DEVELOPMENT OF A WEARABLE FALL DETECTION SYSTEM FOR THE ELDERLY BY USING ESP32

AHMAD AFFIQ BIN MOHD ASNIZAM

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEVELOPMENT OF A WEARABLE FALL SYSTEM FOR THE ELDERLY BY USING ESP32

AHMAD AFFIQ BIN MOHD ASNIZAM

This report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Electronics Engineering Technology with Honours

Faculty of Electronics and Computer Technology and Engineering
Universiti Teknikal Malaysia Melaka

2025

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJURUTERAAN ELEKTRONIK DAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : DEVELOPMENT OF A WEARABLE FALL SYSTEM FOR THE ELDERLY BY USING ESP32

Sesi Pengajian : 2024/2025

Saya AHMAD AFFIQ BIN MOHD ASNIZAM mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

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

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

Disahkan oleh:

(TANDATANGAN PENULIS)

(COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: ....

TS. DR. NUR ALISA BINTI ALI
Pensyarah Kanan
Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer
Universiti Teknikal Malaysia Melaka

Tarikh : 26 Januari 2025

Tarikh : 26 Januari 2025

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

DECLARATION

I declare that this project report entitled "Development Of A Wearable Fall Detection System For The Elderly Using ESP32" is the result of my 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	:	AHMAD AFFIQ BIN MOHD ASNIZAM
Date	:	26/1/2025

APPROVAL

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

Signature :

Supervisor Name : Ts. Dr Nur Alisa Binti Ali

Date : 26.1.2025

Signature :

Co-Supervisor Name (if any) : Ts. Dr Abd Shukur Bin Jaafar

Date : 26.1.2025

DEDICATION

To my beloved mother, Siti Esah Binti Abdul Rashid, and father, Mohd Asnizam Bin Asari

ABSTRACT

Falls are a significant health problem, especially among the elderly, and are often the cause of serious injuries and reduced independence. This project entitled "Development of Fall Detection System for the Elderly using ESP32" involved the development of a wearable fall detection system accounting for advanced sensor technologies for fall detection that are reliable and accurate. The falling hazards to older individuals are very likely to occurs. Most falls lead to severe injuries and decrease the quality of life. This project presents MPU6050 sensors for motion detection, integrating an accelerometer and a gyroscope to monitor the movements. The data is processed in real-time using the ESP32. At the same time, another esp32 with a global positioning system (GPS) establishes its use for wireless communication and sending the location of the incident. This helps to notify the caregivers or emergency services in a short time and in the right location. In this project, elderly safety and welfare will be improved through immediate response to falls using the wearable device. This system aims to overcome the limitations of current methods of fall detection which include improvised surveillance, society and individual safety.

ABSTRAK

Jatuh merupakan masalah kesihatan yang besar, terutamanya di kalangan orang tua, dan sering menyebabkan kecederaan serius dan kurang kebebasan. Projek ini bertajuk "Pembangunan Sistem Pengesanan Jatuh untuk Orang Tua menggunakan ESP32" melibatkan pembangunan sistem pengesanan jatuh yang mengandungi teknologi sensor yang canggih untuk pengesan jatuh yang boleh dipercayai dan tepat. Daripada semua bahaya kepada individu yang lebih tua, jatuh adalah sering berlaku . Kebanyakan jatuh menyebabkan kecederaan yang teruk dan mengurangkan kualiti hidup. Projek ini memperkenalkan sensor MPU6050 untuk pengesanan pergerakan dengan integrasi akselerometer dan giroskop untuk memantau pergerakkan. Data diproses dalam masa nyata menggunakan ESP32-C3. Dalam masa yang sama satu lagi ESP32 dengan sistem penempatan global (GPS) menetapkan penggunaan untuk komunikasi tanpa wayar dan menghantar lokasi insiden. Ini membantu untuk memaklumkan penjaga atau perkhidmatan kecemasan dalam masa yang singkat dan di tempat yang betul. Dalam projek ini, keselamatan dan kesejahteraan orang tua akan ditingkatkan melalui tindak balas segera pada masa-masa jatuh menggunakan peranti berpakaian. Sistem ini bertujuan untuk mengatasi keterbatasan kaedah pengesanan jatuh semasa yang membawa kepada penambahbaikan pengawasan, masyarakat dan keselamatan individu.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Ts. Nur Alisa Binti Ali and co-supervisor for their precious guidance, words of wisdom and patience throughout this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support through the end which enabled me to accomplish the project. Not forget my colleagues for their willingness to share their thoughts and ideas regarding the project.

My highest appreciation goes to my parents, and family members for their love and prayer during the period of my study.

Finally, I would like to thank all the staff, colleagues, classmates, faculty members, and other individuals not listed here for being cooperative and helpful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

A	-	Value of accelerometer
α	-	acceleration
ADL	-	Activity daily life
G	-	Value of gyroscope
αX	-	x-axis of accelerometer
αY	-	y-axis of accelerometer
αZ	-	z-axis of accelerometer
gX	-	x-axis of gyroscope
gY	-	y-axis of gyroscope
gZ	-	z-axis of gyroscope
MAG	-	Magnitude
TP	-	Number of falls detected
TN	-	Number of falls undetected
FP	-	ADL detected as a fall event
FN	-	ADL not detected as a fall event
WHO	-	World Health Organization
RNN	-	Recurrent Neutral Network
LSTM	-	Long Short-term Memory
DSM		Degrees, minutes, seconds

LIST OF ABBREVIATIONS

m/s	-	Meter per second
V	-	voltage
mA	-	milliampere
dBm	-	Decibel meter
MB	-	Mega Bytes
KB	-	Kilo Bytes
mm	-	millimetre
Hz	-	Hertz
dB	-	Decibel
°/s	-	Degree per second
g	-	gravity

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Falls are a significant concern for elderly individuals, as they can lead to serious injuries, hospitalisation, and even death. According to the World Health Organization (WHO), falls are the second leading cause of accidental or intentional injury deaths worldwide, and the risk of falling increases with age. Elderly individuals often have compromised physical abilities, vision, and balance, making them more susceptible to falls[1]. Moreover, falls can have a profound impact on the quality of life of elderly individuals, leading to decreased mobility[2], increased dependence on caregivers, and reduced independence.

Developing a fall detection system for the elderly using Internet of Things (IoT) technology involves designing and implementing a system that detects falls and sends notifications and location information to caregivers or emergency services. The system uses an accelerometer and gyroscope to measure the device's acceleration and orientation, respectively. If a fall is detected, the microcontroller sends notification and location information to the Wi-Fi module, which connects to the internet and sends the data to caregivers or emergency services.

The system is designed to be portable and can be fixed in various spaces. It aims to provide real-time monitoring of the elderly individual's location and activities, detect falls, and send alerts to caregivers or emergency services.

1.2 Addressing Societal Issue of Development Fall Detection System for the Elderly

Falls among the older population remain a serious public health issue, with serious injuries, hospitalization, and loss of independence at stake. Around the world, the human population is aging, and this demands more effective and low-cost solutions for fall prevention. This Arduino technology-based development of a fall detection system will provide a cost-effective, efficient, and user-friendly solution for older adult safety and well-being.

As the leading cause of injury falls make a severe impact physically as well as psychologically. A fall-detecting system monitors continuously detects the falls and informs instantly for ultimate safety and independence. The injuries caused by falls also incur heavy healthcare costs; its early detection and prompt intervention reduce the intensity of the injury and thereby decrease the cost. They reassure elderly people, hence encouraging active life and social involvement, consequently, improving the quality of life of this group.

Most systems already existing are very costly and, in most cases, unavailable, particularly in low-income setups. It can be widely developed and render fall detection technology available to a broader section of the population. The digital divide could limit some elderly people from using modern health monitoring systems. Towards this end, the system has friendly interfaces with clear messages of alerts based on its ability on educational programs, which improve digitization literacy and empowerment. The rapid alert of the emergency contact provides timely medical response. This helps to better the community health outcomes.

1.3 Problem Statement

The increasing aging population and the need for dependable, cost-effective, and accessible fall-detection solutions for the elderly have prompted the development of this project. The current fall detection systems are often limited in effectiveness, accessibility, and affordability.[3], [4]. There is a need for a dependable, affordable, and user-friendly fall detection system that can be easily implemented in the homes of elderly individuals. In buildings such as health care, there will be Wi-Fi-dead zones which means the Internet is out of range in certain areas such as toilets and more.

1.4 Project Objective

The objectives of this project are:

1. To design and implement a fall detection system that can reliably and efficiently identify falls among the elderly.
2. To develop a system that can automatically alert healthcare providers or designated emergency contacts upon the detection of a fall, enabling prompt response and assistance.
3. To develop a cost-effective and user-friendly solution that can be easily adopted by the elderly and their caregivers.

1.5 Scope of Project

The scope of the project are :

1. The MPU6050 sensor, Neo-6m GPS module and two ESP32 microcontrollers selected and integrated to develop a small-sized but compact and effective fall detection system.
2. Primary functions involve the development of detecting falls in the algorithm with data furnished through the MPU6050 sensor. Real-time alert functionality shall be integrated into the system, wherein it triggers on-site indicators with remote notifications sent to on-duty caregiver's devices.
3. Two ESP32s were selected to overcome the problem of WI-FI dead zones when the elderly are in a building. The first ESP is used as a wearable device and then sends the data obtained from the MPU6050 sensor and Neo-6m GPS module to another ESP32 that sets a main microcontroller. The ESP32 sends the data as a notification to the caregiver.
4. The system is cost-effective. It uses low-cost components, and its manufacturing and maintenance are very efficient. The system aimed to be very inexpensive so it can be purchased by as many of the users as possible, even in low-income settings, through appropriate pricing and distribution.
5. An interface that is simple and intuitive with alerts designed to allow the elderly and caregivers to do so without any problem .

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The advancement of wearable technology has opened new avenues for healthcare, particularly in monitoring and improving the safety of the elderly population. As the global demographic shifts towards an aging population, the need for effective fall detection systems becomes increasingly critical. Falls are a leading cause of injury among seniors.[2], often resulting in severe physical and psychological consequences. Therefore, reliable and accurate fall detection systems are essential to enhance the quality of life and ensure the safety of elderly individuals living independently. Recently, the techniques of the Internet of Things (IoT) and mobile communications have been developed to gather human and environmental information data for a variety of intelligent services and applications.[5-7].

This chapter discussed a literature review to explore the current state of research and development in wearable fall detection systems. It seeks to provide a comprehensive overview of existing technologies, methodologies, and innovations that have been proposed and implemented in recent years. This chapter examined various approaches to fall detection, including sensor-based, machine-learning, and hybrid methods, highlighting their strengths, limitations, and practical applications.

2.2 Fall incident to the elderly

A fall is defined as the movement of a person from a higher to lower level under the force of gravity, typically suddenly, swiftly, and without control (World Health Organization (WHO), 2007). According to the WHO, falls are the second leading cause of unintentional injury deaths worldwide, with adults aged 65 and older experiencing the greatest number of fatal falls. The elderly, with a higher risk of falling, are more likely to have serious injuries from falls.[8,9]. The consequences of falls in the elderly can be severe and multifaceted. Physically, falls can lead to injuries such as fractures, particularly hip fractures, head injuries, and bruises. Figure 2.2 below demonstrates the common symptoms leading to falls for the elderly.[10].

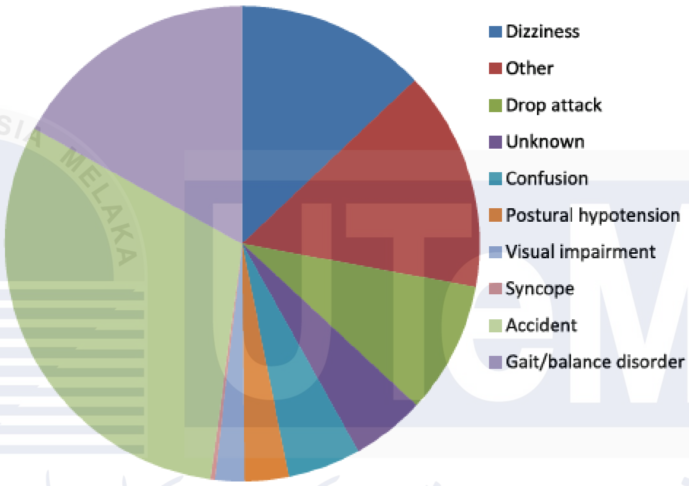


Figure 2.1 Causes of falls in the elderly[8]

Based on Figure 1 above, accidents are the most common cause of falls in the elderly while syncope is the least common cause of falls in the elderly, mostly caused by accident and the lowest caused by syncope.

2.3 Existing Fall Detection Technologies

To make things clear on how to develop a fall detector system, this section compiled the progress from other research that has been developed throughout the years. The discussion of this topic includes the existing fall detection technologies which have three types which are wearable devices, camera-based devices, and ambience devices.

Wearable devices

Wearable devices are the most common type of fall detection technology[11]. These devices are worn by the user and use sensors such as accelerometers, gyroscopes, and magnetometers to detect falls.[12]. They are often small, discreet, and easy to use, making them popular among elderly individuals who want to maintain their independence. Recently, there have been many developments in the wearable fall detection system.

Research for developing an improved wearable fall detection system that can automatically alert the caregiver in the event of an elderly fall via IoT technology that would enable faster response and medical assistance for lighter injuries[11]. This research can automatically detect falls and send alerts without user input to improve the response time. It uses a low-cost and lightweight wearable sensor (triaxial sensor, MPU6050) for easy use by the elderly and real-time IoT notifications to caregivers via email and mobile applications.[6]. Unfortunately, this system has the potential for false alarms for misclassifying intense activities such as falls. Then, the range of wireless transmission is limited due to only using WI-FI and it only detects falls at the chest.[13]. Figure 2.2 below shows the prototype design of a wearable fall devices system using MPU6050 sensor at chest.

