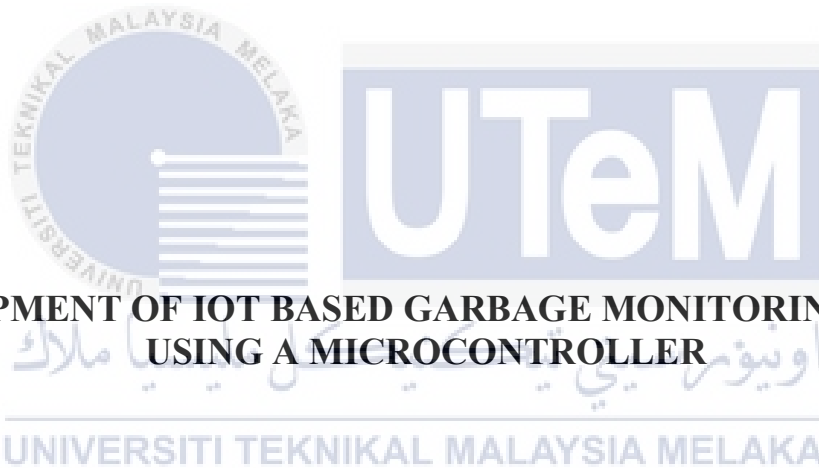**Faculty of Electrical and Electronic Engineering Technology**

DEVELOPMENT OF IOT BASED GARBAGE MONITORING SYSTEM USING A MICROCONTROLLER

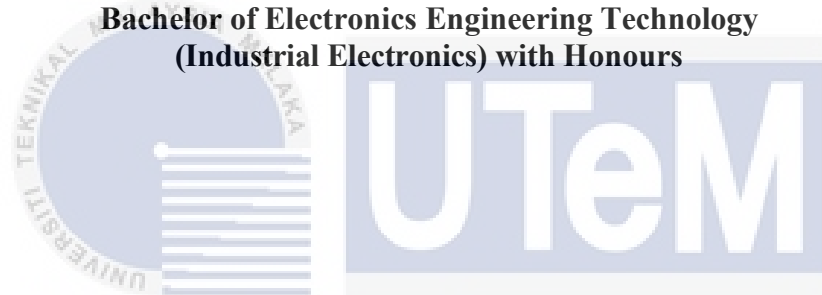**NUR AZREEN ATIKAH BINTI ZULKIFLEE**

Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours

2023

DEVELOPMENT OF IOT BASED GARBAGE MONITORING SYSTEM USING A MICROCONTROLLER

NUR AZREEN ATIKAH BINTI ZULKIFLEE

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

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2023

DECLARATION

I declare that this project report entitled "Development of IoT Based Garbage Monitoring System Using A Microcontroller" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature   :

Student Name   :   Nur Azreen Atikah Binti Zulkiflee

Date   :   13 January 2023

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 ElectronicsEngineering Technology (Industrial Electronics) with Honours.

Signature   :

Supervisor Name : Syed Mohamad Shazali Bin Syed Abdul Hamid

Date   : 13 January 2023

DEDICATION

This research is dedicated to my parents, Zulkiflee Bin Subari and Aidah Binti Siam, who have always encouraged me. They have given me the discipline support and motivation I need to approach this task with eagerness and dedication. Without their love and support, this project would not have been possible.

ABSTRACT

The neat and clean surroundings are the main driving force for any university residential college to be called a "clean university." The increasing number of students in universities generates a bunch of garbage. As a result, trash collection and garbage management in universities have to deal with this critical issue, and this matter must be addressed by university administrators. To maintain the residential college's clean environment, the residential college's management needs to take care of the cleanliness of the bins placed on every level of the residential college's block by checking and collecting the garbage at every single dustbin daily. This will require more human work and time. Other than that, the garbage will overflow and a bad smell will arise if the cleaning staff does not come to collect the garbage right on time because they do not know the status of the dustbin. So, this project's concept is to design the "Development of an IoT-Based Garbage Monitoring System Using Microcontroller" by continuously monitoring the garbage level and notifying cleaning staff wirelessly by sending a notification via the Blynk app on their smartphone every time the garbage bin is full. By using this system, the issue of garbage overflow can be solved. The suggested system included ultrasonic sensors that were strategically positioned under the lid of the dustbin to identify the garbage level and compare it to the depth of the dustbins. To save data, the system displays the status of garbage on a liquid crystal display (LCD) and Blynk application in real time and sends an alert notification to the cleaning staff. The system is expected to create a cleaner and more comfortable environment due to its ability to monitor and control garbage collection intelligently through the Internet of Things (IoT).

ABSTRAK

Keadaan persekitaran yang kemas dan bersih menjadi penggerak utama bagi mana-mana kolej kediaman universiti untuk digelar sebagai "universiti bersih". Peningkatan bilangan pelajar di universiti menjana sekumpulan sampah. Akibatnya, kutipan sampah dan pengurusan sampah di universiti terpaksa berhadapan dengan isu kritikal ini dan perkara ini perlu ditangani oleh pihak pentadbir universiti. Bagi menjaga kebersihan persekitaran kolej kediaman, pihak pengurusan kolej kediaman perlu menjaga kebersihan tong sampah yang diletakkan di setiap peringkat blok kolej kediaman dengan memeriksa dan mengutip sampah di setiap tong sampah setiap hari. Ini akan memerlukan lebih banyak kerja dan masa manusia. Selain itu, sampah akan melimpah dan timbul bau busuk sekiranya petugas pembersihan tidak datang mengutip sampah tepat pada masanya kerana tidak mengetahui status tong sampah. Jadi, konsep projek ini memperkenalkan untuk mereka bentuk "Pembangunan Sistem Pemantauan Sampah Berasaskan IoT Menggunakan Mikropengawal" dengan memantau tahap sampah secara berterusan dan memaklumkan secara wayarles kepada kakitangan pembersihan dengan menghantar pemberitahuan melalui aplikasi Blynk pada telefon pintar mereka setiap kali tong sampah penuh. Dengan menggunakan sistem ini, isu limpahan sampah dapat diselesaikan. Sistem yang dicadangkan termasuk penderia ultrasonik yang diletakkan secara strategik di bawah penutup tong sampah untuk mengenal pasti paras sampah dan membandingkannya dengan kedalaman tong sampah. Untuk menjimatkan data, sistem memaparkan status sampah pada paparan kristal cecair (LCD) dan aplikasi Blynk dalam masa nyata dan menghantar pemberitahuan amaran kepada kakitangan pembersihan. Sistem ini dijangka dapat mewujudkan persekitaran yang lebih bersih dan nyaman kerana berkebolehan untuk memantau dan mengawal kutipan sampah secara bijak melalui Internet of Things (IoT).

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

LIST OF SYMBOLS

kg   -   kilogram

s   -   second

m   -   meter

cm   -   centimeter

mm   -   millimeter

V   -   voltage

A   -   ampere

M   -   Mega

Hz   -   Hertz

LIST OF ABBREVIATIONS

IoT	Internet of Things
GSM	Global System for Mobile Communications
LCD	liquid-crystal display
IR	Infrared
VCC	Voltage Common Collector
VDD	Voltage Drain
RX	Receiver
TX	transmitter
PWM	Pulse Width Modulation
Vin	Voltage Input
GND	Ground
USB	Universal Serial Bus
I/O	Input and Output
IDE	Integrated Development Environment
API	Application Program Interface
IDE	Integrated Development Environment
Wi-Fi	Wireless Fidelity

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1   Background

Nowadays, everything must be completed in a timely and effective manner. The Internet of Things has emerged as the most popular innovation right now. IoT is nothing more than a technology that connects all physical objects, mostly through wireless networks, with the least amount of human intervention possible to keep the environment clean. With the influence of the Internet of Things (IoT), tasks and systems are fusing to create a more rapid and efficient working system. The IoT is an emerging paradigm that is a crucial part of our lives. The term "Internet of Things" was coined by Kevin Ashton in 1999, when he included it in the title of a presentation he made at Procter & Gamble [1]. For example, with the implementation of smart devices, it is possible to develop a garbage monitoring system that can monitor the level of garbage in a dustbin through the IoT. This shows that IoT is becoming essential to our lives.

The management of solid waste, which affects the environment and health of our society, has been one of the key environmental problems. The goal of smart or interconnected gadgets is to close the gap between the real world and the digital one, raising living standards in the process. The creation of a "smart city," where machines will regularly interact with people to carry out numerous daily tasks, is the most significant application. Our goal is to create an IoT garbage monitoring system that will ensure garbage collection is done on time and the problem of overflowing garbage can be avoided. The proposed system should overcome such problems by alerting users to the status of garbage in the dustbin.

1

In today's world, efficient garbage disposal has become a major source of concern in many Malaysian universities and institutions. Due to the ever-increasing number of students, the amount of waste generated has also increased. However, due to ineffective management, we observed a lot of rubbish spilling out of the dustbin, which has left the area smelling unpleasant. If this garbage is not disposed of promptly, various illnesses may spread. A dustbin is considered a basic need to maintain the rate of cleanliness in residential colleges, so it is very important to clean all the dustbins as soon as they get filled. Garbage is classified into two types: dry waste and wet waste. Wet waste is any garbage that is made from food leftovers. In residential colleges, there is a very common situation where the garbage is overloaded and spilled out. This eventually leads to pollution.

Waste management in residential colleges at this time is still limited. The cleaner will clean up at a specified time according to the schedule. This is very ineffective because the delay in garbage collection will cause the garbage in the dustbin to overflow and stink since the dustbin was full before the garbage collection schedule. To maintain the residential college's clean environment, the residential college's management needs to take care of the cleanliness of the bins placed on every level of the college block. So, garbage monitoring is critical for optimizing management and resources and reducing the staff required to conduct garbage collection activities. The traditional method of manually monitoring the garbage in the dustbin is a time-consuming procedure that necessitates more human labor, time, and money, all of which can be avoided with today's technologies, which provide a real-time indication of the garbage level in a dustbin at any given time by monitoring it via the Blynk app on the smartphone. With this technology, the number of insects, germs, and viruses that may infect people would increase rapidly as a result of the waste volume created by ineffective waste management.

Smartphones have become a highly frequent resource and the primary means of communication for everyone in the world to communicate with each other. So, of course, everyone has a smartphone. Therefore, we can take advantage of the existing smartphone by developing a garbage monitoring system that can monitor the level of garbage in the dustbin using only a smartphone. This appears to save them time. With the benefit of cleaners collecting garbage on time, the rate of cleanliness at residential colleges would rise proportionally. Smartphone development encourages users to prefer mobile apps. IoT allows communication between networking devices based on requirements. This mobile app was created to allow you to check the current level of garbage in the trash can wherever you are.

1.2   Problem Statement

As students, we can see that there is a lot of garbage at the residential colleges. This is because there is no novel or systematic method for monitoring the level of garbage in dustbins so that garbage can be collected promptly [4]. The detection, monitoring, and management of garbage are some of the primary problems of the present era. The cleaning staff must check every level of each residential college block to collect the garbage daily. The task will become more tedious if the residential college block has many levels and requires more human work and time. Besides, if the cleaning staff does not come to collect the garbage on time because they do not know about the status of the dustbin, the garbage will overflow and a bad smell will arise. Other than that, sometimes the garbage falls at area around the dustbin while trying to open the dustbin lid to throw away the garbage. These problems are sanitary issue that might cause diseases and may raise the chance of contracting diseases such as cholera, dengue fever, typhoid fever, food poisoning, gastroenteritis, and other serious illnesses caused by flies.

1.3 Project Objective

The objectives of this project are:

a) To design a prototype dustbin with an IoT-based garbage monitoring system that can monitor the level of the garbage in the bin and display the status of the dustbin in Blynk and at LCD attached to the dustbin.

b) To create a prototype dustbin with an automatic lid that opens and closes.

c) To develop a smart alert system for garbage clearance by giving notifications via the Blynk application to the cleaner's mobile phone.

1.4 Scope of Project

To avoid any uncertainty about this project due to some limitations and constraints, the scope of the project is defined as follows:

a) In this project, the NodeMCU microcontroller serves as the brain, controlling the components and sensors.

b) ESP8266 WiFi was used for communication between the NodeMCU and the mobile application developed using the Blynk application to monitor the garbage level and notify the person in charge when the dustbin is full.

c) Using two ultrasonic sensors, one is placed on the lid of the dustbin to detect the level of garbage, and another is placed in front of the dustbin to make the lid of the dustbin open and close automatically.

d) This dustbin is implemented for localized and small-scale cases, such as residential colleges.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, an IoT-based garbage monitoring system has been explained from a selection of papers and journals that describe comparable approaches and subjects. The flow of the study started with collecting sources, for instance, journal articles, books, documents from websites, and conference papers, and analyzing them to extract useful information. The information collected was tabulated to show the comparison between each find clearly. Other researchers have compared their comparable current system, identical sensors, and IoT board. Lastly, the analyzed and synthesized information was evaluated by comparing their trade-offs to find the best approach that can be applied to this project. The outcome of the evaluation was then summarised and written about in the last part of this section.

