerometer): 1
04:57:49.188 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: -0.0 rps
04:57:49.232 -> **********************************************************
04:57:50.368 -> Patient name: John
04:57:51.395 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:51.395 -> Body temperature = 40.61°C
04:57:51.395 -> Body temperature = 105.10°F
04:57:51.893 -> High Temperature
04:57:51.893 -> Buzzer State (Temperature): 1
04:57:52.033 -> ----------------------------------------------------------
04:57:53.044 -> Accelerometer X: 10.3 m/s^2, Y: 0.2 m/s^2, Z: 1.2 m/s^2
04:57:53.044 -> Gesture:Water
04:57:54.071 -> Buzzer State (Accelerometer): 1
04:57:54.071 -> Gyroscope X: -0.1 rps, Y: -0.0 rps, Z: 0.0 rps
04:57:54.071 -> **********************************************************
04:57:55.232 -> Patient name: John
04:57:56.246 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:57:56.246 -> Body temperature = 32.81°C
04:57:56.246 -> Body temperature = 91.06°F
04:57:56.746 -> Low Temperature
04:57:56.746 -> Buzzer State (Temperature): 1
04:57:56.886 -> ----------------------------------------------------------
04:57:57.907 -> Accelerometer X: 10.0 m/s^2, Y: 0.4 m/s^2, Z: 1.0 m/s^2
04:57:57.907 -> Gesture:Water
04:57:58.921 -> Buzzer State (Accelerometer): 1
04:57:58.921 -> Gyroscope X: 0.0 rps, Y: 0.0 rps, Z: -0.1 rps
04:57:58.921 -> **********************************************************
04:58:00.088 -> Patient name: John
04:58:01.117 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:58:01.117 -> Body temperature = 35.01°C
04:58:01.117 -> Body temperature = 95.02°F
04:58:01.117 -> Optimal Temperature
04:58:01.117 -> Buzzer State (Temperature): 0
04:58:01.208 -> ----------------------------------------------------------
04:58:02.170 -> Accelerometer X: 4.5 m/s^2, Y: -8.8 m/s^2, Z: 3.4 m/s^2
04:58:02.170 -> Gesture:Hungry
04:58:03.222 -> Buzzer State (Accelerometer): 1
04:58:03.222 -> Gyroscope X: 0.2 rps, Y: 0.1 rps, Z: 1.6 rps
04:58:03.222 -> **********************************************************
04:58:04.392 -> Patient name: John
04:58:05.403 -> ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
04:58:05.403 -> Body temperature = 34.37°C
04:58:05.403 -> Body temperature = 93.87°F
04:58:05.901 -> Low Temperature
04:58:05.901 -> Buzzer State (Temperature): 1
04:58:06.039 -> ----------------------------------------------------------
```

Figure 4.59: Serial Monitor Output of ESP32

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.5 Blynk Notification (Automation)

Figure 4.60 shows the Low_Temperature_Alert notification triggers when the patient's body temperature falls below a safe level (<35°C). High_Temperature_Alert activates when the patient's body temperature rises above a safe range (>37.5°C). Low_BPM_Alert notifies if the heart rate gets too slow (<50). High_BPM_Alert alerts if the heart rate becomes too fast (>100). Gesture_Alert_1 (Medicine/Emergency) is a specific gesture for calling for help or indicating the need for medication or urgent assistance. Gesture_Alert_2 (Hungry) is a signal when the patient feels hungry. Gesture_Alert_3 (Water) indicates when the patient feels thirsty and needs water. Gesture_Alert_4 (Washroom) alerts when the patient needs to use the restroom. These notifications collectively serve as a comprehensive communication and monitoring system for a paralysis patient, allowing them to signal emergencies, express basic needs, and monitor health parameters even with limited physical ability.

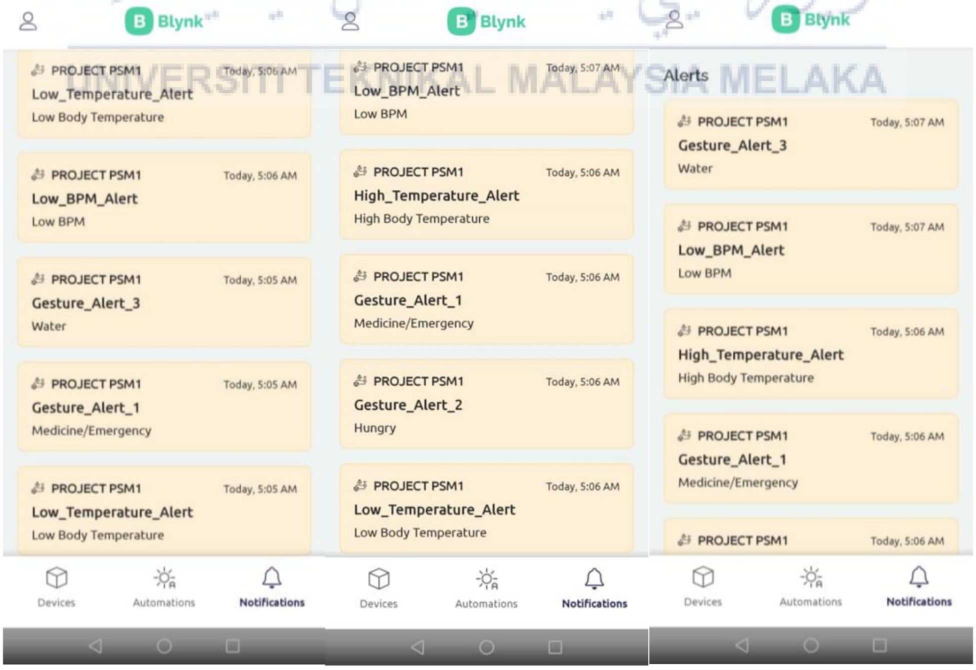
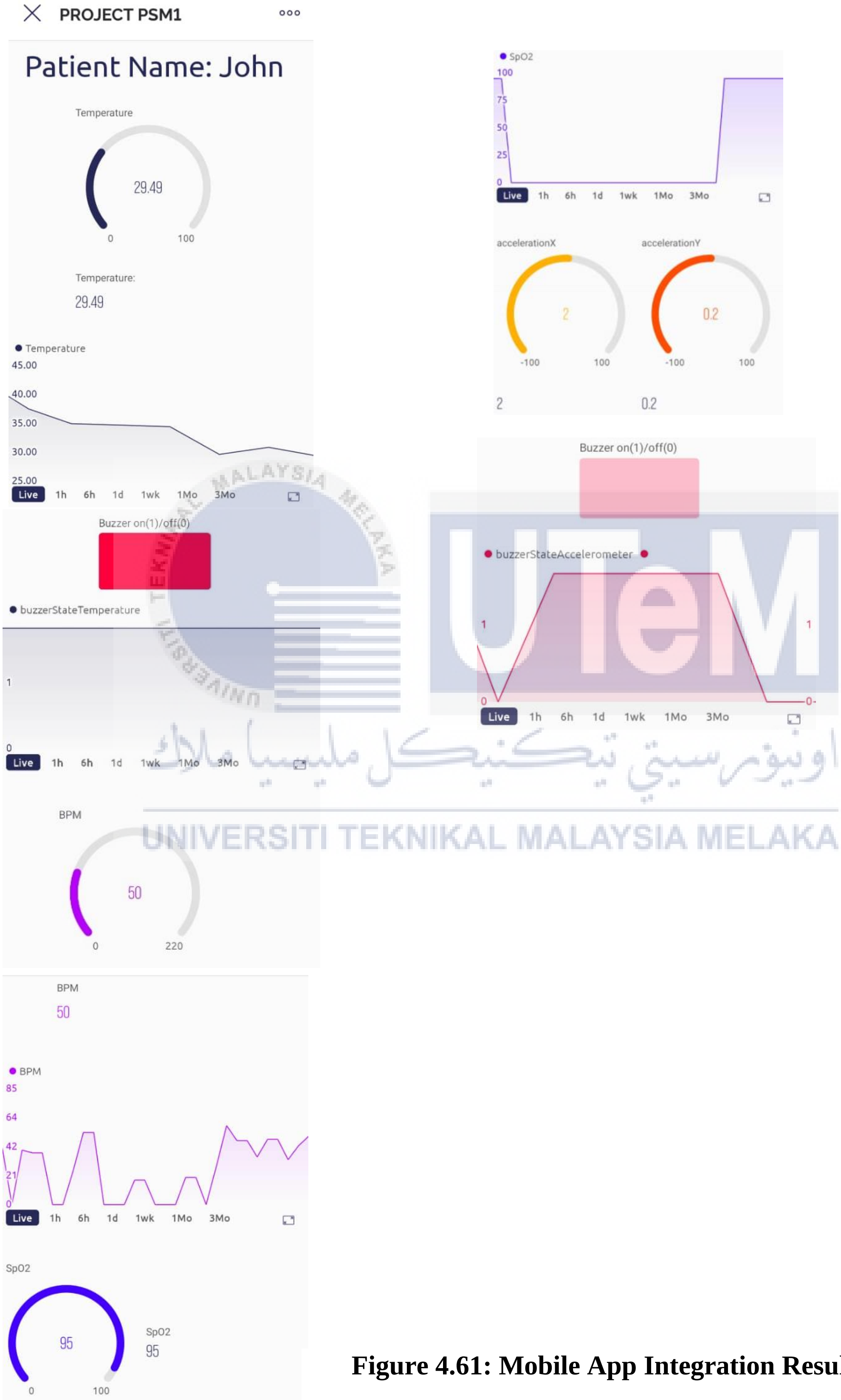


Figure 4.60: Sample of Blynk Notification

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.6 Mobile App Integration Results



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.61: Mobile App Integration Results

The Figure 4.61 above shows the system boasts an array of robust functionalities to cater to diverse user needs. Real-time monitoring capabilities offer live updates on temperature, heart rate, and gestures, ensuring immediate awareness of current conditions. Additionally, the platform enables instant alert notifications, promptly informing users when critical thresholds related to temperature, BPM, or specific gestures are breached, ensuring swift responses to potential issues. Historical analysis is facilitated through Superchart widgets, allowing users to delve into historical data trends. This functionality aids in comprehensive long-term analysis or health tracking. Moreover, the customizable interface empowers users to configure the dashboard layout and establish personalized alerts, aligning the system precisely with their specific requirements or preferences for a tailored user experience.

The comprehensive nature of this system delivers substantial benefits to both patients and caregivers. It facilitates remote monitoring of vital health parameters, granting caregivers immediate access to real-time data. This capability empowers caregivers to take prompt action based on alerts or readings, ensuring timely interventions if any concerning changes occur. By providing insights into the patient's health status, the system offers caregivers a comprehensive view, enabling them to proactively address potential health concerns or needs. Through timely notifications and access to vital data, caregivers can make informed decisions, enhancing the quality of care and support provided to the patient.

