Development of IoT-Based Agriculture Monitoring Framework using Node-RED

By

Foong Theen Wui

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)
COMMUNICATIONS AND NETWORKING
Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2023

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Development of IoT-Based Agriculture Monitoring Framework by using Node -RED

Academic Session: JAN 2023

I FOONG THEEN WUI
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:
290, Persiaran Tawas,
Kampung Tawas, 30010
Ipoh, Perak.

Goh Hock Guan
Supervisor's name

Date: 24/04/2023

Date: 26/4/2023

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

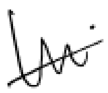
Date: 24/04/2023

SUBMISSION OF FINAL YEAR PROJECT

It is hereby certified that FOONG THEEN WUI (ID No: 20ACB04334) has completed this final year project entitled "Development of IoT-Based Agriculture Monitoring Framework using Node-RED" under the supervision of Ts Dr. Goh Hock Guan (Supervisor) from the Department of Computer and Communication Technology, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,



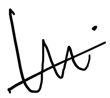
____

(*FOONG THEEN WUI*)

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**Development of IoT-Based Agriculture Monitoring Framework using Node-RED**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : FOONG THEEN WUI

Date : 24/04/2023

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Ts Dr. Goh Hock Guan, who has given me this bright opportunity to participate in this project. He gave a lot of insightful comments and suggestions which helps me during the process of the development. A million thanks to you.

Besides, my appreciation also goes to my friends for their patience and standing by my side during hard times. Finally, I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

In this era of globalization, the implementation of internet technology has been adapted in majority of the sectors. By adapting it in our daily task, it is making the execution process of our task more convenient and efficiently. Recently, the food security problem has become one of the main issues since the war that happening in some places and the limited amount of land for agriculture in the country. The production of the crops in our country is affected seriously and the peoples had to buy the vegetables with a higher price. Since agriculture sector is one of the main contributors to the GDPs of our country, we need to ensure its sustainability and making use of land by introducing the development of indoor farming with the integration with IoT.

Thus, this project aims to build a IoT-based Agriculture Monitoring Framework that can receive the environment data and automate some of the processes of indoor farming. In this project, Raspberry Pi and Node-RED will be studied and used to build the framework. Since the environment of indoor farming are required to monitor by the user, few sensors will be attached with the Raspberry Pi so that the data such as temperature, humidity, water level and current value can be collected. The collected data will be stored in database and the user can access it through a dashboard. Furthermore, a chatbot will be created to ease the process of farm management for the user since it supports remote monitoring. In a nutshell, a framework will be built to ensure the crops can live in a suitable environment and makes the indoor farming activities more efficient.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT	Internet of Things
GDP	Gross Domestic Product.
MQTT	Message Queuing Telemetry Transport
GPIO	General-Purpose Input Output
AC	Alternating Current
DC	Direct current
OS	Operating System
RTOS	Real-Time Operating System
PC	Personal Computer
TSDB	Time Series Database
RDBMS	Relational Database Management Systems
SQL	Structured Query Language
API	Application Programming Interface
CPU	Central Processing Unit
EC	Electrical Conductivity
RAM	Random Access Memory
GUI	Graphical User Interface
SDLC	Software Development Life Cycle

CHAPTER 1

Introduction

1.1 Problem Statement and Motivation

Due to the war in Ukraine, food security has become a global issue that concerned by every nation. Although Malaysia is full of resources, but it is still unable to ensure the food for self-sufficient and the nation still highly depends on the import food. Based on [1], the population of Malaysia has been increased to 35.8 million in 2020 and it will keep increasing in the next few years. Despite the agricultural land is increasing, but the land that used to grow agri-food crops has been decreased 2.2% compare with the industrial commodities such as rubber and oil palm. In order to resolve the issue of growing agri-food crops, the idea of self-sustainability farms has been introduced to the people in the nation to increase the availability of the vegetables. The concept of the self-sustainable farm helps the farmer to solve the land problem efficiently since the crops can be planted vertically and supports hydroponic.

However, the crops that grow in indoor are require more attentions from the farmer compare with the outdoor farming. Different with the traditional agriculture activities, planting the crops in indoor requires a lot of data insights to monitor the condition of the plant. As stated in [2], there are few factors that the farmer is required to notice, such as temperature, humidity, nutrient value, light sensitivity, and oxygen content. Without those data, the farmer only can depend on guessing to make decision during the planting process and it will affect the health condition of the crops.

Besides, the health condition of the crops is highly depending on the planting environment which it must be maintained carefully to ensure that it is suitable for the crops to grow. Back to present, the traditional agriculture activities was not maintained efficiently since it is highly relied on manpower to manage. In the indoor environment, the temperature and the humidity are required to maintained in certain values it is suitable for the plant to grow. Thus, the operating method for the electrical appliances that require the user to control manually is no longer efficient.

Furthermore, the indoor environment and the framework has to be monitored all the time to decrease the financial lost that caused by emergency incident. Although the environment monitoring is important, but the user is lack of countermeasure to handle emergency situations since it is impossible for the farmer to work all the time because they are not available during non-office hours. Thus, the productivity of the crops may be affected if something bad happens in the farm during non-office hour.

1.2 Project Objectives

The aim of this project is to develop a IoT-based Agriculture Monitoring Framework that able to provide a suitable environment for the crops to grow in indoor farm and create smart features by enable user to perform monitoring and remote access through pc or smartphone.

The first objective of the project is to integrate IoT technologies in the agriculture monitoring framework. By connecting the Raspberry Pi with the sensors and actuators, the framework is able to perform data collection, processing and produce output. In this project, the environment condition will be measured by the sensors that attached with the Raspberry Pi so that the user can gain the data insight and observe the environment condition of the indoor farm and determine the output by controlling the actuators.

Secondly, the next objective is to enable the automation feature in the framework. Through automation, the actuators such as grow light and exhaust fan in the framework are allowed to work by themselves based on the data that collected from the sensors. By receiving the data with high accuracy, the actuators can be controlled more efficiently with least human interference.

Lastly, the third objective is to create an all-time operating chatbot via Telegram. By implementing the bot, it can be configured to give response the based-on user command for environment monitoring or actuators controlling. Other than that, the bot is capable of sending alert message to the user when something uncommon is happening in the indoor environment.

1.3 Project Scope

The project will mainly focus on some of the major parts which will be used to build a IoT-based Agriculture Monitoring Framework by using the Raspberry Pi 3b+, Node-RED, sensors, actuators, InfluxDB and Telegram.

Firstly, the project will concentrate on one framework that will be used to plant water spinach. In order to create a suitable environment for the water spinach to grow, the sensor DHT22, TDS, water level sensors will be connected with the Raspberry Pi to collect the temperature, humidity, TDS, and water level value in the hydroponic container. After that, the data will be captured by using Node-RED and send to the monitor device via MQTT. Once the monitoring device receive the data, it will be stored in a time-series database that known as InfluxDB.

In the meantime, the data that stored in the InfluxDB will be pulled out and display on the Node-RED dashboard for real-time virtualization. Other than that, the environment data also will be used to control the output of the actuators such as exhaust fan, grow lights and water pump. On dashboard, the respective buttons for the actuators will be created in order to let user to control by sending action payload via MQTT.

Additionally, a telegram chatbot will be created and configured in Node-RED for enabling the user to perform environment monitoring and actuators controlling by sending some basic commands. Besides, the telegram chatbot will also be configured to send alert message to the user when there are some uncommon data is detected.

After the framework has been built, the reliability and the performance of the agriculture monitoring framework will be studied. The seed of the water spinach that has been germinated will be moved in the hydroponic container and place inside the framework. Then, the growing process of the water spinach will be observed and determined whether it can grow healthy or not.

1.4 Contributions

In order to proof the self-sufficient farm is feasible, it is very important for the agriculture sector to implement IoT technologies into their present system. By creating a suitable indoor environment for farming purpose, the productivity and quality of the crops can be ensured. Meanwhile, the project will prove that how the agriculture monitoring framework is going to work.

Firstly, the agriculture monitoring framework that integrated with Raspberry Pi and sensors can generate some useful data insights for the farmer. Instead of performing some guessing behaviour on decision making, now the user will have the actual data that represent the actual situation of the environment, which the framework will be more efficient to manage. The framework will collect the conditions from the environment and display all the related data on the dashboard so they can monitor the farm easily.

Secondly, the enabling of the automation in the agriculture monitoring framework will reduce the manpower efficiently. Instead of depends on the manpower to control the actuators manually, the output that produce by the automation system would be more accurate because it will be based on the data set that collected from the sensors. It also means that the user no longer needs to waste time on actuators controlling, but now they can put more thoughts to take care of the crops.

Lastly, the implementation of the Telegram chatbot that operating all the time will ensure the user can deal with emergencies effectively. With the chatbot, the user can request for the real-time condition of the farm by typing some basic command. Other than that, the user is also allowed to control the actuators no matter they are located at anywhere or anytime. They even no longer need to worry about the farm condition when they are off duty. If there are unusual data detected, the chatbot will send an alert message to the user so that they can perform immediate decision.

1.5 Report Organization

The report is organized into seven chapters, which is Introduction, Literature Review, System Methodology, System Design, System Implementation, System Evaluation & Discussion and Conclusion & Recommendation. In the first chapter which is the Introduction, the problem statements, motivation, project objectives, project scope, and the expected contribution will be discussed. In the second chapter which is Literature Review, the hardware and software technologies that will be used in the project will be reviewed. Besides, there are four existing system that related to the project will also be reviewed in this chapter.

Furthermore, the third chapter which is System Methodology will discuss about the development model that has been used in this project. The hardware and software that used to build the system will be stated clearly in this chapter. In the fourth chapter which is System Design, the design of the system will be elaborated. After that, the functional modules and the system flows will also be stated clearly in this chapter.

In the fifth chapter which is System Implementation, the setup of the hardware and software including the setting and configuration will be discussed. Then, the system operating will also be described clearly step-by-step. Additionally, the sixth chapter which is System Evaluation and Discussion will review about the system based on the performance metrics and objectives. Lastly, the seventh chapter which is Conclusion and Recommendation will conclude what has been done throughout the whole project and provides some recommendation that would improve the system.

1.6 Concluding Remark

The purpose of writing this chapter is to present the overall concept of the project to the reader. It provides the reader with an understanding on what is the problems that the user is facing, what type of objectives that the project is planning to achieve and how the project is scope. Aside from that, the reader will also know the expected contribution of the project and how the agricultural monitoring system will help user.

CHAPTER 2

Literature Reviews

2.1 Review of the Technologies

2.1.1 Hardware platform

In this project, there will be a main hardware which is the microcontroller or microprocessor board that will be used to control the whole system in terms of collecting data from sensors and controlling the actuators.

- **Raspberry Pi 3B+**

  Raspberry Pi is a microcontroller board that used by many developers to explore the possibility of IoT-related system. Initially, the Raspberry Pi was very popular to use by the student to explore the knowledge of IoT because the price was low and friendly for the beginner. Due to the pandemic for the last two years, the price of the microcontroller has been risen due to the shortage of resources for production purpose.

Figure 2.1.1: Raspberry Pi 3B+

Based on the figure above, Raspberry Pi 3B+ comes with 40 General-Purpose Input Output (GPIO) pins that supports the connection from varieties of sensors and actuators. Besides, the microcontroller was equipped with Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz. Other than that, 1GB LPDDR2 SDRAM is built in the board to meet the minimum requirement for the program to run smoothly.

In term of operating system, it supports Linux-based OS and capable of running multiple programs. It even supports dual-band wireless local area network (LAN) and fast ethernet port that enable the connection of the microcontroller board to the internet. To power up the microcontroller board, it requires a 5V/2.5A DC power input through a Micro USB power port. By utilize the microcontroller properly, there are very high possibility for the user to develop many interesting IoT-related projects.

- **Arduino Uno Rev3**

  Arduino Uno is also a microcontroller board that based on ATmega328P. Similar with Raspberry Pi, it also very popular among the group of developers and students because of its price. It is also beginner-tolerant because the user no need to worry too much in doing something wrong. Even in the worse-case scenario, the user only requires to replace a few dollars chip to start over.

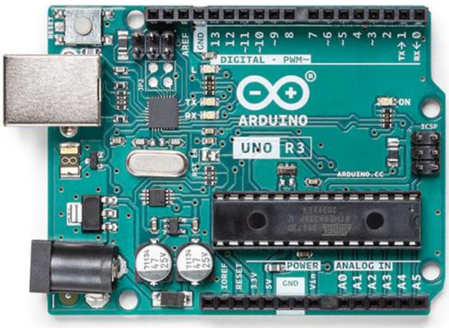


Figure 2.1.2: Arduino Uno Rev3

Besides, Arduino Uno comes with 14 GPIO pins that used for connection purpose. Different with Raspberry Pi, it has 6 analog inputs that support analog signal in the readable format from the sensors. The microcontroller board require 5V of operating voltage to run through the power jack. To get started, the microcontroller board can be configured by the user with a USB cable to computer and AC to DC adapter.

- **ESP8266**

  A low-cost microcontroller which comes with WiFi-enabled capability that produced by the Espressif Systems that located in China. ESP8266 is also equipped with Tensilica Xtensa® 32-bit LX106 RISC that supports Real-Time Operating System (RTOS) which allows the board to operate at clock frequency between 80 to 160 MHz. Furthermore, a Wi-Fi transceiver that known as 802.11b/b/n is incorporated with ESP8266, which it allows the controller itself to connect to a Wi-Fi network and access internet.

Figure 2.1.3: ESP8266

Besides, ESP8266 contains total of 30 pins and 17 of them are GPIO pins that can be assigned with few peripheral duties such as UART, I2C, SPI and etc. Same with Adruino Uno, ESP8266 can be programmed via the Adruino IDE platform by connecting to the desktop through USB cable.

2.1.2 Firmware/OS

- **Raspberry Pi OS**

  In order to ensure the Raspberry Pi hardware to function, a Raspberry Pi OS imager must be flashed inside a SD card and install in the hardware. The latest version of the OS optimized the Linux OS for the Raspberry Pi board to serve as an interface so that the user can interact with the microcontroller board easily.

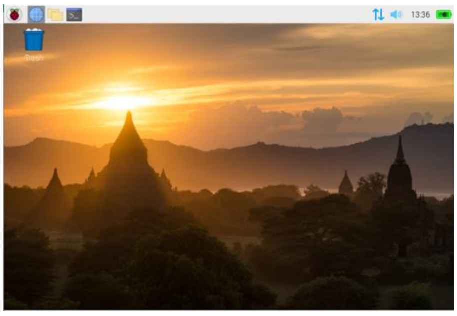


Figure 2.1.4: GUI of Raspberry Pi OS

As shown in the figure, the OS provides the user with a convenient desktop environment that consist of application menu, web browser, file manager and terminal. Besides, the time and Wi-Fi setting is also will be displayed on the desktop to ease the user operation.

- **Arduino IDE**

Different with Raspberry Pi, the Arduino Uno cannot operate an OS because it is a simple board that allows user to program a particular microcontroller chip which is the ATmega328P. However, there is small amount of code is preloaded in the flash memory of the chip and it known as bootloader. The Arduino Uno is relying of the bootloader enables the software in the user's PC to load the code of the program onto the flash memory.

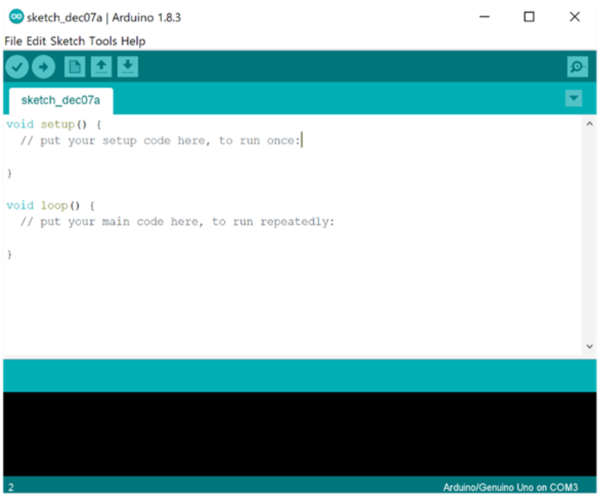


Figure 2.1.5: Arduino IDE Application

As shown in the figure, Arduino IDE provides a simple and friendly interface for the user. The user can write codes in the sketch file and use the library resources that available in the software. Once the code for the whole program is wrote, the user can upload the content in the sketch file to the flash memory of the Arduino Uno so it can produce output accordingly.

2.1.3 Database

- **MySQL**

  SQL, which also known as Structured Query Language is a programming language that mainly used in data management in Relational Database Management Systems (RDBMS). In present, MySQL is widely used for query relational database which it is a collection of tables that stores a specific set of structured data. The data will be stored in rows and columns and the tables among the database are somehow linked in various ways. The database offers the user to access the records by only entering a single command to retrieve a great amount of data.

- **InfluxDB**

  InfluxDB is time-series database that optimized to store and retrieve time-series data which are tracked or monitored over time. The data was indexed with time order which arranged in the sequence of data point. The database is designed to support and store a large volume of time-series data. Besides, it also consists of measurements such as monitored, tracked, and refinement of data. It also can be used to analyze data, read data from sensors, or even stock exchange data.

2.1.4 Programming Language

- **Python**

  A general-purpose programming language that uses for high-level programming. It is also a kind of object-oriented approach which helps the developers to build small- or large-scale project with simple coding techniques. Different with other programming languages, Python used the significant indentation to increase the readability of the code. It helps the user to produce a clear and logical code that is easy for understanding.

  Besides, there are a large number of libraries that are available in Python which helps the developers to build project. The user is only needed to import the library before writing the script and those functions will be ready for use. For embedded devices like Raspberry Pi, the user can write the program conveniently with the helps of Python libraries to interfacing the board with the sensors and actuators.

- **Node.js**

  Node.js is an open-source JavaScript runtime environment that runs the codes outside of a web browser. It is a popular framework that widely used by the developer and big companies to build application because it supports cross-platform. Besides, the developer us Node.js to writing command line for scripting on the server sides. It is mainly uses for API and applications such as real-time and streaming web. Node.js is a best choice for writing a backend server and database because the code can be running as standalone application instead of being evaluated in a browser environment.

  Furthermore, the Node-RED programming tool was built on Node.js because it supports light-weight runtime and taking full advantage of event-driven model. It provides a browser-based editor for the user to wire the flow and connecting the nodes to deploy a system easily. It also makes the programming tool become ideal because it can run at the edge of network on the hardware like Raspberry Pi.

2.1.5 Summary of the Technologies Review

Hardware platform

Description	**Firmware/OS**	**Strength**	**Weakness**
Raspberry Pi 3B+	Raspberry Pi OS	- Fast CPU performance - Supports Linux OS - Consist of 40 GPIO pins - Supports many programming languages	- High power consumption
Arduino Uno Rev3	Arduino IDE	- Low power consumption - Supports analog input - Low cost	- Low CPU performance - Not support Linux OS - Consist of fewer pins - Only support C/C++

Esp8266	Arduino IDE	- Low cost - Wi-Fi enabled - Supports analog input	- Low CPU performance - Not support Linux OS

Table 2.1.1: Summary of Hardware platform

Database

Description	**Strength**	**Weakness**
MySQL	- Store relational data - Fast delivering - Support heavy load	- Complex query optimization - Performance drop when large data is stored
InfluxDB	- Store time-series data - Support real-time analytics - Simple to store and manage	- Limited scaling - Support single series

Table 2.1.2: Summary of Database

Programming Language

Description	**Strength**	**Weakness**
Python	- High-level language - Used for developing large project	- Need to write a lot of codes to run the system
Node.js	- Require minimum effort of code to run the system - Fast understandable	- Not suitable for big project

Table 2.1.3: Summary of Programming Language

2.2 Review of the Existing Systems/Applications

2.2.1 IoT based modern agriculture system using Raspberry Pi

Based on [10], the author wanted to modernize the agriculture industry with the involvement of technologies. Since the agriculture is hindering the development of the country, thus a better production, distribution, and cost control must be ensured. Besides, the author has designed an agriculture system that will makes an efficient use of electricity and water usage. The embedded device Raspberry Pi and Python programming language is used in the project.

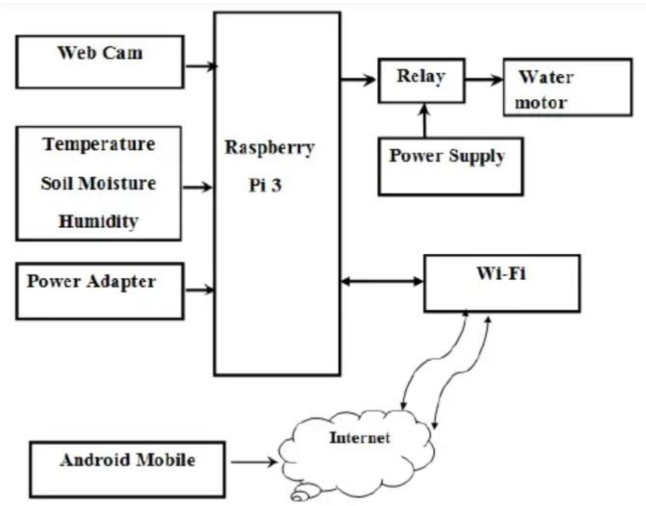


Figure 2.2.1: Block diagram of IoT based modern agriculture system

In the figure, a Raspberry Pi connected with various sensors such as temperature, soil moisture and humidity. Other than that, a relay is used to control the water motor to keep the environmental climate constant. In term of functionalities, the data that collected by the analog sensor will be stored in Raspberry Pi internal memory (RAM) and will be sent to the android phone through Wi-Fi. Besides, the water motor can be controlled through the application on user's android phone. Lastly, a Pi camera was attached to the mainboard so the user can monitor the real-time situation of the farm.

2.2.2 Design smart home hydroponic gardening system using Raspberry Pi 3

The author from [11] has designed an inexpensive automatic hydroponic system in the household. The purpose of the system is to encourage people to plant their own vegetables at home to avoid harmful chemicals in the food. Besides, growing crops by using hydroponic method allows the consumer to harvest freshly natural.

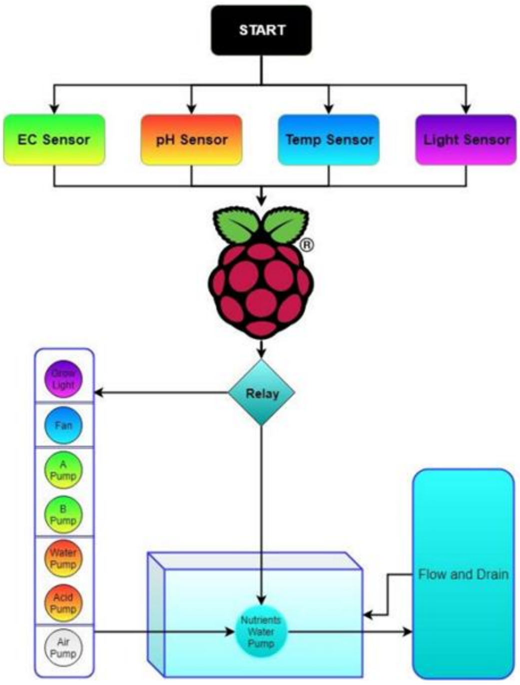


Figure 2.2.2: Block diagram of smart home hydroponic gardening system

Based on the figure above, the hydroponic gardening system was integrated by using Raspberry Pi as the operating system, some sensors and water pump. There are few controlling parameters in the system that used to monitor the condition of the plant, that includes EC control, pH control, light intensity control, and temperature control. Other than that, the system will use that collected data to control the actuators such as grow light, fan, water, and air pump to ensure the environment is suitable for crops growing.

2.2.3 IoT-based Automated Indoor Agriculture System Using Node-RED and IBM Bluemix

After accessing the project from [12], we know that the author aims to develop an IoT based Automated Indoor Agriculture System to ensure the agriculture industry can be more sustainable. The system is capable of automate the processes of indoor farming by monitoring the parameters that contribute to a healthy crop growth such as temperature, humidity, light intensity, and pH levels. Besides, the agriculture system was designed with the integration of central processing system, Node-RED and IBM Cloud so it could work efficiently.