2.2 Research

2.2.1 Monitoring System

Extensive research has been done on waste management throughout the world. The majority of them were concerned with manual garbage collection, distribution, and recycling. People have begun to use technology in recent years to monitor and effectively collect garbage. The essential premise of a monitoring system is to allow users to acquire data, process it, and disseminate it systematically. Based on data obtained in the field, the monitoring system allows us to measure trends in numerous variables. An effective and efficient monitoring system is required to examine system activity, determine if applications are running well, and inform the user if necessary.

5

2.3 Review and Comparison of Previous Work

2.3.1 Wireless Communication Technologies

Wireless communication technology is information that can send out data across long distances without much hindrance.

Table 2.1   Comparison between GSM, Wi-Fi, and Bluetooth

NO.	Name	Operating Frequency	Range	Transfer Rate	Cost	Power
1.	GSM [2][4][6]	850 - 900MHz	2 – 35 Km	168 Kbps	High	Low
2.	Wi-Fi [1][5]	2.4 - 5GHz	10 – 100 m	11 Mbps	Low	High
3.	Bluetooth [11]	2.4 – 2.485 GHz	10 m	1 Mbps	Low	Low

Based on table 2.1, a GSM module is used to communicate between a computer and a GSM system. This wireless communication technology will incur a lot of monetary issues and also have issues with the database and usage [27]. GSM needs to solve the connecting issue. They need to consider all the garbage bins and apply a SIM card to each of the garbage systems. This raises the budget for data subscriptions [27]. Besides, it generally requires a lot of energy to operate since it needs to communicate with the user server consistently. As for Bluetooth, it is generally used for short-range communication [28]. It works at 2.4 to 2.485 GHz, consumes very little power, is available at a very economical price, and is a very simple yet effective communication technology. Bluetooth Low Energy (BLE) and Bluetooth 4.0 are the latest versions of this technology, which consume much less power than the former version [28]. Wi-Fi is a communication technology used to connect devices wirelessly, provide internet access, and also to connect different devices to a wired network. Wi-Fi can be used to share the internet, files, and other resources between devices. It has a range of more than 100 meters and operates at either 2.4 or 5 GHz. This high frequency can carry more data [28].

6

2.3.2 Type of Sensors

From previous research, there are several techniques for detecting the level of garbage in the trash can by employing various types of sensors. IR and ultrasonic sensors are active sensors that need power to operate [22]. Each sensor has its unique strengths and weaknesses. Table 1 below lists some of the sensors used in the prior study.

Table 2.2   Comparison between Ultrasonic Sensor, IR Sensor, and Weight Sensor

Point of Comparison	Ultrasonic Sensor [34] [1] [2] [3] [4] [5] [6]	IR Sensor [3] [10] [20]	Weight Sensor [20]
Function	• Determines the quantity of garbage in the dustbin. • To detect the presence of people throwing garbage into the dustbins	• Determine how much garbage is in the dustbin. • To detect the presence of people throwing garbage into the dustbin	• To detect the weight of the garbage in the dustbin.
Measuring Distance	< 16 meters	< 6 meters	-
Resolution	High	Variable according to distance	-
Detecting Angle Range	Narrow or wide according to its model	Small	-
Output Linearity	Linear	Non-Linear	-
Mobility	Portable	Portable	-

Table 2.2   Comparison between Ultrasonic Sensor, IR Sensor, and Weight Sensor

Point of Comparison	Ultrasonic Sensor [34] [1] [2] [3] [4] [5] [6]	IR Sensor [3] [10] [20]	Weight Sensor [20]
Price	RM 3.30	RM 1.90	RM 5.20
Advantages	• Accurate distance measurement. • It can work in direct sunlight. • Excellent performance whether indoors or outdoors.	• Less expensive Low power consumption	• It is securely implanted under the dustbin rather than affixed to a cover.
Disadvantages	• May become inaccurate when encountering an absorbing obstacle • More expensive than other similar sensors	• Not accurate ranging • Narrow beam width Cannot be used while exposed to the sun	• Inaccurate measurement Cannot detect the level

8

From table 2.2 above, there are three sensors used from previous research, which are ultrasonic, IR, and weight sensors. The selection of the sensor is based on the following criteria: accuracy (how close the reading is to the real distance) and resolution (the minimum or maximum fluctuation of a reportable reading). Ultrasonic and infrared sensors are commonly employed in navigation systems for people, mobile robots, and vehicle-related applications for contact-free, mid-range distance measurements [19].

Ultrasonic can be chosen if the project needs an accurate and numerical representation of distance. The advantages of an ultrasonic sensor are that it is not affected by object color and transparency and cannot be affected by any environmental factor like dark or light, dust, rain, or snow [22]. Besides, an ultrasonic sensor can measure the distance with high accuracy and stable readings [5]. It can measure distances from 2 cm to 400 cm [5]. The downsides of ultrasonic sensors include that some things, such as sloped surfaces that divert the echo away from the sensor or permeable targets such as sponges, foam, and soft clothes, can make detection more difficult. Curved or slanted objects can scatter the bulk of ultrasonic waves directed at them, resulting in a feeble echo return. Larger angular deviations are possible on round or stiff surfaces.

IR sensors show the level of waste in the dustbin, and three different levels of IR sensors are planted on the surface of the dustbin to display the true quantity of garbage present. [20]. This sensor is equivalent to human visual senses and may be used to identify obstacles, making it one of the most prevalent real-time applications [21]. The range of the IR sensor can be adjusted by rotating the potentiometer. If the project only needs to know the present location of an object, then an IR sensor is easier to implement. The IR sensor is good at defining the edges of an area. The weakness of the IR sensor is that its reading can be affected by the color of the obstacle material [20]. There is a downside to utilizing an IR sensor since its results are unsatisfactory in low-light environments. An IR-detecting component locates bright surface

9

things or objects better than dim surface things or objects [22]. An infrared sensor cannot function in low-light conditions [22]. It can also control only one device at a time.

From the previous work, some projects use weight sensors. To determine the weight of the garbage in the bin, a weight sensor is installed beneath the dustbin. [20]. When the IR sensor's threshold level is exceeded, the weight sensor is engaged to transmit its signal [20]. The LOAD cell will continually provide weight values in voltage format. This sensor cannot be used to measure the level of garbage, and its measurements are inaccurate.

2.3.3   Type of Microcontroller

Sensors are devices that convert physical quantities into electrical currents. Therefore, a microcontroller must be used to transform it into information that is understandable for the user.

Table 2.3   Comparison between NodeMCU and Arduino

Points of comparison	ESP8266 NodeMCU [34][25][35]	ESP32 Node MCU [36]	Arduino UNO [4] [5] [9] [11] [12] [13]	Arduino Mega [37]
Microcontroller	ESP8266	ESP32	ATmega328p	ATmega2560
Operating Voltage	3.3V	3.3V	5V	5V
Power supply	7V – 12V	7V – 12V	7V – 12V	7V – 12V
Current consumption	15 µA – 400 mA	20 mA – 240 mA	45 mA – 80 mA	50 mA – 200 mA
Digital I/O pins	11 pins	36 pins. 36 with PWM	14 pins. 6 with PWM	54 pins. 15 with PWM
Analog Input Pins	1 pin	15 pins	6 pins	16 pins
Flash Memory	4 MB	4 MB	32 KB	256 KB
Wi-Fi	Yes	Yes	No	No
Bluetooth	No	Yes	No	No
Programming Language	C, C++	C, C++	C++	C, C++
Price	RM 14.90	RM 29.00	RM 29.00	RM 90.00

Based on table 2.3 above, we can see that four types of microcontrollers have been used in previous research, which are the ESP8266 NodeMCU, ESP32 NodeMCU, Arduino UNO, and Arduino Mega. NodeMCU is a microcontroller used as a controller on the other side that receives control commands through a Wi-Fi signal. NodeMCU has a built-in ESP8266 or ESP32 Wi-Fi receiver that can process and analyze Wi-Fi signals and input them into the microcontroller. The Wi-Fi receiver and microcontroller are integrated into one kit for Internet of Things projects. It goes by the name "NodeMCU." The NodeMCU is programmed using the Arduino Integrated Development Environment (IDE) and the Universal Serial Bus interface (USB) to instruct it on what to do.

The ESP32 is the ESP8266's successor [38]. The ESP32 has built-in temperature and hall effect sensors, as well as touch-sensitive pins. Although the ESP32 is significantly more expensive, both boards are quite affordable [38]. In addition to having more GPIOs with greater capabilities, the ESP32 is significantly more powerful than the ESP8266 and supports Bluetooth. It also has quicker Wi-Fi. Because it is more sophisticated than the ESP8266, the ESP32 is often perceived as being more difficult to operate. Unlike NodeMCU, which only supports 3.3V modules, Arduino UNO supports both 3.3V and 5V modules, making it simple to interact with various sensors and actuators.

2.3.4 Platform

ThingSpeak is an open-source IoT platform comprised of a central server that enables data gathering, processing, and analysis, as well as libraries and APIs to support IoT devices [24]. It is an open internet program that allows us to see a graphical depiction of any project [25]. This platform may be deployed locally or in the cloud, and it offers both free and paid options. [24]. Besides, it is compatible with a broad range of devices, including smartphones and Internet of Things hardware platforms like Arduino, Raspberry Pi, Electric Imp, Particle Photon, and ESP8266 [24].

ThingSpeak distinguishes itself with its innovative data processing and visualization capabilities [24]. The Blynk app is a platform that comprises iOS and Android applications for remotely controlling Arduino, Raspberry Pi, and other similar devices. It is a digital dashboard that allows you to construct a graphical interface for your project by dragging and dropping widgets.

Blynk is not restricted to a single board or shield. Instead, it's supporting the hardware of your choice. Blynk will get you online and ready for the Internet of Things, whether your Arduino or Raspberry Pi is connected to the Internet by Wi-Fi, Ethernet, or this new ESP8266 chip. Blynk is intended for use on the Internet of Things. It can operate hardware remotely, show sensor data, save data, visualize it, and perform a variety of other interesting things.

MIT App is an open-source Android application that uses the Firebase and Database extensions to allow users and the public to store and access data on Google [25]. This Android software app is used to obtain the level of dustbin fill and represent it with a small animation in a graphical format, after which a notification is sent to the mobile device on which the app is installed [25]. It can send the same data to other mobiles where the app is installed [25].

13

2.4 Summary

From all the information gathered above, it can be concluded that for wireless communication technology, using the Wi-Fi module (ESP8266) is more applicable than using the GSM module (SIM900A) and Bluetooth for this proposed project. As for Bluetooth, this communication technology is not suitable for this project because the range is short, which is only 10 meters.

A Wi-Fi receiver is required to connect the system to the Internet. So, as for the Wi-Fi module, it is a low-cost wireless communication technology that can be a communication medium between NodeMCU and the dashboard online. Besides, Wi-Fi can be used to share the internet, files, and resources between devices because this project is about collecting garbage at a residential college, which has many dustbins in that area. So, when using Wi-Fi, the status level of every dustbin can be known by all the cleaning staff.

The GSM module is a good choice because it works in a wider range, but it requires a more expensive module and also has recurring costs from the cellular carrier and running a publicly accessible server, which costs money. If the product is designed to be worn outside, it had better go down the GSM route, as it works in a wider range.

However, Wi-Fi only covers the inside and immediate vicinity of the building where the access point is installed, but it requires only a cheaper module, and because it is, after all, LAN, there are no recurring costs other than the electricity bill (which you would have to pay anyway if you went the GSM route). Referring back to the project scope, this dustbin was designed for localized and small-scale indoor cases. If it is used indoors only, Wi-Fi is more appropriate. As a result, GSM data is more expensive and slower than Wi-Fi. When Wi-Fi is unavailable, a backup solution like GSM data is a good option.

Based on the information from previous research, we can conclude that an ultrasonic sensor is a suitable sensor for this project. We need to find a suitable sensor to detect the presence of garbage in the dustbin for this project. When it comes to waste monitoring systems, ultrasonic sensors enable more reliable detection. Instead of employing infrared LED light, which may fail when exposed to sunlight, it determines the level of the dustbins by generating ultrasound, and the reflected ultrasonic will return to the ultrasonic sensor.

We need to place the sensor inside the dustbin, which is in a dark place. One of the IR sensor's shortcomings is that they cannot function in low-light conditions. So, we cannot use an IR sensor for this project. As for the weight sensor, we cannot use it because this sensor only detects the weight of the garbage, but this project wants to detect the volume of garbage in the dustbin.