The incorporation of these functionalities into Blynk provides a user-friendly interface, creating a robust health monitoring solution. This comprehensive approach ensures proactive alerts and continuous monitoring, ultimately contributing to the user's overall care and well-being.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.7 OLED Results

4.7.1 ESP32 OLED Result

The ESP32 OLED display presents vital information: the patient's name, temperature readings in Celsius and Fahrenheit, temperature status indicating low, optimal, or high levels, readings for the X-axis and Y-axis acceleration, and associated gesture messages, such as washroom, water, hungry, medicine, or emergency alerts.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.62: Patient name 'John'

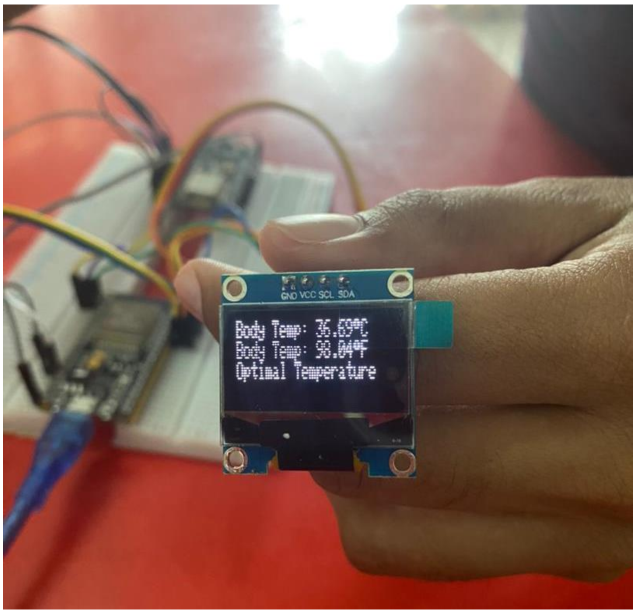
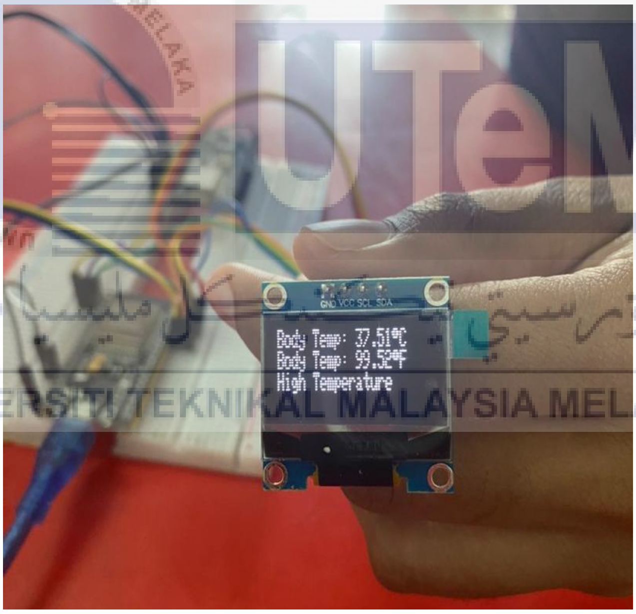


Figure 4.63: Body Temp in °C and °F (Optimal)



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.64: Body Temp in °C and °F (High)

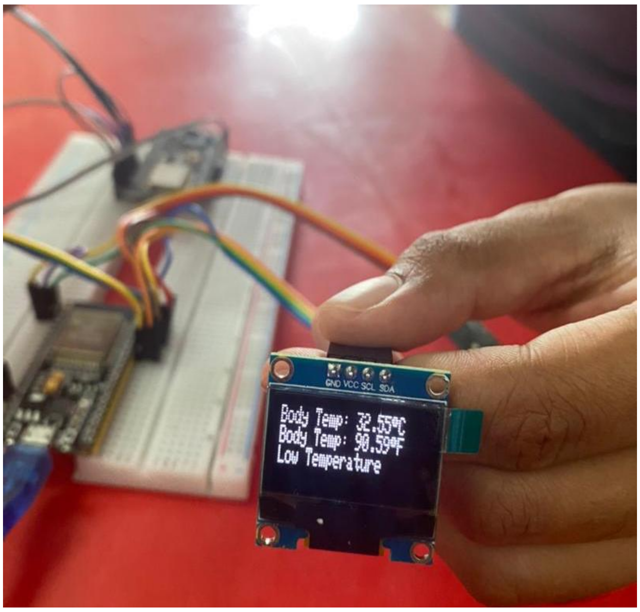


Figure 4.65: Body Temp in °C and °F (Low)

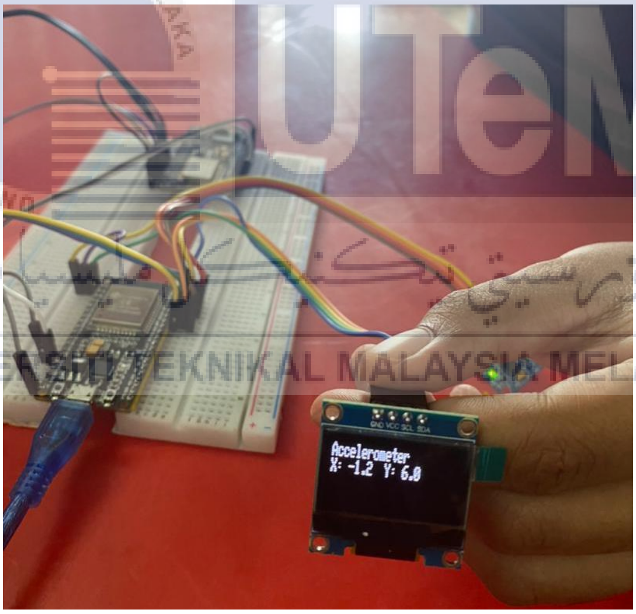


Figure 4.66: Accelerometer X-axis and Y-axis value

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

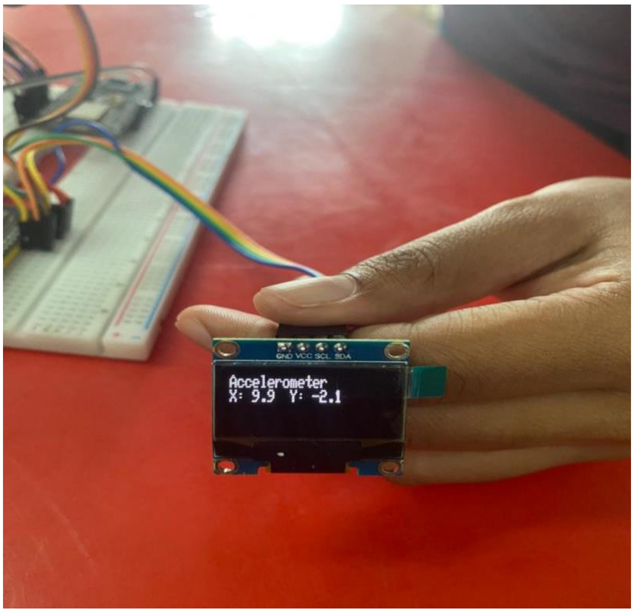


Figure 4.67: Accelerometer X-axis and Y-axis value

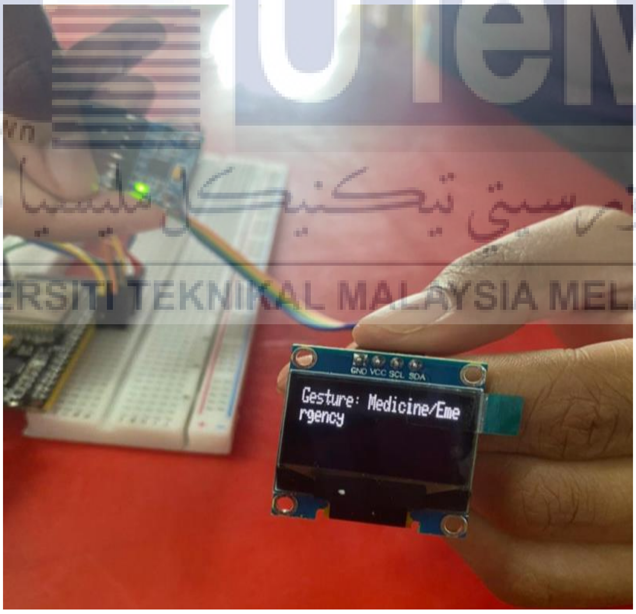


Figure 4.68: Gesture 'Medicine/Emergency'

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

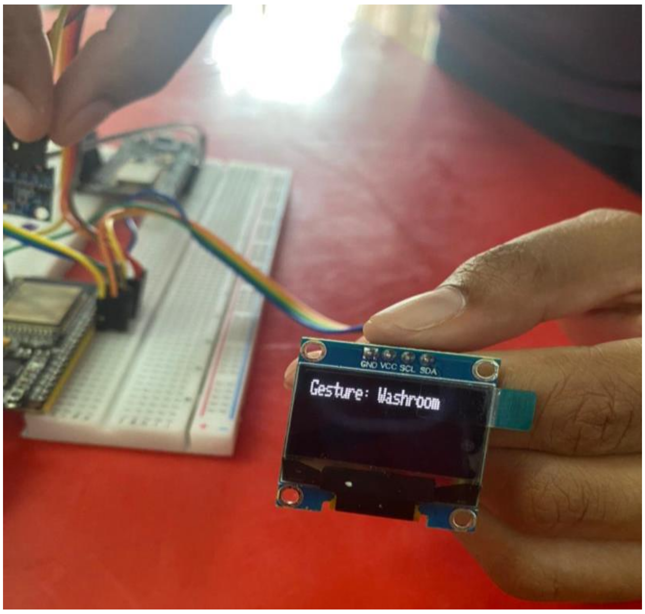


Figure 4.69: Gesture 'Washroom'

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

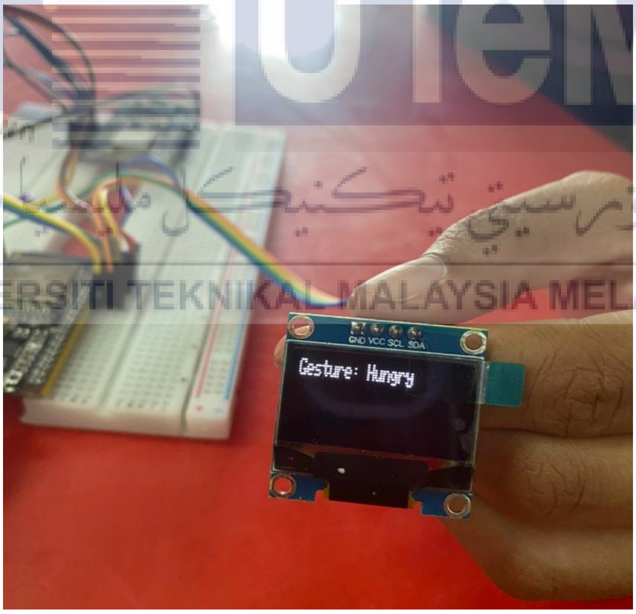


Figure 4.70: Gesture 'Hungry'

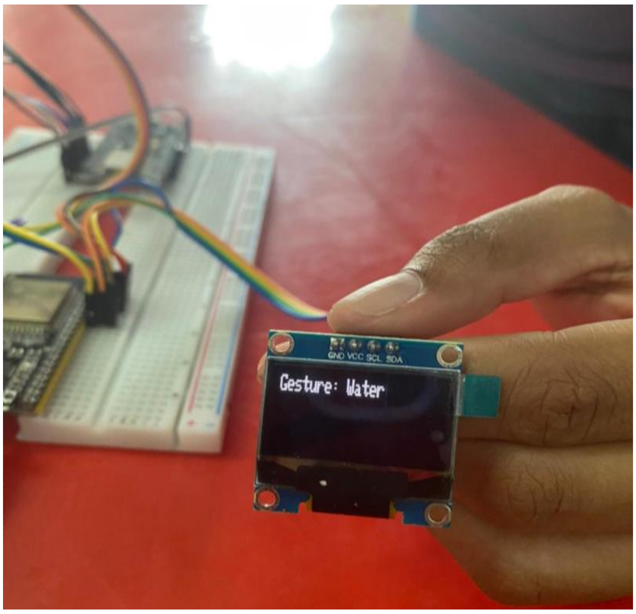


Figure 4.71: Gesture 'Water'

4.7.2 ESP8266 OLED Result

The ESP32 OLED display showcases the heart rate value in beats per minute (bpm) alongside the corresponding SpO2 value, indicating the heart rate status as high, optimal, or low.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

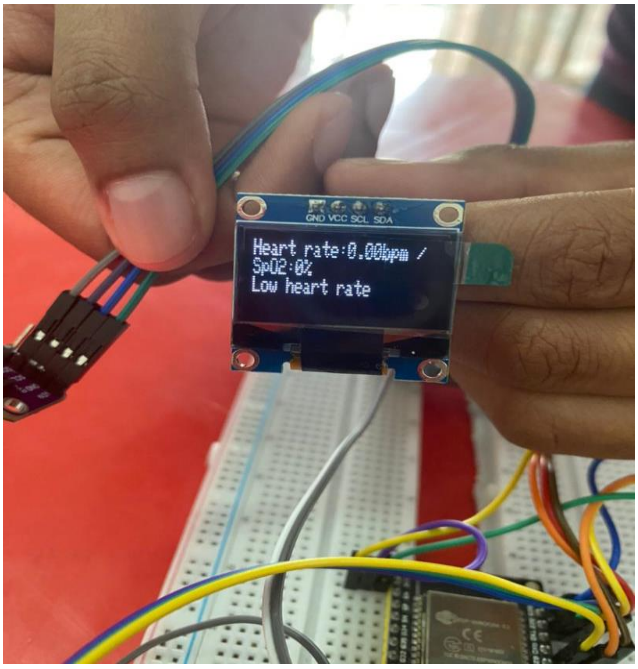


Figure 4.72: Low heart rate

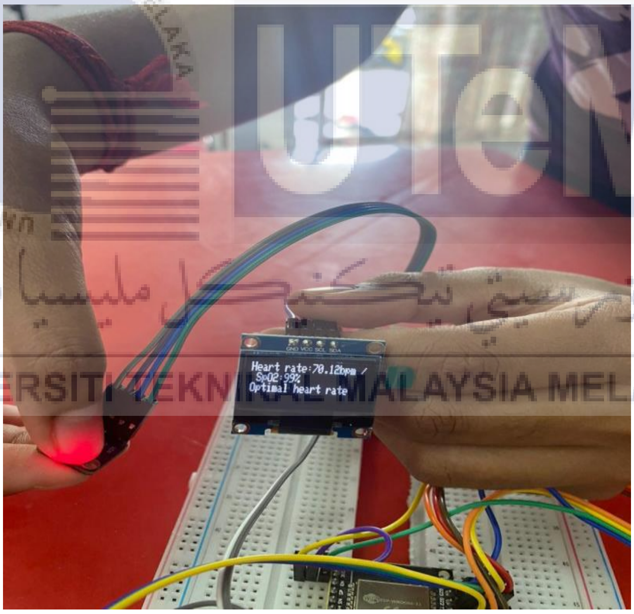


Figure 4.73: Optimal heart rate

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

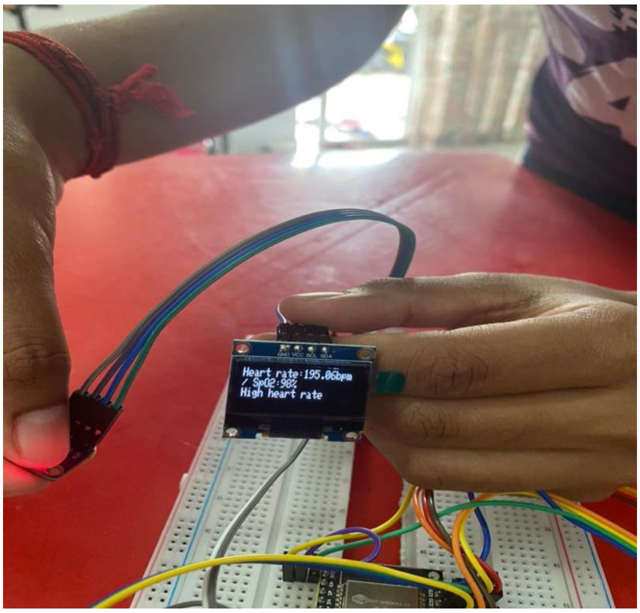


Figure 4.74: High heart rate

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.8 System Design

In a breadboard setup, the components ESP32, ESP8266, MLX90614 (temperature sensor), MPU6050 (accelerometer/gyroscope), MAX30100 (pulse oximeter), OLED display, buzzer, and green LED interconnect through a series of wiring to form a system. This arrangement integrates the sensors and modules, facilitating communication and functionality within the setup.

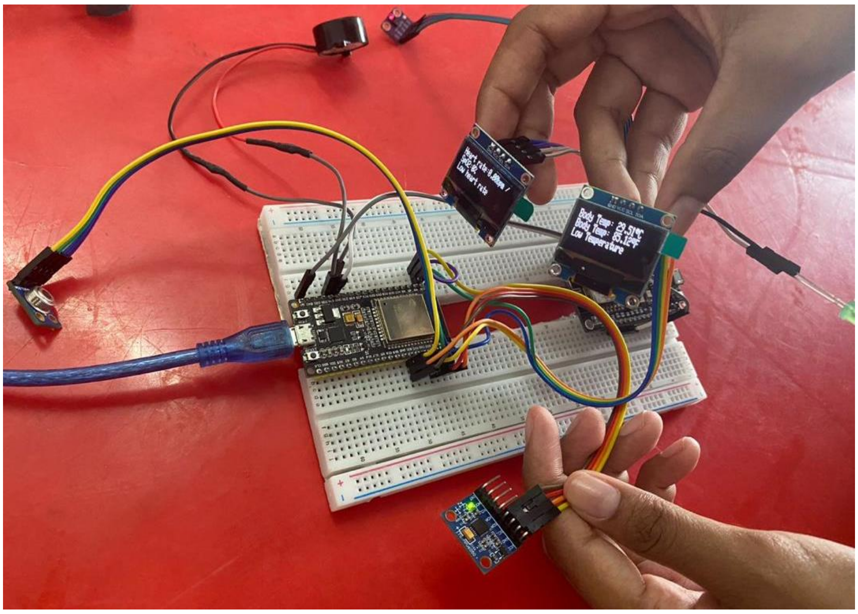


Figure 4.75: System Design 1

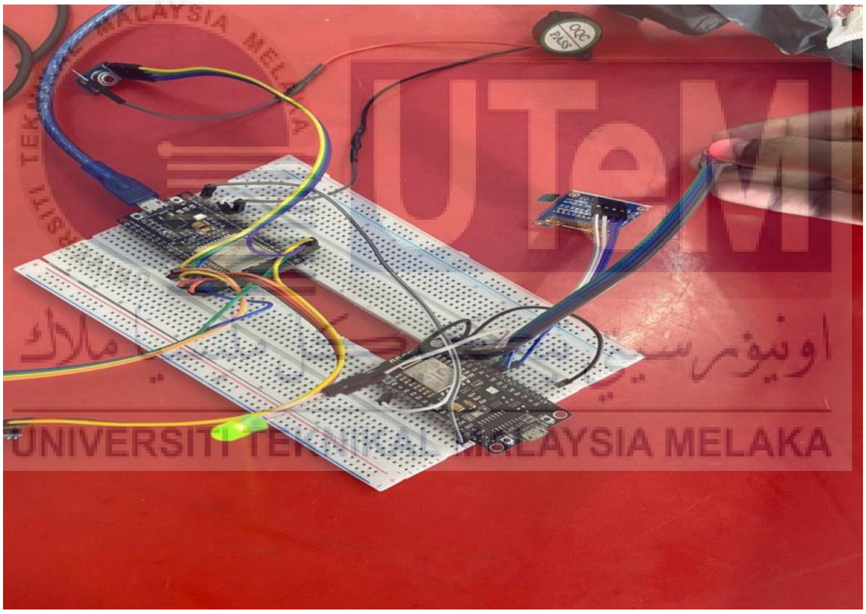


Figure 4.76: System Design 2

Integrating the breadboard connections into an electronic project box involves transitioning from the prototype stage to a more finalized setup. The components previously arranged on the breadboard will now be secured and organized within the project box.

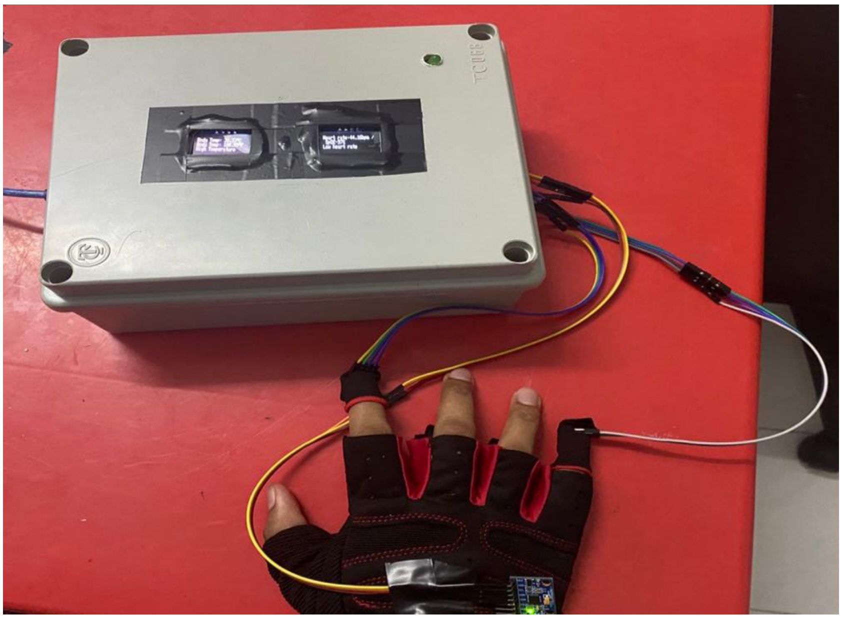


Figure 4.77: System Design 3

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.78: System Design 4

Figure 4.79 System Design 4

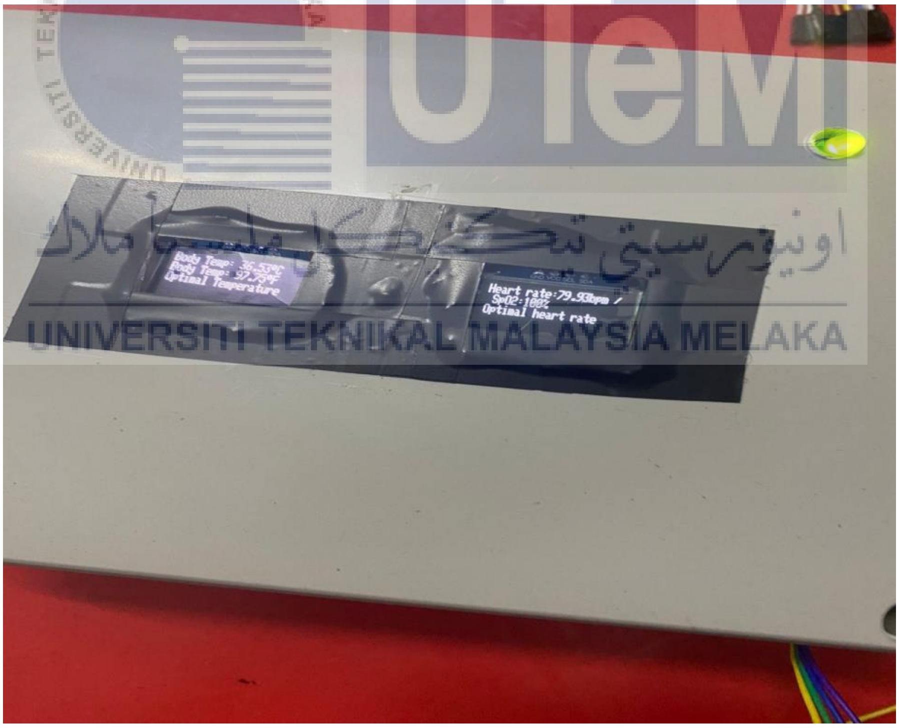


Figure 4.80: System Design 5

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.9 Result Discussion

The integration of ESP32 and ESP8266 microcontrollers alongside a suite of sensors such as the MAX30100 pulse oximeter, MLX90614 temperature sensor, and MPU6050 gyroscope and accelerometer showcased a highly efficient glove-based system. This combination facilitated seamless data capture and transmission. The synchronized operation of these components allowed for the accurate collection of pertinent information. Upon detection through the MLX90614, MAX30100, and MPU6050 sensors, the OLED screen sequentially presented critical details including the patient's name, body temperature, gesture messages, BPM, and SpO2 values, offering a user-friendly interface for real-time vital sign monitoring.

Beyond local display, this system enabled the transmission of collected sensor data encompassing body temperature, BPM/SpO2, and gesture messages over the Internet. This information was directed towards notifying caregivers promptly. Leveraging the Blynk application on a mobile phone as the receiving end, the system provided a comprehensive display of relevant data, facilitated timely notifications, and ensured convenient storage in the Blynk database for easy accessibility.

The MLX90614 temperature sensor is calibrated for room temperatures typically ranging from 28 to 30 degrees Celsius (82 to 86 degrees Fahrenheit). It provides readings for both Celsius and Fahrenheit body temperatures, accompanied by corresponding buzzer states. These buzzer states fluctuate between 1 and 0 based on specific conditions, likely linked to predefined temperature thresholds. They serve as alerts, possibly indicating temperature ranges that warrant attention.

Activation of the buzzer occurs when temperatures fall below 35°C or exceed 37.5°C. In such instances, the buzzer is triggered and set to 1. A temperature reading of 29.97 initiates "Low_Temperature_Alert" notifications, signaling low body temperature conditions. Conversely, a temperature reading of 38.09 triggers "High_Temperature_Alert" notifications. When the temperature registers at the optimal 35.89, it displays as the ideal temperature, and the buzzer is set to 0, indicating normal conditions.