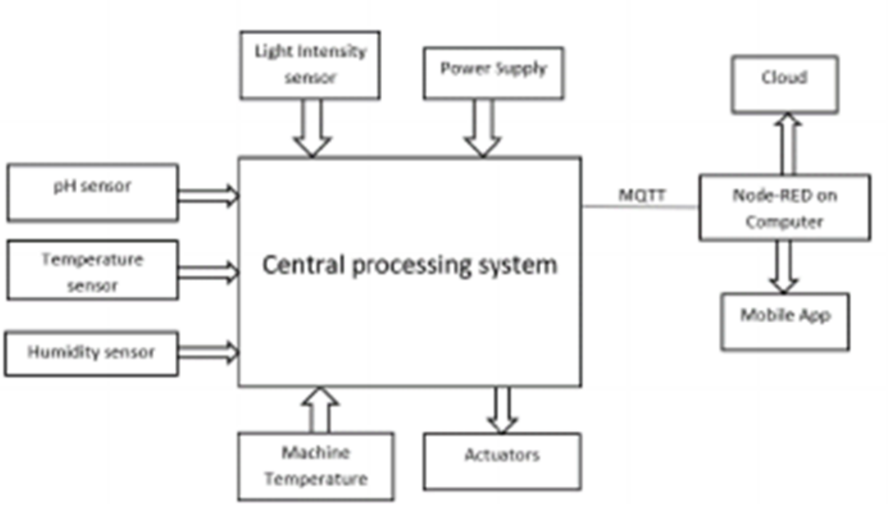


Figure 2.2.3: Block diagram of IoT-based Automated Indoor Agriculture System

Based on the figure above, there are few IoT sensors was connected to the central processing system. The differences of this system with the previous system are it integrated Node-RED which is a low-code developing tool that used to wiring and controlling the hardware devices. The data gathered from the sensors will be transfer from the Raspberry Pi then to Node-RED.

Besides, the Node-RED service has been integrated with the IBM cloud so that the data that collected from the sensors can be stored and access through the application. If the data is found to be out of optimal range, a notification will be sent to the farm owner via the mobile app so the farmer can take immediate action.

2.2.4 Automated smart hydroponics system using internet of things

The author from [13] presented a design of an automated smart hydroponic system by using IoT to increase the crops demand. By implementing IoT, the production from agriculture can be more sustainable to fulfil the needs of people. The system was implemented with NodeMcu, Node-RED, MQTT and few sensors.

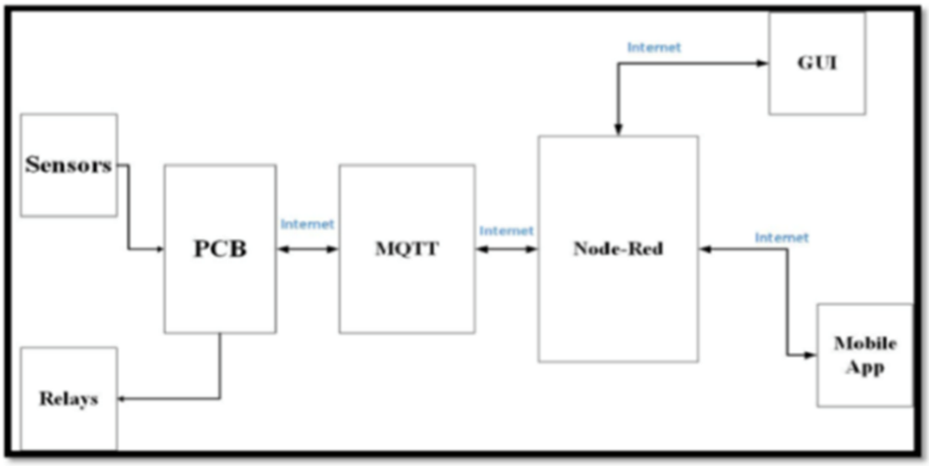


Figure 2.2.4: Block diagram of Automated smart hydroponics system

Based on the diagram above, the main components of the system are the microcontroller that connected with sensors and relays that can be used to further control the actuators. Besides, the block diagram has shown that how the software and hardware are communicated. The data acquired by the sensors will be transferred through the internet via MQTT broker and then to the Node-RED. In Node-RED, the data will be processed and used to decide on which actions should be taken. Besides, the sensors data from the environment will be taken and monitored on the cloud-based webpage with mobile application. By accessing to the data, the user can make decision immediately when somethings wrong.

2.2.5 Summary of the Existing Systems

No.	Existing System	Advantages	Disadvantages	Critical Comments
1	IoT based modern agriculture system using Raspberry Pi	- Accessible to real-time data of the farm condition - Enable remote access to control actuators	- The framework exposed to external environment - Lack of reliable power supply - Required high effort for coding	This project aims to develop an agriculture monitoring system that rely on power supply from inhouse power plug.
2	Design smart home hydroponic gardening system using Raspberry Pi 3	- Low development cost - Automation is enabled in the system	- Data collected did not stored in database - Lack of GUI for user interaction	This project aims to develop a hydroponic gardening system which only run by itself automatically.
3	IoT-based Automated Indoor Agriculture System Using Node-RED and IBM Bluemix	- Easy to program - Data stored in IBM cloud - Remote access through mobile application	- Lack of air flow controlling module	This project aims to develop an automated agriculture system with minimum effort for coding and enable remote access
4	Automated smart hydroponics system using internet of things	- Easy to program - Accessible to data via webpage	- Lack of sensors to monitor water quality	This project aims to develop a hydroponic system that allows user to monitor through the virtualization on webpage

Table 2.2.1: Summary of Existing systems

2.3 Concluding Remark

In this chapter, there are few technologies has been discussed based on their pros and cons which is hardware platforms, firmware/OS, databases and programming languages. Besides, there are four existing system that related to the project has been studies. After reviewing, the information that gathered from these few sections will be very important since it will be applied in this project.

CHAPTER 3

System Methodology

3.1 System Development Models

In the section, there are few models from Software Development Life Cycle (SDLC) will be defined and the details of the project development process will be mentioned clearly.

3.1.1 System Development Model 1: Waterfall Model

As stated in [14], waterfall model is the first model that widely used by most of the software engineer to ensure the project can be developed successfully. The model separates the whole software development process into several phases which is requirement analysis, system design, implementation, testing, deployment and maintenance. The special of this model is the user only allowed to move to next phase once the current phase of the project is completely done.

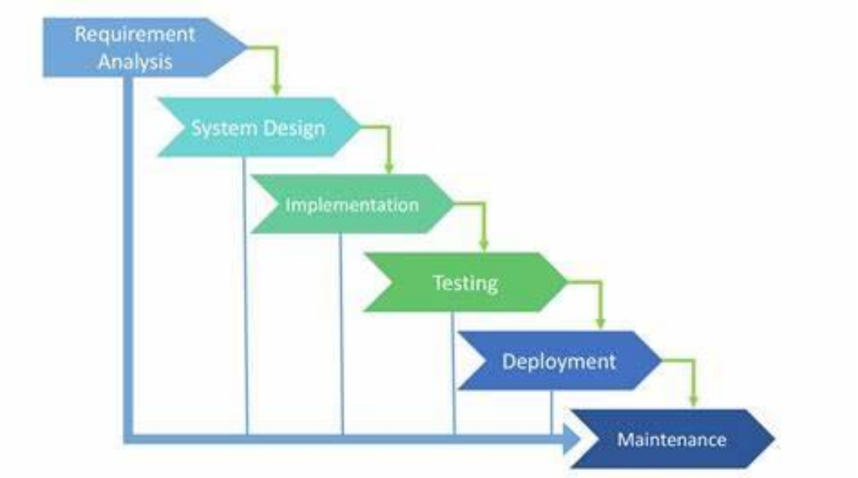


Figure 3.1.1: Waterfall Development Model

Besides, the waterfall model is simple and easy to understand for the user to execute. It has ensured the deliverables output from each of the phases and the documentation are well defined. However, the model is considered as high risk and uncertain since it is not suitable for the complex projects. The model is static because the requirements from the previous phases cannot be changed anymore once the user moved to the next phase.

3.1.2 System Development Model 2: Iterative Model

Iterative model is a model that focus on primarily growth and design by allowing the system development to start with a small set of requirements. Once applying the model, the development process of the software system will be segmented into smaller pieces, which each process will includes four phases, such as requirements, design, implementation and review.

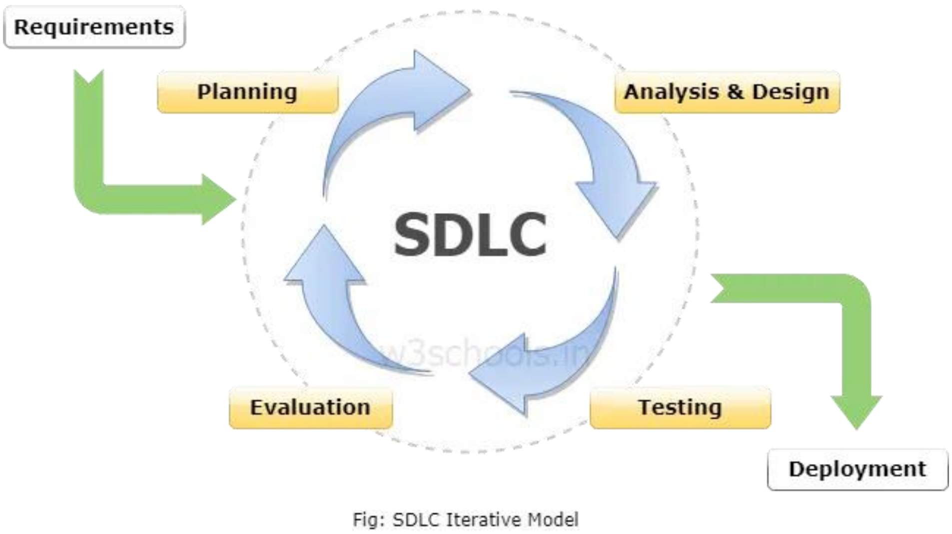


Figure 3.1.2: Iterative Development Model

During an iteration process, [15] stated that the context of entire system specification is not required because only a part of the software will be implemented. Other than that, a new version of the system will be made after every iteration process because the additional requirements will be further reviewed every time. Each iteration process will take two to six weeks.

3.1.3 System Development Model 3: Spiral Model

Spiral Model is a model that combines the concept of iterative development with the waterfall model which allows the incremental releases of the system. Based on [16], the development process of the model has been separated into four phases, which are planning, risk analysis, engineering and evaluation.

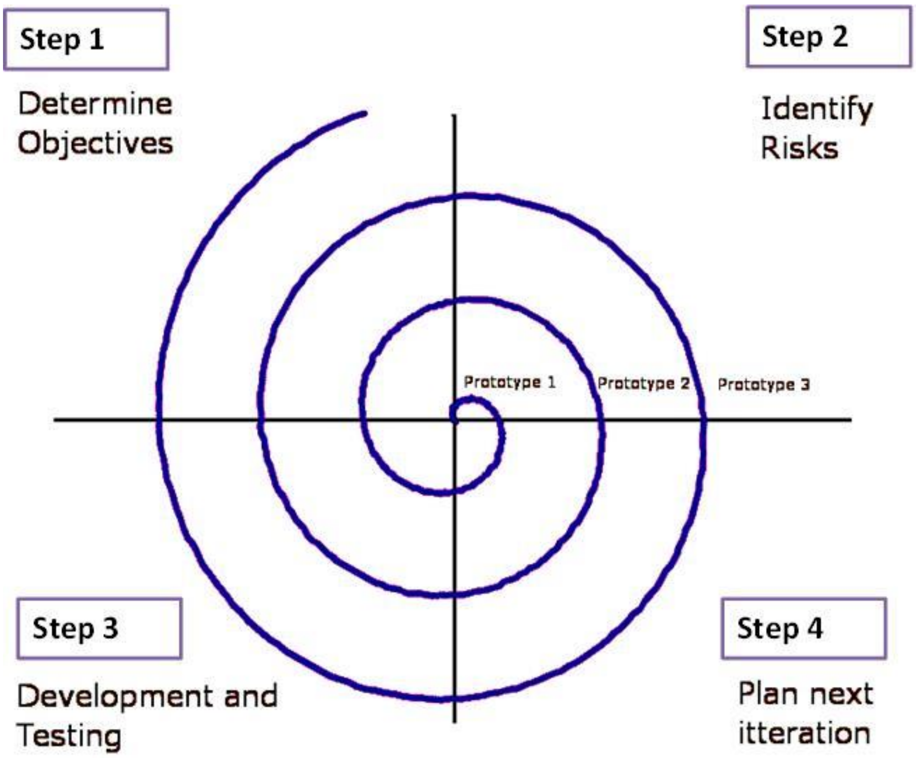


Figure 3.1.3: Spiral Development Model

The spiral model is commonly used in large and complicated software development because it allows continuous enhancement. When a specific activity is done in one iteration, then it will output a small prototype of the entire software. Then, the same activity will continue to repeat in the iteration and make improvement until the whole software is completely built.

3.1.4 System Development Model 4: Kanban Model

Kanban model is one of a kind of agile approach that used to track progresses of the system development. As mentioned in [17], the details of the system development will be stated clearly and virtualize into few sections, such as backlog, planning, design, implementation, test and complete. The task that stated in the board will be moved around between the sections based on its progress.

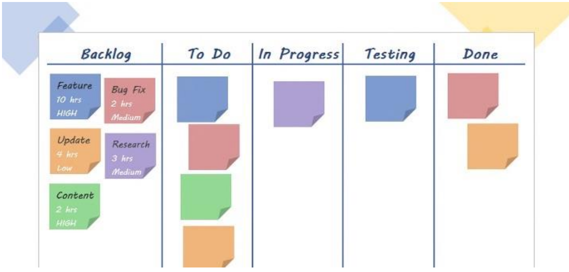


Figure 3.1.4: Kanban Development Model

By virtualizing the progresses, those stated task can act as a reminder for the user to keep track of the workload either it is currently running or completed. Since the model is very flexible, the workflow and be improved or add in anytime and it helps to minimize the bottlenecks of the development process. If the model is executed properly, the result of the system development will be high quality and faster rate.

3.1.5 Selected Model

In this project, Kanban Model is selected due to its flexibility and the ease of progress tracking. Although the project is required to plan properly at the initial phase, but it allows the user to add in new requirements or task anytime by not affecting the entire development process.

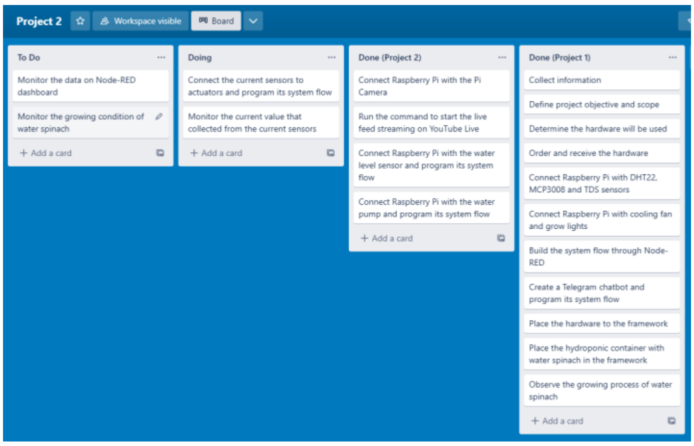


Figure 3.1.5: Kanban board that used to record the progress of the project

3.2 System Requirement

3.2.1 Hardware

During the development process of the project, there are few hardware has been used, which is computer, embedded devices, sensors and camera.

- **Laptop**

Figure 3.2.1: MSI GF65 Thin Model

Description	Specifications
Model	MSI GF65 Thin
Processor	Intel(R) Core (TM) i5-10500H
Operating System	Windows 10
Graphic	NVIDIA GeForce RTX 3060, GDDR6 6GB
Memory	DDR IV 8GB*2 (3200MHz)
Storage	512GB NVMe PCIe Gen3x4 SSD

Table 3.2.1: Specifications of laptop

- **Smart Phone**

Figure 3.2.2: Huawei Nova 5T Model

Description	Specifications
Model	Huawei Nova 5T
Processor	Kirin 980 (7 nm)
Operating System	Android 10, EMUI 10
Memory	8GB
Storage	128GB

Table 3.2.2: Specification of smart phone

- **Raspberry Pi 3B+**

Figure 3.2.3: Raspberry Pi 3B+ Model

Description	**Specifications**
Model	Raspberry Pi 3B+
Processor	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
Operating System	Raspberry Pi OS
Graphic	Cortex-A72(ARM v8)
Memory (RAM)	1GB DDR2 SDRAM
Storage (ROM)	16GB Micro SD Card
Connectivity	- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac WLAN - Gigabit Ethernet (Maximum throughput 300Mbps)
Onboard Port	- 40 – Pin GPIO Header - CSI Camera Port

Table 3.2.3: Specifications of Raspberry Pi 3B+

- **DHT22 Module**

Figure 3.2.4: DHT22 module

Description	Specifications
Input Voltage	3.3V - 6V
Sensing element	Polymer capacitor
Operating range	Humidity: 0 to 100% RH Temperature: -40 to 80 °C
Accuracy	Humidity: +-5% RH Temperature: <+-0.5 °C
Long-term stability	+-0.5% RH/year

Table 3.2.4: Specifications of DHT22

- **5MP Night Vision Pi Camera**

Figure 3.2.5: 5MP Night Vision Pi Camera

Description	Specifications
Photosensitive Module	OV5647
Resolution	5 Million Pixels
Dimension	25mm x 24mm
Support	Access to infrared light or fill light

Table 3.2.5: Specifications of Pi Camera

- **Water Level Sensor**

Figure 3.2.6: Water Level Sensor

Description	Specifications
Sensor Type	Analog
Input Voltage	3.3V – 5V
Operating Current	< 20mA
Detection Area	40mm x 16mm
Operating temperature	10℃- 30℃

Table 3.2.6: Specification of Water Level Sensor

- **8-Channel relay module**

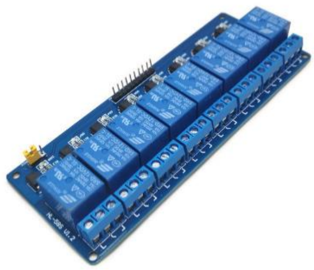


Figure 3.2.7: 8-Channel relay module

Description	Specifications
Maximum Current	10A
Maximum Voltage	250V AC or 30V DC
Input Voltage	5V

Table 3.2.7: Specifications of 8-Channel relay module

- **Micro Submersible Water Pump**

Figure 3.2.8: Micro Submersible Water Pump

Description	Specifications
Input Voltage	3.3V – 5V
Operating Current	0.1 - 0.2A
Size	45mm x 30mm x 25 mm
Weight	30g
Operating temperature	80℃

Table 3.2.8: Specification of Micro Submersible Water Pump

- **MCP3008 Analogue to Digital Converter**

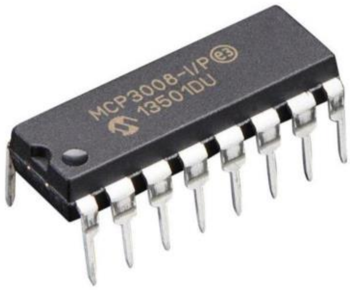


Figure 3.2.9: MCP3008 Analogue to Digital Converter

Description	Specifications
Operating Voltage	3.3V - 5V
ADC Channels	8
Interface	Serial Peripheral Interface (SPI)

Table 3.2.9: Specification of MCP3008

- **ACS712 Current Sensor Module**

Figure 3.2.10: ACS712 Current Sensor Module

Description	Specifications
Supply Voltage	4.5V - 5.5V
Measure Current Range	-5A - 5A
Sensitivity	180mV/A - 190mV/A
Dimension	31mm x 12mm

Table 3.2.10: Specification of ACS712

3.2.2 Software

- **Raspberry Pi OS**

  An operating system that required be installed in Raspberry Pi hardware to enable the interaction between user. It is a Linux-based OS that provide GUI for the user to perform configuration, write code and even running the script.

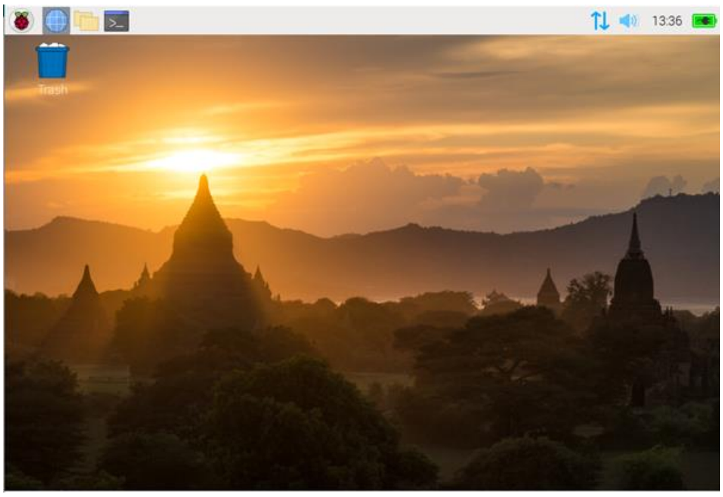


Figure 3.2.11: The user interface of Raspberry Pi OS

- **Node-RED**

  A flow-based development tool that used for visual programming. It allows the user can perform wiring hardware devices, APIs and web services by connecting the node together. The browser-based editor and the clear interface ease the user to wire the flows together by using the wide range of nodes in the palette to deployed to deploy a system.

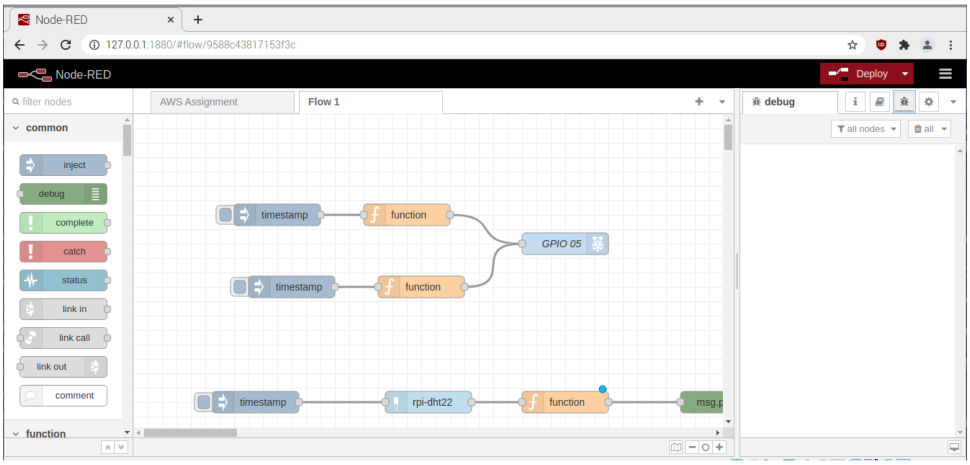


Figure 3.2.12: The GUI of Node-RED

- **HiveMQ**

  A MQTT broker which allows the user to publish and subscribe messages. It is a server that receive all the messages from the publisher client and direct them to the subscriber client.

Figure 3.2.13: The logo of HiveMQ

- **InfluxDB**

  An open-source time-series database (TSDB) that has been developed to store and retrieve the time-series data. There are few functionalities that supported by InfluxDB, such as operations monitoring, applications metrics, collect IoT sensor data, and perform real-time analysis.

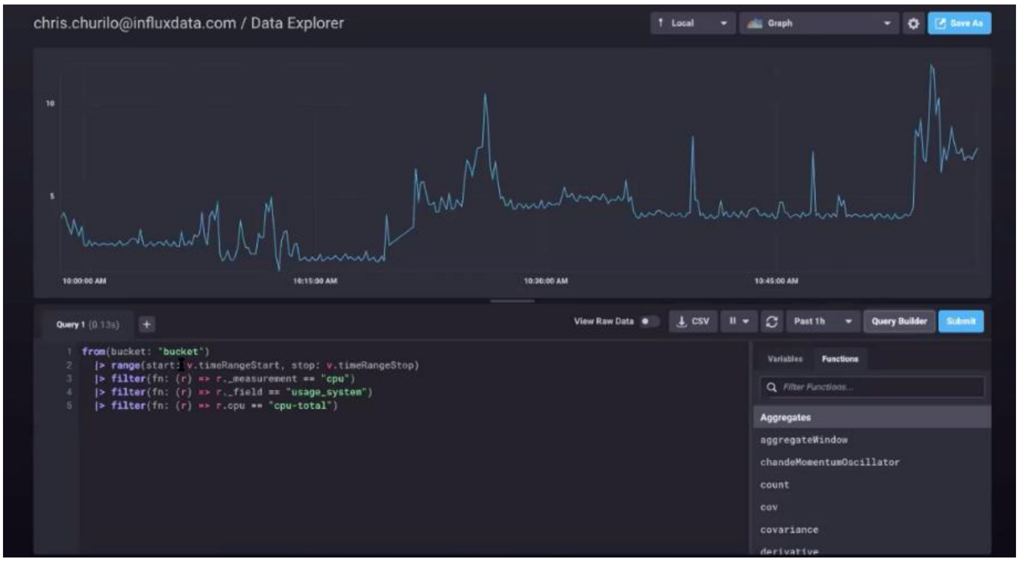


Figure 3.2.14: The GUI of InfluxDB

- **Telegram**

A freemium, cross-platform, cloud-based instant messaging (IM) service. The messaging application is main focus on speed and security which super-fast, implemented with simple interface and free for all users. The messages of the user will sync seamlessly across any number of phones, tablets or computers.

