IoT SMART LABORATORY SYSTEM USING NODE-RED AND RASPBERRY PI

CARLOS ANAK ENSERIBAN

BACHELOR OF ENGINEERING TECHNOLOGY (ELECTRICAL) WITH HONORS

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	CARLOS ANAK ENSERIBAN
Date of Birth		
Title	:	STUDENT
Academic Session	:	2021/2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_______________________ (Student's Signature)

_______________________ (Supervisor's Signature)

New IC/Passport Number
Date: 25/2/2022

AMRAN BIN ABDUL HADI
Date: 25/2/2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

MAKLUMAT PANEL PEMERIKSA PEPERIKSAAN LISAN

(only for Faculty of Computer's student)

Thesis ini telah diperiksa dan diakui oleh
This thesis has been checked and verified by

Nama dan Alamat Pemeriksa Dalam :
Name and Address Internal Examiner

Nama dan Alamat Pemeriksa Luar :
Name and Address External Examiner

Nama dan Alamat Pemeriksa Luar :
Name and Address External Examiner

Disahkan oleh Penolong Pendaftar IPS :
Verified by Assistant Registrar IPS

Tandatangan : Tarikh :
Signature *Date*

Nama :
Name

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Bachelor of Engineering Technology (Electrical) with Honors.

____

(Supervisor's Signature)

Full Name : AMRAN BIN ABDUL HADI

Position : LECTURER

Date : 25/02/2022

____

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student's Signature)

Full Name : CARLOS ANAK ENSERIBAN

ID Number : TB18036

Date : 25/02/2022

IoT SMART LABORATORY SYSTEM USING NODE-RED AND RASPBERRY PI

CARLOS ANAK ENSERIBAN

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Engineering Technology (Electrical) With Honors

Faculty of Electrical & Electronics Engineering
UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2022

ACKNOWLEDGEMENTS

In performing our project, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this project give us much pleasure. We would like to show our gratitude DR. WAHEB A. JABBAR AL-AREEQI, the Course Coordinator of BTS3813 Senior Design Project, Universiti Malaysia Pahang, for giving us a good guideline for project throughout numerous consultations.

In addition, we would like to expand our deepest gratitude to our supervisor, ENCIK AMRAN BIN ABDUL HADI, who have been invested a lot of time and effort to have some meeting with us, supervised our progress and guided us a lot on knowledges, designs, and analysis throughout the project.

Many people, especially our course mates and team members itself, have made valuable comments and suggestions on this thesis which gave us an inspiration to improve our project. We appreciated all the people for their help directly and indirectly in completing our project.

ABSTRAK

Teknologi IOT (Internet of Things) sedang mengalami pertumbuhan yang luar biasa apabila ketersambungan berkembang ke seluruh dunia, dan ia terus memecah sempadan untuk menunjukkan perkara yang boleh dilakukan oleh teknologi canggih. Kebanyakan makmal kurang menggunakan teknologi IOT dan masih membayar kos yang tinggi untuk memastikan keselamatan. Dalam senario yang sangat biasa, peralatan makmal seperti lampu dan penghawa dingin dibiarkan menyala, walaupun tidak digunakan. Ia membawa kepada peningkatan dalam penggunaan kuasa dan pembaziran tenaga di makmal. Oleh itu, projek yang dicadangkan ini bertujuan untuk membangunkan sistem makmal pintar berasaskan IOT yang menjimatkan kos dan boleh dipercayai yang mengurangkan penggunaan kuasa dan memberikan kemudahan kepada pengguna untuk mengawal peralatan makmal walaupun pada jarak yang jauh. Protokol MQTT ialah komunikasi antara pelayan dan klien untuk menerbitkan dan melanggan data. ESP32 bertindak sebagai pelanggan dan pelayan bertindak sebagai broker MQTT. Sistem ini menggunakan Node MCU untuk menyambungkan beberapa penderia berbeza dan memindahkan data mereka ke pelayan tempatan Raspberry Pi menggunakan protokol Wi-Fi, yang kemudiannya akan dihantar ke pelayan awan Lautan Digital. Ia membolehkan pengguna terus berhubung dengan makmal dari semasa ke semasa dengan mengakses laman web antara muka mesra pengguna melalui telefon pintar atau komputer riba. Data masa nyata yang dikumpul akan dipaparkan pada papan pemuka yang dibangunkan dalam Node-RED. Sistem makmal pintar yang dicadangkan berhasrat untuk mengawal peralatan menggunakan Internet dengan cekap dengan kemudahan menyemak data dari tapak web pada bila-bila masa, di mana-mana sahaja.

ABSTRACT

The IOT (Internet of Things) technology is experiencing tremendous growth as the connectivity expands across the world, and it keeps on breaking the boundaries on showing what advanced technology can do. Most of the laboratories are lack of using IOT technology and still paying a high cost to ensure security. In the very common scenario, the appliances of the laboratory such as lights and air conditioners are kept left on, even when not in use. It leads to a rise in power consumption and wastage of energy in the laboratory. Therefore, this proposed project aims to develop a cost-effective and reliable IOT-based smart laboratory system that reduces power consumption and provides convenience for users to control laboratory appliances even at a distance. MQTT protocol is the communication between the server and client to publish and subscribe to the data. The ESP32s act as clients and the server acts as MQTT broker. This system utilizes Node MCU to connect several different sensors and transfer their data to the Raspberry Pi local server using Wi-Fi protocol, which will then send to the Digital Ocean cloud server. It enables users to stay connected with the laboratory from time to time by accessing the user-friendly interface website through smartphones or laptops. The real-time collected data will be displayed on the dashboard developed in Node-RED. The proposed smart laboratory system intends to efficiently control appliances using the Internet with the convenience of checking data from the website anytime, anywhere.

TABLE OF CONTENT

DECLARATION	
TITLE PAGE	
ACKNOWLEDGEMENTS	**ii**
ABSTRAK	**iii**
ABSTRACT	**iv**
TABLE OF CONTENT	**v**
LIST OF TABLES	**viii**
LIST OF FIGURES	**ix**
LIST OF SYMBOLS	**xi**
LIST OF ABBREVIATIONS	**xii**
LIST OF APPENDICES	**xiii**
CHAPTER 1 INTRODUCTION	**1**
1.1 Background	1
1.2 Problem Statement	3
1.3 Objectives	3
1.4 Scopes	3
CHAPTER 2 LITERATURE REVIEW	**4**
2.1 Introduction of Smart Laboratory System	4
2.2 Results and Findings	4
2.3 Existing Smart Devices using IoT	6
3 claps light control bulb	6
Xiaomi Mi Smart Home kit	8
2.4 Advantages Smart Laboratory System	10

2.5 Smart Laboratory System Component 11
Raspberry Pi 11
NodeMCU-ESP32 11
1-Way Relay Module 12

CHAPTER 3 METHODOLOGY 13
3.1 Project details 13
3.2 Block diagram of Smart IOT Laboratory System 14
3.3 Flowchart of Smart Laboratory System using Node-RED 15
3.4 Implementation using Raspberry Pi 20
3.5 Hardware and Software Description and Specifications 20
Raspberry Pi 20
3.6 Node-RED 21
3.7 Project / Hardware Cost 24

CHAPTER 4 RESULTS AND DISCUSSION 25
4.1 Introduction 25
4.2 Area 1 and Area 2 26
4.3 MYSQL setting in Node-RED 28
4.4 Security Camera 29
4.5 PIR Sensor 30
4.6 Twilio setting with PIR sensor in Node-RED 31
4.7 Light Bulb 32
4.8 MQTT Node-RED Integration 34