The MPU6050, via its accelerometer sensor, captures various gestures characterized by distinct movements along both the x-axis and y-axis. If y exceeds 5, it will prompt a Medicine/Emergency message. When y is less than -5, it triggers a message indicating hunger. Similarly, if x is greater than 5, it initiates a water message. Conversely, when x is less than -5, it activates a message for the washroom. For example, readings like x=10.0, y=-0.9 indicate a "Water" gesture, prompting an alert with the buzzer set to state 1. This triggers the Gesture_Alert_3 notification. Similarly, movements such as x=-9.3, y=-0.1 signify the "Washroom" gesture, prompting a response with the buzzer in state 1, and triggering the Gesture_Alert_4 notification.

Other gestures like x=-4.4, y=-10.0 represent a "Hungry" gesture, setting the buzzer state to 1 and triggering the Gesture_Alert_2 notification. Conversely, x=2.4, y=9.5 denotes a "Medicine/Emergency" gesture, setting the buzzer to state 1 and initiating the Gesture_Alert_1 notification. Furthermore, minimal activity instances, such as x=-0.9, y=-0.4, prompt an indication of "No movement detected," maintaining the buzzer state at 0, signaling rest or inactivity.

The MAX30100 pulse oximeter sensor not only displays BPM/SpO2 readings but also regulates its LED based on heart rate thresholds. If the heart rate falls below 50 beats per minute or exceeds 100 beats per minute, the LED turns off. Conversely, when the heart rate falls within the optimal range of 50 to 100 beats per minute, the LED remains illuminated.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

For instance, when the heart rate is 48.81 or 102.42, outside the ideal range, the LED turns off to indicate either a low or high heart rate. Specifically, a heart rate of 48.81 triggers the 'Low_BPM_Alert', signifying a heart rate below the acceptable threshold. Meanwhile, a heart rate of 102.42 triggers the "High_BPM_Alert", indicating a heart rate above the recommended limit. However, when the heart rate is within the ideal range, such as 89.24 or 95.09, the LED stays on to signify an acceptable heart rate between 50 and 100 beats per minute.

Accuracy of MAX30100 and MLX90614 readings relies on proper finger placement. Incorrect placement leads to inaccurate sensor outputs, highlighting the need for correct hand positioning. The patient is limited to moving their hand solely in the upward, downward, leftward, and rightward directions. They are not permitted to move their hand in any other direction.

The code for the MAX30100 pulse oximeter sensor encounters challenges in incorporating delays due to the usage of multiple devices in the ESP32. To address this issue, an ESP8266 has been introduced to execute the MAX30100 code without relying on delays. The selection of the ESP8266 is based on its suitability as a single-core processor, which is well-suited for handling a single process and effectively running the MAX30100 pulse oximeter sensor code.

In summary, the integration of ESP32 and ESP8266 microcontrollers with diverse sensors ensures robust data acquisition, with OLED displays showcasing the recorded information. The system sends notifications when triggered, alerting caregivers to specific conditions. Addressing challenges like timing conflicts and communication mismatches involved segregating tasks and considering LED alternatives for better sensor performance. Emphasizing precise hand positioning for accurate sensor readings, the system showcases effective functionality through strategic solutions despite its complexities.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In this project, the development of IoT-based paralysis patient healthcare using ESP32 has been partially developed. The project was divided into three main objectives as outlined in section 1.4. The achievements for each objective are described and summarized in this chapter.

The first objective, focused on crafting an affordable wearable device for paralyzed patients, has been successfully achieved. The methodology presented a comprehensive framework outlining the selection of components known for their cost-effectiveness, ensuring accessibility for paralyzed individuals. The wearable glove design was specifically chosen for its practicality and ease of use by paralyzed patients, marking the successful accomplishment of the first objective.

The second objective aimed at creating a paralysis healthcare system using IoT and ESP32 for communication has been accomplished. This system effectively reads patient vital signs such as body temperature, BPM, and SpO2. Furthermore, it interprets the patient's hand gestures (up, down, left, right) to display pre-coded messages on the OLED, enabling patients to communicate their needs effectively. This system significantly improves communication capabilities for paralysis patients, addressing their challenges in conveying requests to others.

The third objective, concerning the development of a mobile application for monitoring patient temperature, BPM, and SpO2, has also been successfully achieved. A dedicated