For this IoT garbage monitoring system, I want to make the NodeMCU control a servo motor to collect the garbage via a Blynk smartphone application and show the level of garbage as measured by an ultrasonic sensor. So, I utilized an ESP8266 that is attached as a built-in to a NodeMCU board that has its firmware. A low-level computer control program is called firmware. There is no denying that the ESP32 has more advantages than the ESP8266, which are that it has more digital input/output pins, more analog pins, and Bluetooth.

The reason I choose NodeMCU ESP8266 over ESP32 is that the ESP8266's digital input/output pins are sufficient for this project, which includes an I2C LCD with two pins for SDA and SCL, two ultrasonic sensors with two pins for trigger and two pins for echo, and a servo motor with one pin for PWM. The total number of digital input/output pins used is seven, making the ESP8266 more suitable for use.

There are several limitations to the ESP32. One of them is that it costs more than the ESP8266. Consequently, the ESP8266 may work well at a low cost if we're creating a

straightforward Internet of Things project [38]. Other than that, this IoT garbage monitoring system project does not need to use Bluetooth, so it is wasteful if ESP32 is used for this project. The reason for not using an Arduino microcontroller is that this project requires Wi-Fi. If you use Arduino as the microcontroller, you'll need to connect and configure the ESP8266 module with Arduino, so NodeMCU is a better choice because this microcontroller already has Wi-Fi built in.

As for the project system platform, Blynk was chosen because it is an app available for download from the Google Play store for Android and the Apple App Store for iPhone. Blynk is a platform that allows you to quickly build interfaces for monitoring hardware projects from an iOS or Android device. It is simpler to create and place project dashboards and other widgets on the screen. This platform is perfect for interfacing with simple projects like monitoring the system. The most important thing is that Blynk supports most Arduino boards and the ESP8266 Wi-Fi module.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter aims to discuss the project's approaches employed from the beginning, which is the planning phase until the finish of this project. A few steps are taken to achieve the project's objective. The project's hardware and approach will also be discussed in this chapter.

3.2 Project Workflow

Flowcharts are an efficient way to develop project management strategies because it provides a visual representation of the project. It is vital to have a great flowchart to provide a successful project. The progress of this research was started by understanding the project title. After that, do a research canvas based on previous papers and journals that relate to the project title and discuss it with the supervisor for approval. Once it was approved, the problems, objectives, questions, theoretical or conceptual framework, a method to answer the question part, expected findings, conclusion, and future work based on the project title were determined. The fundamental knowledge of the project's elements was revised. Research the appropriate components for this project. Once the components used for this project have been identified, the analysis procedure will be carried out using the desired parameters and characteristics corresponding to their advantages and disadvantages. Then, plans and discuss with the supervisor the flow system, design, and construction of the project. After getting approval, the scheduled experiments were carried out to the desired result. Finally, a proper conclusion was reached for this project, allowing us to determine future work that can be done to improve it. Figure 3.1 depicts the above-mentioned explanations as a flow chart.

17

```
                        ┌─────────────┐
                        │    Start    │
                        └─────────────┘
                               │
                               ▼
                  ┌──────────────────────────────┐
                  │  Understand the project title │
                  └──────────────────────────────┘
                               │
                               ▼
            ┌────────────────────────────────────────────┐
            │  Do research canvas based on previous paper and │
            │  journals that related to the project title     │
            └────────────────────────────────────────────┘
                               │
                               ▼
            ┌────────────────────────────────────────────┐
            │  Discuss the findings with supervisor        │
            └────────────────────────────────────────────┘
                               │
                               ▼
        No            ◇ Agreed? ◇
                               │ Yes
                               ▼
 ┌──────────────────────────────────────────────────────────────────────────┐
 │ Determine the problems, objectives, questions, theoretical or conceptual   │
 │ framework, a method to answers the question part, expected findings,       │
 │ conclusion and future work based on own project title                      │
 └──────────────────────────────────────────────────────────────────────────┘
                               │
                               ▼
         ┌────────────────────────────────────────────────┐
         │ Study theory of the elements involved in the project │
         └────────────────────────────────────────────────┘
                               │
                               ▼
         ┌────────────────────────────────────────────────┐
         │ Organize the flow, design and construction of the system │
         └────────────────────────────────────────────────┘
                               │
                               ▼
                  ┌──────────────────────────┐
                  │  Discuss with supervisor  │
                  └──────────────────────────┘
                               │
                               ▼
        No            ◇ Agreed? ◇
                               │ Yes
                               ▼
            ┌────────────────────────────────────┐
            │  Conduct the experiments planned    │
            └────────────────────────────────────┘
                               │
                               ▼
            ┌────────────────────────────────────────────┐
            │ Obtain and discuss the results of the experiments │
            └────────────────────────────────────────────┘
                               │
                               ▼
            ┌────────────────────────────────────┐
            │  Proper conclusion of this project  │
            └────────────────────────────────────┘
                               │
                               ▼
            ┌────────────────────────────────────────────┐
            │ Future work that can be improve for this project │
            └────────────────────────────────────────────┘
                               │
                               ▼
                        ┌─────────────┐
                        │     End     │
                        └─────────────┘
```
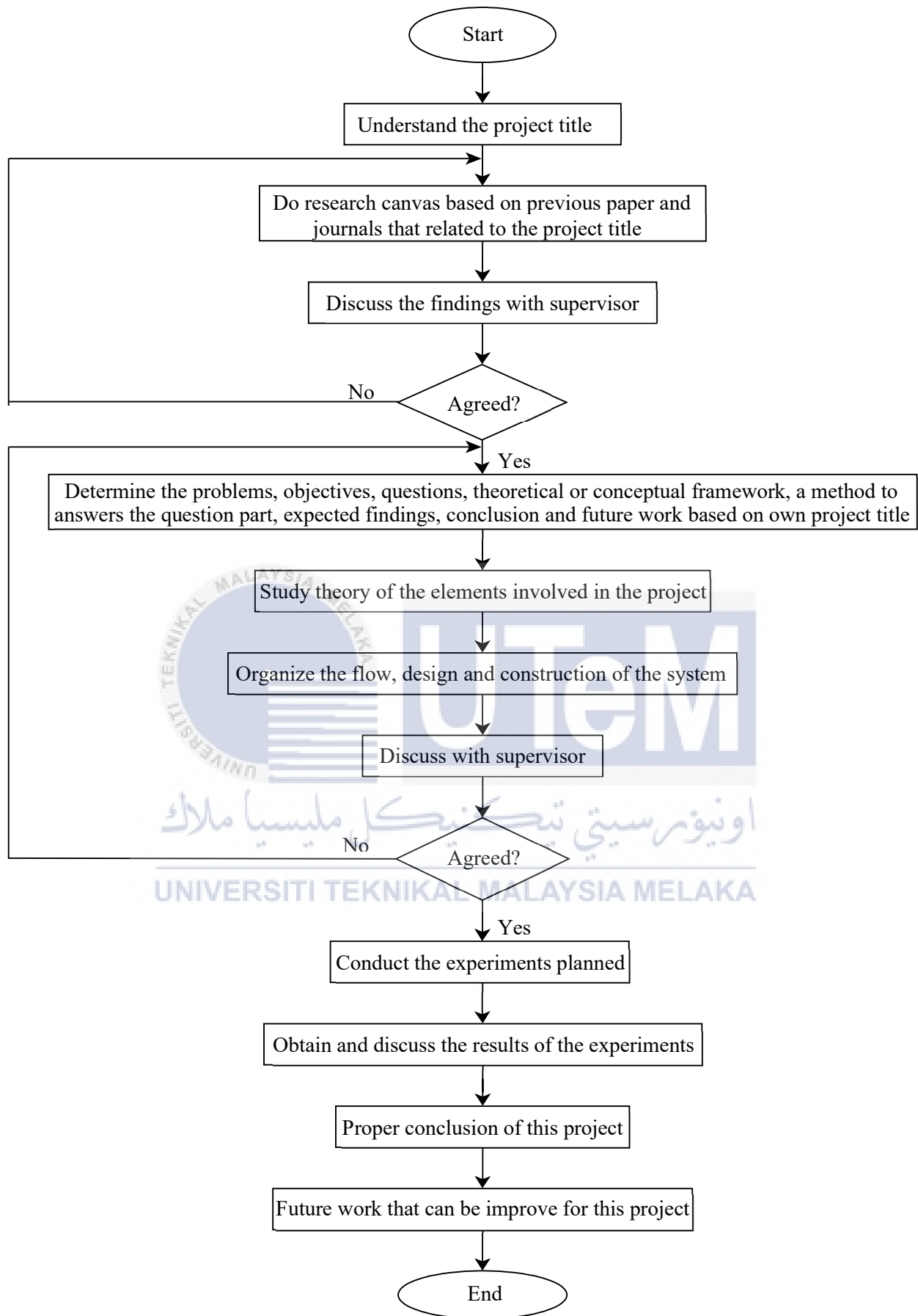
Figure 3.1   General flowchart of the project

18

3.3   Process Flow of System

This project uses two ultrasonic sensors, one servo motor, and one liquid crystal display. Figure 3.2 show the flowchart of how this IoT garbage monitoring system function. First, two ultrasonic sensors will detect and read the distance of the object. There are three conditions with different distances of ultrasonic sensor2. If the distance of ultrasonic sensor2 is more than 20 cm, LCD at dustbin and Blynk application will display "Dustbin Not Full". If the distance of ultrasonic sensor2 is at a range of 10 cm to 20 cm, LCD at dustbin and Blynk application will display "Dustbin Half Full". After displaying the status garbage level in LCD, it will check whether the distance of ultrasonic sensor1 is equal to or less than 30 cm. If yes, the dustbin lid will open automatically, and if no, the dustbin lid does not open. After that, it will go back to detect and read the value from the ultrasonic sensor1 and ultrasonic sensor2.

The last condition of ultrasonic sensor2 is a distance of less than 10 cm. If the distance is less than 10 cm, LCD at the dustbin and Blynk application will display "Dustbin Full", the notification will be sent to the cleaner staff to collect the garbage, and the dustbin lid will remain closed until the cleaner staff collect the garbage. To collect the garbage, cleaner staff need to click a button in the Blynk application and the dustbin lid will open. If the cleaner staff did not collect the garbage, the notification will always be sent to the cleaner staff's smartphone until they collect the garbage. After the cleaner collects the garbage, the system will go back to measure the distance of an object. The Blynk application will always display the percentage level of garbage to be monitored and the garbage distance value that is detected.
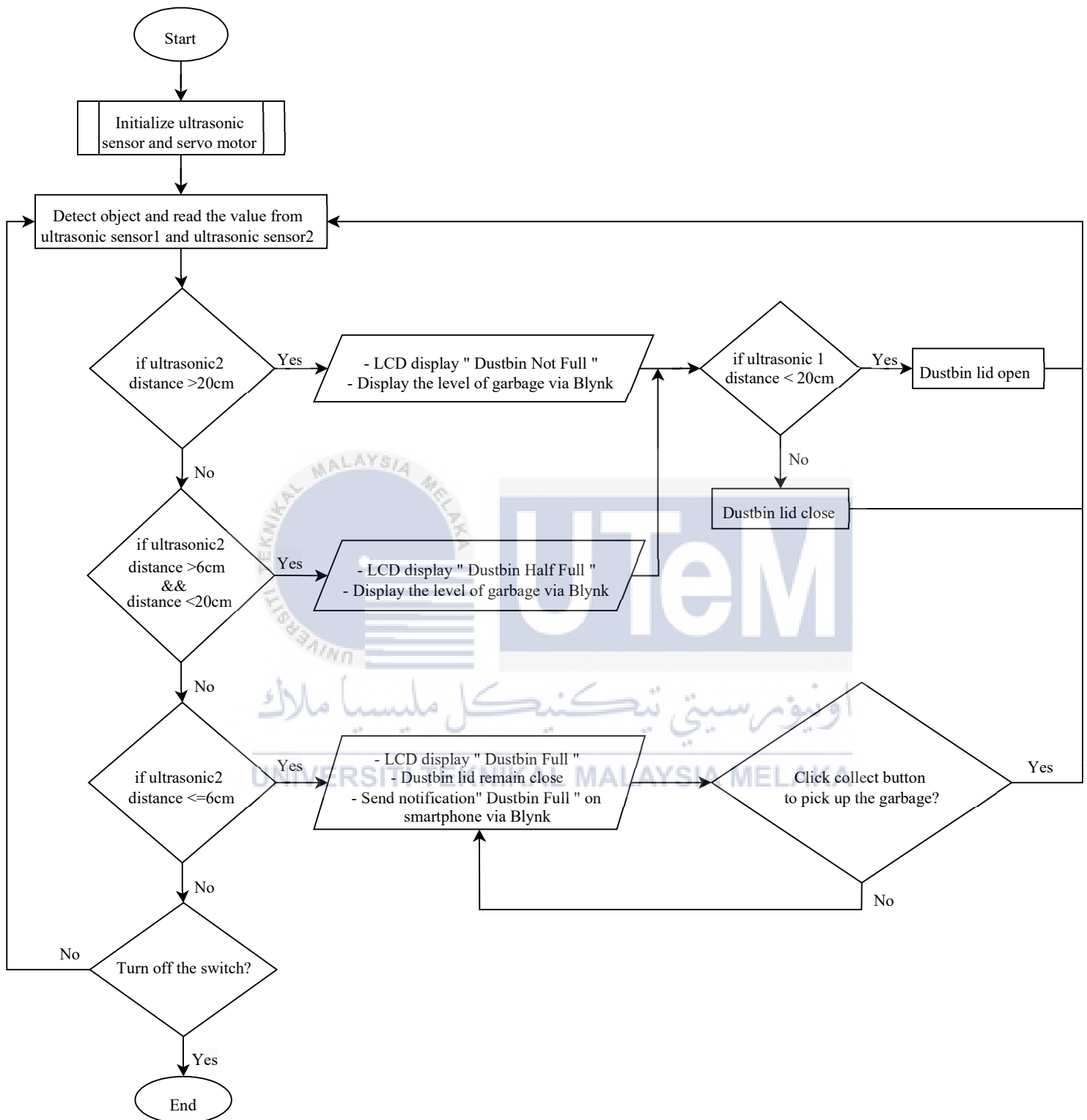
Figure 3.2   IoT garbage monitoring system flowchart
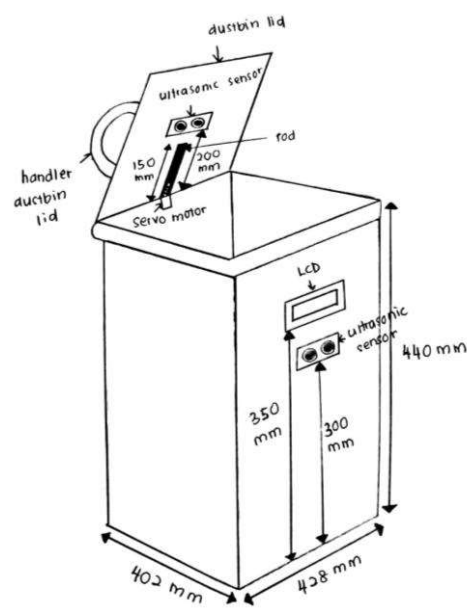
3.4 Project Design

