IOT - BASED WATER QUALITY MONITORING SYSTEM FOR FISHERY

MOHD SYAHMI BIN SALIM

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

IoT - BASED WATER QUALITY MONITORING SYSTEM FOR FISHERY

MOHD SYAHMI BIN SALIM

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Computer Networking) with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2021

DECLARATION

I hereby declare that this project report entitled

IoT - BASED WATER QUALITY MONITORING SYSTEM FOR FISHERY

is written by me and is my own effort and that no part has been plagiarized

without citations.

STUDENT : ________________________ Date : __9/9/2021____

(MOHD SYAHMI BIN SALIM)

I hereby declare that I have read this project report and found

this project report is sufficient in term of the scope and quality for the award of

Bachelor of Computer Science (Computer Networking) with Honours.

SUPERVISOR : ________________________ Date : __9/9/2021___

(DR. WAHIDAH BINTI MD SHAH)

DEDICATION

Thanks to Allah S.W.T,

for giving me a strength to accomplish this project.

To my supervisor,

Dr. Wahidah Binti Md Shah,

Thank you for being supportive and give non-stop guidance throughout this journey to complete this project.

To my wonderful parents,

Salim Bin Zakaria and Jamaliah Binti Abdul Razak,

Thank you for your prayers and moral support.

To my friends,

I would like to thank you for your constantly advice and support.

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Merciful, the Most Compassionate all praise be to Allah, the Lord of the universes and prayers and peace be upon Muhammad His servant and messenger. I am grateful to numerous members and individuals who have assisted me in the completion of this project.

I would like to thank my supervisor, Dr. Wahidah Binti Md Shah, who has always led and provided guidance throughout this project. The project would not have been completed without her, and the progress of the project would not have been known.

Special thanks to my family, Salim Bin Zakaria, Jamaliah Binti Abdul Razak, and all my siblings for their moral support while studying at UTeM to complete this project. I had many problems and mistakes in making this project a great project, which made me inactive. With their presence, they have boosted my enthusiasm by giving me endless inspiration to make me emotionally stronger to complete my project.

Finally, I would like to express my appreciation to my friends who are always willing to help me by sharing their incredible ideas. This project would not have been completed on time without them.

ABSTRACT

Water plays a significant role include hydration, agriculture, electricity, and aquaculture use. Water quality is an important aspect of fish farming production. Poor water quality led to slow fish growth and result in fish death. Fishpond water quality monitoring could be hassle and time consuming which requires breeders to obtain their own water samples and test it using special meters. The IoT-based system is proposed to assist fish breeders in monitoring the fishponds water and alert them about poor water quality conditions. The system consists of Raspberry Pi, pH sensor, and temperature sensor. A pH sensor is used to detect the pH of the water while a temperature sensor is used to detect the water temperature of fishponds, Raspberry Pi is a microprocessor that processes these water parameter readings. This system uses Wi-Fi for the data transmission process as it uses the built-in Wi-Fi module on the Raspberry Pi, fish breeders can perform monitoring via mobile apps on their android smartphones. If the pH and temperature values exceed the set action levels, fish breeders will get alert notifications from mobile apps. This will help breeders to carry out the process of maintaining the fishpond. The results are collected through testing, and the system's efficacy judgments are drawn from the findings for which sensor readings could be performed with a moderate degree and notification alerts functioned with excellent response.

ABSTRAK

Air memainkan peranan penting termasuk hidrasi, pertanian, elektrik, dan penggunaan akuakultur. Kualiti air adalah aspek penting dalam pengeluaran ternakan ikan. Kualiti air yang buruk menyebabkan pertumbuhan ikan lambat dan mengakibatkan kematian ikan. Pemantauan kualiti air kolam ikan dapat menjadi kesulitan dan memakan masa yang memerlukan penternak untuk mendapatkan sampel air mereka sendiri dan mengujinya dengan menggunakan meter khas. Sistem berasaskan IoT dicadangkan untuk membantu penternak ikan dalam memantau air kolam ikan dan memberi tahu mereka tentang keadaan kualiti air yang buruk. Sistem ini terdiri dari Raspberry Pi, sensor pH, dan sensor Suhu. Sensor pH digunakan untuk mengesan pH air sementara sensor suhu digunakan untuk mengesan suhu air kolam ikan, Raspberry Pi adalah mikropemproses yang memproses pembacaan parameter air ini. Sistem ini menggunakan Wi-Fi untuk proses penghantaran data kerana menggunakan modul Wi-Fi bawaan pada Raspberry Pi, penternak ikan dapat melakukan pemantauan melalui aplikasi mudah alih di telefon pintar android mereka. Sekiranya nilai pH dan suhu melebihi tahap tindakan yang ditetapkan, penternak ikan akan mendapat pemberitahuan amaran dari aplikasi mudah alih. Ini akan membantu penternak menjalankan proses pemeliharaan kolam ikan. Hasilnya dikumpulkan melalui ujian, dan penilaian keberkesanan sistem diambil dari penemuan yang mana bacaan sensor dapat dilakukan dengan tahap sederhana dan amaran pemberitahuan berfungsi dengan respons yang sangat baik.

TABLE OF CONTENTS

LIST OF TABLES

PAGE

LIST OF FIGURES

PAGE

LIST OF ABBREVIATIONS

FYP	**-**	**Final Year Project**
IoT	**-**	**Internet of Things**
RPI	**-**	**Raspberry Pi**
GPIO	**-**	**General-Purpose Input/Output**
RAM	**-**	**Random Access Memory**
Wi-Fi	**-**	**Wireless Fidelity**
MATLAB	**-**	**Matrix Laboratory**
API	**-**	**Application Program Interface**
APK	**-**	**Android Package**
DIN	**-**	**Data In**
DOUT	**-**	**Data Out**
VREF	**-**	**Voltage Reference**
VDD	**-**	**Voltage Drain-to-Drain**
VCC	**-**	**Voltage Common Collector**

LIST OF ATTACHMENTS

PAGE

CHAPTER 1: INTRODUCTION

1.1 Introduction

One of the most crucial aspects of a healthy environment is water quality. Clean water is essential for the survival of a wide range of plants and animals. The quality of the water used in fish farming is crucial. Poor water quality is a low water quality or standard that it is in bad condition. The status of various water parameters like pH and temperature cannot be overlooked for maintaining a healthy aquatic environment (Bhatnagar and Devi, 2013). The neglection of water quality can lead to fish death, this happens to fish breeders where water pollution is mostly caused by waste generated from agricultural activities. This is because, the main water source is from a nearby river that has been polluted (Astro Awani, 2019).

To determine the water quality, the water manually tested using complete equipped to test the water before it can be used for fish farming purposes. Nowadays, most fish breeders, especially rural breeders, use manual methods in monitoring the water of their fishponds. This water quality monitoring is done by using special equipment that requires the breeders themselves to go down to the location of the pond to obtain water samples for testing. Once the results were obtained, all records were kept in a book by hands for future reference. This monitoring process is done periodically either once a week or once every two weeks. This poses difficulties to breeders, additionally, this water analysis process requires some time to obtain water quality results.

In the agricultural sector perspective, the use of sensors is used for water monitoring processes. A system developed for measuring parameters water such as turbidity, solute, pH and temperature. The developed system is equipped with suitable sensors for measuring water parameters. Therefore, Water Quality Monitoring System is designed to monitor the condition of the water quality in the fishponds based on the water parameters (R. Verma and D. Kiran, 2019). Just like other livestock sectors, fish also needs good monitoring of water parameters especially tilapia which is the focus in this project for better fish growth and production. By leveraging the use of sensors, the system will detect the pH level and temperature of the water in the fishponds as a benchmark for the water parameters. This system will alert the fish breeder if the water parameters exceed the action level. Values that exceed this level of action are expected to have negative effects on fish survival and quality. Therefore, with the use of this system, breeders can anticipate and perform water maintenance on the fishpond more efficiently.

1.2 Problem Statement (PS)

In fish farming, it is difficult for breeders to monitor water for the purpose of water maintenance process. The need for fully equipped tools and a lengthy process to test water causes breeders to have time constraints with other matters in managing their premises. Easiest way could be by manually look at the color or turbidity of the water. However, this method is not accurate since other parameters such as pH and temperature cannot directly observable. This can endanger the fishpond environment and interfere with fish growth. Despite the color of the water is clear, does not guarantee good water quality in the fishpond. Thus, manual tools are used for measuring, for example water samples obtained from the ponds by the breeders themselves which are then evaluated manually using special meters for pH and temperature readings. This also pose problem and hard time to fish breeders in monitoring their fishpond. To summarize this, the problem statements for this project are shown in Table 1.1.

Table 1.1: Summary of Problem Statement

PS	Problem Statement
PS_1	Hard to monitor the quality of water using the manual method
PS_2	No indicator for water maintenance and only based on the physical change of water color
PS_3	Lack of notification to identify the poor water quality

1.3 Project Question (PQ)

Project questions were used to identify questions on monitoring water quality conditions. Based on several studies, it can be concluded that there are some difficulties in determining whether water quality reaches the required level for fish farming purposes. Table 1.2 shows the summary of the project question.

Table 1.2: Summary of Project Question

PS	PQ	Project Question
PS_1	PQ_1	How to monitor water quality for tilapia farming more effectively?
PS_2	PQ_2	How to determine water quality conditions based on water parameters?
PS_3	PQ_3	How to warn fish breeder if water quality exceeds the level of action?

1.4 Project Objective (PO)

Project objectives are tailored to the problem statement and project questions. The objectives of this Water Quality Monitoring System project are shown in the Table 1.3 below.

Table 1.3: Summary of Project Objective

PS	PQ	PO	Project Objective
PS_1, PS_2, PS_3	PQ_1, PQ_2, PQ_3	PO_1	To identify water quality parameters for tilapia fish farming.
		PO_2	To design and develop IoT based monitoring system that able to monitor and alert water quality.
		PO_3	To test the effectiveness of the system functionality, response time, and accuracy.

1.5 Project Scope

1. Focus on the use of fish breeders in maintaining the water quality of their fishponds. Fish breeders can view water quality indicators through their mobile phone.
2. A smart IoT project that focuses on the development of features that can be offered through the system by detecting water parameters in terms of pH level and water temperature using sensors. Initially, the dissolved oxygen water parameter also includes as part of the project plan but cannot be done due to finance constraint. Alerting system that can notify the mobile phone user when the parameters values exceed the action level.

1.6 Project Contribution (PC)

Project contribution defines the expected outcome from this project. This project may help many fish breeder especially modest breeder by providing the monitoring system for their fishpond water quality maintenance. The main goal of this project is to monitor the water quality for fish farming purposes.

The utilization of IoT with mobile application technology eases the project development in monitoring and alerting to the mobile phone. Detecting the parameters of the water helps to increase the water quality that meets the requirement for the use of fish farming. Table 1.4 below shows the summary of project contribution.

Table 1.4: Summary of Project Contribution

PS	**PQ**	**PO**	**PC**	**Project Contribution**
PS_1, PS_2, PS_3	PQ_1, PQ_2, PQ_3	PO_1	PC_1	Provide a solution in detecting the parameter of the water quality.
		PO_2	PC_2	Provide mobile application that can monitor and alert.
		PO_3	PC_3	Provide an effective system for consumerism.

1.7 Report Organization

Chapter 1: Introduction

This chapter discuss about the purpose in developing the Water Quality Monitoring System which includes project background, problem statement, project question, and project objective to clarify the intention of the system.

Chapter 2: Literature Review

This chapter discuss about other topics that have similar fields and can correlated to this project. Previous work or related work that using different tools and methods been compared. In this chapter will make changes to the existing work and identify the improvements needed in this project.

Chapter 3: Project Methodology

This chapter consists of a preview to the project methodology and methods that can be done on this project. This project uses Agile model and every stage of this methodology will be described in this chapter. The project milestones will be planned for proper project planning to make sure every task and phases are being through smoothly.

Chapter 4: Analysis and Design

This chapter will cover all the requirement that need to be analyzed for the project which include the Data Requirement, Functional Requirement, and Non-functional requirement. The results of the analysis and detailed design will be described. This chapter can be considered important where it is necessary to formulate a good analysis and design in order to produce a successful project.

Chapter 5: Implementation

In this chapter contain the implementation of the project design that has been determined previously. The system development begins using the required software and hardware. The implementation status and progress for each of the component or module been described.

Chapter 6: Testing

Through this chapter, testing phase and testing strategy that been adopted in the project are described. Testing will be made according to the testing strategy which include the Test Plan, Test Strategy, Test Design, and Test Results & Analysis. This allows the developed project system to be analyzed for any weaknesses and improvements that need to be made.

Chapter 7: Project Conclusion

This chapter explains the conclusions of this project on how the objectives are achieved based on the implementation and the testing phase. Significant results obtained in this project are summarized including the weaknesses and strengths of this project.

1.8 Conclusion

In conclusion, this project helps to improve water quality for the purpose of fish farming. In this chapter, all objectives, problem statements, project questions, project scope, and project contributions can be identified in detail. Water Quality Monitoring System can solve the problems faced by fish breeders in monitoring the use of their water quality in fishponds. This project can provide an overview of how the sensor method can detect the parameters of the water to be captured and transferred to the mobile phone. When the project is complete and ready to use in real situations, it can be useful and provide effective solutions to solve problems faced by fish breeders.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter discuss about the related works that relevant to the specific area of the research chosen. The literature will briefly describe in detail the significance of monitoring the water quality especially for the purpose of the fish farming by referring some certified sources. This provides a critical summary for the research -related topics published. Developing a Water Quality Monitoring System will be the focus of this chapter. All explanation and information about the sensors and applications is explained.

2.2 Related Work

Related work explained about the domain that related to the water monitoring system. The purpose of this related work is to identify and analyze the domain that can enhance the knowledge and understanding in aiding the developing process.

2.2.1 Water Quality Requirements of Tilapia

Water quality plays an important role in determining the production of healthy fish products. Fish live depend entirely on the water in which they live for all their needs. If the water quality is in accordance with the needs of the fish, then it will be more beneficial for the fish to get the maximum weight and size. The pH level essentially determines the acidic or basic nature of the water. Figure 2.1 shows the perception from the fish farming community, most of the fish breeders agree and aware about the water qualities in term of pH, Dissolved oxygen, CO_2, and Soluble salts can influence the good or bad of the water.

However, based on statistics shows that only a few of the respondents practice the adoption of water quality maintenance and some of them do not apply it. This may be closely related to the lack of knowledge and information, time constraints, and high cost of getting a fully equipped tools which implies the effort in adopting the water quality.

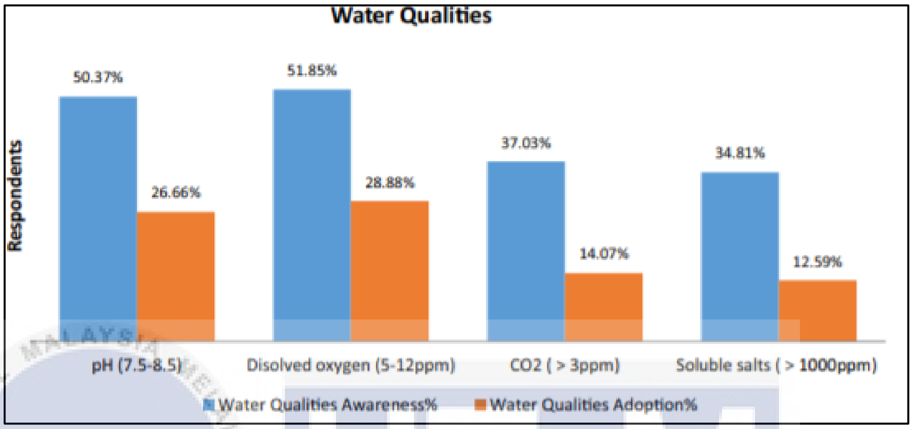


Figure 2.1: Water Perceptions from Fish Farming Community

Different fish species have different specific aspects of water quality. Tilapia is no exception in this regard, aspects of water quality including pH and temperature determine where they can survive, grow, and reproduce. Tilapia usually stops feeding when the water drops below 17 ◦C and death occurs below 11 ◦C (Atwood et al., 2003; Popma and Masser, 1999). The optimal water temperature for tilapia growth is between 29 and 31 ◦C (Popma and Masser, 1999). The temperature outside this level of comfort reduces appetite, growth, increased stress, and incidence of disease to tilapia. Higher temperature increases the rate of bio-chemical activity of the micro biota, plant respiratory rate, and so increase in oxygen demand (Bhatnagar, A., & Devi, P., 2013).

Regarding pH, readings below 4 can result in low survival rates to tilapia, increased mucus secretion, irritation and swelling of the gills can occur. The pH of water should be maintained between 4 and 9. Below 4 and above 10.5 can indicate a significant mortality rate.

Therefore, fish breeders must always be sensitive to changes that occur in the water. Based on figure 2.2 shows an indication of the level of action that needs to be given attention by breeders. Once the value exceeds the value of the action level, the breeder must take action to make the water conservation. Values exceeding these levels of action are expected to have a negative impact on fish survival and production quality.

Parameter	Optimal	Action level
Temperature (°C)	29–31	<12–13
Dissolved oxygen (mg/L)	>5–6[e]	<3.1
Carbon dioxide (mg/L)	<20[e]	>60[e]
ΔP (mm Hg)	<30–40[e]	>85–100
Un-ionized ammonia (mg/L)	<0.43	>1.0[e]
pH	4–9	<3.9 and >10.1

[e] Estimated value.

Figure 2.2: Water Parameter Indication

Fish waste and food surplus are broken down into ionized ammonium (NH4), or non -ionized ammonia (NH3). Ammonium is harmless to fish, whereas ammonia is toxic. The pH of water is the determining factor that causes fish waste and food to decompose into ammonium, or ammonia. A lower pH level will have a higher ammonium concentration, and a higher pH level will have a higher ammonia concentration. Its concentration level is directly proportional to pH. As the pH increases, the concentration (toxicity) of ammonia also increases. When the pH level rises above 9, most of the ammonium in the water is converted into toxic ammonia (NH3), which can kill fish. This stage can also cause gill and kidney damage, impaired growth, and decreased resistance to disease (Chewy Editorial, 2021). Beside of that, rain can be the reason that influence the value of pH water. There's no doubting that rain is good for ponds since it provides a free and gentle water source that's devoid of pollutants like chlorine and chloramine. However, the presence of acid rain can endanger fishponds. Simply put, acid rain is a mixture of wet and dry particles (nitric and sulfuric acids). Technically it refers to all types of rainfall with a pH less than 7, posing a threat to fish. Acid rain is caused by pollution in the air, which is mostly caused by vehicles, power plants, and the burning of coal and oil (Nualgi America, Inc., 2020).