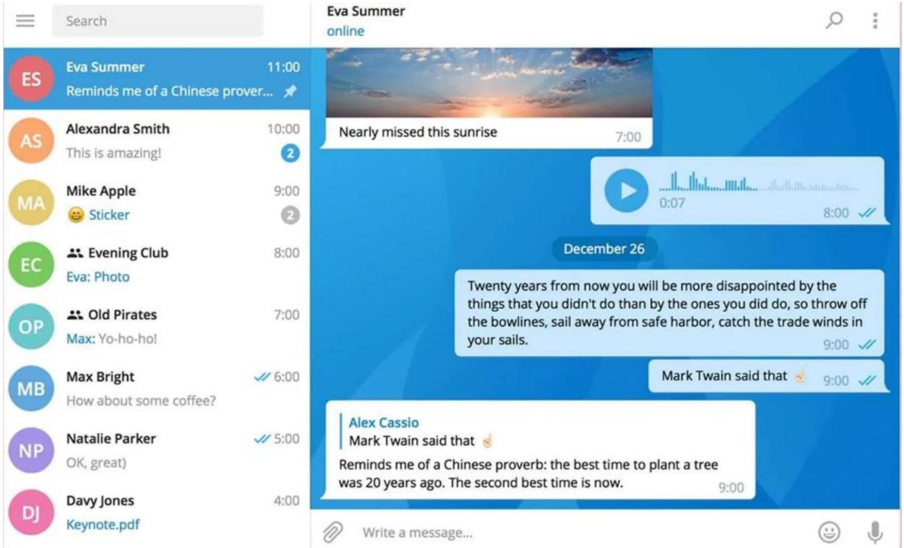


Figure 3.2.15: The GUI of Telegram

- **The Botfather**

A third-party application that run in Telegram. It allows the user to create and configure a chatbot to receive responses by entering some basic commands.

Figure 3.2.16: The logo of Botfather

3.3 Functional Requirement

No.	Description
1	Raspberry Pi should collect the data from the temperature and humidity sensor and publish them through a MQTT every minutes.
2	Raspberry Pi should stream the real-time video feed that captured by the Pi Camera on YouTube Live Channel.
3	Raspberry Pi should turn on and off the grow light on specific time.
4	Raspberry Pi should turn on and off the exhaust fan when the temperature has exceeded or lower than specific value.
5	Raspberry Pi should turn on and off the water pump when the water level has exceeded or lower than specific value.
6	InfluxDB will store the real-time sensors data that receive from Raspberry Pi through MQTT.
7	The Node-RED dashboard should display all the data that pulled from InfluxDB.
8	Node-RED will provide output such as sensors data based on the user command that send through Telegram.
9	Node-RED will send alert message to the user via Telegram when unusual data has been detected.

Table 3.3.1: Description for Functional Requirements

3.4 Project Milestone

FYP1

Task Description	Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
FYP1														
Project Analysis														
Collect information regarding the project														
Define the project objective and scope														
Determine the hardware that will be used														
Order and receive the sensors and actuators														
Project Implementation														
Connect Raspberry Pi with the sensors & actuators														
Create the flow of the system by using Node-RED														
Publish the data to user device through MQTT														
Create a chatbot to control the system via Telegram														
Project Testing														
Attach the hardware to the framework.														
Run the system overnight in the lab														
Place the water spinach in the framework														
Observe the growing process of water spinach														

Table 3.4.1: Milestone of FYP1

FYP2

Task Description	Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
FYP2														
Project Review														
Review the weakness of current system														
Add water level sensor and water pump to the system														
Add pi camera to the system & stream the live feed on YouTube														
Plantation Process														
Place the water spinach in the framework														
Monitor the growing conditions of water spinach														
Record the data and condition of water spinach														
Additional Integration														
Add DHT22 to the system and place nearby the IoT Devices														
Add current sensors by connecting it with actuators														
Monitor the stability of the framework														
Project Conclusion														
Prepare final project report														
Present the final project														

Table 3.4.2: Milestone of FYP2

3.5 Estimated Cost

The following tables shows the hardware that purchase from the local market which is Cytron and Autobotic online store. The cost of the hardware for commercialization are dynamic since it is based on the production supply and the discount campaign that held by the seller.

No.	Product Description	Quantity	Price (RM)	Total (RM)
1	Raspberry Pi 3 Model B+	1	156.00	156.00
2	5MP Night Vision Camera for Raspberry Pi	1	50.10	50.10
3	DHT22 Sensor Module Breakout	2	15.50	31.00
4	Water Level Sensor	1	1.40	1.40
5	8 Channel Relay Module with Opto-Isolator (5V)	1	26.50	26.50
6	40 Ways Male to Male Jumper Wire	1	2.50	2.50
7	40 Ways Male to Female Jumper Wire	1	2.50	2.50
8	40 Ways Female to Female Jumper Wire	1	2.50	2.50
9	DC Jack(Female) to Screw Terminal Adapter	2	2.00	4.00
10	Adapter 12V 2A (UK Plug)	2	9.00	18.00
11	Breadboard 16.5x5.5cm (830 Holes)	2	3.90	7.80
12	DC 12V 2Pin Desktop / CPU Cooling Fan	2	4.90	9.80
13	12V LED Grow Light	2	14.95	29.9
14	MCP3008 - Analogue to Digital Converter	1	19.50	19.50
15	ACS712 5A Current Sensor Module	2	9.50	19.00
16	Water Quality TDS Meter Sensor	1	58.00	58.00
17	DC 12V to 5V Converter	1	8.00	8.00
18	Micro Submersible Water Pump DC 3V-5V	1	3.90	3.90
			Grand Total:	**450.40**

Table 3.5.1: Estimated cost of the project

3.6 Concluding Remark

In this chapter, there are few system development models has been studied and one of them has been chosen and applied in this project. Then, the hardware and software that used to develop the system has been examined. There are few functional requirements has been stated clearly to ensure direction of the system development will be correct. Besides, the project milestone for FYP1 and FYP2 has been shown by differentiate the task and its respective time of execution. Finally, the estimated cost that will be used to buy the hardware that applied in this project has been stated clearly in a table.

CHAPTER 4

System Design

4.1 System Architecture

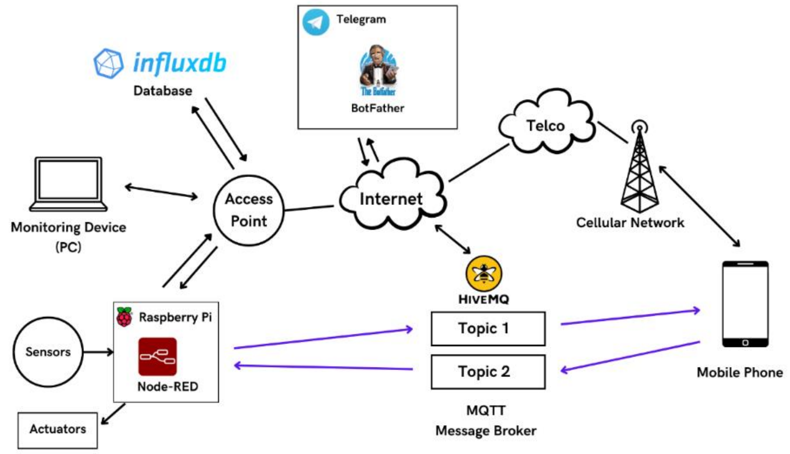


Figure 4.1.1: Architecture diagram of the project

In this project, Raspberry Pi will be used as main controller to control the whole system and Node-RED will be installed and act as a back-end engine to support the whole system flow. Besides, there are few sensors will be connected with Raspberry Pi to collect the environment data. The environment data that collected from the sensors will be pushed to the HiveMQ Message Broker through MQTT Topic 1.

On the other side, the monitoring device will also install and use Node-RED to pull the data from HiveMQ through MQTT to further store the data in the time-series database that known as InfluxDB. Then, the data that store in the InfluxDB will be pulled out to show on the Node-RED dashboard for virtualization purpose. Other than that, the user can control the actuators that connected with Raspberry Pi through the Node-RED dashboard. The action that commanded by the user will be push to HiveMQ through MQTT Topic 2.

Lastly, a Telegram chatbot will be created through Bot Father in the Telegram application. The chatbot will be configured and able to provide response by entering some simple commands. The bot is supported with some functions such as environment condition checking, automation controlling, and sending alert message.

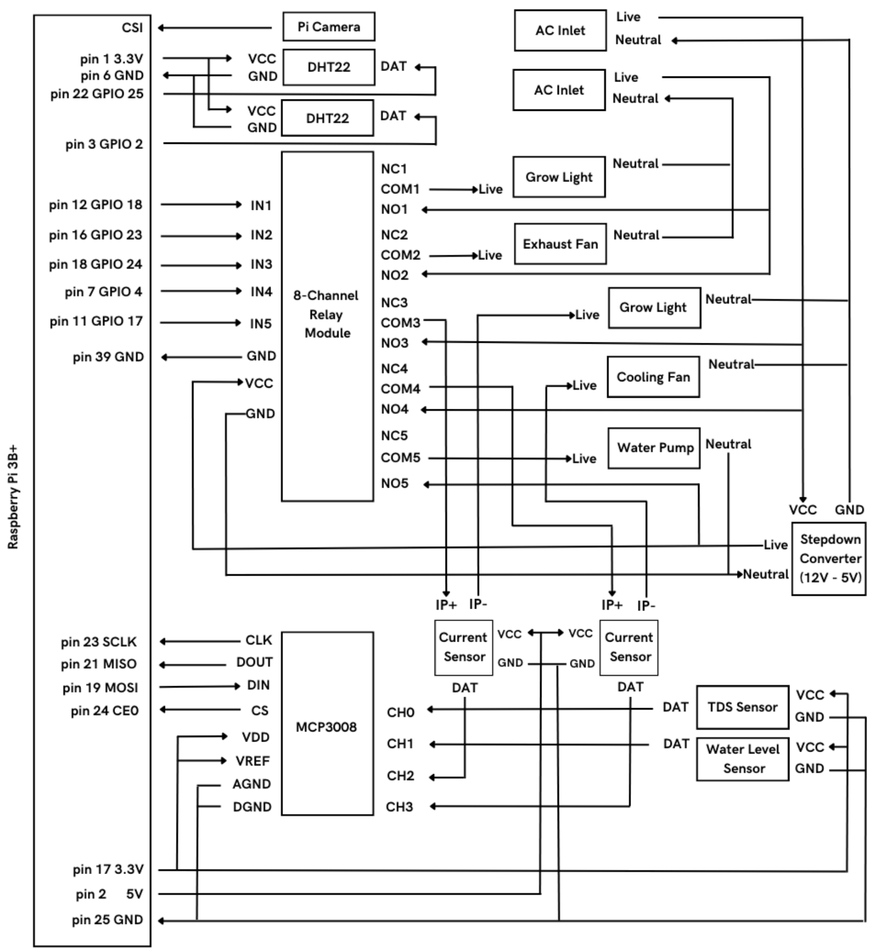


Figure 4.1.2: Hardware connections between Raspberry Pi with sensors and actors

Furthermore, the figure above shows the connections details between the Raspberry Pi with the sensors and actuators. Firstly, a Pi Camera will be attached to the CSI port of the Raspberry Pi to snap picture and capture video feeds of the environment. Secondly, two DHT22 Breakout Modules will be connected with Raspberry Pi to collect the temperature and humidity value from the environment.

Besides, an 8-Channel Relay Module will also be connected to act as a switch to control the responses of the actuators, which include fans, grow lights and water pump. In order to avoid the Raspberry Pi overloaded from power supply, the power source for the relay module will be pulled from the power plug by going through a 12V to 5V stepdown converter.

Lastly, an Analog to Digital Converter (ADC) MCP34008 has been used to connect with Raspberry Pi to read the analog input from the sensors, which include TDS sensor, Water Level sensor, and Current sensors.

4.2 Functional Modules in the System

- Module 1 (Data Collection):
  - The data that collect from the sensors will transmit to the monitoring device via MQTT and further store in InfluxDB database. The sensors data that will be collected includes temperature, humidity, EC TDS, water level and current value.
- Module 2 (Live Stream)
  - A night vision Pi Camera is required to attach with Raspberry Pi. By running the raspivid command, the video feed that recorded by the pi camera will be streamed to YouTube Live through FFmpeg. Besides, the secret stream key has to be stated in the command so the live stream can be linked to the user YouTube account.
- Module 3 (Automation Control):
  - The automation will include few actuators. First, the grow light will be turned on between 7am to 8pm everyday to simulate as the sunlight. Besides, the exhaust fan will be turned on when the surrounding temperature value exceeds 25 ℃ to maintain the environment temperature to drop to normal range. The cooling fans will be turning on and off every 5 minutes to stimulate as the real wind that blowing the plant. Finally, the water pump will pump water to the hydroponic container when the water level is dropped to 2.
- Module 4 (Data Visualization):
  - Node-RED will be configured to pull the sensors data from InfluxDB and display them on Node-RED dashboard. Besides, the YouTube live stream and control button for actuators will also be shown.
- Module 5 (Telegram Chatbot):
  - The chatbot will respond based on the command that enter by the user through Telegram. The chatbot will provide few functions, which is retrieve the environment condition, enable the automation of the framework and snap photo of the environment. An alert message will also be sent to the user when uncommon data is detected.

4.3 System Flow

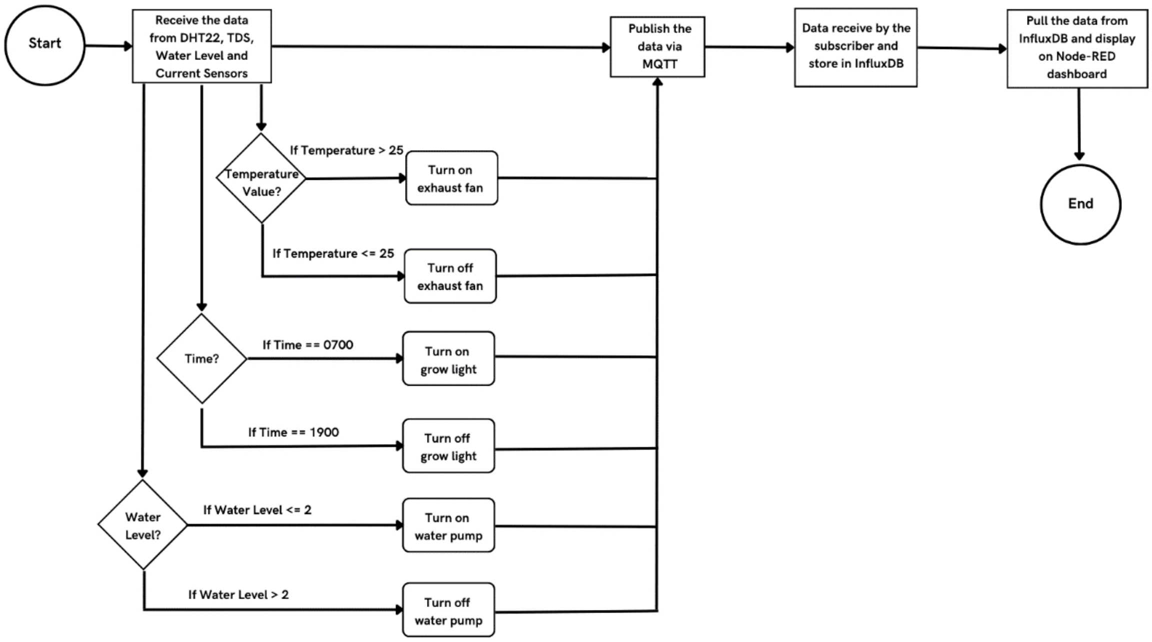


Figure 4.3.1: The system flow of the monitoring framework

Once the system is start, the Raspberry Pi will collect the temperature, humidity, TDS, water level and current value from the sensors that connected with the microprocessor. After that, the system will process the data by going through 3 switch cases to determine the response of the actuators, which is exhaust fan, grow light and water pump.

Firstly, the exhaust fan will be turned on if the temperature value is exceeded 25°C and remain as off status if the temperature value is equal or below 25°C. Secondly, the system will turn on or off the grow light automatically with the implementation of the time-range node. The grow light will be turned on when the time reach 7am and will be turned off when the time is after 8pm. At the meantime, the water pump will operate based on the water level value. When the water level value is equal or below 2, then water pump will be turned on to refill the water in the hydroponic container. If the water level value is higher than 2, then the water pump will remain as off status.

Furthermore, all the data that collected from the sensors will be published to the subscriber through MQTT. Once the subscriber receives those data, it will further store it into InfluxDB. By pulling the data from InfluxDB, the data can be display on the Node-RED dashboard to ease the monitoring process for the user.

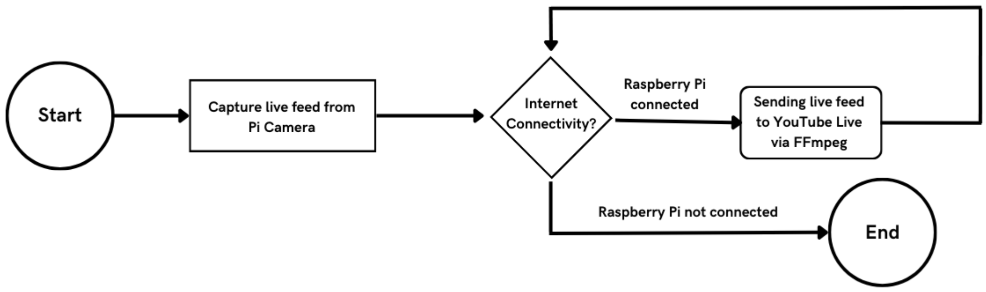


Figure 4.3.2: The system flow of the live feed streaming

In order to stream the framework environment to YouTube Live, a Pi Camera has to attached with the Raspberry Pi to capture the live feed. After that, the internet connectivity will be checked to ensure the live feed can be sent to YouTube live continuously. Once the Raspberry Pi is disconnected from internet access, then the streaming process will be ended immediately.

4.4 GUI Design

The GUI of the monitoring framework will be implemented through Node-RED dashboard. The data that collected from the sensor such as temperature, humidity, TDS and water level will be pulled from InfluxDB and display on the dashboard. By visualising the data, the user now can monitor the environment condition conveniently. Besides, the button of the actuators such as fans, grow lights and water pump will be synchronized so that the user can always know that whether they are turning on or off. There is also a main switch which use to enable the whole system to run automatically.

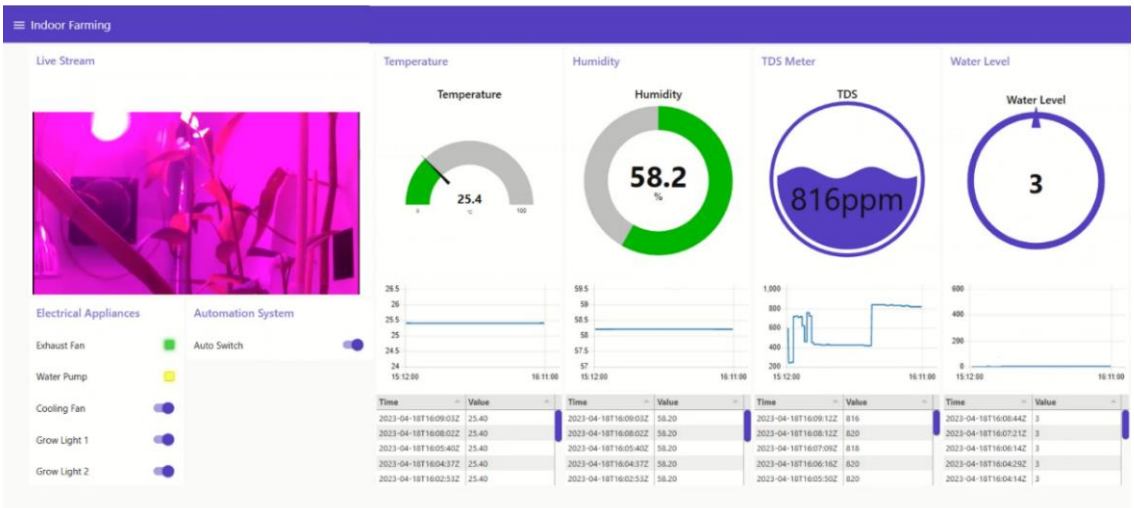


Figure 4.4.1: The Node-RED dashboard of farm condition

Additionally, there is also an IoT Device Monitoring interface which allows the user to monitor the data of the sensors that placed around the IoT devices to ensure their running status and functionality.

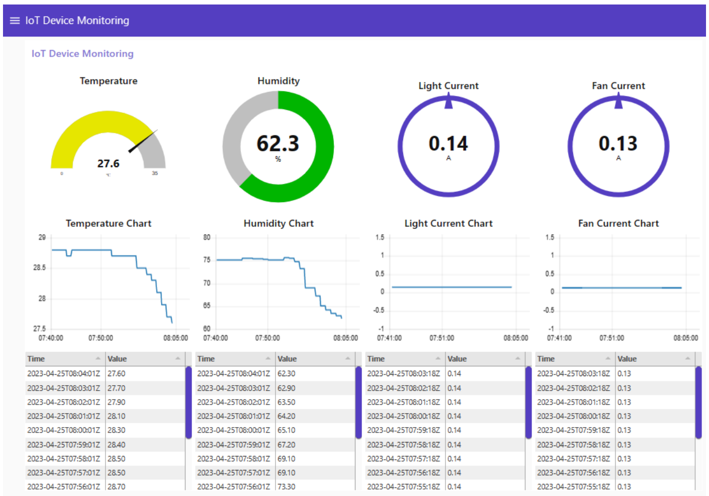


Figure 4.4.2: The Node-RED dashboard of IoT devices condition

Other than that, the Telegram chatbot is also consider as an interface for the user to control the system. Thus, a menu was configured to shows the features of the chatbot clearly. Throughout the menu, the user can choose to access the features such as environment condition, enable automation, or snap photo. The user only need to click the highlighted word and the command will be entered automatically.

Figure 4.4.3: The interface of Telegram chatbot

4.5 Concluding Remark

In this chapter, the architecture of the system has been discussed to understand how the system are going to work technically. The connection between the hardware has been stated clearly and the functional modules has explained about what the function of the system is going to offer. Then, the system flow shows the process of how the system run and the GUI of the system for both Node-RED dashboard and Telegram chatbot has been displayed.

CHAPTER 5

System Implementation

5.1 Hardware Setup

The project is focus on build a prototype of agriculture monitoring framework by using the minimum resources to reduce the chance of failure and malfunction.

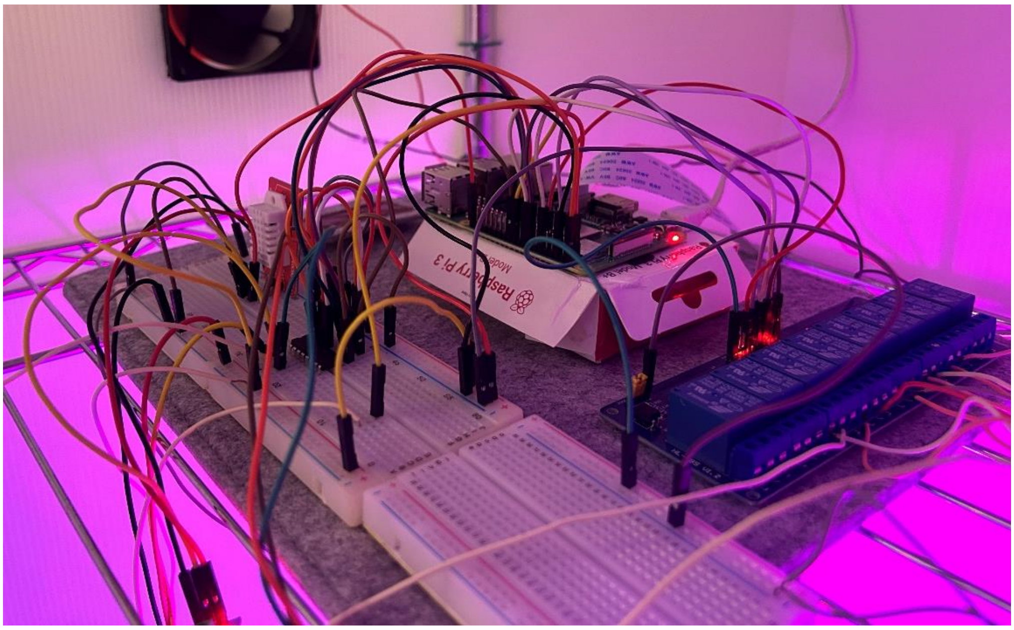


Figure 5.1.1: The connection of Raspberry Pi to breadbox and relay module.

