Faculty of Electrical and Electronics Engineering Technology

EARLY DETECTION SYSTEM FOR HOME FLOOD PREVENTION USING ARDUINO AND IOT

EZZAT MUHAMMAD SYAHMY BIN ASRI

Bachelor of Electronics Engineering Technology (Telecommunications) with Honours

2022

EARLY DETECTION SYSTEM FOR HOME FLOOD PREVENTION USING ARDUINO AND IOT

EZZAT MUHAMMAD SYAHMY BIN ASRI

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

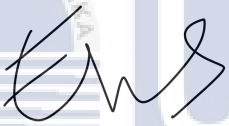
Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this project report entitled “EARLY DETECTION SYSTEM FOR HOME FLOOD PREVENTION USING ARDUINO AND IOT” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : 

Student Name : EZZAT MUHAMMAD SYAHMY BIN ASRI

Date : 11/1/2022

APPROVAL

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

Signature : .....

Supervisor Name : IR. DR. MOHD MUZAFAR BIN ISMAIL

Date : 11/1/2023

.....

DEDICATION

To my beloved mother, Rohana Binti Zainun, and my father, Asri Bin Redzuan,

Thank you for supporting me when I continue my studies

for bachelor's degree in UTeM.

To my sister, Nur Aiman Ezzaty Binti Asri,

Thank you for providing your creativity expertise and suggestion

for completing this project.

To my brother, Muhammad Eiman Bin Asri,

Thank you for your emotional support.

ABSTRACT

During this of globalization, floods are the major disaster that are always occurs in the country, Malaysia. The country is affected by heavy rainfall largely during the end of the year which are the typical for seasonal monsoon floods. Flood disasters such as flash floods and tidal floods that occurs mostly on houses or residential that are close to the seaside and rivers can be devastating to those who choose to live there. The damage done to the property and even innocent lives that are loss to this natural disaster occurred every year can be potentially avoided with a system. The system that can signal if the water around the residential area is rising rapidly and have a blockade for the flood water to enter the residential place. With that, no to minimal damage will be done to the places. This system could also be used in industrial places with using the same concept, albeit in much bigger scale compared to residential area system. The expected outcome of this system is to have minimal property damages done to homes and safety to resident from floods. An Ultrasonic sensor is chosen in this project because of having high sensing distance while having no moving parts thus making it more reliable compared to other alternative water sensors. This system is main equipped with Arduino UNO connected to water sensors to track water level on simulated water drainage. When specified water level is reach, it will trigger the water gate system. The data from water sensors is also connected to NodeMCU ESP8266. The user will be able to do monitoring for the water level by using the mobile application. This project is aimed at developing and design flood monitoring systems with an automation for water gate, with addition of able to monitor in real-time using mobile phone application through IoT platforms. The project is presented in a small-scale concept of residential houses, simulated water drainage and functioning prototype of the system. This project is suitable for residential and industrial area for safety measure against flood disaster. IoT functionality are efficient for monitoring area from afar. For future works, the system can be implemented and tested in real world residential house and add a reliable uninterrupted power source such as solar powered battery.

ABSTRAK

Dalam era globalisasi ini, banjir merupakan bencana besar yang selalu berlaku di negara, Malaysia. Negara ini terjejas oleh hujan lebat sebahagian besarnya pada akhir tahun yang merupakan tipikal untuk banjir monsun bermusim. Bencana banjir seperti banjir kilat dan banjir pasang yang kebanyakannya berlaku di rumah atau kediaman yang berhampiran dengan tepi laut dan sungai boleh memudaratkan mereka yang memilih untuk tinggal di situ. Kerosakan yang dilakukan kepada harta benda dan juga nyawa yang tidak berdosa yang merupakan kerugian akibat bencana alam ini berlaku setiap tahun berpotensi dapat dielakkan dengan sistem; itu boleh memberi isyarat jika air di sekitar kawasan perumahan itu naik dengan cepat, dan mempunyai sekatan untuk air banjir memasuki kawasan perumahan. Dengan itu, tiada kerosakan yang minimum akan berlaku kepada tempat-tempat tersebut. Sistem ini juga boleh digunakan di kawasan perindustrian dengan konsep yang sama, walaupun dalam skala yang lebih besar berbanding sistem kawasan kediaman. Hasil yang diharapkan daripada sistem ini ialah kerosakan harta benda yang minimum dilakukan kepada rumah dan keselamatan kepada penduduk daripada banjir. Penderia Ultrasonik dipilih dalam projek ini kerana mempunyai jarak penderiaan yang tinggi sementara tidak mempunyai bahagian yang bergerak sehingga menjadikannya lebih dipercayai berbanding dengan penderia air alternatif yang lain. Sistem ini dilengkapi dengan Arduino UNO yang disambungkan kepada penderia air untuk mengesan paras air pada saliran air simulasi. Apabila paras air tertentu dicapai, ia akan mencetuskan sistem pintu air. Data daripada penderia air juga disambungkan ke NodeMCU ESP8266. Pengguna akan dapat melakukan pemantauan paras air dengan menggunakan aplikasi mudah alih. Projek ini bertujuan untuk membangunkan dan mereka bentuk sistem pemantauan banjir dengan automasi untuk pintu air, dengan tambahan boleh memantau dalam masa nyata menggunakan aplikasi telefon mudah alih melalui platform IoT. Projek ini dibentangkan dalam konsep berskala kecil rumah kediaman, simulasi saliran air dan prototaip sistem yang berfungsi. Projek ini sesuai untuk kawasan perumahan dan perindustrian bagi langkah keselamatan terhadap bencana banjir. Fungsi IoT adalah cekap untuk memantau kawasan dari jauh. Untuk hasil kerja akan datang, sistem ini boleh dilaksanakan dan diuji di rumah kediaman dunia sebenar dan menambah sumber kuasa tanpa gangguan yang boleh dipercayai seperti bateri berkuasa solar.

ACKNOWLEDGEMENTS

To complete this project, many people had helped and inspired me. I have received a lot of support from each of them.

First and foremost, I would like to express my gratitude to my supervisor, IR. DR. MOHD MUZAFAR BIN ISMAIL for their precious guidance, words of wisdom and patient throughout this project.

I am also my fellow colleague and housemates for the willingness of assisting me regarding the project. They are never relentless to share their knowledge, information, and experience with me.

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

An honourable mention also goes to both of my long-time friends for brainstorming together to form this project idea.

TABLE OF CONTENTS

DECLARATION ..... d
APPROVAL ..... e
DEDICATION ..... f
ABSTRACT ..... g
ABSTRAK ..... h
ACKNOWLEDGEMENTS i
TABLE OF CONTENTS i
LIST OF TABLES iii
LIST OF FIGURES iv
LIST OF ABBREVIATIONS, SYMBOLS AND NOMENCLATURE vi
LIST OF APPENDICES vii
CHAPTER 1 INTRODUCTION 1
1.1 Research Background 1
1.2 Problem Statement 3
1.3 Project Objective 3
1.4 Scope of Project 3
CHAPTER 2 LITERATURE REVIEW 4
2.1 Introduction 4
2.2.1 Flood disaster indicator of water level monitoring system 5
2.2.2 A Low-Cost Flash Flood Monitoring System 5
2.2.3 SMS based flood monitoring and early warning system 6
2.2.4 Flood Monitoring and Early Warning System Using Ultrasonic Sensor 7
2.2.5 Web Camera Sensor Coupled With Lidar Data Flood Map For Flood Warning System 8
2.3 Journal Comparison for Relevant Previous Research 9
2.4 Summary 14
CHAPTER 3 METHODOLOGY 15
3.1 Introduction 15
3.2 Project Workflow 15
3.2.1 Project Planning and Flow Chart 16
3.3 System Operation 18
3.4 System Development 20
3.4.1 NodeMCU ESP8266 21

3.4.2 Arduino 23
3.4.3 Ultrasonic Sensor HC-SR04 25
3.4.4 Blynk 26
3.5 Electrical Diagram 28
3.6 Preliminary Result 30
3.7 Summary 33
CHAPTER 4 RESULTS AND DISCUSSIONS 34
4.1 Introduction 34
4.2 Software Setup 34
4.3 Hardware Setup 36
4.4 Result Analysis 40
4.5 Summary 51
CHAPTER 5 CONCLUSION 52
5.1 Conclusion 52
5.2 Future Works 53
REFERENCES 54
APPENDICES 56
Appendix A: Project Gantt Chart Bachelor Degree Project 1 56
Appendix B: Project Gantt Chart Bachelor Degree Project 2 57
Appendix C: Arduino UNO coding 58
Appendix D: NodeMCU ESP8266 coding 62

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1	Journal Comparison	10
Table 3.1	List of Devices	20
Table 3.2	LoLin NodeMCU ESP8266 Specification	22
Table 3.3	Arduino UNO Specification	23
Table 3.4	Ultrasonic sensor specification	25
Table 4.1	Output displayed on Water Gate system	48
Table 4.2	Output displayed on Blynk IoT application in mobile phone	49

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 3.1	Process Flow	15
Figure 3.2	Project Flowchart PSM 1	16
Figure 3.3	Project Flowchart PSM 2	17
Figure 3.4	System structure of Early Home Flood Prevention System	18
Figure 3.5	Flowchart of the Early Home Flood Prevention System	19
Figure 3.6	LoLin style NodeMCU ESP8266	21
Figure 3.7	LoLin NodeMCU Pinout with label	21
Figure 3.8	Arduino UNO	23
Figure 3.9	Arduino UNO board with label	24
Figure 3.10	Ultrasonic Sensor HC-SR04	25
Figure 3.11	Blynk IoT Platform	26
Figure 3.12	Blynk flow diagram	27
Figure 3.13	Electrical Diagram	28
Figure 3.14	Circuit Design	29
Figure 3.15	Proof of concept of the system on TinkerLab	30
Figure 3.16	Proof of concept of the system on TinkerLab	31
Figure 3.17	Testing connection of the project	32
Figure 3.18	Testing the pre-built of the project	32
Figure 4.1	NodeMCU ESP8266 coding in Arduino IDE	35
Figure 4.2	Main System Hardware Set-Up	37
Figure 4.3	Main System Hardware when cover closed	37

Figure 4.4	Simulated Water Drainage System Hardware Set-Up	38
Figure 4.5	Water Gate System Hardware Set-Up	38
Figure 4.6	Project Connection Hardware Set-Up	39
Figure 4.7	Sensor range for Simulated Water Drainage system	41
Figure 4.8	LCD display on the system	41
Figure 4.9	UI display on Blynk mobile phone application	42
Figure 4.10	System in Normal State	43
Figure 4.11	System in Orange State	44
Figure 4.12	System in Alert State	45
Figure 4.13	Blynk Notification through the mobile phone application	46
Figure 4.14	Blynk Notification through the mobile phone application	47
Figure 4.15	Water Gate Lowered	48
Figure 4.16	Water Gate Raised	48
Figure 4.17	Blynk Mobile Dashboard at 25%	49
Figure 4.18	Blynk Mobile Dashboard at 50%	49
Figure 4.19	Blynk Mobile Dashboard at 100%	50

LIST OF ABBREVIATIONS, SYMBOLS AND NOMENCLATURE

A	-	Ampere Current
API	-	Application Programming Interface
EEPROM	-	Electrically Erasable Programmable Read-Only Memory
GSM	-	Global System Mobile Communication
GUI	-	Graphic User Interface
IDE	-	Integrated Development Environment
I/O	-	Input / Output
IoT	-	Internet of Thing
SMS	-	Short Message Service
UI	-	User Interface
UPS	-	Uninterruptible Power Supply
USB	-	Universal Serial Bus
V	-	Voltage
W	-	Watt, Unit of Power

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
Appendix A	Project Gantt Chart Bachelor Degree Project 1	54
Appendix B	Project Gantt Chart Bachelor Degree Project 2	55
Appendix C	Arduino UNO coding	56
Appendix D	NodeMCU ESP8266 coding	60

CHAPTER 1

INTRODUCTION

This chapter will discuss about the background of the project, problem statement, objective, scope of the project and the project outline.

1.1 Research Background

Arduino board is a type microcontrollers kits or single board that have been use in variety type of project. Most of prototype and even final product have taken benefit of the flexibility using Arduino board as it is much inexpensive and already comes with and open supply hardware and software with the board design use a variety of microprocessors and controllers. Digital and analogue input/output (I/O) pins on the boards can be used to connect to expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards have serial communications interfaces, as well as USB ports on some variants, which are used to load programmes. The Arduino programming language, which is based on Processing and comes with a modified version of the Processing IDE, may be used to programme microcontrollers using C and C++ programming languages and a standard API. In addition to standard compiler toolchains, the Arduino project provides an integrated development environment (IDE) and a command line tool built in Go, a compiled programming language developed by Google. [1]

The Harvard architecture is used in the Arduino board's processor, which includes separate memory for programme code and programme data. Programme memory and data memory are the two types of memories available. Data is stored in data memory, whereas code is stored in flash programme memory. With 32kb of flash memory, 2kb of SRAM, and 1kb of EPROM, the Atmega328 microcontroller runs at 16MHz. [2]

The main circuit for the project is tested and developed using SimulIDE. The coding to be written in propriety Arduino IDE software. It will then test in SimulIDE with sensor and other circuits connected to simulated Arduino board loaded with the code written. A Wi-Fi module are implemented to the system for IoT system using Blynk that is used to monitoring via the Internet.

The main objective of this paper is to bring notice that matter to prevent proprietary damage to residential area can be avoided with an efficient system built and implemented properly. In addition, the system can also be implemented in other purpose if the customer demands for other usage. This study may prove study to be useful to those who live in residential area with a high risk of flood within every year.