CHAPTER 5 CONCLUSION 36

5.1	Introduction	36
5.2	Conclusion	36
5.3	Recommendation	36
REFERENCES		**38**
APPENDICES		**40**

LIST OF TABLES

Table 3.1 Project Cost 24
Table 5.1: Project Gannt Chart 46

LIST OF FIGURES

Figure 2.1: 3 claps light control bulb 7
Figure 2.2: Reviews of clap light control bulb 7
Figure 2.3: Mi Smart Home Kit 8
Figure 2.4: Reviews of Xiaomi Mi Smart Home Kit 8
Figure 2.5: Wi-Fi CCTV Camera 9
Figure 2.6: Reviews of CCTV camera 9
Figure 2.7: Raspberry Pi 11
Figure 2.8: NodeMCU-ESP32 11
Figure 2.9: 1-Way Relay Module 12
Figure 3.1: Layout of Smart IOT Laboratory at FTKEE 14
Figure 3.2: System architecture of the Smart Laboratory System 15
Figure 3.3: Flowchart of the Smart IoT Laboratory using Node-RED 16
Figure 3.4: Flow of Control Lamp 17
Figure 3.5: Flow of PIR Sensor 17
Figure 3.6: Flow of Security Camera 18
Figure 3.7: Flow of Humidity and Temperature 19
Figure 3.8: Raspberry Pi 20
Figure 3.9: Node-RED 22
Figure 3.10: Node-RED 22
Figure 3.11: MQTT Node 23
Figure 3.12: Gauge Node 23
Figure 3.13: Dashboard 24
Figure 4.1: Area 1 26
Figure 4.2: Area 2 26
Figure 4.3: Flow Area 1 in Node-RED 27
Figure 4.4: Flow Area 2 in Node-RED 27
Figure 4.5: MYSQL function Node 28
Figure 4.6: Security Camera 29
Figure 4.7: Flow in Node-RED 29
Figure 4.8: PIR Sensor 30
Figure 4.9: Flow of PIR Sensor with twilio in Node-RED 30
Figure 4.10: Twilio Node 31
Figure 4.11: Twilio function node 32

Figure 4.12: LAMP 32
Figure 4.13: Flow of light bulb in Node-RED 33
Figure 4.14: MQTT Node-RED Integration 34
Figure 4.15: Security tab 35
Figure 5.1: Hardware of Lamp system 41
Figure 5.2: Website of Smart Laboratory 41
Figure 5.3: Prototype of PIR Sensor 42
Figure 5.4: Prototype of ESP Camera 42
Figure 5.5: Prototype of Temperature and Humidity Sensor 43
Figure 5.6: Account of Twilio 44
Figure 5.7: SMS from Twilio 44
Figure 5.8: Reading of Temperature and Humidity sensor in database 45

LIST OF SYMBOLS

LIST OF ABBREVIATIONS

IoT	Internet of Things
MQTT	Message Queuing Telemetry Transport
SMS	Short message services

LIST OF APPENDICES

Appendix A: Overall Actual Product	41
Appendix B: Twilio	44
Appendix C: Phpmyadmin	45
Appendix D: Gantt Chart Project Planning for SDP2	46
Appendix E: Coding of PIR Sensor	47

CHAPTER 1

INTRODUCTION

1.1 Background

The invention of Smart IoT Laboratory system is to upgrade the standard by applying the advanced technology available. It brings many advantages for people in this era that always asked for fast and easy. The existing devices had maximized the functionality to fulfil all the human needs but there are still some spaces for improvement. Smart IoT Laboratory System should be used as a multifunction device that can control several appliances in laboratory. At the same time, it ensures the laboratory safety and increase laboratory security.

The security of the laboratory should be strengthened with the advanced technology available, which is IoT technology. A camera and motion sensor are suggested to use in this system to ensure the security of users. Besides, wastage of energy has become the biggest problem as people tend to forget to switch off the appliances after leaving from one place. People nowadays are living a busy life, and it is very time-consuming to switch off all the appliances as the sockets are normally placed separately. The fact that most of the people do not turn off them when they leave causes wastage of energy. Real-time data monitoring and control using IoT are one of the ways to reduce energy wastage. To overcome this problem, a system with automation control and a monitoring system should be designed and developed so that people can control and monitor the appliances wirelessly.

A spike in electricity usage can be obviously seen from the monthly electricity bill. The monthly electricity bill comes from cumulative daily consumption. In the very common scenario, the appliances of the laboratory such as lights and air conditioners are kept left on, even when not in use. Imagine as time passes, it eventually leads to a rise in power consumption and wastage of energy in the laboratory. With so much help from the technology, it is only able to solve parts of the problem. If the root cause of the problem of high consumption of energy is not addressed, the problem will still exist, and it will get worse and worse over the time as people do not put much attention to its consequences and choose to ignore it. Therefore, the action of spreading public awareness on energy savings is important. People should acknowledge energy efficiency, which refers to using less energy to produce the same amount of services or useful output. People are encouraged to starts by changing their own behaviour such as practising always turn off the appliances whenever no in use.

This study presents an IoT-based smart laboratory system with the design, fabrication, and validation. It enables IoT automation and monitoring laboratory by using Node-MCU as a microcontroller and Wi-Fi gateway. Several sensors are used to control and monitor various environmental parameters such as temperature and humidity. An ESP32-camera and a PIR motion sensor are attached to the system to enhance laboratory security. A few relays are used to control the activities of laboratory appliances, such as lights. The MQTT protocol acts as the communication between the server and Node-MCU to publish and subscribe the data. Furthermore, a user-friendly interface website is developed to bring convenience in the aspect of the interaction between the users and the smart laboratory as the real-time collected data will be displayed on the dashboard developed in Node-RED. Laboratory users are allowed to view and control the appliance through the website. A simple prototype is constructed to implement and validate the effectiveness of monitoring the appliances of the proposed smart laboratory system. The developed system aims to develop a cost-effective and reliable IoT-based smart laboratory system to automate laboratory appliances, strengthen security, and upgrade life quality with technologies.

To summarize, this project is focus on developing a Smart IoT Laboratory System and monitoring system controlled by Raspberry Pi through the Internet of Things (IoT). By having this smart IoT laboratory system, it provides high efficiency as many home devices are connected to the system through internet instead of wired connection. It also provides more safety and security as it can notify the user through SMS or email during emergency. The smart laboratory system will be able to solve the problems in a high efficiency, low cost, and user-friendly way.

1.2 Problem Statement

1) High cost of installing many security systems

2) Safety and security are not secured as no notifying when the user is away from home

3) Busy user who did not switch off home appliances when not in use caused wastage of energy

1.3 Objectives

1) To display the real output into dashboard by using Node-RED.

2) To establish communication between local/cloud server and Node-MCU using MQTT protocol

3) To control and monitor the appliances using Node-RED dashboard

1.4 Scopes

The scope for this project is focus on the implementation of IOT in the smart laboratory system using Raspberry Pi that can control variety of laboratory appliances. The system is designed to control and monitor the laboratory appliances by using only one website through wireless connection instead of wired connection. Raspberry Pi boards can perform very well to provide complete web server functions with a low budget. LAMP web server (Linux, Apache, MySQL and PHP) will be installed in Raspberry Pi. Besides, MQTT protocol acts as the communication between the local server and Node-MCU to publish and subscribe the data.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction of Smart Laboratory System

Internet of things (IOT) is an effective system where users can publish and subscribe to the database, especially in the industrial sector. With an internet connection, humans will only need to monitor the system and control the functions such as temperature, humidity, and security system in a more convenient way.

2.2 Results and Findings

Some of articles were reviewed for the current literature review, with all the summaries and implications of this report. The purpose of the investigation was to identify trends in the literature that may lead to contradictions or gaps/weaknesses that need to be filled. Analysis of each piece of literature has been organised based on the themes being discussed and chronologically, on the publication date from oldest to newest.

(Chooruang & Mangkalakeeree, 2016; Tsao et al., 2022) presents the use of Message Queuing Telemetry Transport (MQTT) for messaging protocol in wireless heart rate monitoring systems. The investigation discovered that the normal HTTP protocol does not satisfy their criteria because it requires more bandwidth and the necessity to keep the server alive to respond to incoming requests from clients, which consumes a lot of power.

The next article presents a Design and Implementation of IOT Based Smart Laboratory (Amruta N. Banagar & Rajshankar Khattar, 2020). This research focuses on the constructed system that uses IOT and mobile communication technologies to monitor the laboratory's overall activity. The system will monitor lab operations such as energy consumption and device utilization using an IOT system. This IOT monitoring system is based on the dashboard developed in Node-RED, which is also linked to an MQTT broker.

In IOT systems, the integration of Node-Red with MQTT is becoming an excellent platform for easily connecting with physical devices. Paper (Domínguez et al., 2020) proposed a remote laboratory for automatic control that enables an easier interconnection and integration of its elements. The study focuses on exhibiting an industrial laboratory controlled by various factors such as an instructor who is a Node-Red client, one student who is a client for each student, and a central broker. The next findings present the IOT sensor working with the Node-RED platform as a way to transfer the data from the sensor on Raspberry Pi (Lekić & Gardašević, 2018). This system transfers data to the IBM Blue mix cloud and demonstrates how effective it is to integrate sensors using the Node-RED platform.

The next findings were proposed a Smart Lab with a wireless Sensor Network (Poongothai et al., 2018). The monitoring system was designed using a mobile application and Node-RED dashboard. The data was collected from each device of the smart lab and transferred by using the MQTT system. The objective of the writer is to show this IOT system can help the user to control the system universally and help to reduce human intervention monitoring the devices.