Based on [Figure 5], the microprocessor Raspberry Pi 3B+ is connected with the sensors and actuators through the header pins. Pi Camera is connected to the CSI port on Raspberry Pi. Besides, the header pins that use to connect the sensors is wired to the breadboard since it is more convenient to manage. There are few sensors that is used for the project, which includes DHT22, MCP3008, TDS Sensor, Water Level Sensor, and Current Sensor.

Firstly, there are 2 DHT22 are connected with the Raspberry Pi to collect the temperature and humidity value from the environment for the hydroponic container and IoT devices. Thus, the pin 3 (GPIO2), pin 22 (GPIO25), pin 1 (3.3V), and pin 6 (GND) from the Raspberry Pi has been used to connect the DHT22 via breadboard.

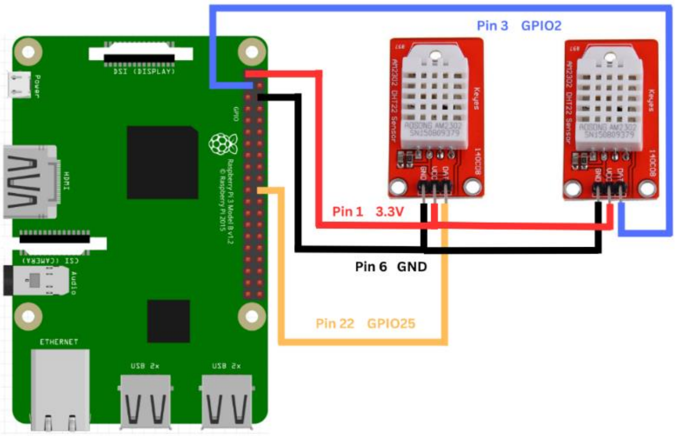


Figure 5.1.2: The connection of Raspberry Pi with DHT22 breakout modules

Besides, the Raspberry Pi are required to connect with the 8-Channels relay module to control the actuators. There is few pins from Raspberry Pi has been used to connect the input pin from the relay module, which are pin 39 (GND), pin 12 (GPIO18), pin 16 (GPIO23), pin 18 (GPIO24), pin 11 (GPIO17), and pin 7 (GPIO4) Additionally, the power source for the relay module will pull from the AC inlet by going through a stepdown converter (12V to 5V) to decrease the burden of Raspberry Pi.

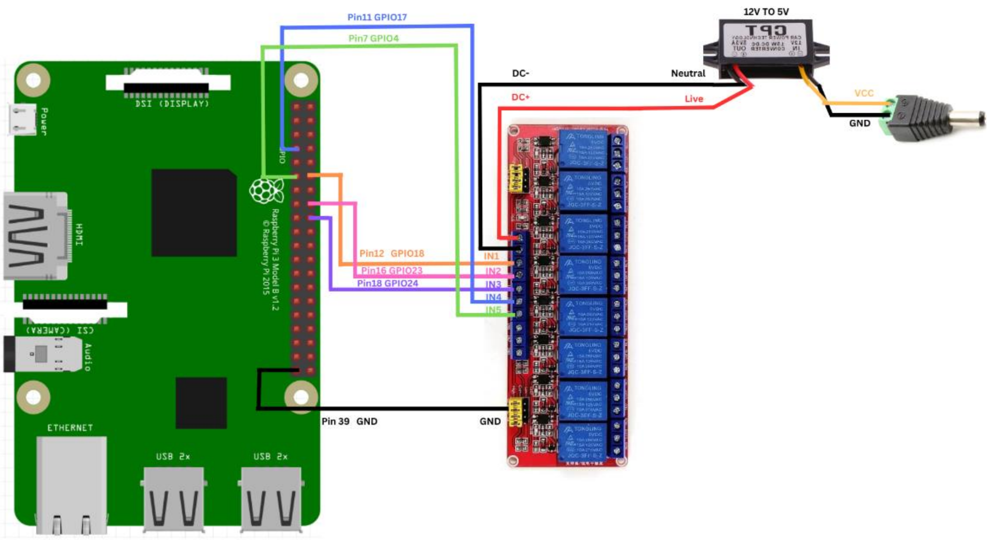


Figure 5.1.3: The connection of Raspberry Pi with 8-Channels relay module

Furthermore, there is an MCP3008 which is an ADC is connected with the Raspberry Pi to read the value from 4 analogue sensors, which include TDS sensor, Water Level sensor, and Current sensors. There are 7 pins from Raspberry Pi are required to use for the SPI connection with the MCP3008, which are pin 6 (GND), pin 1 (3.3V), pin 2 (5V), pin 23 (SCLK), pin 21 (MISO), pin 19 (MISO), and pin 24 (CE0).

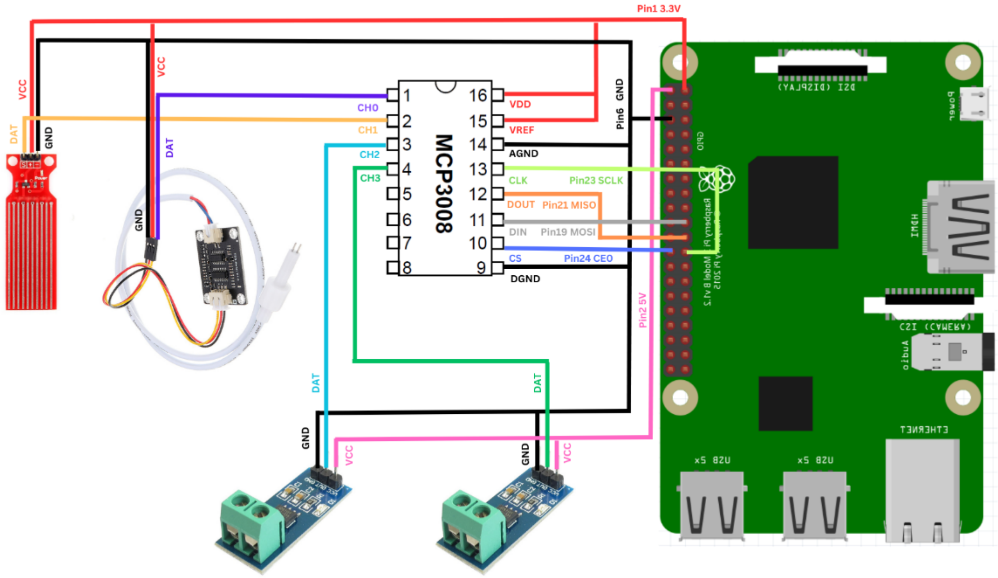


Figure 5.1.4: The connection of Raspberry Pi with MCP3008 and the analog sensors

In term of convenient, the wiring has been done in the breadboard for easy management. The analog sensors such as water level, TDS, and current sensor has been connected to the Raspberry Pi via MCP3008. Then, the MCP3008 will convert the analog value that collected from those sensors to digital value which is readable by the Raspberry Pi.

5.2 Software Setup

In this project, the set up for the software has been separated into 3 sections, which are Raspberry Pi, Laptop and Telegram.

Software Setup for Raspberry Pi

Before starting up the Raspberry Pi, the user is required to install the Raspberry Pi Operating System (OS) to a microSD card and attach it with the microcontroller. The Raspberry Pi OS can be easily installed by using the Raspberry Pi Imager which can download from the official website, ***https://www.raspberrypi.com/software/***.

Figure 5.2.1: User interface of Raspberry Pi Imager

After the Raspberry Pi has been booted up, the user now can install Node-RED with Node JS 14 LTS by entering the following command at the terminal.

$ bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)

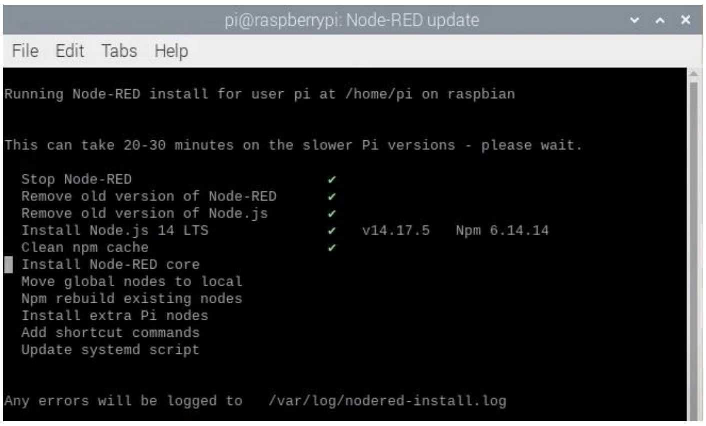


Figure 5.2.2: Installation process of Node-RED tool

Once the Node-RED installation process is completed, run the Node-RED console and access to the Node-RED panels by typing ***http://localhost:1880/*** in the browser.

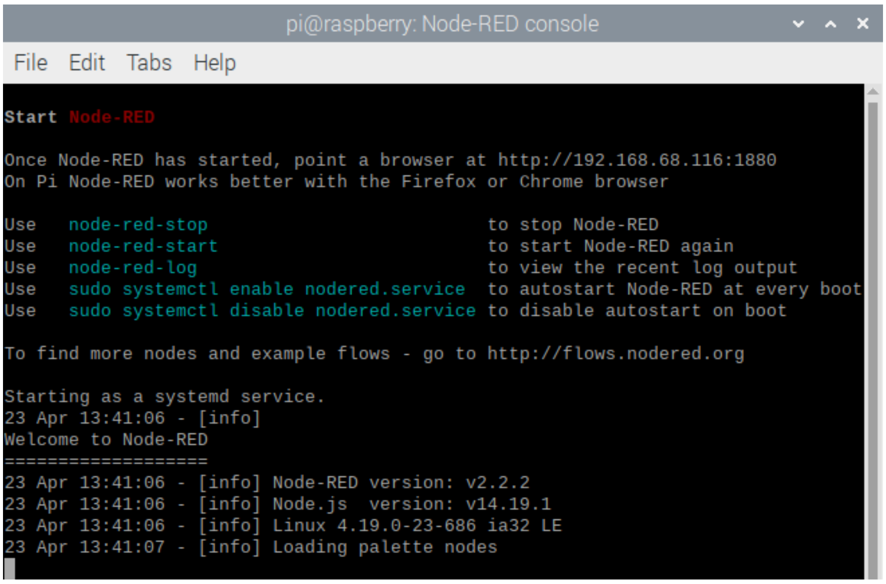


Figure 5.2.3: Node-RED console

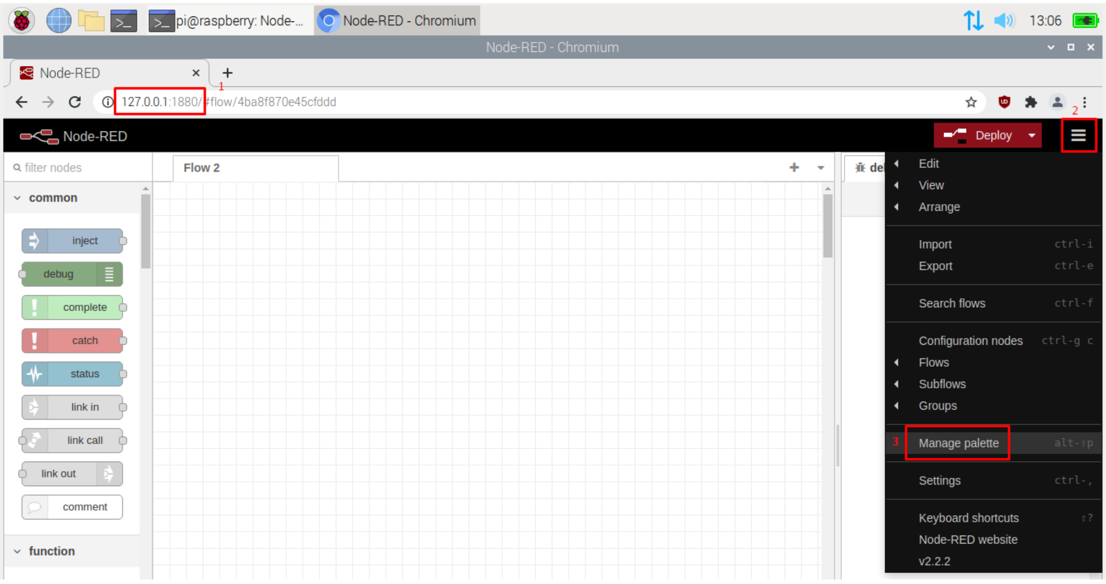


Figure 5.2.4: Node-RED panels

In order to ensure the system to work properly, there are few palettes that has been shown below are required to install by the user.

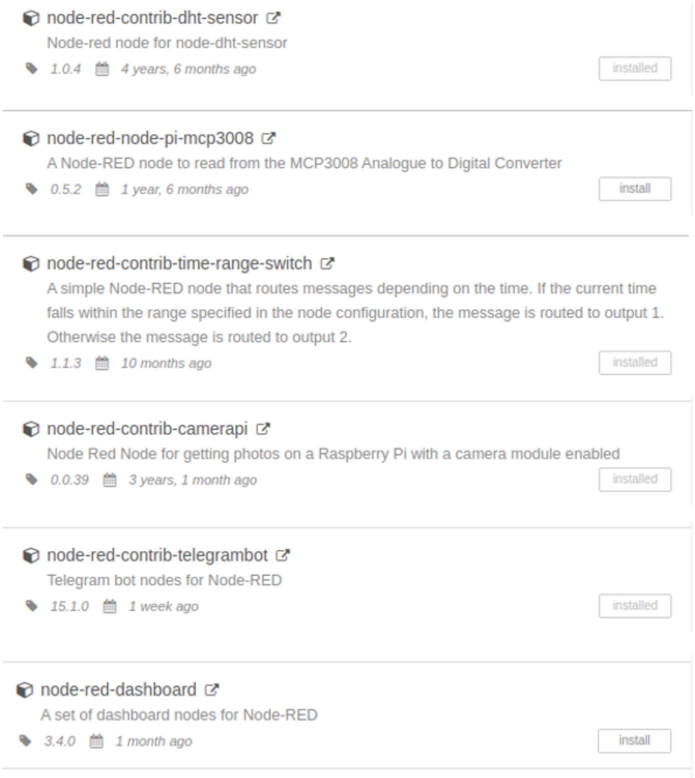


Figure 5.2.5: Installed palettes in Node-RED tool

Software Setup for Laptop

Since the laptop will be used as the monitoring device that receive data from Raspberry Pi, it has to install Node-RED and InfluxDB to support the system.

Firstly, download wget (64-bit EXE) from ***https://eternallybored.org/misc/wget/***.

Download InfluxDB by typing the following command in the command prompt.

wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.0.7-windows-amd64.zip

Once the download is completed, unzip the folder. Open a command prompt and enter the command *cd **C:\files\influxdb2*** to access the folder and execute the file by typing ***influxd*** to run InfluxDB as local host.

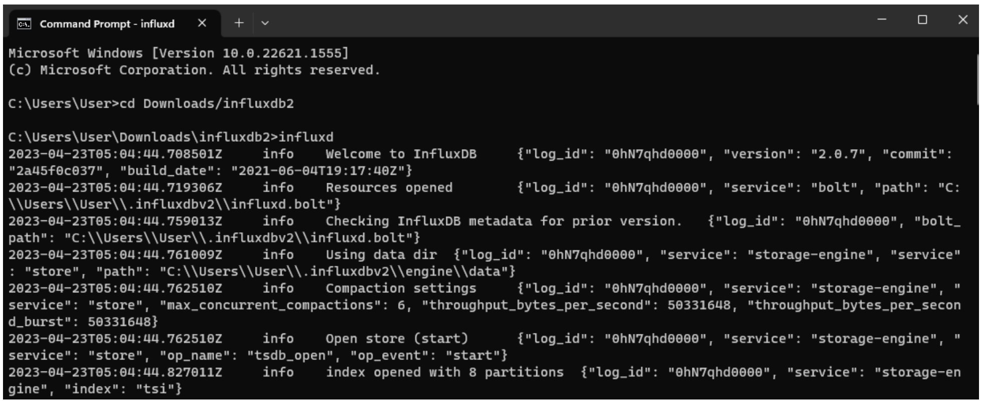


Figure 5.2.6: Command prompt that running InfluxDB

The user now can access InfluxDB UI by type ***http://127.0.0.1:8086/*** in the browser.

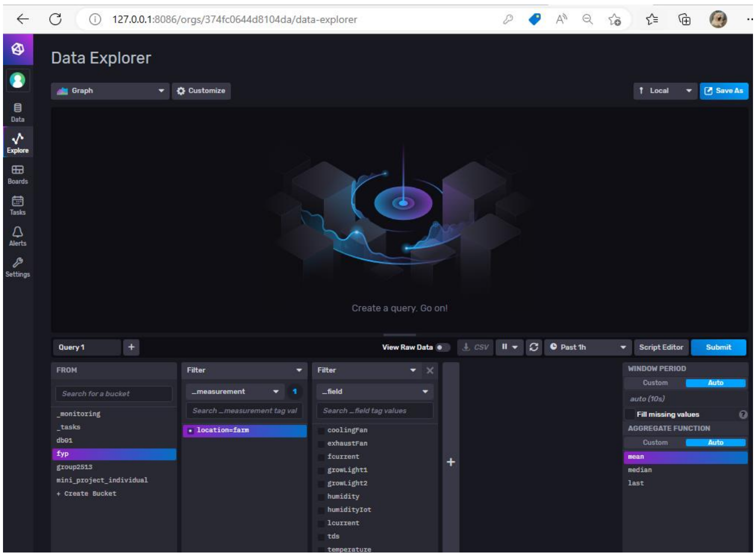


Figure 5.2.7: InfluxDB platform

To install Node-RED in laptop, the user has to access *https://nodejs.org/en/* and download the **Recommended For Most Users** version and install it.

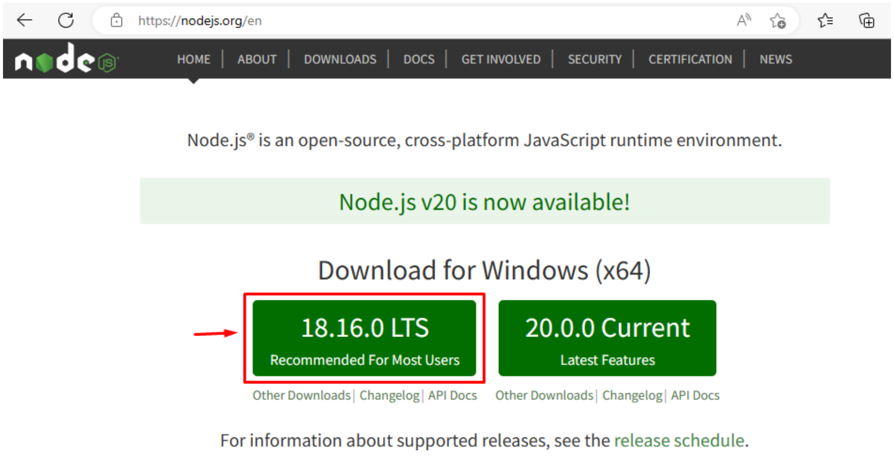


Figure 5.2.8: Noje.js download platform

Then, open a command prompt and type ***npm install -g --unsafe-perm node-red*** to install Node-RED. After the installation, type ***node-red*** to run the Node-RED tool.

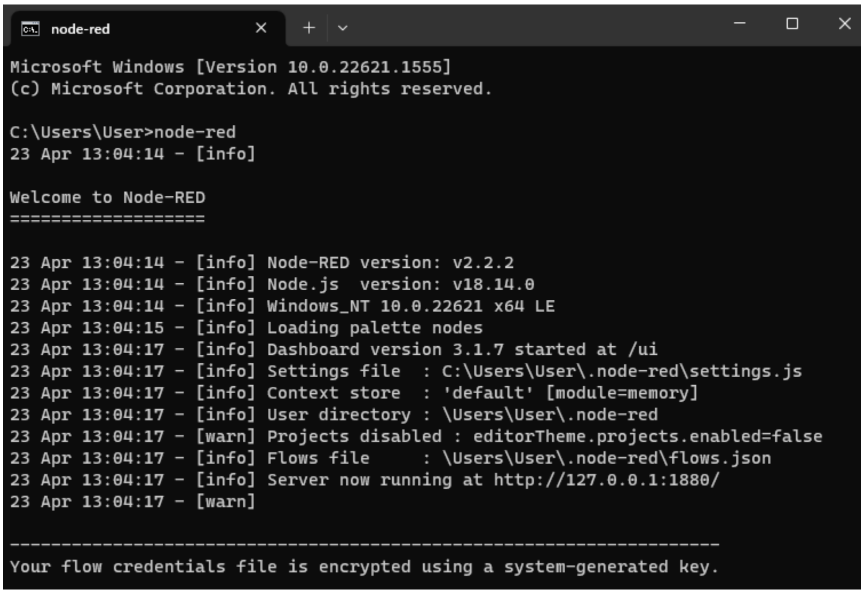


Figure 5.2.9: Node-RED console from laptop view

Open a browser and connect to Node-RED tool by typing ***http://127.0.0.1:1880/***.

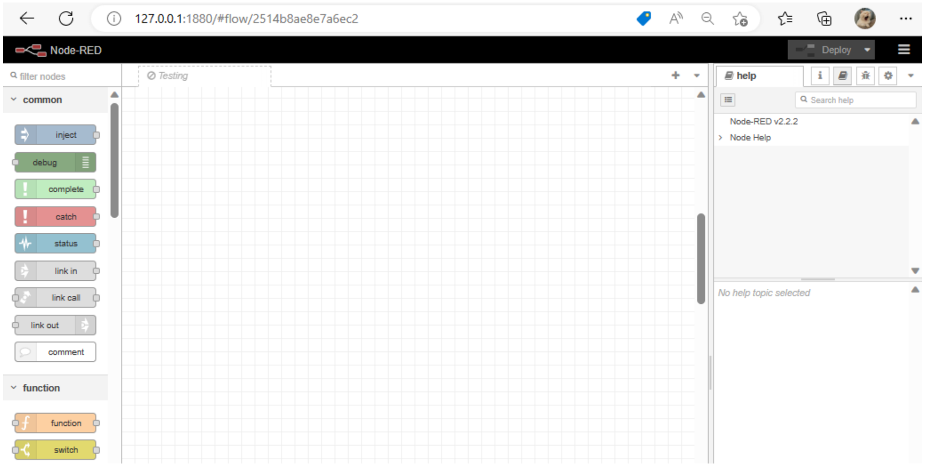


Figure 5.2.10: Node-RED panels from laptop view

Software Setup for Telegram

In order to create a Telegram chatbot, the user is required to install the Telegram application from the Google Playstore either in their desktop or smartphone. After the installation, search for Bot Father and start the conversation by typing ***/start***.

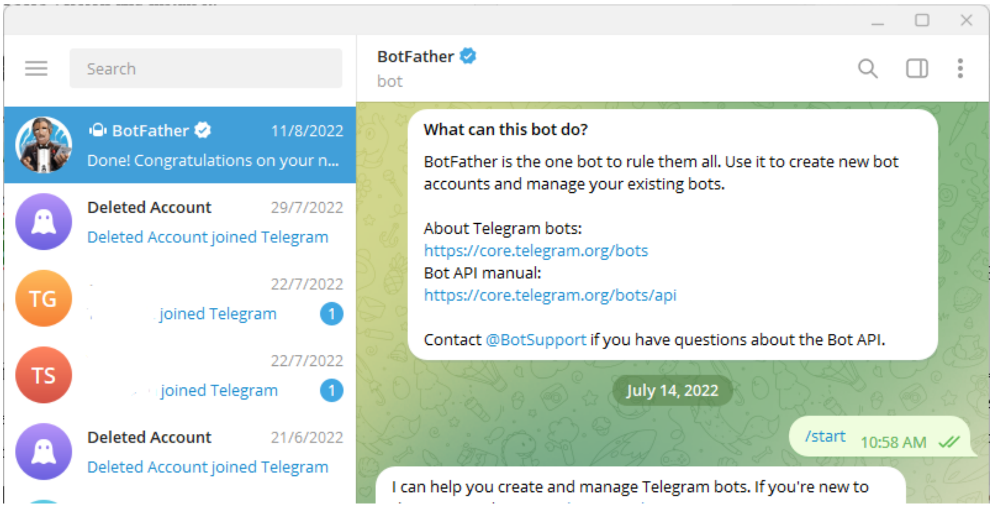


Figure 5.2.11: Bot Father conversation in Telegram

After that, enter the command /newbot to start the procedure of creating a Telegram chatbot. Once the chatbot has been created, save the token that provided by Bot Father for further usage.