Figure 3.3   Illustration of an isometric view of a garbage monitoring system

Figure 3.3 above is the design for the development of an IoT-based garbage monitoring system using a microcontroller. The size of the dustbin used as a prototype for this project is 21 cm x 27 cm x 37 cm (20 liters), which is the smallest size of a dustbin. For this project, this size has been chosen as a prototype because it is the cheapest, but this garbage monitoring system can be developed for the largest dustbin size, which is 59 cm x 76 cm x 100 cm (240 liters) because an ultrasonic sensor can measure distance up to 16 meters to determine the level of garbage in the dustbin. The MG996R servo motor was chosen because its maximum stall torque is 11kg/cm according to the datasheet. When it detects people, it can lift the dustbin lid to open and close it automatically. To determine the precise level of garbage in the bin, this project employs two ultrasonic sensors, one under the dustbin lid and one in the center. Another ultrasonic sensor is placed at the front of the dustbin to detect the presence of objects. So, to avoid the lid opening automatically when the ultrasonic sensor detects the presence of a cat, this sensor is placed 30 cm from the dustbin base because the height of a cat is about 23 cm to 25 cm. This prototype can be placed outdoors because it uses an ultrasonic sensor that can be exposed to direct sunlight.

21

3.5   Block Diagram

The block diagram in Figure 3.4 depicts the whole system, which includes the dustbin equipped with a sensor unit. The ultrasonic sensor and NodeMCU with Wi-Fi module ESP8266 are the first components of the proposed system.

The first ultrasonic sensor placed at the front of the dustbin was used to detect the presence of people. After NodeMCU receives input data from the ultrasonic sensor, it will control the servo motor to make the dustbin open and close automatically. When the person's distance from the dustbin is equal to or less than 20 cm, NodeMCU sends the data to the servo motor, which rotates 90 degrees and opens the dustbin lid. Then, after the ultrasonic sensor detects a person's distance is greater than 20 cm, the servo motor will rotate back to its initial condition of 0 degrees, which is the dustbin lid closed.

The other ultrasonic sensor is placed on top of the dustbin facing the bottom to sense the distance between the dustbin lid and the top of the garbage within the dustbin. Set a threshold value based on the size of the dustbin. When an ultrasonic wave contacts any object, the ultrasonic transmitter emits an ultrasonic wave that passes through the atmosphere and is reflected toward the sensor. The reflected wave is detected by the ultrasonic receiver module. The sensor's echoes detect waves and calculate object distance.

The LCD placed in front of the dustbin will display the status of the garbage in the dustbin as either not full, half full, or full. The ESP8266 Wi-Fi module that is attached to NodeMCU is used to connect the functions that are at the dustbin with the Blynk application to upgrade it to the "Internet of Things," which can be monitored and controlled by using a smartphone alone.
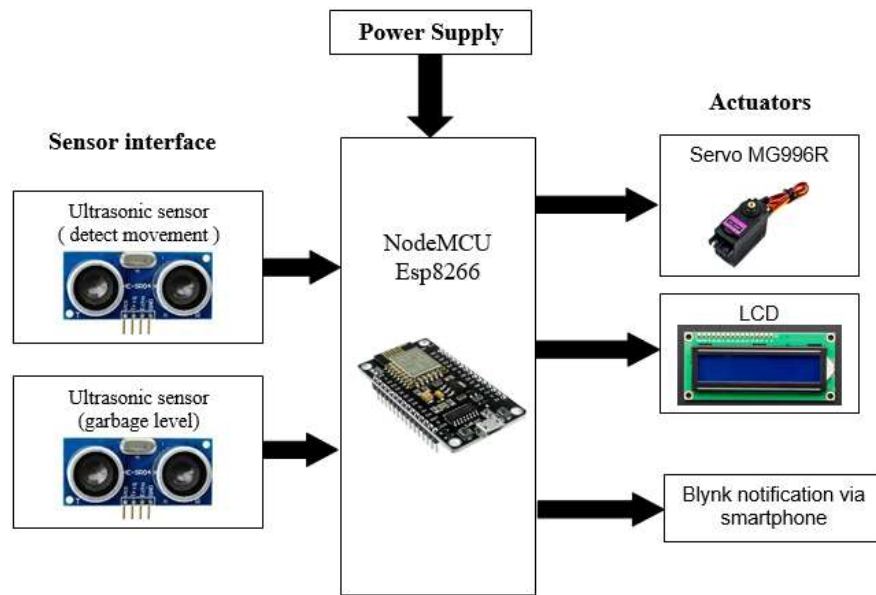
Figure 3.4   Block diagram of the proposed system

3.6   Equipment

3.6.1   Hardware Development

This section covers microcontrollers, sensors, and actuators for a garbage monitoring system. The function and application of each of the components are discussed in this chapter.

3.6.1.1   NodeMCU ESP8266

NodeMCU in figure 3.5 is an open-source IoT platform. The Wi-Fi module is used for controlling the microcontroller toward the actuator through a mobile application. This module is the most cost-effective solution for all types of wireless communication currently available on the market. It consumes very little power to operate, which is 3.3V or 5V. The Wi-Fi module has safety features due to authentication and encryption. NodeMCU provides a way to connect different sensors to their controllers wirelessly via Wi-Fi. Since it is an improved version of the ESP8266, it has better and easier programming, with better voltage stability and more reliability.
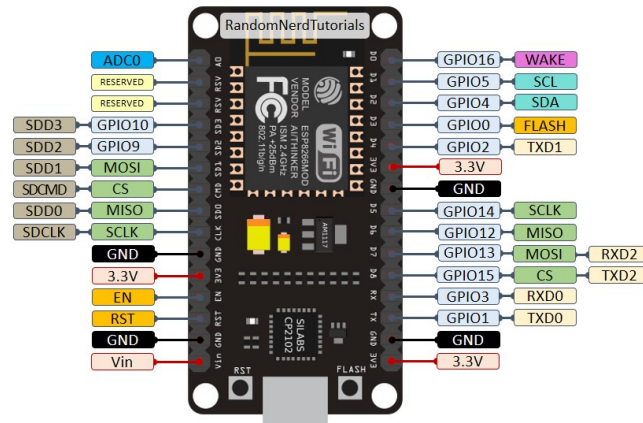
Figure 3.5   NodeMCU pinout

3.6.1.2   Ultrasonic Sensor

In this project, HC-SR04 is an ultrasonic sensor that is used for measuring the level of garbage in the dustbin by measuring the distance between the top of the lid to the top of the garbage and detecting the presence of people. Table 3.3 below is the HC-SR04 Sensor features:

Table 3.1   HC-SR04 Ultrasonic Sensor Features

Operating voltage	+5V
Theoretical measuring distance	2cm to 450cm
Accuracy	3mm
Measuring angle covered	<15°
Operating current	<15mA
Operating frequency	40Hz

The HC-SR04 ultrasonic sensor is a widely common sensor that is utilized in many applications that require measuring distance or sensing things. The module includes two projections in the front that look like eyeballs and form the ultrasonic transmitter and receiver. The sensor operates using a basic high school algorithm.

24

$$\text{Distance} = \text{Speed} \times \text{Time}$$

The ultrasonic transmitter sends out an ultrasonic wave, which travels through the air and is reflected back toward the sensor when it encounters any obstacle. This reflected wave is detected by the ultrasonic receiver module. The trig pin is attached to the NodeMCU output pin, and it is used to burst the microwave from the sensor to the target. By referring to figure 3.6, echo is linked to the microcontroller's input pin and is used to receive data reflected from the opposite side. VCC is connected to a 5v power supply, while GND is connected to the ground.
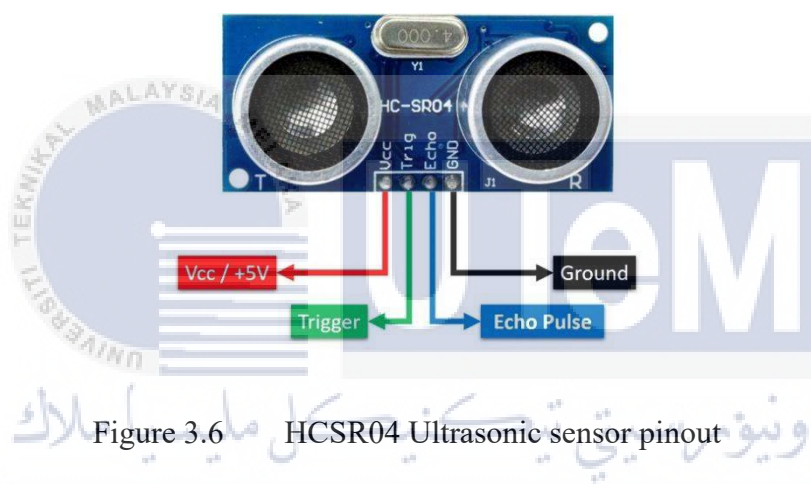

Figure 3.6   HCSR04 Ultrasonic sensor pinout

3.6.1.3   Liquid Crystal Display

The 16x2 I2C liquid crystal display (LCD) in figure 3.7 below is a normal screen incorporated with the I2C module for displaying text and numbers. The usual LCD usually requires a lot more line coding and complicated wire soldering and connection. It will be a problem if do not have many I/O pins available, like NodeMCU. Therefore, the I2C LCD uses 2 pins only, such as SDA and SCL. The serial data pin is SDA, while the clock pin is SCL. The rest pins are for the power supply, VCC, and ground. The pin connection uses a normal jumper wire and can be connected directly. Hence, this is its advantage because it doesn't complicate

the design. The potentiometer is just to adjust the LED backlight. This LCD can display all the current situations of the I/O device.

Figure 3.7   I2C LCD pinout

3.6.1.4   Servo Motor

The MG996R is a metal-gear servo motor with a maximum stall torque of 11 kg/cm. Like other servo motors, it rotates from 0 to 180 degrees based on the duty cycle of the PWM wave supplied to its signal. If the application requires stronger and longer-running motors, metal gears are a better choice. To make this motor rotate, we have to power the motor with +5V using the red and brown wires and send PWM signals to the orange color wire. By delivering a coded signal to the servo, this shaft may be precisely positioned at certain angular positions. The angular location of the shaft varies when the coded signal changes. In this project, the servo motor is used to open and close the dustbin lid automatically after the ultrasonic sensor detects the presence of an object.