2.2.2 Sensors for Water Monitoring

Water quality monitoring systems require sensors to obtain readings of water quality parameters. A single sensor has the ability to provide different parameters such as for pH level and water temperature. By connecting the sensor with the Raspberry Pi in series to get each value. Each value is decoded and split up using python programming. Water quality can be easily predicted using pH and temperature values.

For use in this project, the DS18B20 Temperature Sensor is an ideal sensor for monitoring water temperature. DS18B20 is a 1 -wire Temperature sensor that can be programmed from a maximum combination. It is widely used to measure temperature in harsh environments such as in chemical solutions or soils.

Figure 2.3: DS18B20 Temperature Sensor

The sensor has a unique single -wire interface method. It only requires a mouth line to connect microprocessors and to establish two-way communication. No peripheral devices are required during use.

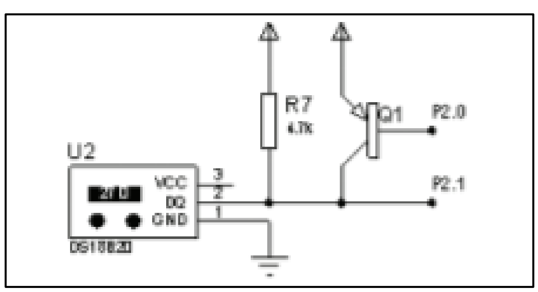


Figure 2.4: DS18B20 Circuit Connection

The power supply is available via a data channel, and the associated voltage ranges from + 3.0V to + 5.5V. The temperature measurement ranges from -55 ° C to +125 ° C, the inherent resolution of the temperature measurement is 0.5 ° C. Digital reading patterns with 9 ~ 12 bits can be realized by programming. The upper and lower alarm thresholds can be set by the user.

The pH of a solution is a measurement of its acidity and alkalinity. The hydrogen ion concentration index is another name for it. The pH scale measures hydrogen-ion activity in solution. Typically, pH is expressed as a number ranging from 0 to 14. pH 7 is considered neutral. A pH of less than 7 is classified as acidic. A pH greater than 7 is then considered alkaline (basic).

Figure 2.5: E-201-C pH Sensor

The pH sensor E-201-C is intended to quantify the pH of a solution and reflect its acidity or alkalinity. This instrument tests the hydrogen-ion activity of a water-based solution and is commonly used to determine the pH of a liquid. Acidic solutions have a high number of hydrogen ions and have a low pH value. Therefore, the more hydrogen ions, the lower the pH, on the contrary, the fewer hydrogen ions, the higher the pH value.

Beside of that, the precautions should be taken when using this sensor. The sensitive glass bubble in the pH probe's head should not encounter the hard substance. The electrode will fail if it is damaged or scratched once touched with the hard material. Moreover, the pH probe needs to be disconnected from the signal converter board once the measurement is complete. For lengthy periods of time, the pH analogue sensor probe should not be attached to the signal converter board without power.

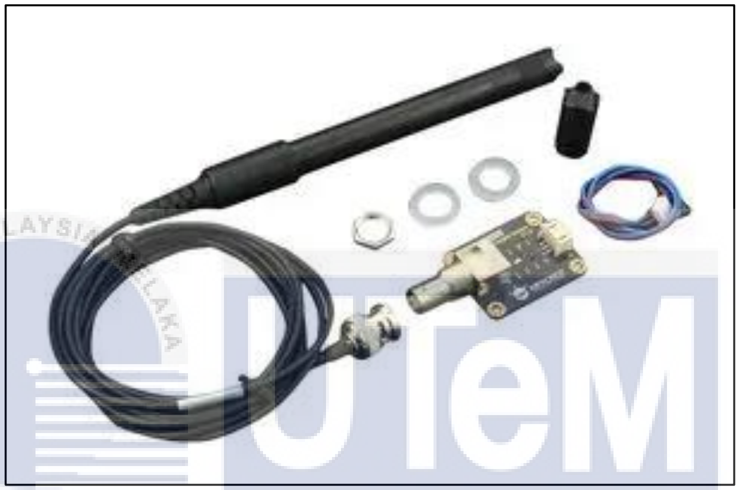


Figure 2.5: Analog Dissolved Oxygen Sensor

Analog Dissolved Oxygen Sensor used to determine the amount of dissolved oxygen in water and hence the water quality. It's used in a variety of water quality applications, including aquaculture, environmental monitoring, and natural science. One of the most important metrics for representing water quality is dissolved oxygen. Low levels of dissolved oxygen in the water will make it difficult for fish to breathe, endangering their lives. However, these sensors will cost a high price and research results this product only available from overseas supplier.

2.2.3 Raspberry Pi

Raspberry Pi is a powerful computer in the size of a credit card. The Raspberry Pi 3 comes with Bluetooth Low Energy (BLE) and Wi-Fi built-in (Balon, B., & Simić, M., 2019). Raspberry Pi 3 has improved power management, with an upgraded switched power source of up to 2.5A, to support more powerful external USB devices. It uses ARM (Advance RISE Machine) with clock speed 1.4 GHz 64Bit quad-core processor.

Figure 2.6: Raspberry Pi 3 Model B+

For years work was done on the Arduino, but the difference is that the Arduino is a microcontroller, and the Raspberry pi is a microprocessor that can process data very efficiently with a 1.4 GHz, 64 Bit Quad-core ARM processor (Nath, O., 2020). Also, the performance of the Pi 3 is about 50-60% faster than the Pi 2 which means it is ten times faster than the original Pi. Table 2.1 shows the detail specification of Raspberry Pi 3 Model B+.

Table 2.1: Specification of Raspberry Pi 3 Model B+

Microprocessor	Raspberry Pi 3 Model B+
SOC Type (Processor)	Broadcom BCM2837B0 (with metal cover)
Core Type	Cortex-A53 (ARMv8) 64-bit
CPU Clock	1.4GHz
RAM	1GB LPDDR2 SDRAM
Wi-Fi	2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN
Bluetooth	Bluetooth 4.2, Bluetooth Low Energy (BLE)

Memory	Micro SD port for loading operating system and storing data
Power Rating	5V/2.5A DC power input
Ethernet	Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
HDMI	Full-size HDMI
USB Port	4 x USB 2.0 ports
POE	Power-over-Ethernet (PoE) support (requires separate PoE HAT)

2.3 Critical Review

In this section of Chapter 2 Literature Review, several work from previous projects will be compared and analyzed to identify appropriate methodologies, techniques, and components that can be used in this project.

2.3.1 Water Quality Monitoring System Based on IoT Platform

According to (Fadel, A. A., & Shujaa, M. I., 2020), proposes a technique for water factory manufacturers by using wireless sensor nodes. This project suggestion due of constant changes in water, either due to seasonal changes in water chemistry or due to the operative circumstances of the manufacturing climate, this dynamic system can be used by the water manufacturers. Therefore, to minimize the risk of contamination, water quality monitoring is defined as a method of obtaining data and information about water from various regions in order to assess water quality over a regular or continuous duration. In order to collect data from water, two types of sensors are used. These sensors are TDS Sensor and pH sensor that are linked to microcontroller devices using Esp32 to handle data collection. The node then sends the status over a wireless network using a specified internet protocol (IP) where all information is transmitted to web application, which will store data on the website.

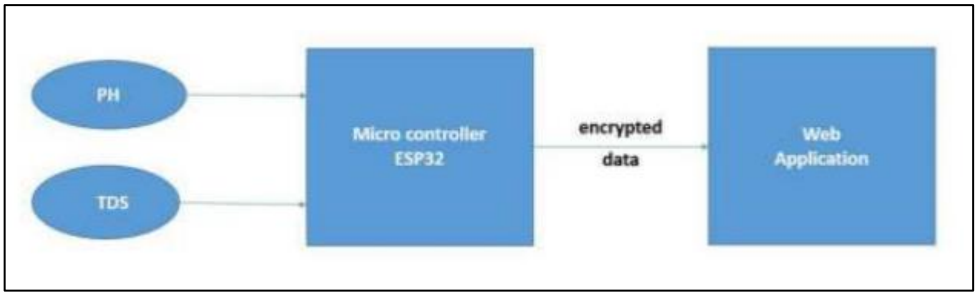


Figure 2.7: Concept Design of Water Quality Monitoring System Based on IoT Platform

The IoT is outlined as networked devices that has the means to sense and collect completely different information from too many locations at a same time. Through the data collected, the system has ability to detect water testing parameters such as pH and TDS with high accuracy and monitor quality of water automatically without human intervention using wireless sensor networks nodes. This reduces the cost and time required to perform laboratory analysis with proposed system.

2.3.2 Monitoring of Water Purification Process Based on IoT

Based on (Khalil, I. M., & Abdulrazzak, H. N., 2019), to get good water quality, a monitoring system is needed that expands the wireless and IoT sensor network. The monitoring device for a drinking water purification system consists of a microcontroller that controls the entire process of the purification system. Besides, LCD notifies and tell the information to the maintenance person, a detector that analyzes data to determine the healthy condition of the filter element, a reminder device that generates a message warn the user about the unhealthy water condition of the filter element.

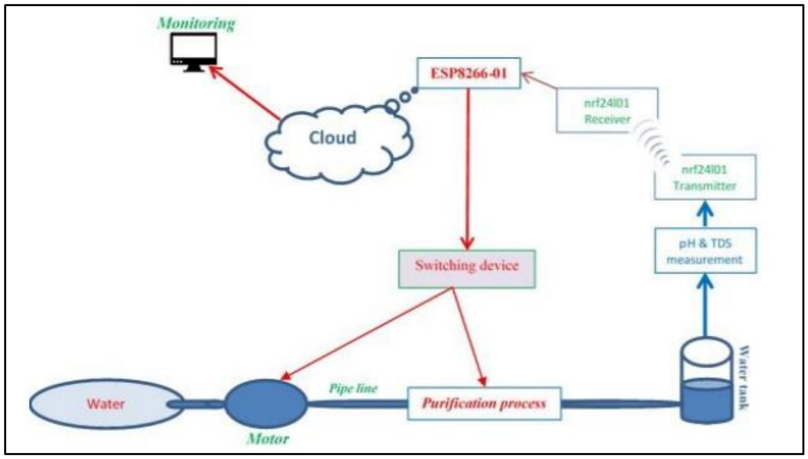


Figure 2.8: Concept Design of Monitoring of Water Purification Process Based on IoT

The system comprised of two microcontrollers, one of which is the Arduino Uno which is used to link the sensing element and transfer the measured value via the NRF24L01, which serves as a transmission node. The other microcontroller is the ESP8266 which is used to obtain the measured value by attaching to another node NRF24L01 and transmitting the purification system status to the global database server. Server then sends the status to the secure web, which notifies the maintenance person, Monitoring is meant to display the measured value and the switch on/off purification mechanism based on the measured value compiled from the sensor. A reminder is made by producing a message and displaying the graphical state of the unhealthy water filtration elements. Based on Figure 2.8, good water quality results were obtained using the following algorithm, which took into account suitable pH and TDS values and indicated the system's status in terms of pH and TDS separately using the equation.

$$SystemStatuswithrespecttopH = \begin{cases} 1 & whenpHwithintheacceptancerange \\ 0 & whenpHoutoftheacceptancerange \end{cases} \quad (1)$$

$$SystemStatuswithrespecttoTDS = \begin{cases} 1 & whenTDS\ within\ the\ acceptance\ range \\ 0 & whenTDS\ out\ of\ the\ acceptance\ range \end{cases} \quad (2)$$

Figure 2.9: Operation Algorithm

2.3.3 Water Quality Monitoring using IoT

By referring to (Sengupta et al., 2019), because of urbanization and pollution, it is now important to track and assess the quality of water entering homes. The project focuses on measuring variables such as pH, turbidity, and water temperature, which can be checked on a regular basis. The pH sensors, turbidity sensors, and temperature sensors are some of the sensors used to measure water quality. The pH and turbidity sensors are analog sensors, while the temperature sensor is a digital sensor. The Raspberry Pi 3 B+ used only accommodate digital inputs. Therefore, the temperature sensor is directly attached to the Raspberry Pi's GPIO pins, while the pH and turbidity sensors are connected to the Analog to Digital Converter (ADC) to receive digital output. The Raspberry Pi module processes the sensor data and sends it to the cloud server. The sensed values are available on the cloud through cloud computing and are used to decide whether or not the data is within a safe range. Depending on whether or not this condition is met, the Raspberry Pi rules the relay machine, which determines whether or not the water supply can be continued. The water quality parameters thus recorded are then shown on web server, where the relevant authority may track and manage the water supply manually.

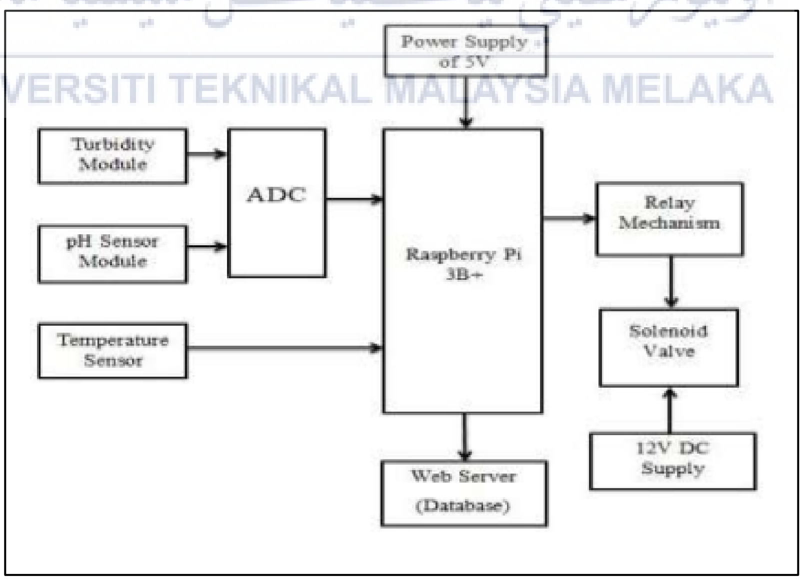


Figure 2.10: Concept Design of Water Quality Monitoring using IoT

Temperature and pH value relations also been discussed. The pH of water is also affected by temperature changes. Since pH varies with temperature, the pH of water at 10°C will not be the same as the pH of water at 25°C. The pH value in water reduces as the temperature rises, making the water more acidic. As seen in Figure 2.11, the pH values vary very little in the acidic region as the temperature rises, but dramatically in the basic region.

pH Range	Temperature		
	0°C	25°C	60°C
Acid	0.99	1.00	1.01
Neutral	7.47	7.00	6.51
Base	14.94	14.00	13.02

Figure 2.11: Change in pH due to increase in temperature

Based on (Fadel, A. A., & Shujaa, M. I., 2020), (Khalil, I. M., & Abdulrazzak, H. N., 2019) and (Sengupta et al., 2019) studies, their projects focus on the web-based platform monitoring. Nowadays, mobile phone has commonly been used and are often held close to the user. Warning alerts via mobile phones make it easier for users to constantly monitor. As a result, fish breeders will be able to get fast warning updates and information about the water quality in their fishpond.

Table 2.2: Comparison of Previous Project

Journal Name / Author	Microcontroller / Microprocessor	Sensor Type	Parameters	Communication Technology	Platform
Water Quality Monitoring System Based On IoT Platform (Fadel, A. A., & Shujaa, M. I., 2020)	ESP 32	Total Dissolved Solid (TDS) Sensor & pH Sensor	TDS & pH Value	Wi-Fi	Web-Based
Monitoring of Water Purification Process Based on IoT (Khalil, I. M., & Abdulrazzak, H. N., 2019)	Arduino Uno & ESP8266	Total Dissolved Solid (TDS) Sensor & pH Sensor	TDS & pH Value	Wi-Fi	Web-Based
Water Quality Monitoring using IoT (Sengupta et al., 2019)	Raspberry Pi 3B+	Turbidity Module, pH Sensor Module, & Temperature Sensor	Turbidity, pH, & Temperature	Not stated	Web-Based

Proposed Solution	Raspberry Pi 3B+	pH Sensor Module, & Temperature Sensor	pH & Temperature	Wi-Fi	Mobile Apps

Based on Table 2.2 above shows the comparison between the previous projects. As been showed, the previous projects used ESP 32, Arduino Uno, ESP8266, and Raspberry Pi 3B+ as their microcontroller or microprocessor. In this project used Raspberry Pi 3B+ as the microprocessor. Total Dissolved Solid (TDS) Sensor, pH Sensor, and Temperature Sensor are among the sensors used in the previous project, hence this project used pH Sensor and Temperature Sensor as the sensor which monitor the pH and temperature water parameters. Moreover, most of the previous projects used Wi-Fi as a communication technology as well in this project. On the other hand, all the previous projects provide a web-based platform, but in this project mobile apps are implemented.

Table 2.3: Functionality Comparison Between Previous Projects

Journal Name / Author	Display Data	Mobile Apps Dashboard (Visualize Data)	Notification
Water Quality Monitoring System Based on IoT Platform (Fadel, A. A., & Shujaa, M. I., 2020)	✓	✗	✗
Monitoring of Water Purification Process Based on IoT (Khalil, I. M., & Abdulrazzak, H. N., 2019)	✓	✗	✓
Water Quality Monitoring using IoT (Sengupta et al., 2019)	✓	✗	✗
Proposed Solution	✓	✓	✓

Table 2.3 shows the gaps from the existing works with the proposed project. It clearly highlights that all the previous project and this proposed project have a display data for the monitoring purpose. Moreover, the main advantage of this proposed project is to provide a mobile application dashboard to users that was not available in the previous project. Besides, most of the previous projects did not have an alert function whereas this proposed project was to provide and use an alert function.

2.4 Proposed Solution

From the critical reviews, this project is aims to improve and develop a system for measuring water quality using pH sensor and temperature sensor good detection accuracy. The pH and temperature are crucial water parameter to be measured to determine the water quality condition in the fishpond. The optimal water temperature for tilapia growth is between 29 and 31 ◦C whereas the pH of water should be maintained between 4 and 9. The value outside this level can affect growth and incidence of disease to tilapia as discussed early in this chapter.

Unclean water can cause pathogens that may affect aquaculture growth and safety (De Belen, M. C., & Cruz, F. R. G., 2017). Monitoring the water can be done by using pH sensor and temperature sensor which will constantly record the pH and water temperature value. The importance of correct pH and temperature values in the preparation of accurate data should not be underestimated. Once water parameters have been recorded, data should be delivered to fish breeders. Breeders are able to monitor the water quality of their fishponds even if they are not in the pond area. This would increase the water quality and the effectiveness with which the fishpond is maintained.