1.2 Problem Statement

The citizen of our country Malaysia has to face horrible natural disaster which is flood every year during the monsoon season. [3] Although preventive measure such as remove the clogging within the drain system or expand the water flow system in the residential area, the heavy rain that comes overnight have able to flood a portion of the houses that are place in low or sloped land area. Furthermore, this study could introduce a type of IOT device that can be used by user to monitor the water level around the residential area.

1.3 Project Objective

The main aim of this project is to propose a systematic and effective methodology to make system to control the water flow of flood by preventing water entering the residential area. Specifically, the objectives are as follows:

a) To design a monitoring flood system and an automated gated system that can be activated when incoming water flow that exceed the limitation given.
b) To detect the level of the water flood using sensor which will be divided into multiple level.
c) To analyse the effectiveness of the automated system with Internet of Things (IoT) system architecture.

1.4 Scope of Project

The scope of this project is as the follows:

a) Study the which hardware of that are suitable for the project.
b) Determine which type of sensors that are suitable for the water level.
c) Design is functioning hardware component of the project
d) Simulate the prototype by using TinkerCAD, Arduino IDE and Blynk.
e) The project would be finished in at least a small-scale prototype as the original scale would be much costly.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Several writers and report mentioned an inevitable cases of flood natural disaster occur every year. Even its seems that these cases have been addressed through government initiative such as the SMART Tunnel [4], it is still a huge issue in rural area around the country that does not have an efficient proper drainage system that can combat the flood problem. [5]

Prototype developed to solve this issue can be made by using a microcontroller such as Arduino. Arduino is an open-source electronics construction and programming platform. It can receive and send data to most devices, as well as command specific electronic equipment over the internet. The board is programmed using an Arduino Uno circuit board and a software programme (Simplified C++). [6]

In this chapter, I have been comparing others project related with the project. It is related to the function and operation of each component. All the comparison process during the literature review were made to complete the Early Home Flood Prevention System.

2.2.1 Flood disaster indicator of water level monitoring system

With the advancement of technology, early warning systems for flood management have been swiftly developed. These systems use the Short Message Service (SMS) via the Global System for Mobile Communications to inform individuals early (GSM). This paper describes a simple, portable, and low-cost early warning system that employs an Arduino board to manage the system and GSM shields to transmit data. The hardware and software components of the system have been designed and implemented. The device uses float switch sensors to assess the water level, then analyses the data to determine the type of hazard present. [7] An alarm message is issued to the user based on the observed level. The GSM network is utilised to send SMS messages between the various system components.

2.2.2 A Low-Cost Flash Flood Monitoring System

A flash flood is a disaster that necessitates rapid identification in order to avoid fatalities. [8] The concept envisioned a low-cost solution based on an Android smartphone and an Arduino board. The system will use an echolocation approach to assess water level, collect data from its sensors, and communicate data to the server over a mobile network or long-range WIFI. Low-level features, such as colour, are analysed at the phone as a network connection decision trigger in order to save network costs. Other high-level characteristics are extracted at the server to identify whether it is a flash flood.

2.2.3 SMS based flood monitoring and early warning system

This study shows how an SMS-based Flood Monitoring and Early Warning system is designed, implemented, and experimentally verified. [9] SMS will be used for tools like credit top-up and maintaining contact numbers. Users would receive texted updates on the water level's height if they requested it. When the level of water exceeds the user-defined threshold value, the system sends an SMS alert to the at-risk or threatened population as well as necessary authorities. The Global System for Mobile Communications (GSM) module is used to deliver text messages to mobile phones, while the Arduino Uno CPU reads the pressure sensor's input and calculates the water level. This basic yet effective warning system is thought to be one of the most efficient and cost-effective ways to notify the appropriate authorities and the susceptible residence. [3]

2.2.4 Flood Monitoring and Early Warning System Using Ultrasonic Sensor

The goal of this research is to create a real-time flood monitoring and early warning system in Isabela's northern region, particularly in the communities bordering the Cagayan River. Ultrasonic sensing techniques have matured to the point that they are now frequently used in engineering and basic science. Ultrasonic sensing has the unique capacity to probe inside an object without causing damage since ultrasound can travel through any medium, including solids, liquids, and gases. The water level detection and early warning system (through website and/or SMS) that warns concerned agencies and individuals to a potential flood occurrence is the subject of this study. Furthermore, an inquiry system is added in this study to make it more participatory, so that members of the community can use SMS keywords to question about the current water level and status of the desired area or location affected by flood. The study's goal is to help individuals be prepared and informed in the event of a flood. The use of Arduino, ultrasonic sensors, GSM module, web-monitoring, and SMS early warning system in assisting stakeholders in mitigating flood casualties is what makes this effort unique. The paper aims to assist flood-prone areas throughout the country, notably in the province's local villages. It is, without a doubt, significant and important in terms of community safety and wellbeing. [10]

2.2.5 Web Camera Sensor Coupled With Lidar Data Flood Map For Flood Warning System

A Flood Warning System is created in this study using Water Level Data imaged from a simple web camera. The system converts the water level image's output data into tables, then connects to the flood map database to choose the appropriate map for presentation as a flood prediction image using a GIS application. The technology uses real-time data processing to deliver reliable flood area predictions to the public. By using this technique, users may predict where areas will be flooded as the water level rises, allowing residents in the affected region to escape or at the very least prepare for the impending flood. This type of early warning system is projected to save not only the lives of residents in the affected area, but also their possessions from the flood calamity. The flood area is split at 1 m intervals from LiDAR data, and the water level is measured every 10 cm. The resulting area, calculated using elevation data, is used to define the flood's greatest extent. Each map layer corresponds to a specific water level captured by the camera. [11]

2.3 Journal Comparison for Relevant Previous Research

Table 2.1 show the comparison table for 5 different research related to this proposed project so that we can improve or avoid the drawback for this project after analysing it.

Between the research had been made, most of the project used the same method of systematic workflow with the module and microcontroller. The GSM module are widely used in previous project as it can be using its own network connectivity without dependence to other connection such as WIFI. It also been easy to setup and use which is why it is out of the common out of choices.

Next, almost all the project that have been researched here are using microcontroller Arduino which is easy to use and program. It also user friendly because it an open-source program. The program also easy to learn from the internet source.

Moreover, most of the research choose a variety of different sensor for its similar problem statement which is flood. For this project, ultrasonic sensor is chosen as it more suitable for the area which is residential houses.

Table 2.1: Journal comparison

Title	Microcontroller	Monitoring system	Type of sensor	Communication system	Function	Advantages	Disadvantages
Flood disaster indicator of water level monitoring system [7]	Arduino Uno board	Mobile phone SMS	Float switch sensors (RSF50 Series)	GSM Module	Uses float switch sensors to assess the water level, then analyses the data to determine the type of hazard present. An alarm message is issued to the user based on the observed level. The GSM network is utilised to send SMS messages between the various system components.	Float switch sensors are known to be cheaper alternative sensors compare to ultrasonic sensor.	Turbulence in the fluid may cause the float to move.

A Low-Cost Flash Flood Monitoring System [8]	IOIO board	Android Module	Echolocation sensor, temperature sensor, camera	GSM Module	The system will use an echolocation approach to assess water level, collect data from its sensors, and communicate data to the server over a mobile network or long-range WIFI.	IOIO board can be use or program using Android system	Noise and variant climate are mainly interference to the system.
SMS based flood monitoring and early warning system [9]	Arduino Uno board	Mobile phone SMS	Barometric pressure sensor (SM5100B)	GSM Module	Users would receive texted updates on the water level's height if they requested it. When the level of water exceeds the user-defined threshold value, the system sends an SMS alert to the at-risk or threatened population as well as necessary authorities.	Pressure sensor are more sensitive and has large measuring range.	Need high power source, cannot use solar powered or low power delivery source.

Flood Monitoring and Early Warning System Using Ultrasonic Sensor [10]	Arduino Uno board	Mobile phone SMS	Ultrasonic Sensor	GSM Module	Ultrasonic sensing has the unique capacity to probe inside an object without causing damage since ultrasound can travel through any medium, including solids, liquids, and gases. The water level detection and early warning system (through website and/or SMS) that warns concerned agencies and individuals to a potential flood occurrence is the subject of this study.	Two places monitoring system, can cover large area for monitoring	Expensive as needed two system, not suitable for residential area coverage.

Web Camera Sensor Coupled with Lidar Data Flood Map For Flood Warning System [11]	Computer	Computer	Ultrasonic sensor, Camera	Computer data collection	A Flood Warning System is created in this study using Water Level Data imaged from a simple web camera. The system converts the water level image's output data into tables, then connects to the flood map database to choose the appropriate map for presentation as a flood prediction image using a GIS application. The technology uses real-time data processing to deliver reliable flood area predictions to the public.	Can cover large area for monitoring	Not suitable with its camera dependant system for residential area coverage, expensive.

2.4 Summary

Various project benefits and drawbacks are mentioned towards the conclusion of this chapter. It's much easier to put past researchers' approaches into practise in this project. Furthermore, the drawbacks can be as small as possible. The findings of earlier researchers' research have a favourable impact on this project since the correct and proper strategy will be taken while keeping the weak point in mind. After reading their research, I would come out with ideas for decision making for suitable component and ideas for rooms of improvement to the system to fit the residential area system.

CHAPTER 3

METHODOLOGY

3.1 Introduction

For a project, a proper data collecting, a system of methods used for producing must be recorded and written appropriately. It is to keep the project workflow more smoothly and keeping it more organized for report achieving.

3.2 Project Workflow

This process is separated into five major stages: planning, research, analysis, design, and execution. The planning stage is the most important. The planning phase establishes the project's expected milestones and timeframe for completion. This will ensure that all operations are completed on time and that the project is completed on time. Figure 3.1 illustrates the high level of project process flow.

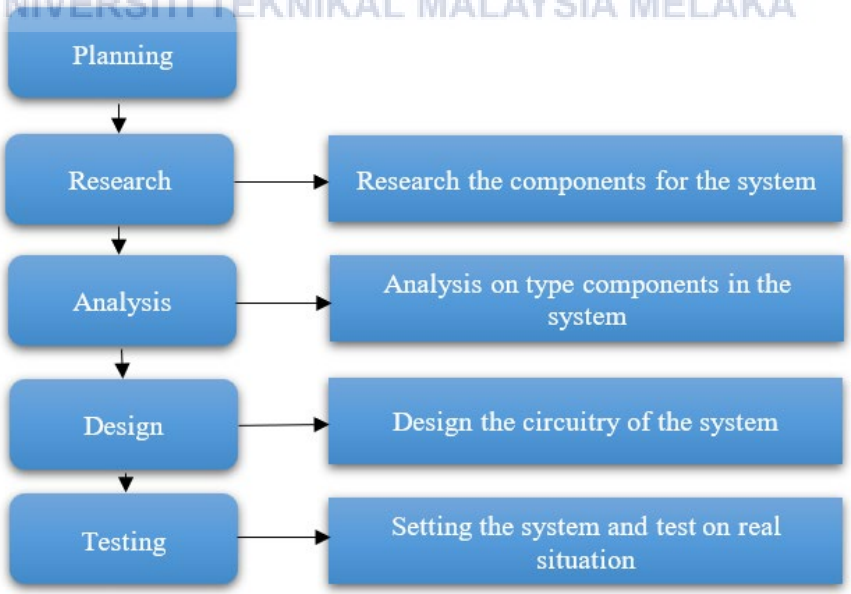


Figure 3.1: Process Flow

3.2.1 Project Planning and Flow Chart

Project is organised to ensure the timeline can be met. Figure 3.2 illustrates the project flowchart while the project Gant Chart which summarized the activities and the timeline are included in the appendix.