Another researcher proposed using ESP32 to monitor a door security system (Revelivan et al., 2019). This microcontroller performs a publish and subscribe from the cloud MQTT so that the communication between the application and the door lock is successful. The hardware of this system included an ESP32 that was linked to a PIR sensor to detect any movement, as well as a magnetic sensor that collected data to determine the state of the door security system. Because the ESP32 has a Wi-Fi connector for connecting to the internet, this system demonstrates that it may be used in a variety of applications.

Next article presents that Internet of things IoT and cloud are widely used in integration of advanced protocols (Balaji et al., 2020; Yang et al., 2020). The study concludes that using MQTT as an IoT protocol is safe for both the sensor and the cloud. According to my observations, it is critical to keep the data from the sensor or database public because anyone can hack the data and publish it as their own on the website. Cloud is an excellent choice IoT platform, especially for individuals or researchers, because it is very secure because each user creates their own password and username.

Article (Tabaa et al., 2018) presents the use of Node-RED in the Industry 4.0. It is very productive because technically it can help to reducing costs and improve the product and services. The study finds this system was developed the industrial architecture based on Node-RED and Modbus. The implementation of this system is connecting wired with all sensors and actuators based on industrial protocol.

From the literature survey, it was observed that the IOT with Node-RED and MQTT gave many positive impacts, especially in monitoring systems because it generally can save time and money for the users. This technology enables the rapid development of applications, particularly those that respond to events such as IOT applications.

2.3 Existing Smart Devices using IoT

3 claps light control bulb

Figure 2.1: 3 claps light control bulb
Source: (Cards et al., n.d.)

Figure 2.1 depicts one of the existing smart home device devices that uses wireless sound on/off and clap detection. This light control bulb is activated by any sound detection, so if you clap your hands or knock on the table three times, the light will switch on/off automatically. This system has been improved so that no phone or remote controller is required. However, because the bulbs are not synced, this light control just requires one bulb per room.

Figure 2.2: Reviews of clap light control bulb
Source: (Cards et al., n.d.)

Figure 2.2 above depicts the overall customer feedback, in which the majority of customers report the same issue, namely that the three claps do not work consistently. This signifies that the product did not meet the expectations of the customers.

Xiaomi Mi Smart Home kit

Figure 2.3: Mi Smart Home Kit
Source: (Security et al., 2022)

Figure 2.3 depicts the Mi smart home kit, which includes a Human Motion Sensor, Door/Window Sensor, and Wireless Switch. According to the product description, it is simple to install, plug and play, does not require any tools and does not damage the decoration, does not take up much room, and can be controlled remotely through smartphone. When the user is not at home, they can still know everything about their residence. It also consumes less power and does not require battery replacement for two years. Shopee Mall has starting pricing as low as RM 32.80.

Figure 2.4: Reviews of Xiaomi Mi Smart Home Kit
Source: (Security et al., 2022)

Figure 2.4 above depicts customer product ratings and reviews. Many of them believe that the majority of the language utilised in the programme is Chinese. This is inconvenient for folks who do not understand Chinese. However, there is a Global version of this Mi Smart Home Kit available for a higher price of RM277.60 from Lazada. However, this product has been discontinued.

Figure 2.5: Wi-Fi CCTV Camera
Source: (Specifications, n.d.)

The wireless CCTV camera connected through Wi-Fi is depicted in Figure 2.5 above. According to the product description, it is compatible with both Android and iOS smartphone devices. The image from the CCTV can also be viewed by multiple users; it is not limited to a single user. Furthermore, it is more advanced when we can manually control the rotation of the camera and monitor it in real time. From the CCTV, we may have a 360° view of our home/places.

Figure 2.6: Reviews of CCTV camera
Source: (Specifications, n.d.)

Figure 2.6 above depicts a customer review in which the majority of them believe that this CCTV camera is difficult to install since it must be installed close to the CCTV itself; otherwise, the video quality of the CCTV will suffer. Overall, the majority of customers are pleased with the product's quality.

2.4 Advantages Smart Laboratory System

A Smart Laboratory systems is a laboratory outfitted with the Internet of Things (IoT) Technology. This system enables the communication between user and control of laboratory appliances. Advantages are included with reduce wastage of energy, time saving, convenience and enhanced security while disadvantages include internet reliance and cost.

First, the smart laboratory system act as a multi-function devices which can control any appliance in laboratory and it results in energy savings. People or laboratory user who always forget to turn off laboratory appliances will need this as user can turn off the appliances even if they are leaving from laboratory. At the same time, we can save the time without checking whether the laboratory appliances have switched off one by one. A simple things is we just go through into the website, and the user can turn off directly all the appliances in laboratory. Convenience is the first point of the bulk of such user no need to download any applications from google store/Apple store because some of user want to reduce the memory storage of the phone. So what we do is we create a website that can be access for user only, and from that they can control and monitoring the laboratory system from the website. As long as the user have a username and password, so they can access the laboratory in any time they want.

There are some disadvantages of smart laboratory system. Internet reliance is the biggest weakness of the smart laboratory. The smart laboratory system does not work whenever the internet connection goes down. It brings trouble to user if the smart laboratory system is the only way to control the laboratory system. For example, in this system we have created a twilio sms where user automatically received the sms if in laboratory detect the motion of human. So one motion detected, the credit account will be deducted around 2 sen. Then if the balance of credit account is zero, the user will not received any sms from the devices. They just only can control manually from the website. The smart laboratory system would possibly need to use various program and apps to manage the technology as not all systems would like to incorporate in these smart laboratory system. It cost much higher to get a centralized platform that allows controlling all the systems in one.

2.5 Smart Laboratory System Component

This chapter will describe the component that we use where is included electrical part in Internet of Things system.

Raspberry Pi

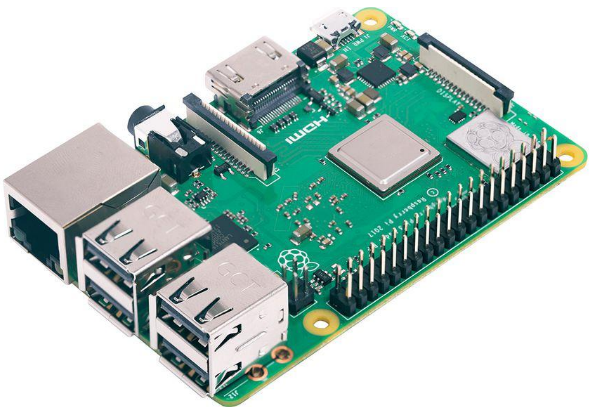

Figure 2.7: Raspberry Pi

The Raspberry Pi is a low-cost, credit-card-sized computer that plugs into a computer display or television and operates with a regular keyboard and mouse. We utilised this raspberry pi as our local server to connect with MQTT for this system.

NodeMCU-ESP32

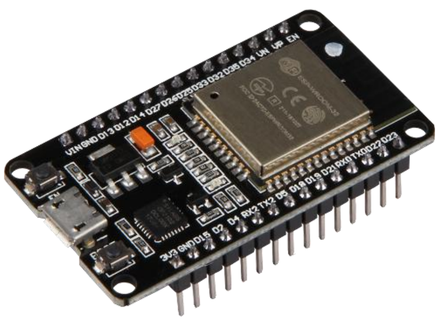

Figure 2.8: NodeMCU-ESP32

The ESP32 is a basic controller that can be programmed using Luascript, the Arduino IDE, and a breadboard. This board features 2.4 GHz dual-mode Wi-Fi and

Bluetooth wireless connectivity. In this project, we simply connect the ESP32 to a PIR sensor, a DHT11, and a lamp system.

1-Way Relay Module

Figure 2.9: 1-Way Relay Module

Relays are electric switches that use electromagnetism to transform tiny electrical inputs into low current. As a result, for this project, this relay was mostly employed in the Lamp system to regulate the on/off of the lights. Electromagnets establish or break existing circuits in response to electrical inputs.

CHAPTER 3

METHODOLOGY

3.1 Project details

The methodology is a system of broad principle of rules from which specific methods or procedures may be derived to interpret or solve. This chapter explained in detail the procedure of the implementation of the control system in the Smart Laboratory System for security, control and monitoring the laboratory system. The methods used in this chapter are aimed to achieve the objectives of the project which will give satisfying results on the performances of the control system in the Smart Laboratory System.

The objective of designing a Smart IOT Laboratory System is by monitoring and control the lab devices using the Node-RED dashboard and also from our own websites. In this system, Laboratory is interfaced with data that is controlled from the IOT system with a Local IP address. All the collected data from each sensor or device of the smart IOT laboratory was transmitted to a website for further analysis. All of the systems in the laboratory communicate using the MQTT (Message Queuing Telemetry Transport) protocol.