The system would have a microprocessor, Raspberry Pi 3 B+, which will read data from pH sensor and temperature sensor to detect pH and temperature values and then send the information to a cloud database platform. Analog-to-Digital Converter can be used to transform data from analog to digital so that it can be read. The data would be submitted to the user's smartphone and made available online. If the water parameters exceed the action level value, a warning message will be sent to the mobile. The fish breeders will get alert notification from their mobile and will take the appropriate actions to keep the fishpond in good water quality condition.

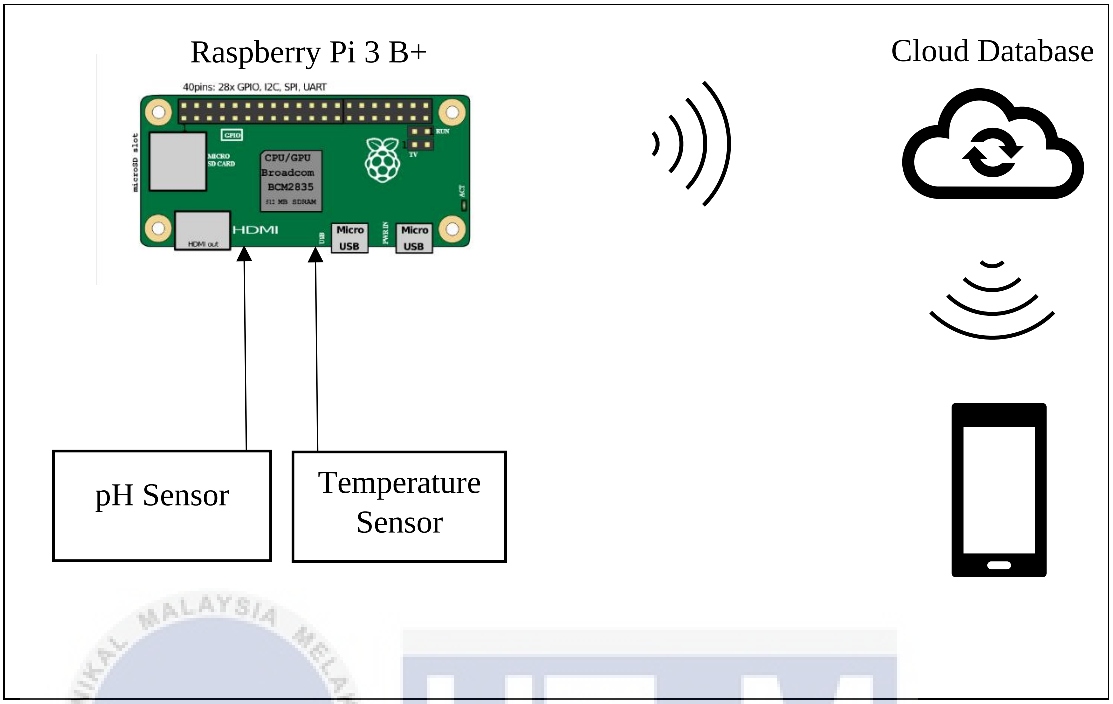


Figure 2.12: Architecture Design for Proposed Solution

2.5 Conclusion

This chapter discusses the project's literature analysis, which is an ongoing project review, in order to complete a successful project. The system architecture review will be explored in depth. The literature review includes data collection and details, as well as problem identification and criteria. Any outcome is a result of the data review, and some conclusions and recommendations have been drawn from the findings of this chapter.

CHAPTER 3: PROJECT METHODOLOGY

3.1 Introduction

This chapter went through in detail about the methodology used as a guide and reference to make sure the project is in track and in the correct order. The methodology offers steps and procedures in achieving the milestones while in the project completion process. There are various types of methodologies that can be selected for use as a reference, using the right methodology can determine the overall success and quality of the project report.

Based on this IoT project, SDLC methodology, namely the Agile model, is seen to be able to provide good effectiveness in developing the project. Agile model can provide mobility and flexible advantages throughout the development phase. Agile model breaks down tasks into iterations or smaller parts. Development and testing are based on continuous iteration throughout the project development life cycle. Both development and testing activities been conduct simultaneous.

3.2 Methodology

The methodology chosen for this project is the Agile model. Just like other methodologies, Agile model also has its own stages in implementing the project. This provides descriptive guidelines for creating efficient performance support in four stages. Agile Model phases consists of Requirements, Architecture and Design, Development, and Testing and Feedback phases. The Agile model offers a flexible guide to build efficient and high-quality system.

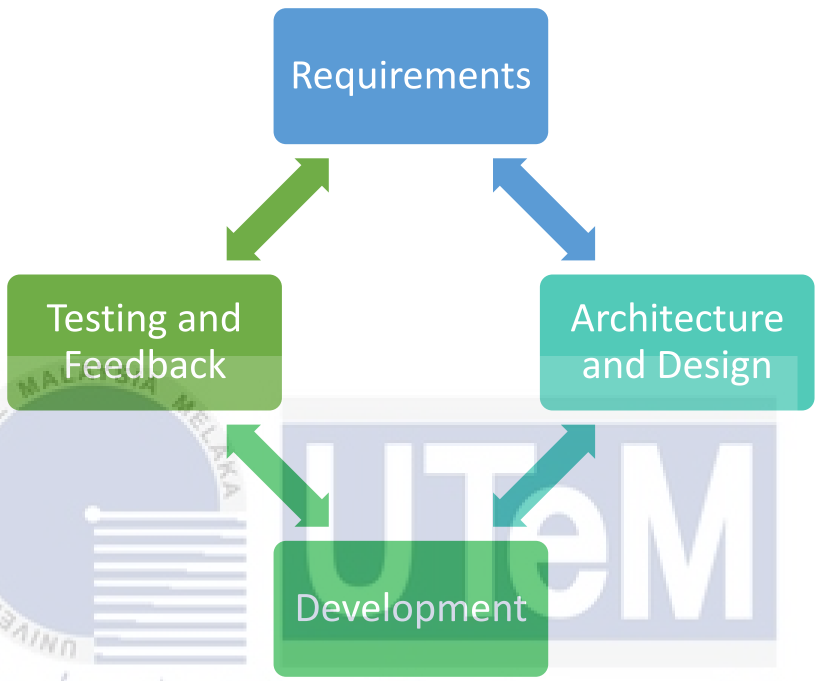


Figure 3.1: Agile Model

Based on Figure 3.1, shows the methodology of Agile model. Agile model starts with the requirements phase where the project requirements are identified. This procedure then moves to the architectural and design phase, when the system setting is drafted. The system development phase will begin in accordance with the project plan. Finally, testing and feedback phases were undertaken in order to obtain any input and enhance the project. Aside from that, this methodology allows for flexibility in the project implementation process. Once completed with the specific phases, able to refer back to the prior phase for any modification and improvement.

3.2.1 Requirements Phase

Each project development life cycle requires to discuss the requirements needed to develop a product. The main objective of this phase is to identify and define the system requirements in detail. All the requirements will be listed and detailed. It is necessary in ensure that process representatives have a clear understanding of the role and how each requirement should be implemented.

Table 3.1: System Requirement

Item	Description
Collect the data	The system able to detect and collect the data from the sensors.
Monitor and alert notification	The system offers to monitor and alert the user when exceeding the normal values via notification.
Testing the effectiveness of system	The system should be effective for consumerism.

The selection of hardware is done after considering based on several factors such as hardware availability, affordable costs, and the suitability of the hardware to support in developing this project system. Based on Table 3.1, system requirement been listed and each of the system requirement been described.

Table 3.2: Hardware Requirement

Item	Description
Raspberry Pi 3 Model B+	This microprocessor makes it an ideal candidate for this IoT project as multiple sensors can be connected to it simultaneously. It used to host Raspbian operating systems and offers to connect and control external electronic devices via a set of GPIOs, or ‘General Purpose Outputs’.
E-201-C pH Sensor	Selection of this sensor model is based on its affordable price compared to other models in addition to being able to provide the same functionality. It is convenient to use with Raspberry Pi to measure the PH of liquids.

DS18B20 Temperature Sensor	This is the only one common sensor available in the market to be used. It is used to measure temperature value in harsh environments like in chemical solutions, mines or soil.
Breadboard	This breadboard be used to act as the platforms that allow to build and test electronic circuits without having to solder anything.
MCP3008	MCP3008 is an Analog-to-Digital Converter used to convert the Analog voltage to digital value. This converter is needed because E-201-C pH Sensor is an analog sensor. The Raspberry Pi computer does not read analog inputs. Therefore, the MCP3008 will translate for Raspberry Pi use.
Jumper Wire	Used to making connections between items on the breadboard and Raspberry Pi pins.
Resistor 4.7K Ohm	Used to reduce current flow

Based on Table 3.2, hardware requirement been listed and the description details of each of the hardware been elaborate.

Table 3.3: Software Requirement

Item	**Description**
Raspbian OS	Free operating system based on Debian, optimized for the purpose of Raspberry Pi hardware use. This software been chosen because of it is official OS for Raspberry Pi.
Kodular	Tools used to build applications for Android devices. Kodular is a web-based platform that does not require a high pc

	specification, this allows application development to be done more easily and economically.
Thingspeak	Thingspeak is a cloud-hosted database used to store and sync data in real-time. It is used to obtain a graph visualization of the data stored and included in the application of this project.

Based on Table 3.3, software requirement been listed and the description details of each of the software been elaborate.

3.2.2 Architecture and Design Phase

Architecture and design are a phase that discuss about the architecture of the project that helps in defining the overall system using the requirement specify in the previous phase which is requirement phase. The prototype design is to use the sensors to measure water quality and send data to the Raspberry Pi. The program code will be written specifically for each sensor. Then, the connection circuit design between the sensor, microprocessor and microchip on the breadboard is drafted.

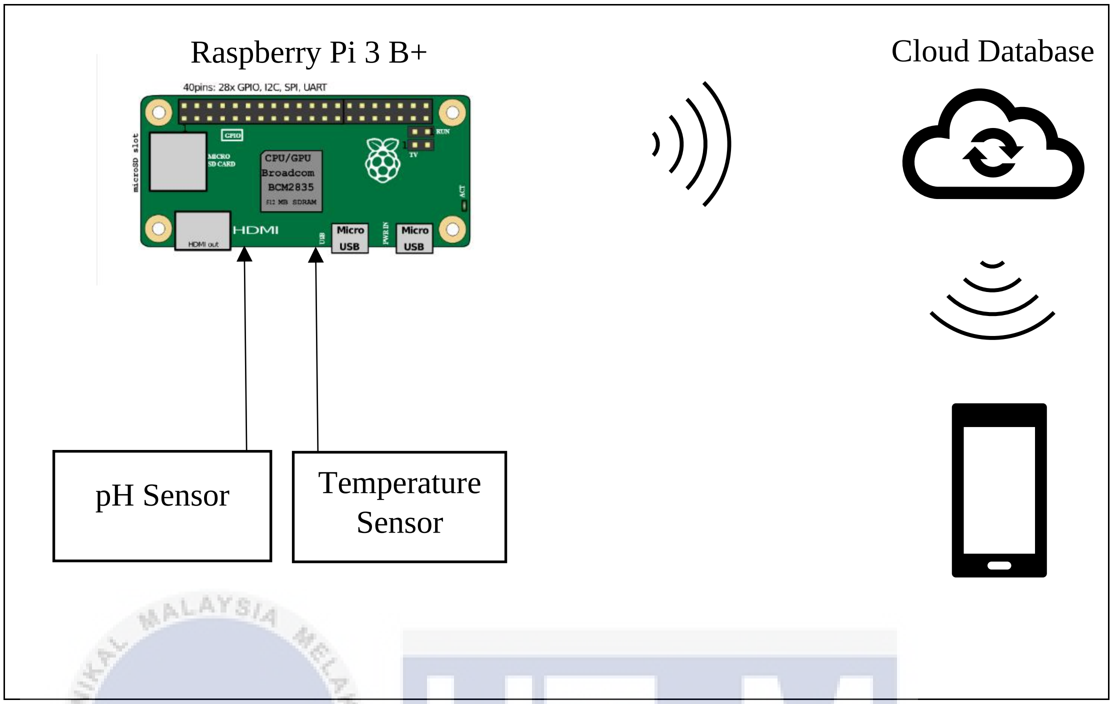


Figure 3.2: Overall System Design

Based on Figure 3.2, shows an overview design of the overall system to illustrate how the system will work. Basically, Raspberry Pi 3 B+ will gained the data from sensors which are pH sensor and Temperature sensor. Raspberry Pi 3 B+ then analyze this data and transmit it to the cloud database. From this, mobile apps will show the output based on the data received from the cloud database.

3.2.3 Development Phase

The development process begins with the connection of each hardware device. Once the connection is complete, the Raspbian OS is installed as the operating system to the Raspberry Pi's SD card. All configurations will be recorded and stored on the SD card; this is where all the code will be written on Raspberry Pi. The connection between the sensor and microprocessor is then tested to obtain an output reading.

Once an accurate read is achieved, it then needs to be transferred to the cloud database. Wi-Fi connection been established on the Raspberry Pi to ensure its connection to the internet. Firebase cloud database then been setting up to receive the data from the

Raspberry Pi. After that, application development will start using Android Studio. This application will be developed to suits the android platform and will be installed on smartphones. The application should receive all the upcoming data gain from the cloud database and display it on the smartphone.

3.2.4 Testing and Feedback Phase

In this phase, the testing process been conducted to ensure that everything goes in line with the expected results and goals. Android emulator have been used to run the testing of the application developed. After that, testing is done on a real smartphone device before it can be deployed on the real situation. Any feedback and review will be sought to improve and correct any issue from the developed application. This testing process is repeated until all critical issues are removed, and the application is developed stably.

3.3 Project Milestones

Table 3.4: Project Milestone

Phase	Activity
Requirements	▪ Project proposal ▪ Proposal assessment and verification ▪ Proposal improvement and correction ▪ Proposal submission ▪ Project objectives, scopes, and problem statement identification ▪ Project requirements identification
Architecture and Design	▪ Design the project architecture ▪ Project hardware connection identification ▪ Construct flow chart for the system flow
Development	▪ Build the hardware connection ▪ Program code been written into the Raspberry Pi ▪ Set up connection with the cloud database

	▪ Develop the application for android platform
Testing and Feedback	▪ Monitor the system for any error and flaw ▪ System is tested into real environment scenarios ▪ Test and collect feedback from users ▪ Improve and correct any issue arise

Table 3.4 shows the phases involved in the project milestone. The phases include Requirement's phase, Architecture and Design phase, Development phase, and Testing and Feedback phase. Activities involved from each of the phases been listed.

Table 3.5: Project Gantt Chart

Task Name	Week																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Requirements																					
Architecture and Design																					
Development																					
Testing and Feedback																					

Based on Table 3.5, shows the project gantt chart. This project is divided into two sessions, namely Final Year Project 1 (FYP 1) and Final Year Project 2 (FYP 2). Basically, Final Year Project 1 (FYP 1) will involve between week 1 to week 14. On the other hand, Final Year Project 2 (FYP 2) will be performed for 7 weeks which will involve between week 15 to week 21. Therefore, Final Year Project 1 will go through the Requirement, Architecture and Design, and half of the Development phases while Final Year Project 2 will go through the Development and also the Testing and Feedback phases.

3.4 Conclusion

To summarize, Agile SDLC model methodology is chosen to assist project development in deliver the project efficiencies to meet the deadline. This chapter consist of the significance of planning by committing the steps determined in the selected methodology. Project milestones are stated for planning use to guide project implementation according to a predetermined timeline. For the next chapter, analysis and design will be discussed and explained. Complete details on the design and structure of the system will be described in the next chapter.

CHAPTER 4: ANALYSIS AND DESIGN

4.1 Introduction

Chapter 4 cover about the analysis and design of the project in determine on the best plans and preparations for the project. The project problem been identified to be analyses before commit with the further analysis and project requirement. Moreover, the project design been developed to achieve the expected outcome or output. Then, the hardware and software that related and required in this project also will been discusses in this chapter.

4.2 Problem Analysis

Water quality is a serious matter that fish breeders need to consider. The production and rearing of fish farms are influenced by good water quality factors in accordance with the needs of the fish. Therefore, care and monitoring of fishpond water quality should always be done from time to time. However, fish farmers have certain obstacles in conducting the monitoring. Among them are manual method used by the fish breeders, they need to perform manual monitoring activities that are somewhat little energy -intensive and time consuming. Therefore, the Raspberry Pi microprocessor and sensors such as pH sensor and temperature sensor will be placed at the location of the fishponds to perform the water monitoring process, this does not need the fish breeders to go down the location every time the monitoring process be done. Besides, no indicator for water maintenance where the water maintenance is only based on the physical changes of the water color. By implementing the system, pH and temperature will be the main indicator of the water parameter by appreciating the pH sensor and temperature sensor.

Moreover, fish breeders are unable to know when the exact time the immediate maintenance can be done when the water quality declines. Therefore, the system will be designed to provide a mobile apps using Kodular platform with a real-time information that can assist the breeders in knowing their fishpond water quality status as well as get notifications when water quality changes occur.

4.3 Requirement Analysis

4.3.1 Data Requirement

In monitoring the water quality, data means significance things that need to be process. Therefore, sensors will act to detect and measure the information such as temperature and pH value. This data will be process locally in the Raspberry Pi. Once then, it will be transferred to be stored in the ThingSpeak cloud database. ThingSpeak is a cloud based IoT analytics tool with built in MATLAB analytics that lets to gather, visualize, and analyze live data streams. Below, Table 4.1 shows the data dictionary of the data that been generated and stored in the ThingSpeak cloud database.