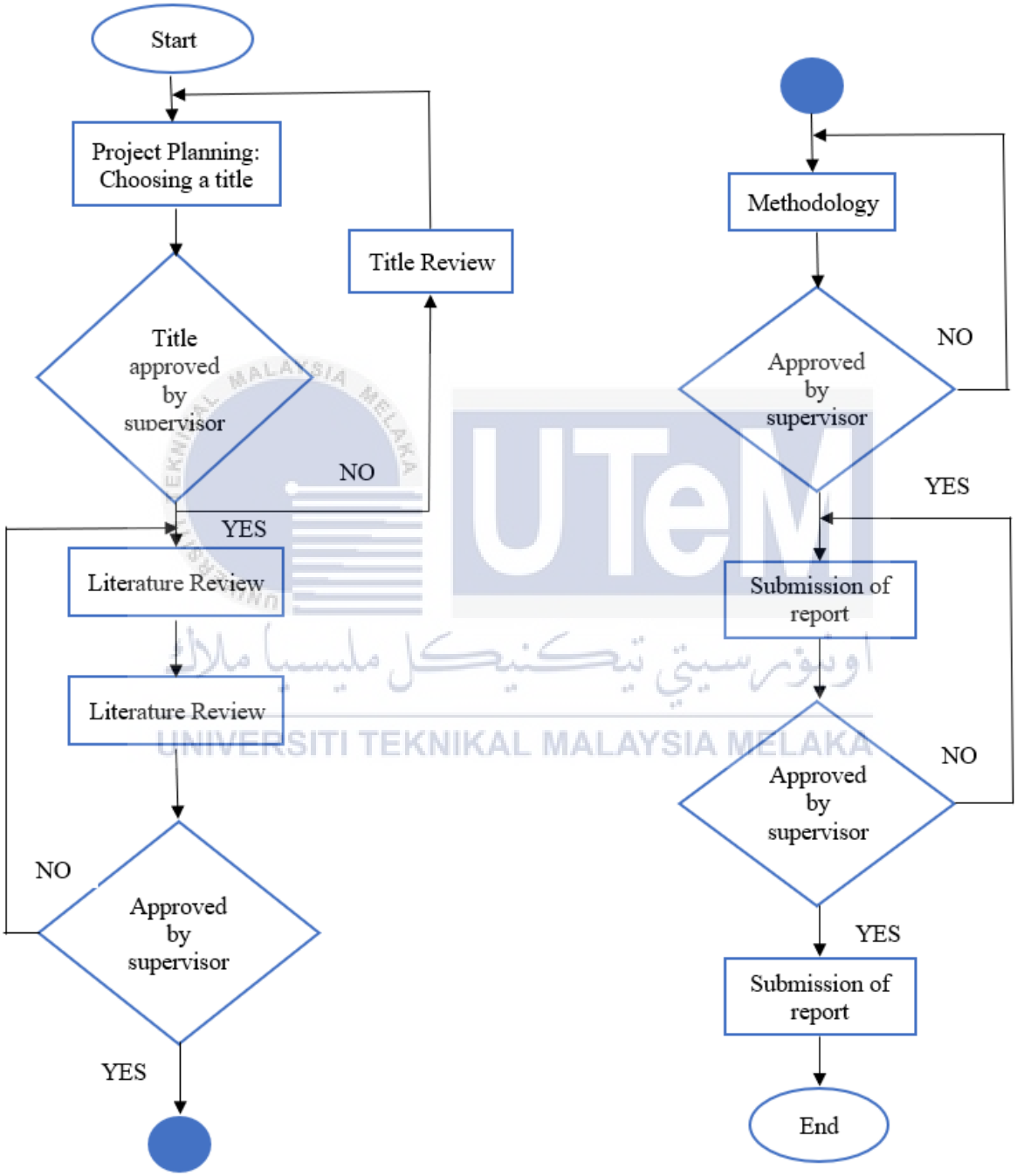


Figure 3.2 Project Flowchart PSM 1

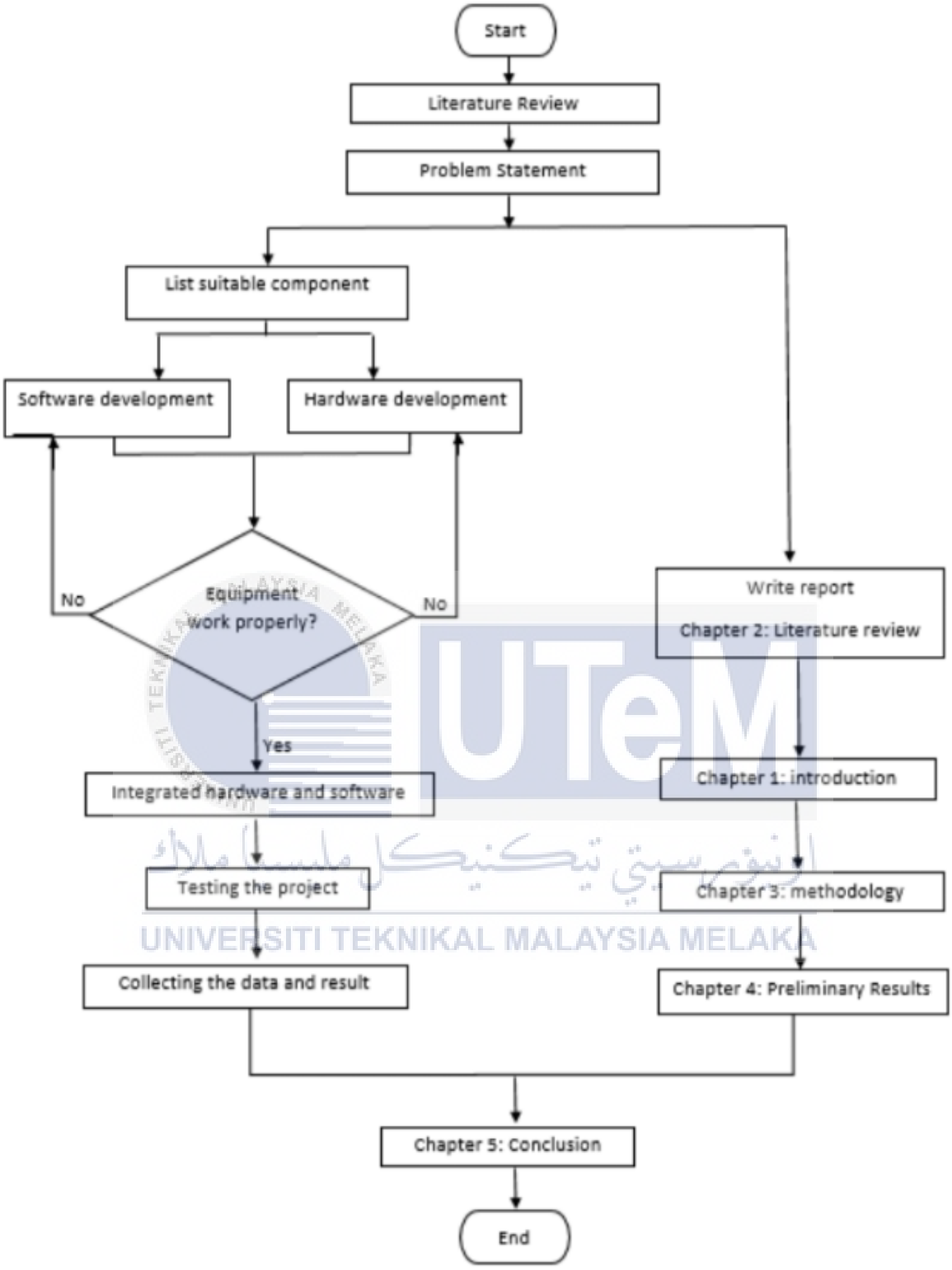


Figure 3.3 Project Flowchart PSM 2

3.3 System Operation

This paper presents a flow on how the project system could work. It could give a greater understanding of how to improve the aspects of the project system. The guidance is consisted of selection of equipment. Figure below will show the flow step by step to accomplish the project.

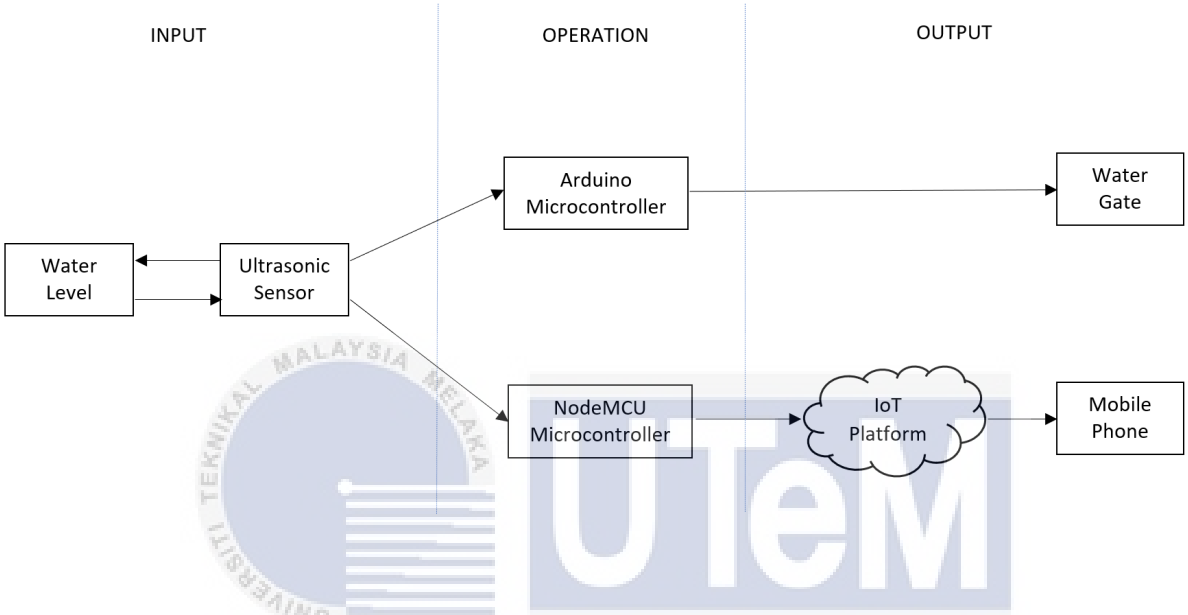


Figure 3.4 System structure of Early Home Flood Prevention System

Ultrasonic sensors would emit waves frequency to the water and the calculate its distance based on the time required for the frequency to reflect back. [12] The information from ultrasonic sensor then be sent to Arduino which it determines if it reaches normal or danger levels while at the same time display the water level information on LCD attached to the Arduino. In the case of water reaches danger level, Arduino will initiate the water gate to close, while at the same time Wi-Fi module sent alerts messages to mobile phone, through IoT platform.

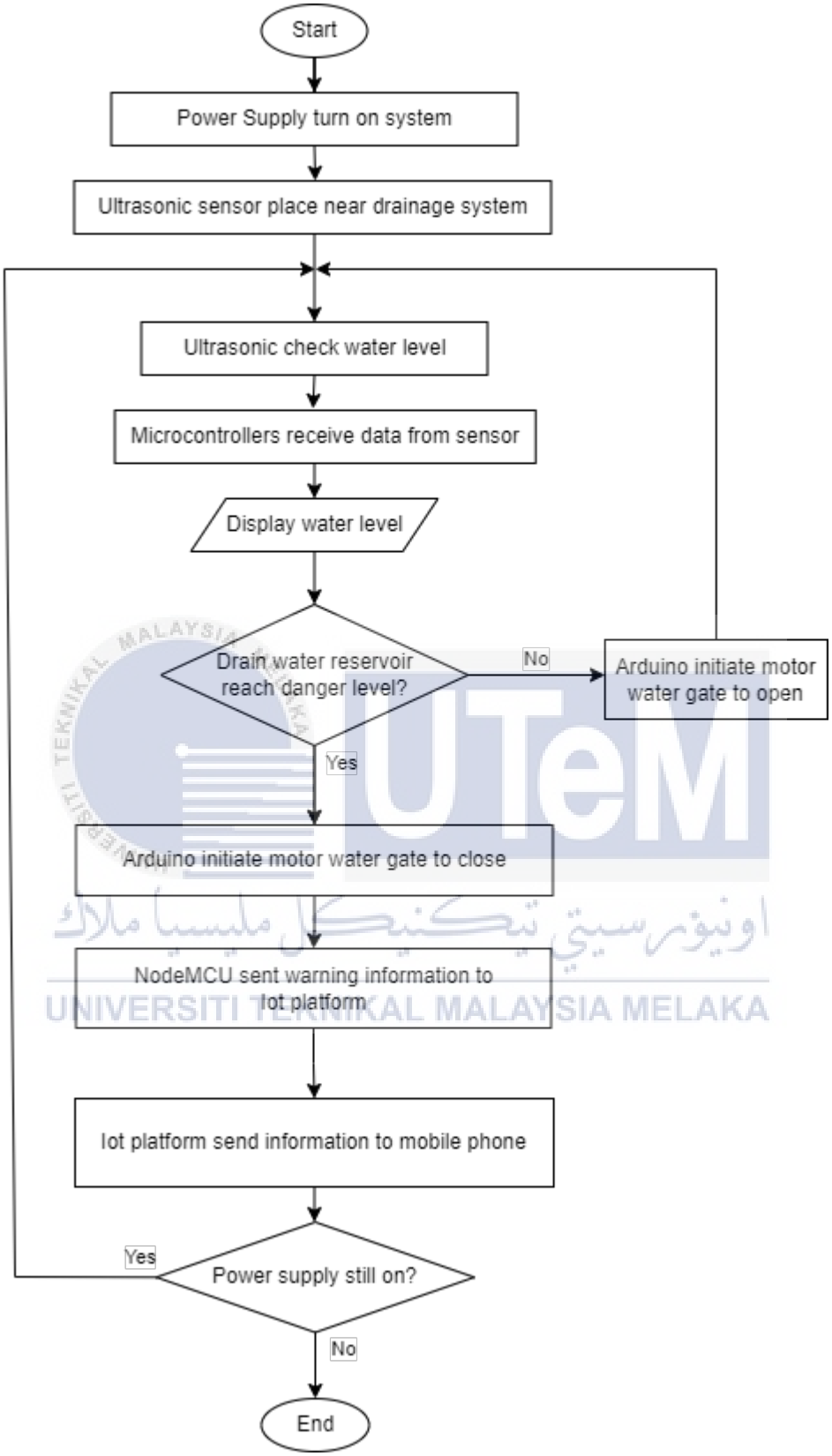


Figure 3.5 Flowchart of the Early Home Flood Prevention System

3.4 System Development

This section will explain the device and materials that used in this small-scale prototype of project. The ultrasonic sensor will give out the height of the flood water, it then feed the information to the Arduino. This information also retrieves by NodeMCU Wi-Fi module for monitoring from mobile phone application through IoT platform. When reach certain requirements, the water gate will initiate while the Wi-Fi module will send out alert notification to mobile phone application. Blynk is chosen as the IoT platform for this project for its easy-to-understand coding construction and implementation of IoT infrastructure

Table 3.1 List of devices

Component	Quantities
NodeMCU ESP8266	1
Arduino UNO	1
LED	3
LCD Display	1
Buzzer	1
Servo Motor 9G	1
Water Sensor	1
Ultrasonic Sensor HC-SR04	1

3.4.1 NodeMCU ESP8266

A low-cost System-on-a-Chip (SoC) called the ESP8266 serves as the foundation of the NodeMCU (Node Microcontroller Unit), an open source IoT platform and hardware development environment. The Espressif Systems-designed and -produced ESP8266 has all of the essential components of a computer, including CPU, RAM, networking (Wi-Fi), and even a contemporary operating system and SDK. This makes it a fantastic option for all types of IoT projects. The form factor model that is chosen for this project is LoLin style NodeMCU measures 58mm x 32mm with a pin spacing of 0.1″ between pins and 1.1″ between rows.