Blynk application was created, facilitating real-time monitoring of the patient's body temperature, BPM, SpO2, as well as the x and y-axis accelerations. The application allows visualization of live and historical data, offering graphs for various timeframes (1 hour, 6 hours, 1 day, 1 week, 1 month, and 3 months). Additionally, the system can generate CSV files, providing structured data for body temperature, BPM, and SpO2, ensuring comprehensive monitoring capabilities.

In conclusion, significant strides have been made in achieving the project objectives, resulting in a robust IoT-based healthcare system catering to the specific needs of paralysis patients in affordable price, facilitating communication, and enabling comprehensive health metric monitoring through a user-friendly mobile application.

5.2 Future Works

There are several potential future directions for the development of IoT-based paralysis patient healthcare using ESP32. The first and foremost is the expansion of sensor capabilities by considering integrating additional sensors to capture a wider range of health parameters. This could include sensors for monitoring respiratory rate, blood oxygen saturation, or body posture. Expanding the sensor capabilities would provide a more comprehensive view of the patient's health status.

Integrating advanced sensors can enhance the system's capabilities that can be used by all paralysis patients without any physical limitation. Consider incorporating sensors that can provide more detailed and specific health data, such as heart rate variability, temperature monitoring, or muscle activity. This additional information can contribute to a more comprehensive understanding of the patient's condition. Expanding remote monitoring capabilities can further improve patient care. Implement features that allow healthcare professionals to access real-time data remotely, enabling them to make informed decisions and adjustments to the patient's care plan without the need for in-person visits.

Continuously work on improving the user interface and experience of the mobile app. Ensure that the app is intuitive, easy to navigate, and accessible to users with varying levels of technological proficiency. Solicit feedback from both healthcare providers and patients to identify areas for improvement. Given the sensitive nature of healthcare data, prioritize robust security measures to protect patient information. Implement end-to-end encryption, secure data transmission protocols, and strict access controls to ensure the confidentiality and integrity of patient data.

Moreover, plan for scalability to accommodate a growing number of users. As the system gains popularity and more patients adopt the technology, it should be able to handle increased data traffic and user demands without compromising performance. Establish a feedback loop with both healthcare professionals and patients to gather insights on the system's performance and identify areas for improvement. Regular updates and enhancements based on user feedback will contribute to the long-term success of the project.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

In conclusion, the future of IoT-based paralysis patient healthcare using ESP32 holds promise with planned expansions in sensor capabilities, including advanced features like respiratory rate and muscle activity monitoring. Remote accessibility, user-friendly interfaces, and robust security measures are focal points. Continuous feedback and scalability planning ensure adaptability to a growing user base, reflecting a commitment to innovation and improving patient care and communication.

5.3 Project Commercialization

The proposed system is suitable to be used by paralysis patients at their homes as a personal health monitoring system to monitor their body's vital measurements. Since the equipment is only available in hospitals which are very large and expensive machines, this system is designed to perform similar functions as the equipment used in hospitals. It can accurately monitor body temperature and BPM/SpO2, allowing patients to track their vital signs over time. Additionally, it can detect patient motion and display pre-coded messages to fulfill their specific needs. Unlike large and expensive hospital equipment, the proposed system utilizes cost-effective components such as the ESP microcontrollers and sensors. This makes it more affordable and accessible, particularly for patients who may have limited financial resources. It's not just a health monitoring system; it's a two-in-one solution that combines health monitoring with hand gesture recognition. The system enables communication between the patient and their caregiver. By incorporating features such as displaying pre-coded messages, the system allows patients to express their needs and communicate with their caregivers, enhancing their overall care and well-being. Users can benefit from a seamless and intuitive experience, allowing them to not only monitor their health but also interact with the system using hand gestures, potentially improving overall user engagement and satisfaction. The patient data stored in the Blynk Cloud guarantees security and privacy by employing authentication through the implementation of an auth token in the code. This ensures that only authorized individuals have access to the data. By facilitating remote monitoring and communication, the proposed system contributes to the overall quality of life for paralysis patients. It offers a sense of independence and empowerment, as patients can actively participate in their own healthcare management and receive timely assistance when needed.

اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

REFERENCES

[1] Jadhav, P. A., Jadhav, S. B., Erudkar, A. P., Bhurke, S. A., & Palekar, N. S. (2021). An IOT based approach in paralysis patient healthcare system. *International Journal of Engineering Applied Sciences and Technology*, *6*(3). https://doi.org/10.33564/ijeast.2021.v06i03.038

[2] Naveena, N., Kirubanandan, S., Gladson, S., Suruthi, D., Preethi, B., & Nandhini, K. (2023, March 20). Designing of automated paralysis patient healthcare system. Retrieved April 17, 2023, from https://www.bohrpub.com/journals/BIJGIM/BIJGIM_20232113

[3] Al-Sawaai, I. S., Lavanya, V., & Frank, A. (2020). Safety Wheel Chair for paralysis patient. *Journal of Student Research*. https://doi.org/10.47611/jsr.vi.977

[4] Viancy, V., Wilson , J. A., & Anusha , P. (2020). Paralyzed Patient Monitoring Equipment – IoT. *International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST)*, *6*(10). https://doi.org/https://www.ijarbest.com/journal/v6i10/2059

[5] Sindagi, K., B. Patil, R., L. Mujawar, R., & M.B. Mulik. (2020). GSM based Paralysis Patient Monitoring System. *International Research Journal of Engineering and Technology (IRJET)*, *7*(6).

[6] Kate, D. M., Ashwini , W., Kajal , V., Sejal , K., Mr. Rushikesh Shrikhande, & Piyush , D. (2022). GSM based health monitoring system for paralysis patients. *International Journal of Advanced Research in Science, Communication and Technology*, *04*(04). https://doi.org/10.48175/ijarsct-3043

[7] Banka, S., Madan, I., & S.S. Saranya. (2018). Smart Healthcare Monitoring using IoT. *International Journal of Applied Engineering Research*, *13*(15).

[8] Ponlatha, D. S., Gowthaman, M., Jayabalaji, K., Karthick, A., & Karthikeyan, V. G. (2023, February). *Patient rescue and condition monitoring system using IOT*. INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY. Retrieved April 17, 2023, from https://www.ijirt.org/master/publishedpaper/IJIRT158183_PAPER.pdf

[9] Sali, S., & Dr. Parvathi C.S. (2021). Real Time Health Monitoring System Using IoT. *International Journal of Creative Research Thoughts*, *9*(4).

[10] Abed, N. J., & Hussein, E. A. (2021). Design and implementation of Real Time Health Care Monitoring System based on IOT. *Journal of Physics: Conference Series*, *1818*(1), 012044. https://doi.org/10.1088/1742-6596/1818/1/012044

[11] Gaikar, D., Porlekar, P., Shetty, D., Shitkar, A., & Kalebere, P. K. (2021, August 8). *Automated paralysis patient healthcare system - IJCRT*. International Journal of Creative Research Thoughts. Retrieved April 17, 2023, from https://ijcrt.org/papers/IJCRT2108139.pdf

[12] Prof.Prasanna , G., Tanuja , K., & M Nethravathi. (2018). Health Monitoring System using IOT and Raspberry Pi. *International Journal of Computing, Communications and Networking*, *7*(2), 188–192. https://doi.org/10.30534/ijccn/2018/34722018

[13] Begum, R. V., & Dharmarajan, K. (2020). Smart Healthcare Monitoring System In IoT. *European Journal of Molecular & Clinical Medicine*, *7*(4), 2647–2661.

[14] Vidya , S., Rappai, K. R. A., Thomas, V., Dubey, A., & Shukla, S. (2021, May). *Automated paralysis patient healthcare system - IJCRT*. International Research Journal

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

of Engineering and Technology (IRJET). Retrieved April 17, 2023, from https://ijcrt.org/papers/IJCRT2108139.pdf

[15] Mohana, M., Priyadharshini, S., Sowmiya, N., & Pavithra Devi, G. (2020). An IoT Based Automated Communication System for Paralyzed Patients using Simple Hand Gestures. *European Journal of Molecular & Clinical Medicine* , *7*(4), 2681–2686.

[16] Krishna Rao, P. V., Niharika, V., Prashanthi, V., Akhila, V., & Gayathri, V. (2019). Hand Gesture Recognition and Voice Conversion System for Dumb and Deaf People. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *6*(4), 152–158.

[17] Josephine Julina Josepha, K., & Thangaswamy, S. S. (2021). Recognition of Hand Signs Based on Geometrical Features using Machine Learning and Deep Learning Approaches.

[18] Wachs, J., Stern, H., Edan, Y., Gillam, M., Feied, C., Smith, M., & Handler, J. (2020). *A Real-Time Hand Gesture Interface for Medical Visualization Applications*.

[19] Muthumeena, K., Santhiya, S., Sri Yamuna, V., & Guhan, T. (2020). *Hand Gesture Recognition For Patients Monitoring*. 326–331.

[20] Garcia, L., Tomas, J., Parra, L., & Lloret, J. (2018). An M-health Application for Cerebral Stroke Detection and monitoring using Cloud Services. *International Journal of Information Management.*, *4*(5), 319–327.

[21] Joshi Manisha, S., Amogh, B., Arpitha, K., Rakshith Shetty, K., & Yogeswarr, S. (2022). Gesture Based Monitoring System for Partially Paralyzed Patients. *International Journal of Engineering Applied Sciences and Technology*, *6*(11), 88–94.

[22] Godavari, K., Rajesh, K., Dhanalakshmi, G., & Subrahmanyam, J. (2022). IoT Paralysis Patient Health Care. *Journal or Emerging Technologies and Innovative Research(JETIR)*, *9*(12), b488–b494.

اونيۆرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[23] Raad, M., Deriche, M., bin Haffedh, A., Almasawa, H., bin Jofan, K., Alsakkaf, H., Bahumran, A., & Salem, M. (2019). *An IOT based Wearable Smart Glove for Remote Monitoring of Rheumatoid Arthritis Patients* . Science and Technology Publications (SCITEPRESS)

[24] Urshlila, R., Talib, A., Rahul, V., Akash, R., & Shivani, S. (2019). An IoT based smart glove. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *6*(5), 486–493.