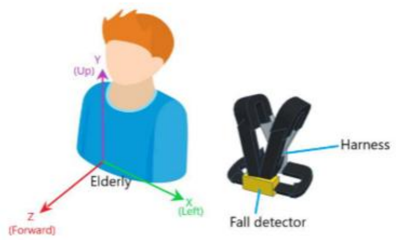


Figure 2.2 The prototype design with sensing axes of MPU6050 on the chest of the person[12]

Camera-based devices

Camera-based devices use cameras to detect falls. They are often used in conjunction with machine learning algorithms to improve accuracy.[14]. The camera-based system is one of the devices that has the most positive impact in detecting and predicting fall activities and individual routing activities. In research work, a novel deep learning algorithm is proposed to detect human falls in the indoor environment automatically using computer vision techniques.[15].

Among all fall detection methods computer vision-based methods are giving better accuracy compared to other traditional methods. They produced lower false alarm rates, and the result can be further improved by the efficient deep learning algorithm.[16]. Based on this research, they used Recurrent Neutral Network (RNN) and Long Short-Term Memory (LSTM) algorithms. The accuracy of LSTM achieved 85.79% while the Gaussian Naive Bayes classifier only got 82% of accuracy.[17]. Figure 3 and Figure 4 show the live capture of the fall event.

a

Figure 2.3 Fall event detection with a camera-based fall detection system for non-fall [16]

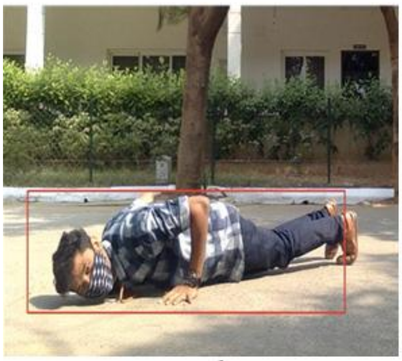


Figure 2.4 Fall event detection with a camera-based fall detection system for fall [16]

Ambience devices

Ambience devices use sensors such as pressure sensors, laser sensors, and inertial sensors to detect falls. They are often used in conjunction with machine learning algorithms to improve accuracy.

There is research[18] has proposed a low-cost but highly accurate machine learning for elderly fall detection using wearable sensors. The system used a triaxial accelerometer worn on the waist to acquire acceleration signals. The research had high accuracy, and low computational cost, and two-segment feature extraction algorithms avoided storing full sensor data to reduce machine memory requirements. However, the system may not capture all aspects of falls using only a single accelerometer, and the accuracy reported was specific to the dataset used, which may vary in real-world scenarios.

Summary of the Other Research

This section summarizes the advantages and disadvantages of some other research including the research already mentioned above in Table 2.1

Table 2.1 The summary of the references for the fall detection system

Author [References]	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[19]	Develop a small, wearable device that can reliably detect if an elderly user has fallen using sensor fusion and fall detection algorithms	The hardware components used are: 1. The MARG sensor 2. Accelerometer 3. Gyroscope 4. Magnetometer 5. microcontroller ATmega328 6. Bluetooth module 7. microSD card.	1. MARG sensors provide accurate 3D motion sensing by fusing data from multiple sensors 2. They can transmit wireless data for users to remote monitoring	1. Battery life may be reduced by wireless transmission and processing demands 2. Experimental falls may not fully replicate real-life uncontrolled falls.
[18]	Propose a low-cost, highly accurate machine learning for elderly fall detection using wearables sensors.	A triaxial accelerometer worn on the waist is used to acquire acceleration signals.	1. High accuracy 2. Low computational cost, 3. Two-segment feature extraction algorithms avoid storing full sensor data, reducing memory requirements.	1. Use only a single accelerometer which may not capture all aspects of falls. 2. The accuracy reported on specific datasets only, 3. Data distribution may vary around the world.

References	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[3]	Develop an improved wearables fall detection system that can automatically alert caregivers in the event of an elderly fall via IoT technology, enabling faster response and medical assistance.	The hardware components used are ESP8266, and MPU6050 sensor (triaxial accelerometer and gyroscope). The software used is Blynk IoT cloud server for data processing and notifications and Arduino IDE	1. Automatically detects falls and sends alerts without user input. 2. Improving the response time. 3. Low-cost 4. Lightweight wearable sensors for easy use by the elderly 5. Real-time IoT notifications to caregivers via email and mobile applications.	1. Potential for false alarms for misclassifying intense activity as a fall. 2. Wireless transmission range is limited due to only the use of Wi-Fi. 3. It only detects falls at the chest.
[20]	The project developed a wearable fall detection system for the elderly using a threshold-based approach applied to data from a 3D accelerometer worn at the waist.	This project uses predefined thresholds on various accelerometer-derived parameters like acceleration, pitch/ roll angles and calculated 'alpha' value to identify falls and normal posture. Using a 3-axis accelerometer sensor to provide acceleration data needed for analysis.	1. Provides a portable and lightweight wearable solution suitable for elderly users. 2. Achieved high accuracy (>90%) in detecting most normal posture and forward falls. 3. wireless alert capability enables remote notification of falls for prompt emergency response.	1. Detection of backward fall was poorer (75% success rate) indicating methods need improvement. 2. Susceptibility to false alerts from similar activities, limited durability, 3. Lack of integration with healthcare systems for medical data collection.

References	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[12]	This research developed an unintrusive fall detection system for the elderly by leveraging inertia sensors integrated discreetly into eyeglasses.	This research uses a threshold-based fall detection algorithm that compares acceleration, and tilt angle data to preset thresholds. This research uses an IMU sensor board integrating accelerometer and gyroscope, microcontroller ATMEGA328P, ESP8266 and a prototype eyeglass fitted with IMU and microcontroller PCB.	1. Discreet and comfortable form factors integrated into eyeglasses. 2. Lightweight and portable for long-term continuous monitoring. 3. Achieve a high accuracy of 95.44% in detecting falls. 4. Wireless alert capability	1. Accuracy could degrade over the long term due to sensor drift or ageing. 2. Placement on the head could be impacted by activities like sleeping. 3. Susceptible to false alerts for similar high-impact movements.
[6]	This project developed an IoT-based fall detection system for the elderly using wearables sensors and achieved high-accuracy fall detection on a deep learning model running on edge devices.	The equipment used for this project is wearable sensor nodes using a NUCLEO-L152RE microcontroller, MetaMotionR inertia sensors, Raspberry Pi, 6L0WPAN wireless sensor network, MQTT protocol, and Cloud service. The software used is Contiki OS for sensor nodes, Apache Flink for stream processing and TensorFlow.	1. This project achieved high accuracy (95.9%) in fall detection on real sensor data. 2. Low-powered wearable sensor node design 3. Identified optimal sensor placements and sampling rates.	1. Algorithm complexity. 2. Validation is done on a limited number of human subjects. 3. Sensitivity of threshold-based approach not analysed.

References	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[11]	This literature review, through a systematic analysis of relevant studies, provided insights into the current research landscape on using IoT/wearable sensor technologies for elderly health monitoring, assistance, and independent living support.	Sources of data: A systematic literature search of papers from 2010-2019. Papers were identified from scientific literature databases like Scopus, Web of Science, and PubMed. Additional papers were identified from references of earlier review papers.	1. Classified papers are systematically based on predefined aspects like health focus, IoT tech, aims, and more. 2. Provided qualitative and quantitative analysis of studies based on these aspects. 3. Identified current research trends, technologies and methods used.	1. Lacks quantitative meta-analysis due to heterogeneity of studies. 2. Search limited to selected databases. 3. Limited discussion on technical performance aspects of devices/methods.
[21]	This project presented the design and implementation of an IoT-enabled system to remotely detect falls in the elderly and notify caregivers, which can help manage emergencies promptly for improved elderly care.	The equipment used for this project is as follows: Hardware: MPU-6050 6-axis motion tracking sensor, WeMos D1 mini microcontroller, U-Blox NEO-6M, GPS module, Lipo battery, Mobile phone with Telegram app Software: Arduino IDE for microcontroller programming, Telegram API for notification functionality.	1. Uses affordable IoT sensors and microcontrollers for lightweight implementation. 2. Achieved a 90% success rate in fall detection during experimental testing. 3. Capable of sending location-embedded notifications to caregivers promptly. 4. Small, lightweight and portable form factor suitable for deployment.	1. Accuracy and latency need to be further evaluated under varied conditions. 2. No integration with external healthcare systems or emergency response. 3. Privacy and security of location sharing needs considerations. 4. Usability and acceptance by elderly users require user studies.

References	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[22]	this project presented the current landscape and trends in the field of using IoT technologies for elderly fall detection. It classified and analysed studies to understand and identify areas requiring further research efforts.	Equipment mentioned in case studies: Wearable sensors: accelerometers, gyroscopes, EMG sensors. Visual sensors: RGB cameras, RGB-D cameras. Ambient sensors: PIR, vibration, RF sensors, acoustic sensors Edge devices: Raspberry Pi, Arduino. Software: Arduino IDE, Machine learning tools.	1. Comprehensive analysis of over 100 papers published in recent years. 2. Classified papers systematically based on well-defined aspects. 3. Provided an overview of trends, methods, and challenges in the field. 4. Discussed technical as well as human aspects like usability.	1. Literature search limited to selected databases. 2. Lacks quantitative meta-analysis due to heterogeneity. 3. Health economics factors not considered. 4. Discussion is mainly based on reviewed work, not new experiments. 5. Scope limited to fall detection, excluding prediction/prevention.
[23]	The project presents the design and implementation of an IoT-enabled fall detection and notification solution for the elderly using wearable sensors, Arduino, machine learning and messaging capabilities to detect falls and promptly alert caregivers.	The equipment used for this project: Hardware: MPU6050 sensor, Arduino Uno microcontroller, GPS module, battery Database: MongoDB Software: Arduino IDE, Python, machine learning algorithms (KNN, decision tree),	1. Proposed a low-cost IoT-based solution using affordable sensors and microcontrollers. 2. Achieved a high accuracy of 98.75% in fall detection testing. 3. Can promptly notify caregivers in case of a fall	1. Small sample size of the testing dataset. 2. No integration with external healthcare systems/emergency response. 3. Limited to fall detection, doesn't include prediction/prevention. 4. Security and privacy aspects not addressed.

References	Description of project	Method / Source/equipment used	Advantage / Strength of project	Any limitation/disadvantage
[7]	This project presents the design of an IoT-enabled accidental fall detection system that integrates wearable sensors, a microcontroller and a smartphone app to automatically detect vehicle falls, check user's condition and promptly alert nearby hospitals to facilitate emergency response.	The equipment used for this project: Hardware: 1. MPU6050 accelerometer sensor 2. Heartbeat sensor 3. Arduino Uno microcontroller 4. Bluetooth modules 5. 6. Battery	1. Propose a low-cost IoT-based solution using affordable sensors and microcontrollers. 2. This project integrates fall and health detection using multiple sensors for accuracy. 3. Successfully detected falls and sent alerts in experimental testing. 4. Provides automated emergency alerting Buzzer to reduce response times	Limited to accident detection and doesn't include prediction or prevention of accidents.

2.4 Sensor technology

The efficacy of wearable fall detection systems depends on the sensors used to monitor the user's movements and detect falls. These sensors provide critical data that allows the system to differentiate between normal activities and fall events. The following sections discuss the various sensors commonly integrated into wearable fall detection systems, with a focus on their functions, advantages, and limitations.

Accelerometer

An accelerometer is used to measure linear acceleration along one or more axes. The accelerometers detect sudden changes in velocity[24], which are characteristic of falls. They measure the rate of change in motion and can identify rapid deceleration followed by a period of low movement, indicating that a person has hit the ground and is lying still. It has several advantages which are high sensitivity to sudden movements, low power consumption, and small size making them ideal for wearable devices.[25]. However, accelerometers alone can sometimes produce false positives, as they may interpret non-fall activities such as sitting down quickly as falls.

Gyroscope

A gyroscope measures angular velocity around one or more axes. It can detect changes in orientation and rotational motion[24]. When combined with accelerometer data, gyroscopes help distinguish between diverse types of falls and other activities, reducing the likelihood of false positives[25]. Gyroscopes provide additional context to accelerometer data, enhancing the accuracy of fall detection. However, they tend to consume more power than accelerometers and may require more complex data processing.[19].

Combined Sensor Unit (MPU6050)

The combined sensor unit such as MPU6050 is used to integrate an accelerometer and gyroscope in a single module.[21]. The MPU6050 is a popular choice for wearable fall-detection systems.[4] Because it combines both types of sensors, providing comprehensive motion and orientation data[21]. This integration simplifies the design and improves the reliability of the system. It is compact, cost-effective, and capable of providing detailed motion analysis by fusing data from both sensors.[2]. However, it requires calibration and sophisticated algorithms to process the combined data effectively.[21].

2.5 Analysis Effective formula

Before the analysis of sensor data, data preprocessing is an important process. It is very important to detect the missing values from the dataset and remove or replace the missing values. If not, it will affect the accuracy of the classification algorithms. Formula (1) and (2) are used to convert the value of accelerometer (A) and gyroscope (G) sensors into magnitude form.[20], [23]:

$$A = \sqrt{aX^2 + aY^2 + aZ^2} \quad (1)$$

$$G = \sqrt{gX^2 + gY^2 + gZ^2} \quad (2)$$

Where aX, aY, and aZ are the readings of the accelerometer while gX, gY, and gZ are the readings of the gyroscope. Then the overall magnitude (Mag) is calculated based on the combination of A and G using formula (3)[[23].

$$Mag = A + G \quad (3)$$

To detect the falls the values of Mag should be larger than the threshold value of the fall to identify it as a fall accident.

For analysis of the effectiveness of the fall detection system based on the parameters used:

Sensitivity: It is defined as the following ratio of the number of correctly classified falls to the number of falls[3], [26]:

$$Sensitivity = \frac{TP}{TP+FN} \qquad (4)$$

Specificity: The system can prevent false alarms which is the system detecting some activities that are similar to a fall[3], [26], [27].

$$Specificity = \frac{TN}{TN+FP} \qquad (5)$$

Accuracy: The system's ability to identify the actual falls while recognising the false falls [14], [26].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (6)$$

False positive Rate: the number of false falls that are identified as actual falls per time which is referred to as the false positive rate. It is calculated as a proportion of the number of false to the total time recordings.

$$False\ Positive\ Rate = \frac{FP}{ADL\ time} \qquad (7)$$

Where TP is the true positive indicated number of falls detected, False positive (FP) indicates false falling events detected. True negative (TN) indicates the system does not detect the falling event and False negative (FN) indicates activities of daily life (ADL) not detect as a fall event.