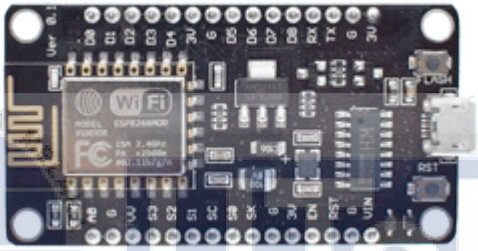


Figure 3.6 LoLin style NodeMCU ESP8266

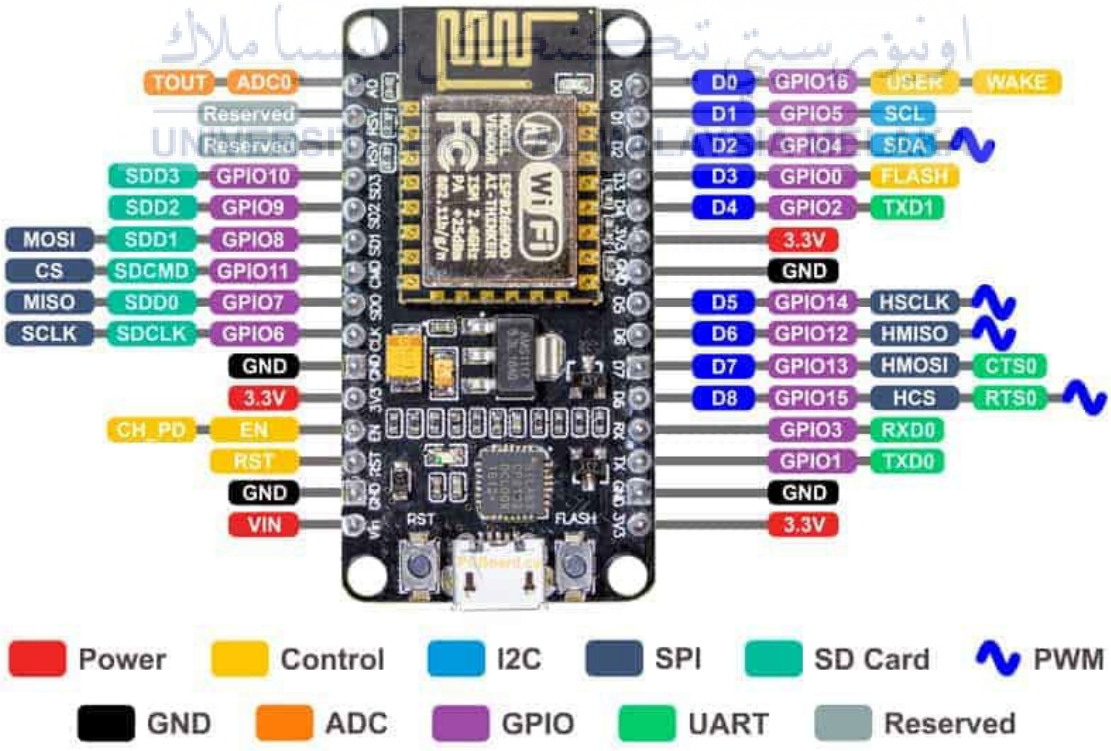


Figure 3.7 LoLin NodeMCU Pinout with label

The specification of LoLin style NodeMCU ESP8266 is as shown below:

Table 3.2 LoLin NodeMCU ESP8266 Specification

Parameter	Specification
Microcontroller	ESP-8266 32-bit
NodeMCU Model	Clone LoLin
NodeMCU Size	58mm x 32mm
Pin Spacing	1.1" (27.94mm)
Clock Speed	80 MHz
USB to Serial	CH340G
USB Connector	Micro USB
Operating Voltage	3.3V
Input Voltage	4.5V-10V
Flash Memory/SRAM	4 MB / 64 KB
Digital I/O Pins	11
Analog In Pins	1
ADC Range	0-3.3V
UART/SPI/I2C	1 / 1 / 1
Wi-Fi Built-In	802.11 b/g/n
Temperature Range	-40 °C - 125 °C

3.4.2 Arduino

Table 3.3 Arduino UNO Specification

Parameter	Specification
Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC current for 3.3V Pin	50mA
Flash Memory	32 KB (ATmega328), 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

Figure 3.8 Arduino UNO

The Arduino system provides digital and analogue input/output (I/O) pins for interfacing numerous expansion boards and additional circuits such as servo motors and

ultrasonic sensors. [17] The USB serves as a serial communication interface, loading the programme and connecting the Arduino to the system's serial monitor. The pins on the Arduino UNO board that contribute to Arduino functionality are shown in the diagram below.

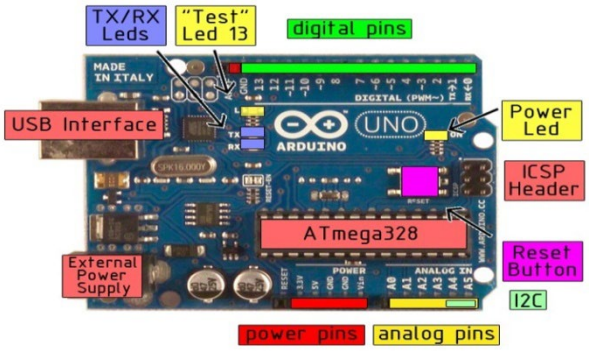


Figure 3.9: Arduino UNO board with label.

In this project, Arduino acts as a link between the other components, for example, it acts as a link between the system's input and output. The Arduino UNO was chosen above other microprocessors for this project because it has a simple operating system, is easy to configure, is inexpensive, and has a huge software library.

3.4.3 Ultrasonic Sensor HC-SR04

The HC-SR04 ultrasonic sensor uses sonar to determine the distance to an object. Ultrasonic sensor is used for detecting the flood water level around the residential area. This sensor is chosen because it can detect more large area and inexpensive. [18] Two ultrasonic transducers are at the heart of the HC-SR04 Ultrasonic Distance Sensor. One acts as a transmitter, turning electrical signals into ultrasonic sound pulses with a frequency of 40 KHz. The receiver listens for the broadcast pulses. If it receives them, it produces an output pulse, the width of which can be used to measure the pulse's travel distance. [19]

Figure 3.10 Ultrasonic Sensor HC-SR04

Table 3.4 Ultrasonic sensor specification

Parameter	Specification
Operating voltage	3.2 – 5.2 V
Operating current	8mA
Ultrasonic frequency	40kHz
Measuring range	2-400cm
Resolution	1cm
Output	PWM
Measurement angle	15°
Working temperature	-10~60 °C

3.4.4 Blynk

A software provider that offers infrastructure for the internet of things is called Blynk. Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things. Blynk invented the no-code method of producing IoT apps in 2014 and became well-known throughout the world for its mobile app editor. Applications that put things, people, and data at the centre of company processes are combined with a cloud platform by Blynk. There are three major components in the platform:

Blynk App - allows to you create amazing interfaces for your projects using various widgets we provide.

Blynk Server - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. It's open source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.

Blynk Libraries - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

Coding to be upload to microcontroller can be written using Arduino IDE software. The Blynk IoT platform will provided additional coding template so the microcontroller having access to Internet can access Blynk platform.

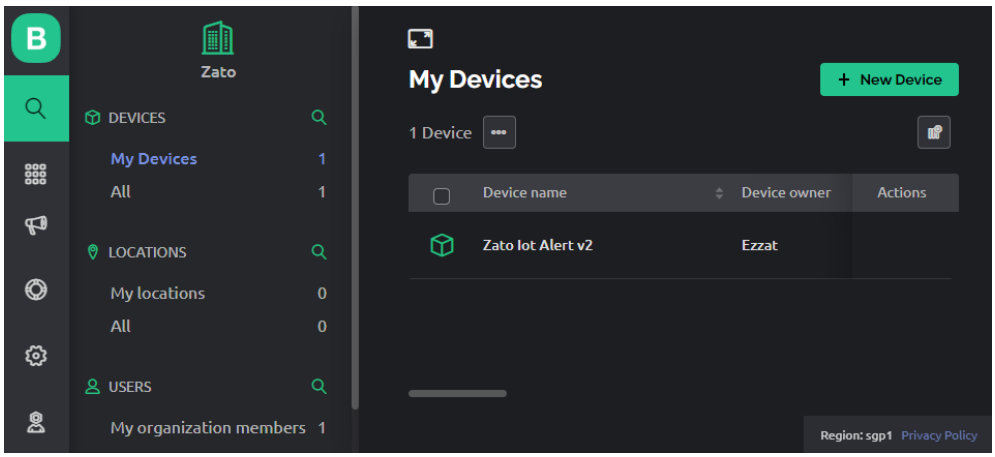


Figure 3.11 Blynk IoT Platform

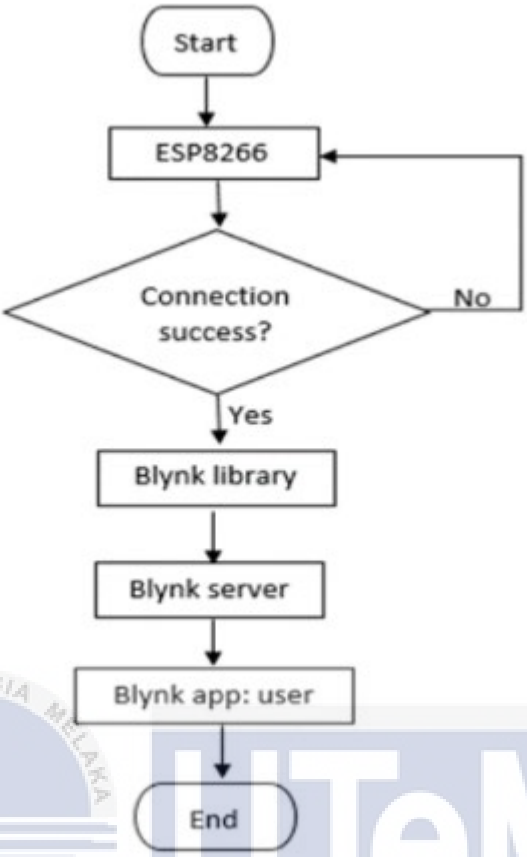


Figure 3.12 Blynk flow diagram

3.5 Electrical Diagram

The Arduino are equipped with LCD display, LED indicator and buzzer. The system for flood monitoring system is using ultrasonic sensor that would input inserted to Arduino. The LCD would display the condition and water level with information provided from the sensor. [20] Wi-Fi module is inserted for online notification services that will send to mobile phones. The servo motor act as water gate system is also then connected to the Arduino, with specified condition it will initiate the gate. NodeMCU ESP8266 will also receive date from ultrasonic sensor to be send to IoT platform.