Figure 3.1 shows the Smart IOT Laboratory System layout at our Electrical Engineering Faculty where we chose the CAD Lab for project testing.

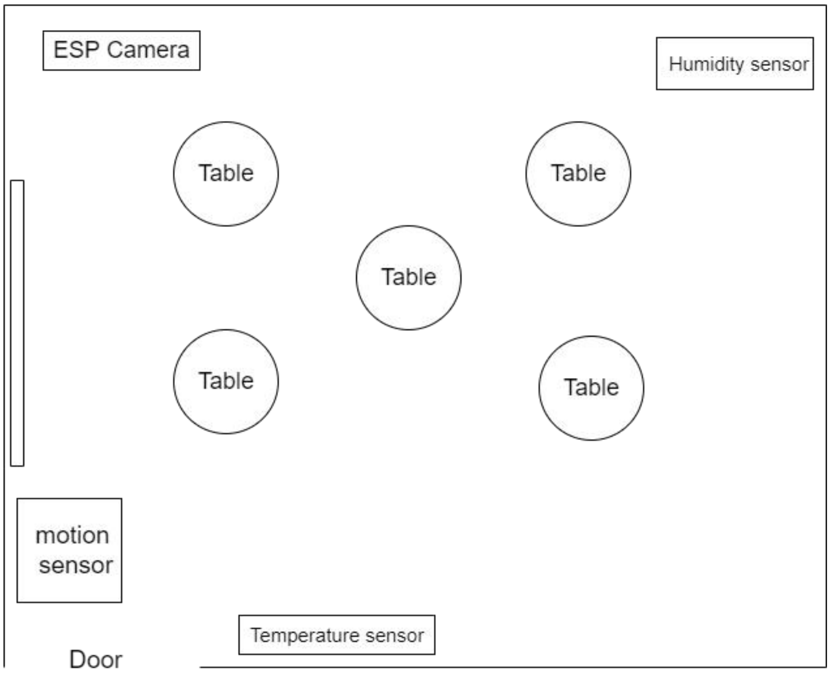

Figure 3.1: Layout of Smart IOT Laboratory at FTKEE

3.2 Block diagram of Smart IOT Laboratory System

Figure 3.2 shows the system architecture of the Smart IoT Laboratory system. The data from the sensor (input) will be sent through Node-Red that was connected with MQTT before the data from the sensor is transmitted to real-time on an interface and stored in a PHPMyadmin database. From Node-RED itself we created a simple Dashboard to manipulate and read the data through IoT systems.

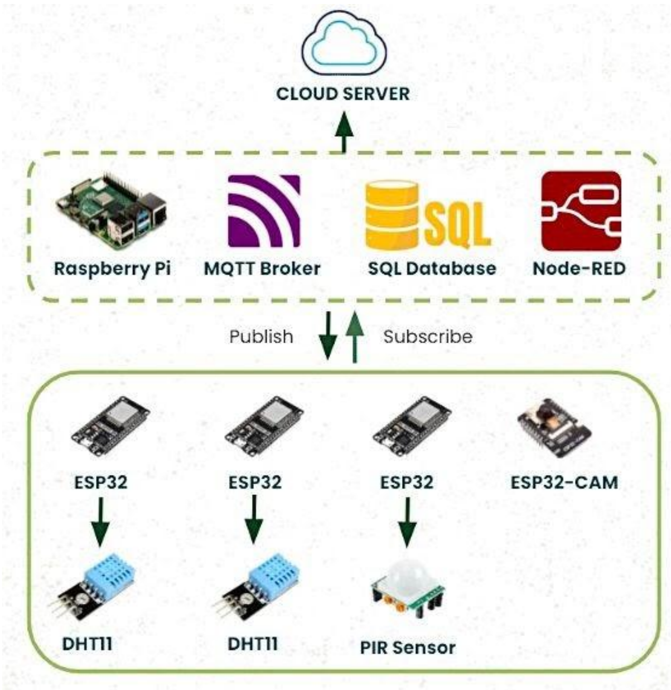

Figure 3.2: System architecture of the Smart Laboratory System

3.3 Flowchart of Smart Laboratory System using Node-RED

Figure 3.3 shows the flowchart of the Smart IoT Laboratory system using Node-RED, which shows the entire Smart Laboratory system in operation. The system flow starts with the raspberry pi linked to Wi-Fi with the local IP address, and then the Node-RED subscribes to the topic from the input, which is a DHT11 sensor, a PIR sensor, and an ESP32 camera. When the input publishes the topic into Node-RED, the MQTT protocol is connected, and the data is stored in the PHPMyAdmin database. All of the systems can also be monitored and controlled via a website established with Digital Ocean.

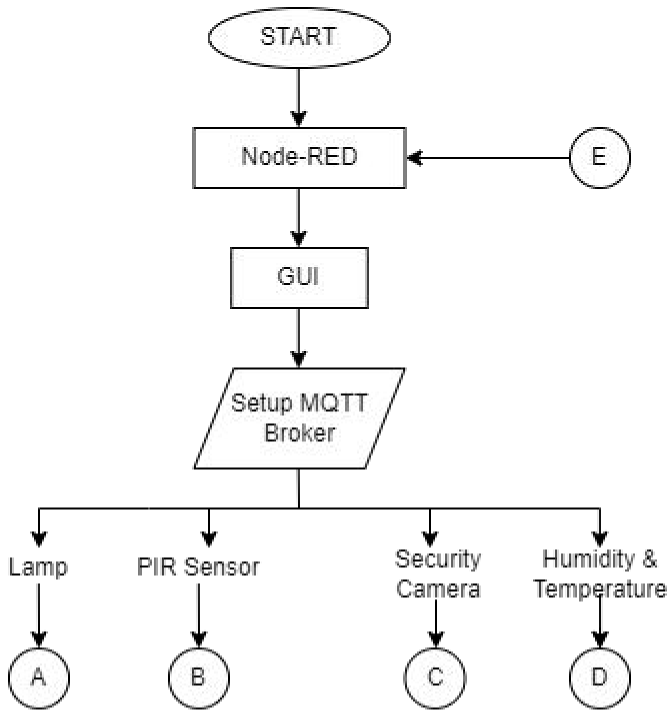

Figure 3.3: Flowchart of the Smart IoT Laboratory using Node-RED

In Figure 3.3 above show the flowchart of the system using Node-RED. Firs step is to connect the MQTT broker where each devices and sensor has their own topic to subscribe before connect to A,B,C and D.

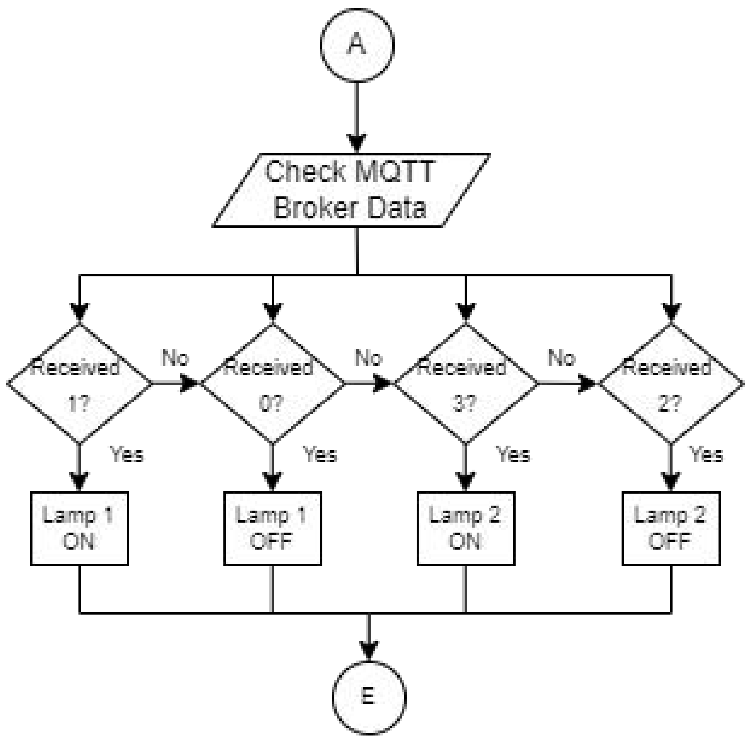

Figure 3.4: Flow of Control Lamp

In Figure 3.4 above show the flowchart of after going through subscribe process. Then, The MQTT system will check either the devices subscribes the topic from sensor and if yes, the control lamp can been used as usual.