Table 4.1: ThingSpeak Data Dictionary

Field Name	Data Type	Description
created_at	Datetime	Time and date will be auto generated once receive the value.
entry_id	Int	Entry number for the received data.
field1	Double	Data value received from Raspberry Pi

4.3.2 Functional Requirement

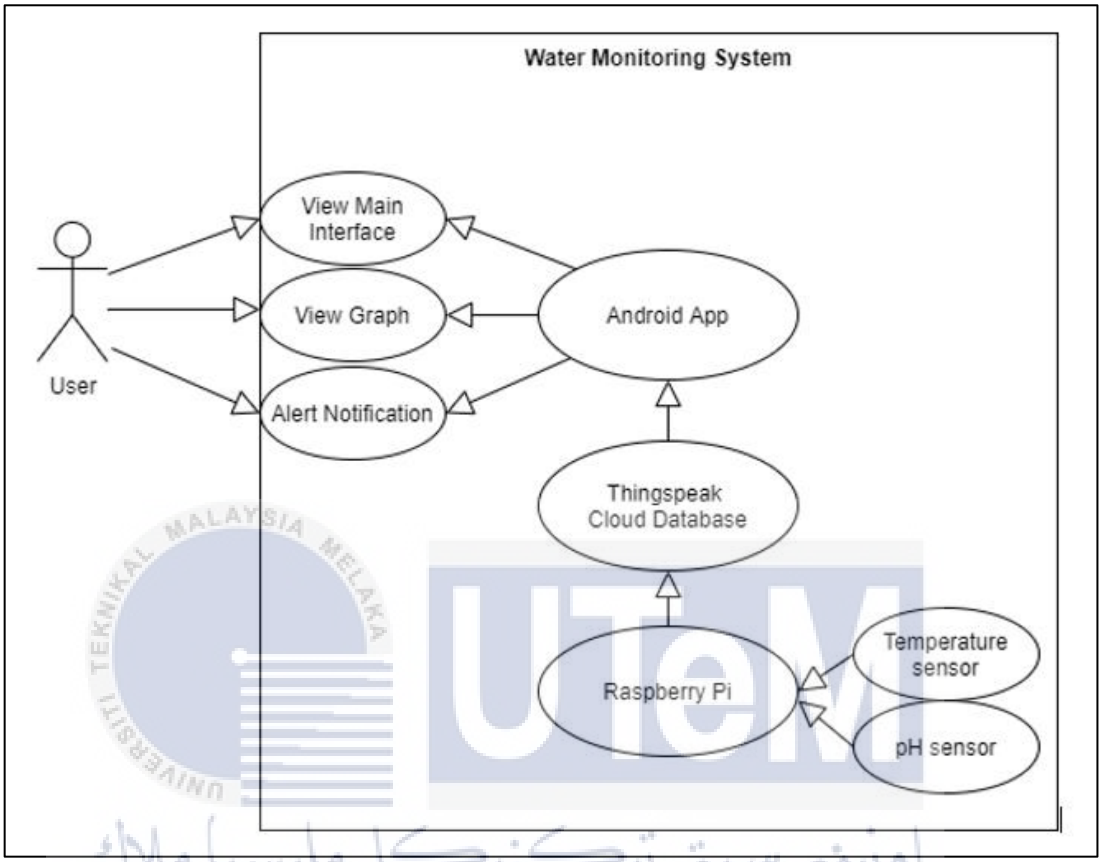


Figure 4.1: System Use Case Diagram

Based on Figure 4.1, shows the use case diagram of the water monitoring system. From this, the functional requirement of this system is to allow the user to view the main interface for monitoring, view the graph, and get the alert notification from the application. Basically, all the information generated in the android app are being processed and gathered from the ThingSpeak cloud database. This data information was previously obtained from the Raspberry Pi that generated from the sensors.

4.3.3 Non-functional Requirement

This system will perform according to few quality requirements such as Consistency, Usability, and Availability:

Consistency

The system will show and display the data consistently as long as the Raspberry Pi and the sensors keep running and generated the value to be transmitted to the cloud database.

Usability

It is important to ensure that the application that has been designed is easy to be used and understand especially for the novice user.

Availability

System services will function as required and will not fail or undergo repair actions when required.

4.3.4 Others Requirement

Software Specifications:

- **Raspbian OS**
  Raspbian OS is a Raspberry Pi operating system based on Debian. The Raspberry Pi Foundation has officially made it accessible as the primary operating system for the Raspberry Pi series of tiny single board computers since 2015. By default, the Raspberry Pi supports C/C++, Python 2/3, and Scratch. On Raspbian OS, however, virtually any language compiler or interpreter may be installed.

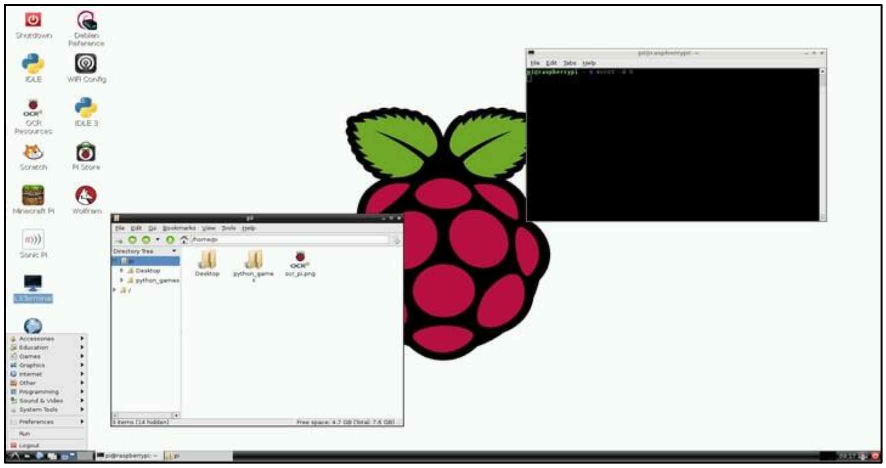


Figure 4.2: Raspberry OS

- **Kodular**

  Kodular is an online suite tools for mobile app development. It mainly provides a free drag-and-drop Android app creator without coding, based on MIT AppInventor. It brings lots of new features like new components and blocks. It also provides a free online app store to share and distribute apps and an extensions IDE for advanced users.

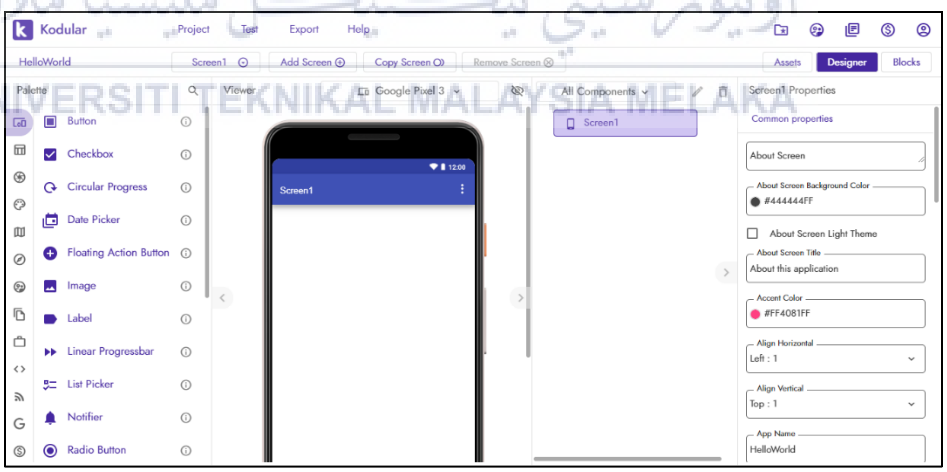


Figure 4.3: Kodular

- **ThingSpeak**

  ThingSpeak is an open-source Internet of Things (IoT) application and API for storing and retrieving data from things over the Internet or over a Local Area

Network utilizing the HTTP and MQTT protocols. ThingSpeak allows developers to create sensor recording apps, location tracking apps, and a social network of things with status updates.

Figure 4.4: ThingSpeak

Hardware Specifications:

- **Raspberry Pi 3 Model B+**
- **E-201-C pH Sensor**
- **DS18B20 Temperature Sensor**

  Raspberry Pi and sensors information have been explained in detail as mentioned in the previous Chapter 2 Critical Review.
- **Breadboard**

  A breadboard is a rectangular piece of circuit board with many mounting holes. It is used to link electronic components to single-board computers and microcontrollers like the Arduino and Raspberry Pi. The connections are not permanent and may be removed and reinstalled at any time.

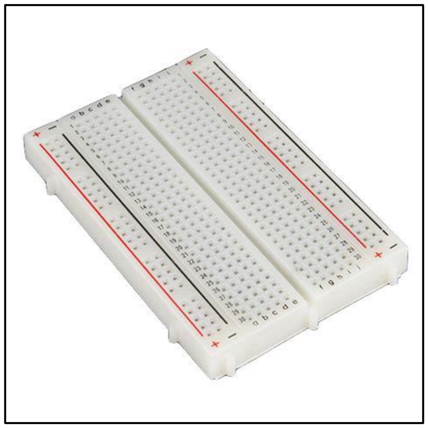


Figure 4.8: Breadboard

- **MCP3008**

  The MCP3008 is well-known for providing analogue input to a Raspberry Pi that is exclusively digital. This chip will give your microcontroller or microcomputer project 8 channels of 10-bit analogue input.

Figure 4.9: MCP3008

- **Jumper Wire**

  A jumper wire is an electrical wire having a connector or pin on both ends that is used to connect the components of a breadboard.

Figure 4.10: Jumper Wire

- **Resistor 4.7K Ohm**

  Resistors are used for a variety of purposes such as reducing the current flow. It's a two-terminal passive electrical component that acts as a circuit element by implementing electrical resistance.

Figure 4.11: Resistor

4.4 High-Level Design

Development of Water Quality Monitoring System interacts with the environmental of the physical sensor which are the pH sensor and the temperature sensor. This interaction requires the interconnection between the physical equipment and the software applications. Therefore, it is a responsible to the software as an intermediary to enable the system to function. Raspbian OS that based on the Raspberry Pi will then be run and execute the code for the system functional.

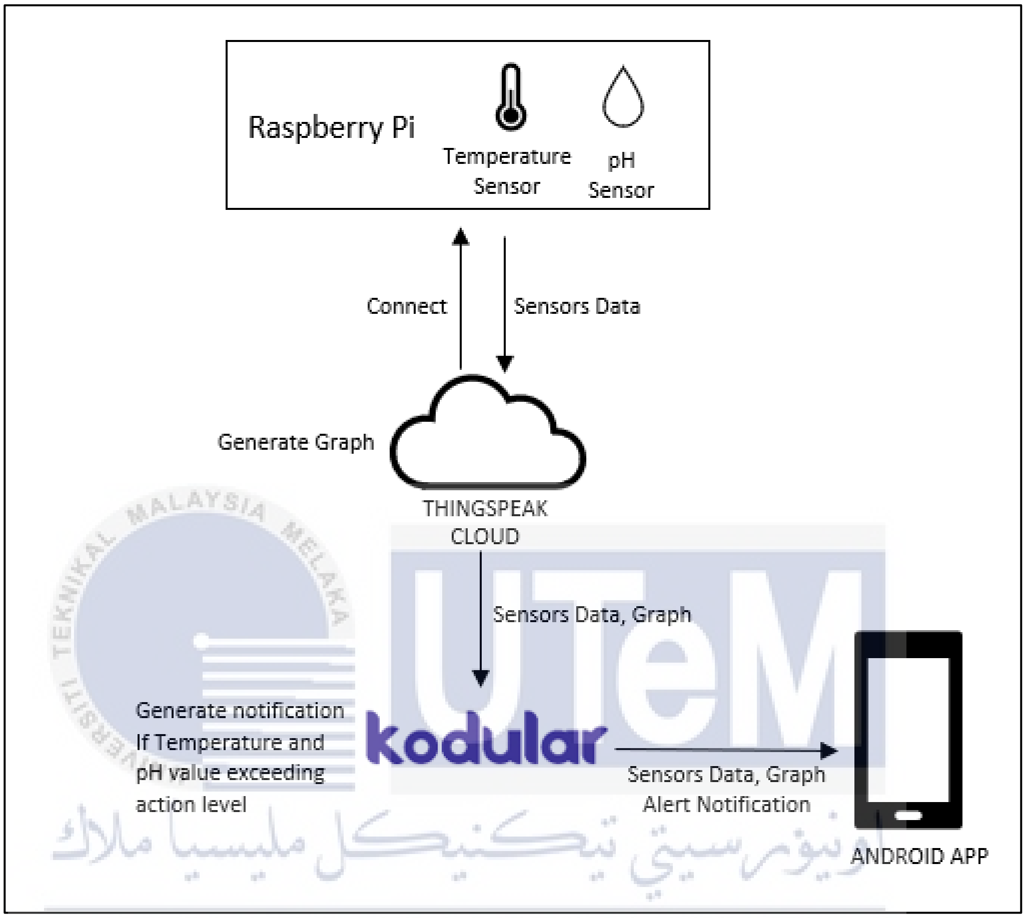


Figure 4.12: High Level Design Diagram

As been illustrated in Figure 4.12 above, once Raspberry Pi read the data gained from the sensors, it will then be connected and transferred the data to the ThingSpeak cloud. Graph will be generated based on the data value received. Once then, Kodular will received the sensor data and the graph, alert notification will be produced if the data values exceed the action level to notify the user.

4.4.1 System Architecture

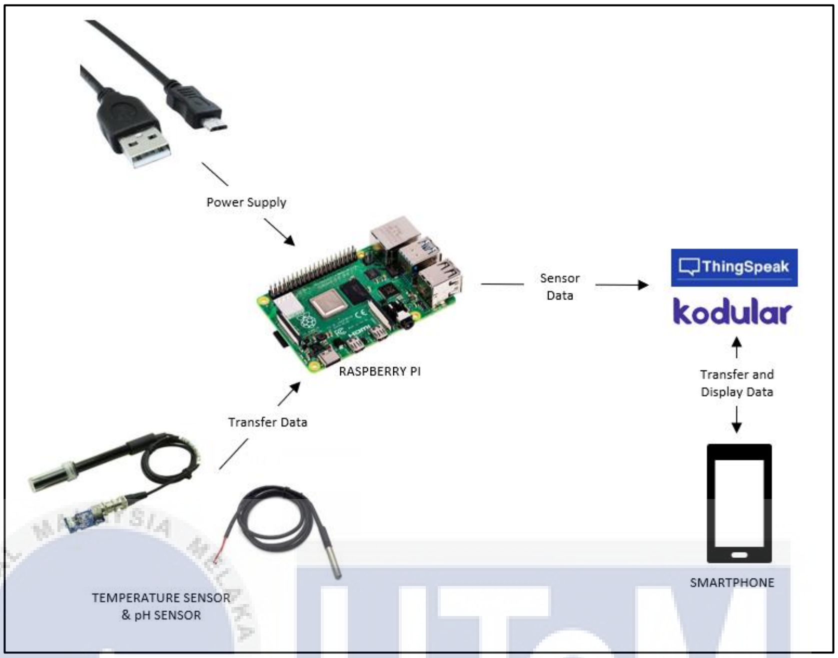


Figure 4.13: System Architecture

Based on Figure 4.13, Micro USB will be connected to power up the Raspberry Pi. This system uses the Raspberry Pi microprocessor which will be programmed to execute and read the data generated from the sensors to obtain the temperature and pH value of the water. This sensors data will be sent to the smartphone user via the built-in Wi-Fi module on the Raspberry Pi to be connected to the cloud database first.

4.4.2 User Interface Design

The visual layout of the items that a user could interact with in an app is referred to as user interface design. User interface designs will be not only appealing to the users, but also practical and designed with them in mind especially for the use of novice user.

Figure 4.14: Main Dashboard

Figure 4.14 above shows the application user interface for the main dashboard of the water quality monitoring system. The main purpose of this interface is to provide users the output value from the temperature sensor & pH sensor, and the real-time graph that present the historical data for a given period of time. As been discussed in Chapter 2 Critical Review, pH of water is affected by temperature changes. The pH value in water reduces as the temperature rises, making the water more acidic. Therefore, the plotted graph of pH and temperature on figure above to some extent undergoing the changes due to the temperature changes that affected by environmental factors.

4.5 Detailed Design

This part depicts the project's detailed design, which includes the structure perspective of the system to be built, which includes the circuit diagram, system flow chart, and pseudocode.

4.5.1 Circuit Diagram

Circuit diagram is a drawing that depict what the physical arrangement of the wires and the components connected. It is used to build and maintain the component and hardware devices. Once finished the prototype, this design can be used as the clear reference and guideline for understanding how it work. Due to avoid the circuit diagram being too crowded and dizzying, this diagram is divided into two diagrams which are the pH sensor circuit diagram and the temperature sensor circuit diagram.

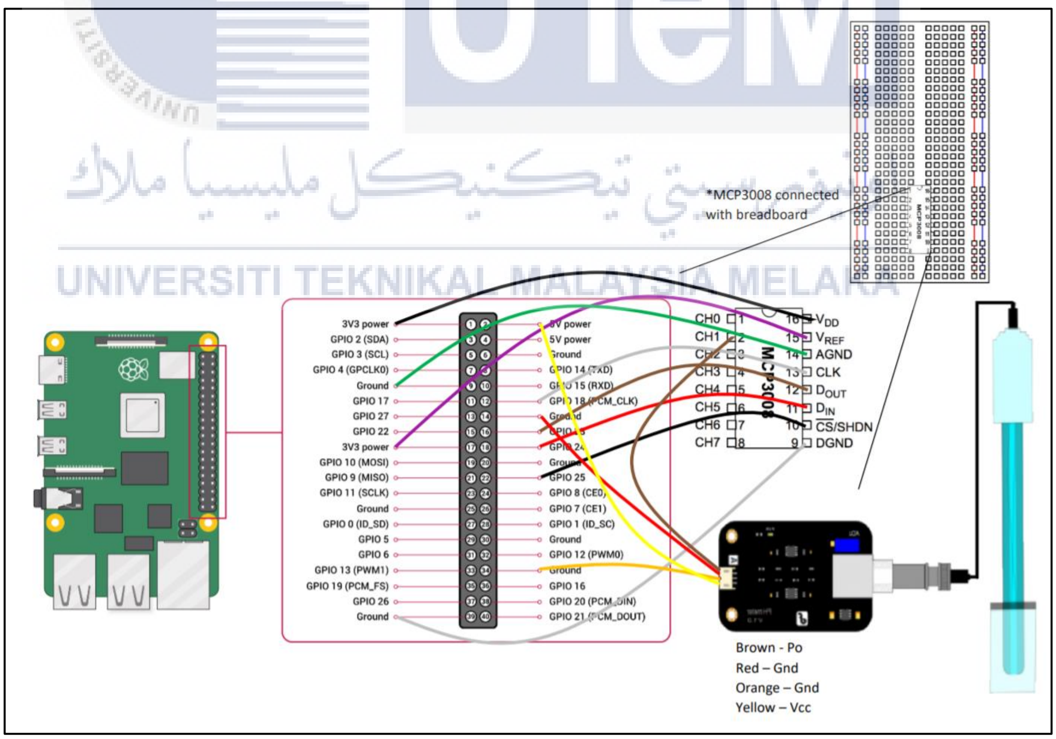


Figure 4.15: pH Sensor Circuit Diagram

Based on Figure 4.15, shows the pH Sensor Circuit Diagram. This diagram consists of wire connections and components that associated with a pH sensor.