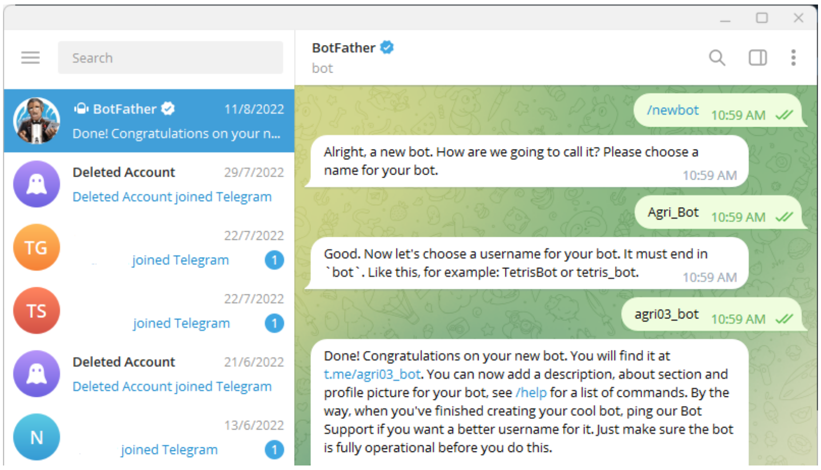


Figure 5.2.12: Telegram chatbot creation process

5.3 Setting and Configuration

Setting and Configuration in Raspberry Pi

Before the configuration, there are few interfaces that are required to enable by the user for convenient purpose, which is Camera, SSH, VNC and SPI. Camera interfaces is required to use for enabling the connection between Raspberry Pi and Pi Camera to perform capture or recording images. Besides, SSH and VNC has to be enabled so that the user can access the Raspberry Pi by command prompt or UI control remotely. Lastly, the SPI interface is enabled for connection between Raspberry Pi and MCP008.

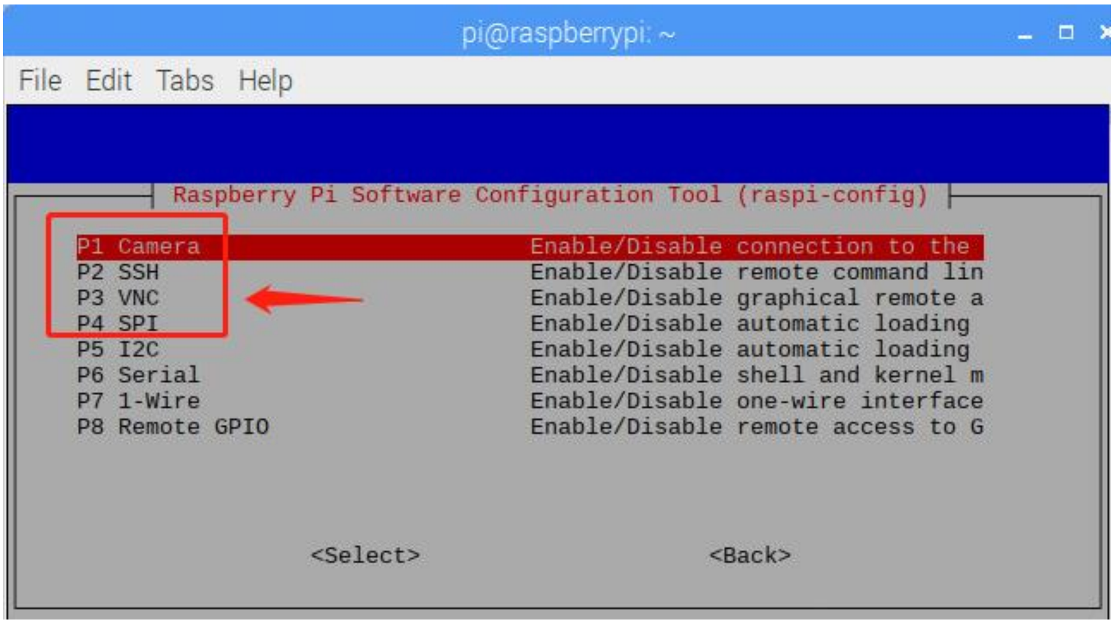


Figure 5.3.1: Configuration Page for Raspberry Pi Interface

After the interfaces has been enabled, the user now can write the programming flow of the agriculture monitoring framework by using Node-RED as follow. First, configure the programming flow to collect the temperature and humidity value from DHT22 that placed nearby hydroponic container and IoT devices.

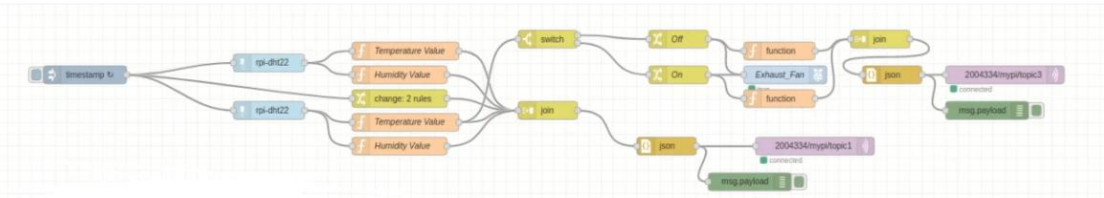


Figure 5.3.2: Programming flow for Raspberry Pi with DHT22 and Exhaust Fan

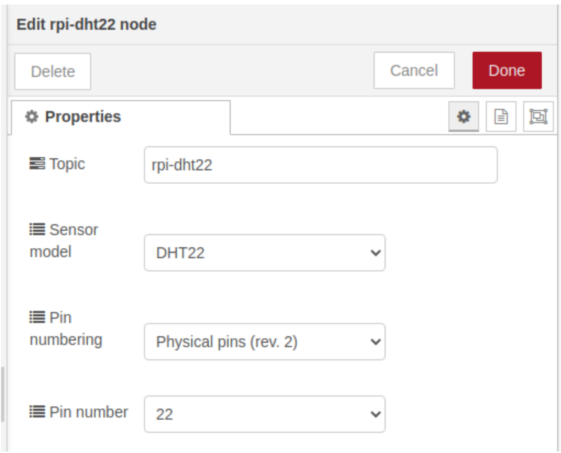


Figure 5.3.3: Configuration of rpi-dht22 node

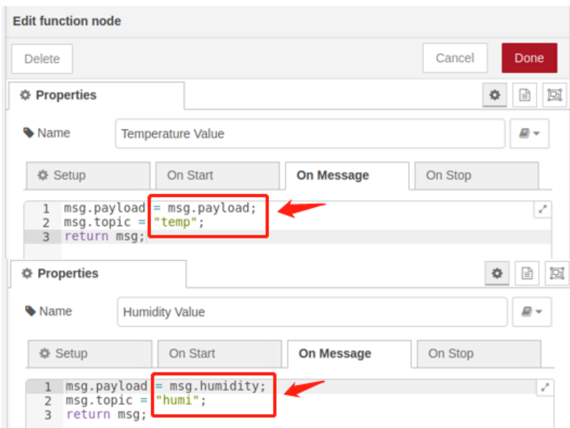


Figure 5.3.4: Configuration of function nodes to collect temperature & humidity value

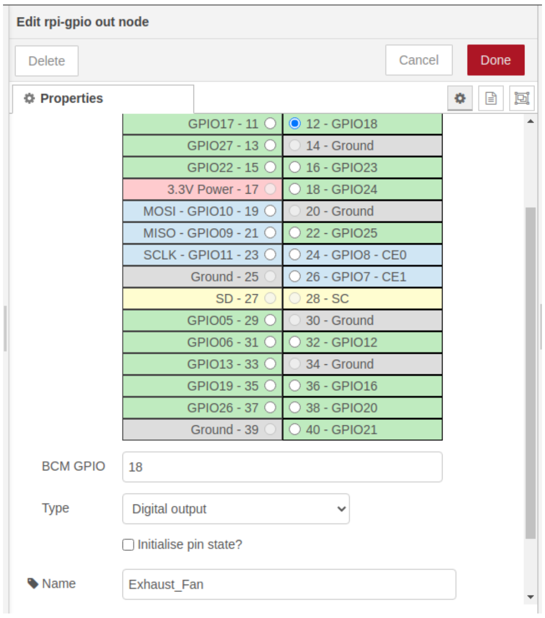


Figure 5.3.5: Configuration of rpi-gpio out nodes to control Exhaust Fan

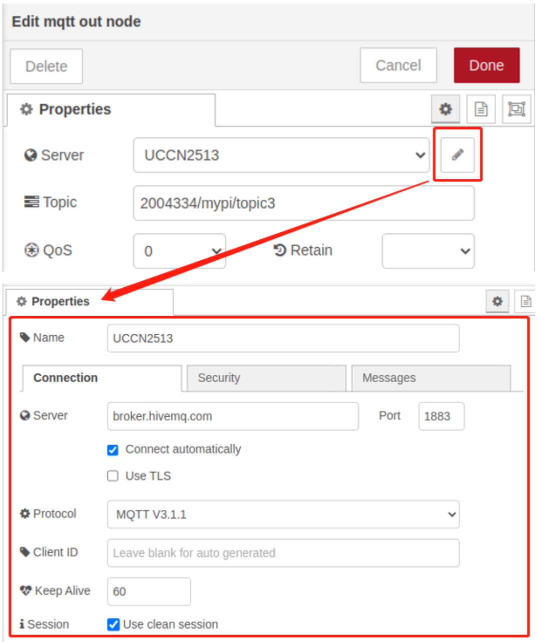


Figure 5.3.6: Configuration of mqtt out node

Secondly, configure the programming flow to collect the data from analog sensors via the ADC which is MCP3008. There will be 4 A/D converter node will be configured for the 4 analog sensors, which are TDS, water level, and 2 current sensors.

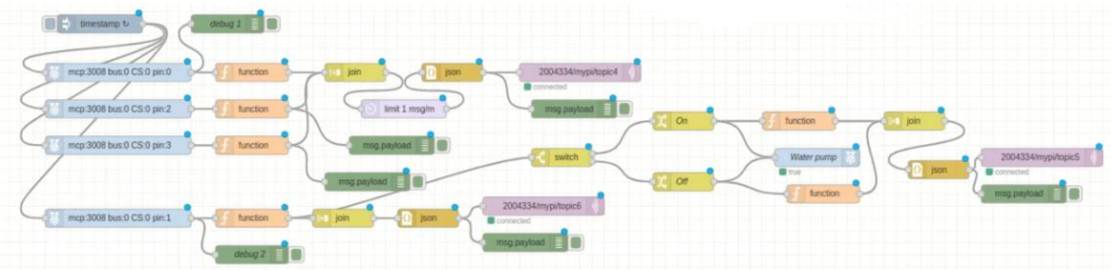


Figure 5.3.7: Programming flow for Raspberry Pi with MCP3008 and Water Pump

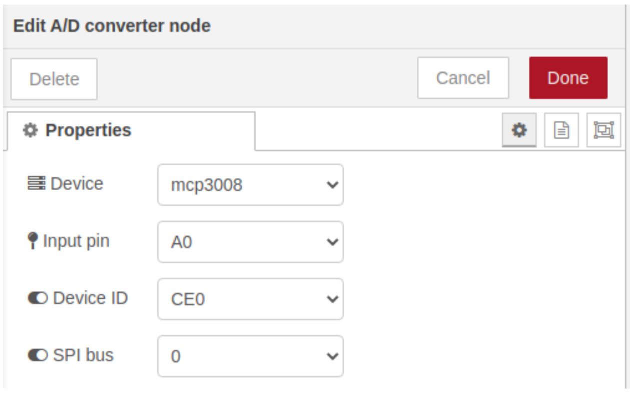


Figure 5.3.8: Configuration of A/D converter node

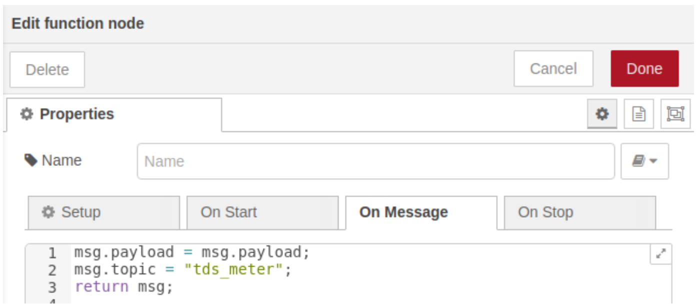


Figure 5.3.9: Configuration of function node

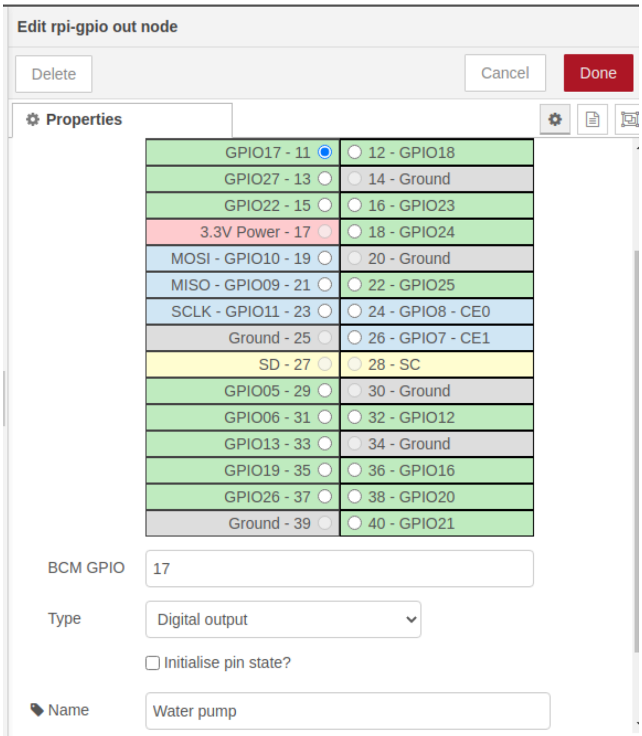


Figure 5.3.10: Configuration of rpi-gpio out nodes to control Water Pump

Thirdly, configure a programming flow which Raspberry Pi will determines the status of the actuators according to the payload value that send by the user via MQTT. There are 2 categories of function nodes, which are automation system and manual control.

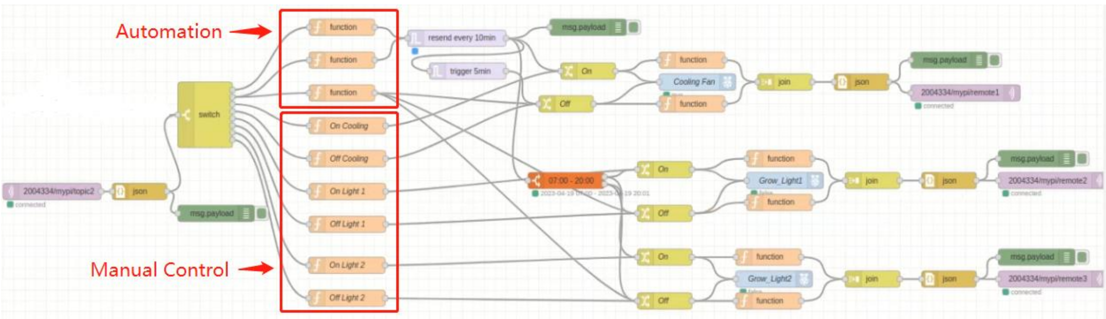


Figure 5.3.11: Programming flow for Raspberry Pi with Cooling Fan and Grow Lights

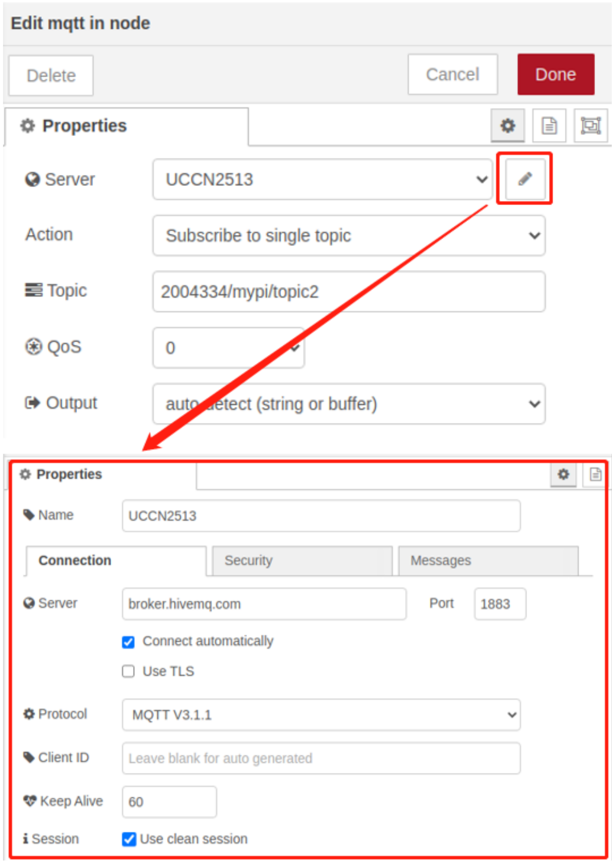


Figure 5.3.12: Configuration of mqtt in node

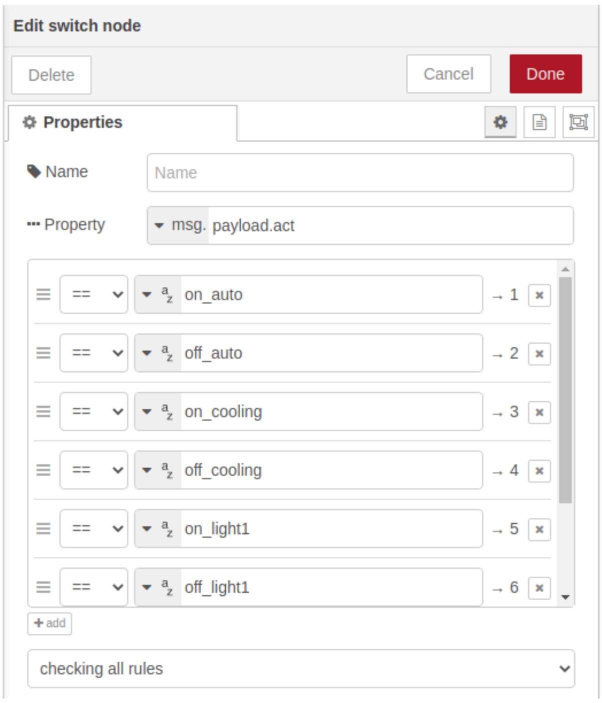


Figure 5.3.13: Configuration of switch node

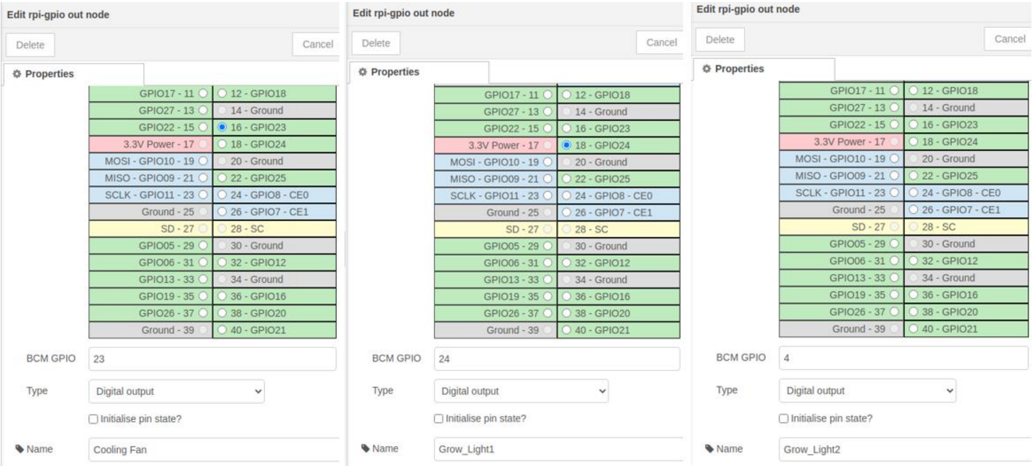


Figure 5.3.14: Configuration of rpi-gpio out nodes to control Cooling Fan and Grow Lights

Lastly, create a programming flow to allow the user to snap the plant photo by sending command to telegram.

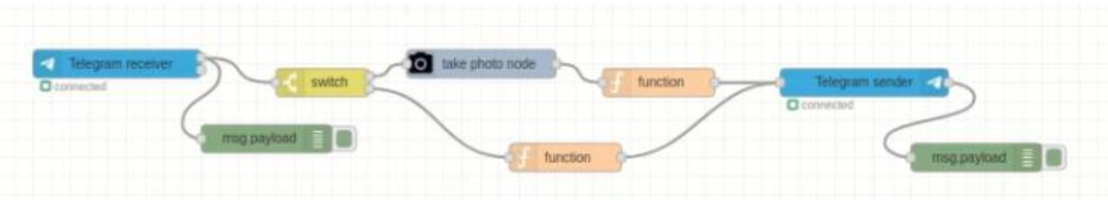


Figure 5.3.15: Programming flow for Telegram Chatbot to snap photo

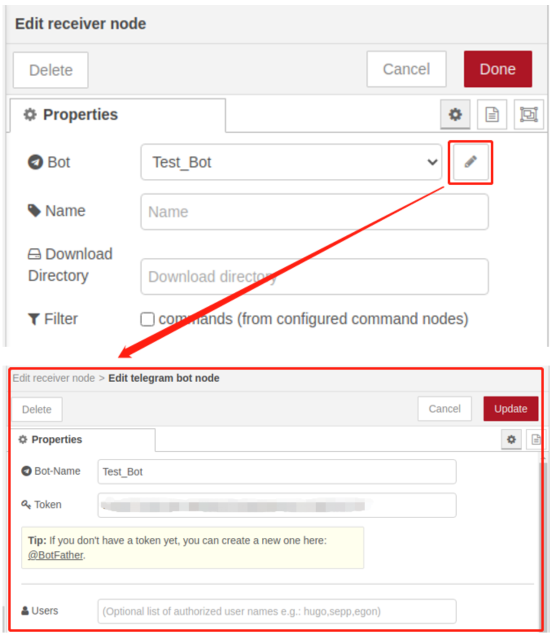


Figure 5.3.16: Configuration of Telegram receiver node

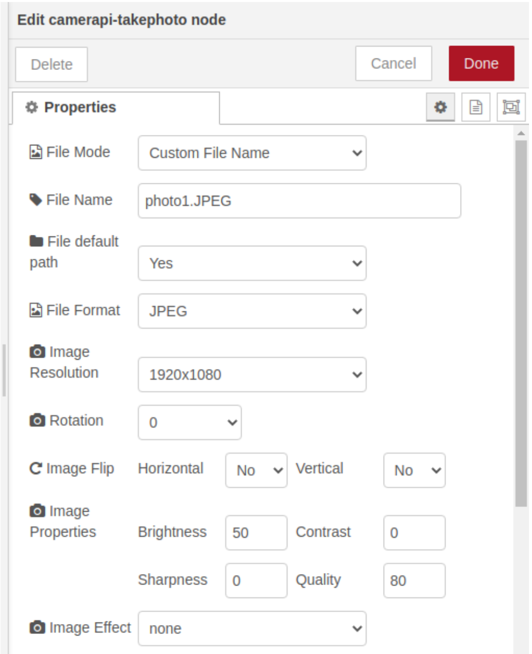


Figure 5.3.17: Configuration of camerapi node

Setting and Configuration in Laptop

As a monitoring device, the user is required to write the programming flow to receive and send data to Raspberry Pi by using Node-RED. First of all, the laptop will be programmed as a receiver to receive the data of sensors and actuators that sent by the Raspberry Pi and further store it into InfluxDB.