2.6 Dead Internet Zone

Dead zones in wireless networks are zones where either the connection is partial, or even absent due to a lack of signal strength. Technologically, it greatly minimizes the efficiency of a network and provides interruptions to communication, particularly in institutional and educational environments. Rambabu A. Vatti and colleagues used Wi-Fi analyzers to map out the strength of signals in networks spanning across the whole campus. Such factors deteriorate the performance of the network and lower the efficiency of communications, most especially in institutional and school environments[28]. Therefore, demands its coverage through either modification of routers' firmware, exchange of antennas, or use of wireless range expanders.

Dead zones can be thought of as experiential territories without connection or, in Nathanael Edward Bassett's words, areas characterized by disconnection that is not just technical malfunction but a site of social exclusion, alienation, and loneliness.[29]. Dead zones, according to Bassett, are metaphors for other issues at stake, among which one can include issues of social alienation, isolation, and obsolescence.[29].

2.7 Challenges for a fall detection system

Fall detection systems have emerged as a promising assistive technology for the elderly population who are at high risk of falls and their consequences. With ageing populations worldwide, falls have become a serious public health issue. However, developing fall detection systems that can perform robustly under real-world conditions presents unique challenges that researchers continue grappling with. This essay aims to discuss the key challenges, practical issues and emerging trends in fall detection as identified in recent literature.

2.7.1 Trends

Recent literature[30] pointed out some emerging trends that attempt to surmount prior shortcomings. Smartphone-based solutions are also on the rise since they leverage existing pervasive computing platforms.[2], [11]. The use of a smartphone in fall detection systems because it is a self-contained device, which presents a mature hardware and software environment for developing pervasive fall detection systems. Smartphones have built-in communication protocols that allow simple data logging to the device and wireless transmission while the price is also significantly reduced due to high production volume.

Moreover, algorithmic techniques such as machine learning methods are increasingly utilized for automatic pattern recognition from sensor data streams.[30]. Machine learning and thresholding techniques are the two basic ways that acceleration signals may be used to identify falls.[22]. Applications built using the first method require little computing effort and are easy to implement. They can recognize falls. The percentage of false positives, however, remains a severe problem. The machine learning method produces higher detection rates and is more advanced. Although these approaches are presently the norm because thresholding methods have proven to be ineffectual, there have been recorded challenges with their implementation.[22] Additionally, within the range of machine learning algorithms, no technique has gained widespread acceptance, and every study offers a unique strategy.

2.7.2 Challenges

One of the major challenges lies in achieving high performance when systems are deployed for actual usage instead of laboratory testing.[30]. Most studies still rely on simulated falls conducted by young volunteers in controlled environments. However, falls in real-life settings can occur from a vast spectrum of postures.[20], [22], involve complicating factors like medical conditions or environmental hazards, and may not resemble simulated counterparts. Ensuring algorithms and sensors can accurately classify falls with similar efficacy even under real-world complexity remains a challenge.

Another crucial challenge is system usability and gaining user acceptance[30]. For any assistive technology to see widespread adoption, users must find the solution convenient and non-intrusive in their daily lives[31]. Aspects like form factor, aesthetics, ease of use and battery maintenance directly impact user experience and likelihood of continued adherence. Additionally, there are open questions about social acceptability and how systems can be designed sensitively without eliciting privacy concerns or a sense of stigma[22]. Addressing usability and user-centric challenges will be paramount for fall detection systems.

2.7.3 Issues

Several practical implementation issues also need to be overcome. Power consumption is a major impediment.[30], [31], especially for wearable devices which require continuous monitoring. Advancements in low-power embedded systems and energy harvesting techniques will be needed to alleviate battery life constraints.[25]. Performing real-time data processing while meeting latency requirements is another issue area given the limits of on-device sensing and computation. Privacy preservation also emerges as a concern, particularly for context-aware systems leveraging video/audio feeds without adequate anonymization and security protocols.[11].

2.8 Summary

In this section, this project discussed the research and process for creating and building a fall detection system for elderly people using Node MCU, ESP32 and Arduino as microcontrollers. The planned system sent alerts to distant family members and caregivers via the Telegram application message service if a fall occurred. The WhatsApp application was chosen to send the alert since it is free software and a widely known program. Everyone can easily use the software, and it is not extremely complicated.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter covered the method used in this project. This chapter will address every aspect of the project including the experimental setup, parameters, tools, equipment, and restrictions as well as the technique and summary for creating a fall detection system for the elderly using ESP32. A methodology offers a theoretical framework for choosing the most appropriate technique, combination of techniques, or best practices.

3.2 Flow Chart Design

The project needs to be in 2 separate parts which are the sender and the receiver. The sender is the part that can be a wearable device while the receiver can be the terminal that can send notifications to caregivers via WhatsApp.

Based on Figure 3.1 the system starts by initializing a 6-axis motion sensor, such as an MPU6050, which reads acceleration and angular velocity along the user's X, Y, and Z axes. The device calculates two important parameters: the magnitude of acceleration and the angular displacement magnitude. It compares these measures to predetermined thresholds to identify significant events. If the acceleration exceeds 20m/s or the angular displacement exceeds 80°, the system presumes the user's status as "normal" and resumes monitoring. If high acceleration is sensed, any resultant movement is checked to see if a fall has occurred. If no movement is detected, the system assumes a fall has occurred and sets the fall state to true. If a fall is detected, the system sends the fall status and GPS coordinates to the receiver's ESP32 using ESP-NOW, which needs to be notified right away. The system moves into a waiting mode for a response via a push button from the user. If the button is pressed, the system resets the fall state and acceleration flags, while if not pressed, the fall state remains active, keeping the incident logged and sending repetitive notifications if necessary Then, the system resets and goes back to its original state, continuing its normal monitoring.

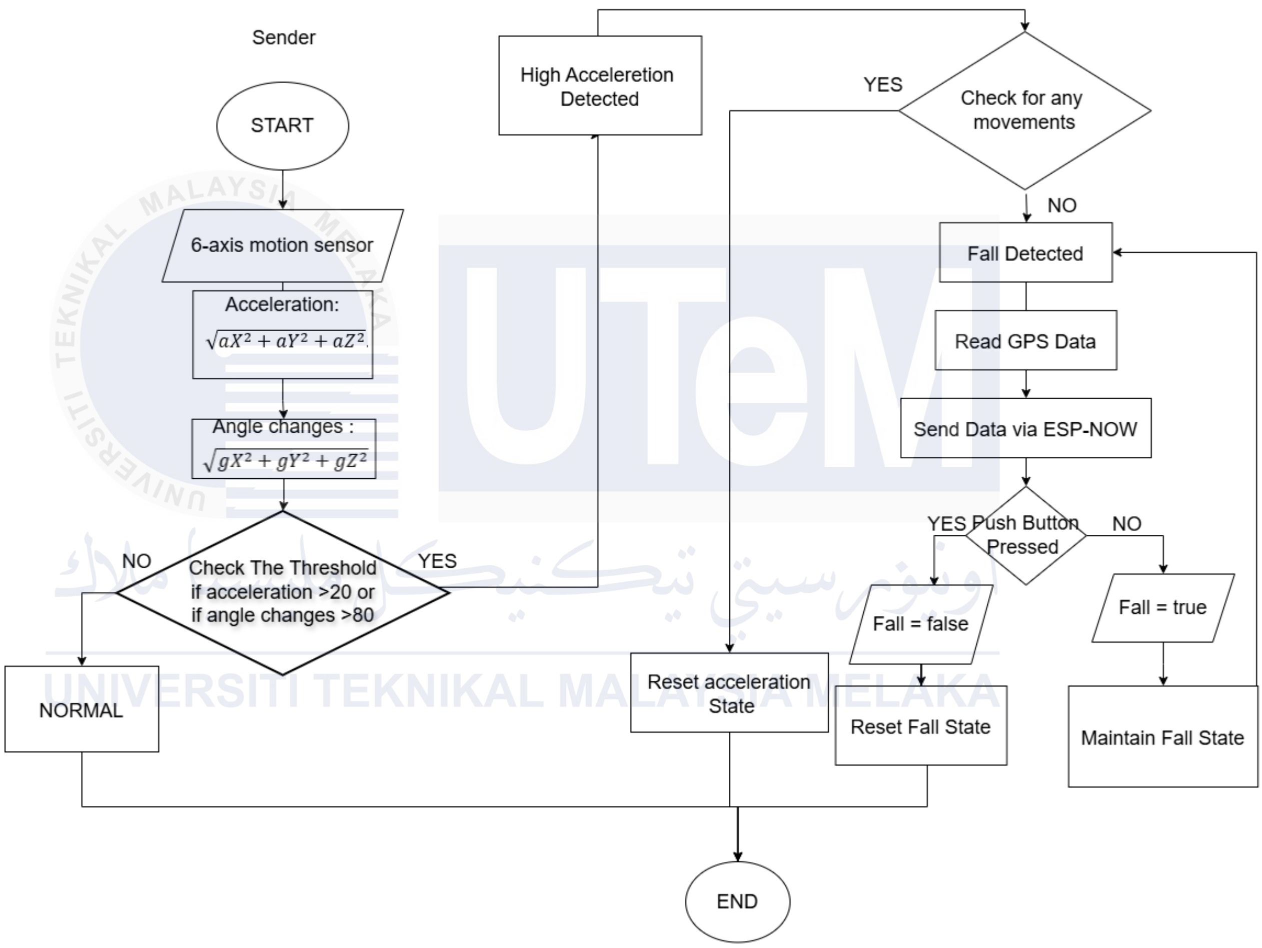


Figure 3.1 Flow Chart of the Sender

Based on Figure 3.2 the receiver module waits for data via the ESP-NOW protocol, providing information about fall detection status, emergency ID, and GPS location. The received data is displayed on an I2C LCD screen, providing immediate feedback. If the condition is non-critical, the system resets to await fresh data to prevent false alarms.

During a critical fall event, the system triggers an audible buzzer alarm, enabling early intervention. The buzzer continues sounding until manually turned off. Once deactivated, the system switches from ESP-NOW protocol communication to Wi-Fi for remote notification, such as sending a WhatsApp message to pre-programmed contacts. The system goes into a temporary standby mode for seamless internet-based communication, sending a WhatsApp message with all critical fall information, including emergency ID and GPS data, and informing family members, and caregivers about the incident. Once the WhatsApp message is sent successfully, the system switches back to ESP-NOW, allowing the module to receive new data from the wearable device. This end-to-end response approach ensures timely help to the user, reduces false alarms, and increases the system's usability and reliability.

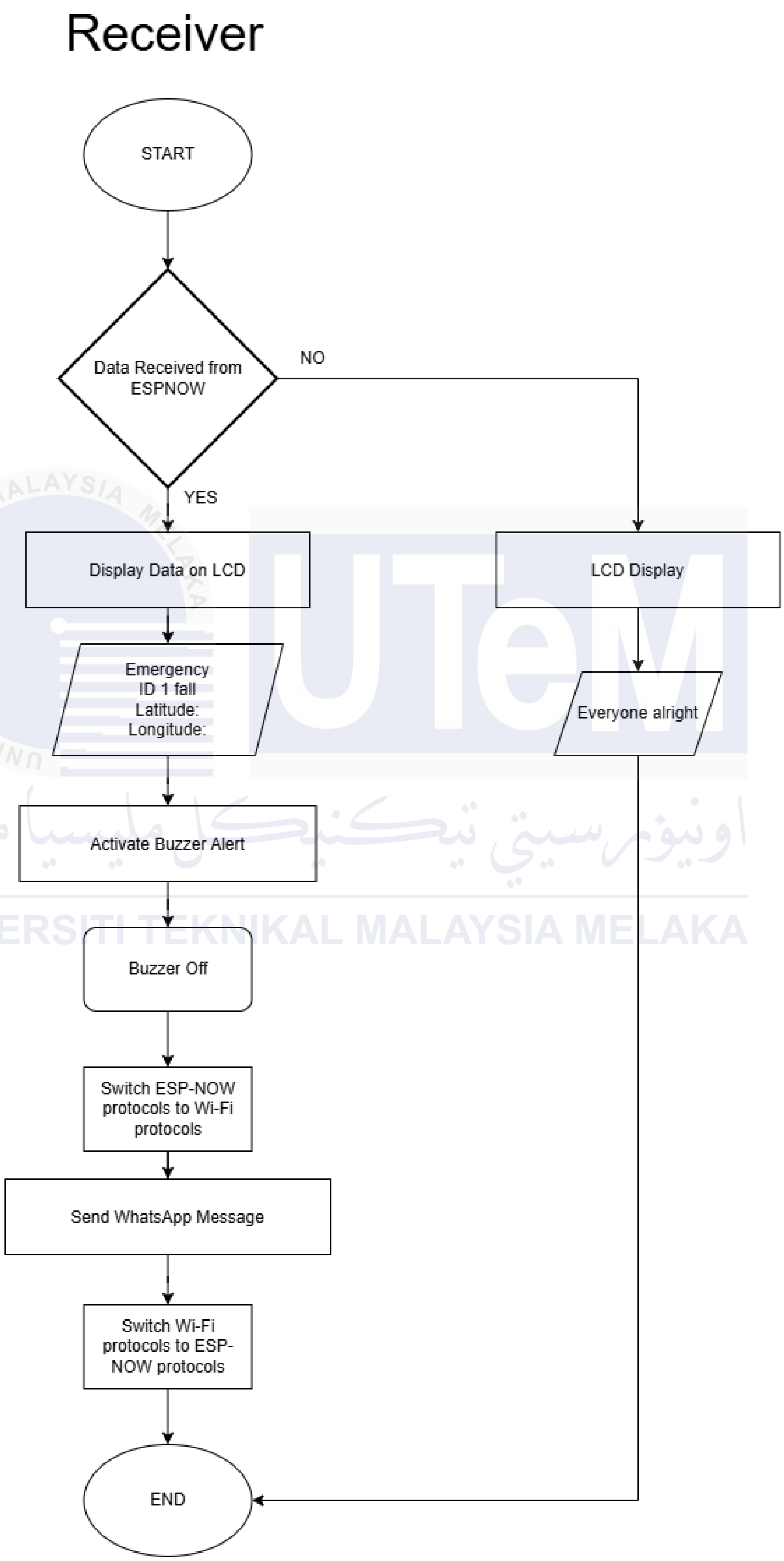


Figure 3.2 The flowchart of the Receiver

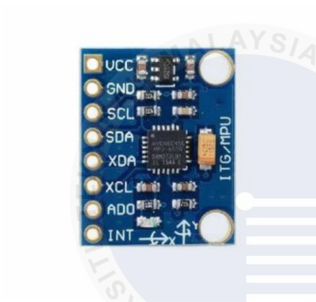
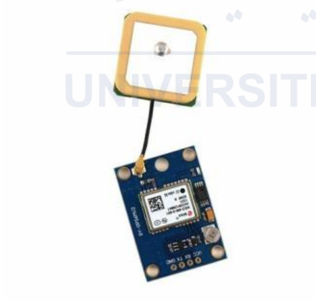
3.3 Software and Hardware Implementation

There are three stages to implementing the hardware and software selected above which is stage 1 which is to determine the hardware components used with their specification. It can help to identify the advantages and limitations of the project. Then, stage 2 is to determine the specific software tools used for software development and software for IoT. It can help to determine the software that needs to be installed on the personal computer for achieving the IoT part of the project. Finally, stage 3 for integrating the software and hardware to build the project. By using the schematic circuit and the simulation, we can build the projects effectively with minimal time.