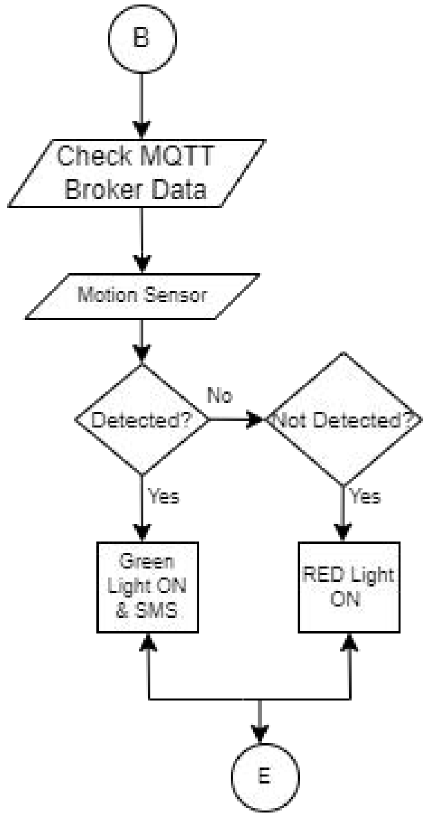

Figure 3.5: Flow of PIR Sensor

In Figure 3.5 above show the flowchart of PIR sensor after subscribe the topic from Node-RED. First, the motion sensor will detect either any movement in laboratory, if yes then the green light turn on it automatically sent sms through twilio. If no, the indicator will turn off.

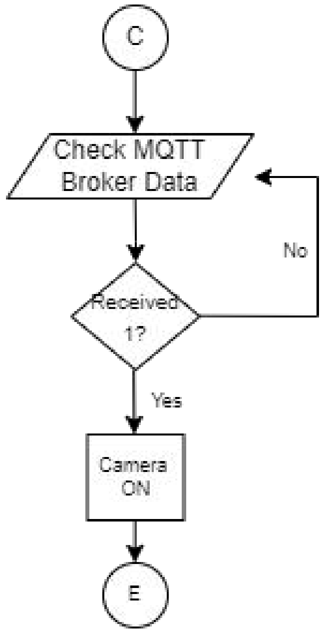

Figure 3.6: Flow of Security Camera

In Figure 3.6 above show the flowchart of security camera. When user or admin subscribe the topic they want, they will publish the topic through MQTT and the security camera can be monitor automatically through Node-RED dashboard.

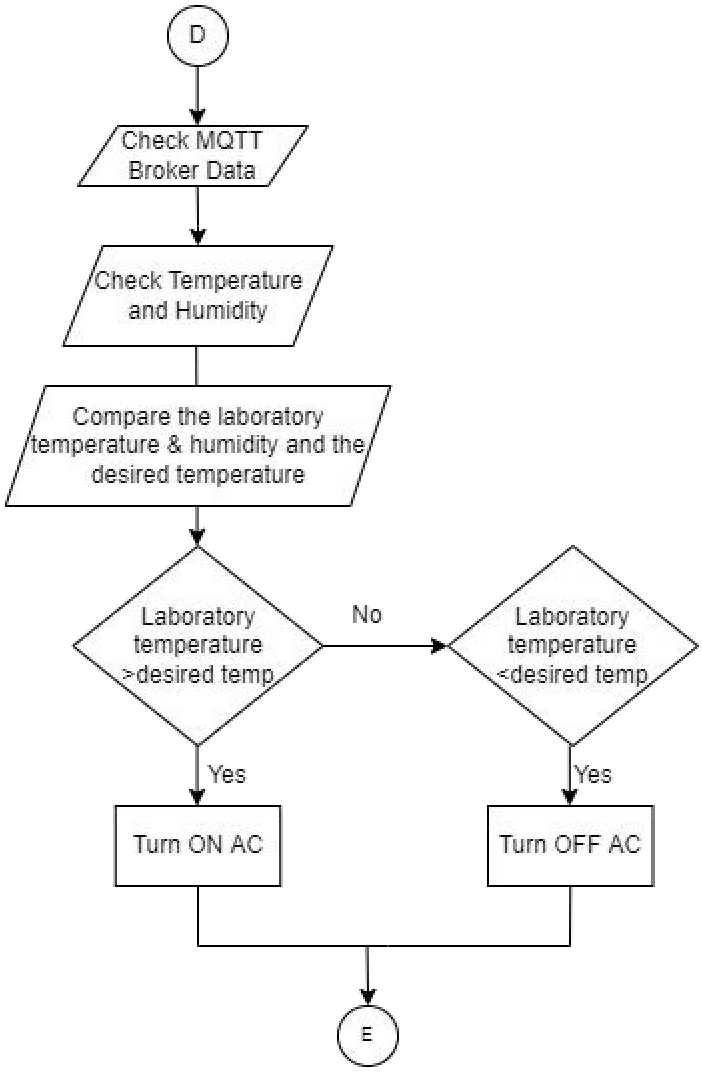

Figure 3.7: Flow of Humidity and Temperature

In Figure 3.7 above show the flowchart of DHT11 sensor with MQTT. If the topic has been subscribe from Node-RED, the sensor will publish the real data into Node-RED. Then, it will compare the laboratory temperature and humidity, and if the laboratory achieved the desired temperature the aircond will turn on. And if the laboratory did not achieved the desired temperature, the aircond will turn off.

3.4 Implementation using Raspberry Pi

The Smart IOT Laboratory system will sense the DHT11 temperature and humidity sensor in different areas which are area 1 and area 2. All the data from the sensors will be sent to the microcontroller Raspberry Pi/Localhost that was linked with MQTT and Node-RED, then it will be sent to the cloud platform for constant monitoring and analyzing purposes. Besides that, the data from the sensor also will be sent into our database which is PHPMyadmin and it is required only the admin can access the database. In this smart laboratory, an ESP32 with a PIR sensor was built. The principle of the system is when any movement is detected, the LED in the dashboard will display green light while if no movement is detected, the LED will display a red light. It is easier for the user because all the systems can be monitored either outside or inside of the laboratory as long as the user connects to the same IP address.

Raspberry Pi is connected to a network and runs this application in it. The control signals can be sent from any device connected in the network to the broker and the client publish the status of the device by the individual appliance to the broker and can be viewed in the dashboard.

3.5 Hardware and Software Description and Specifications

Raspberry Pi

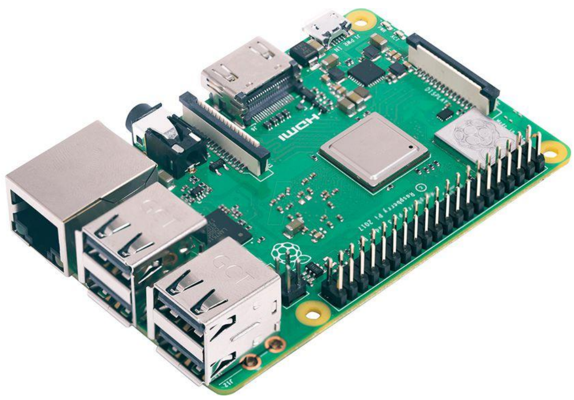

Figure 3.8: Raspberry Pi
Source: (Pi et al., n.d.)

Figure 3.8 depicts the microcontroller, which serves as a local server in this smart home system. This Raspberry Pi has been connected to a computer through an HDMI connection. Unfortunately, it does not work with some devices. The VNC viewer can be used to monitor this microcontroller, which is the solution to this problem. It is a small computer that is very competent of performing tasks; it is not only used for software but can also perform other tasks like a computer. For example, we can use Google and play a game as usual.