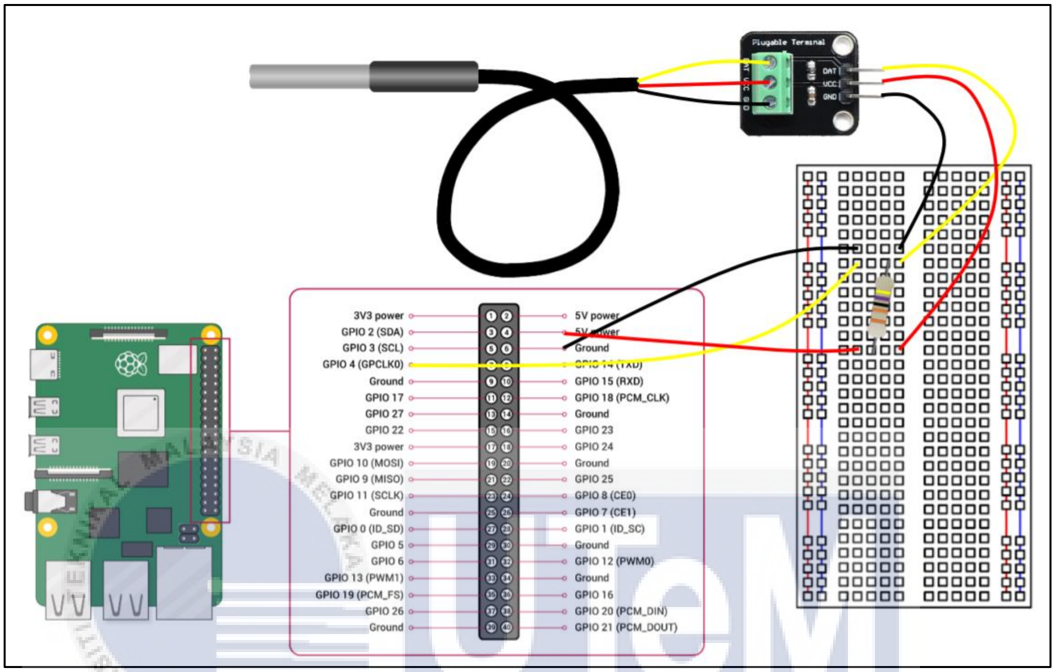


Figure 4.16: Temperature Sensor Circuit Diagram

Based on Figure 4.16, shows the Temperature Sensor Circuit Diagram. This diagram consists of wire connections and components that associated with a temperature sensor.

4.5.2 Flow Chart

A flowchart is a diagram that shows how a workflow or process works. A flowchart is a diagrammatic depiction of an algorithm, or a step-by-step procedure for completing certain process.

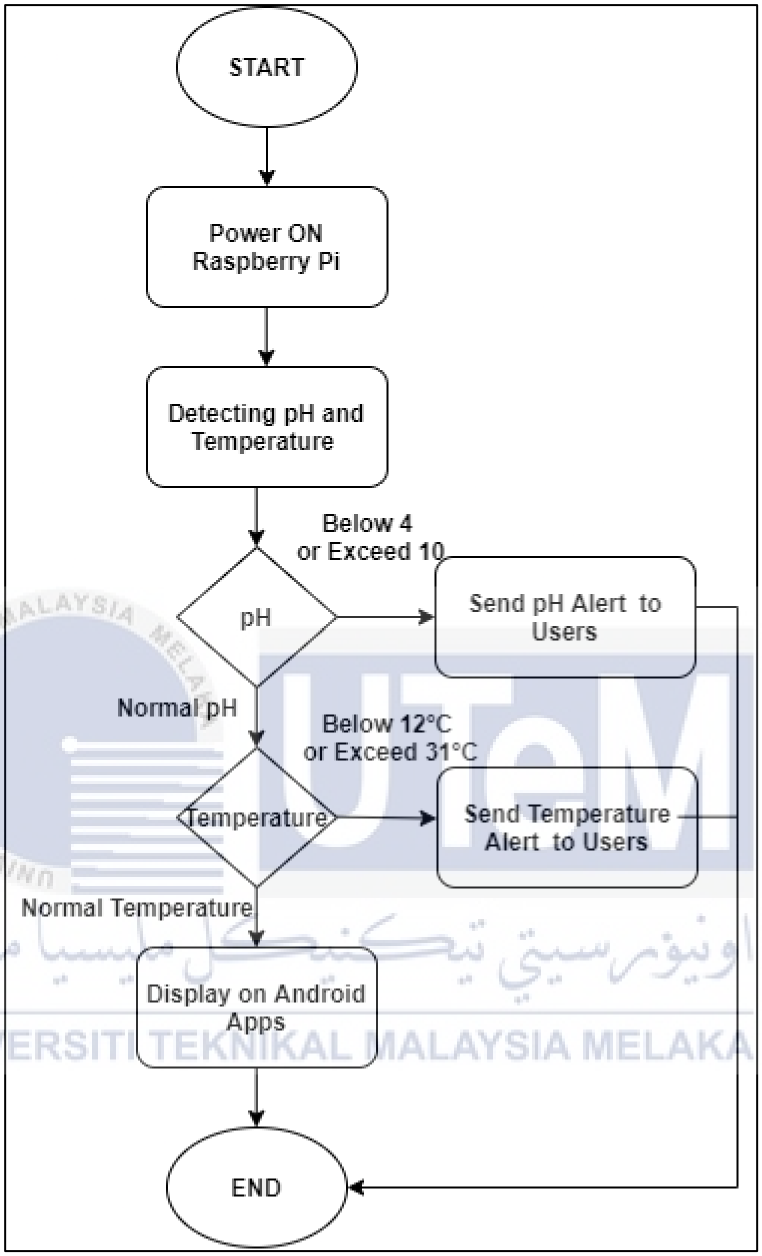


Figure 4.17: Flow Chart

Based on Figure 4.17, shows the Flow Chart diagram that explaining the process of the system flow. The system starts by power on the raspberry pi, then it will be functioning to detect the pH and temperature value. The conditional decision has determined based on the previously discussed in Chapter 2 Literature Review. Once then, the conditional decision of pH and temperature value will be finalized, if unnormal value been detected will send an alert notification to the users. The value will been display on the android apps that can be monitored by users.

4.5.3 Pseudocode

Pseudocode is basically a common language explanation of an algorithm or code. It is designed so that humans can understand and not machines to understand its code. Pseudocode is better understood than any algorithm code.

```
Program to detect pH and Temperature
Initialize
    Input
        pH
        Temperature

Start loop
    IF pH < THEN
        OUTPUT "Low pH value"
    ELSEIF pH > THEN
        OUTPUT "High pH value"
    ELSE
        Display pH value
    IF Temperature < THEN
        OUTPUT "Low Temperature value"
    ELSEIF Temperature > THEN
        OUTPUT "High Temperature value"
    ELSE
        Display Temperature value
End loop
```

Figure 4.18: Pseudocode

Based on Figure 4.18, shows the pseudocode. It is used to design an algorithm by sketching out the program's structure before moving on to the actual coding implementation.

4.6 Conclusion

The stages of project development include analysis and design. Prior to design implementation, all software and hardware specifications must be defined and validated. This chapter is the application's pre-preparation phase, in which the overall flow of the system is described so that it may be better understood before implementation. This chapter also covers all the critical design requirements, which will be executed and tested in the following chapter. It also discusses the intended project's architectural design.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

In this chapter, the design and analysis from the previous chapter be implemented. The implementation of the project discussed in depth in this chapter. This chapter also cover project environment setup and configuration management, which ensure that the project runs smoothly and efficiently. The Raspberry Pi is used in this project to monitor the water quality, and the user been notified when the pH and temperature levels exceed the threshold. This project also makes use of additional software and hardware that is part of the system.

5.2 Development Environment Setup

A development environment is a set of methods and tools used to create, test, and debug a program or application. By setting up the development environment is to allow an environment in which any changes to the system are being applied. Development environment setup cover of Raspberry Pi environment, Desktop environment, and Water Quality Monitoring System environment.

5.2.1 Raspberry Pi Environment

To begin the system building, the Raspberry Pi is one of the most important components to be setup. This section will go over the Raspberry Pi environment settings in detail. The Raspberry Pi is the most important component of this project since it allows everything to work. It serves as a microprocessor to ensure that the water quality monitoring system runs smoothly and efficiently. Therefore, to make the Raspberry Pi completely functional to use, Raspberry Pi development environment would need to be setup by following a few simple steps.

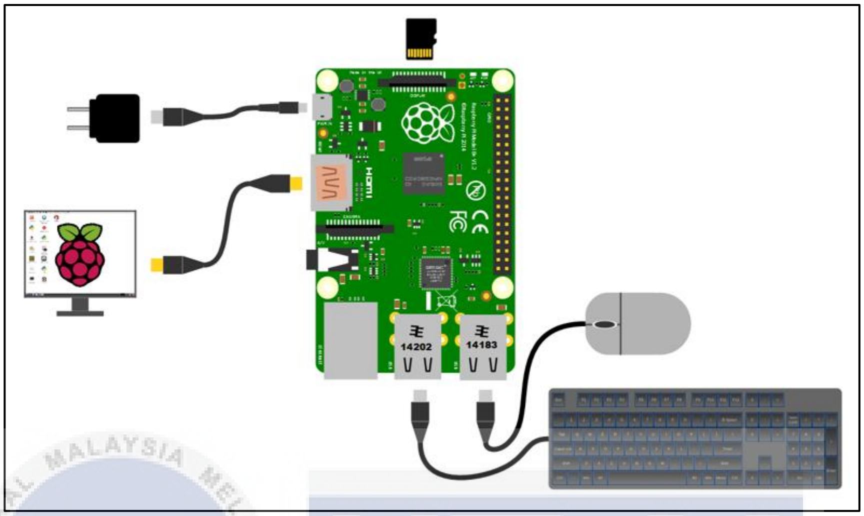


Figure 5.1: Raspberry Pi Development Environment Setup

Based on Figure 5.1, shows the development environment setup for the Raspberry Pi. The Raspberry Pi has an HDMI connector built in, so it able to connect to any monitor or TV that has an HDMI port for providing the display of the Raspberry Pi interface. Besides, to powered up the Raspberry Pi is by using the standard USB charging cable on the micro-USB port. Keyboard and mouse are needed for the initial set-up and capability of control to the RPI. MicroSD card is required to stored Operating System and the working files for the use of the system. On the other hand, other easy ways to access and use RPI is by remotely access using the VNC viewer via the desktop, whereas the VNC Server is built into Raspberry Pi OS.

5.2.2 Desktop Environment

Desktop computers work as a main development environment in developing the water quality monitoring system. Any requirement to access the RPI using VNC remote method and development of the mobile apps are done by using this desktop. Based on Table 5.1 below shows the specification of the desktop computer been used in this project.

Table 5.1: Desktop Specification

Item	**Value**
OS	Microsoft Windows 10
Model	Acer Swift 1
Processor	Intel Pentium CPU N4200 @ 1.10GHz
RAM	4.00 GB
ROM	128.00 GB

5.2.3 Water Quality Monitoring System Environment

To form a complex and well -functioning system, some components need to be setup and connected. Water Quality Monitoring System is a system that are formed from several components among which are Raspberry Pi, E-201-C pH sensor, DS18B20 temperature sensor, MCP3008, breadboard, jumper wire, and transistor.

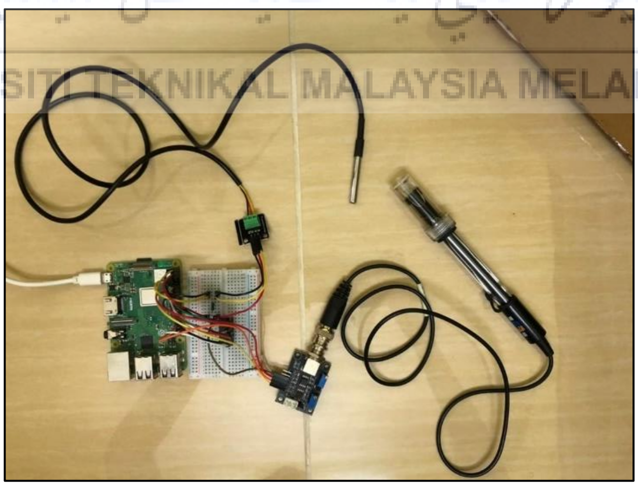


Figure 5.2: System Development Environment Setup

Based on Figure 5.2 above shows the environment setup of the Water Quality Monitoring System. This setup is not too difficult, nor too easy to be developed. However, the connection on the part of the pH sensor and the Temperature sensor requires good scrutiny. All the connection requires a connection between the sensors and the Raspberry Pi GPIO pins. Previously in Chapter 4, Figure 4.15 and Figure 4.16 already illustrate the circuit diagram for the connection. Based on Table 5.2 below, shows the details of the connection between several components to the RPI as the main component using the jumper wire. It is advisory best practice to use the same color of jumper wire on the same connection function to avoid the confusion.

Table 5.2: System Component Connection

Component	**Connection Details**
pH sensor	▪ Both GND pins connect to RPI Ground ▪ VCC pins connect to RPI 5V power ▪ PO pins connect to MCP3008 CH1
Temperature sensor	▪ DAT pins connect to RPI GPIO 4 (GPCLK0) ▪ VCC pins connect to RPI 5V power ▪ GND pins connect to RPI Ground
MCP3008	▪ V_{DD} pins connect to RPI 3V3 power ▪ V_{REF} pins connect RPI 3V3 power ▪ AGND pins connect to RPI Ground ▪ CLK pins connect to RPI GPIO 18 (PCM_CLK) ▪ D_{OUT} pins connect to RPI GPIO 23 ▪ D_{IN} pins connect to RPI GPIO 24 ▪ CS/SHDN pins connect to RPI GPIO25 ▪ DGND pins connect to RPI Ground ▪ CH1 pins connect to pH sensor PO

5.3 Software Configuration Management

Software configuration management is the process of discovering and defining the software configuration elements in the system. Configuration management, when effectively done, guarantees that the technological assets are setup correctly and that they are related to one another.

5.3.1 Configuration Environment Setup

Configuration Environment Setup cover the configuration in the Raspberry Pi, Desktop, and the Water Quality Monitoring System itself.

5.3.1.1 Raspberry Pi Configuration

The first and most important thing to do is make sure that the Raspberry Pi is operated with Rasbian OS. Latest Rasbian OS will been download and write into the MicroSD card. This MicroSD card will then be inserted into the Raspberry Pi and been power up. The Raspbian OS will boot straight to the desktop, and the username *pi* and password *raspberry* will been automatically set as the default credential.

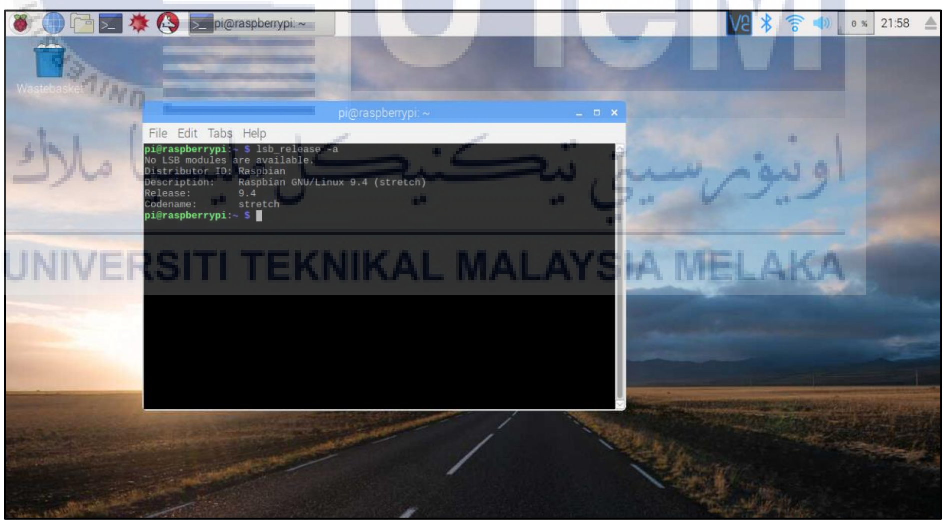


Figure 5.3: Raspbian OS

Figure 5.3 shows the installed Raspbian OS in the Raspberry Pi which indicate that the version used in this project is the Raspbian GNU/Linux 9.4 version.

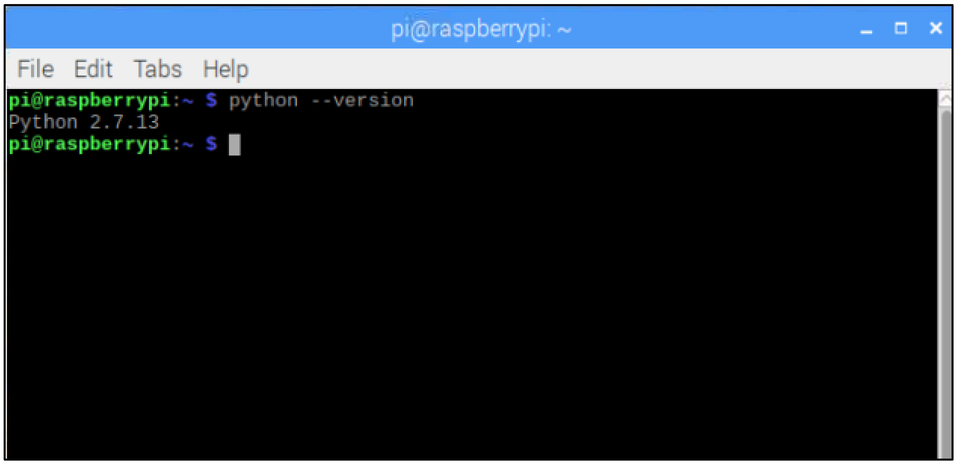


Figure 5.4: Python Version

Based on Figure 5.4, shows the python version in the Raspberry Pi which is Python 2.7.13 version. By default, Python 2 is already installed on the Raspbian OS. Basically, Python will be used to write and execute the code on the Raspberry Pi especially for the use of pH sensor and temperature sensor functions.

5.3.1.2 Desktop Configuration

Thingspeak and Kodular are web-based platform that will been used as part of the tools used in this project. Thingspeak act as an analytic platform to retrieve the data from Raspberry Pi and then generate the plotted graph. Whereas Kodular is an online tool to develop the android mobile apps in this project. Both web-based platform will only been access and develop by using the browser via desktop.