Figure 3.8   Servo Motor pinout

26

3.6.1.5 Measuring Tape

The measuring tape with 0.1 cm of sensitivity (figure 3.9) is used to measure the distance between the ultrasonic sensor and the object. This measuring tape is also used to do the calibration for an ultrasonic sensor to compare the measurements between manual measurements, readings from the serial monitor in the Arduino IDE, and the Blynk application.

Figure 3.9   Measuring tape

3.6.2 Software Development

3.6.2.1 Arduino IDE

The open-source Arduino Software (IDE) is a cross-platform application (for Windows, macOS, and Linux) that is written in functions from C and C++ that makes it easy to write code and upload it to the Arduino or NodeMCU microcontroller board. It allows the user to program a microcontroller by simply selecting the relevant port and microcontroller model in the program settings, and the coding is retrieved into the microcontroller. This application has a basic user interface (Figure 3.10) and is easy to create; thousands of libraries can be readily loaded and utilized. This software is compatible with all Arduino boards. A text editor for writing code, a message box, a text terminal, a toolbar with buttons for basic tasks, and a series of menus are all included in the Arduino development environment. It interacts with the Arduino or NodeMCU hardware and uploads programs to it.
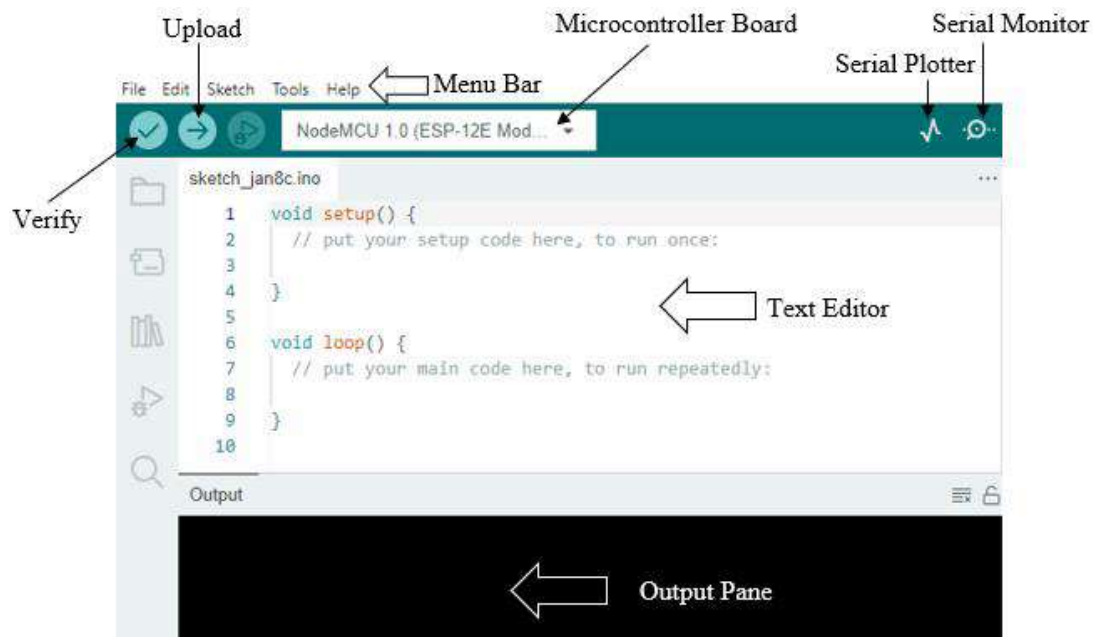
27

Figure 3.10   Arduino IDE interface

3.6.2.2   Blynk

Blynk is a platform that includes iOS and Android apps for controlling Arduino, Raspberry Pi, and other similar devices via the Internet. This app can be downloaded at the Google store for Android and Apple store for iPhone. It's a digital dashboard where you may drag and drop widgets to create a graphical interface for your project. The platform is made up of three primary parts:

- Blynk App - allows you to develop stunning interfaces for your projects by utilizing the numerous widgets that we offer. Refer to figure 3.11.

- Blynk Server - in charge of all communications between the phone and the hardware. It is open-source, capable of supporting thousands of devices, and can even be run on a Raspberry Pi. Refer to figure 3.12.

- Blynk Libraries - enable contact with the server and process all incoming and outgoing commands on all major hardware systems.

28

Notification is the function utilized in this project. When the garbage is full, the Blynk alerts the cleaner staff. This project will send an alarm every 5 minutes starting when the garbage is full. The notified signal will stop delivering the message to the cleaner staff when the garbage in the dustbin is empty.
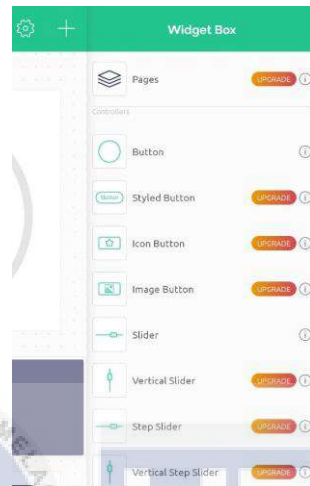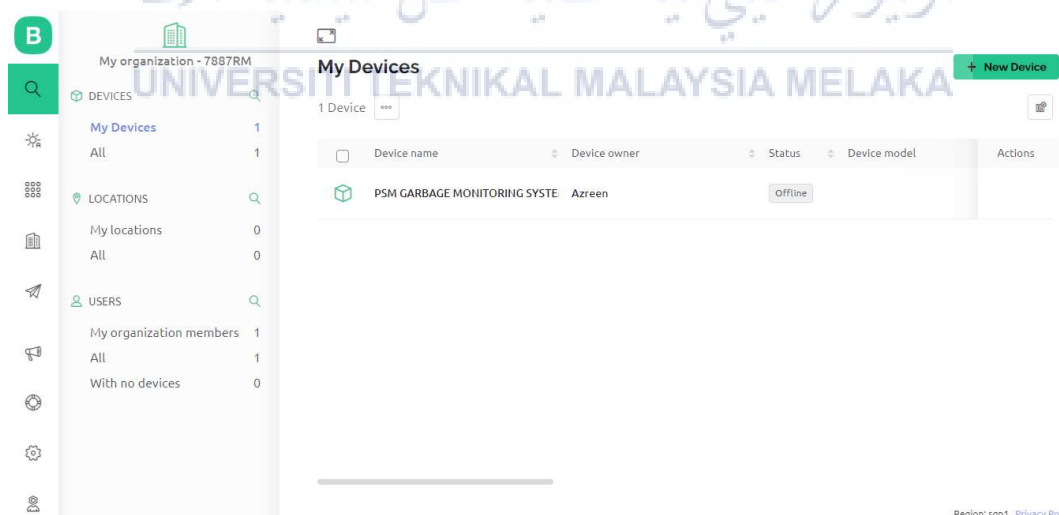

Figure 3.11   Blynk App interface

Figure 3.12   Blynk server

3.7   Simulation Circuit of Project

This is the preliminary circuit for the IoT garbage monitoring system. This simulation circuit have been constructed to test the function of the project before applying in hardware design.
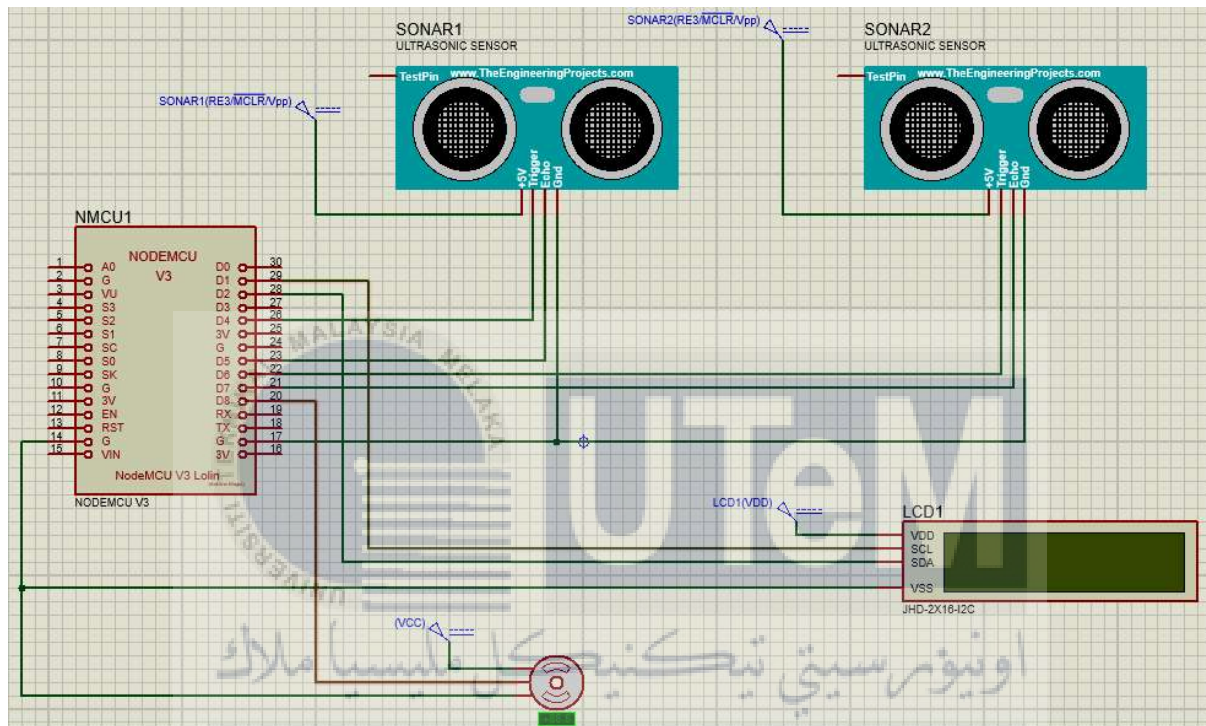

Figure 3.13   Simulation design of the project

3.8   Experiment Setup

This section is for planning project testing to collect the necessary data for this project. The hardware part has to be tested physically. It is necessary to check whether the system is working properly or not and whether the readings are accurate. This testing will be conducted using a measuring tape to measure the distance pointed out by the ultrasonic sensor.

3.8.1   Ultrasonic Sensor Calibration Testing

Objective: To make sure to get the same distance value pointed out by the ultrasonic sensor.

Procedures:

1.   Place the ultrasonic sensor on the breadboard.
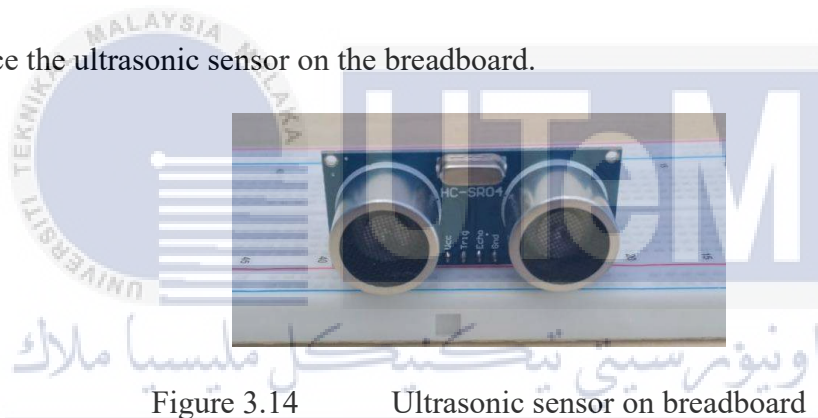
Figure 3.14   Ultrasonic sensor on breadboard

2.   Place the object 5 cm away from the ultrasonic sensor.

3.   Using a measuring tape to measure the distance.

Figure 3.15   Measuring tape

4.   Record the reading get from the serial monitor at Arduino IDE, LCD, and Blynk application.

31

Figure 3.16   Sensor calibration setup
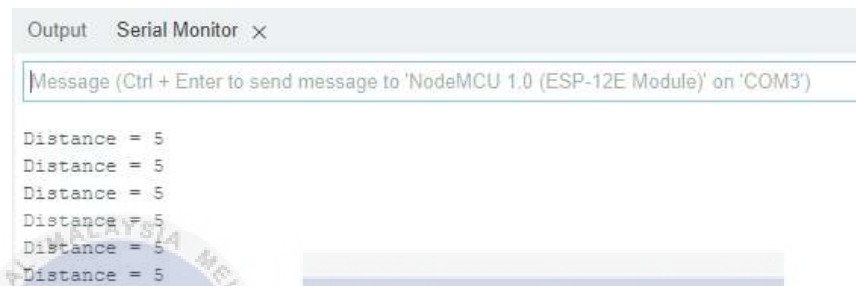
Figure 3.17   Serial monitor

Figure 3.18   LCD

Figure 3.19   Blynk application

5.   All the steps above were repeated by increasing the distance between the ultrasonic

       sensor and the object to 10 cm, followed by 15 cm, until 50 cm.