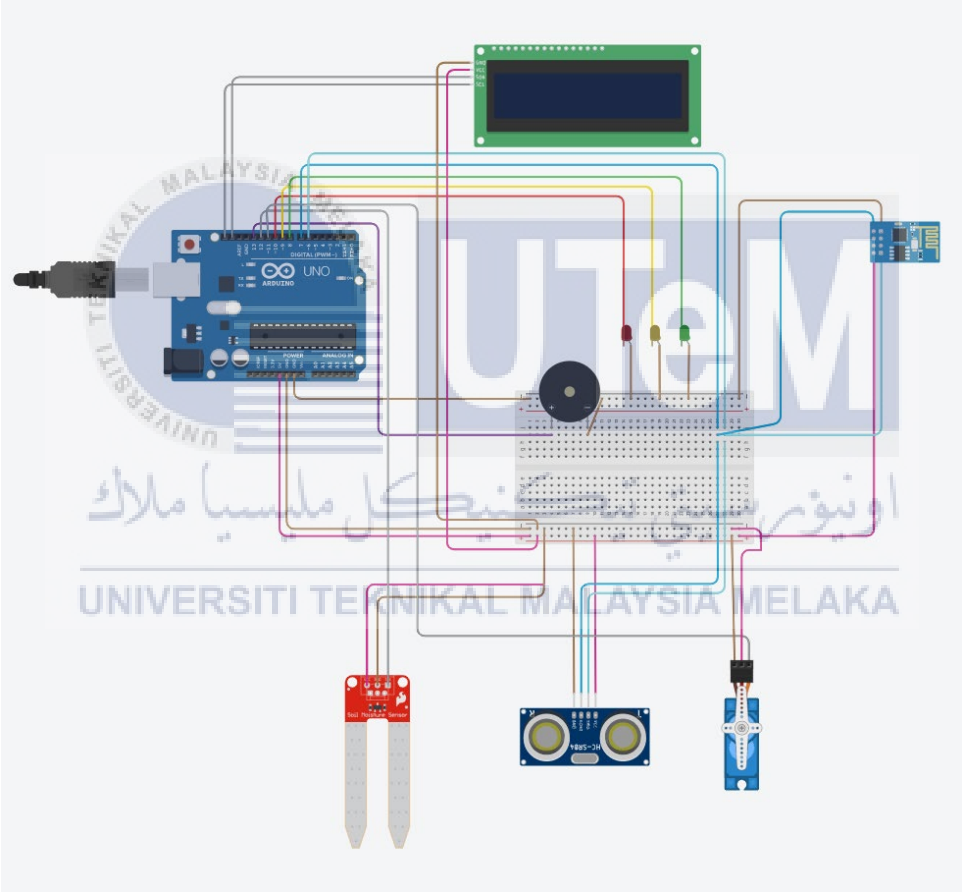


Figure 3.13 Electrical Diagram

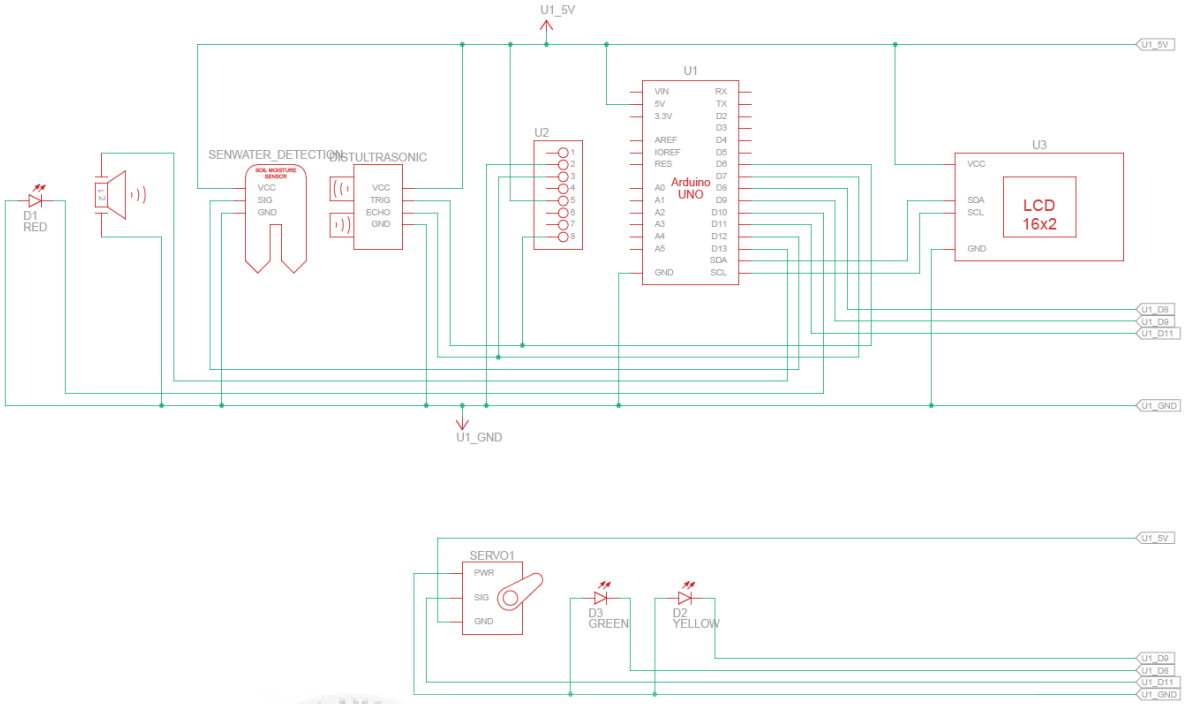


Figure 3.14 Circuit Design

3.6 Preliminary Result

The preliminary result is based on previous testing of the project during PSM 1 semester. It is the foundation of the finished product that will be made available. Based on these preliminary results, the draft coding that are uploaded to Arduino UNO work as intended on TinkerCAD. These codes are then tested on real hardware which consist of component that will be used on the final product result. These codes will also be neatly modified to be used to be used for NodeMCU module for its IoT platform usage.