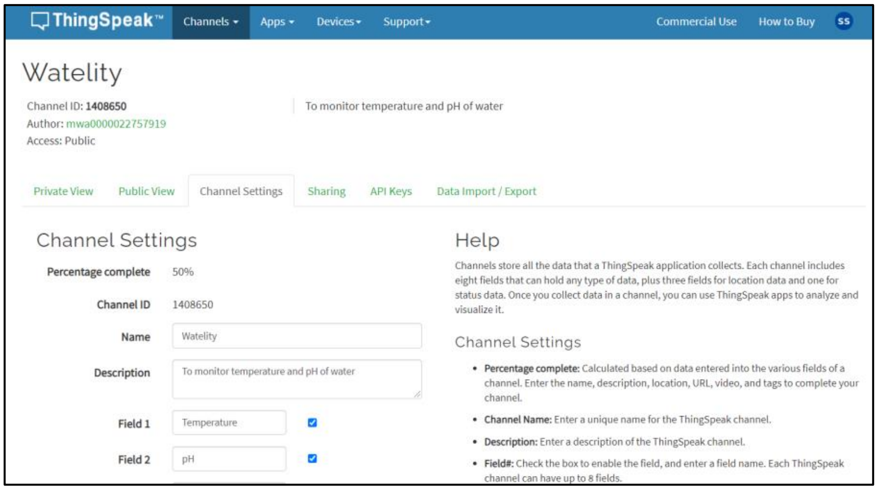


Figure 5.5: Thingspeak Channel

Based on Figure 5.5, shows the created Thingspeak account channel to retrieve both data value of pH and temperature from the Raspberry Pi. These channels will be detected and authorized using the API Keys that will be configured through the sensor code in the Raspberry Pi.

Figure 5.6: Kodular Project

Figure 5.6 shows the created Kodular project to develop the android mobile apps in monitoring the water quality. It is compulsory to having an account first before proceeding with the mobile app's development. The design work will be done in this section, it requires some skill and focus in visualizing the interface design for these apps.

User-friendly, consistency and simple interface should be considered in designing the user interface to make users feel more comfortable and able to get things done quickly.

5.3.1.3 Water Quality Monitoring System Configuration

- **Sensors Coding**

  To enable the pH sensor and temperature sensor to function, Raspberry Pi need to be configured with the sensors code. Based on Figure 5.7, shows the sensors code that have been write in Raspberry Pi. This code is intended to make sensors read water data readings as well as make a few calculations before the data is sent to the Thingspeak database using the Thingspeak Write Key that already been set in the code. The code can be run by simply enter '*sudo python watelity3.py'* through the Raspberry Pi terminal.

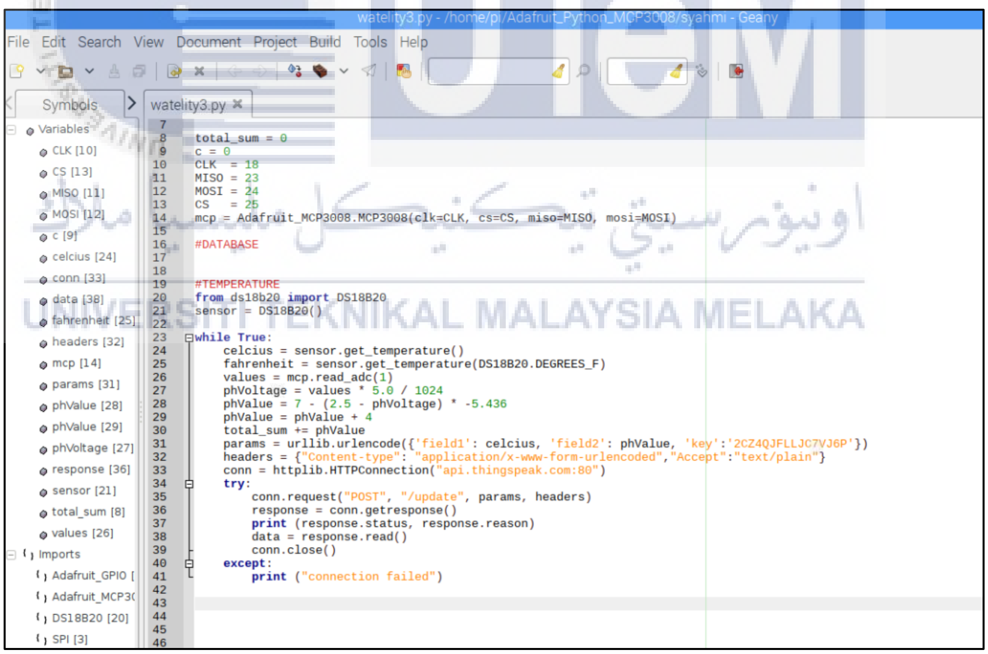


Figure 5.7: Sensors Code

- **Main Dashboard Output**

  Main dashboards play a vital part in this project to provide the users with monitoring of water parameter data. Therefore, in this section, an interface has been designed which then the blocks are built to collect the data from ThingSpeak.

Figure 5.8 below shows the blocks to retrieve the pH and temperature value from ThingSpeak by using the ThingSpeak URL and ThingSpeak Read Key. This is intended for an identification and authorization process before data can be retrieved.

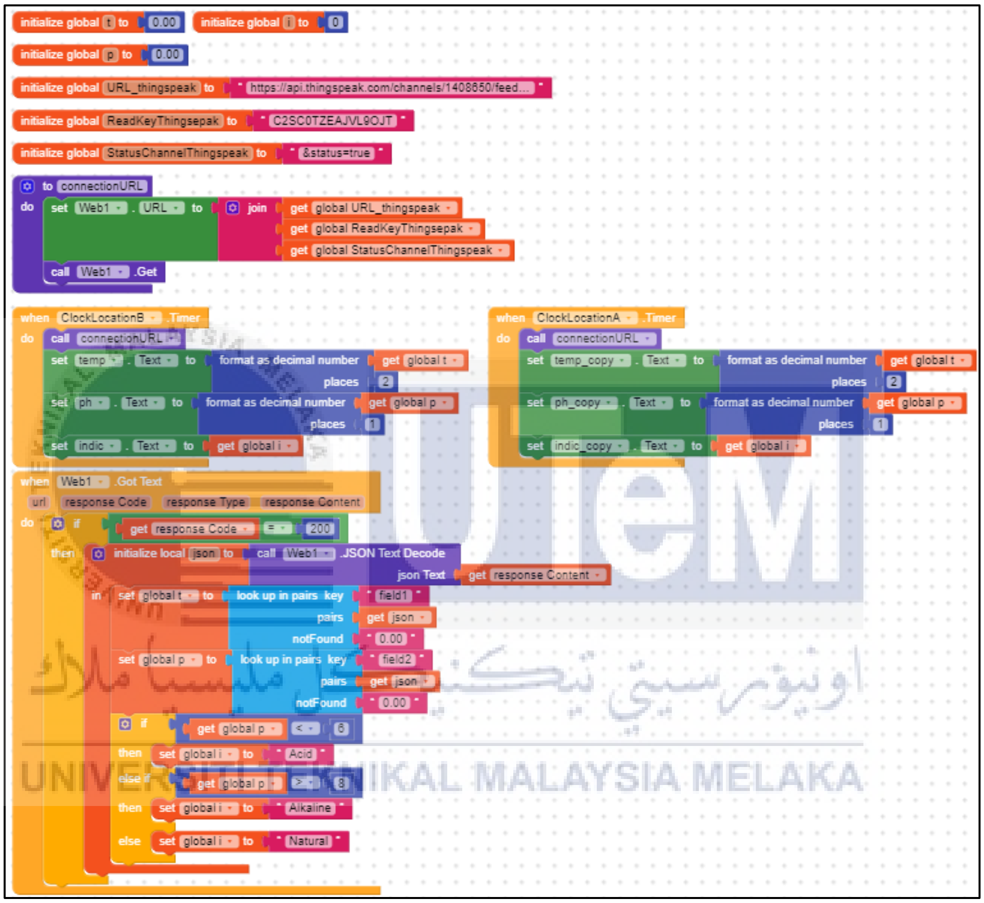


Figure 5.8: Main Dashboard Blocks

Based on Figure 5.9 shows the example of output value on the main dashboard after been run on the real android device. The pH indicator automatically been assigned by considering the if else condition statement from the blocks that have been built.

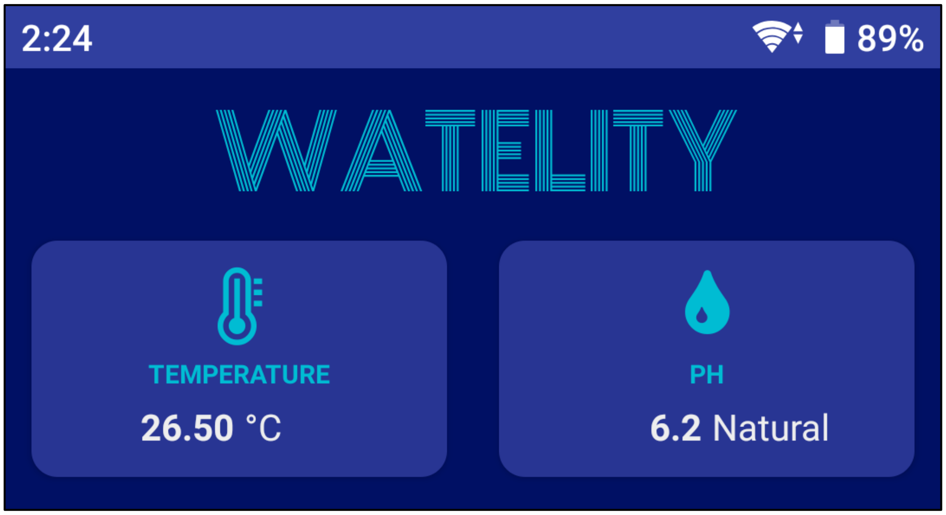


Figure 5.9: Main Dashboard Output

- **Alert Notification Function**

  As this project objective to warn fish breeder about the water changes that are not suitable for the fishpond environment. Alert notification function can assist the users to been notified once the water quality exceeds the level of action. Based on Figure 5.10 below, shows the alert function blocks that been build up in the Kodular. To be simplify, the blocks will been set with the level of action value that already been considered by referring the Chapter 2 Literature Review, this blocks values will be compared with the received values from the sensors. Once values from the sensors exceed the blocks values, it will trigger a notification according to the pH perspective or the temperature perspective. Once the notification triggered, the time interval will automatically been sign up intended for the next notification inspection.

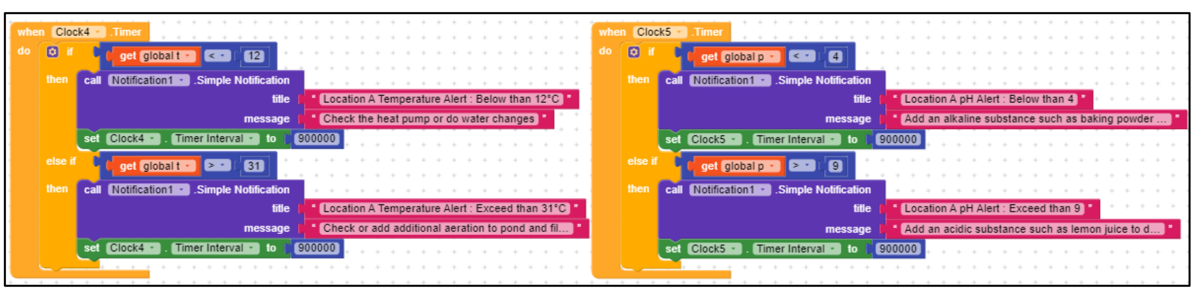


Figure 5.10: Alert Function Blocks

Based on Figure 5.11, shows the example of alert notification function. The notification title and message been showed in the bar notification of the phone according to the blocks that have been build. The simple action been suggested within the notification to provide the fish breeders some idea in overcoming the issues.

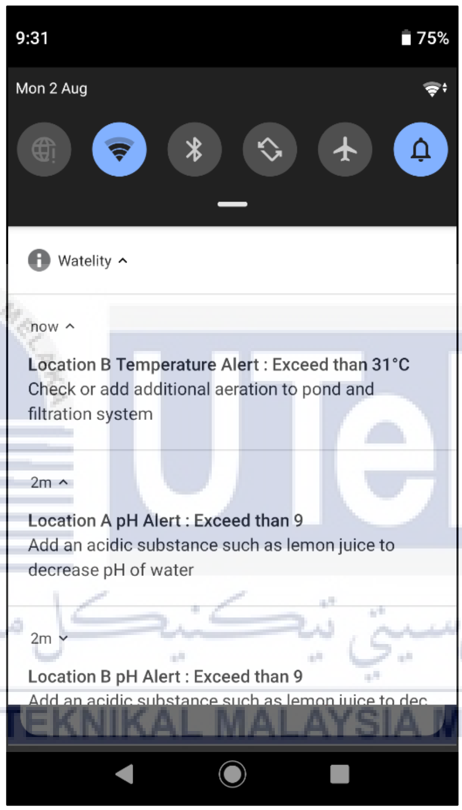


Figure 5.11: Alert Function Example

- **Multiple Location Dashboard**

To monitor multiple fishpond locations, the mobile app interface needs to be designed with options that allow users to switch dashboards between locations. In Figure 5.12, spinner component been used in the Kodular to provide the dropdown function in enabling the users to switch the dashboard. This dashboard will provide users the selected location interface from the dropdown option.

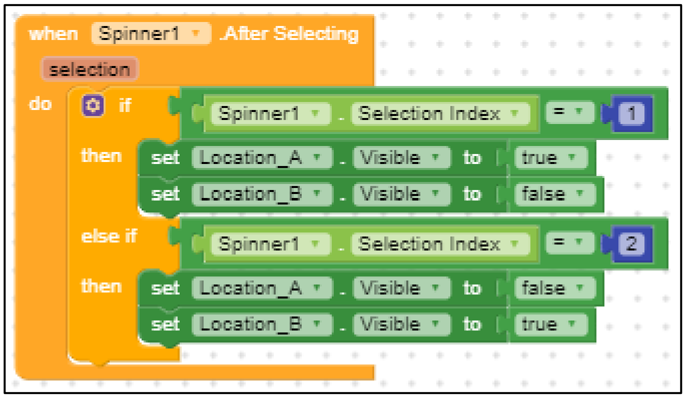


Figure 5.12: Location Spinner Function Blocks

In Figure 5.13, the dropdown option of the location will allow the users in switching their fishpond monitered dashboard. Logically, this is necessary because most fish breeder are not only focused on one fishpond, but also several other fishponds to be monitored.

Figure 5.13: Location Dropdown Function

- **Graph Function**

Graph is intended to show the analyzed and summary of the data within a certain period of time. ThingSpeak web-based platform provide the plotted graphs according to the received data value from the sensors. These plotted graphs can be used for use in the mobile apps. On the other hand, these graphs also can be plotted in the specific period such as the daily average values by changing in the graph setting. In Figure 5.14, highlight the share button that contain the graph URL to be used in the Kodular using the Web Viewer component.

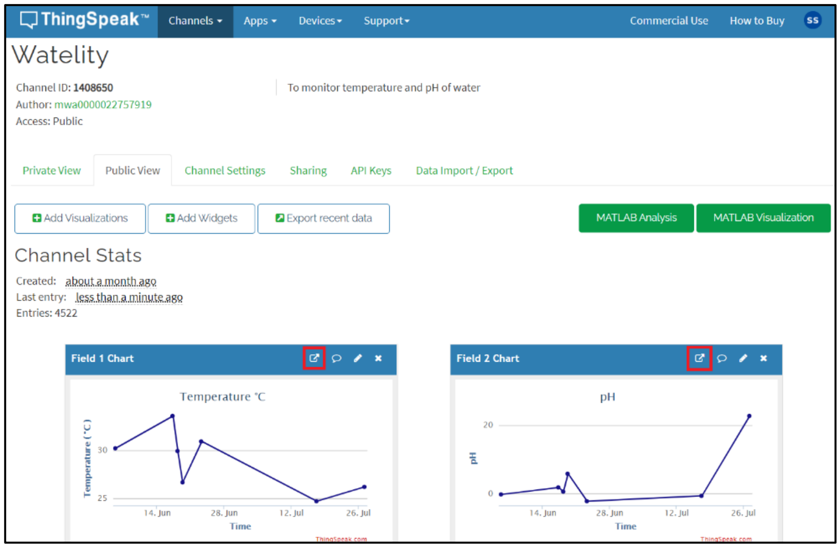


Figure 5.14: *ThingSpeak* Graph

Based on Figure 5.15, shows the example of the setting in the web viewer component. By using the web viewer component, can be used to display the ThingSpeak graph in the mobile apps. The previous ThingSpeak graph URL will be assigned in the Home Url setting.

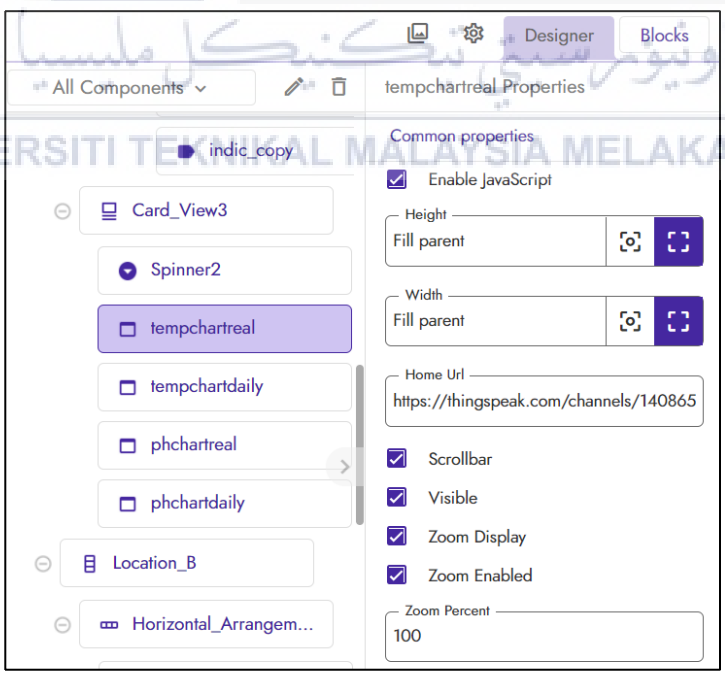


Figure 5.15: Web Viewer Component Setting

Figure 5.16 shows the example of the graph display output in the mobile apps. Simply put, there is no difference between displays on the web and in mobile apps. If any changes are to be made to the graph, it can be made directly through ThingSpeak platform and the reconfiguration needs to be done again.