Specifications:

- Microcontroller: Raspberry Pi model 3
- Storage: microSD
- GPIO: 40-pin header
- Full size HDMI
- 4 USB 2 Ports
- 4 pole stereo output and composite video port
- Bluetooth 4.1 classic, Bluetooth low energy

3.6 Node-RED

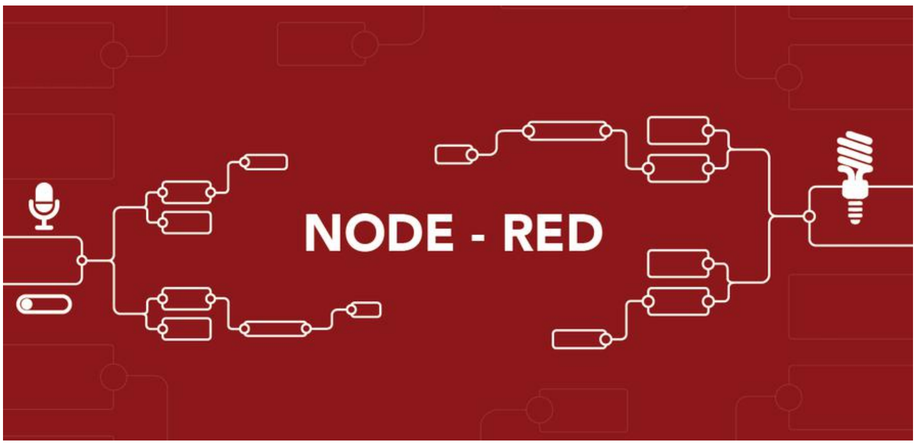

Figure 3.9: Node-RED
Source: (WolkAbout, 2019)

Figure 3.9 above shows the Node-RED where our smart laboratory system was successfully build with this software. Inside this Node-RED we can arrange the flow of the smart laboratory. For example in Figure 3.10 we can see flow of Temperature and Humidity in Area 1. Where we assign the publish and subscribe topic from MQTT node, each sensor has their own topic like esp32/dht/temperature and esp32/dht/humidity. The ESP32 is publishing the temperature readings every 10 seconds on the esp32/dht/temperature and esp32/dht/humidity topics.

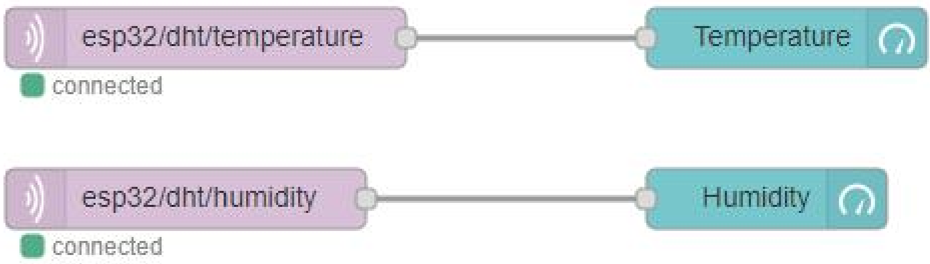

Figure 3.10: Node-RED

In Figure 3.11 below shows the MQTT Node interface where we can set up the server with our local IP Address. The insert the topic that we want to be subscribed to for example esp32/dht/temperature.

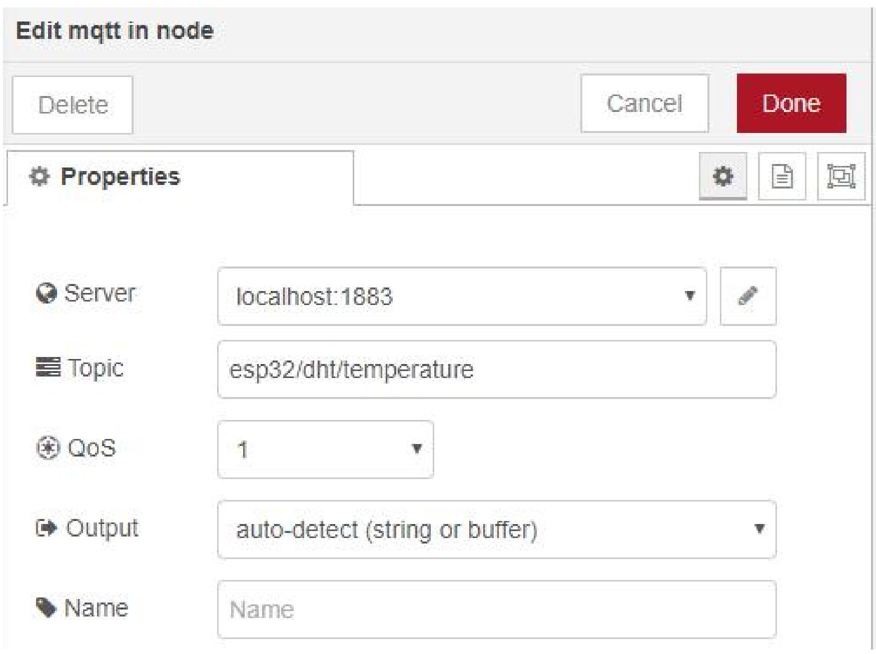

Figure 3.11: MQTT Node

In Figure 3.12 below show the gauge nodes and from there we can edit and insert properties for each reading. And the following nodes is set for the temperature readings.

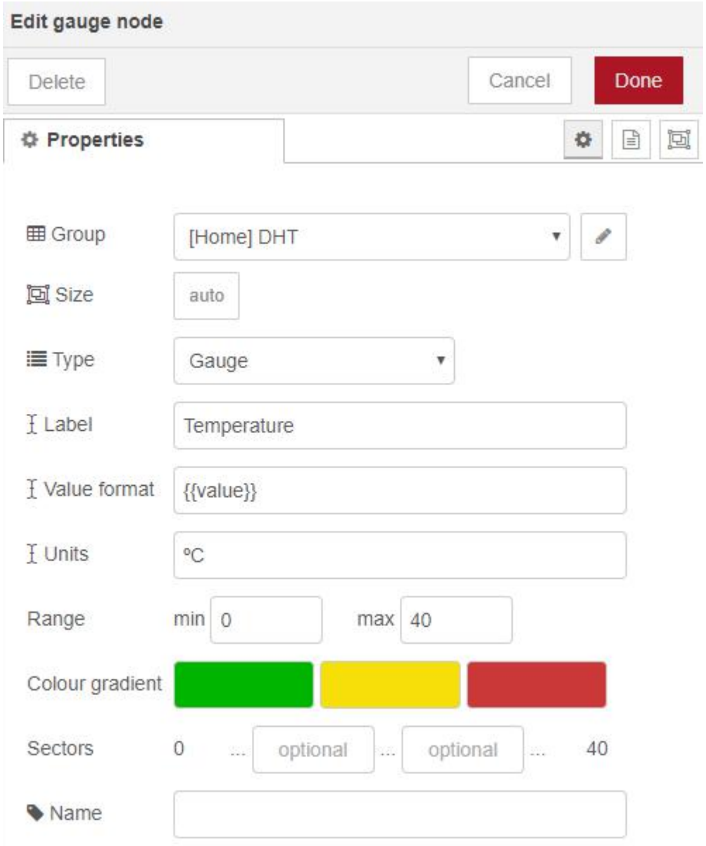

Figure 3.12: Gauge Node

After all has been completely setup, then next step is build a dashboard in Node-RED. To open the dashboard, we can access from localhost:1880/ui and supposedly we can read the current temperature and humidity readings in dashboard. The dashboard can be see in the Figure 3.13 below.

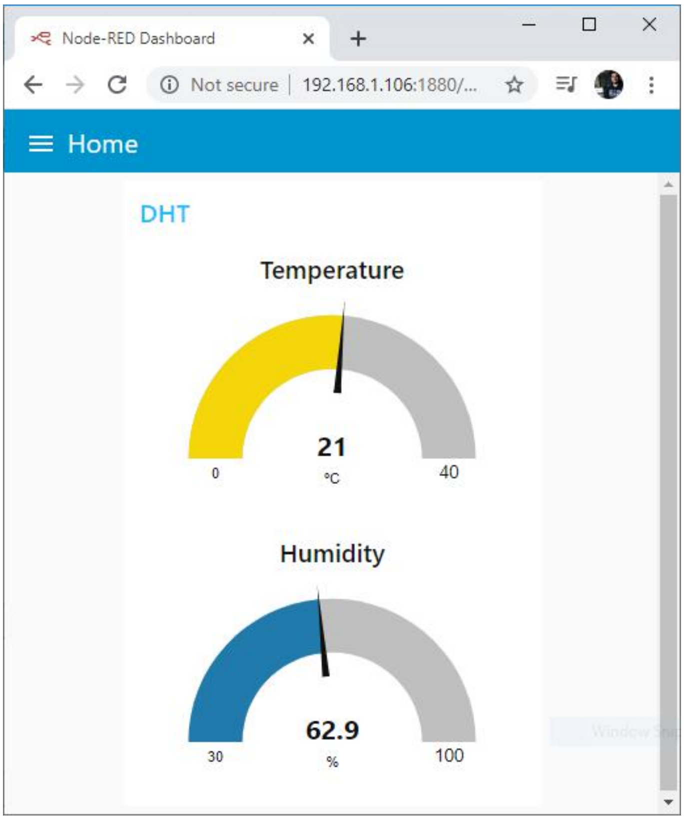

Figure 3.13: Dashboard

3.7 Project / Hardware Cost

Table 3.1 Project Cost

Items	Unit	Price/Unit (RM)	Total Price (RM)
Raspberry Pi	1	174.00	174.00
PIR Sensor	1	16.60	16.60
ESP32	5	26.10	130.50
ESP32 Camera	1	39.80	39.80
Relay	2	3.90	7.80
Overall Price (RM)			368.70

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

After completing all the method from the previous method chapter, this smart laboratory will be analyse with Node-RED. In this chapter shows all the results obtained from this study. Interface and figures are included in this chapter. The detailed explanation of interface and figures are also provided. The program and the interface will be program using Node-RED through Linux in Raspberry Pi.