Stage 1: Hardware Implementation

Based on the research in the Literature Reviews section, the hardware such as sensors and microcontrollers had been decided for the project. Additionally, several components can be used as the outputs. The decided components needed for the sender were a microcontroller ESP32-C3, MPU6050, push button and Neo-6m GPS module, While the receiver, consists of a microcontroller ESP32, Liquid Crystal Display with I2C, and buzzer. Table 3.1 below shows the components used for this project with their specification and their purpose in the project.

Table 3.1 The list of the components used for the project

Component	Specification	Purpose
Buzzer 	Operating Voltage:3-5V Sound Output: ~85 dB. Frequency: ~2300 Hz Current Consumption: ~10 mA	The buzzer is used to provide audible alerts in case of a detected fall
MPU6050 	Sensors: 3-axis accelerometer and 3-axis gyroscope Gyroscope Range: ±250, ±500, ±1000, ±2000 °/s Accelerometer Range: ±2g, ±4g, ±8g, ±16g	The MPU 6050 is used to detect motion and orientation. Its accelerometer and gyroscope data are crucial for implementing the fall detection algorithm by analyzing movement patterns.
Neo-6M 	Channels: 50 Sensitivity: -161 dBm Update Rate: 1 Hz Operating Voltage: 2.7-3.6V	The Neo-6M GPS module provides location data, which can be used to track the patient's location in case of an emergency or fall, enabling quick assistance.
ESP32 	Microcontroller: ESP32 Operating Voltage: 3.3V Digital I/O Pins: 17 Flash Memory: 4 MB RAM: 128 KB Wi-Fi: 802.11 b/g/n	ESP32 is used for wireless communication over Wi-Fi, allowing the wearable device to send data to cloud servers or smartphone apps for remote monitoring and alerts.

Component	Specification	Purpose
Push Button 	Type: Normally open (NO) Operating Voltage: up to 12V Current Rating: 50mA Dimensions: 12x12mm	The push button is used for user interactions, such as resetting the device and acknowledging alerts.
Seeedstudio Xiao ESP32-C3 	Microcontroller: ESP32-C3 (RISC-V single-core processor) Operating Voltage: 3.3V Digital I/O Pins: 11 Flash Memory: 4 MB RAM: 400 KB SRAM Wi-Fi: 802.11 b/g/n (2.4 GHz)	ESP32-C3 is used for wireless communication over ESP-NOW protocols, allowing the wearable device to send data to another ESP32
Battery Li-Po 	Voltage: 3.7V Capacity: 1200mAh	To supply power to the ESP32-C3 as a wearable device.
Liquid Crystal Display with I2C 	Display Type: 16x2 character LCD Operating Voltage: 5V Communication: I2C interface Power Consumption:<50mA	The 16x2 LCD with I2C is used for showing information like fall detection, user ID, and GPS location. It gives quick updates to the caregiver.

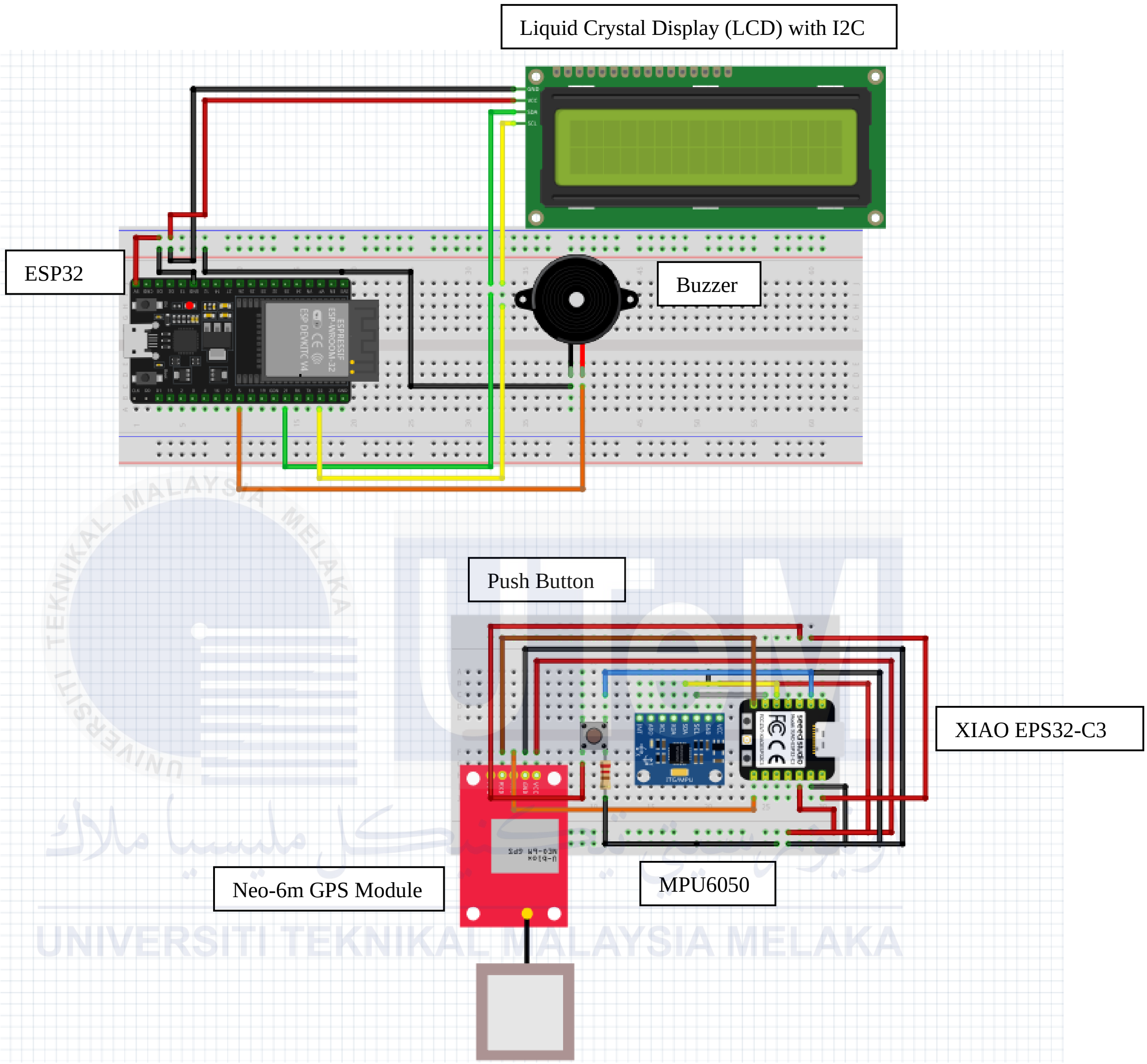


Figure 3.3 The schematic circuit design for the project

Based on Table 3.1, a schematic circuit was made in Figure 3.3. The hardware was selected based on the researched literature in Chapter 2. First, ESP32 is a microcontroller that can read and calculate the incoming input data from a 3-axis accelerometer and 3-axis gyroscope sensor and handle the GPS module, Neo-6m, Then, ESP32(1) will send the data obtained to the ESP32(2). Finally, the ESP 32(2) can send the notification via selected software such as telegram or Arduino Cloud.

The buzzer is an alarm system that when the Arduino nano reads and calculates the input sensor of MPU6050 and it detects it as a fall incident, the buzzer will be on as an alarm for nearby people. The push button is as to confirm when the false alarm was triggered or when the wearer is in good condition and does not need any help from the caregiver and people nearby. Lastly, the LEDs are an indicator of the system, when the green LED is on it indicates that the system is on and online.

Stage 2: Software Implementation

Table 3.2 The list of the software used for the project

Software	Purpose
Arduino IDE	To write, compile, and upload code to the Arduino boards. Provides libraries for interfacing with various sensors and modules.
WhatsApp	WhatsApp helps connect different apps and devices. Using this app and appropriate coding, the microcontroller Esp32 can interact with devices or smartphones.

Based on Table 3.2, there are two types of software involved in building the project. Arduino IDE software is used to write and upload the code into ESP32 and ESP32-C3. WhatsApp application is selected as a medium for ESP32 to send notifications and locations to the user as an alarm and reminder by using the appropriate code for the microcontroller.

Stage 3: Software and Hardware Implementation

The block diagram in Figure 7 represents a system where MPU6050 is used to measure and calculate the acceleration and angle changes of a person by Xiao ESP32C3. Then, Neo-6m is used to determine the person's location with the Internet and send the notification to the caregiver.

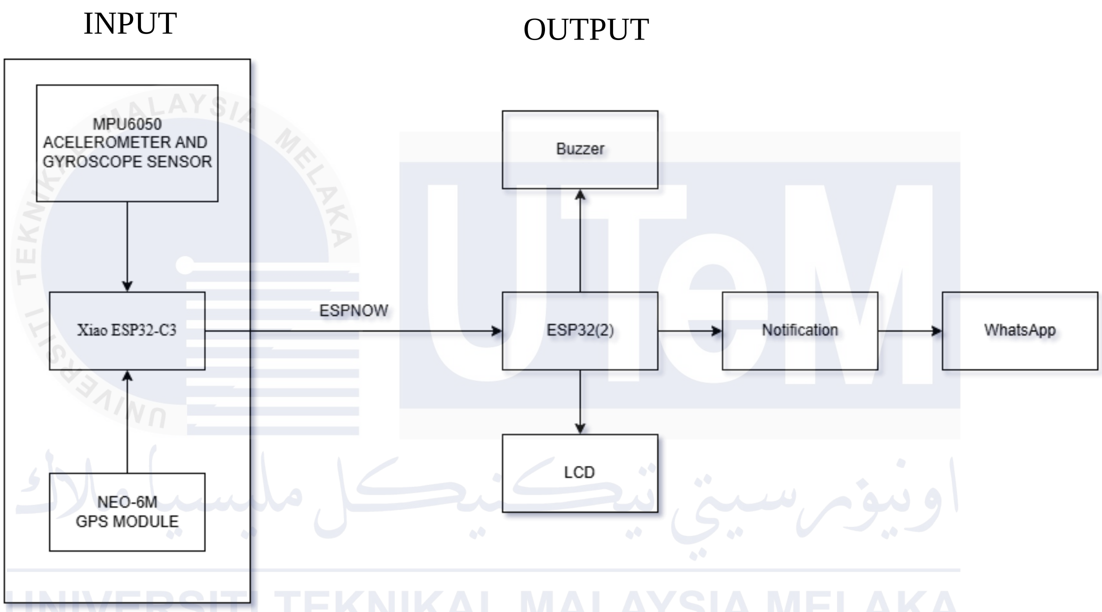


Figure 3.4 Block diagram of the project

3.4 Data Collection

The data collection involved the 2 microcontrollers which are Esp32-C3 and ESP32. It needs to connect the physical components, store data of desired calculated value parameters, and show the outputs that are triggered by the sensor. The outputs are a buzzer and the location when it is sent via the WI-FI module. Nowadays, the community of research and development on fall detector systems has developed multiple systems and algorithms to detect falls and avoid false alarms indicated by false fall incidents by the microcontroller.

3.5 Development Stage

This project's purpose is not to prevent falls and the injuries that can occur to elderly people, but this project objective is to notify the caregiver or family member for immediate response when the elderly fall. It can prevent the incident from becoming worse. To differentiate the fall incidents and the human daily activities which are walking, jumping sitting, running, lying, and sleeping. In this section, there are some stages for developing this project. The Gant chart for the planning progress for the PSM in both PSM 1 and PSM 2 is in Appendix A.

6-axis sensor reading

MPU6050 is a 6-axis motion sensor that combines a 3-axis accelerometer and a 3-axis gyroscope. It is used to detect changes in motion, acceleration, and rotation. The accelerometers measure the static acceleration due to gravity and dynamic acceleration caused by motion such as daily activities, shock, and vibration.

Theoretically, in a static object, the acceleration over the Z-plane is equal to the gravitational force which is 9.8 ms^{-2} while the X-plane and Y-plane should be zero. Furthermore, a gyroscope is used to measure rotational velocity or the rate of change of the angular position over time, along the X, Y, and Z-axis which are indicated as roll as the X-axis, pitch as the Y-axis and yaw as the Z-axis.

Combining the information from both accelerometer and gyroscope will make the information about sensor orientation more accurate. The accelerometer provides static acceleration data, while the gyroscope provides dynamic rotation data.

Microcontroller

During the development of fall detection systems, many varieties of microcontrollers can be used as the system's ‘brain’. For this project, ESP32 was chosen with a series of ESP32C3, which is smaller and more suitable for wearable devices.