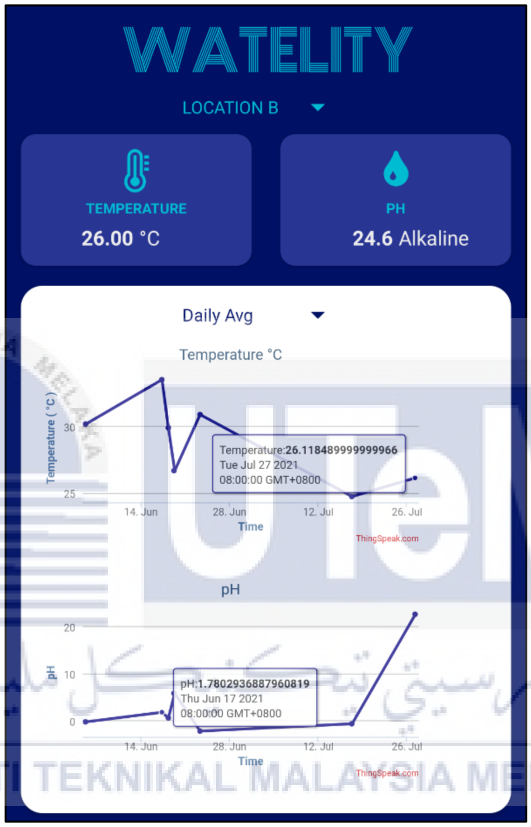


Figure 5.16: Daily Average Graph

5.4 Implementation Status

In this section, the progress of the development status for each of the module been described as the Table 5.3 below.

Table 5.3: Implementation Status Details

No.	Module	Description	Duration	Date Completed
1	Assemble Hardware	Collecting all the components and hardware for the project.	7 Days	24 April 2021

2	Building the Prototype	All components successfully been connected to breadboard and Raspberry Pi.	7 Days	10 May 2021
3	Software Configuration & Sensors Code Implementation	Sensor’s code configured using necessary libraries and connect with the ThingSpeak.	14 Days	24 May 2021
4	Mobile Apps Development	Simple interface mobile apps developed successfully connected and received the values from Raspberry Pi.	7 Days	30 May 2021
5	Alert Notification Function	Alert notification function block been built, and the bar notification successfully display on the phone.	7 Days	20 July 2021
6	Real-time & Daily Average Graph	Setting up the graphs setting in ThingSpeak and Kodular to be display in the mobile apps.	7 Days	22 July 2021
7	Multiple Location Dashboard	Designing and built up the blocks for the dropdown function to switch the different locations dashboard.	7 Days	24 July 2021

5.5 Conclusion

For the conclusion, the implementation phase demonstrates how the project development process works in detail. The procedures and activities involved in this project are broken down step by step to ensure that the project runs smoothly and efficiently. Furthermore, in order to be more successful, the environment and configuration setup must be carefully designed at this phase. The outcome from this chapter is obtained and used in preparation for testing and analysis in the next chapter.

CHAPTER 6: TESTING

6.1 Introduction

A brief discussion on project implementation was held in the previous chapter. After the installation is complete, the testing and evaluation of the project will continue. This is necessary to guarantee that the project is completed, and the objectives of water quality monitoring are met. Several test plans will be explained in this chapter, such as Test Plan, Test Strategy, Test Design, and Test Results Analysis. The verification method is critical to ensuring that the finished product fulfills the required standards and functions properly while furthering the project's aim of achieving the project objectives.

6.2 Test Plan

This section will describe the basis of each system test. It involves the project's scope, including the organization of the test, the environment of the test and the test schedule.

6.2.1 Test Organization

Testing Organization states about the individuals involved when the test is performed. For this project, the testing process will only involve two individuals: the system developers and the user. System developer is the person responsible for developing and providing functions system, in addition to deciding which test subjects to use and test cases to employ to find any problems in the system. The user will be any adult to do the system testing that can provide insights and opinions for any improvements of the system.

6.2.2 Test Environment

Testing location for this project is carried out at home using several different water samples. These different water samples represent water solutions with different pH and temperatures values. This location environment has an internet connection and power source. Raspberry Pi uses Wi-Fi technology that requires an internet connection for the data transmission process and the power source is needed to power up the Raspberry Pi. In addition, an android smartphone is required to perform water quality monitoring testing through mobile apps that have been developed. Tools such as pH meter and temperature meter are also needed to represent the manual method to make comparisons with the system that has been implemented.

6.2.3 Test Schedule

In general, the tests must be scheduled in such a way that the best results are obtained for each testing process. It also has the advantage of making the testing more methodical. If the tests are done in a timely manner, the results will be more accurate. The test is scheduled for two weeks to enable the testing process performed in a more flexible way. In this project, testing is dynamically conducted on Week 4 and Week 5 of FYP 2. 10 sensor readings from the testing be taken and recorded for comparison with the manual method. If any faults are discovered throughout the testing process, the process will return to the implementation phase to identify and remedy the problem. This method will be repeated until the desired outcome is achieved.

6.3 Test Strategy

Test Strategy explained the strategy selected to test the system such as bottom-up or top-down and black-box/white-box classes of tests. Black box testing strategy is used and been selected in this project. Testing the system with no prior knowledge of its internal workings is known as Black Box testing.

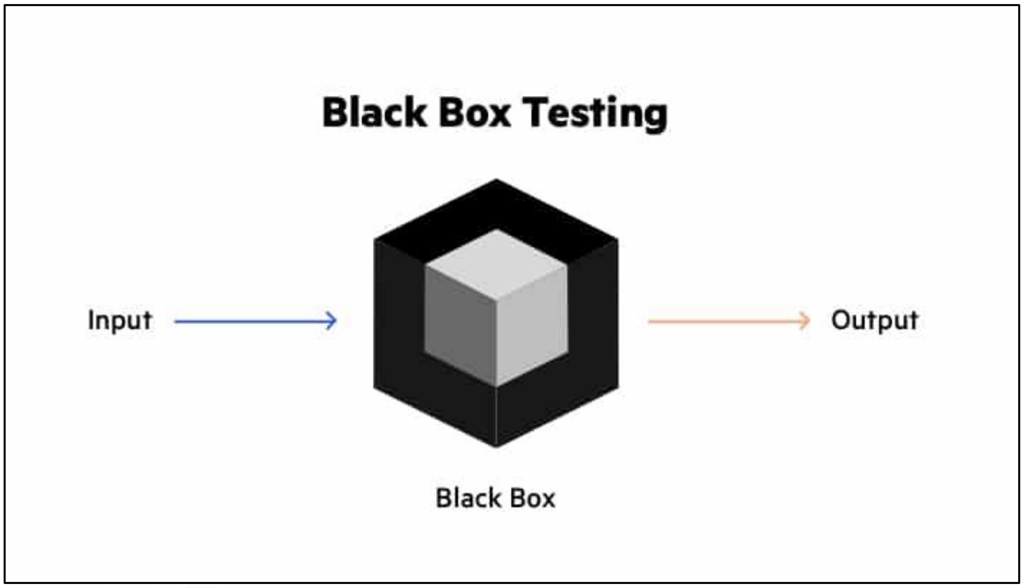


Figure 6.1: Black Box Testing

Figure 6.1 shows the illustration cycles of Black Box testing. Tester gives the system under test an input and watches the output it produces. This allows the system to react to expected and unexpected user activities and the system reaction time, usability difficulties, and reliability concerns. This is a powerful testing approach as it puts the system through its steps from start to finish. Tester able to mimic user activity and see if the system lives up to its promises. End users did not know how the system is built or constructed and hoped to obtain an acceptable answer to their request. In addition, the accuracy of the sensor readings is one of the things to be achieved in this test. Therefore, 10 sensor readings will be selected based on repeated readings and nearest to the manual method readings.

6.3.1 Classes of Tests

Classes of test include any involved tests in this project such as output correctness or Functionality test, Security test, Stress test, etc. Due to the selected test strategy is Black Box testing, the functionality test will be applied. The functionality test ensures that the Raspberry Pi and the sensors, which include a pH sensor and a temperature sensor, are fully functioning. Testing began when the Raspberry Pi receives data from the sensors, it sends it to the developed mobile apps. Besides, the mobile apps functionality such as alert notification and graph display also be tested in the functionality test.

6.4 Test Design

Test Design is the process of evaluating the correctness and reliability of the system components or modules. Each component must be examined and the comparison tests between the system and manual method performed to validate the system. This testing is required to confirm that the system meets the requirements and functions correctly.

6.4.1 Test Description

The project test cases and expected outcomes are created and documented in this part. All tests assigned with Test Case ID for structured documentation and reference purposes.

6.4.1.1 Connectivity Test between Raspberry Pi and Desktop

Table 6.1: Connectivity of Raspberry Pi and Desktop

Test Case ID	**TC01**
Test Functionality	Test the connectivity between Raspberry Pi and Desktop
Prerequisites	▪ Both are power on and connect within the same network ▪ Desktop installed with VNC Viewer ▪ Identify the IP address of Raspberry Pi through local network
Execution Steps	1. Launch VNC Viewer on desktop. 2. Enter the VNC Server address of Raspberry Pi. 3. Enter username and password of Raspberry Pi.
Expected Result	Desktop access the main display of Raspberry Pi via VNC Viewer

6.4.1.2 Connectivity Test between System Component and ThingSpeak

Table 6.2: Connectivity of the System Component and ThingSpeak

Test Case ID	**TC02**
Test Functionality	Test the connectivity between the System Component and ThingSpeak
Prerequisites	▪ All component of the system includes the sensors is connected ▪ Raspberry Pi is power on ▪ Sensors Source code contain with ThingSpeak Write API Key
Execution Steps	1. Launch terminal in the Raspberry Pi. 2. Change the working directory to the source code file directory in the terminal. 3. Run the source code file using *sudo python* command.
Expected Result	Raspberry Pi terminal produce 'OK' statement indicate successfully connect and transmit data to ThingSpeak

6.4.1.3 Sensors Reading and Display Test

Table 6.3: Sensors Reading and Display Test

Test Case ID	**TC03**
Test Functionality	Test the sensors reading and display the data through mobile apps
Prerequisites	▪ Source code is executed in Raspberry Pi ▪ Android smartphone already installed with mobile apps APK
Execution Steps	1. Download the APK file from Kodular into the android smartphone. 2. Install the APK file of the developed mobile apps. 3. Run the mobile apps.

Expected Result	Mobile apps display pH and temperature data seamlessly and with near -accurate readings

6.4.1.4 Alert Notification Test

Table 6.4: Alert Notification Test

Test Case ID	**TC04**
Test Functionality	Test the alert notification generate by the mobile apps
Prerequisites	▪ Source code is executed in Raspberry Pi ▪ Android smartphone installed with mobile apps APK ▪ Prepare cold/hot water or high acidic/alkaline solution
Execution Steps	1. Run the mobile apps in android smartphone. 2. Soak the sensors into cold/hot water or high acidic/alkaline solution
Expected Result	Mobile apps triggered the alert notification

6.4.1.5 Graph Display Test

Table 6.5: Graph Display Test

Test Case ID	**TC05**
Test Functionality	Test the real-time and daily average graph generate by ThingSpeak and display in mobile apps
Prerequisites	▪ Source code is executed in Raspberry Pi ▪ Setting up the graphs setting in ThingSpeak and Kodular ▪ Android smartphone installed with mobile apps APK
Execution Steps	1. Run the mobile apps in android smartphone. 2. Tap the dropdown option to switch the real-time and daily average graph display.
Expected Result	Mobile apps display the real-time and daily average graph generate by ThingSpeak

6.5 Test Results and Analysis

6.5.1 Test Case 01

Table 6.6: Test Case 01 Result

Test Case Identification	Tester Identification	Results (Success/Fail)
Connectivity between Raspberry Pi and Desktop	▪ Raspberry Pi red power LED lights up ▪ VNC Viewer access the display of Raspberry Pi	Success

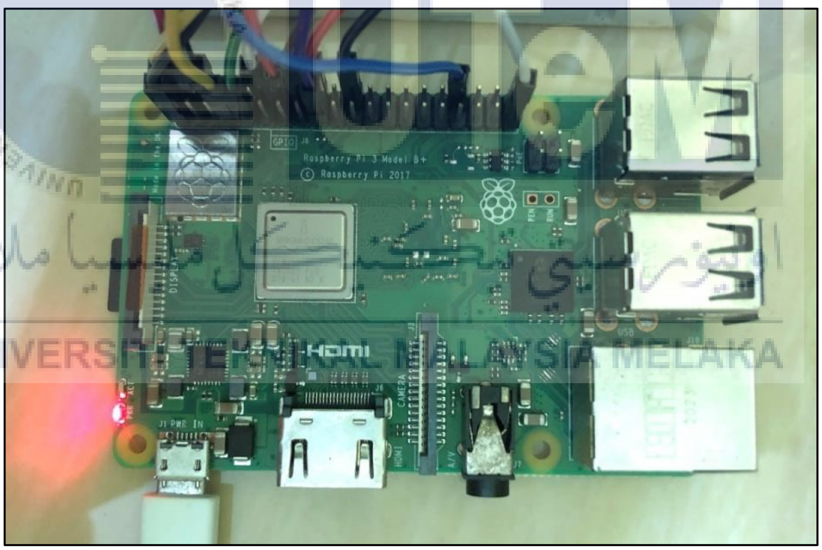


Figure 6.2: Raspberry Pi LED Light

Based on Table 6.6, shows the connectivity test between the Raspberry Pi and the Desktop that resulted in a successful test. Figure 6.2 shows the Raspberry Pi red power LED lights up, this indicates that it has an active power supply and there are absolutely no problems with the Raspberry Pi.

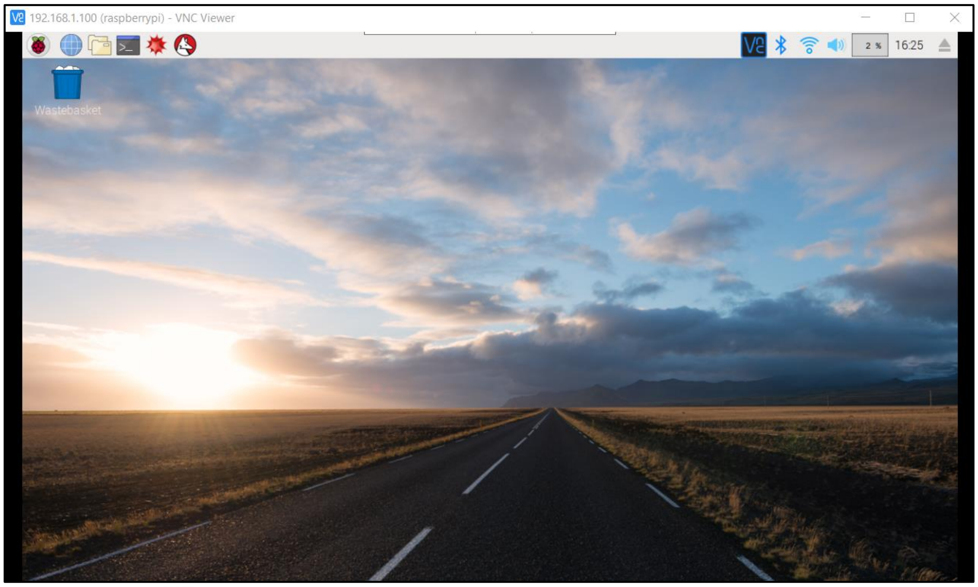


Figure 6.3: Raspberry Pi Display

Figure 6.3 shows the VNC Viewer in the Desktop able to access the display interface of Raspberry Pi, this indicates that the Raspberry Pi successfully connected with the Desktop and ready to be used.

6.5.2 Test Case 02

Table 6.7: Test Case 02 Result

Test Case Identification	Tester Identification	Results (Success/Fail)
Connectivity between the System Component and ThingSpeak	▪ Raspberry Pi terminal produce 'OK' statement	Success

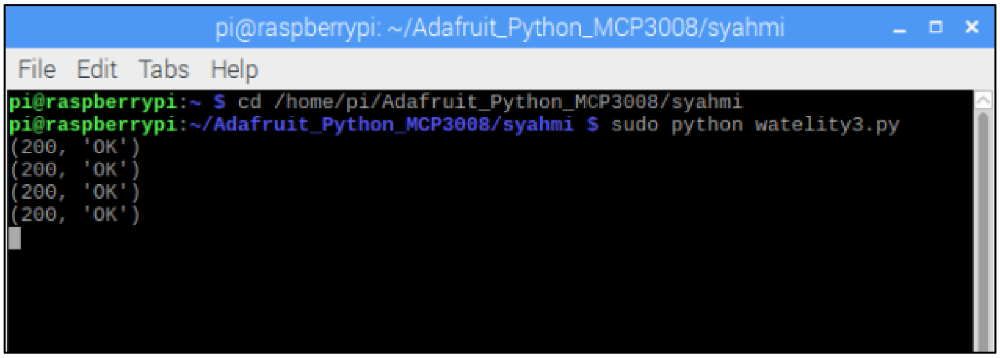


Figure 6.4: Raspberry Pi Terminal

Table 6.7 shows the connectivity test between the System Component and ThingSpeak that resulted in a successful test. Figure 6.4 above, shows the terminal of Raspberry Pi that run the source code file using *sudo python* command. The Raspberry Pi then successfully connect and transmit data to ThingSpeak by produce the 'OK' statement.