The optimization method usage and interpretation of its results are obtained based on the detailed study of the usage of the software involve. First, make the flow of the smart laboratory function. Then, make the MQTT protocol by using publish and subscribe node in Node-RED. This flow is going to use as the input and output for the ESP32, PIR sensor, camera, Raspberry Pi, and light bulb. This Node-RED is user friendly and reliable. After successfully make the flow of the smart laboratory system, it will connect to the dashboard whether the flow is function or not. Analysis and discussion were focused more on the Node-RED. The results is extracted from the real data in the dashboard.

4.2 Area 1 and Area 2

Figure 4.1: Area 1

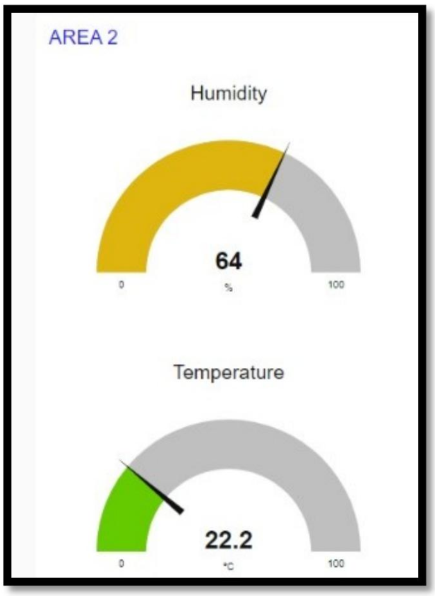

Figure 4.2: Area 2

Figure 4.1 and Figure 4.2 above shows the result of ESP32 with temperature and humidity sensor after all the flow in Node-RED successfully build. Each area have their own topic where ESP32 will subscribe the topic from Node-RED. Then, the MQTT protocol will develop to communicate from ESP32 and Node-RED. Beside that, the

MYSQL node also build to store and save the data into our database for personal analyse.

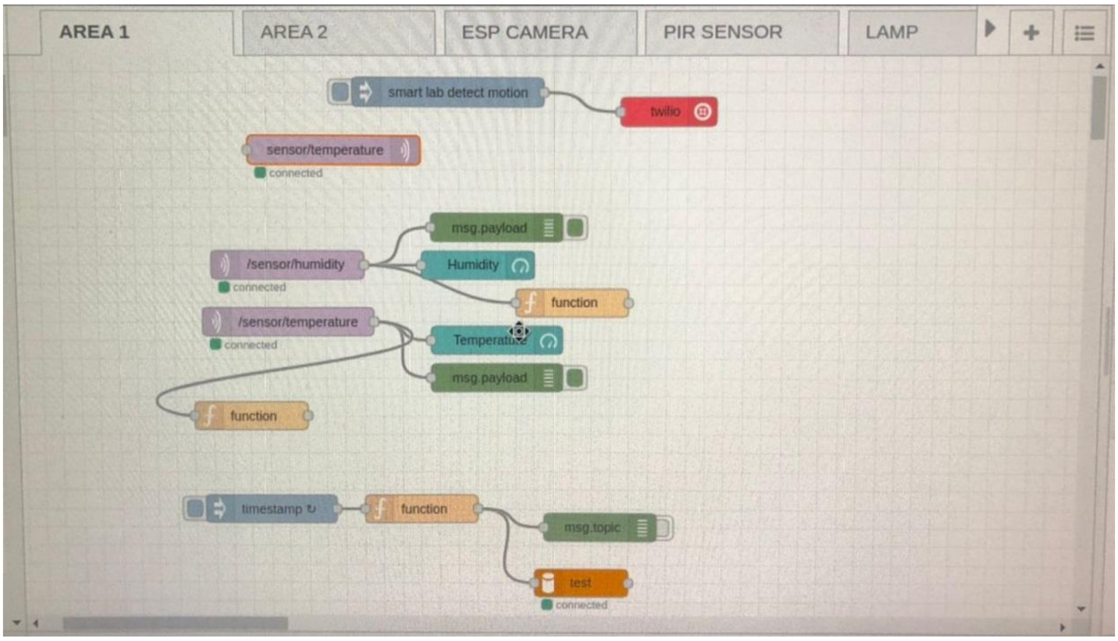

Figure 4.3: Flow Area 1 in Node-RED

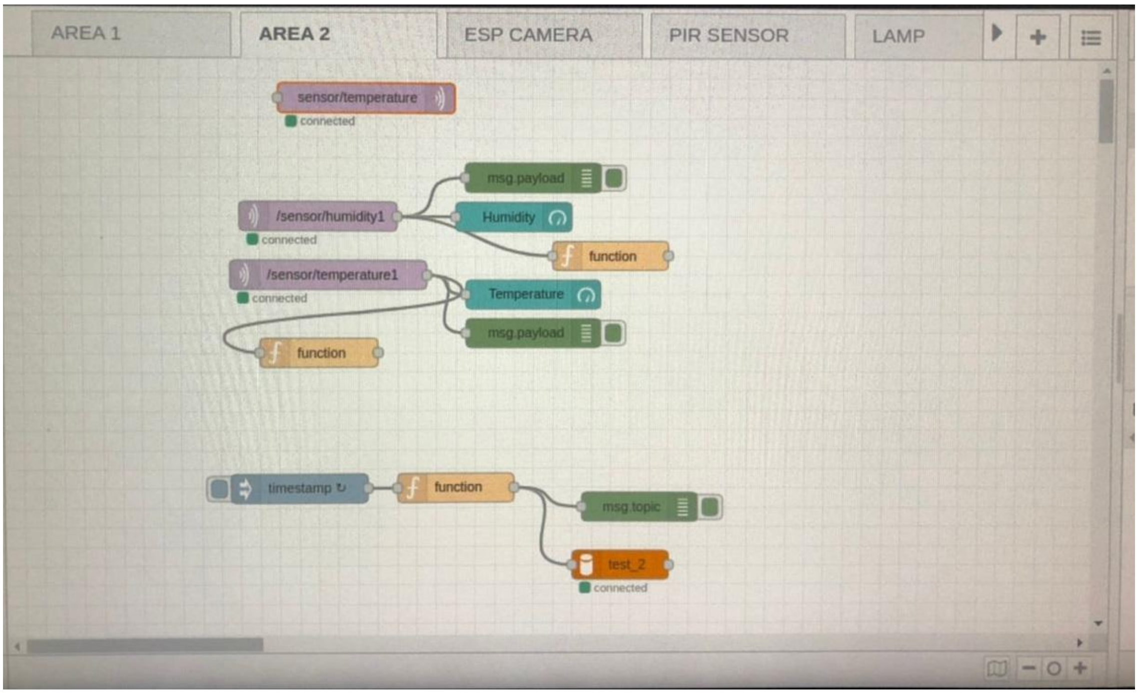

Figure 4.4: Flow Area 2 in Node-RED

As we can see in Figure 4.3 and Figure 4.4 above, the function call has been connected with MQTT node. Where this function is to set the global variable as shown in Figure 4.5. The timestamp node is required in this flow to publish the reading of temperature and humidity in 5 seconds. Which means every 5 seconds, the reading will appear and store in phpmyadmin database as shown in Figure 5.8.

4.3 MYSQL setting in Node-RED

For temperature and humidity sensor, the MYSQL was added to store the reading from sensor into phpmyadmin database. The flow from Node-RED can be seen in Figure 4.3 and Figure 4.4.