3.5.1.1 Seedstudio Xiao ESP32C3

XIAO ESP32C3 by SeeedStudio is an ESP32 microcontroller that is compact, and powerful based on the Espressif ESP32-C3 chip. It features a RISC-V core with built-in WiFi and Bluetooth 5 protocols. This board includes with external antenna to increase the signal strength for wireless communication such as WiFi, Bluetooth, Radio signal, and more. The Xiao ESP32-C3 is small in size and can be used as a limited-space microcontroller with many features. The reasons for choosing this microcontroller are this board compatible with Arduino IDE for programming, small for limited space for wearable devices, has low power consumption, and has numerous GPIOas which are 11 digital Inputs/outputs that can be used as PWM pins, 3 analogue input/output that can be used as ADC pins.

3.5.1.2 ESP32

ESP32 is a microcontroller developed by Espressif Systems. It is also widely recognized by the dual-core processor, besides supporting Wi-Fi and Bluetooth, which will be very easy to integrate into IoT and embedded systems. With its variety of peripherals that may interface with a broad range of sensors and actuators, including ADCs, DACs, PWM, UART, SPI, and I2C interfaces. With wide community support and compatibility with several development environments, the ESP32 has become one of the favourite solutions for applications for control systems.

ESP NOW Protocol

Espressif Systems developed ESP-NOW, a low-power wireless communication protocol designed especially for ESP32 and ESP8266 devices. This protocol allows for efficient, low-latency communication between multiple devices without requiring a regular Wi-Fi network. It uses the 2.4 GHz frequency range and runs on a peer-to-peer architecture. Additionally, broadcast and unicast communications are supported by ESP-NOW. Furthermore, it can be applied to device-to-device communication, sensor networks, and remote monitoring systems. For situations requiring real-time data transfer with minimal setup, ESP-NOW is excellent.

The ESP-NOW protocol was tested based on the range of connectivity of 2 ESP32s. The test consists of an ESP32 sending 100 data to another ESP32 at different distances between the two ESP32s. The test was done in Land of Sight (LoS) and Non-Land of Sight (NLoS). LoS is the empty area where the ESP32s face each other without any obstacles for example open field area. While NLoS is an area that has no obstacles between two ESP32s. For this testing for the NLos area was made in the residential where houses act as obstacles between 2 ESP32s. Table 3.3 with figures 3.4 and figure 3.5 indicates the result of the testing number of packet loss during transmission data between 2 ESP32s in two different areas.

Table 3.3 The result of testing ESPNOW packet loss in NLos and LoS

Distance(m)	Packet Loss in (NLoS)	Packet Loss in (LoS)
25	47	0
50	79	2
100	100	96
150	100	99
200	100	100

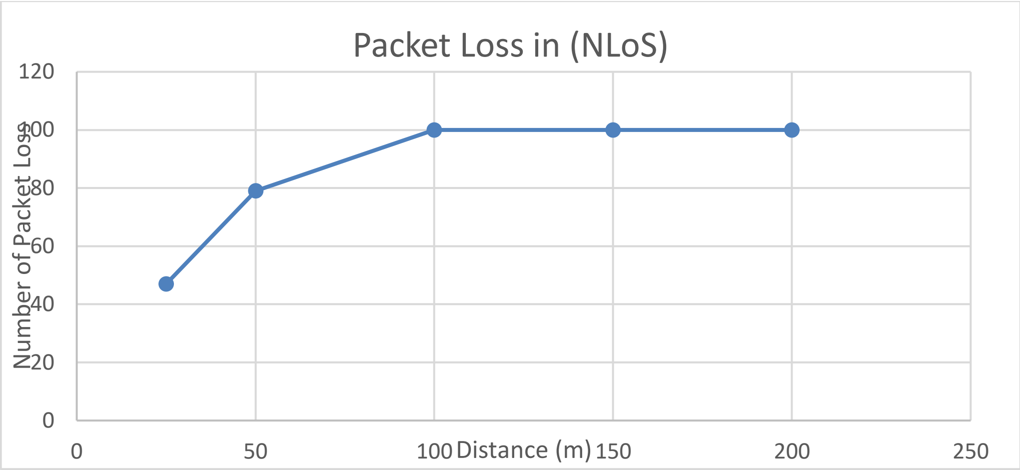


Figure 3.5 Graph of the number of packet losses in NLos based on distance

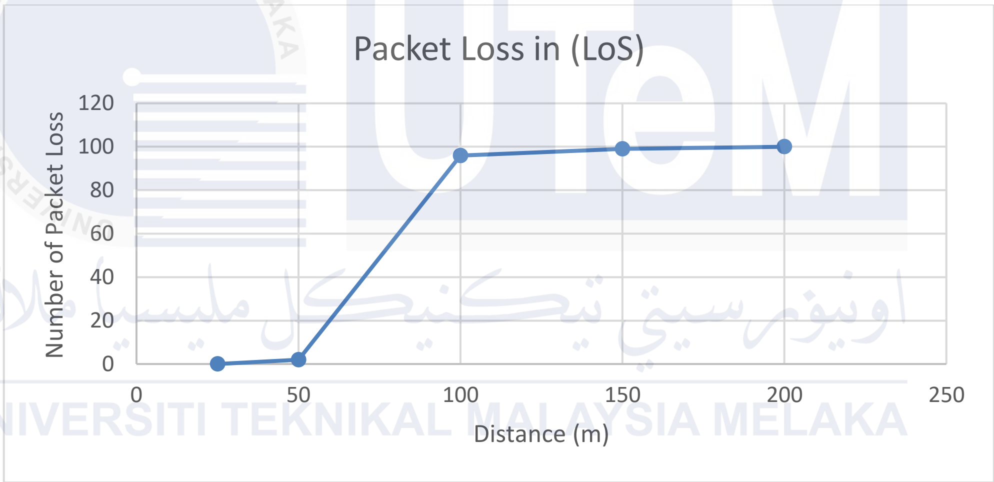


Figure 3.6 The graph of the number of packet losses in Los based on the distance

During the NLoS area shows that the number of packets lost during the transmission of data was 47 at 25 metres. Then, the number of packet losses increased significantly at 50 metres which was 79 and increased to 100 at a distance of 100 metres and more. It indicated that there are no more data were received at 100 metres, so after 200 metres the test stopped.

During the LoS area shows that there were 0 packet loss at 25 metres. Then, at 50 metres a few packets of data were lost during the transmission which was two packets. At 100 metres, there are 96 packets of data lost with the number of packet losses was increased to 200 metres. At 200 metres, the packet loss was 100 percent and there were no data received from the sender.

Based on the result, in LoS, the data can be transferred without any loss of packet of data at distances below 50 metres while in NLoS there are chances of losing the data that can be sent from the sender.

Simulation

The project was first built in the simulation. Conducting a simulation before building a physical prototype offers several significant benefits that enhance the overall development process. Firstly, it is cost-efficient, as simulations allow you to test and refine your design without the need for physical components, thereby avoiding the expense of wasted materials on faulty prototypes. This process also mitigates risks, ensuring the safety and viability of the design before physical implementation. It also helps detect potential errors early, reducing the likelihood of failure in the actual prototype.

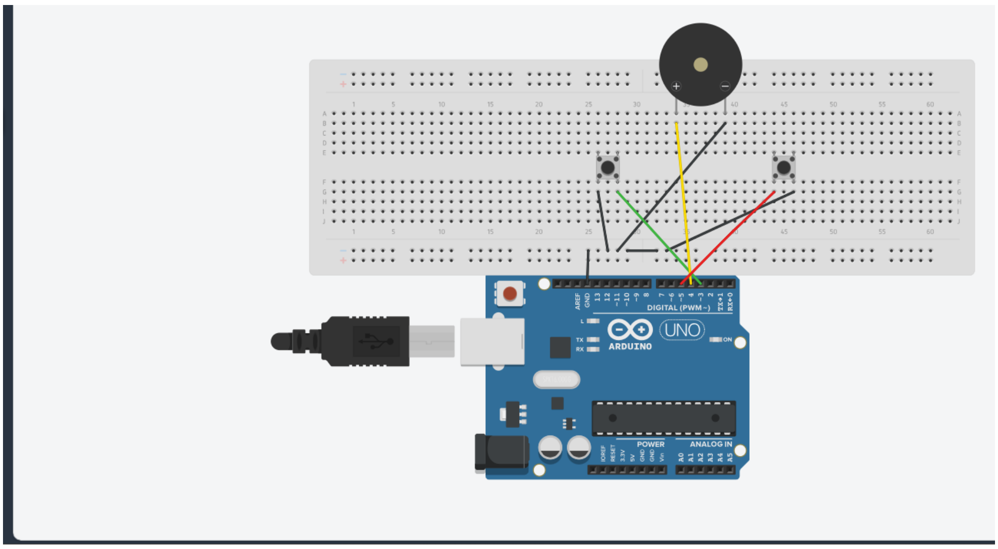


Figure 3.7 Simulation on TinkerCAD website.

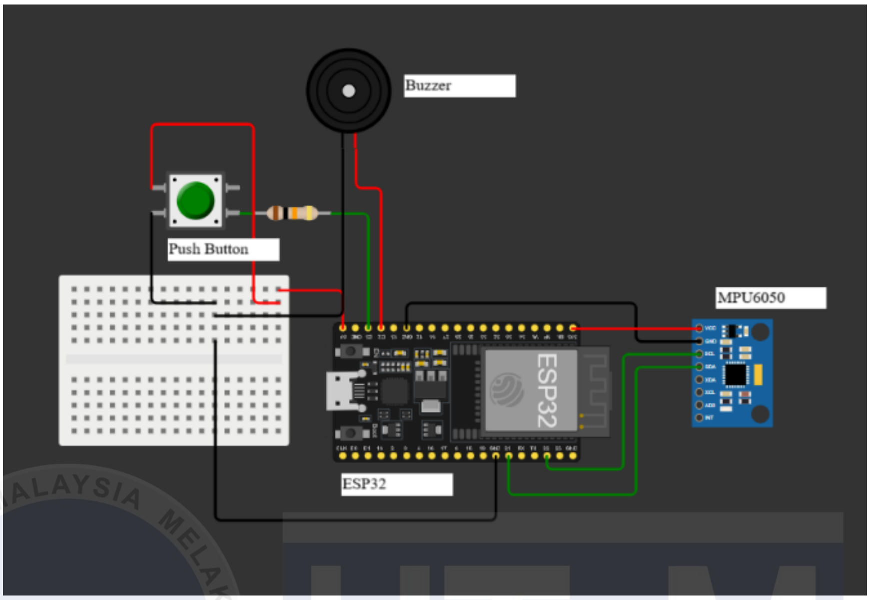


Figure 3.8 Simulation using WOKWI website.

In tinkerCAD as in Figure 8, ESP32 is not in the library, Arduino Uno is used to replace the microcontroller by connecting to two push buttons one of it set as a trigger input to replace the MPU6050 sensor. This simulation is to simulate when the system triggers the fall event in Button 1 the buzzer will stay on for a fixed time while waiting for a response in Button 2, When Button 2 is pushed, the buzzer will be off immediately.

On the WOKWI website as in Figure 9, the circuit can be simulated with an MPU6050 sensor that integrates with ESP32. The buzzer is set as output, and the push button is set as an input to off the buzzer when the system triggers a fall event.

3.6 Threshold for fall

Determination of fall detection thresholds with an MPU6050 accelerometer.were the thresholds were calculated based on acceleration and change in angles during a range of activities, such as static (static), walking, and rapid walking. The thresholds are important in falling detection and differentiation with routine activity, in such a way that the system will be reliable and accurate.

Experimental Setup

Experiments in a laboratory environment determined the thresholds. The device, with the MPU6050 accelerometer, experienced four scenarios:

a. Static State: Device in a static position in a horizontal plane in a view to obtain baseline values.
b. Walking: Device at waist level when the subject moves at a normal pace.
c. Brisk Walking: Device at waist level when the subject moves at a high pace.

The acceleration in the X, Y, and Z axes and angular displacement (roll and pitch) were measured through a sensor and then analyzed in a view to monitor trends and variations between routine activity and a fall. The acceleration value is calculated by using the formula the unit g as acceleration due to gravity:

$$Acceleration(g) = \sqrt{aX^2 + aY^2 + aZ^2}$$

The angle change value is calculated using the formula:

$$Angle\ Changes\ (°) = \sqrt{gX^2 + gY^2 + gZ^2}$$

Data Analysis

. The analysis of acceleration and change in angle over 30 seconds of acquired observations in three activities: static, walking, and brisk walking.The table below shows the test value of the acceleration and angle changes based on three different activities.

Table 3.4 The test value of the acceleration and angle changes based on three different activities which were static, walking and brisk walking

Time (S)	Activity					
	Static		Walking		Brisk Walking	
	acceleration	angle changes	acceleration	angle changes	acceleration	angle changes
1	9	2	9	22	9	24
2	9	3	9	81	9	59
3	9	3	9	107	9	30
4	9	3	9	26	9	38
5	9	2	9	56	10	23
6	9	3	9	69	7	22
7	9	3	9	41	7	44
8	9	3	9	6	20	41
9	9	3	9	18	8	24
10	9	3	9	6	11	11
11	9	3	9	94	8	26
12	9	2	9	45	20	13
13	9	2	9	34	7	38
14	9	2	9	70	11	41
15	9	3	9	36	9	54
16	9	3	9	65	9	52
17	9	3	9	76	6	4
18	9	3	9	35	8	55
19	9	3	9	18	13	57
20	9	3	9	34	10	31
21	9	2	9	5	5	37
22	9	2	9	39	19	42
23	9	2	9	39	4	42
24	9	3	9	26	6	23
25	9	3	9	30	5	27
26	9	2	9	70	5	14
27	9	2	9	77	20	16
28	9	3	9	44	10	9
29	9	3	9	104	12	17
30	9	3	9	141	15	11

For the static position, acceleration remains constant at 9g units for 30 seconds, and this indicates that the device is not in motion and experiences little or no activity. Angle variation in such a position is also minimal, between 3° and 5°. This minimal variation informs one that any detected change in angles is most likely a function of minor vibratory motion or small tilts and not active motion.

Walking entails acceleration between 9g and 141g units, indicative of the impact forces of one's steps. This variation mirrors a walking rhythm, with acceleration changing and decreasing with motion taking place. Angle variation is between 9°and107°, depicting the change in postures and swinging motion of a limb when walking.