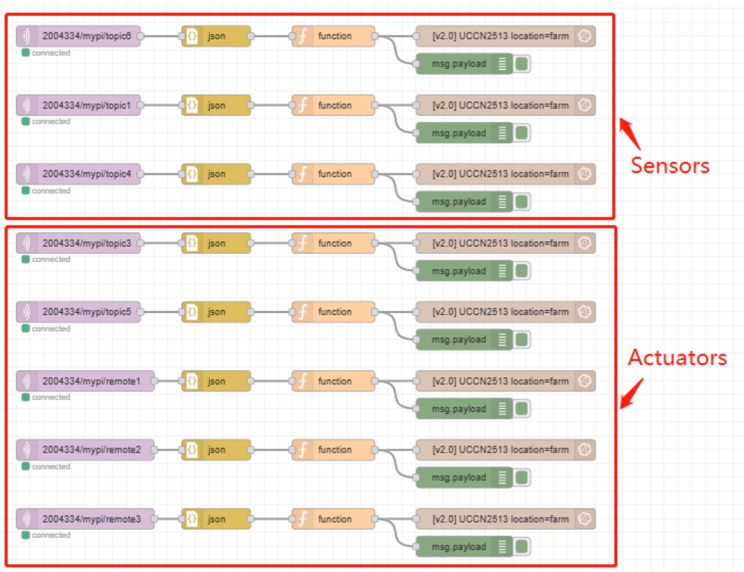


Figure 5.3.18: Programming flow for receiving data from Raspberry Pi and further store in InfluxDB

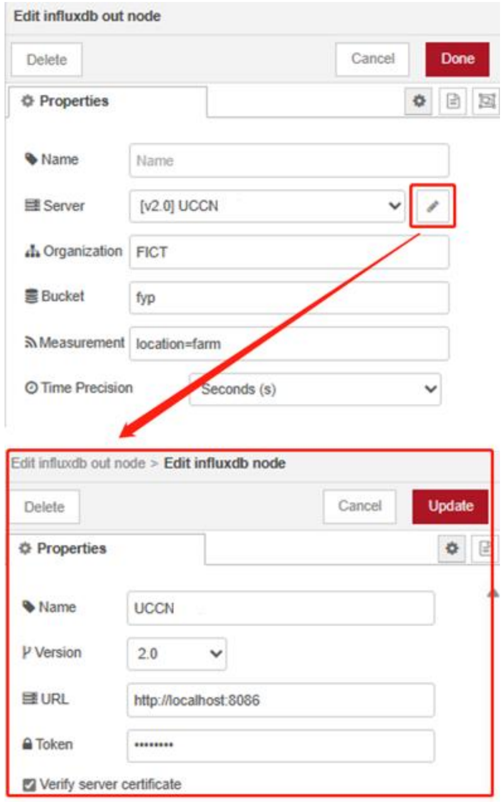


Figure 5.3.19: Configuration of influxdb out node

After that, create a programming flow to pull the sensors and actuators data from InfluxDB and further display it on the Node-RED Dashboard.

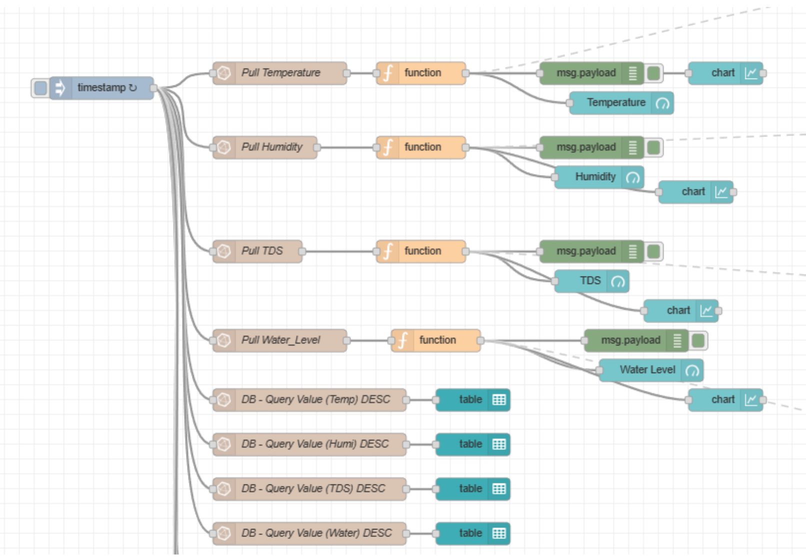


Figure 5.3.20: Programming flow for pulling data from InfluxDB and display on dashboard

Additionally, there are some extension flows for the previous program flow to configure by connecting to a switch to determine the unusual data value and send alert message to the user immediately via Telegram chatbot.

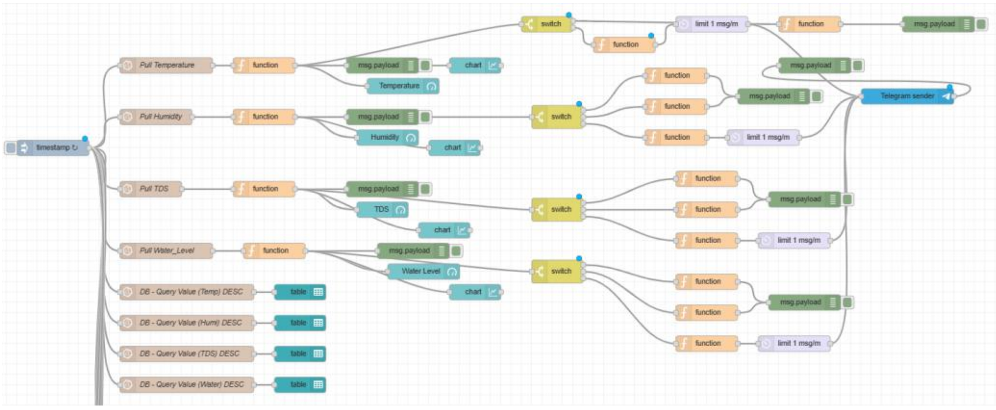


Figure 5.3.21: Programming flow for sending alert message

Besides, the user need to create another program flow that are similar to the previous one to pull the data of sensors from InfluxDB that are used to monitor IoT devices.

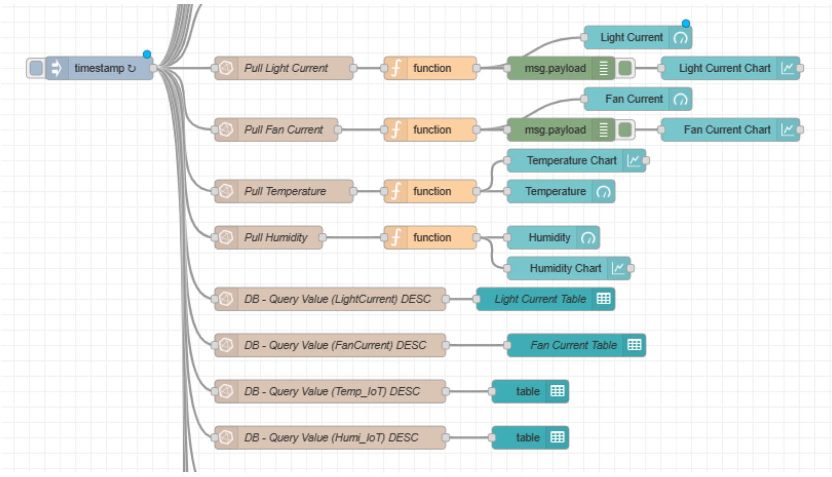


Figure 5.3.22: Programming flow for pulling IoT devices sensors data from InfluxDB

Furthermore, create a program flow for storing and pulling the status of the actuators from the InfluxDB and further display on dashboard by connecting them with switch node. The user is unable to control exhaust fan and water pump since they are fully controlled by the Raspberry Pi for independent automation. In the other way round, the user is able to manually control the cooling fan and both grow lights by clicking the respective switch. There is also an Automation System switch that allows the user to decide whether they want to automate the framework.

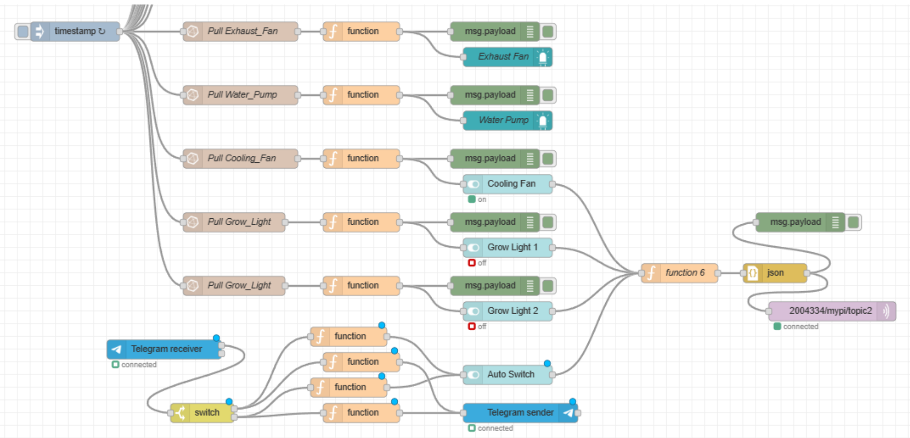


Figure 5.3.23: Program flow for display actuators data on dashboard

Finally, create a program flow that indicating the menu or guideline in Telegram chatbot to enable the user to retrieve the sensors data by only using smart phone.

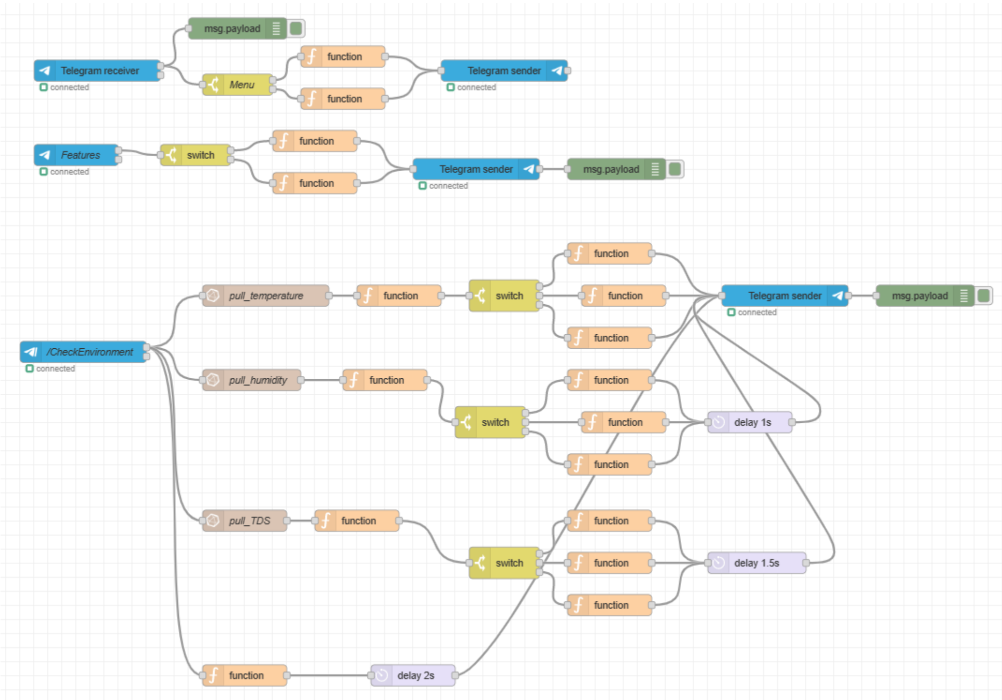


Figure 5.3.24: Program flow for Telegram chatbot menu

5.4 System Operation

There are 3 main aspects that contributes to the whole system to run seamlessly, which are the operation in Raspberry Pi, the laptop that act as a monitoring device, and Telegram chatbot.

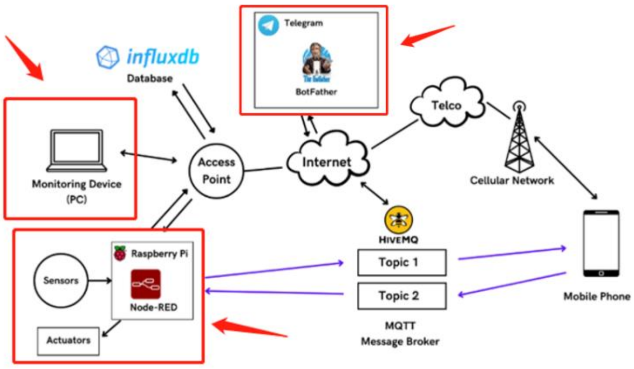


Figure 5.4.1: Architecture diagram of the project

Operation in Raspberry Pi

Firstly, run the Node-RED console and access to the Node-RED panels, the programming flows that has been configured earlier will run automatically.

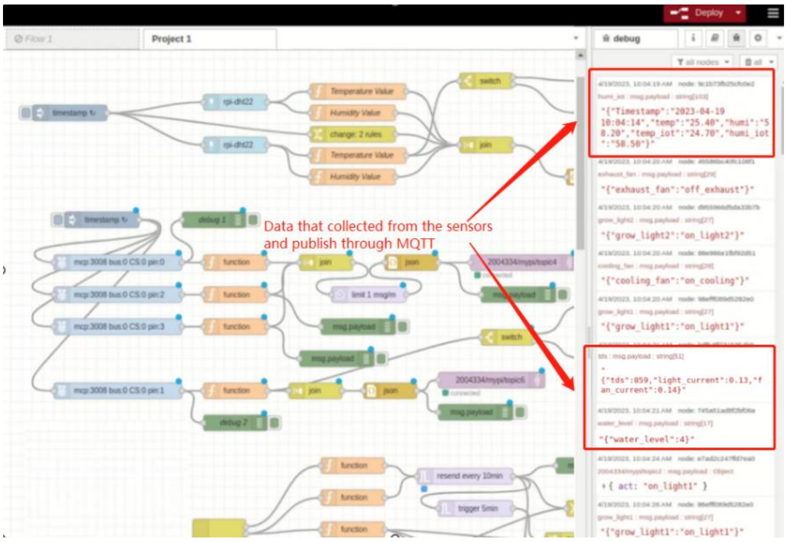


Figure 5.4.2: Node-RED panels in running state

Besides, the live streaming operation is also happening in Raspberry Pi. By connecting to the Pi Camera, it has been used to capture the live feed of the farm environment and stream to YouTube Live by running ***raspivid*** & ***ffmpeg***. By entering the command below, the parameter and orientation of the live feed will be set and stream through rmtp with the help of ffmpeg.

raspivid -o - -t 0 -hf -fps 30 -b 6000000 | ffmpeg -re -ar 44100 -ac 2 -acodec pcm_s16le -f s16le -ac 2 -i /dev/zero -f h264 -i - -vcodec copy -acodec aac -ab 128k -g 50 -strict experimental -f flv rtmp://a.rtmp.youtube.com/live2/enter-stream-key

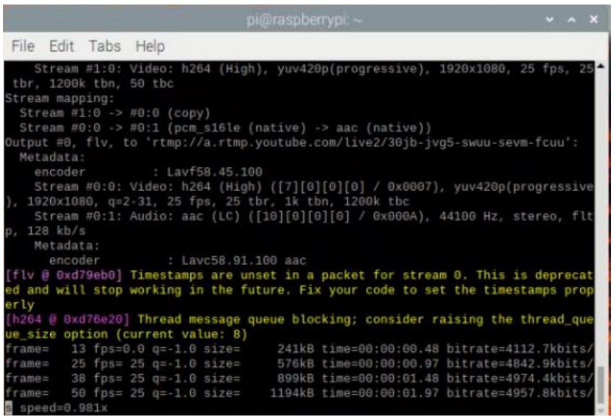


Figure 5.4.3: Streaming Process to YouTube Live

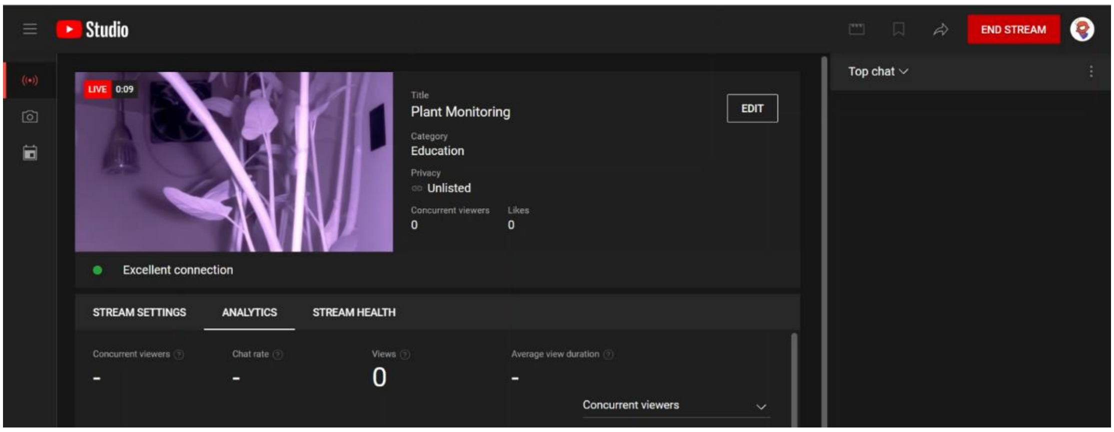


Figure 5.4.4: YouTube Live control room

Operation in laptop

Firstly, run the database which is InfluxDB in localhost by entering the command ***influxd*** in its directory. Node-RED will store the data of sensors and actuators that received from Raspberry Pi into the database first before further process.

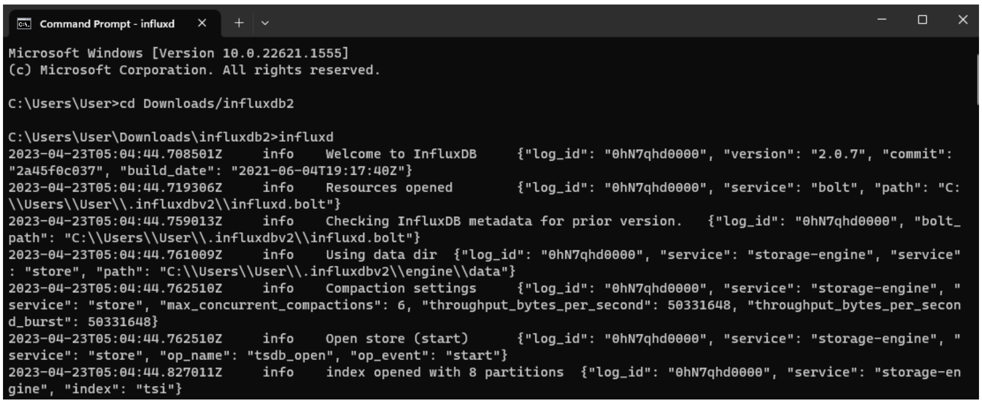


Figure 5.4.5: Command prompt that running InfluxDB

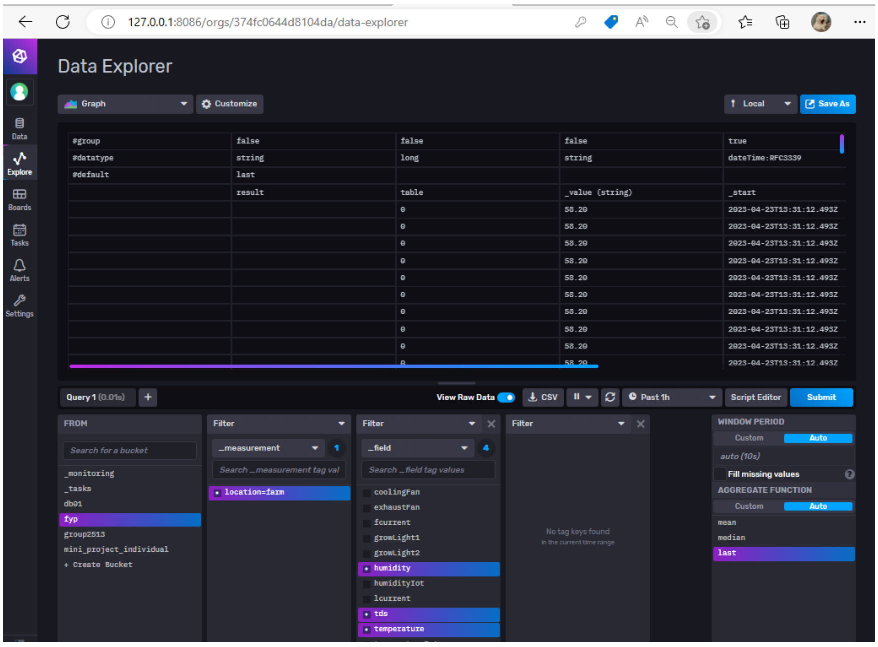


Figure 5.4.6: User Interface of InfluxDB

Secondly, type the command ***node-red*** in the command prompt (***cmd.exe***) to run the Node-RED tool in local host.

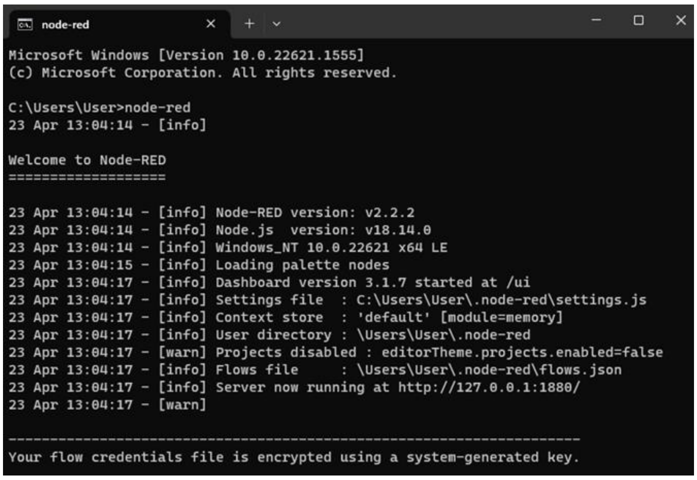


Figure 5.4.7: Command prompt that running Node-RED

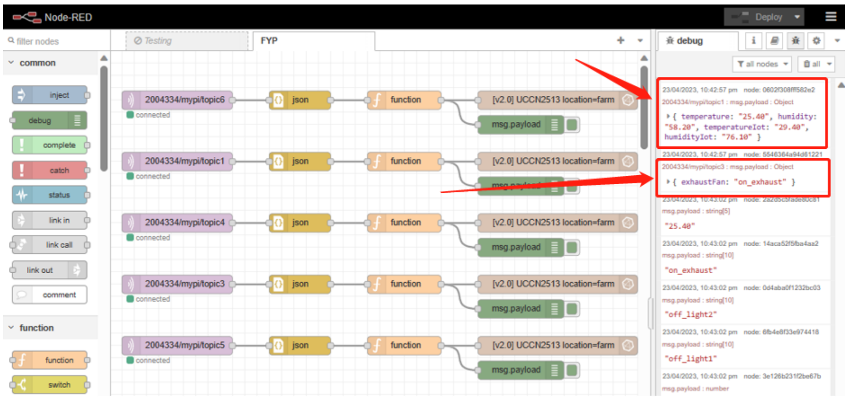


Figure 5.4.8: Node-RED panels that in running state from laptop

Other than that, the Live Stream video from YouTube has been embedded into the html template node from Node-RED Dashboard by using YouTube embedded function.

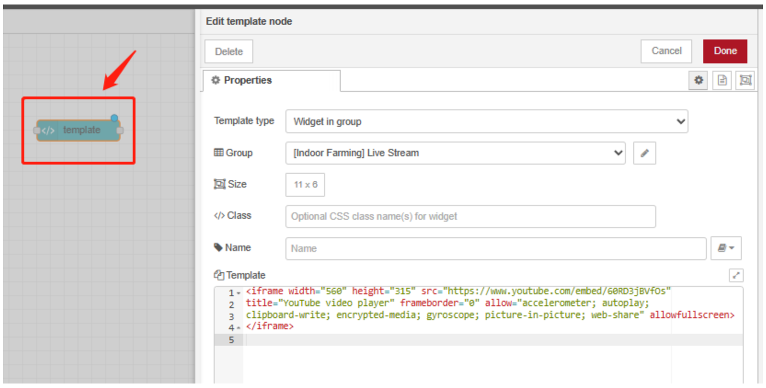


Figure 5.4.9: Template node from Node-RED Dashboard

By accessing ***http://127.0.0.1:1880/ui*** via the browser, now the user can see the agriculture monitoring interface and IoT device monitoring interface that build on Node-RED dashboard.