[25] Faisal, F., & Hossain, S. A. (2019, August 1). IoT based remote medical diagnosis system using NodeMCU. *2019 13th International Conference on Software, Knowledge, Information Management and Applications, SKIMA 2019*. https://doi.org/10.1109/SKIMA47702.2019.8982509

[26] Julius, A., & Jian-Min, Z. (2019). IoT Based Patient Health Monitoring System Using LabVIEW and Wireless Sensor Network. *International Journal of Science and Research*, *6*, 2319–7064. https://doi.org/10.21275/ART20171643

[27] Senthil Kumar, R., Leninpugalhanthi, P., Rathika, S., Rithika, G., & Sandhya, S. (2021). Implementation of IoT Based Smart Assistance Gloves for Disabled People. *2021 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021*, 1160–1164. https://doi.org/10.1109/ICACCS51430.2021.9441855

[28] Kumar. R, H., & Padmaja, K. (2021). IoT based Paralyzed Patient Health and Body Movement Monitoring System. *International Journal for Research in Applied Science and Engineering Technology*, *9*(VI), 4495–4500. https://doi.org/10.22214/ijraset.2021.35721

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[29] Kad, S., Joshi, A., Bajgude, S., & Naiknawre, D. (2022). IOT based Paralysis Patient Healthcare. *International Journal for Research in Applied Science and Engineering Technology*, *10*(5), 3628–3633. https://doi.org/10.22214/ijraset.2022.43182

[30] Lee, B. G., & Lee, S. M. (2018). Smart Wearable Hand Device for Sign Language Interpretation System with Sensors Fusion. *IEEE Sensors Journal*, *18*(3), 1224–1232. https://doi.org/10.1109/JSEN.2017.2779466

[31] Bhavana, M., Geethika, D., Alekhya, S., Gompa, H., & Anusha, V. (2019). *Smart glove for hand gesture recognition*. *5*(2), 963–966.

[32] Abinayaa, R., Gayathiri, G. U., & Kaveri, G. (2021). Speaking System for Speech Impaired People. *International Journal for Research in Applied Science and Engineering Technology*, *9*(1), 172–177. https://doi.org/10.22214/ijraset.2021.32742

[33] Oracle. (2020). *What is the Internet of Things (IoT)?* Oracle.com. https://www.oracle.com/my/internet-of-things/what-is-iot/

[34] Ordr. (2023). *10 internet of things (IoT) healthcare examples*. Ordr. https://ordr.net/article/iot-healthcare-examples/

[35] *0.96" OLED Display (Blue, I^2C, 4-Pin)*. (n.d.). Www.elektor.com. Retrieved November 29, 2023, from https://www.elektor.com/0-96-OLED-display-blue-i2c-4-pin

[36] *Arduino Guide for MPU-6050 Accelerometer and Gyroscope | Random Nerd Tutorials*. (2021, February 16). https://randomnerdtutorials.com/arduino-mpu-6050-accelerometer-gyroscope/#:~:text=The%20MPU%2D6050%20IMU%20(Inertial

[37] *MLX90614 Non-Contact Infrared Temperature Sensor (GY-906 IR)*. (n.d.). Maker Portal. https://makersportal.com/shop/mlx90614-non-contact-temperature-sensor-infrared

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[38] *ESP32*. (n.d.). WeMakeIoT. https://www.wemakeiot.com/iot-technology-solutions/esp-32/#:~:text=The%20ESP32%20microcontroller%20enables%20devices

[39] *Paralysis vs Paresis: What's The Difference?* (2023, March 26). Https://Www.Pmrihomeo.Com/Paralysis-vs-Paresis/.

[40] *Types of paralysis - MEDizzy*. (n.d.). Types of Paralysis - MEDizzy. Retrieved June 18, 2023, from https://medizzy.com/feed/25918444

[41] Fraiwan, L., Basmaji, T., & Hassanin, O. (2018). A Mobile Mental Health Monitoring System: A Smart Glove. *2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 235–240. https://doi.org/10.1109/SITIS.2018.00043

[42] *What is the Internet of Things (IoT)?* (2021, November 23). Https://Www.Globalsign.Com/En-Sg/Blog/What-Internet-Things-and-How-Does-It-Work.

[43] *MAX30100 Heart Rate Oxygen Pulse Sensor Pinout, features, datasheet, working, applications*. (n.d.). Components101. https://components101.com/sensors/max30100-heart-rate-oxygen-pulse-sensor-pinout-features-datasheet

[44] Merriam, R. (2016, June 6). *Sending Data using Cheap RF Modules*. Https://Hackaday.Com/2016/06/06/Sending-Data-Using-Cheap-Rf-Modules/.

[45] *What is a Buzzer : Working & Its Applications*. (n.d.). Https://Www.Elprocus.Com/Buzzer-Working-Applications/.

[46] Youngblood, J. (2017, March 1). *Blynk IoT Cloud & App Quick Start Guide*. Https://Ncd.Io/Blog/Blynk-Iot-Cloud-App-Quick-Start-Guide/.

[47] *Installing ESP32 Board in the Arduino IDE*. (n.d.). Https://Lastminuteengineers.Com/Esp32-Arduino-Ide-Tutorial/.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

[48]*Installing ESP32 Board in the Arduino IDE*. (n.d.). Https://Lastminuteengineers.Com/Esp32-Arduino-Ide-Tutorial/.

[49] Notes, E. (n.d.). *Understanding LED Specifications & Characteristics » Electronics Notes*. https://www.electronics-notes.com/articles/electronic_components/diode/light-emitting-diode-led-datasheet-specifications-parameters-characteristics.php

[50] *THE 17 GOALS | Sustainable Development*. (n.d.). https://sdgs.un.org/goals

[51] I. (2017, September 30). *Getting Started With the ESP8266 ESP-01*. Instructables. https://www.instructables.com/Getting-Started-With-the-ESP8266-ESP-01/

[52]*Table 7 : Normal range of a resting heart rate*. (n.d.). ResearchGate. https://www.researchgate.net/figure/Normal-range-of-a-resting-heart-rate_tbl3_228733372

[53]*Normal body temperature and its variations*. (n.d.). https://notes.medicosnotes.com/2016/05/normal-body-temperature-and-its.html

اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPENDICES

Appendix A: ESP32 CODE

ESP32 Code