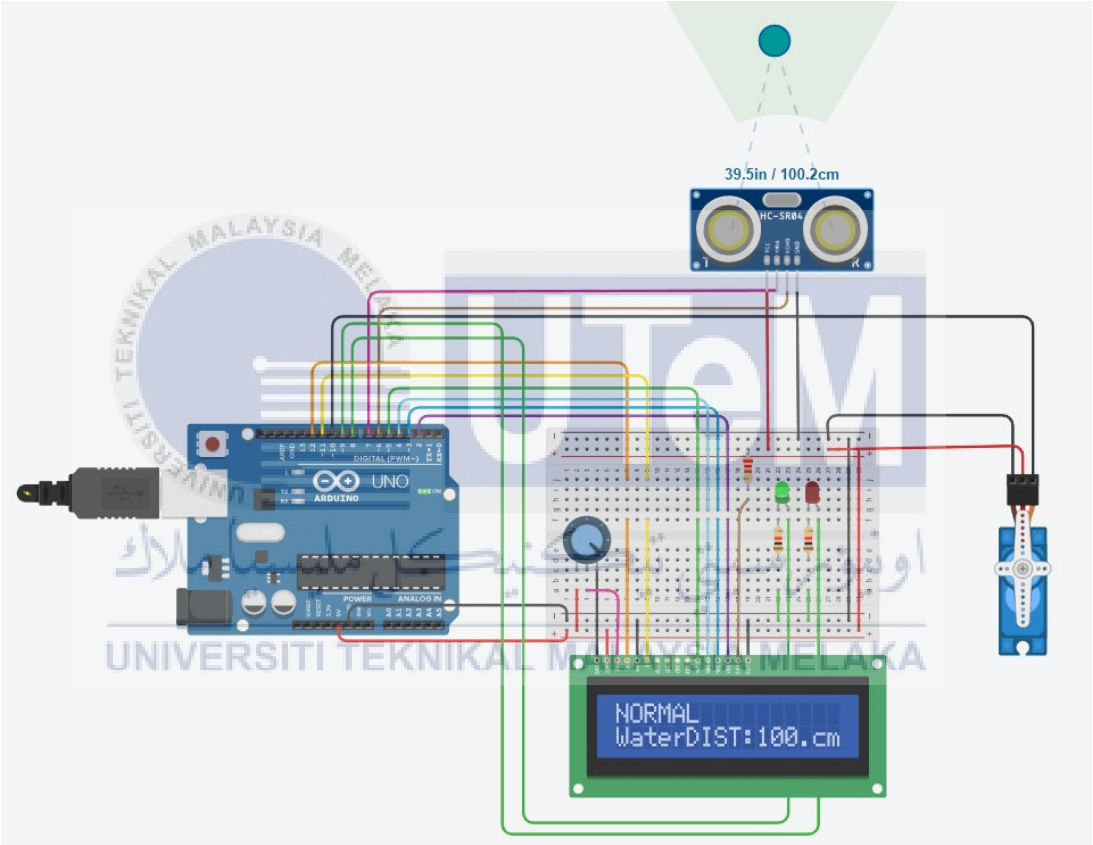


Figure 3.15 Proof of concept of the system on TinkerLab

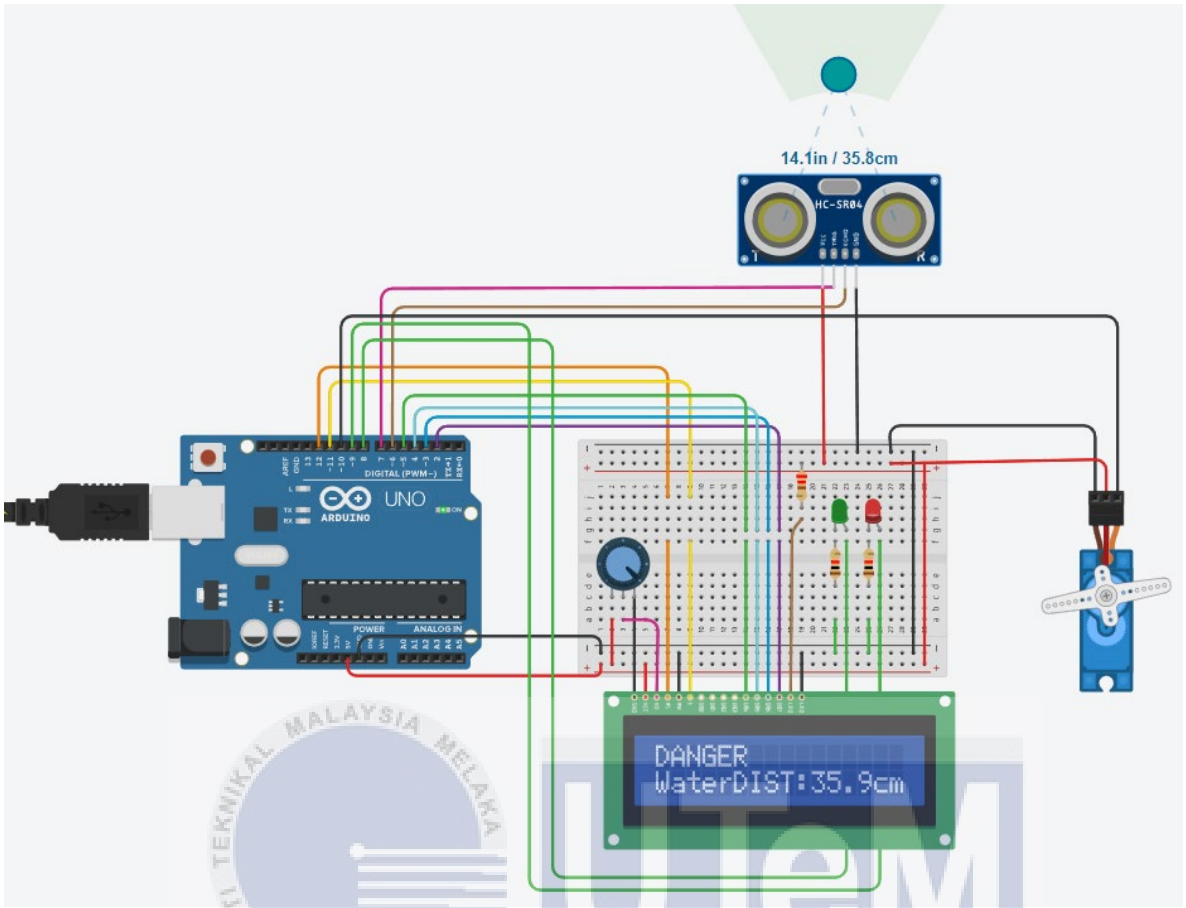


Figure 3.16 Proof of concept of the system on TinkerLab

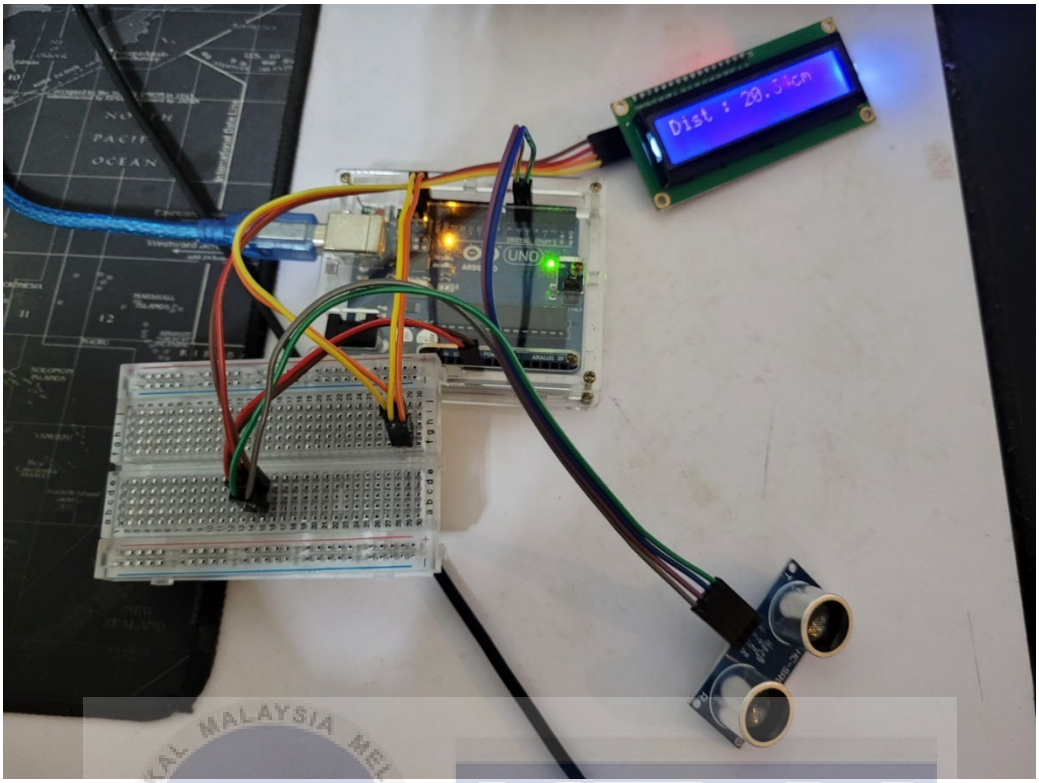


Figure 3.17 Testing pre-built connection of the project

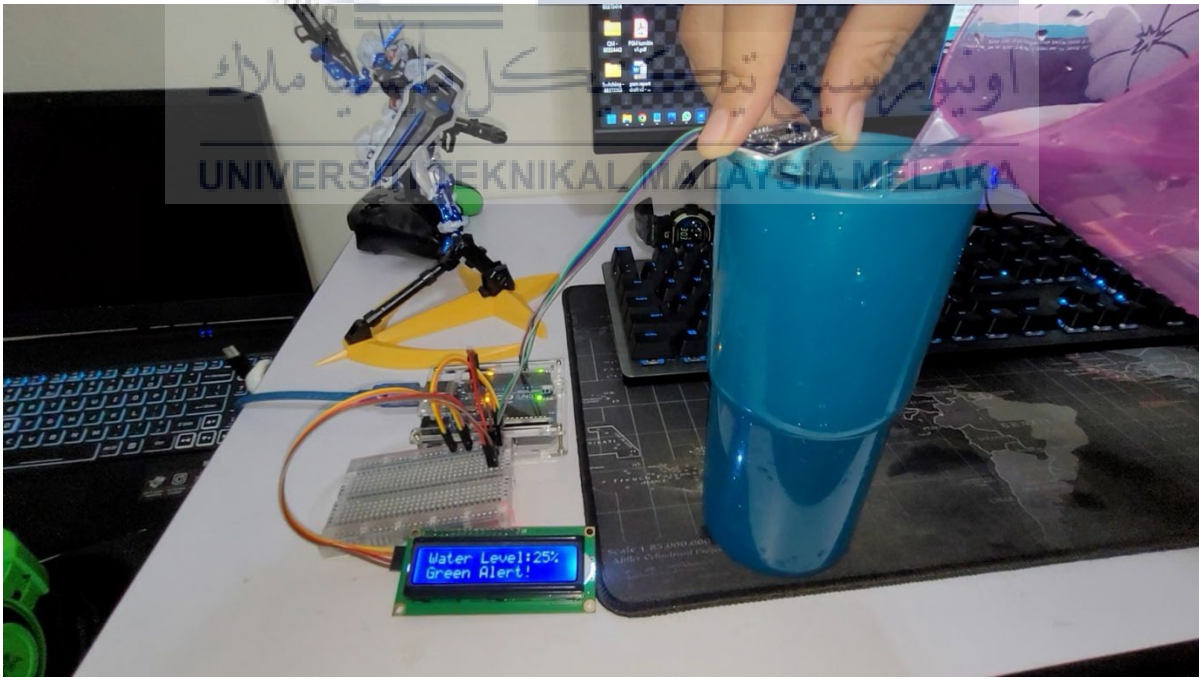


Figure 3.18 Testing the pre-built of the project

3.7 Summary

This chapter present the proposed methodology of the Early Home Flood Prevention System. Object tracking systems based on ultrasonics are created using methods and processes that have already been explored. The basic component hardware for this project is an Arduino Uno, an ultrasonic sensor, and a water gate. The project's block diagram and layout are designed, and a layout prototype is built based on the layout design. Furthermore, the flowcharts for the system development process are divided into two parts: hardware development and software development. Both of these things are intertwined. The proof of concept for the connection of the system have been tested on TinkerLab software which would work in theory. From the proof of concept, hardware design would be built and fitted to a small-scale prototype which be presented in the end. The coding for the system would be also modified and improved to reach the intended objective of the project.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter explains the results obtained from the project that was implemented. This chapter is important in order to prove whether all objectives of project that stated in chapter one is fulfil or not. As this project is working properly, the output and results can be created. The hardware implementation and software implementation for this project, which is Early Detection System For Home Flood Prevention Using Arduino and IoT will be clarified in this chapter. After the hardware have been assembled and software have been setup properly, it will be tested to see if the output would be as expected. The output from this project would be obtained to be used for result analysis.

4.2 Software Setup

In this section, the implementation or design of the software will be discussed. For this project, two-part of software side had be set up which are coding for the microcontrollers and set up for the IoT platform. For writing and uploading programmes to boards that are compatible with Arduino or other vendor development boards, utilise the Arduino integrated development environment (IDE). Most of the logic and instruction for the system is written within the coding. Coding written is shown Appendices section of this project report. 3 sets of states contain an instruction when water level reach certain capacity and its conditions are met. The main coding for this project is then uploaded to Arduino UNO.

For NodeMCU ESP8266, a slightly modification made for it to integrate with Blynk IoT platform. Using its Blynk.Console proprietary app, template device have been set up and line up with the hardware configuration of Ultrasonic sensor

connection to NodeMCU ESP8266. Blynk device IoT platform later set up using template device that have been made obtain its firmware configuration coding. This code would later be placed to NodeMCU ESP8266 coding which consist of Template ID, Device Name, and AuthToken to be declared at the very top of the firmware code. Figure 4.1 illustrates how the system is being programmed to enable communication with the IoT platform. An optional command to enable additional in-depth printouts is define BLYNK DEBUG, which ought to be the first line of code written. The use of define BLYNK PRINTS was done in order to suppress prints and conserve space. Since NodeMCU has SPI hardware on 4 pins, the purpose of the SPI library is to enable communication between SPI devices and Arduino. The ESP8266WiFi.h file must then be included in the system for this module to be connected to a Wi-Fi network and begin sending and receiving data. In order for the BLYNK application to start a new project, a request for an authentication token must be made. Next, a dashboard for User Interface are made within Blynk IoT mobile phone applications.

```
1   #define BLYNK_TEMPLATE_ID "
2   #define BLYNK_DEVICE_NAME "
3   #define BLYNK_AUTH_TOKEN "
4
5   #define TRIGGERPIN D1
6
7   #define ECHOPIN D2
8   // Comment this out to disable prints and save space
9   #define BLYNK_PRINT Serial
10
11
12  #include <ESP8266WiFi.h>
13  #include <BlynkSimpleEsp8266.h>
14  #include <Wire.h>
15
16  char auth[] = BLYNK_AUTH_TOKEN;
17
18  // Your WiFi credentials.
19  // Set password to "" for open networks.
20  char ssid[] = "
21  char pass[] = "
```

Figure 4.1 NodeMCU ESP8266 coding in Arduino IDE

4.3 Hardware Setup

In this section, the implementation or design of the hardware will be discussed. For this project, three-part of hardware had been set up which are the main system, simulated water drainage and water gate side. Figure 4.2 shows the main system. Figure 4.4 show the simulated water drainage system. In Figure 4.5 shows a small-scale model consist of residential house with water gate system. The main system which will connected to simulated water drainage system and water gate system as shown in Figure 4.6.

Sensors that are attached to the simulated water drainage system shown in Figure 4.3 will be send to both Arduino UNO and NodeMCU ESP8266 within the main system shown in Figure 4.2. Then, the data collected will be processed to show water level in simulated drainage. The data will be displayed to LCD display attached to main system shown in Figure 4.3. NodeMCU ESP8266 will send the data through Blynk IoT Platform to display the water level at mobile phone. LEDs and Buzzer at main system function as output indicators for users. The water gate placed in the small-scale model shown in Figure 4.4 is connected to Servo Motor which then connected to main system. The distance range for simulated water drainage system can be detected by Ultrasonic sensor is minimum distance 2cm to maximum distance 10cm from ultrasonic sensor. The system will not be getting data from the sensor if it distance range is over 10cm or below 2cm from Ultrasonic sensor. The maximum distance range for by Ultrasonic sensor can be change in the coding.

The small-scale model of residential house is to show how the system would work. Between the walls enclosing the residential area are placed water gates. The purpose of the water gate on the small-scale model is to show how the system would function if it were connected to the actual water gate's scale in the real world. The water gate would, in principle, prevent floodwater from entering the residential area in a real-world scenario when the project would be implemented. Simulated water drainage systems are used to replicate the house's outside drainage.

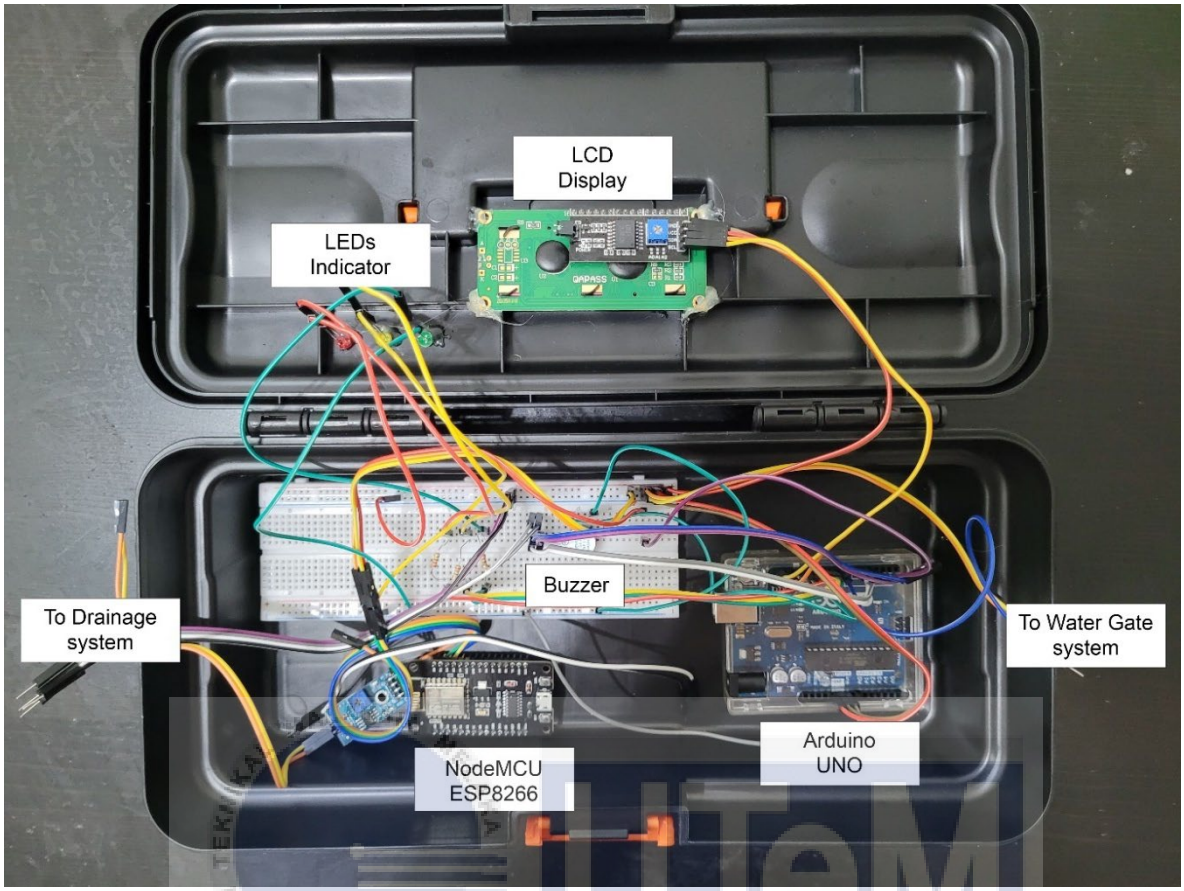


Figure 4.2 Main System Hardware Set-Up

Figure 4.3 Main System Hardware when cover closed

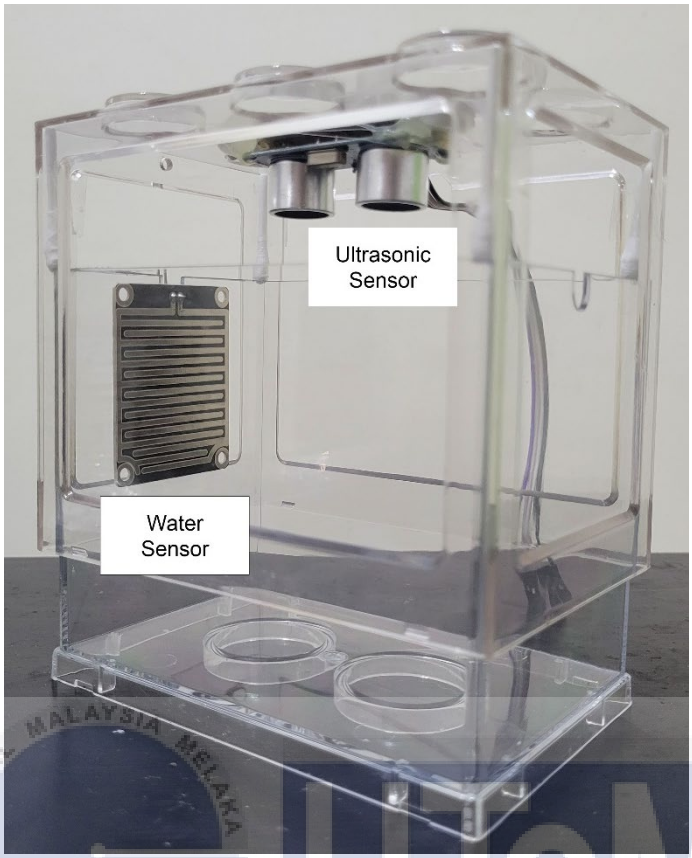


Figure 4.4 Simulated Water Drainage System Hardware Set-Up

Figure 4.5 Water Gate System Hardware Set-Up

Figure 4.6 Project Connection Hardware Set-Up

4.4 Result Analysis

In this section, the output obtained from the project are shown. This includes the output from Blynk IoT platform. The distance range for simulated water drainage system can be detected by Ultrasonic sensor is minimum distance 2cm to maximum distance 10cm from Ultrasonic sensor as shown in Figure 4.7. The system will not be getting data from the sensor if it distance range is over 10cm or below 2cm from Ultrasonic sensor. The maximum distance range for by Ultrasonic sensor can be change in the coding. First, Ultrasonic sensor will collect water level data from Simulated Water Drainage tank. Water sensor attached to the side of the tank act as false error sensor if the Ultrasonic sensor detect other foreign object that are not water or liquid. These data will be sent to both microcontrollers; the Arduino UNO and NodeMCU ESP8266. Microcontrollers will process these data to calculate the water level in percentage form. Servo motor attached to water gate that is placed on small-scale model will follow the instruction of Arduino UNO depending on the water level. In normal state, the water gate should be in lowered state. LCD display will be always showing the water level data received from Arduino UNO as shown in Figure 4.7. Blynk App on mobile phone will also showing the data received from NodeMCU ESP8266 through Blynk IoT platform as shown in Figure 4.8.

Within these microcontrollers have been set an instruction when water level reach certain capacity. If the water capacity is below 50% as shown in Figure 4.9, the system will act as normal with displays showing the water level, LED indicator attached to the main system will light up green. If the water capacity is between 50% to 70% as shown in Figure 4.10, “Orange Alert” will display, buzzer would be outputting tone and LED indicator will light up yellow. When water capacity reach above 70% as shown in Figure 4.11, a few instructions from microcontrollers will be initiated. On this state which is call “Danger Alert”, Arduino UNO will initiate command to servo motor so the water gate will rise. The LED indicator will light up red while buzzer outputting tone. NodeMCU ESP8266 will give notification through the Blynk App on mobile phone to notify user of the alert as shown in Figure 4.12. This state of the system will continue until Ultrasonic sensor have given a data to microcontroller to indicate that the water capacity dropped below the danger water level capacity.