6.5.3 Test Case 03

Table 6.8: Test Case 03 Result

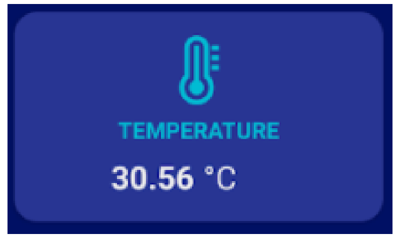
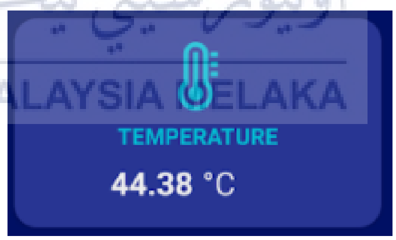
Test Case Identification	**Tester Identification**	**Results (Success/Fail)**
Cold Water Test	▪ The low value of water temperature display in the mobile apps	Success
Room Temperature Water Test	▪ The moderate value of water temperature display in the mobile apps	Success
Hot Water Test	▪ The high value of water temperature display in the mobile apps	Success
Neutral Solution Test	▪ The value of pH and indicator for natural solution display in the mobile apps	1st test – Fail 2nd test – Success

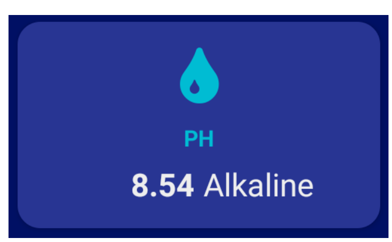
Acidic Solution Test	▪ The value of pH and indicator for acidic solution display in the mobile apps	1st & 2nd test – Fail 3rd test – Success
Alkaline Solution Test	▪ The value of pH and indicator for alkaline solution display in the mobile apps	1st test – Fail 2nd test – Success

Table 6.8 shows the sensors reading and data display through mobile apps that resulted in a successful test. Temperature sensor has directly successful results within the cold and hot water whereas the pH sensor has an inconsistent result to neutral, acidic, and alkaline solutions. Several revisions and improvements have been attempted especially on the source code and physical parts of the pH sensor. However, these unstable results are likely due to the degradation in the quality of sensor components.

Table 6.9: Manual Method & Developed System Comparison

Test Case Identification	**Manual Method**	**Developed System**	
Cold Water Test	21°C 	21.06°C 	10 Reading
			21.56
			21.25
			21.12
			21.06
			21.12
			21.19
			21.25
			21.31
			21.38

			21.44
Room Temperature Water Test	32°C 	30.56°C  	10 Reading 30.62 30.56 30.50 30.56 30.50 30.44 30.50 30.56 30.50 30.56
Hot Water Test	45°C 	44.38°C  	10 Reading 44.61 44.62 44.50 44.38 44.31 44.12 44.11 44.38 44.35 44.31
Neutral Solution Test	7.03	7.21	10 Reading 7.39

			7.40 7.38 7.21 7.21 7.21 7.20 7.21 7.21 7.22
Acidic Solution Test	2.03 	2.33 	10 Reading 2.47 2.45 2.38 2.33 2.33 2.35 2.33 2.34 2.32 2.32
Alkaline Solution Test	8.69	8.54 	10 Reading 9.00 8.54 8.54 8.50 8.50 8.54

			8.61
			8.59
			8.54
			8.42

As detailed in Table 6.9, the comparison results between the manual method and the developed system are shown. The manual method is represented using pH meter and temperature meter. The test performed several times which almost exceeded 10 readings, repeated good readings and close to the manual readings were recorded directly as stated. As shown, the cold-water test yields 21.0°C on manual methods and 21.06°C on developed systems. Result for room temperature water shows a difference of 1.44°C where 32°C on manual method and 30.56°C on developed system. Hot water results show a difference of 0.62°C where 45°C on manual method and 44.38°C on developed system.

Testing the pH sensor readings, on the other hand, gives a different experience. Testing was performed over 10 trials to obtain an acceptable reading. Neutral solution results the difference of 0.18 with 7.03 for the manual method and 7.21 for the developed system. Result for acidic solution shows a difference of 0.30 with 2.03 for manual method and 2.33 in the developed system. Moreover, the manual method yielded an alkaline solution of 8.69, whereas the developed system yielded an alkaline solution of 8.54, indicating a 0.15 difference between the two.

6.5.4 Test Case 04

Table 6.10: Test Case 04 Result

Test Case Identification	**Tester Identification**	**Results (Success/Fail)**
Alert Notification Test	▪ Mobile apps triggered the alert notification in the smartphone bar notification	Success

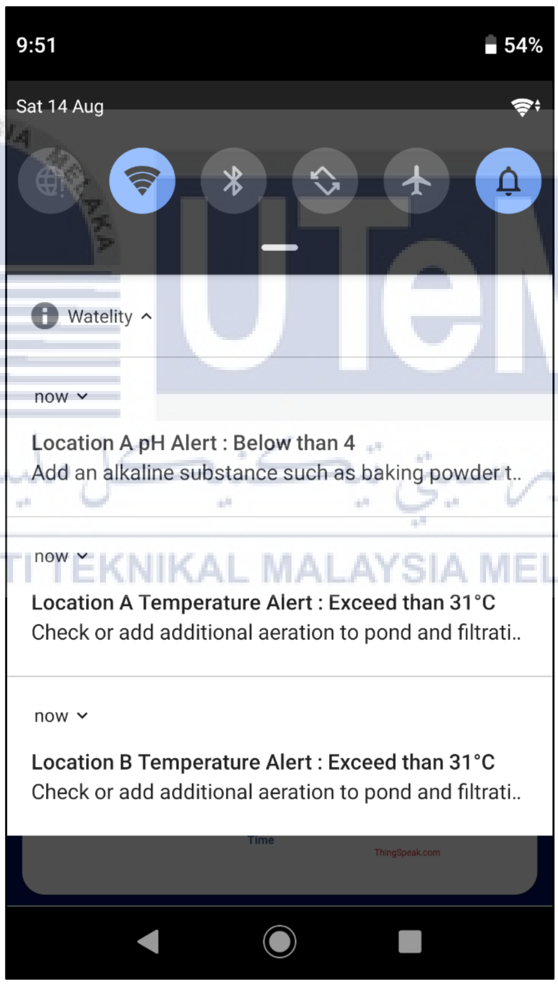


Figure 6.5: Alert Notification Function

Table 6.10 shows the alert notification function test on the developed system that resulted in a successful test. Figure 6.5 shows the display of the alert notification that been triggered by the system to be displayed on the bar notification of the android smartphone. The alert notification be triggered and responded immediately upon detection of abnormal pH and water temperature conditions. Its response rate takes less than 2 seconds after the abnormal condition of water is identified. It takes 15 minutes for the next alert notification to be displayed on the notification bar depending on whether the water condition is still abnormal, this is as has been set during the development of these mobile apps. If the water condition is good, the next warning notification will not appear until new abnormal circumstances are discovered.

6.5.5 Test Case 05

Table 6.11: Test Case 05 Result

Test Case Identification	**Tester Identification**	**Results (Success/Fail)**
Graph Display Test	▪ Mobile apps able to display the real-time and daily average graph generate by ThingSpeak	Success

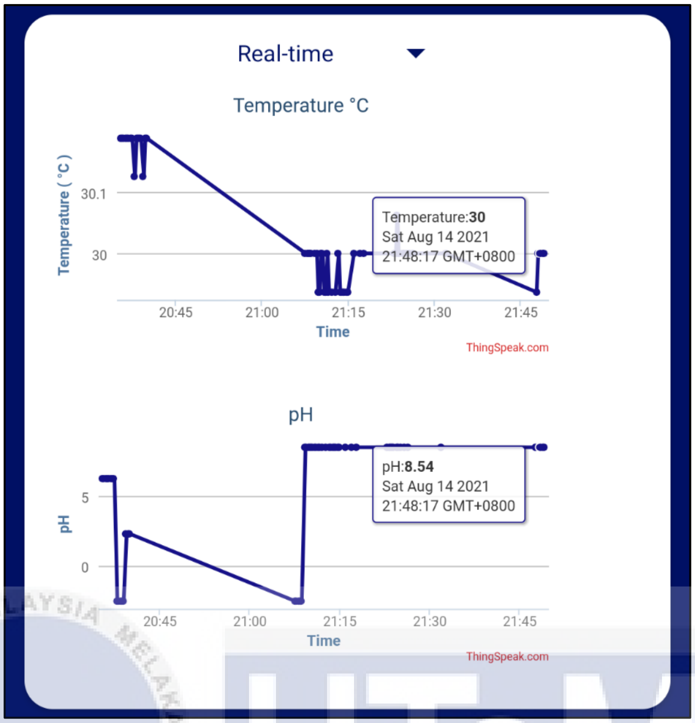


Figure 6.6: Real-time Graph Display

Table 6.11 shows the graph display function test on the developed system that resulted in a successful test. Figure 6.6 above, shows the example of the real-time graph display on the developed system. Significant changes are visible on the graph due to differences in water conditions detected by the sensors which are subsequently plotted on the graph. The change in water condition is due to tests performed where the sensors are placed in different solutions. In fact, ambient temperature is also a factor influencing the change of the graph plot as occurs in the water temperature graph. Users can monitor and view the pH value and temperature of water as well as see the pattern of changes in pH and temperature through the graph for the pass few minutes.

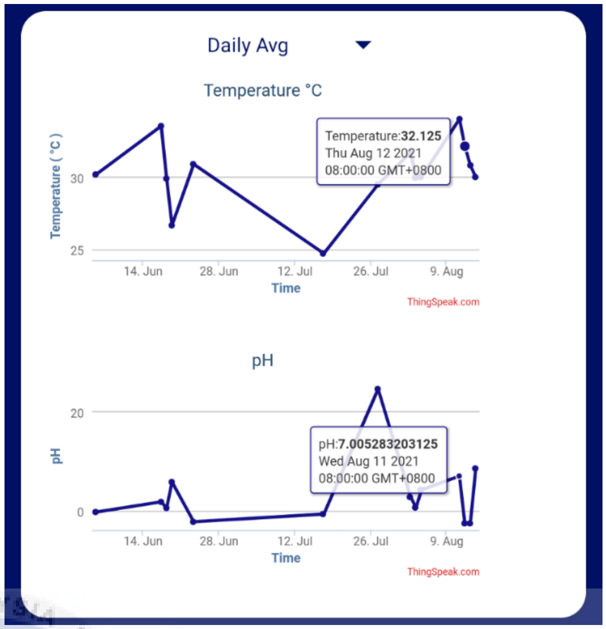


Figure 6.7: Daily Average Graph Display

Figure 6.7 shows the example of the daily average graph display on the developed system. The average value of the pH and temperature for each day are shown on the graph. The ups and downs plots on the graph are the result from ThingSpeak calculations to obtain daily average values for pH and temperature. These average values are plotted on the graph by date. This patterns of changes in pH and temperature values can be seen and compared over the past few days. Switching between the daily average graphs and real - time graphs also possible and works well in this system.

6.6 Conclusion

This chapter describes the project overall results. It may be concluded that the system capable of detecting water pH and temperature and distinguishing between normal and abnormal water conditions through the alert notification function. However, a few errors occur during testing, such as the pH value being unstable. This is likely due to the degradation in the quality and performance of the pH sensor components. Replacement of new components or using a better version of pH sensors may help to obtain more stable and accurate readings.

CHAPTER 7: PROJECT CONCLUSION

7.1 Introduction

The project progress and conclusion are summarized in this chapter. This chapter discusses the development and achievements of the project, as well as the contributions, limitations, and future improvements of the project. In addition, discussing the summarization of project help to clarify and understand all the specifics of the project. As the result, this chapter also discuss the possible upgrades and implementation to ensure project continuity in the future.

7.2 Project Summarization

This project is based on IoT which integrate with Raspberry Pi, pH sensor, and temperature sensor to create a water quality monitoring system. This device not only monitors water quality but also sends out alerts if a certain condition of water is recognized by an algorithm. Besides, this system designed to enhance water quality while also aiding in more effective fishpond water monitoring.

The first objective of this project is to identify water quality parameters for tilapia fish farming. This objective is achieved by finding a suitable water parameter to be used in this project. Temperature and pH are the water parameters focused on this project due to it play an important role in determining water quality for fish farming as discussed in Chapter 2 previously.

Second objective of this project is to design and develop IoT based monitoring system that able to monitor and alert water quality. The integration of the Raspberry Pi, pH Sensor, and Temperature Sensor assist in build the system that allows this water quality monitoring to be done. Water quality alert also be implemented through android mobile apps that have been built based on the Kodular framework. All of this has been achieved as detailed in Chapter 5 previously.

The last objective of this project is to test the effectiveness of the system functionality, response time, and accuracy. Once the system is successfully built, testing is carried out to see how well the system functionality, respond time, and accuracy can operate. This has been accomplished in Chapter 6 where all the tests can be executed well although there are some tests that are encountered with little difficulty.

Overall, this project accomplished all the project objectives. As a result, this initiative improves water quality monitoring and increase the fishpond water quality. Therefore, brief assessment about the weaknesses and strength of this project are stated as Table 7.1 below.

Table 7.1: Project Strength & Weaknesses

Project Strength	Project Weakness
▪ Mobile Apps Visualize Data ▪ Alert Notification ▪ Enhance Water Quality Monitoring Effectiveness	▪ Equipment Deterioration ▪ Financial Constraints

7.3 Project Contribution

This project is intended to assist the fish breeders in monitoring their fishpond water quality. It keeps track of the temperature and pH of the water. The monitoring of fishponds water quality through mobile apps is a good added value in the implementation of this project, the use of smartphones has become a daily routine for everyone. As a result, it is a better way to save the time and alert the fish breeders to take an action and avoid the worst-case scenario. Fish breeders no longer need to be around fishponds all the time and early signs of poor water quality in the fishpond can be identified. So, this is a facility that fish breeders can take advantage of to observe their fishponds.

7.4 Project Limitation

There are some limitations found in this project. The quality and type of sensor to some extent affect in the accuracy of the readings, better sensors are required and the price for these sensors costs more. In fact, most of these sensors are sold by suppliers from overseas, shipping costs should also need be considered. As a student, this factor becomes a major limitation.

In addition, existing sensors especially the pH sensor are among the constraints faced. This pH sensor is very sensitive and needs to be taken care of carefully. The deterioration of the quality of this sensor causes the functionality of the sensor to decrease. As a result, the accurate readings expected from the sensor become difficult. Besides, only pH and temperature are among the water parameters focused on this project. Thus, the more and varied water parameters are better.

7.5 Future Works

In the future, this project can be improved by using better sensors with more accuracy in obtaining the data. The sensor released by the supplier from DFRobot is better and it is more compatible for use with Arduino. This can help the data be obtained easily and provide the better results. Of course, it costs more, but the efficiency of the system can be improved.

In addition, system offerings for iOS smartphone users can be taken into account. Since application development using Kodular can only be offered to android users, other initiatives to develop apps for iOS users may can be considered in the future.

Furthermore, other water parameter monitoring options may be added to this system. Monitoring in terms of dissolved oxygen can be done with the addition of a dissolved oxygen sensor. This is because the water quality monitoring for fish farming can be improved due the dissolved oxygen in the water is one of the factors that affect the needs of fish life.

7.6 Conclusion

At the end of this project, the objectives stated in Chapter 1 were successfully achieved. Through mobile apps, the system can monitor water parameters such as temperature and pH while deliver alert notifications to the users for certain conditions. In the meantime, the system performance is reduced as a result of several weakness, which has an impact on the accuracy of the readings. To sum up, no perfect system can be developed; however, improvements can be made to achieve the level of perfection described.

REFERENCES

Atwood, H.L., Tomasso, J.R.,Webb, K., Gatlin, D.M., 2003. Low-temperature tolerance of Nile tilapia, Oreochromis niloticus: effects of environmental anddietary factors. Aquaculture Research 34, 241–251

Astro Awani. (2019, August 7). Ikan mati akibat sumber air tercemar, 17 penternak rugi lebih RM500,000. https://www.astroawani.com/berita-malaysia/ikan-mati-akibat-sumber-air-tercemar-17-penternak-rugi-lebih-rm500000-214425

Balon, B., & Simić, M. (2019, May). Using Raspberry Pi computers in education. In 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 671-676). IEEE.

Bhatnagar, A., & Devi, P. (2013). Water quality guidelines for the management of pond fish culture. International journal of environmental sciences, 3(6), 1980.

Chewy Editorial. (2021, January 21). The role of pH in the aquarium nitrogen cycle. BeChewy. https://be.chewy.com/the-role-of-ph-in-the-aquarium-nitrogen-cycle/

Colt, J., & Kroeger, E. (2013). Impact of aeration and alkalinity on the water quality and product quality of transported tilapia—a simulation study. Aquacultural engineering, 55, 46-58.

Components101. (2018, May 7). DS18B20 temperature sensor. https://components101.com/sensors/ds18b20-temperature-sensor

De Belen, M. C., & Cruz, F. R. G. (2017, December). Water quality parameter correlation in a controlled aquaculture environment. In 2017IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (pp. 1-4). IEEE.

EXISTEK. (2018, April 25). SDLC models explained: Agile, waterfall, V-shaped, iterative, spiral. Offshore Software Development Company | Offshore Software Development Services. https://existek.com/blog/sdlc-models/

Fadel, A. A., & Shujaa, M. I. (2020, November). Water Quality Monitoring System Based on IOT Platform. In IOP Conference Series: Materials Science and Engineering (Vol. 928, No. 3, p. 032054). IOP Publishing.

Khalil, I. M., & Abdulrazzak, H. N. (2019). Monitoring of Water Purification Process Based on IoT.

M. M, K. K. B M, R. Verma and D. Kiran, "Design and Development of Real-Time Water Quality Monitoring System," 2019 Global Conference for Advancement in Technology (GCAT), 2019, pp. 1-6, doi: 10.1109/GCAT47503.2019.8978414.

Muddassir, M., Noor, M. A., Ahmed, A., Aldosari, F., Waqas, M. A., Zia, M. A., ... & Jalip, M. W. (2019). Awareness and adoption level of fish farmers regarding recommended fish farming practices in Hafizabad, Pakistan. Journal of the Saudi Society of Agricultural Sciences, 18(1), 41-48

Nath, O. (2020) REVIEW ON RASPBERRY Pi 3b+ AND ITS SCOPE. International Journal of Engineering Applied Sciences and Technology, 2020 Vol. 4, Issue 9, ISSN No. 2455-2143, Pages 157-159

Nualgi America, Inc. (2020, February 17). Protect your pond's pH levels from the effects of (Acid) rain. Nualgi Ponds. https://nualgiponds.com/protect-ponds-ph-level-from-effects-of-acid-rain/

Popma, T., Masser, M., 1999. Tilapia–Life History and Biology. Southern Regional Aquaculture Center, SRAC Publication No. 283.

RASPBERRY PI FOUNDATION. (n.d.). Raspberry Pi 3 Model B+. https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/

Sengupta, B., Sawant, S., Dhanawade, M., Bhosale, S., & Anushree, M. (2019). Water Quality Monitoring using IoT. Int. Res. J. Eng. Technol., 6, 695-701.

ThingSpeak. (2021, March 1). The Things Industries. https://www.thethingsindustries.com/docs/integrations/cloud-integrations/thingspeak/

Xiong, F. (2015, June). Wireless temperature sensor network based on DS18B20, CC2420, MCU AT89S52. In 2015 IEEE International Conference on Communication Software and Networks (ICCSN) (pp. 294-298). IEEE.

APPENDIX A – APPS DEVELOPMENT FULL BLOCK