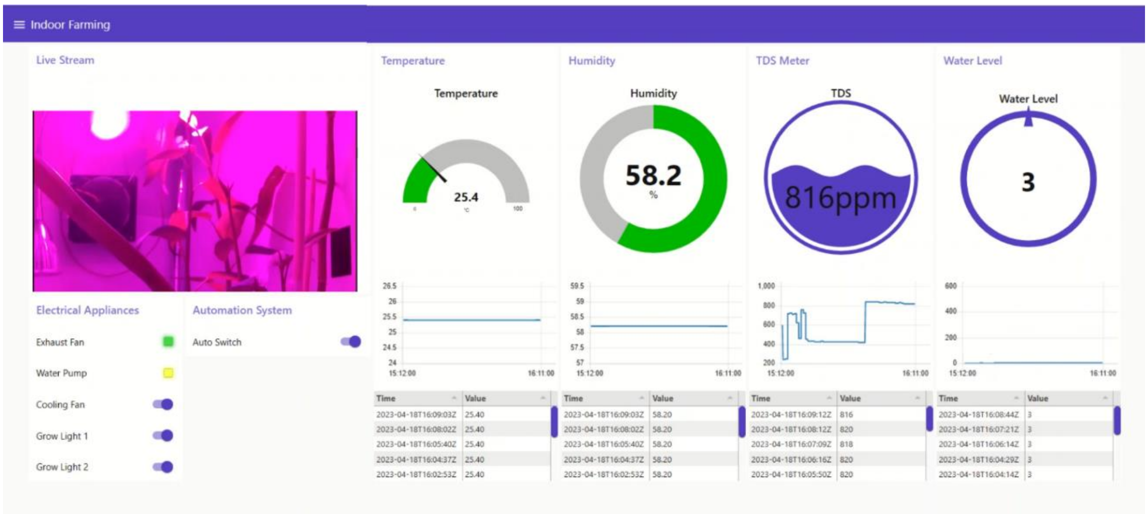


Figure 5.4.10: Node-RED dashboard that showing farm condition

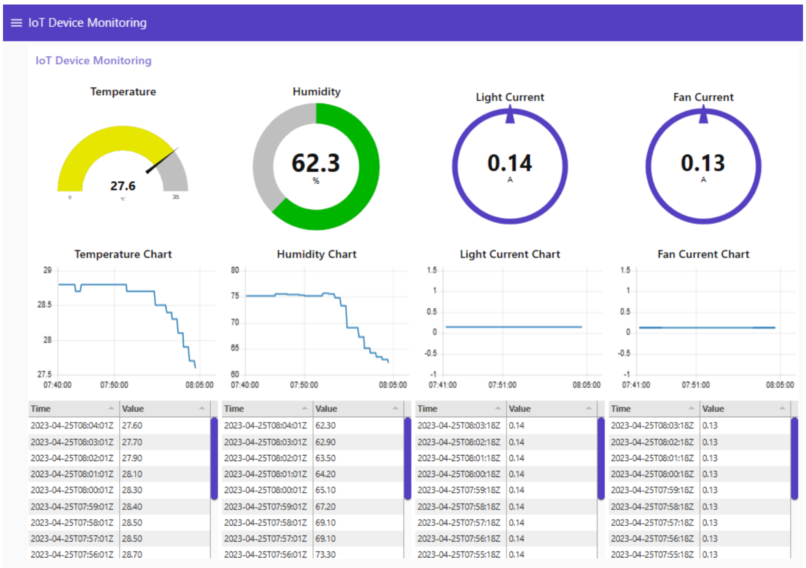


Figure 5.4.11: Node-RED dashboard that showing IoT device condition

Operation in Telegram

Firstly, the user needs to access Telegram application via laptop or smart phone. They can start the conversation with chatbot by entering command ***/begin***. After that, the user needs to enter subsequence command based on the information they want to retrieve.

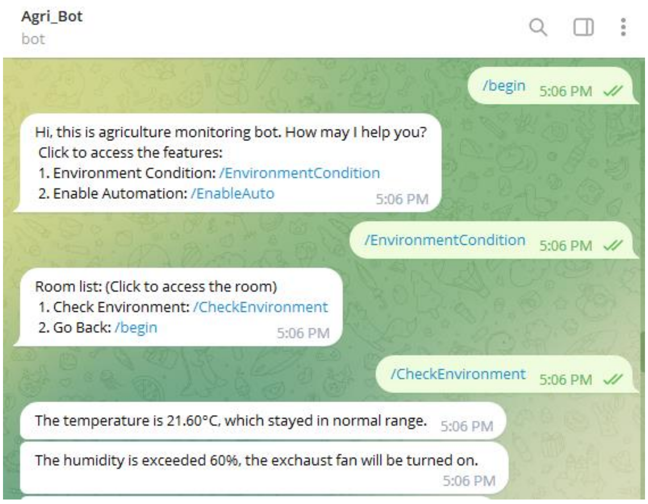


Figure 5.4.12: Operation in Telegram chatbot

Besides, the Telegram chatbot will send alert message immediately when the unusual data has been detected.

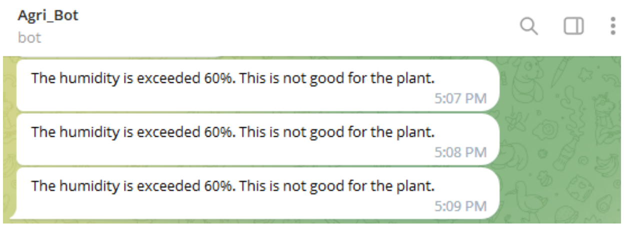


Figure 5.4.13: Alert message from Telegram chatbot

5.5 Concluding Remark

In this chapter, the set-up procedure of the hardware and software for the Raspberry Pi and laptop has been described clearly. Besides, the setting and configuration that has been conducted is also stated so the next person who interested on this topic can build the same framework conveniently. Lastly, the screenshot of the system flow has been provided so the user can understand how the system works accordingly.

CHAPTER 6

System Evaluation and Discussion

6.1 System Testing and Performance Metrics

In order to ensure the stability of the system in daily basis, system testing will be conducted to analyse the operation and performance of the system. In this project, there are three metrics has been selected to evaluate the performance of the system, which is the accuracy of the sensors, the continuity of live feed streaming and the growing condition of the water spinach.

Firstly, the accuracy of the sensors is very important for agriculture monitoring system because it will directly reflect the condition of the farm. In this project, the temperature, humidity, TDS, water level and current value will be collected by the respective sensors for indicating the environment of the framework. The value of data will directly affect the performance of system since it will be used to decide the operation of actuators.

Besides, the continuity of live feed streaming is also crucial to the project since the user is expecting to get the real-time information of the farm immediately. If there are unusual incident happens in the farm, then the person in-charge can immediately observe the situation in real-time and have enough time to handle the incident.

Lastly, the growing condition of the water spinach is the main indicator for this project since it will direct reflect the effectiveness of the framework. There are few conditions will be observed during the process, which is the height of plant and the number of leaves.

6.2 Testing Setup and Result

Three metrics has been selected to evaluate the performance of the system, which is the accuracy of the sensors, the continuity of live feed streaming and the growing condition of the water spinach.

The accuracy of the sensors

Comparing the temperature and humidity value that measured by DHT22 & Thermometer.

DHT22		**Thermometer**	
Temperature (℃)	**Humidity (%)**	**Temperature (℃)**	**Humidity (%)**
24.7	57.0	24.4	55.0
25.4	58.2	25.2	57.0
26.3	60.1	26.0	59.0
27.8	65.0	27.2	63.0
29.3	76.0	29.1	75.0

Table 6.2.1: Comparison between DHT22 & Thermometer

Comparing the Total Dissolved Solids (TDS) and Electric Conductivity (EC) value that measured by TDS sensor & TDS meter.

TDS Sensor		**TDS Meter**	
TDS (ppm)	**EC**	**TDS (ppm)**	**EC**
816	1632	786	1572
820	1640	802	1604
818	1636	795	1590
821	1642	803	1606
819	1638	801	1602

Table 6.2.2: Comparison between TDS sensor and TDS meter

Comparing the current value that measured by Current sensor and Multimeter.

Current Sensor	Multimeter
Current (A)	**Current (A)**
0.11	0.14
0.10	0.13
0.14	0.13
0.13	0.14
0.13	0.13

Table 6.2.3: Comparison between Current sensor and Multimeter

Overall, there are small range of differences between the actual value that measured by the sensors and the measuring tools, thus the sensors that has been used in the project is consider reliable.

The continuity of live feed streaming

Based on the live streaming record that can be viewed from the content section, the range of the live streaming period for every stream is quite different. The continuity of live streaming is depending on the Raspberry Pi that running the ***raspivid*** and ***ffmpeg*** commands. Besides, the stability and the bandwidth of the connected network would also affect the quality of the live stream.

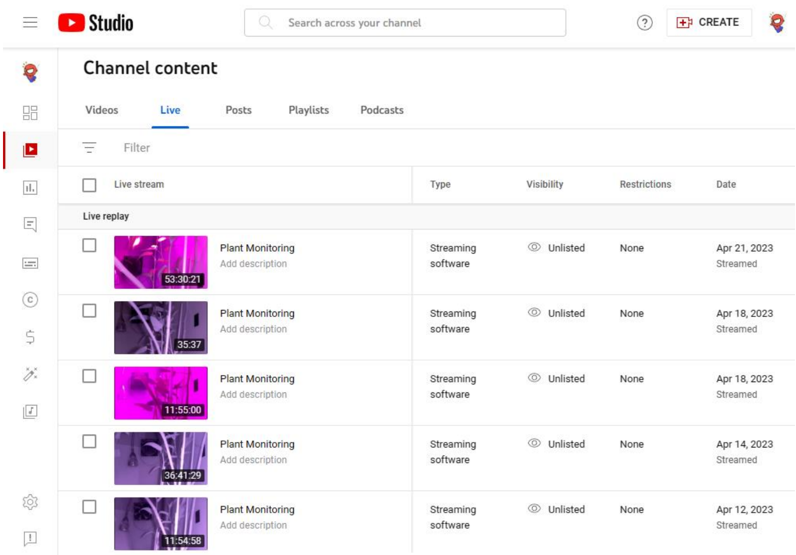
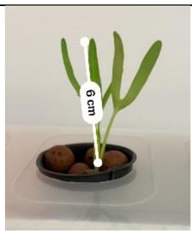
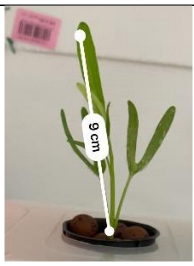


Figure 6.2.1: Live streaming records from YouTube

The Raspberry Pi is placed in the Final Year Project (FYP) lab and connected to the lab Wi-Fi which shared among the students who doing their project in there. Based on the observation, most of the live streaming session has been stopped during the daytime. In contrast, the live streaming went smoothly during night-time which the time that there are nobody using the Wi-Fi. As shown in the record, the longest live streaming period was maintained for around 53 hours. It indicates that the live streaming can last as long as the user want based on the network bandwidth that may consumed by other users.

The growing condition of the water spinach

The growing conditions of the water spinach has been measured based on the length of plant and the number of leaves.

Week	Conditions	Length (cm)	Number of leaves
1		4	4
2		6	4
3		9	6
4		15	8

5		17	8
6		24	8
7		26	10
8		30	10

Table 6.2.4: Growing conditions of water spinach

Based on the observation, the data from the table proved that the framework is suitable for the water spinach to grow since the length of the plant and the number of leaves is increasing in a healthy way.

6.3 Project Challenges

In the project development process, it is not possible that everything will run smoothly, and the developers should always be ready to face the problems that would occur. Throughout the whole testing and executing phases, there are some challenges has been faced, which are the plantation knowledge, environment condition, and network connectivity.

Firstly, the user must be equipped with basic plantation knowledge to ensure the plant has fulfilled the basic requirement to grow healthily. Planting is not as easy as ABC, the user needs to consider several aspects before starting, such as what kind of plant is suitable for soil or hydroponic medium, which kind of fertilizer or soil mixing should be used as the base for planting.

Other than that, the environment condition such as temperature and humidity value is beyond human control. The framework that used for planting must be placed in a right place, so the environment condition is friendly for the plant to grow. In this project, the framework is placed in the FYP lab which the air conditioner and ventilation system only will be operating on weekdays, thus the temperature and humidity value will be higher on the weekends. In order to decrease the impact of environment condition to the plant, the actuator such as exhaust fan has been attached to the framework by decreasing the temperature and humidity value.

Lastly, the network connectivity to the internet is also a challenge for the project since the data transmission between the Raspberry Pi and monitoring device is depending on the network to operate. Once there is a network issue, the connection between the Node-RED in laptop and Raspberry Pi will be disconnected and the system will stop operating. Same to the live streaming on YouTube, the live feed recording will stop immediately once there is no internet connection. At the end of day, the user may have to perform troubleshooting by configure the hardware physically in order to make the system operate again.

6.4 SWOT

Strengths

By using Node-RED, the development process has become simple and straight forward since the user only require minimum effort on programming setup to build the system. Besides, the system also supports data visualisation which build on top of Node-RED dashboard. All the data that collected from sensors and actuators includes the live streaming video will be displayed on the dashboard tidily which convenient for user to monitor.

Weakness

There are also some uncontrollable factors which would affect the performance of the system. Firstly, the data transmission between the Raspberry Pi and monitoring device is highly dependent on the network connectivity. Besides, the uncontrollable environment condition such as temperature and humidity in where the framework has been placed in would also affect its operating performance.

Opportunities

The framework is developed by using the accessories and components which is affordable by most of the people. This would encourage the people who interested in the related project to take the first step and start to build their own indoor farming framework. In term of compatibility, the Node-RED contains of various type of palette modules that support the types of connection between the Raspberry Pi and the sensors or actuators.

Threats

Since the framework is connected to the wireless network that operating at the environment where it has been placed, there is a possibility where the connection status might be affected due to the number of network users and the unpredictable factors such as lightning strike and network error. Besides, the uncontrollable environment condition would also affect the system performance since the temperature of the IoT devices would increase if the air circulation in the room is not functioning.

6.5 Objectives Evaluation

In chapter one, the objectives of the project have been listed out to act as the direction that the system should follow to achieve the goals.

1. **Integrate IoT technologies in the agriculture monitoring framework.**
   The objective has been successfully achieved because the framework is built by the connection between the microprocessor Raspberry Pi with the sensors and actuators. By running the Node-RED program that has been designed, the system is operating by collecting the data from the sensors that further transmit it to the remote device that act as a receiver and process the data.
2. **Enable the automation feature in the framework.**
   The objective has partially achieved since the framework can run automatically and require least manpower to manage. Once the automation function is enabled, the actuators such as grow light, exhaust fan and water pump will operate based on the value that collected from the respective sensors. In the other hand, the framework still requires minimum manpower to change the water and mix the nutrient solution for the plant to grow healthily.
3. **Create an all-time operating chatbot via Telegram.**
   The objective has been successfully achieved. The Telegram chatbot has been treated as an alternative way for the user to monitor the framework without accessing the monitoring device. The user can get the latest information of the framework by entering specific command to the chatbot. If there is unusual data is collected from the sensors, the Telegram chatbot will send alert message to the user immediately.

6.6 Concluding Remark

In this chapter, there are few metrics has been listed to measure the performance of the system. There are few tests has been conducted and the result is stated in the report clearly. Besides, the challenges that has been faced throughout the whole development process is discussed. Lastly, a SWOT analysis about the project has been conducted and the achievement of project objectives has been evaluated.

CHAPTER 7

Conclusion and Recommendation

7.1 Conclusion

Throughout the research, the project indicates to develop a self-sufficient farm that operate in indoor environment to solve the limited land for agri-food issue and ensure the production of the crops. In the existing agriculture monitoring system, some problems that will affect the effectiveness of the system has been encountered, which is lack of data insights, highly relied on manpower, and lack of countermeasure to handle unusual incident.

In this project, there are few objectives has been set for system improving, which is integrate IoT technologies in the monitoring framework, enable the automation feature, and create an all-time respond chatbot. Besides, the project has been scoping down to develop a small framework by integrating Raspberry Pi with Node-RED, HiveMQ, InfluxDB and Telegram. After the project is completed, it is expected that some contribution will be made to the user, which is provides useful data insights, reduce manpower efficiently and monitor the framework conveniently with the chatbot.

Besides, some of the related technologies that will be used to build the framework has been studied and examined, such as hardware platform, firmware OS, database and programming language. Same as planning, the technologies such as Raspberry Pi, Node-RED and InfluxDB will be used in this project after reviewed. Other than that, there are four existing system has been studied to generate more ideas on the project. The strength and weakness of those systems has been emphasized so that it can improve the implementation of this project.

Furthermore, there are four development model has been examined in order to make sure the project can be developed on time, which is waterfall model, iterative model, spiral model and Kanban model. Among those models, Kanban development model has been chosen for the project since it is very flexible for the user to add in new requirement no matter in which development phases. For the system requirement, the hardware and software that will be used in the project has been listed clearly. The functional requirement also has been stated to ensure the direction of the project development is correct. Then, the project milestone for FYP1 and FYP2 has been stated clearly in the respective tables to ensure the progress of the project development can be achieved. Since some hardware will be used in the project, the estimated cost of the respective items and their prices is listed clearly.

Moreover, the architecture of the designed system has been drawn and explained to ensure that the reader will know how the whole system is going to work. The connection between the Raspberry Pi and the hardware has been drawn and the details has been mentioned carefully. Additionally, the functional modules of the system have been written, which is data collection, automation control, data virtualization and Telegram chatbot. Next, a system flow diagram has been drawn to indicate the incident that will happen during the system process. Finally, the GUI design of the system from the Node-RED dashboard and Telegram chatbot has been taken and paste in the report.

Additionally, the set-up procedure of the hardware and software for Raspberry Pi and laptop including the setting and configuration has been stated so the next person who interested on this topic can build the same framework conveniently. Then, the screenshot of the system flow has been described clearly so the user can understand how the system works accordingly.

Last but not least, there are few system tests has been conducted and measured by few performance metrics which are the accuracy of the sensors, the continuity of live feed streaming and the growing condition of the water spinach. The result of the system testing shows that the system is reliable and suitable for planting. There are also few challenges that has been faced throughout the development process, which are lack of plantation knowledge, environment condition and network connectivity. Finally, a SWOT analysis and objectives evaluation has been conducted to ensure the system is completed and operating normally.

7.2 Recommendation

Although the agriculture monitoring framework that has been designed and built in this project is achieved most of its objectives, there are still some improvements can be made to further extend its utilization purposes for the farmers.

- **Automate the dosing system for the nutrient solution.**
  In the current framework, it still requires manpower to mix the nutrient solution in the hydroponic container physically after the person in-charge notice that the TDS value is low through the monitoring dashboard. The dosing system is expected to work by mixing the nutrient solution in the hydroponic container automatically once a low TDS value has been detected. By integrating an automated dosing system in the framework, it will become more convenient for the farmers since it will further decrease the burden of manpower.
- **Test the suitability of other crops species in the framework.**
  The crops that have been planted in the framework is water spinach, which the project proved that it also can grow healthily same as the water spinach that grows in outdoor soil medium. In order to further utilize the framework, the user can consider to place the other crops species in the framework and observe their growing condition.

REFERENCES

[1] R. A. Dardak, "Malaysia's Agrofood Policy (NAP 2011-2020)- Performance and New Direction," *FFTC Agricultural Policy Platform (FFTC-AP)*, Apr. 11, 2019. https://ap.fftc.org.tw/article/1368#:~:text=Malaysia%27s%20Agrofood%20Policy%20(NAP%202011%2D2020)%2D%20Performance%20and%20New%20Direction (accessed Aug. 11, 2022).

[2] O. Robbins, "Hydroponics: How It Works, Benefits and Downsides, & How to Get Started," *Food Revolution Network*, Feb. 24, 2021. https://foodrevolution.org/blog/hydroponics/#:~:text=Hydroponics%20is%20a%20type%20of%20agriculture%20or%20gardening (accessed Aug. 12, 2022).

[3] R. P. (Trading) Ltd, "Raspberry Pi 3 Model B+," *Raspberry Pi*. https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/ (accessed Aug. 12, 2022).

[4] Arduino.cc, "Arduino Uno Rev3," Arduino Online Shop. https://store-usa.arduino.cc/products/arduino-uno-rev3?selectedStore=us (accessed Aug. 13, 2022).

[5] R. P. (Trading) Ltd, "Raspberry Pi OS," *Raspberry Pi*. https://www.raspberrypi.com/software/ (accessed Aug. 13, 2022).

[6] B. Anderson and B. Nicholson, "SQL vs. NoSQL Databases: What's the Difference?," www.ibm.com, Jun. 15, 2021. https://www.ibm.com/cloud/blog/sql-vs-nosql (accessed Aug. 15, 2022).

[7] A. Nair, "Introduction to InfluxDB: A time-series database," wearecommunity.io, Mar. 21, 2021. https://wearecommunity.io/communities/india-java-user-group/articles/891 (accessed Aug. 15, 2022).

[8] Technostacks, "Node.js vs Python: Which Should Be Used In 2022?," Technostacks Infotech, Sep. 22, 2021. https://technostacks.com/blog/node-js-vs-python/ (accessed Aug. 18, 2022).

[9] H. Mishra, "Python vs NodeJS — Which is Best for your Web Application?," Geek Culture, Oct. 29, 2021. https://medium.com/geekculture/python-vs-nodejs-which-is-best-for-your-web-application-bd21a33da0eb (accessed Aug. 18, 2022).

[10] K. Otale, "IOT Based Modern Agriculture System Using Raspberry Pi," www.academia.edu, vol. 4, no. 5, Jan. 2017, Accessed: Aug. 19, 2022. [Online]. Available: https://www.academia.edu/65271047/IOT_Based_Modern_Agriculture_System_Using_Raspberry_Pi

[11] P. Choklikitamnuay and P. Netinant, "DESIGN SMART HOME HYDROPONIC GARDENING SYSTEM USING RASPBERRY PI 3 1," International Journal of Electrical, Electronics and Data Communication, vol. 7, no. 7, Jul. 2019, Accessed: Aug. 19, 2022. [Online]. Available: http://www.iraj.in/journal/journal_file/journal_pdf/1-583-156819209229-33.pdf

[12] V. David, V. India, R. K. Duraiswamy, H. Ragu, and S. P, "IoT based Automated Indoor Agriculture System Using Node-RED and IBM Bluemix," 2021 6th International Conference on Inventive Computation Technologies (ICICT), Jan. 2021, Accessed: Aug. 20, 2022. [Online]. Available: https://www.academia.edu/78955528/IoT_based_Automated_Indoor_Agriculture_System_Using_Node_RED_and_IBM_Bluemix

[13] S. K. Selvaperumal, R. Lakshmanan, M. D. Guedi, S. Perumal, and R. Abdulla, "Automated Smart Hydroponics System Using Internet of Things," International Journal of Electrical and Computer Engineering, vol. 10, no. 6, Dec. 2020. Accessed: Aug. 20, 2022. [Online]. Available: https://core.ac.uk/download/pdf/333845609.pdf#:~:text=The%20proposed%20system%20here%20is%20an%20automated%20smart,interface%20based%20on%20the%20shown%20real%20time%20parameters.

[14] Tutorials Point, "SDLC Waterfall Model," www.tutorialspoint.com, 2019. https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm (accessed Aug. 21, 2022).

[15] W3schools, "SDLC Iterative Model," www.w3schools.in. https://www.w3schools.in/sdlc/iterative-model/#:~:text=What%20is%20SDLC%20Iterative%20Model%3F%20The%20popular%20iterative (accessed Aug. 22, 2022).

[16] “Spiral Model - What is SDLC Spiral Model?,” Software Testing Help, Aug. 07, 2022. https://www.softwaretestinghelp.com/spiral-model-what-is-sdlc-spiral-model/#:~:text=What%20is%20SDLC%20SPIRAL%20MODEL%3F%20The%20spiral%20model (accessed Aug. 30, 2022).

[17] “Kanban | SDLC | DevMaking,” www.devmaking.com, Mar. 29, 2021. https://www.devmaking.com/learn/sdlc/kanban/ (accessed Aug. 30, 2022).