```
#include <Adafruit_MLX90614.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_SSD1306.h>
#include <Wire.h>
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL6JEVBCWO4"
#define BLYNK_TEMPLATE_NAME "PROJECT PSM1"
#define BLYNK_AUTH_TOKEN "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn"
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
BlynkTimer timer;

char auth[] = "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn";
char ssid[] = "HUAWEI nova 3i";
char pass[] = "sggayathiri";

Adafruit_MLX90614 mlx = Adafruit_MLX90614();
Adafruit_MPU6050 mpu;
Adafruit_SSD1306 OLED(128, 32, &Wire);

const int BUZZER_PIN = 13;
bool isBuzzerActive = false; // Declare isBuzzerActive in the global scope
// Define variables to track buzzer states
int buzzerStateTemperature = 0;
int buzzerStateAccelerometer = 0;

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
void setup()
{
  Serial.begin(115200);
  while (!Serial);

  pinMode(BUZZER_PIN, OUTPUT);

  if (!mlx.begin())
  {
    Serial.println("Error connecting to MLX sensor. Check wiring.");
    while (1);
  }

  if (!mpu.begin())
  {
    Serial.println("Sensor init failed");
    while (1)
    yield();
  }

  if (!OLED.begin(SSD1306_SWITCHCAPVCC, 0x3C))
  {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;)
      ;
  }

  OLED.display();
  delay(500);
  OLED.setTextSize(1);
  OLED.setTextColor(SSD1306_WHITE);

  OLED.clearDisplay();
  OLED.display();

  Blynk.begin("gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn", ssid,pass);
  Serial.println("");
  delay(100);
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
void loop()
{
  String patientName = "John";
  Serial.print("Patient name: ");
  Serial.println(patientName);
  OLED.clearDisplay();
  OLED.setCursor(0, 0);
  OLED.print("Patient name: ");
  OLED.println(patientName); // Print the patient name on the next line
  OLED.display(); // Update the OLED display
  delay(1000);
  Serial.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~");
  Blynk.run();
  temperatureLoop();
  accelerometerGyroscopeLoop();
}

void temperatureLoop()
{
  String tempState = "Optimal"; // Default state is Optimal
  float objectTemperature = 0.0;
  Serial.print("Body temperature = ");
  float objectTempC = mlx.readObjectTempC();
  Serial.print(objectTempC);
  Serial.println("°C");

  Serial.print("Body temperature = ");
  float objectTempF = mlx.readObjectTempF();
  Serial.print(objectTempF);
  Serial.println("°F");

  if (objectTempC >= 35 && objectTempC <= 37.5) {
    noTone(BUZZER_PIN);
    isBuzzerActive = false;
    buzzerStateTemperature = 0;
    Serial.println("Optimal Temperature");
    tempState = "Optimal Temperature";
  }
  else if (objectTempC < 35)  {
    playBuzzerSound();
    buzzerStateTemperature = 1; // Update buzzer state for temperature
    Serial.println("Low Temperature");
    tempState = "Low Temperature";
  }
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
 else if (objectTempC > 37.5)  {
    playBuzzerSound();
    buzzerStateTemperature = 1; // Update buzzer state for temperature
    Serial.println("High Temperature");
    tempState = "High Temperature";
  }

  // Print the buzzer state
  Serial.print("Buzzer State (Temperature): ");
  Serial.println(buzzerStateTemperature);

  //blynk notification
  if (objectTempC > 37.5) {
    Blynk.logEvent("high_temperature_alert","High Body Temperature");
  }
  else if (objectTempC < 35) {
    Blynk.logEvent("low_temperature_alert","Low Body Temperature");
  }
  objectTemperature = objectTempC;
  Blynk.virtualWrite(V0, objectTemperature);
  Blynk.virtualWrite(V6, buzzerStateTemperature); // Send buzzer state
for temperature to V6

  OLED.clearDisplay();
  // Set the cursor position for temperature
  OLED.setCursor(0, 0);
  OLED.print("Body Temp: ");
  OLED.print(objectTempC, 2);
  OLED.print((char)247);   // Degree symbol
  OLED.print("C");
  OLED.println(" ");
  OLED.print("Body Temp: ");
  OLED.print(objectTempF, 2);
  OLED.print((char)247);   // Degree symbol
  OLED.print("F");
  OLED.println(" ");
  OLED.println(tempState);
  OLED.display();
  Serial.println("-------------------------------------------------------
--");
  delay(1000);
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
 void accelerometerGyroscopeLoop()
{
  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  float accelerationX = a.acceleration.x;
  float accelerationY = a.acceleration.y;
  OLED.clearDisplay();

  OLED.setCursor(0, 0);