For brisk walking, Acceleration and angle variation record larger changes for quick walking than for regular walking. For acceleration, values range between 9g and 206 g units, indicative of increased intensity and force in terms of a faster motion pace. Angle variation is also larger, between 17° and 115°, indicative of sharper and swifter motion of the body.

Results and Final Threshold

The analysis showed a fall could be detected with high accuracy with the following thresholds:

Acceleration: 20g and more

Angle Change: 200° and more

3.7 Creating WhatsApp Bot

CallMeBot provides an unofficial way to send WhatsApp messages by using API. It relies on authenticating phone numbers to send messages. One of the advantages of using this bot is that it allows you to interact with ESP32 remotely, without having to physically access it. This can be useful if the user wants to control the ESP32 from a different location or wants to monitor it remotely. There are steps to create the CallMeBot API :

i. Save CallMeBot numbers into phone contacts

ii. Send a Message to the CallMeBot number with the text “I allow callmebot to send me messages”.

iii. Waiting for a reply. The CallMeBot will respond with personal API keys.

iv. Then, use the API keys provided to send messages and integrate code into the system.

The figure below indicates the result of connecting the WhatsApp API Bot with some testing messages to ensure that the messages can be sent through an ESP32 receiver that can be used as an alert to caregivers and relatives.

Figure 3.9 The messages are received through the WhatsApp application from an ESP32 receiver.

3.8 Summary

In this section, the process for building a fall detection system for the elderly using Arduino as a microcontroller was presented. The system will send notification and location of the incident to family members and the caregiver via Node MCU to their phones. The Arduino will detect the fall incident when it detects the high acceleration by sensor MPU6050, and a buzzer will be on as an alarm to the people surrounding it. The family members and caregivers will get notified and immediately give an emergency service for the elderly if the push button is not pushed in 40 seconds. The hardware and software are implemented for this project with the data collection for the sensor MPU6050.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This study undertook to address two objectives of this project, as identified in Chapter 1 section 1.3 which aimed at developing an easy monitoring wearable device, developing low-cost alternatives to nursing, and integrating software and hardware into building the fall detector. Results are therefore presented based on the three key areas of concern namely; the process of developing a fall detector using ESP32 for the elderly, and the study on the project's effectiveness. The results in each area shall be explained and discussed in detail throughout this chapter.

4.2 Result

Results sections were created to show the design and interface of the project to give a clear image of the finished product with some testing of the prototype in a controlled environment.

Designed Prototypes

The prototype for this project had been designed. The sender had been designed as compact as possible for more lightweight and comfortable wearable devices. This is because a wearable device should be as comfortable as possible, especially for elderly people who going to wear it for a long time. The prototype was tested before it soldered on donut boards. As shown in Figure 4.1 and Figure 4.3 below the circuits that had been built based on the schematic circuit from Figure 3.3. The sender was built and soldered on the donut board to make it more compact and small size. The size of the sender circuit is 85mm x 50mm. In Figure 4.2, the sender prototype circuit is put into an arm strap as a wearable medium. The wearable device is put on the arms because of the comfort and accessibility. The user is almost familiar with wearing devices such as smartwatches or fitness trackers on their arms which makes it more comfortable and natural. Figure 4.4 shows the casing for the terminal which had inserted the prototype circuit of the receiver in it. Then, Figure 4.5 shows the wearable device that consists of the sender circuit in the arm strap.

Figure 4.1 The prototype design for the sender that can be used as a wearable device

Figure 4.2 The prototype that was put into arm-straps to become a wearable device

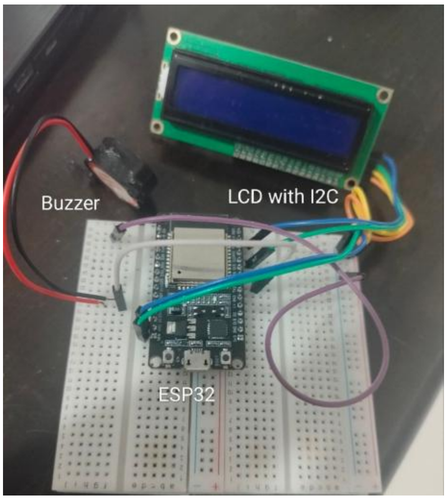


Figure 4.3 The prototype circuit for the receiver

Figure 4.4 The prototype of the receiver

Figure 4.5 The sender was put into arm straps that can be worn at the arm.

Testing In Control Environment

During the testing in a controlled environment, there are several tests for several types of falls: forward fall, backward fall, and sideward fall. The project also tested some daily activities or activities of Daily Life (ADL) which consists of daily activities that people do, such as walking, running, sitting, lying, and doing chores. The table below shows the results during the control testing of the project.

Table 4.1 The result of a control testing Environment which consists of several types of falls Activity Daily Life (ADL).

Test Scenario	Number of Tests	Fall Detected / True Positive (TP)	False Positive (FP)	Missed Falls / True Negative (TN)	Accuracy (%)	Latency (Seconds)
Forward Falls	20	18	0	2	90	10
Backward Falls	20	18	1	1	94	9
Sideward Falls	20	17	2	1	94	9
Activity Daily Life (ADL)	30	8	2	N/A	N/A	N/A
Overall	60	32	5	4	93	9

The fall detection system was evaluated across multiple test scenarios to assess its ability to accurately detect falls, avoid false alarms, and respond within an acceptable timeframe. The test results highlight various metrics, including True Positives (TP), False Positives (FP), True Negatives (TN), accuracy percentages, and latency. These metrics provide a comprehensive view of the system's performance in different scenarios.

For forward falls, 20 tests were conducted. The system successfully detected 18 falls, resulting in a True Positive rate of 90%. However, 2 falls were missed, and the average latency for detection was 10 seconds. Although the system performed adequately, forward falls had slightly lower accuracy compared to other scenarios, indicating room for improvement in detecting this type of motion. Backward falls demonstrated a stronger performance, with 18 out of 20 falls correctly identified. Only 1 false positive was recorded, and 1 fall was missed, resulting in a high accuracy rate of 94% and a latency of 9 seconds.

Sideward falls were also tested with 20 trials. The system correctly detected 17 falls, with 2 false positives and 1 missed fall. This resulted in an accuracy rate of 94%, comparable to the backward fall scenario, but with a slightly higher false positive rate. The latency for sideward falls was also 9 seconds, reflecting consistent responsiveness across these types of falls. For normal daily life activities (ADL), which aimed to simulate non-fall scenarios, the system erroneously detected 8 falls, with an additional 2 false positives. While no true falls were expected in this scenario, these results highlight the need for better differentiation between regular movements and falls.

When considering overall performance across all test scenarios, the project demonstrated an accuracy of 93%, with 32 falls correctly detected, 5 false positives, and 4 missed falls out of 60 tests. The average latency for fall detection was 9 seconds, indicating the system is capable of responding within a reasonable timeframe. These results show that the system is reliable for most fall scenarios, with a low false positive rate and high detection accuracy.

The Receiver

The sender sends the data when it detects a fall accident through the ESP-NOW protocol to the receiver. Then receiver's ESP32will disconnect the ESP-NOW protocols to connect the set Wi-Fi to connect to the Internet. This is because ESP32 can do two or more Wi-Fi protocols at the same time. Furthermore, ESP-NOW protocol and Wi-Fi use the same radio signal in ESP32. After the receiver receives the data from the sender. The LCD will be changed from "Everyone Good" to emergency alert, user's IDs with location of the user by latitude (Lat) and longitude (Long). Then, the buzzer will be on with a tone that can be an alarm for the caregivers.

Table 4.2 The Output of the LCD when Fall detected

The output at the sender	**Display**
	Emergency ! ID 1 fall
	Lat: 2.332725 Long: 102.295120

The WhatsApp Notification

Based on the result, it takes almost 5 to 9 seconds depending on the Internet connection strength, The receiver ESP32 will switch from Wi-Fi mode ESPNOW to connect to the Wi-Fi that had been set. Then, ESP32 will send the data received from the sender through the link of the URL with the registered API key and number phone that is connected to the WhatsApp application. The messages consisted of the ID of the user, and the latitude and longitude of the user based on data received from the Neo-6m GPS module. In addition, the Google map link contains the latitude and the longitude to easier for the receiver especially the caregivers to track down the user’s location. Based on figures below show the message and the Google link from the user’s location. When the caregiver clicks the link given in the message. The Google Map will open with the data of the Latitude and Longitude of the sender, As the figures below, the latitude and longitude are 2.332725 and 102.295120 respectively. In Google Maps, they convert the latitude and longitude in decimal format into degrees, minutes and seconds (DMS) format. Google Maps show the converted latitude and longitude in DMS format which are 2°19’57.8” N and 102°17’42.4” E respectively.

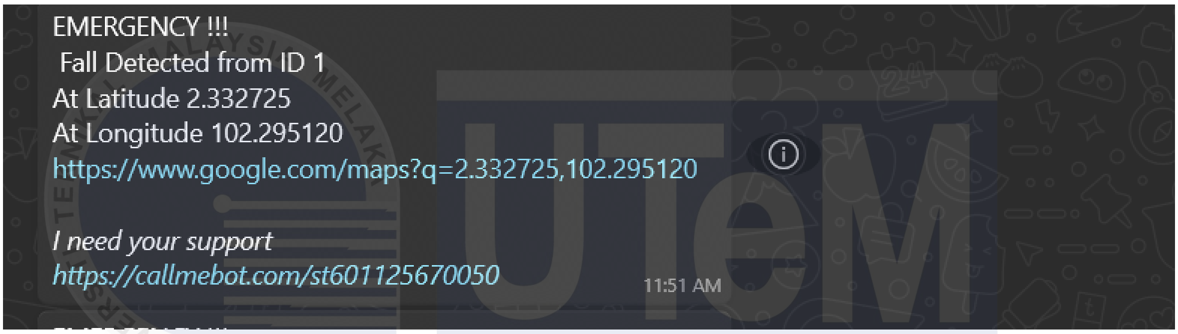


Figure 4.6 The WhatsApp message was received from the ESP32's receiver.

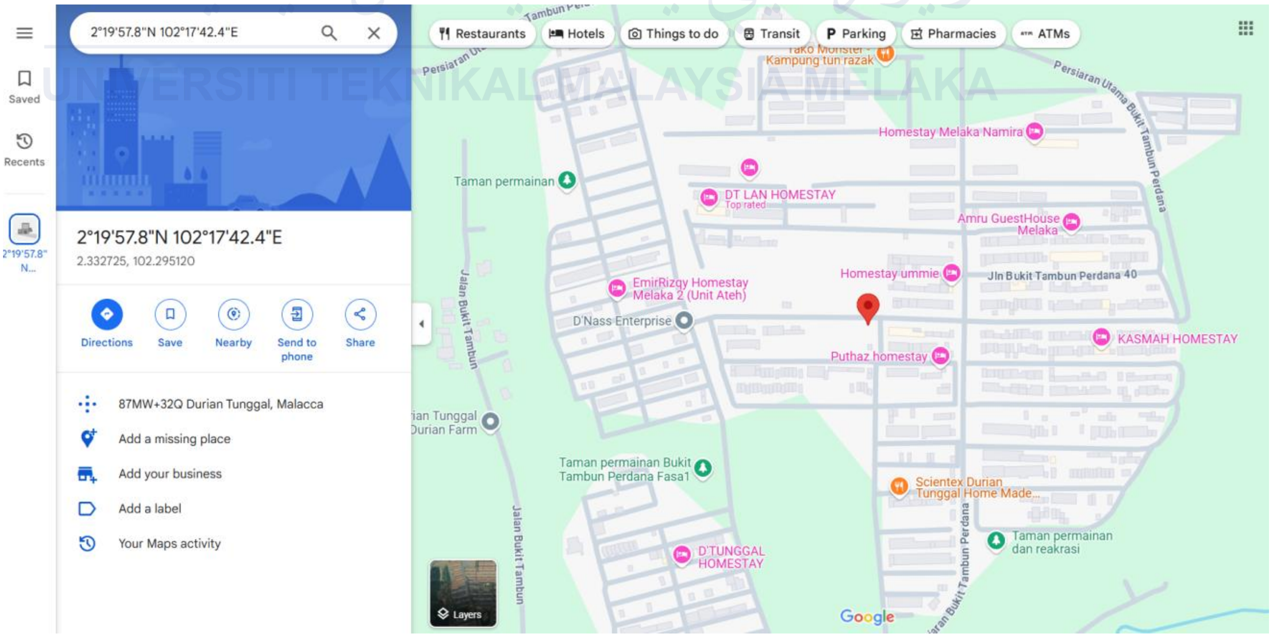


Figure 4.7 The location can be obtained via the Google Map link from WhatsApp messages.

Unfortunately, the GPS data from the Sender cannot be sent because the GPS Module at the sender sometimes had problems to connect to the satellites. Neo-6m GPS module needs at least four to ten satellites to be able to get the GPS data. In the indoors, the problems sometimes occur based on the building. If it is 2 or more floors building, the GPS data sometimes will be not accurate or will not get the GPS data from the satellites. While, when the sender is outdoors, the GPS data can be sent accurately to the receiver.

To overcome the problem, when the latitude and longitude indicated 0 or no GPS data received. It can be set that the user was at home. It is because most of the elderly are not as active as young age people who will go everywhere and move from one place to another place. The figure below indicates WhatsApp messages that will be received when there is no data received from the sender that indicates the user was in a building or home.