32

3.8.2   Experiment Detects Object Presence Near the Dustbin

Objective: To ensure the dustbin lid opens automatically according to a predetermined distance.

Procedures:

1. Place the ultrasonic sensor in front of the dustbin.

Figure 3.20   Ultrasonic sensor in front of the dustbin

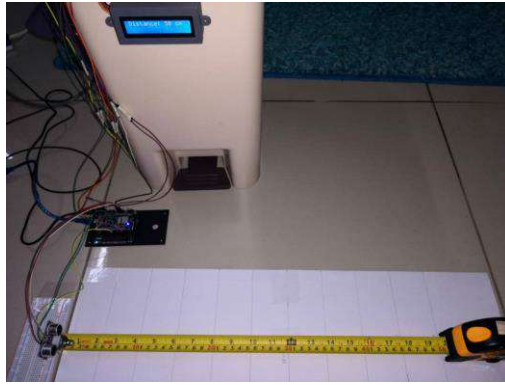2. Set the threshold value of 20 cm in the coding.

```
24    const int THRESHOLD = 20;
```

Figure 3.21   Threshold value

3. Place the object 5 cm away from the ultrasonic sensor.

Figure 3.22   Measure the distance

4. Observe the condition of the dustbin lid (either open or closed).

5. All the steps above were repeated by increasing the distance between the ultrasonic sensor and the object to 10 cm followed by 15 cm until 30 cm.

33

3.8.3 Experiment Dustbin Status Identification

Objectives: To determine the distance accuracy of garbage level in the dustbin.

Procedures:

1. Place the ultrasonic sensor at the top of the dustbin lid.

Figure 3.23   Ultrasonic sensor position

2. Fill in the dustbin with garbage.

3. Measure the distance between the ultrasonic sensor with the top of the garbage.

Figure 3.24   Measure the level of garbage

4. Record the actual measurement and measurement from the Blynk application.

   Using this formula to calculate actual measurement:

   Distance = height of dustbin – height of garbage

5. Observe the dustbin status display at the actual LCD and Blynk LCD. At the same time, monitor the percentage level of dustbins at Blynk.

3.8.4 Experiment Dustbin Lid Action

Objectives: To observe the dustbin lid either open or closed according to the status of the dustbin.

Procedures:

1. Stand away from the dustbin less than 20cm.

2. Fill in the dustbin with garbage.

3. Stand away from the dustbin for more than 20cm.

Figure 3.25   Way to measure

4. Observe the status of the dustbin.

Figure 3.26   Display the status of the dustbin

5. Repeat step 1, and observe the action of the dustbin lid either opening or remaining closed.

6. Record the action of the dustbin lid.

7. Then repeat steps 1 until 6.

3.8.5   Testing Different Types of Material

Objectives: The purpose of the experiment is to test whether several materials can be detected by ultrasonic sensors.

Procedures:

1. Place the vegetable into the dustbin.
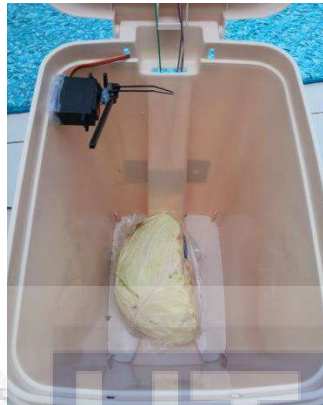
Figure 3.27   Vegetable in dustbin

2. Observe and record whether the ultrasonic sensor can detect it.

3. Repeat all the steps above and change with the several types of material.

3.8.6 Experiment Send Notification to Smartphone When Dustbin is Full.

Objectives: To observe the time taken to send a notification sent to a smartphone when the dustbin is full.

Procedures:

1. Fill in the dustbin with garbage until full.

Figure 3.28   Full-filled dustbin

2. Observe the time taken to get a notification when the dustbin starts full by using a stopwatch.

3. Observe how many times Blynk apps will send notifications in one hour.

3.9   Summary

In short, if all the phases have been finished and all of the aforementioned conditions have been satisfied, including software and hardware, this project may be realized. Any issues that develop throughout the project's execution will be addressed to accomplish the goals. As detailed in the next chapter, data will be gathered and examined.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This chapter aims to discuss the analysis of the results of the performance of the IoT garbage monitoring system. Therefore, ultrasonic, servo motor and get notifications from Blynk apps will be tested.

4.2 Results and Analysis

4.2.1 Calibration Sensor

4.2.1.1 Ultrasonic reading accuracy
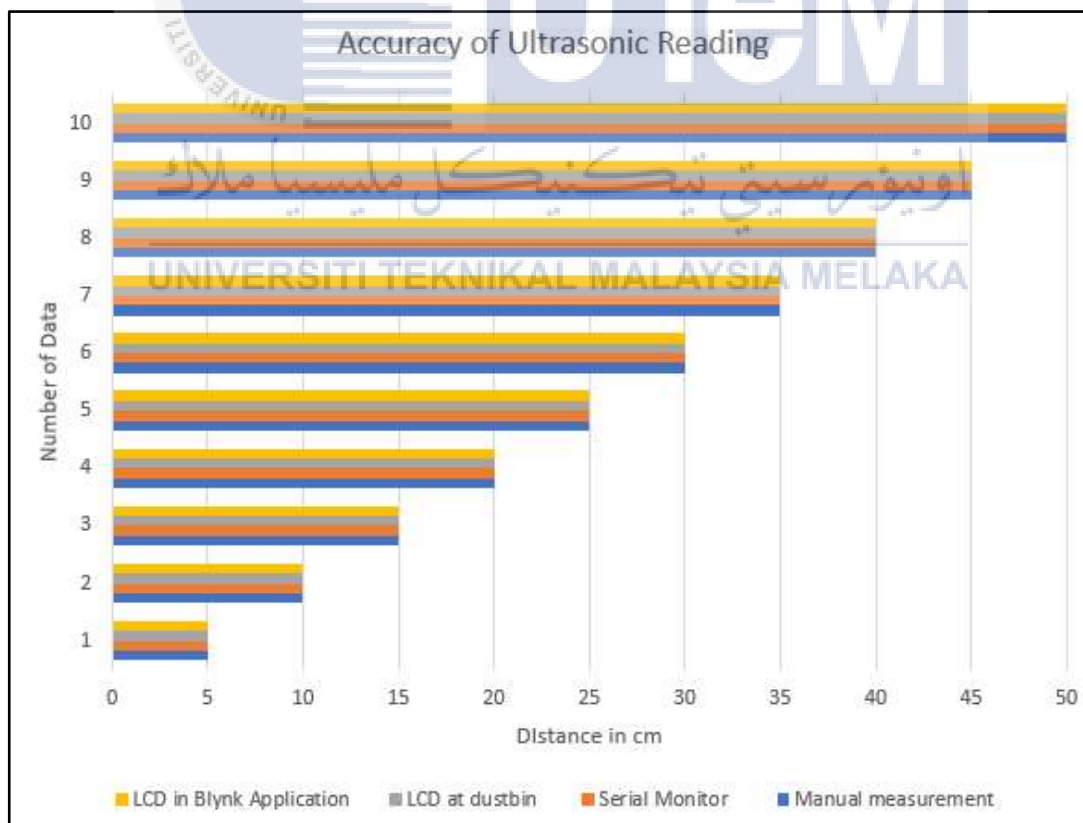
Figure 4.2   Accuracy of ultrasonic reading

The graph in Figure 4.1 depicts the accuracy of ultrasonic sensor reading that get from manual measurement, serial monitor, LCD at dustbin, and LCD in Blynk application. The graph shows that the ultrasonic sensor reading display at a serial monitor, LCD dustbin, Blynk application, and manual measurement is perfectly accurate.

4.2.1.2   Time Response of ultrasonic sensor
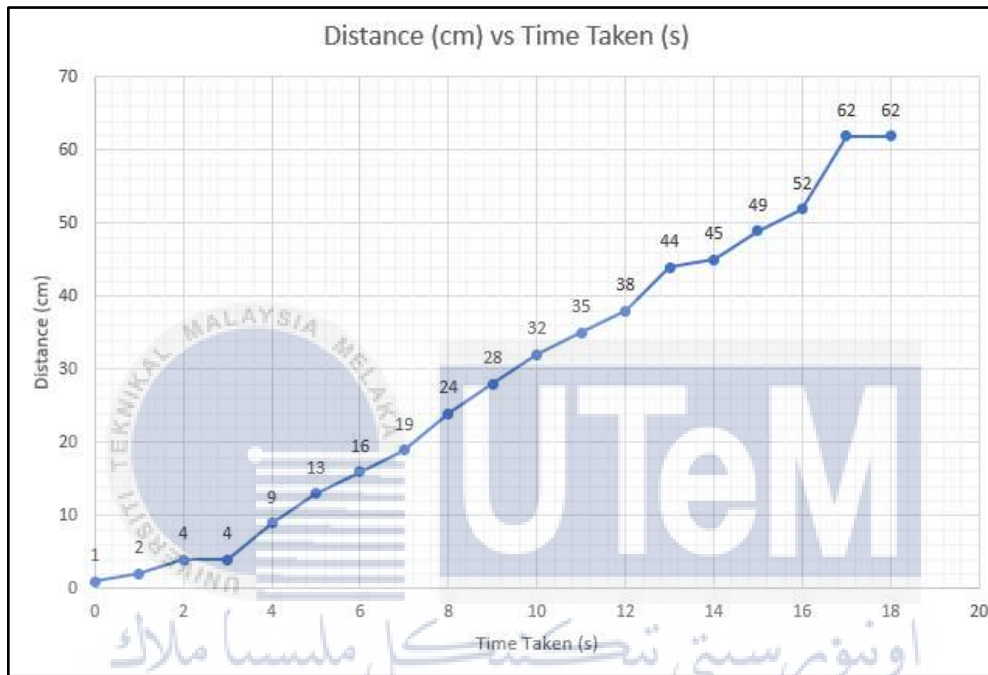
Figure 4.2   Time response of the ultrasonic sensor

Figure 4.2 shows the graph of the time response of an ultrasonic sensor. This graph was created with the Arduino IDE's serial plotter. So, based on the graph above, it can be concluded that the ultrasonic transmitter takes 1 second to transmit an ultrasonic wave, which is reflected back toward the sensor when it encounters any obstacle, and that this time remains constant at 1 second regardless of distance from the substance. An ultrasonic sensor's formula is distance=speed x time. The hypothesis that can be observed from this graph is that when the distance of a substance from the ultrasonic increases, the speed of the sound wave increases.

4.2.2 Detect Object Presence Near the Dustbin.

Figure 4.3.1

Object distance 5cm

Figure 4.3.2

Object distance 35cm

Figure 4.3.3

Threshold distance 20cm

Table 4.1   Distance of object and dustbin lid action

No.	Distance	Dustbin Lid Action
1.	10 cm	Open
2.	20 cm	Open
3.	30 cm	Close
4.	40 cm	Close
5.	50 cm	Close

Table 4.1 shows the result of the distance affecting the action of the dustbin lid. The threshold value set in the coding is 20cm because this is the proper distance for people to throw the garbage. If people want to throw the garbage at a distance of more than 20cm, of course, the garbage will not go into the dustbin. So, Table 4.1 shows that the dustbin lid will open automatically when the distance between people and the dustbin is equal and less than 20cm. The dustbin lid does not open if the distance is more than 20cm to avoid the dustbin lid opening automatically when people walk through the dustbin.

40

4.2.3 Dustbin Status Identification

Table 4.2   Dustbin status identification

Test case	Actual measurement	Input experimental visual	Status of dustbin display	Blynk display
Not full	26 cm		Dustbin Not Full	23% Distance: 27cm Dustbin Not Full
Full	14 cm		Dustbin Half Full	57% Distance: 15cm Dustbin HalfFull
Half full	2 cm		Dustbin Full	94% Distance: 2cm Dustbin Full

41

The IoT garbage monitoring system was experimentally analyzed for performance. Experimental results of the IoT garbage monitoring system identifying the dustbin status in Table 4.2. As an initial condition, the dustbin shows a status of "Dustbin Not Full". This status can be accessed by the user through the smartphone via the Blynk application and also in the web dashboard Blynk cloud. Garbage filling starts as soon as the dustbin is placed in an accessible location, and the status is updated dependent on how full it is. The status updates of the dustbin under various garbage-filling circumstances are shown in Table 4.2.