  Serial.print("Accelerometer ");
  Serial.print("X: ");
  Serial.print(a.acceleration.x, 1);
  Serial.print(" m/s^2, ");
  //oled x
  Serial.print("Y: ");
  Serial.print(a.acceleration.y, 1);
  Serial.print(" m/s^2, ");
  //oled y
  Serial.print("Z: ");
  Serial.print(a.acceleration.z, 1);
  Serial.println(" m/s^2");

  String message;
  if (a.acceleration.y > 5)
  {
    message = "Medicine/Emergency";
    Blynk.logEvent("gesture_alert_1","Medicine/Emergency");
  } else if (a.acceleration.y < -5)
  {
    message = "Hungry";
    Blynk.logEvent("gesture_alert_2","Hungry");
  } else if (a.acceleration.x > 5)
  {
    message = "Water";
    Blynk.logEvent("gesture_alert_3","Water");
  } else if (a.acceleration.x < -5)
  {
    message = "Washroom";
    Blynk.logEvent("gesture_alert_4","Washroom");
  } else
  {
    message = "No movement detected";
  }
```

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
  Serial.println("Gesture:" + message);

  OLED.clearDisplay();
  OLED.setCursor(0, 0);
  OLED.println("Accelerometer ");
  OLED.print("X: ");
  OLED.print(a.acceleration.x, 1);
  OLED.print("  ");
  OLED.print("Y: ");
  OLED.println(a.acceleration.y, 1);
  OLED.display();
  delay(500);

  OLED.clearDisplay();
  OLED.setCursor(0, 0); // Move cursor to the top-left corner
  OLED.println("Gesture: " + message); // Print Gesture message
  OLED.display();


  if (message != "No movement detected") {
    playBuzzerSound();
    buzzerStateAccelerometer = 1; // Update buzzer state for
accelerometer
  } else {
    noTone(BUZZER_PIN);
    isBuzzerActive = false;
    buzzerStateAccelerometer = 0; // Update buzzer state for
accelerometer
  }
// Print the buzzer state
  Serial.print("Buzzer State (Accelerometer): ");
  Serial.println(buzzerStateAccelerometer);

  Serial.print("Gyroscope ");
  Serial.print("X: ");
  Serial.print(g.gyro.x, 1);
  Serial.print(" rps, ");
  Serial.print("Y: ");
  Serial.print(g.gyro.y, 1);
  Serial.print(" rps, ");
  Serial.print("Z: ");
  Serial.print(g.gyro.z, 1);
  Serial.println(" rps");
  Serial.println("*******************************************************
** ");
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
  Blynk.virtualWrite(V2,accelerationX);
  Blynk.virtualWrite(V3,accelerationY);
  Blynk.virtualWrite(V7, buzzerStateAccelerometer); // Send buzzer state
for accelerometer to V7

  OLED.display();
  delay(1000);
}

void playBuzzerSound()
{
  tone(BUZZER_PIN, 1000, 500);
  delay(500);
  noTone(BUZZER_PIN);
  isBuzzerActive = true; // Update the buzzer state variable
}
```

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix B ESP8266 CODE

ESP8266 Code

```
#include <Wire.h>
#define BLYNK_PRINT Serial
#define BLYNK_TEMPLATE_ID "TMPL6JEVBCWO4"
#define BLYNK_TEMPLATE_NAME "PROJECT PSM1"
#define BLYNK_AUTH_TOKEN "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn"
#include <SPI.h>
#include <Ethernet.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "MAX30100_PulseOximeter.h"
#include <Adafruit_SSD1306.h>
BlynkTimer timer;
Adafruit_SSD1306 OLED(128, 32, &Wire);
#define REPORTING_PERIOD_MS 1000
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1
char auth[] = "gvu7EMIfmnpjN6LyKSjQlCwaKg-OkpJn";
char ssid[] = "HUAWEI nova 3i";
char pass[] = "sggayathiri";
const int GREEN = 16;     // for Green LED
PulseOximeter pox;
uint32_t tsLastReport = 0;
void onBeatDetected()
{
Serial.println("Beat!");
}
void setup()
{
Serial.begin(9600);
pinMode(GREEN, OUTPUT);
Blynk.begin(auth, ssid, pass);
Serial.print("Initializing pulse oximeter..");
if (!pox.begin())
{
Serial.println("FAILED");
for (;;)
;
}
else{
Serial.println("SUCCESS");
}
```

```
pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
pox.setOnBeatDetectedCallback(onBeatDetected);
if (!OLED.begin(SSD1306_SWITCHCAPVCC, 0x3C))
{
Serial.println(F("SSD1306 allocation failed"));
for (;;)
;
}
OLED.display();
OLED.setTextSize(1);
OLED.setTextColor(SSD1306_WHITE);
OLED.clearDisplay();
OLED.display();
}
void loop()
{
Blynk.run();
pox.update();
if (millis() - tsLastReport > REPORTING_PERIOD_MS)
{
Serial.print("Heart rate:");
Serial.print(pox.getHeartRate());
Serial.print("bpm / SpO2:");
Serial.print(pox.getSpO2());
Serial.println("%");
OLED.clearDisplay();
OLED.setCursor(0, 0);
OLED.print("Heart rate:");
OLED.print(pox.getHeartRate());
OLED.print("bpm / SpO2:");
OLED.print(pox.getSpO2());
OLED.println("%");
OLED.display();
int rate = pox.getHeartRate();
int sp = pox.getSpO2();
tsLastReport = millis();

if (rate >= 50 && rate <= 100 )
{
digitalWrite(GREEN, HIGH);
Serial.println("Optimal heart rate");
OLED.println("Optimal heart rate");
OLED.display();
Serial.println("LED STATUS: 1");
}
```

```
else if (rate <50)
{
digitalWrite(GREEN, LOW);
Serial.println("Low heart rate");
OLED.println("Low heart rate");
OLED.display();
Serial.println("LED STATUS: 0");
}
else if (rate >100)
{
digitalWrite(GREEN, LOW);
Serial.println("High heart rate");
OLED.println("High heart rate");
OLED.display();
Serial.println("LED STATUS: 0");
}
Blynk.virtualWrite(V1, rate);
Blynk.virtualWrite(V5, sp);

if (rate > 100)
{
Blynk.logEvent("high_bpm_alert", "High BPM");
}
else if (rate <50)
{
Blynk.logEvent("low_bpm_alert", "Low BPM");
}
}
}
```

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Appendix C Normal Range of Resting Heart Rate

Age or fitness level	*Beats per minute (bpm)*
Babies to age 1:	100–160
Children ages 1 to 10:	60–140
Children age 10+ and adults:	60–100
Well-conditioned athletes:	40–60

Appendix D Range of Body Temperature

	Degree Celsius	Degree Fahrenheit
• Normal temperature	36.6-37.2	98-99
• Febrile	> 37.2	>99
• Hyperpyrexia	> 41.6	>107
• Subnormal	< 36.6	< 98
• Hypothermia	< 35	< 95

MedicosNotes.com

	Measurement site		
	Mouth / armpit	Ear / forehead	Rectum
Low temperature	< 35.8	< 35.7	< 36.2
Normal temperature	35.9 - 37.0	35.8 - 36.9	36.3 - 37.5
Increased temperature	37.1 - 37.5	37.0 - 37.5	37.6 - 38.0
Light fever	37.6 - 38.0	37.6 - 38.0	38.1 - 38.5
Moderate fever	38.1 - 38.5	38.1 - 38.5	38.6 - 39.0
High fever	38.6 - 39.5	38.6 - 39.4	39.1 - 39.9
Very high fever	39.6 - 42.0	39.5 - 42.0	40.0 - 42.5

MedicosNotes.com

Low temperature	Consult a doctor.
Normal temperature	You are perfectly well.
Increased temperature	You should get some rest.
Light fever	Check your temperature regularly and rest.
Moderate fever	Check your temperature regularly. Consult a doctor if you get worse or if the fever lasts for more than three days.
High fever	Consult a doctor, especially if the fever lasts for more than one day.
Very high fever	Go to emergency ward of a hospital!

PSM 2 REPORT GAYATHIRI SELVARAJU

by Gayathiri Selvaraju

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Submission date: 12-Jan-2024 11:31AM (UTC+0900)
Submission ID: 2269705722
File name: PSM_2_REPORT_removed_4.pdf (6.54M)
Word count: 24465
Character count: 137044

PSM 2 REPORT GAYATHIRI SELVARAJU

ORIGINALITY REPORT

10%	6%	5%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.semanticscholar.org Internet Source	1%
2	N. J. Abed, Ehab Abdulrazzaq Hussein. "Design and Implementation of Real Time Health Care Monitoring System Based on IoT", Journal of Physics: Conference Series, 2021 Publication	<1%
3	www.icoess.com Internet Source	<1%
4	www.scilit.net Internet Source	<1%
5	Submitted to Universiti Teknikal Malaysia Melaka Student Paper	<1%
6	www.scitepress.org Internet Source	<1%
7	www.revistaclinicapsicologica.com Internet Source	<1%

UNIVERSITI TEKNIKAL MALAYSIA MELAKA