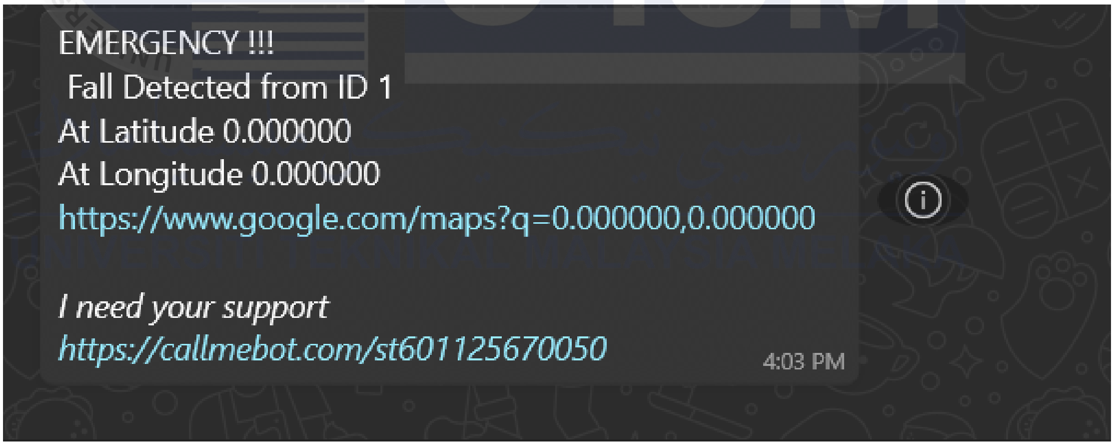


Figure 4.8 The WhatsApp messages are received when there are no GPS data.

4.3 Analysis

For analysis of the effectiveness of the fall detection system based on the TP, the true positive indicates the number of falls detected, False positive (FP) indicates false falling events detected. True negative (TN) indicates the system does not detect the falling event and False negative (FN) indicates activities of daily life (ADL) are not detected as a fall event:

Sensitivity

Sensitivity is defined as the following ratio of the number of correctly classified falls to the number of falls.

$$Sensitivity = \frac{TP}{TP + FN} \, x \, 100\%$$

$$Sensitivity = \frac{53}{60} \, x \, 100\%$$

$$Sensitivity = 88.33\%$$

Accuracy

Accuracy: The system's ability to identify the actual falls while recognising the false falls.

$$Accuracy = \frac{TP}{TP+TN} \times 100$$

Forward falls:

$$Accuracy = \frac{18}{18 + 2} \times 100$$

Accuracy = 90%

Backward falls:

$$Accuracy = \frac{18}{18 + 1} \times 100$$

Accuracy = 94%

Sideward Falls:

$$Accuracy = \frac{17}{17+1} \times 100$$

Accuracy = 94%

4.4 Summary

The result demonstrates the effectiveness of the fall detection system across various scenarios. Overall, the system achieved a sensitivity of 88.33%, indicating that the system successfully detected most of the falls during testing in a controlled environment. The overall accuracy of above 90% indicates the system's reliability to differentiate between the actual falls and non-fall events based on three types of falls which are forward fall (90% accuracy), Backward falls (94% accuracy) and sideward falls (94% accuracy). The average of response time, or latency of the system was 9 seconds which is appropriate for real-time fall detection

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This thesis explains the method of development of a fall detection system for the elderly, which applies to the field of 'Safeguarding the elderly' with immense progress. Besides, immediate alerts were generated regarding a fall through on-site warnings. The objectives of this project can be achieved which is to develop an Arduino-based fall detection system for the elderly, ensuring efficient fall identification, prompt response, and user-friendliness for healthcare providers. This project eventually enhanced the quality of life for elderly individuals by allowing them to live more independently and confidently without the constant fear of falling. Once implemented successfully, the system can help improve public health concerns in society, especially when the population around the world is ageing.

5.2 Future Works

After the development of a wearable fall detection system for the elderly, some room for improvement can make it more reliable and accurate for user experience or the Caregiver. Among the improvements that can be made are:

1. Enhancing Accuracy and Reliability: To enhance the accuracy of sensors for fall detection by integrating multiple sensors such as barometer, gyroscope, and magnetometer to differentiate between falls and non-fall activities.
2. Insert the acknowledgement in ESP-NOW protocols: To ensure the data was sent from the sender and receiver, the acknowledgement methods can make the data from the sender resend the data if the sender does not receive the data.

3. Power Optimization: Implement algorithms that can minimize power consumption by reducing unnecessary computing in programs or enabling sleep modes during static activities.
4. Real-time Communication: Develop a mobile application or web server for the caregiver to monitor the user's location and status in real-time.
5. Usability and Comfort: Design the devices as compact, lightweight, and aesthetically pleasing wearable devices such as watches, pendants, or belt clips for better user acceptance.
6. Expand the functionality: Fall detection systems, can be improved by adding more features and functions such as health monitoring systems to monitor the heart rate, oxygen saturation, or blood pressure to improve the safeguarding of the elderly.

5.3 Potential Commercialization

The commercialization of the developed fall detection system for the elderly is promising, particularly considering an ageing population and increasing cases of falls among older adults. The system features fall detection with real-time notifications, thus having high market potential because it is wearable and integrates necessary features such as GPS tracking and automation of alerts via WhatsApp. It makes it very comfortable and affordable both for personal or healthcare purposes with a performance that outperforms the majority of standard fall detection wearables. Equipped with ESP-NOW protocol to efficiently transfer the data, this system includes a buzzer and an LCD which allows immediate local alerting, even improving more usability and dependability of this system.

It can also be meant for those older adults with a high risk of falling or even those who generally have any form of mobility problems, including all active caregivers and relatives in need of providing safety for the ones they care about. The use of notifications through WhatsApp will make such immediately familiar and highly user-friendly as most likely one platform that's taken up by every caregiver. The reason why this feature will make the system particularly appealing is that not everyone can afford more complicated infrastructure for monitoring.

Commercialization potential will be strengthened due to the scalability and adaptability of the system. In this respect, the development of technology that would integrate AI and machine learning in future versions could advance the detection of falls, reduce false alarms, and enable predictive analytics. This could be just that missing link this market needs to offer a reliable, user-friendly, and cost-effective solution that will go a long way in reducing health costs attributed to falls, while at the same time reassuring users and their families.

REFERENCES

[1] C. B. Pereira and A. M. K. Kanashiro, 'Falls in older adults: a practical approach', *Arq Neuropsiquiatr*, vol. 80, 2022, doi: 10.1590/0004-282X-ANP-2022-S107.

[2] N. T. Newaz and E. Hanada, 'The Methods of Fall Detection: A Literature Review', 2023. doi: 10.3390/s23115212.

[3] Y. K. Sheng and N. Zainal, 'Development of Wearable Sensor-Based Fall Detection System for Elderly using IoT', *Evolution in Electrical and Electronic Engineering*, vol. 3, no. 2, pp. 69–79, 2022, doi: 10.30880/eeee.2022.03.02.009.

[4] S. C. S, S. H. D, S. Muteeba, and A. Professor, 'Smart Fall Detection System for Elderly People Using Arduino Uno"', 2023. [Online]. Available: www.ijrti.org

[5] T. Vaiyapuri, E. L. Lydia, M. Y. Sikkandar, V. G. Diaz, I. V. Pustokhina, and D. A. Pustokhin, 'Internet of Things and Deep Learning Enabled Elderly Fall Detection Model for Smart Homecare', *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3094243.

[6] P. Kulurkar, C. kumar Dixit, V. C. Bharathi, A. Monikavishnuvarthini, A. Dhakne, and P. Preethi, 'AI based elderly fall prediction system using wearable sensors: A smart home-care technology with IOT', *Measurement: Sensors*, vol. 25, Feb. 2023, doi: 10.1016/j.measen.2022.100614.

[7] Sri Shakthi Institute of Engineering and Technology, Institute of Electrical and Electronics Engineers. Madras Section, India Electronics & Semiconductor Association, and Institute of Electrical and Electronics Engineers, *Intelligent Accident Detection and Alert System for Emergency Medical Assistance*. 2017.

[8] H. C. Lin, M. J. Chen, C. H. Lee, L. C. Kung, and J. T. Huang, 'Fall Recognition Based on an IMU Wearable Device and Fall Verification through a Smart Speaker and the IoT', *Sensors*, vol. 23, no. 12, 2023, doi: 10.3390/s23125472.

[9] G. C. Ang, S. L. Low, and C. H. How, 'Approach to falls among the elderly in the community', *Singapore Med J*, vol. 61, no. 3, 2020, doi: 10.11622/smedj.2020029.

[10] R. Scuccato, 'Falls in the elderly', *Recenti Prog Med*, vol. 109, no. 7–8, pp. 401–404, Jul. 2018, doi: 10.1177/1755738015612824.

[11] T. G. Stavropoulos, A. Papastergiou, L. Mpaltadoros, S. Nikolopoulos, and I. Kompatsiaris, 'Iot wearable sensors and devices in elderly care: A literature review', *Sensors (Switzerland)*, vol. 20, no. 10, May 2020, doi: 10.3390/s20102826.

[12] C. L. Lin, W. C. Chiu, F. H. Chen, Y. H. Ho, T. C. Chu, and P. H. Hsieh, 'Fall Monitoring for the Elderly Using Wearable Inertial Measurement Sensors on Eyeglasses', *IEEE Sens Lett*, vol. 4, no. 6, Jun. 2020, doi: 10.1109/LSENS.2020.2996746.

[13] L. Ren and Y. Peng, 'Research of fall detection and fall prevention technologies: A systematic review', *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2922708.

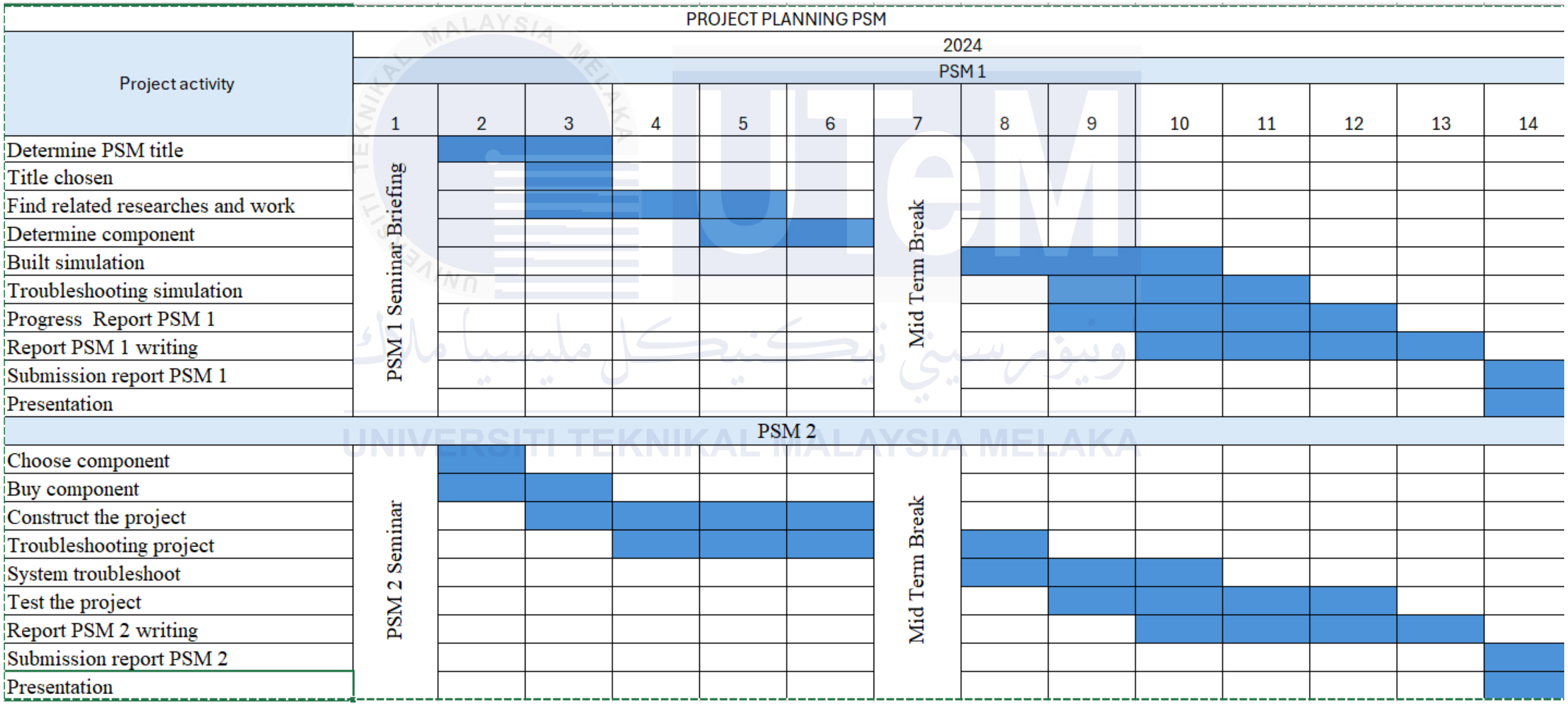
[14] T. H. Tsai and C. W. Hsu, 'Implementation of Fall Detection System Based on 3D Skeleton for Deep Learning Technique', *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2947518.

[15] F. Lezzar, D. Benmerzoug, and I. Kitouni, 'Camera-Based Fall Detection System for the Elderly With Occlusion Recognition', 2020.