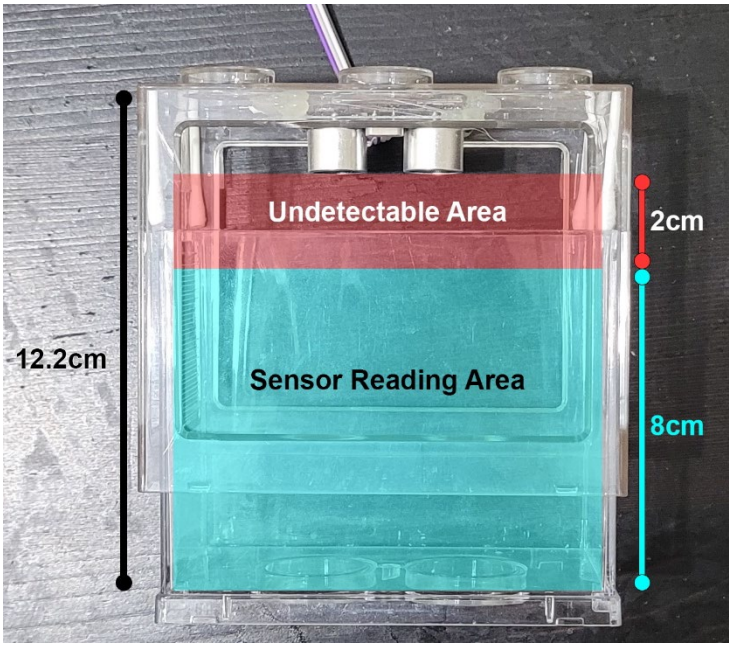


Figure 4.7 Sensor range for Simulated Water Drainage system

Ultrasonic sensor able to read within the sensor reading area (highlight in blue colour).

Figure 4.8 LCD display on the system

The flood monitoring system LCD display will output its current water level based on the data received from ultrasonic sensor then through processed by Arduino UNO.

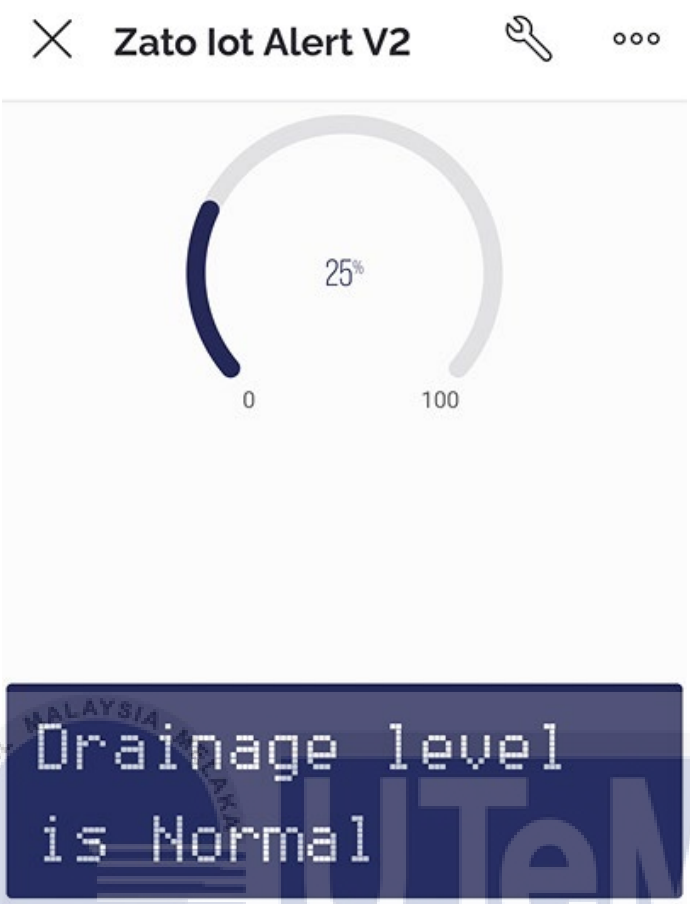


Figure 4.9 UI display on Blynk mobile phone application

This is the user interface mobile dashboard of the device from Blynk IoT mobile phone application. The gauge which range from 0% to 100% are virtual gauge display of the water level from simulated water drainage system. The virtual LCD display under the gauge are used to display the status of the simulated drainage system. NodeMCU ESP8266 output data that processed based on data received from ultrasonic sensor. The processed data then are displayed on this mobile dashboard.

Figure 4.10 System in Normal State

When the water level are detectible by ultrasonic sensor to be below the 49% of the simulated drainage water capacity, the flood monitoring system LCD display will output its current water level. LED indicator would light up green. Water gate are also in lowered state.

Figure 4.11 System in Orange State

When the water level are detectible by ultrasonic sensor to be in between at 50% to 69% of the simulated drainage water capacity, the flood monitoring system LCD display will output “Orange Alert” and its current water level. This is to indicate to user that water capacity almost reaches dangerous level. LED indicator would light up orange. Buzzer would start outputting tone that will repeat until the status of the water capacity change to lower than 50% However, water gate would still be in lowered state as the water capacity is not in dangerous level yet.

Figure 4.12 System in Alert State

When water level are detectible by ultrasonic sensor to be in between at 70% to 100% of the simulated drainage water capacity, the system check the water sensor attached to side of water capacity tank for liquid. This is for system to avoid false alarm detected from ultrasonic sensor. The flood monitoring system LCD display will output "Danger Alert" and its current water level. This is to indicate to user that water capacity reaches dangerous level. LED indicator would light up Red. Buzzer would be outputting tone that will repeat until the status of the water capacity change to lower than 50%. Water gate would be in raised state as the water capacity is in dangerous level yet.

Figure 4.13 Blynk Notification through the mobile phone application

In "Danger Alert" status; when water level are detectible by ultrasonic sensor to be in between at 70% to 100% of the simulated drainage water capacity, Blynk would notify the user through Blynk IoT platform to Blynk IoT mobile phone application. In the notification it notifies the water level is/has exceed danger limit of simulated drainage water capacity. This is the example of notification that would be shown to user.

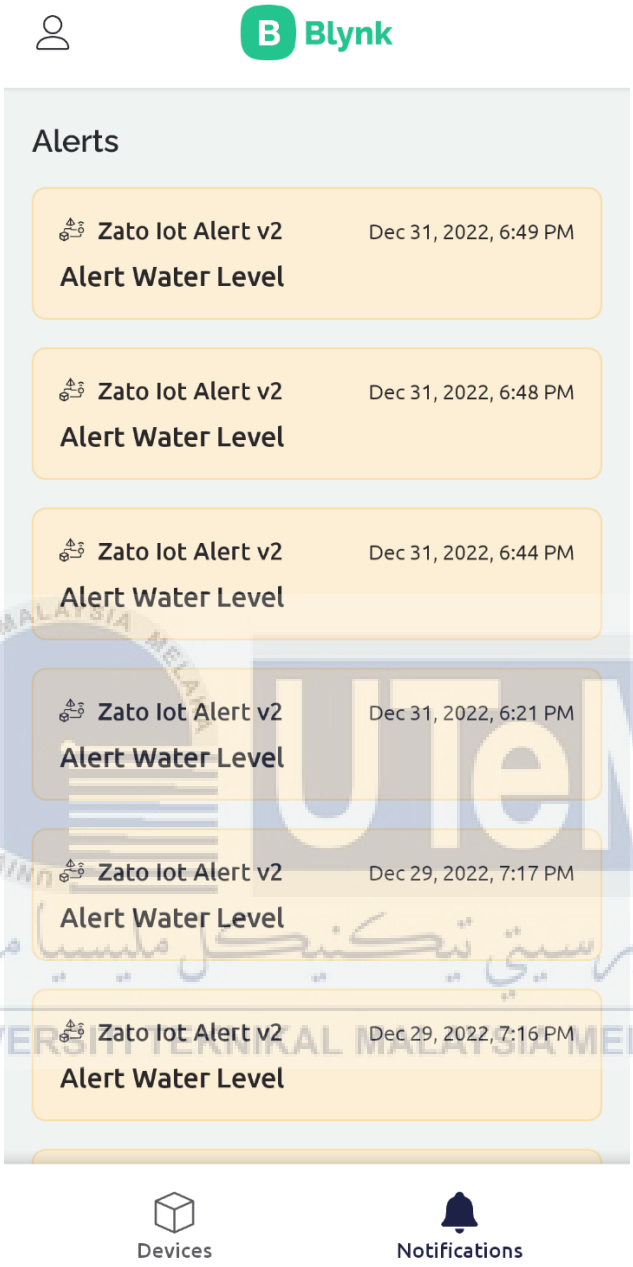


Figure 4.14 Blynk Notification through the mobile phone application

All the notification history and log that are received from Blynk IoT platform would display from Blynk IoT mobile phone application. It includes the date and time of when those notification are received.

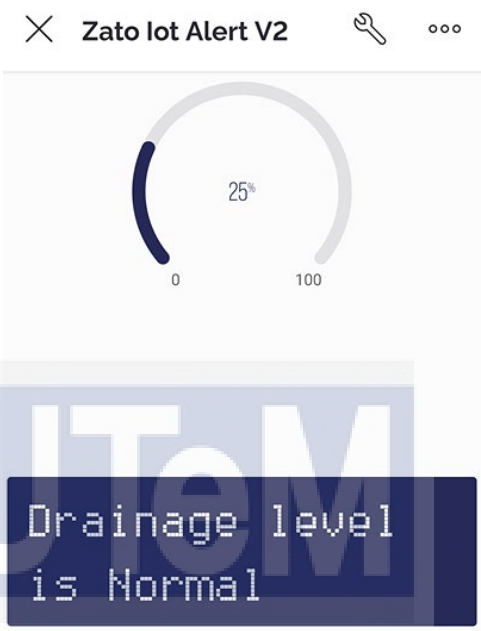
Water gate system are controlled by Flood Monitoring Main System. The water gate will be raised or lowered base on the water capacity of simulated water drainage system. Output for the water gate system are shown below: -

Table 4.1 Output displayed on Water Gate system

Description	Diagram
Water Capacity at 69% and below.	 **Figure 4.15 Water Gate Lowered**
Water Capacity at 70% and above.	 **Figure 4.16 Water Gate Raised**

NodeMCU ESP8266 output data that processed based on data received from ultrasonic sensor. This output data can be received and seen using Blynk IoT mobile phone application. Output displayed on Blynk IoT app in mobile phone are shown as below: -

Table 4.2 Output displayed on Blynk IoT application in mobile phone

Description	Diagram
Water Capacity at 49% and below.	 **Figure 4.17 Blynk Mobile Dashboard at 25%**
Water Capacity at 50% to 69%.	 **Figure 4.18 Blynk Mobile Dashboard at 50%**

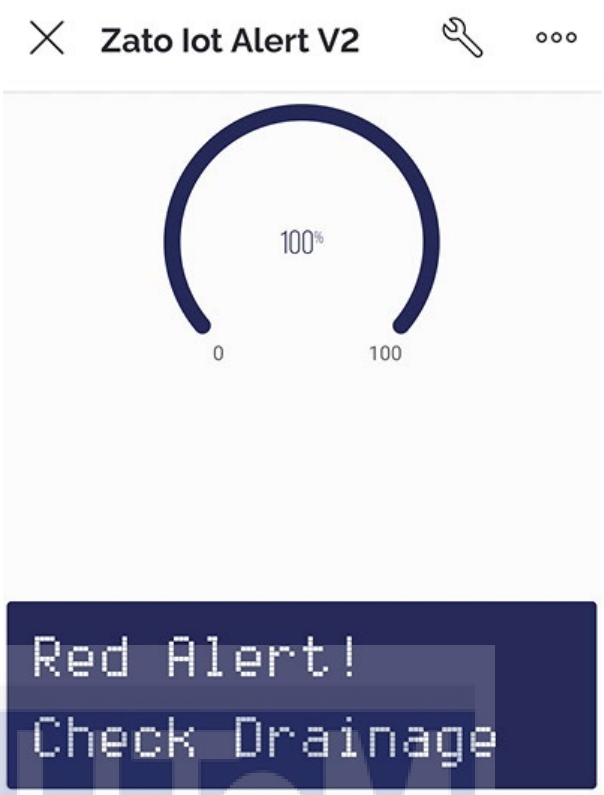
Water Capacity at 70% and above.	

Figure 4.19 Blynk Mobile Dashboard at 100%

When water level is detectible by ultrasonic sensor to be in between at 70% to 100% of the simulated drainage water capacity, Blynk would notify the user through Blynk IoT platform to Blynk IoT mobile phone application.

4.5 Summary

In this chapter, it can prove that the project hardware and software side are working as intended. The ultrasonic sensor is suitable for the drainage tank that shape is regular or symmetrical. Although, some of the data gathering from the ultrasonic sensor are unstable. It is probably due to obstruction during pouring water to the simulated water drainage system. The water capacity formula that used in the coding are probably to making the ultrasonic sensitivity not accurate as it should. Due to limitation of Blynk IoT platform free account, notification cannot be received more for accurate demonstration of the system. The small-scale model of residential house is to show how the system would work. Water gate are place between the wall surrounding the residential area. Water gate placed on the small-scale model is to represent how the system would interact if it's connected to real world scale of actual water gate. In real world situation on how the project would work, the water gate would block overflowing flood water from entering the residential area in theory. Simulated water drainage system are to simulate the drainage that would surround the house.