This IoT garbage monitoring system was tested by introducing approximately 50 percent of the garbage fill into the dustbin, then the garbage monitoring system was tested for status identification. Here, the result from the garbage monitoring system was "Half Full" indicating a visual display on the LCD screen and in the Blynk application. In this case, when the IoT garbage monitoring system was tested by filling the garbage completely into the dustbin, the developed system performs effectively by indicating the monitoring "Full" status.

There is a slight difference in distance readings obtained from the actual measurement and readings display in the Blynk application. The top of the garbage surface does not flat and this causes some difficulty in getting an accurate measurement reading. After conducting the experimental studies on the developed IoT garbage monitoring system, it was identified that all the defined test cases were validated.
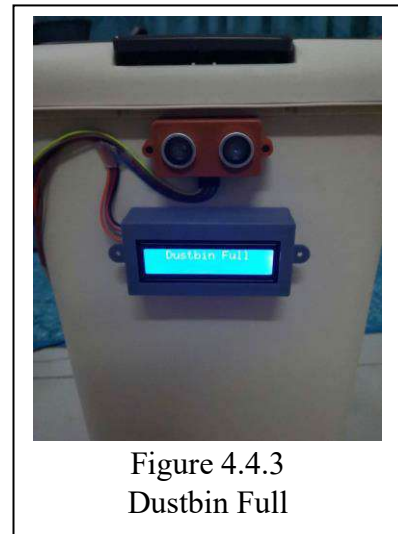
42

4.2.4 Dustbin Lid Condition

| Figure 4.4.1 | Figure 4.4.2 | Figure 4.4.3 |
| Dustbin Not Full | Dustbin Half full | Dustbin Full |
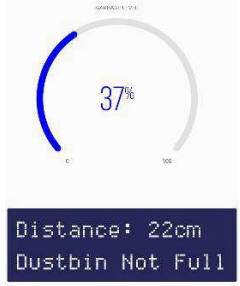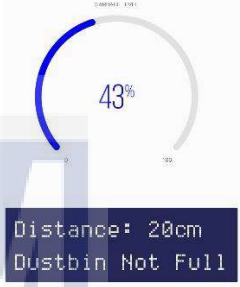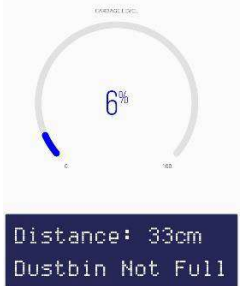
Table 4.3   Dustbin lid action

No.	Status of dustbin	Dustbin Lid Action
1.	Dustbin Not Full	Open
2.	Dustbin Half Full	Open
3.	Dustbin Full	Close

Referring to Figures 4.4.3 and 4.4.2, when there are obstacles with a distance of less than or equal to 20cm, the dustbin lid opens automatically because the dustbin has not reached the full level yet. As for Figure 4.4.3 which is the dustbin has reached the full level, the dustbin lid has remained close though there are obstacles with a distance of less than or equal to 20cm. This system can prevent the occurrence of garbage overflow. The results of this experiment indicate that the function of keeping the dustbin lid closed when the status of the dustbin reaches the full level is successful.

4.2.5 Different Types of Material.

Table 4.4   Types of material detected

Types of material	Result	Actual measurement	Blynk display
Vegetable	Detect	20 cm	37% Distance: 22cm Dustbin Not Full
Paper box	Detect	19 cm	43% Distance: 20cm Dustbin Not Full
Plastic bag	Detect	16 cm	46% Distance: 19cm Dustbin HalfFull
Glass	Detect	30 cm	6% Distance: 33cm Dustbin Not Full

44

Figure 4.5.1 Glass	Figure 4.5.2 Paper box	Figure 4.5.3 Plastic bag	Figure 4.5.4 Vegetable
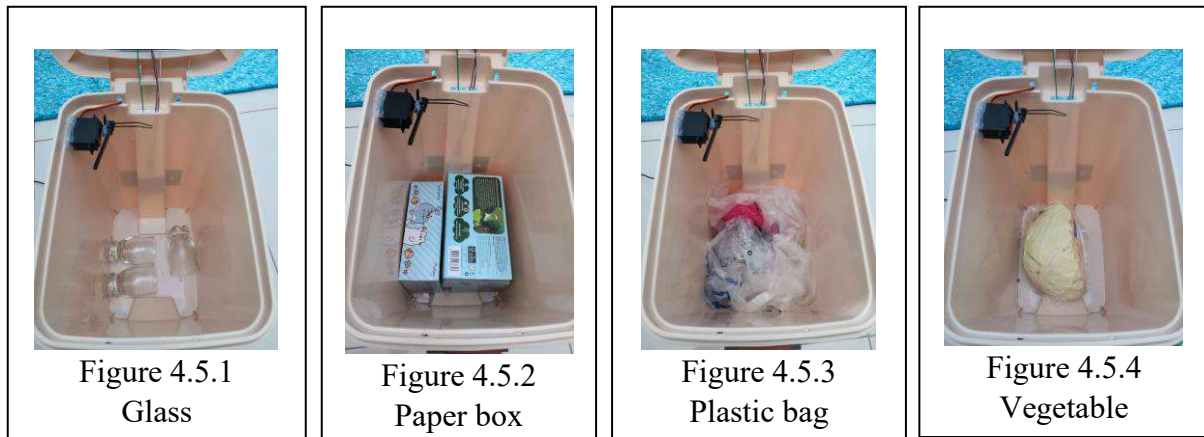
Table 4.4 shows that all the testing materials can be measured by ultrasonic sensor but exist reading errors with less than 2 cm to 3 cm between actual measurement and display at Blynk application.

4.2.6   Send Notification to Smartphone When Dustbin is Full.
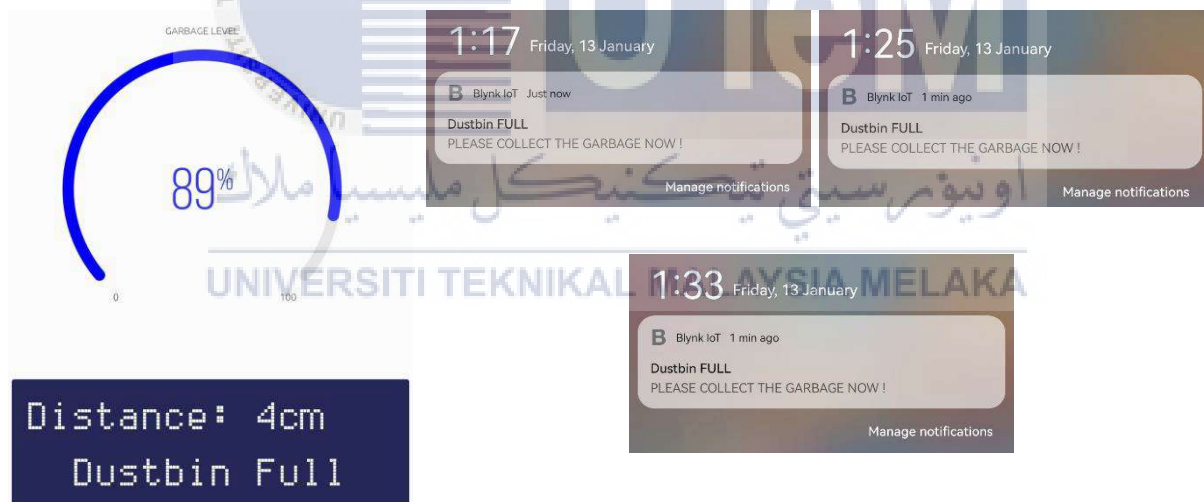
Figure 4.6   Blynk monitoring   Figure 4.7   Notification send via Blynk

When the level of garbage in the dustbin is full which is a distance less than or equal to 6 cm, the Blynk application will send a notification to the user. The notification will be sent every 7 minutes if the user did not collect the garbage yet.

4.3   Summary

It is concluded that this project was successful in designing and developing an IoT garbage monitoring system using a dustbin prototype. The dustbin prototype has an automatic opening and closes the lid. The system has sensors that measure the level of garbage and display them on the user's smartphone for monitoring purposes. Thus, the user does not need to manually check the level of garbage in the dustbin. All of the important experiment data is analyzed in real-time as it is collected from the sensors and sent through the internet via the Wi-Fi module. This system's design has been registered as a development of IoT based garbage monitoring system using a microcontroller.

CHAPTER 5

CONCLUSION

5.1   Introduction

This chapter discusses the conclusion and future work for the IoT garbage monitoring system.

5.2   Conclusion

In conclusion, the IoT garbage monitoring system is indeed a viable and reliable system that can be used to overcome the garbage overflow problem. The main objective of this project is to maintain the level of cleanliness in a residential college and create an environment that is better for living. As stated in the introduction, the first objective of this research is to design a prototype dustbin with an IoT-based garbage monitoring system that can monitor the level of garbage in the dustbin. This objective was successfully achieved. In this project, implementation is done only for a dustbin. The hardware prototype is constructed with a NodeMCU ESP8266 module and an ultrasonic sensor to monitor the actuators of the servo motor and LCD. For the software part, the language used in this project is C++ programming. Then the Blynk cloud application was used to construct the function set in the Blynk mobile app. Mobile applications enabled users to preview the garbage level and receive notifications when the garbage was full.

As for the second and third objectives of this research, which are to open the dustbin lid automatically and develop a smart alert system for garbage clearance by giving notification via the Blynk app to a smartphone, three experiments were carried out. The first experiment

47

was to detect the presence of humans and make the dustbin lid open automatically. From this experiment, it can be concluded that this dustbin only opens the lid when it detects an object at a distance of less than 20 cm. The second experiment was the testing of a notification sent to the user's mobile phone via the Blynk app when the dustbin is full, and at the same time, the dustbin lid will remain closed until a cleaner collects the garbage. The observation result shows that starting from 6 cm or less is the most suitable distance to send a notification because it has extra space so the dustbin lid can close properly.

Lastly, the outcome of the third experiment, which tested different types of garbage material, proved that ultrasonic can detect and measure all types of garbage material, such as vegetables, paper boxes, plastic, and glass, but that there is a reading error of less than 2 to 3 cm in the actual value. Besides that, testing for a suitable distance sends a notification to the collection center. Overall, the IoT garbage monitoring system performed admirably, but its performance fell short of expectations. So, if there are any possibilities to improve the performance of this system, then it would be highly recommended.

5.3 Future works

For future works, the IoT garbage monitoring system can be enhanced as follow:

i. An ultrasonic sensor was used to detect the level of the garbage. Although the results are considered promising most of the time, sometimes it will produce data that is not accurate. This can be improved by researching a more suitable sensor or improving the design of the system to minimize the data inaccuracy of the system.

ii. This system can be implemented by smartly controlling garbage overflow by separating it into dry and wet parts.

iii. Implement a global positioning system (GPS) to this IoT garbage monitoring system to locate the dustbin location.

iv. Improvement by installing a compressor inside the dustbin to compress the garbage. So that it can hold more garbage before reaching the level of "Dustbin Full".
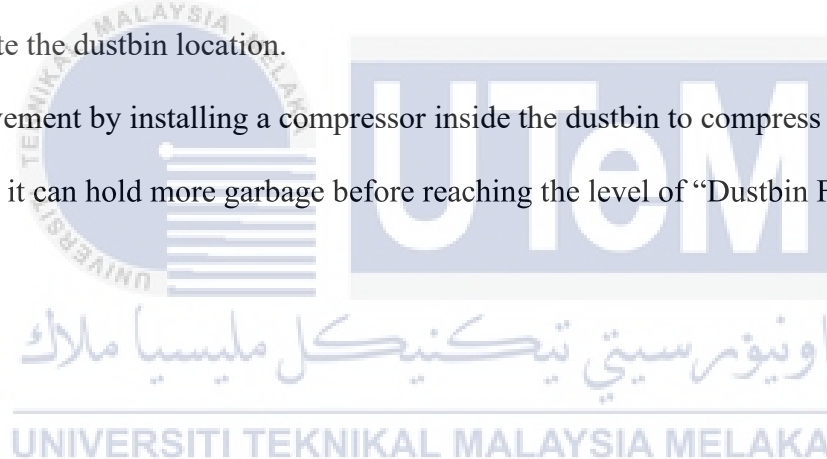
REFERENCES

[1]   Viral Rambhia, Aman Valera, Rahul Punjabi, "Smart Dustbins - Automatic Segregation & Efficient Solid Waste Management using IoT Solutions for Smart Cities", International Journal of Engineering Research & Technology (IJERT), 2019.

[2]   Mayuri Dongre, Shruti Walde, Rupali Babde, Charul Vaidya, Varsha Lanjekar, "IOT Based Garbage Monitoring System", International Journal of Innovations in Engineering and Science, 2019.