[18] L. M. Engineers, “Getting Started with ESP8266,” *Last Minute Engineers*, Jan. 25, 2023. https://lastminuteengineers.com/getting-started-with-esp8266/ (accessed April. 20, 2023)

APPENDIX A

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 2
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE

- Review the weakness of current system
- Add water level sensor and water pump to the system

2. WORK TO BE DONE

- Add pi camera to the system & stream the live feed on YouTube

3. PROBLEMS ENCOUNTERED

- No problem was encountered

4. SELF EVALUATION OF THE PROGRESS

- The progress of the project is on schedule

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 4
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE

- Add pi camera to the system & stream the live feed on YouTube

2. WORK TO BE DONE

- Place the water spinach in the framework

3. PROBLEMS ENCOUNTERED

- No problem was encountered

4. SELF EVALUATION OF THE PROGRESS

- The progress of the project is on schedule

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 6
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE - Place the water spinach in the framework
2. WORK TO BE DONE - Monitor the growing conditions of water spinach - Record the data and condition of water spinach
3. PROBLEMS ENCOUNTERED - No problem was encountered
4. SELF EVALUATION OF THE PROGRESS - The progress of the project is on schedule

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 8
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE - Monitor the growing conditions of water spinach - Record the data and condition of water spinach
2. WORK TO BE DONE - Add DHT22 to the system and place nearby the IoT Devices - Add current sensors by connecting it with actuators
3. PROBLEMS ENCOUNTERED - No problem was encountered
4. SELF EVALUATION OF THE PROGRESS - The progress of the project is on schedule

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 10
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE

- Add DHT22 to the system and place nearby the IoT Devices
- Add current sensors by connecting it with actuators

2. WORK TO BE DONE

- Monitor the stability of the framework
- Working on the FYP2 report

3. PROBLEMS ENCOUNTERED

- No problem was encountered

4. SELF EVALUATION OF THE PROGRESS

- The progress of the project is on schedule

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Trimester 3, Year 3	**Study week no.:** 12
Student Name & ID: Foong Theen Wui, 20ACB04334	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Development of IoT-Based Agriculture Monitoring Framework using Node-RED	

1. WORK DONE

- Monitor the stability of the framework

2. WORK TO BE DONE

- Finalizing the FYP2 report
- Prepare for FYP2 presentation

3. PROBLEMS ENCOUNTERED

- No problem was encountered

4. SELF EVALUATION OF THE PROGRESS

- The progress of the project is on schedule

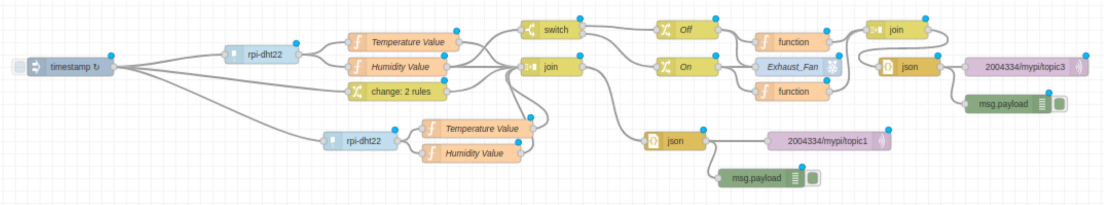
Supervisor's signature

Student's signature

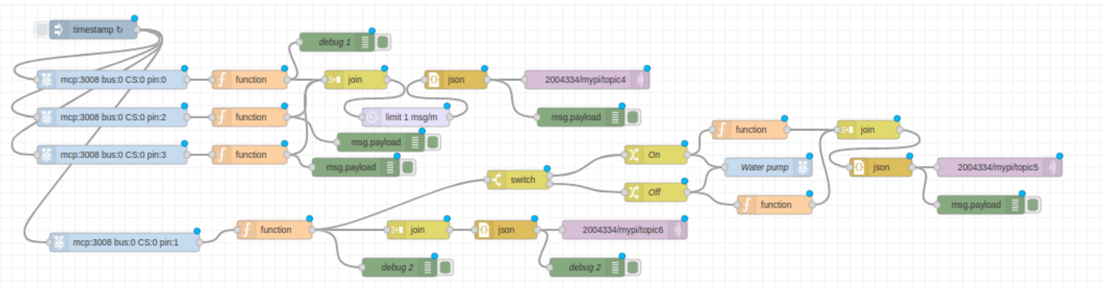
APPENDIX B

The Node-RED flow in Raspberry Pi

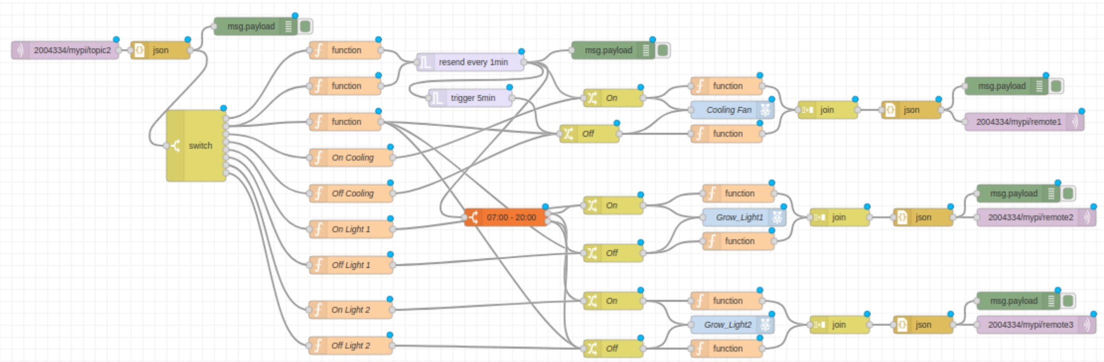
Program flow to collect temperature and humidity value from DHT22 that placed nearby hydroponic container and IoT Devices. The temperature value that received from nearby IoT Devices is further extend and connect with the switch node to control the exhaust fan.



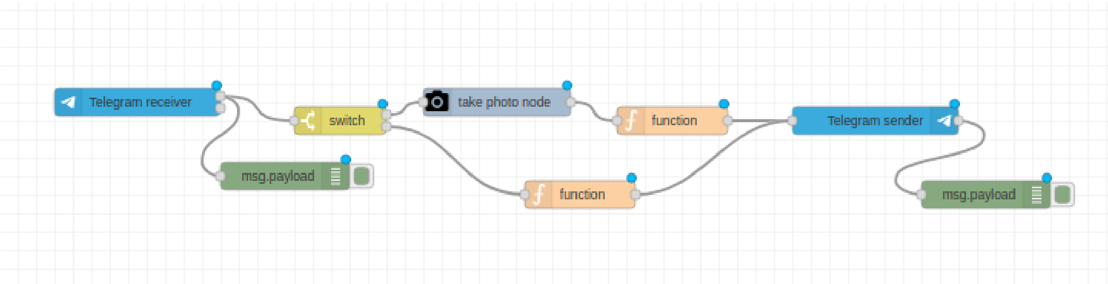
Program flow to read the digital value that converted from the analog sensors via MCP3008. The sensors which include are TDS, water level, and current sensors. The water level is further extend and connect with the switch node to control water pump.



Program flow to receive the controlling action that performed from the remote device via MQTT. There are 3 actuators involved, which is cooling fan and both grow lights. The cooling fan will be turn on and off every 5 minutes to stimulate the wind to blow the crops. Besides, the grow light will turn on at 7am and turn off at 8pm every day.

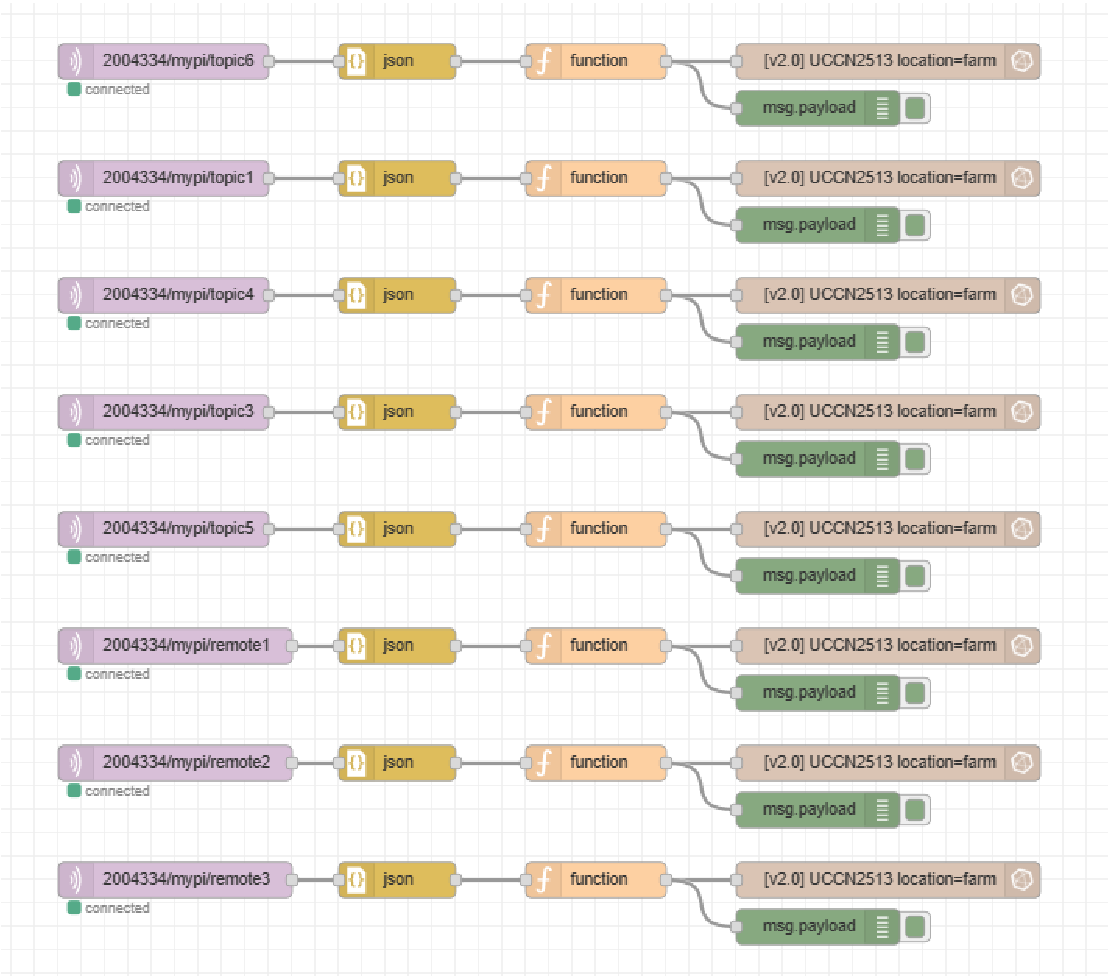


Program flow which allows the user to snap photo by controlling the pi camera via Telegram chatbot.

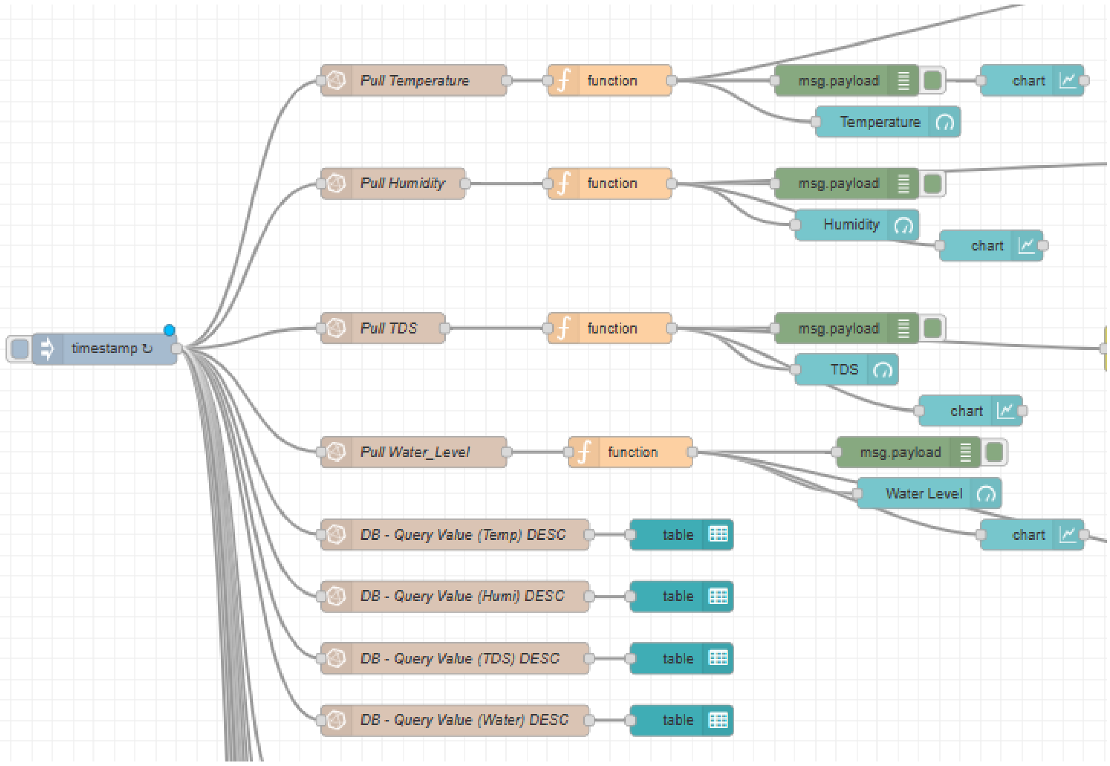


The Node-RED flow in laptop (remote device)

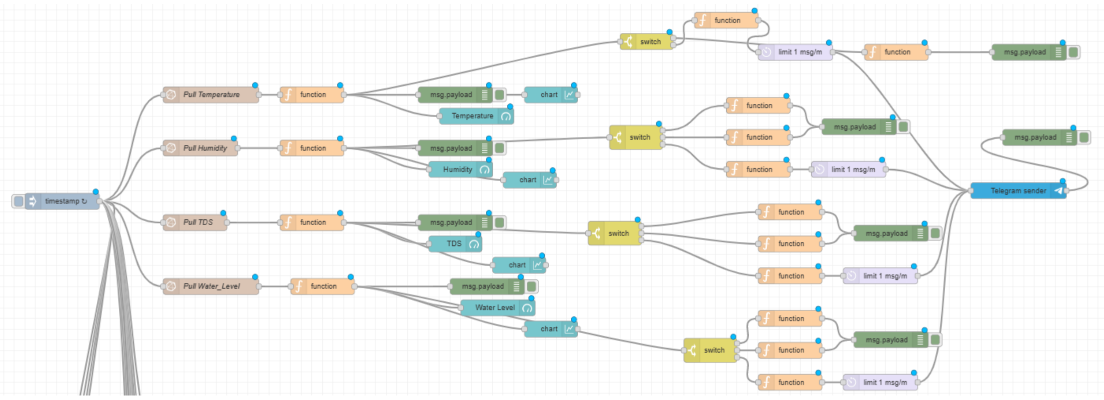
Program flow which receiving the sensors and actuators data that sent by Raspberry Pi and further store it into InfluxDB.



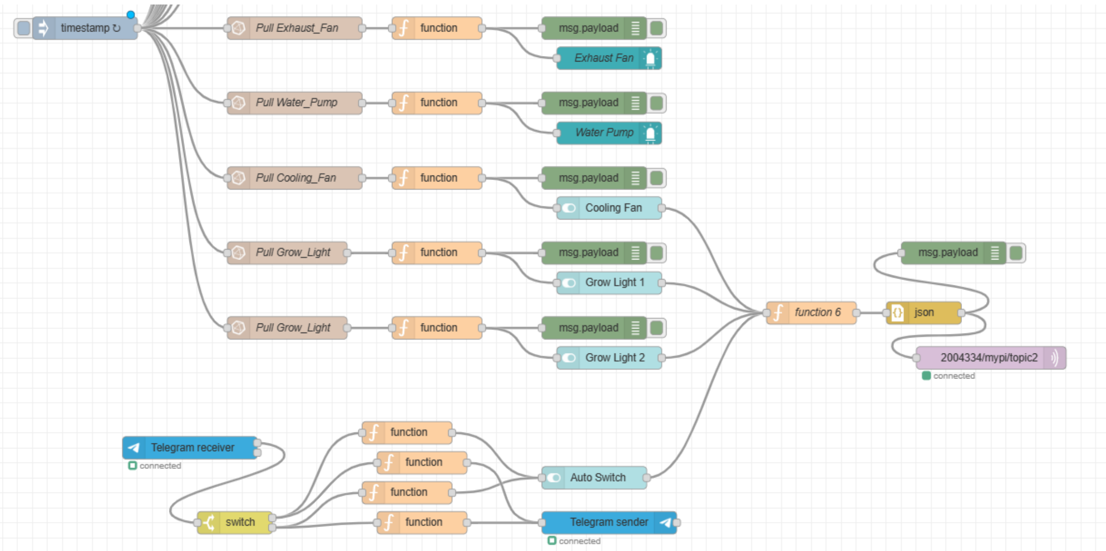
Program flow which pulls the sensors data from InfluxDB and display it on Node-RED dashboard as gauge, chart and table form.



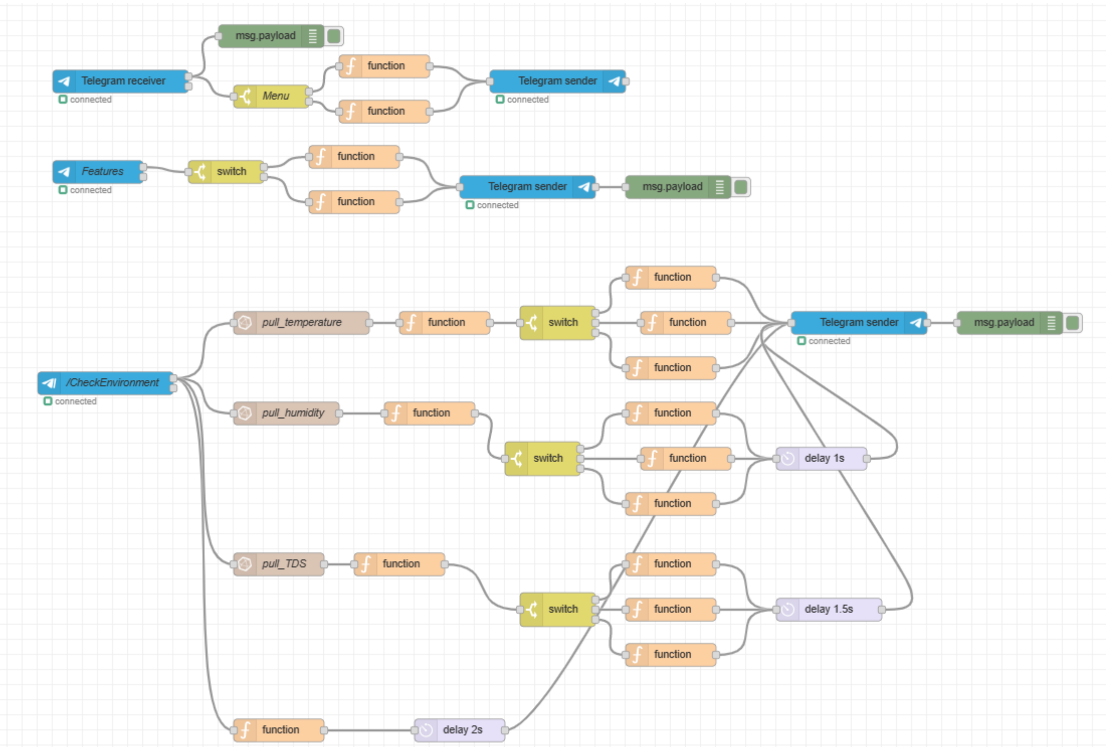
Besides, there is a further extension of the previous program flow which connecting to a switch to determine the usual data value and send alert message to the user immediately.



Program flow which pulls the status of actuators from InfluxDB and display it on Node-RED dashboard as switches. Besides, there is also an automation switch to let the user enable the automation system. All the action that done by the remote user will be sent to the Raspberry Pi via MQTT.

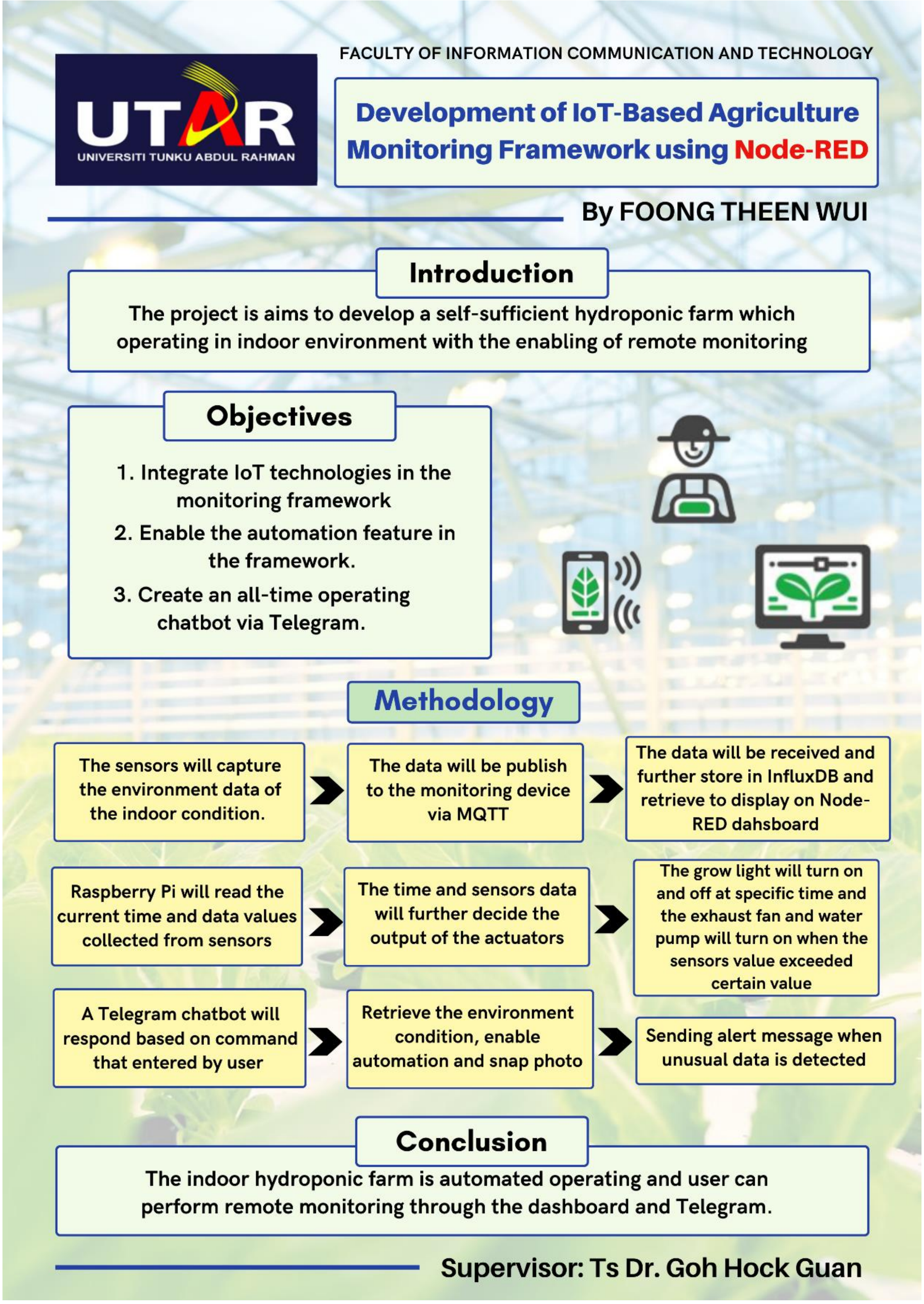


Program flow which creates a menu in Telegram chatbot to enable the user retrieve the sensors data by only using smart phone.



APPENDIX C

POSTER



PLAGIARISM CHECK RESULT

FYP2 Report

by THEEN WUI FOONG

Submission date: 25-Apr-2023 01:03PM (UTC+0800)
Submission ID: 2073747882
File name: FYP2_Report.pdf (4.97M)
Word count: 16786
Character count: 87669

PLAGIARISM CHECK RESULT

FYP2 Report

ORIGINALITY REPORT

7%	5%	4%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijsr.net Internet Source	1%
2	Submitted to Colorado Technical University Online Student Paper	1%
3	Submitted to University of College Cork Student Paper	1%
4	Submitted to HELP UNIVERSITY Student Paper	1%
5	repositorium.uminho.pt Internet Source	1%
6	www.semanticscholar.org Internet Source	1%

Exclude quotes	On	Exclude matches	< 8 words
Exclude bibliography	On		

Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

UTAR UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	FOONG THEEN WUI
ID Number(s)	20ACB04334
Programme / Course	Bachelor of Information Technology (Honours) Communications and Networking
Title of Final Year Project	Development of IoT-Based Agriculture Monitoring Framework using Node-RED

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceed the limits approved by UTAR)**
Overall similarity index: 7 **%** **Similarity by source** Internet Sources: 5 % Publications: 4 % Student Papers: 5 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required, and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note: Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Goh.

______________________________ Signature of Supervisor

Name: Goh Hock Guan

Date: 26/4/2023

______________________________ Signature of Co-Supervisor

Name: ______________________

Date: ______________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB04334
Student Name	FOONG THEEN WUI
Supervisor Name	TS DR. GOH HOCK GUAN

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

____
(Signature of Student)
Date: 24/04/2023