CHAPTER 5

CONCLUSION

5.1 Conclusion

Effective early flood warning systems can benefit the community and serve as a precautionary measure for victims, preventing death, disaster-related trauma, and property damage. The water gate system is an excellent complement to the existing excellent flood detecting system, and it has the ability to improve user safety while also reducing property damage. Also, IoT platform is an excellent alert communication strategy that can help flood victims in a specific place get the news out. It would be fantastic if this system had a data mining and artificial intelligence component that could help with water level data processing and generate huge graphs for better analysis. This finding will aid in the forecasting of future water levels. The findings enabled the project's objectives to be met, and the project architecture both hardware and software was completely functional. The data was sent to the mobile phone over Wi-Fi. All water levels, including those that are alert, moderate danger, and high risk, are detected by the ultrasonic sensor. Data from the input sensor and the Arduino Uno microcontroller were sent to the IoT platform system. Although, huge flaw of the system is ultrasonic sensor is unable to get a reading when the water is too close, approximately 2cm and nearer. The hardware component used in the system are not waterproof, making the whole system would not be able to work if it is submerged underwater.

The methodology is separated into three sections, which include testing, hardware development, and the three objectives outlined in Chapter 1. Ultrasonic sensor is proven to be effective for this project to sense water level in drainage. Arduino UNO flexibility are shown in this project for its available port making its viable for data input and system output. Due to its built-in Wi-Fi and open source IoT platform, NodeMCU is more ideal for IoT projects. Blynk was selected as the IoT platform since it is a free app for creating interfaces. Overall, the project, Early Detection System for Home Flood Prevention Using Arduino And IoT perform well and has succeeded.

5.2 Future Works

For further improvement of Early Detection System For Home Flood Prevention Using Arduino And IoT could be enhanced as follows:

1) In this project, the power supply aspect is not touch upon. An efficient portable can be implemented to the system such as solar powered battery.
2) Data collection such as in graph form can be made for water level. It can be used for estimated time of how long it takes until water level reach dangerous level.
3) Made the water gate can also be controlled manually to raised or lowered using IoT platforms.
4) Add other value parameter data to the system such as weather forecasts or temperature.
5) Made a small-scale model with functional drainage to show effectiveness usage of water gate.
6) Implement other type of connectivity such as GSM or Bluetooth for its IoT implementation.
7) Implement this system to a real world uses for testing the effectiveness of this project.

REFERENCES

[1] "Arduino." Wikipedia, Wikimedia Foundation, 21 February 2007, https://en.wikipedia.org/wiki/Arduino.

[2] Admin, "Arduino technology architecture and its applications," WatElectronics.com, 31-May-2022. [Online]. Available: https://www.watelectronics.com/arduino-technology-architecture-and-applications/. [Accessed: 02-Jun-2022].

[3] Noor Suraya Romali, Zulkifli Yusop; Flood damage and risk assessment for urban area in Malaysia. Hydrology Research 1 February 2021; 52 (1): 142–159. doi: https://doi.org/10.2166/nh.2020.121.

[4] "SMART Tunnel." Wikipedia, Wikimedia Foundation, 5 May 2006, https://en.wikipedia.org/wiki/SMART_Tunnel.

[5] "SMART Project," Department of Irrigation and Drainage, 21-Apr-2017. [Online]. Available: https://www.water.gov.my/index.php/pages/view/430. [Accessed: 02-Jun-2022].

[6] Y. A. Badamasi, "The working principle of an Arduino," 2014 11th International Conference on Electronics, Computer and Computation (ICECCO), 2014, pp. 1-4, doi: 10.1109/ICECCO.2014.6997578.

[7] W. H. Wan Hassan, A. Z. Jidin, S. A. C. Aziz, and N. Rahim, "Flood disaster indicator of water level monitoring system," International Journal of Electrical and Computer Engineering, vol. 9, no. 3, pp. 1694–1699, Jun. 2019, doi: 10.11591/ijece.v9i3.pp1694-1699.

[8] O. Intharasombat and P. Khoenkaw, "A Low-Cost Flash Flood Monitoring System," 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE), 2015.

[9] B. Sharma, S. Azid, K. Raghuwaiya, A. Chand, S. Prasad, and A. Jacquier, "SMS BASED FLOOD MONITORING AND EARLY WARNING SYSTEM," ARPN Journal of Engineering and Applied Sciences, vol. 10, no. 15, 2015, [Online]. Available: www.arpnjournals.com

[10] J. G. Natividad and J. M. Mendez, "Flood Monitoring and Early Warning System Using Ultrasonic Sensor," IOP Conference Series: Materials Science and Engineering, vol. 325, no. 1, Mar. 2018, doi: 10.1088/1757-899X/325/1/012020.

[11] I. Riyanto et al., "WEB CAMERA SENSOR COUPLED WITH LIDAR DATA FLOOD MAP FOR FLOOD WARNING SYSTEM," IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019.

[12] L. Deng, "Study on Ultrasonic Ranging System Design," 2009 Fifth International Conference on Information Assurance and Security, 2009, pp. 414-416, doi: 10.1109/IAS.2009.352.

[13] B. Kotiyal, I. (12E. T. Baig, M. Muzamil, and S. (12E. T. Dalvi, "Home Automation using Arduino WIFI module ESP8266," Apache2 Ubuntu Default Page: It works, 01-May-2016. [Online]. Available: http://ir.aiktclibrary.org:8080/xmlui/handle/123456789/1558. [Accessed: 6-Jun-2022].

[14] "BEST PRACTICES: Recommendations for Repositories Fourth Edition." Accessed: Jun. 6, 2022. [Online]. Available: http://www.ihmt.unl.pt/wp-content/uploads/2021/03/ISBER.pdf

[15] Wikipedia Contributors, "Uninterruptible power supply," Wikipedia, May 29, 2022. https://en.wikipedia.org/wiki/Uninterruptible_power_supply (accessed Jun. 6, 2022).

[16] "Backup UPS Systems UT800EG," Backup UPS Systems UT800EG, 2019. https://www.cyberpower.com/my/en/product/sku/ut800eg (accessed Jun. 6, 2022).

[17] "Arduino Uno Specification," TOMSON ELECTRONICS, 2018. https://www.tomsonelectronics.com/blogs/news/arduino-uno-specification (accessed Jun. 10, 2022).

[18] B. Anbarasu and G. Anitha, "Altitude Estimation based on Multi-Sensor Fusion for Indoor Navigation of UAV," Journal of Basic and Applied Engineering Research Print, vol. 2, no. 7, pp. 540–543, May 2015.

[19] "Arduino Ultrasonic Sensor Overview and Tutorials - Latest Open Tech From Seeed," *Latest Open Tech From Seeed*, Nov. 04, 2019. https://www.seeedstudio.com/blog/2019/11/04/hc-sr04-features-arduino-raspberrypi-guide/ (accessed Jun. 6, 2022).

[20] Last Minute Engineers, "How Water Level Sensor Works and Interface it with Arduino," *Last Minute Engineers*, Nov. 29, 2019. https://lastminuteengineers.com/water-level-sensor-arduino-tutorial/ (accessed Jun. 10, 2022).

APPENDICES

Appendix A: Project Gantt Chart Bachelor Degree Project 1

PROJECT ACTIVITIES	STATUS	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15	WEEK 16
BDP Briefing	E									M							S
	A																
Meeting with Supervisor	E									I							T
	A																
Distribution of project titles	E									D							U
	A																
PSM 1 Rubrics Explanation	E									B							D
	A																
Project planning	E									R							Y
	A																
Proposal preparation	E									E							
	A																
Abstract	E									A							W
	A																
Literature Review	E									K							E
	A																
Design Project	E																E
	A																
Flowchart	E									S							K
	A									E							
Construct the project	E									M							
	A																

E – Estimated, A – Actual

- All project activities that have been estimated are in line with the actual timeline plan.

Appendix B: Project Gantt Chart Bachelor Degree Project 2

PROJECT ACTIVITIES	STATUS	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15	WEEK 16
Draft Material List	E									M							S
	A																
Meeting with Supervisor	E									I							T
	A																
Test Hardware	E									D							U
	A																
Analyse Result	E									B							D
	A																
Complete Chapter 4:Result and Discussion	E									R							Y
	A																
Complete Chapter 5:Conclusion	E									E							
	A																
Submit draft report	E									A							W
	A																
Prepare Project Poster	E									K							E
	A																
Preparation for presentation	E																E
	A																
Presentation	E									S							K
	A									E							
Submit Final Report	E									M							
	A																

E – Estimated, A – Actual - All project activities that have been estimated are in line with the actual timeline plan.

Appendix C: Arduino UNO coding

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <Servo.h>

LiquidCrystal_I2C lcd(0x27,16,2);

const int in=6;  //echo
const int out=5;   //trig
const int green=10;
const int orange=11;
const int red=12;
const int buzz=13;
const int water=4;
Servo servo_8;

void setup()
{
Serial.begin(9600);
lcd.begin();
lcd.backlight();
pinMode(water, INPUT);
pinMode(in, INPUT);
pinMode(out, OUTPUT);
pinMode(green, OUTPUT);
pinMode(orange, OUTPUT);
pinMode(red, OUTPUT);
pinMode(buzz, OUTPUT);
digitalWrite(green,LOW);
digitalWrite(orange,LOW);
digitalWrite(red,LOW);
digitalWrite(buzz,LOW);
servo_8.attach(8);
servo_8.write(10);
lcd.setCursor(0,0);
lcd.print("Flood Monitoring");
lcd.setCursor(0,1);
lcd.print("Alerting System");
delay(5000);
lcd.clear();
```

```
}

void loop()
{
long dur;
long dist;
long per;
digitalWrite(out,LOW);
delayMicroseconds(2);
digitalWrite(out,HIGH);
delayMicroseconds(10);
digitalWrite(out,LOW);
dur=pulseIn(in,HIGH);
dist=(dur*0.034)/2;
per=map(dist,10.5,2,0,100);
if(per<0)
{
  per=0;
}
if(per>100)
{
  per=100;
}


Serial.println(String(per));
lcd.setCursor(0,0);
lcd.print("Water Level:");
lcd.print(String(per));
lcd.print("%  ");

if (per>=90 && digitalRead(water) == LOW)
{
  digitalWrite(red,HIGH);
  digitalWrite(green,LOW);
  digitalWrite(orange,LOW);
  servo_8.write(65);
  lcd.setCursor(0,1);
  lcd.print("DANGER ALERT");
  SendBuzz();
  delay(500);
```

```
}

else if(per>=70 && digitalRead(water) == LOW)
{
  lcd.setCursor(0,1);
  lcd.print("Red Alert!   ");
  digitalWrite(red,HIGH);
  digitalWrite(green,LOW);
  digitalWrite(orange,LOW);
  servo_8.write(65);
  tone(buzz,1000);
  delay(500);
  noTone(buzz);
  delay(500);
}


else if(per>=50 && digitalRead(water) == LOW)
{
  lcd.setCursor(0,1);
  lcd.print("Orange Alert!  ");
  digitalWrite(orange,HIGH);
  digitalWrite(red,LOW);
  digitalWrite(green,LOW);
  tone(buzz,1000);
  delay(500);
  noTone(buzz);
  delay(1000);
  }

else
{
 lcd.setCursor(0,1);
 lcd.print("Green Alert!  ");
 digitalWrite(green,HIGH);
 digitalWrite(orange,LOW);
 digitalWrite(red,LOW);
 digitalWrite(buzz,LOW);
 servo_8.write(10);
}

delay(1000);
}
```

```
void SendBuzz()
{
  for(int i = 0; i <= 2; i++)
  {  tone(buzz,1000);
  delay(500);
  noTone(buzz);
  delay(500);}
}
```

Appendix D: NodeMCU ESP8266 coding

```
#define BLYNK_TEMPLATE_ID "xxxxxxxxxxxxxxxxxxx"
#define BLYNK_DEVICE_NAME "Zato Iot Alert"
#define BLYNK_AUTH_TOKEN "xxxxxxxxxxxxxxxxxxxx"

#define TRIGGERPIN D1

#define ECHOPIN D2
// Comment this out to disable prints and save space
#define BLYNK_PRINT Serial


#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Wire.h>

char auth[] = BLYNK_AUTH_TOKEN;

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "WIFI SSID";
char pass[] = "WIFI PASSWORD";

bool alert_wifi = false;

WidgetLCD lcd(V1);

void setup()
{
  // Debug console
  Serial.begin(115200);
pinMode(TRIGGERPIN, OUTPUT);
  pinMode(ECHOPIN, INPUT);
  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  //Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
  lcd.clear(); //Use it to clear the LCD Widget
}

void loop()
```

```
{
long dur;
long dist;
long per;
digitalWrite(TRIGGERPIN,LOW);
delayMicroseconds(2);
digitalWrite(TRIGGERPIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGERPIN,LOW);
dur=pulseIn(ECHOPIN,HIGH);
dist=(dur*0.034)/2;
per=map(dist,10.5,2,0,100);
if(per<0)
{
  per=0;
}
if(per>100)
{
  per=100;
}

Serial.println(String(per));
Blynk.virtualWrite(V0, per);

if(per>=70 && alert_wifi == false)
{
lcd.print(0,0, "Red Alert!      ");
lcd.print(0,1, "Check Drainage    ");
Blynk.logEvent("alert_wifi", "Water Level Exceed Danger Limit");
alert_wifi == true;
}

else if(per>=50)
{
lcd.print(0,0, "Orange Alert!     ");
lcd.print(0,1, "Check Drainage    ");
alert_wifi == false;
}

else
{
  lcd.print(0,0, "Drainage level  ");
  lcd.print(0,1, "is Normal       ");
```

```
  alert_wifi == false;
}

  Blynk.run();

}
```