[3]   S.Sivajothi Kavitha,K.Hemalatha V.Jamuna, "A Real-Time Smart Dumpsters Monitoring and Garbage Collection System Using Iot", International Journal of Recent Technology and Engineering (IJRTE), 2019.

[4]   B. Sridhar, G. Manasa, A. Mohini, B. Naveen, "Development of IoT Based Smart Dustbin Monitoring System", International Journal of Engineering and Advanced Technology (IJEAT), 2020.

[5]   Anitha A, "Garbage monitoring system using IoT", ICSET, 2017.

[6]   Kuhu Vaish, Shivani Kashyap, Shivani Nagar, Swati Goel, "A Survey on Waste Management Monitoring System Based on IoT", International Journal of Innovative Research in Technology & Science, 2020.

[7]   Deepti Patole, Darshan Panchal, Krishna Sampat, Saurabh Nagare, "IoT based Garbage Monitoring System", International Journal of Computer Applications, 2018.

[8]   Shabanaafreen Hussain, Samiha Shaikh, Ashish Jadhav, Prof. Jadhav V.J, "Smart City-Garbage Monitoring System Using IoT", IJARIIE-ISSN, 2018.

[9]   Amit Kumar Sinha1, Somesh Kumar Saxena, Nitin Kumar, Abhishek Kumar Maurya, "Arduino Based Garbage Monitoring System using IoT", International Research Journal of Engineering and Technology (IRJET), 2019.

[10]   Prof. P. Arunadevi, S. Karthika, V.V. Swetha raj, K. Aamir Suhail, "Smart Garbage Monitoring System Using Internet of Things", International Research Journal of Engineering and Technology (IRJET), 2019.

[11]   Prof. Devika Deshmukh, Ashwini Karme, Bhavana Nagmote, Megha Joshi, Nisha Dekate, Prachi Mishra, "IoT Based Garbage Monitoring Using Location Tracking System", IJARIIE, 2018.

[12]   Ms. Komal V. Yawale, Prof. G.D.Dalvi, "Automated Garbage Monitoring System Using GPS,GSM and IoT", International Journal Of Creative Research Thoughts (IJCRT), 2021.

[13]   N. A. Ali, A.R. Syafeeza, A. S. Ja'afar, Norihan Abdul Hamid, M. Ridzuan, "Garbage monitoring systems based on Internet-of-Things application", ICTEC, 2019.

[14]   Thangavel Bhuvaneswar, J. Hossen, NurAsyiqinbt. Amir Hamzah, P. Velrajkumar, Oo Hong Jack, "Internet of things (IoT) based Smart Garbage monitoring system", Indonesian Journal of Electrical Engineering and Computer Science, 2020.

[15]   Siti Sarah Md Ilyas, Muhammad Amirul Rosyad Mohd Halim, Nadia Abdul Wahab, Norfiza Ibrahim, "IoT-based Smart Garbage Monitoring", Journal of Computing Research and Innovation (JCRINN), 2021.

[16]   A. Hipparkar, Shivani Ahiwale, Vidya Dhumal, Shweta Gadekar, Supriya Nikude, "IoT Based Smart Garbage Collection System", Journal of Applied Science and Computations (JASC), 2019.

[17]   Ms. Harshita A. Gawad, Mr. Darsh Katkoria, Mr. Suraj Kadam, Ms. Dona T. Jain, "Garbage Monitoring System Using IoT", International Journal of Engineering Sciences & Research Technology (ISSN), 2018.

[18]   S. Jaya Kumar, B. Sai Praneeth, B. Sai Sumanth, M. Venkatesh, Puli Lohith, "Ultrasonic Assisted Smart Garbage Monitoring System", Journal of Network Communications and Emerging Technologies (JNCET), 2018.

[19]   Adarsh S, Mohamed Kaleemuddin S, Dinesh Bose, K I Ramachandran, "Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications", Centre for Computational Engineering and Networking (CEN), 2016.

[20]   S.S.Navghane, M.S.Killedar, Dr.V.M.Rohokale, "IoT Based Smart Garbage and Waste Collection Bin", International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), 2016.

[21]   Pooja Ajmera, "A Review Paper on Infrared sensor", International Journal of Engineering Research & Technology (IJERT), 2017.

[22]   Ali Khan, "Review of techniques and methods for object detection", International Journal of Advances in Computer Science and Technology, 2019.

[23]   Mr.Varun Chaudhary, Mr. Rohit Kumar, Mr. Anil Rajput, Mr.Manvendra Singh , ER. Thakurendra Singh, "Smart Dustbin", International Research Journal of Engineering and Technology (IRJET), 2019.

[24]   Luca De Nardis, Giuseppe Caso and Maria Gabriella Di Benedetto, "ThingsLocate: A ThingSpeak-Based Indoor Positioning Platform for Academic Research on Location-

Aware Internet of Things", Department of Mobile Systems and Analytics, Simula Metropolitan Center for Digital Engineering, 2019.

[25] K. Lova Raju, Member, IEEE, Md. Abbas Hussain, KVNS. Umesh Chandra, V. Naga Phanendhra, D. Narendra, "IoT Based Dust bin Monitoring System Using Node MCU", Innovations in Power and Advanced Computing Technologies, 2019.

[26] Ng Chee Ping & Lee Te Chuan, "Prototyping Smart Dustbin Using Arduino Uno Devices", Research in Management of Technology and Business, 2020.

[27] Ohwo, Onome Blaise, Dr. Alao D. Olujimi, "A Comparative Review of Emerging Wireless Technology", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2020.

[28] Muhammad Nasir Khan, Fawad Naseer, "IoT based University Garbage Monitoring System for Healthy Environment for Students", IEEE 14th International Conference on Semantic Computing (ICSC), 2020.

[29] Telugu Maddileti, Harish Kurakula, "Iot Based Smart Dustbin", International Journal of Scientific & Technology Research, 2020.

[30] Krishna Nirde, Prashant S. Mulay, Uttam M.Chaskar, "IoT based solid waste management system for smart city", International Conference on Intelligent Computing and Control Systems ICICCS, 2017.

[31] Xiang-Rui Huang, Shih-Sian Yang, Wei-Sheng Chen, Yong-Qi Zhang, Chun-Te Lee and Liang-Bi Chen, "An IoT-Based Smart Trash Cans Monitoring System", IEEE 10th Global Conference on Consumer Electronics (GCCE), 2021.

[32] Dr.T.M.N.Vamsi, Mr.G.Kalyan Chakravarthi, Mrs.Pratibha Lanka, Mr.B.Divakar, "An IoT Based Smart Garbage Monitoring and Disposal Support System", International Conference on Computing Methodologies and Communication (ICCMC), 2021.

[33] Riyan Hadi Putra, Feri Teja Kusuma, Tri Nopiani Damayanti, Dadan Nur Ramadan, "IoT: smart garbage monitoring using android and real time database", TELKOMNIKA, Vol.17, No.3, 2019.

[34] Suwetha I.G., "NodeMcu based smart garbage monitoring system", International Journal of Advance Research, Ideas and Innovations In Technology, 2019.

[35] Paleti Surya Teja, Motapothula Murali Krishna, Venkata Ratnam Kolluru, "Development of IoT based Garbage Management System using NodeMCU", International Journal of Engineering and Advanced Technology (IJEAT), 2019.

[36] Anthony Anggrawan, Sirojul Hadi, and Christofer Satria, "IoT-Based Garbage Container System Using NodeMCU ESP32 Microcontroller", Journal of Advances in Information Technology, 2022.

[37] P. Ramchandar Rao, S. Sanjay Kumar, Ch. Rajendra Prasad, "Garbage Monitoring System using Arduino", International Journal of Trend in Scientific Research and Development (IJTSRD), 2017.

[38] Mohamed Fezari, Neili Zakaria,Ali-Aldahoud, "Comparative study between two Powerfull NodeMCU Modules: ESP32 and ESP8266", Badji Mokhtar Annaba University, faculty of Engineering, 2019.

APPENDICES

Appendix A   Gantt Chart BDP 1 and BDP 2

BDP 1	WEEKS													
Task / Activities	1	2	3	4	5	6	7	8	9	10	11	12	13	14
PSM1 Briefing	■								M					
Find similar project		■							I					
Find journal/research paper			■						D					
Chapter 1				■	■									
Do comparison table from previous project					■	■								
Flow chart and block diagram project						■	■		S					
Chapter 2							■	■	E	■				
Finalise component used and design project									M	■				
Chapter 3											■	■		
Chapter 4									B		■	■		
Chapter 5									R			■		
Draft report submission									E			■		
Slide presentation									A				■	
Final report submission									K				■	
Presentation BDP 1														■

BDP 2	WEEKS													
Activities	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Talk introduction BDP 2	■													
Meeting supervisor		■	■	■	■	■	■	■	M	■	■	■	■	■
Hardware and software development	■	■	■	■					I					
System development				■	■				D					
System testing				■	■	■	■							
Troubleshoot						■	■	■						
Data tabulate and analysis									S	■				
Poster and slide									E		■	■		
BDP 2 submission									M				■	
Presentation BDP 2														■

Appendix B   IoT Based Garbage Monitoring System Full Coding

```cpp
#include <ESP8266WiFi.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#define BLYNK_PRINT Serial
#include <BlynkSimpleEsp8266.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);

char auth[] = "9H4RZqy6TIgIKBEx5t1YfUMourhki63t";
char ssid[] = "yeen";
char pass[] = "yeen1606";

Servo servo;
BlynkTimer timer;

#define trigPin D4 //Ultrasonic1
#define echoPin D5 //Ultrasonic1
#define trigPin2 D6 //Ultrasonic2
#define echoPin2 D7 //Ultrasonic2
#define BLYNK_MAX_SENDBYTES 256 //256 Bytes
WidgetLCD blynklcd(V2);
const int THRESHOLD = 20;
int MaxLevel = 35;
int binLevel = 0;

void setup()
{
 Serial.begin(9600);
 lcd.init();
 lcd.backlight();
 pinMode(trigPin,OUTPUT);
 pinMode(echoPin,INPUT);
 pinMode(trigPin2,OUTPUT);
 pinMode(echoPin2,INPUT);
 Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
 timer.setInterval(1000L, sendSensor);
}
```

```
void sendSensor()

{

  // ultrasonic 1

  long duration, distance;

  digitalWrite(trigPin, LOW);

  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);

  delayMicroseconds(10);

  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);

  distance = 0.017 * duration;


  // ultrasonic 2

  long duration2, distance2;

  digitalWrite(trigPin2, LOW);

  delayMicroseconds(2);

  digitalWrite(trigPin2, HIGH);

  delayMicroseconds(10);

  digitalWrite(trigPin2, LOW);

  duration2 = pulseIn(echoPin2, HIGH);

  distance2 = 0.017 * duration2;

  binLevel = map(distance2, 35,0,0,100);

  Blynk.virtualWrite(V0, binLevel);


  blynklcd.print(0,0,"Distance: " + String(distance2) + "cm ");

  Serial.print("Distance 1 = ");

  Serial.print(distance);

  Serial.println();

  Serial.print("Distance 2 = ");

  Serial.print(distance2);

  Serial.println();
```

```
if (distance >THRESHOLD || distance <= 0) // lid close
  {
  servo.attach(D8);
  servo.write(180);


  if (distance2 >= 20)
  {
   blynklcd.print(0,1,"Dustbin Not Full");
   lcd.setCursor(0,0);
   lcd.print("Dustbin Not Full");
   delay (1000);
   lcd.clear();
   blynklcd.clear();
  }
  else if (distance2 > 6 && distance2 < 20)
  {
   blynklcd.print(0,1,"Dustbin HalfFull");
   lcd.setCursor(5, 0);
   lcd.print("Dustbin");
   lcd.setCursor(4, 1);
   lcd.print("Half Full");
   delay(1000);
   lcd.clear();
   blynklcd.clear();
  }
 else if(distance2 <=6)
  {
   servo.detach();
   blynklcd.print(2,1,"Dustbin Full");
   lcd.setCursor(2, 0);
```

```
    lcd.print("Dustbin Full");

    delay(1000);

    lcd.clear();

    blynklcd.clear();

    Blynk.logEvent("dustbin_full","PLEASE COLLECT THE GARBAGE NOW !!!");

  }

}

  else

  {

  servo.write(0); //lid open

  }

    delay (500);

}

//open manually

  BLYNK_WRITE(V1) // this command is listening when something is written to V1

{

  int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable

  if (pinValue == 1) {

   servo.attach(D8);

   servo.write(0);

  } else if (pinValue == 0) {

   servo.attach(D8);

   servo.write(180);

  }

}

void loop ()

{

  Blynk.run();

  timer.run();

}
```

Appendix C   Blynk Mobile Interface

Appendix D   Blynk Web Interface