[16] K. De Miguel, A. Brunete, M. Hernando, and E. Gambao, 'Home camera-based fall detection system for the elderly', *Sensors (Switzerland)*, vol. 17, no. 12, 2017, doi: 10.3390/s17122864.
[17] G. Anitha and S. BaghavathiPriya, 'Surveillance Camera Based Fall Detection System Using Long Short Term Memoryfor Elderly People', 2021. doi: 10.4108/eai.7-6-2021.2308571.
[18] M. Saleh and R. L. B. Jeannes, 'Elderly Fall Detection Using Wearable Sensors: A Low Cost Highly Accurate Algorithm', *IEEE Sens J*, vol. 19, no. 8, pp. 3156–3164, Apr. 2019, doi: 10.1109/JSEN.2019.2891128.
[19] P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti, 'A High Reliability Wearable Device for Elderly Fall Detection', *IEEE Sens J*, vol. 15, no. 8, pp. 4544–4553, Aug. 2015, doi: 10.1109/JSEN.2015.2423562.
[20] A. Kurniawan, A. R. Hermawan, and I. K. E. Purnama, 'A Wearable Device for Fall Detection Elderly People Using Tri Dimensional Accelerometer', 2016.
[21] I. Goldwin, A. Gunadhi, D. Lestariningsih, H. Pranjoto, P. R. Angka, and L. Agustine, 'Fall detection and notification system to fast emergency management for the elderly', in *E3S Web of Conferences*, EDP Sciences, Jan. 2024. doi: 10.1051/e3sconf/202447502014.
[22] X. Wang, J. Ellul, and G. Azzopardi, 'Elderly Fall Detection Systems: A Literature Survey', Jun. 23, 2020, *Frontiers Media S.A.* doi: 10.3389/frobt.2020.00071.
[23] S. K. Bhoi *et al.*, 'FallDS-IoT: A Fall Detection System for Elderly Healthcare Based on IoT Data Analytics', in *Proceedings - 2018 International Conference on Information Technology, ICIT 2018*, Institute of Electrical and Electronics Engineers Inc., Dec. 2018, pp. 155–160. doi: 10.1109/ICIT.2018.00041.
[24] T. P. Filgueiras, C. R. P. Torres, and P. B. Filho, 'Low Cost System for Fall Detection in the Elderly', in *Proceedings - IEEE 20th International Conference on Bioinformatics and Bioengineering, BIBE 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 697–700. doi: 10.1109/BIBE50027.2020.00117.
[25] X. Hu and X. Qu, 'Pre-impact fall detection', 2016. doi: 10.1186/s12938-016-0194-x.
[26] H. Ramirez, S. A. Velastin, I. Meza, E. Fabregas, D. Makris, and G. Farias, 'Fall Detection and Activity Recognition Using Human Skeleton Features', *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3061626.
[27] L. La Blunda, L. Gutiérrez-Madroñal, M. F. Wagner, and I. Medina-Bulo, 'A Wearable Fall Detection System Based on Body Area Networks', *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3032497.
[28] Shrutika S. Shete, Rambabu A. Vatti, and Karishma V. Raigandhi, *Dead Zone Identification In IEEE 802.11G Network*. Institute of Electrical and Electronics Engineers, 2017.
[29] N. E. Bassett *et al.*, 'Dead Zones: A Phenomenology of Disconnection', 2020.
[30] R. Igual, C. Medrano, and I. Plaza, 'Challenges, issues and trends in fall detection systems', *Biomed Eng Online*, vol. 12, no. 1, 2013, doi: 10.1186/1475-925X-12-66.
[31] E. Casilari, J. A. Santoyo-Ramón, and J. M. Cano-García, 'Analysis of public datasets for wearable fall detection systems', *Sensors (Switzerland)*, vol. 17, no. 7, 2017, doi: 10.3390/s17071513.

APPENDICES

APPENDIX A



APPENDIX B

Sender

```
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <TinyGPS++.h>
#include <esp_now.h>
#include <WiFi.h>
#include <esp_mac.h>
#include <HardwareSerial.h>

//#define I2C_SDA 4
//#define I2C_SCL 5
#define GPS_BAUD 9600

HardwareSerial gpsSerial(0); // Use Serial0 (GPIO 16 for RX, GPIO 17 for TX)
Adafruit_MPU6050 mpu;
TinyGPSPlus gps;

// REPLACE WITH YOUR RECEIVER MAC Address
uint8_t broadcastAddress[] = {0x10, 0x06, 0x1c, 0x85, 0xa0, 0x6c};

// Structure example to send data
// Must match the receiver structure
typedef struct struct_message {
  int id;
  float x;
  float y;
} struct_message;

// Create a struct_message called myData
struct_message myData;

esp_now_peer_info_t peerInfo;

// callback when data is sent
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
  Serial.print("\r\nLast Packet Send Status:\t");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" :
"Delivery Fail");
}

const int MPU_addr = 0x68;  // I2C address of the MPU-6050
int16_t AcX, AcY, AcZ, Tmp, GyX, GyY, GyZ;
float ax=0, ay=0, az=0, gx=0, gy=0, gz=0;
bool fall = false; //stores if a fall has occurred
bool trigger1=false; //stores if first trigger (lower threshold) has occurred
```

```
bool trigger2=false; //stores if second trigger (upper threshold) has occurred
bool trigger3=false;//stores if third trigger (angle changes) has occurred

byte trigger1count=0; //stores the counts past since trigger 1 was set true
byte trigger2count=0; //stores the counts past since trigger 2 was set true
byte trigger3count=0; //stores the counts past since trigger 2 was set true
int angleChange=0;

const int Button = D10;

void setup() {
  Wire.begin(D4,D5);
  Wire.beginTransmission(MPU_addr);
  Wire.write(0x6B);  // PWR_MGMT_1 register
  Wire.write(0);     // Set to zero (wakes up the MPU-6050)
  Wire.endTransmission(true);
  // Init Serial Monitor
  Serial.begin(115200);
  // Start Serial 2 with the defined RX and TX pins and a baud rate of 9600
  gpsSerial.begin(GPS_BAUD, SERIAL_8N1, -1, -1);

  // Set device as a Wi-Fi Station
  WiFi.mode(WIFI_STA);

  // Init ESP-NOW
  if (esp_now_init() != ESP_OK) {
    Serial.println("Error initializing ESP-NOW");
    return;
  }
  // Once ESPNow is successfully Init, we will register for Send CB to
  // get the status of Trasnmitted packet
  esp_now_register_send_cb(OnDataSent);
  // Register peer
  memcpy(peerInfo.peer_addr, broadcastAddress, 6);
  peerInfo.channel = 0;
  peerInfo.encrypt = false;

  // Add peer
  if (esp_now_add_peer(&peerInfo) != ESP_OK){
    Serial.println("Failed to add peer");
    return;
  }

  pinMode(Button, INPUT_PULLUP); // Assuming a pull-down resistor is used with the
button
}

void loop(){
  mpu_read();
  gps_read();
```

```
 int buttonStateD1 = digitalRead(Button);
 // values may be different for you
 ax = (AcX-2050)/16384.00;
 ay = (AcY-77)/16384.00;
 az = (AcZ-1947)/16384.00;

//270, 351, 136 for gyroscope
 gx = (GyX+270)/131.07;
 gy = (GyY-351)/131.07;
 gz = (GyZ+136)/131.07;
// calculating Amplitute vactor for 3 axis

 float raw_amplitude = pow(pow(ax,2)+pow(ay,2)+pow(az,2),0.5);
 int acceleration = raw_amplitude * 10;  // Mulitiplied by 10 bcz values are between 0 to
1
 Serial.print("Acceleration: ");
 Serial.println(acceleration);
 delay(500);
 int angleChange = pow(pow(gx,2)+pow(gy,2)+pow(gz,2),0.5);
 Serial.print("Angle Changes: ");
 Serial.println(angleChange);
 delay(500);

 if (acceleration <=11 && trigger3==false){ //if AM breaks lower threshold (0.4g)
   trigger1=true;
   trigger2=true;
   Serial.println("NORMAL");
   }
 if (trigger1==true){
   trigger1count++;
   if (acceleration>= 20){ //if AM breaks upper threshold (3g)
     trigger3=true;
     Serial.println("HIGH ACCELERATION DETECTED");
     trigger1=false;
     trigger1count=0;
     }

   if (angleChange>=200){ //if AM breaks upper threshold (3g)
     trigger3=true;
     Serial.println("MASSIVE ANGLE CHANGES DETECTED");
     trigger2=false;
     trigger2count=0;
     }
 }
 if (trigger3==true){
   trigger3count++;
    if (trigger3count>=4){
       if ((angleChange>=0) && (angleChange<=30)){ //if orientation changes remains
between 0-10 degrees
```

```
          fall=true;trigger3count=0;
          }
       else{ //user regained normal orientation
          trigger3=false; trigger3count=0;
         }
       }
     }
 if (fall == true){
   Serial.println("FALL DETECTED");
   Serial.print("Latitude: ");
   Serial.println(gps.location.lat(), 6); // Print latitude with 6 decimal places
   Serial.print("Longitude: ");
   Serial.println(gps.location.lng(), 6);
   senddata1();
      if (buttonStateD1 == LOW) {
        Serial.println("Button Pressed - Fall condition reset");
        fall = false; // Reset the fall state
        trigger3 = false;
     } else{
      fall = true;
     }
   }

//It appears that delay is needed in order not to clog the port
 //(200);
}

void mpu_read(){
    Wire.beginTransmission(MPU_addr);
    Wire.write(0x3B);  // Starting with register 0x3B (ACCEL_XOUT_H)
    Wire.endTransmission(false);
    Wire.requestFrom(MPU_addr, 14, true);  // Request a total of 14 registers
    AcX = Wire.read() << 8 | Wire.read();  // 0x3B (ACCEL_XOUT_H) & 0x3C
(ACCEL_XOUT_L)
    AcY = Wire.read() << 8 | Wire.read();  // 0x3D (ACCEL_YOUT_H) & 0x3E
(ACCEL_YOUT_L)
    AcZ = Wire.read() << 8 | Wire.read();  // 0x3F (ACCEL_ZOUT_H) & 0x40
(ACCEL_ZOUT_L)
    Tmp = Wire.read() << 8 | Wire.read();  // 0x41 (TEMP_OUT_H) & 0x42
(TEMP_OUT_L)
    GyX = Wire.read() << 8 | Wire.read();  // 0x43 (GYRO_XOUT_H) & 0x44
(GYRO_XOUT_L)
    GyY = Wire.read() << 8 | Wire.read();  // 0x45 (GYRO_YOUT_H) & 0x46
(GYRO_YOUT_L)
    GyZ = Wire.read() << 8 | Wire.read();  // 0x47 (GYRO_ZOUT_H) & 0x48
(GYRO_ZOUT_L)
}

void gps_read(){
  // GPS
```

```
unsigned long start = millis();
  while (millis() - start < 1000) {
    while (gpsSerial.available() > 0) {
      gps.encode(gpsSerial.read());
    }
  delay(200);
}
}

void senddata1(){
  myData.id = 1;
  myData.x = gps.location.lat();
  myData.y = gps.location.lng();
  // Send message via ESP-NOW
  esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &myData,
sizeof(myData));

  if (result == ESP_OK) {
    Serial.println("Sent with success");
  }
  else {
    Serial.println("Error sending the data");
  }
  delay(2000);
}
```

APPENDIX C

Receiver

```
#include <WiFi.h>
#include <esp_now.h>
#include <HTTPClient.h>
#include <UrlEncode.h>
#include <LiquidCrystal_I2C.h>

#define Buzzer 5

// Replace with your CallMeBot API details
String phoneNumber = "601125670050"; // Include country code
String apiKey = "2363977"; // Your CallMeBot API key

static const char* ssid = "Really frfrfr";
static const char* password = "freaddy435";

bool isDatarecv = false;
// Initialize the LCD
LiquidCrystal_I2C lcd(0x27, 16, 2); // Change the address if needed

// Structure to hold incoming data
typedef struct struct_message {
   int id;          // Integer
   float x;   // Float 1
   float y;   // Float 2
} struct_message;

struct_message myData;

// Callback function to handle incoming ESPNOW data
void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
   memcpy(&myData, incomingData, sizeof(myData));
   Serial.printf("ID %u", myData.id);
  // Update the structures with the new incoming data
   Serial.printf("x LATITUDE: ");
   Serial.println(myData.x);
   Serial.printf("y LONGITUDE: ");
   Serial.println(myData.y);
   isDatarecv = true;
}

// Function to send WhatsApp message using CallMeBot API
void sendWhatsAppMessage(const String &message) {

  WiFi.disconnect();       // Stop ESP-NOW temporarily
  WiFi.begin(ssid, password);
```

```
  //check wi-fi is connected to wi-fi network
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.print(".");
  }
    if (WiFi.status() == WL_CONNECTED) {
    String url = "https://api.callmebot.com/whatsapp.php?phone=" + phoneNumber +
"&apikey=" + apiKey + "&text=" + urlEncode(message);
    HTTPClient http;
    http.begin(url);

  // Specify content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

  // Send HTTP POST request
    int httpResponseCode = http.POST(url);
    if (httpResponseCode == 200){
    Serial.print("Message sent successfully");
    }
    else{
      Serial.println("Error sending the message");
      Serial.print("HTTP response code: ");
      Serial.println(httpResponseCode);
    }

  // Free resources
    http.end();
}
    else {
        Serial.println("WiFi Disconnected");
    }
    //wifi disconnect
    WiFi.disconnect();
    //initialize espnow
    while (esp_now_init() != ESP_OK) {
    Serial.println("Reinitializing ESPNOW...");
    delay(1000);
  }

  // Register the receive callback again
  esp_now_register_recv_cb(esp_now_recv_cb_t (OnDataRecv));
}

void setup() {
  Serial.begin(115200);
    // Set up WiFi
  WiFi.mode(WIFI_STA);
    // Initialize ESPNOW
  if (esp_now_init() != ESP_OK) {
      Serial.println("Error initializing ESPNOW");
```

```
    return;
   }

   // Register the receive callback
  esp_now_register_recv_cb(esp_now_recv_cb_t (OnDataRecv));
  delay(1000);

 // WiFi.disconnect();
  lcd.init();
  lcd.backlight();
  pinMode(Buzzer, OUTPUT);
  digitalWrite(Buzzer,LOW);
}

void loop() {
  if (isDatarecv == true){
   // Display on LCD
   lcddisplay();
   isDatarecv = false;
   // Send data via WhatsApp
   String message = "EMERGENCY !!!\n Fall Detected from ID "+ String(myData.id) +
"\nAt Latitude " + String(myData.x, 6) + "\nAt Longitude " + String(myData.y, 6) +
"\nhttps://www.google.com/maps?q=" + String(myData.x, 6) +","+ String(myData.y, 6);
   sendWhatsAppMessage(message);

   for (int i = 0; i < 10; i++){
    digitalWrite(Buzzer, HIGH);
    delay(1000);
    digitalWrite(Buzzer, LOW);
    delay(100);}

   delay(2500);
  }
  else{
    lcd.print("Everyone Good");
    delay(1000);
    lcd.clear();
    delay(1000);
    digitalWrite(Buzzer, LOW);
  }
}

void lcddisplay(){
  lcd.setCursor(0, 0);
  lcd.print("EMERGENCY !");
  lcd.setCursor(0, 1);
  lcd.print("ID " + String(myData.id) + " fall");
  delay(2000);
  lcd.clear();
```

```
lcd.setCursor(0, 0);
lcd.print("Lat: ");
lcd.print(myData.x, 6);
lcd.setCursor(0, 1);
lcd.print("Long: ");
lcd.print(myData.y, 6);
delay(2000);
lcd.clear();
}
```