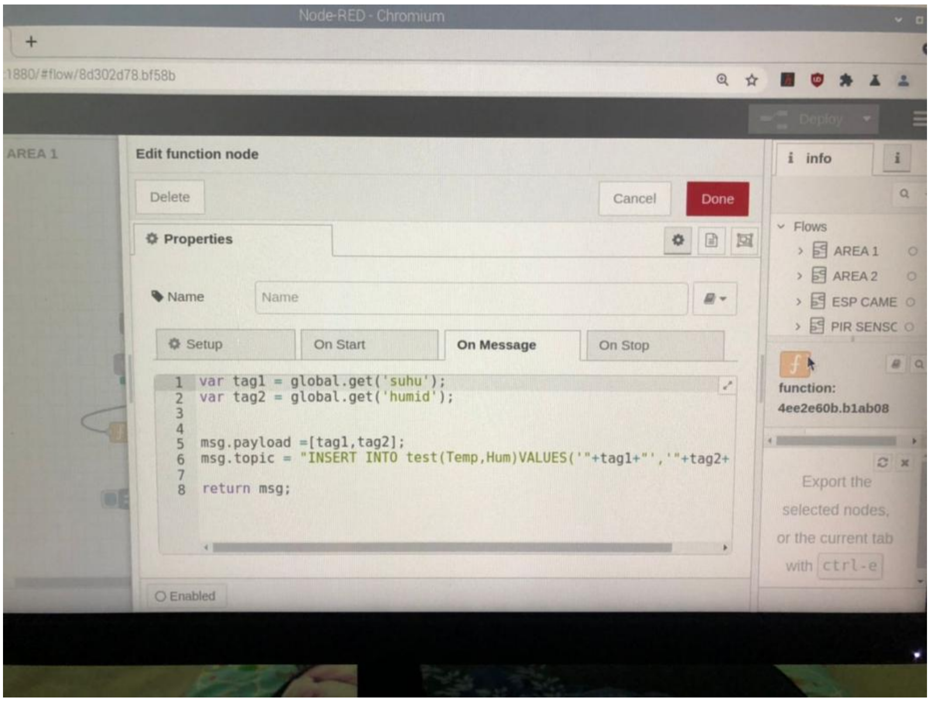

Figure 4.5: MYSQL function Node

As seen in Figure 4.5, the 'suhu' and 'humid' variables were set to global variables. As a result, if any instruction makes use of a global variable, the system will display data from nodes with the same global name. The message payload will then act as tags 1 and 2 to display temperature and humidity data. The data will then be passed into the database in phpmyadmin from the message topic itself, with INSERT INTO "test" as the database name in phpmyadmin. As a result, the data will be automatically sent to phpmyadmin every 5 seconds, as determined by the interval time in Node-RED.

4.4 Security Camera

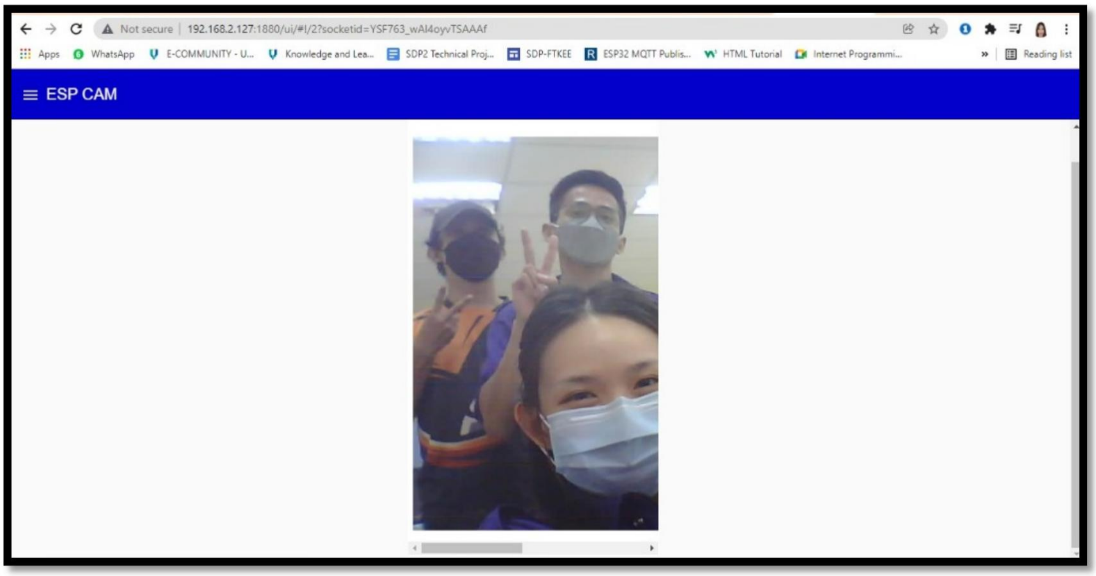

Figure 4.6: Security Camera

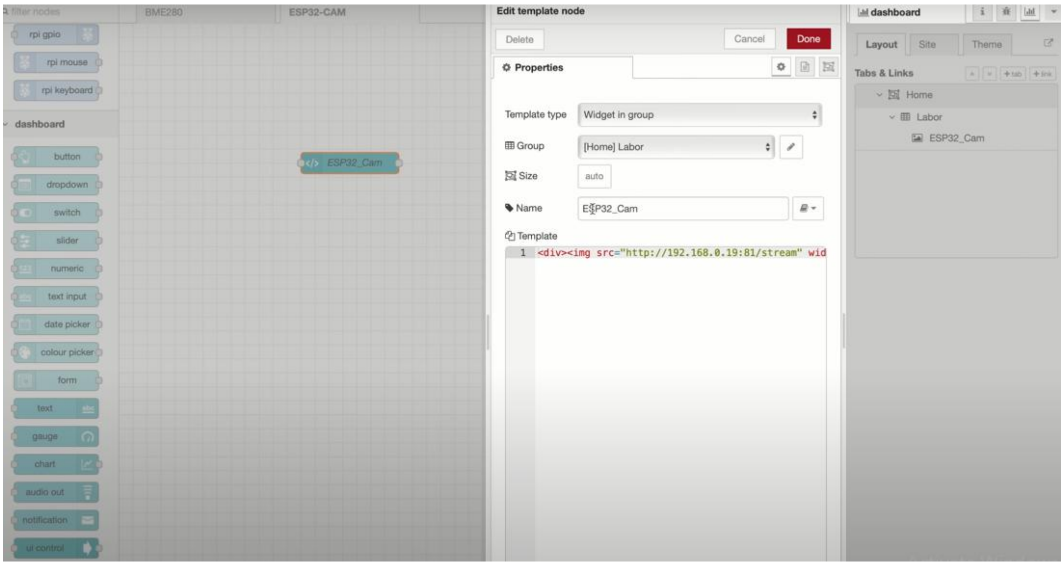

Figure 4.7: Flow in Node-RED

Figure 4.6 above shows the result of security camera in dashboard. Figure 4.7 is the setting or the program of the ESP camera where we can see IP Address in the template node, the IP Address only can be get through the output after run the coding in Arduino IDE. Then it follows by the width and height of the display of the camera and lastly is deploy the system and the results is can be view as in the Figure 4.7 above.

4.5 PIR Sensor

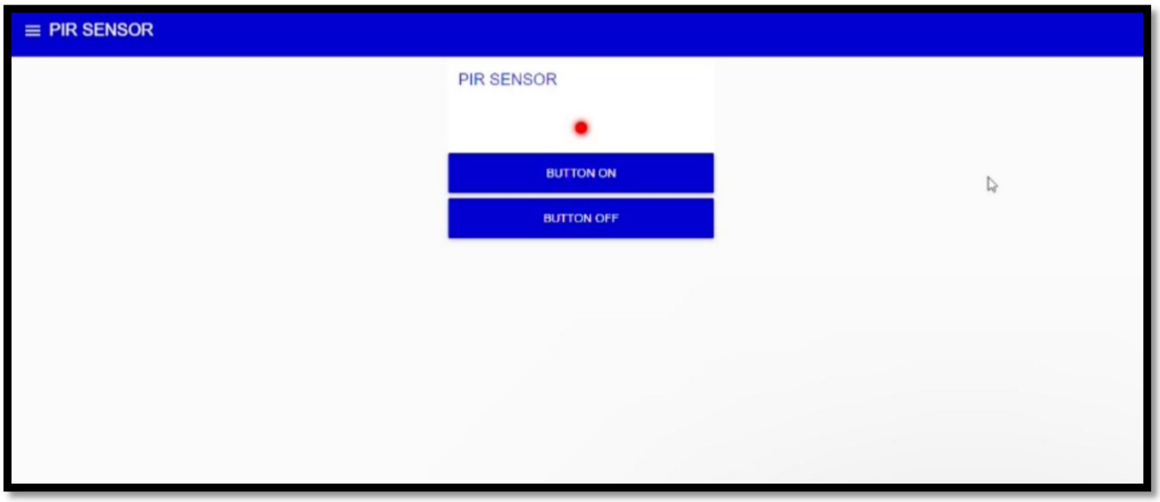

Figure 4.8: PIR Sensor

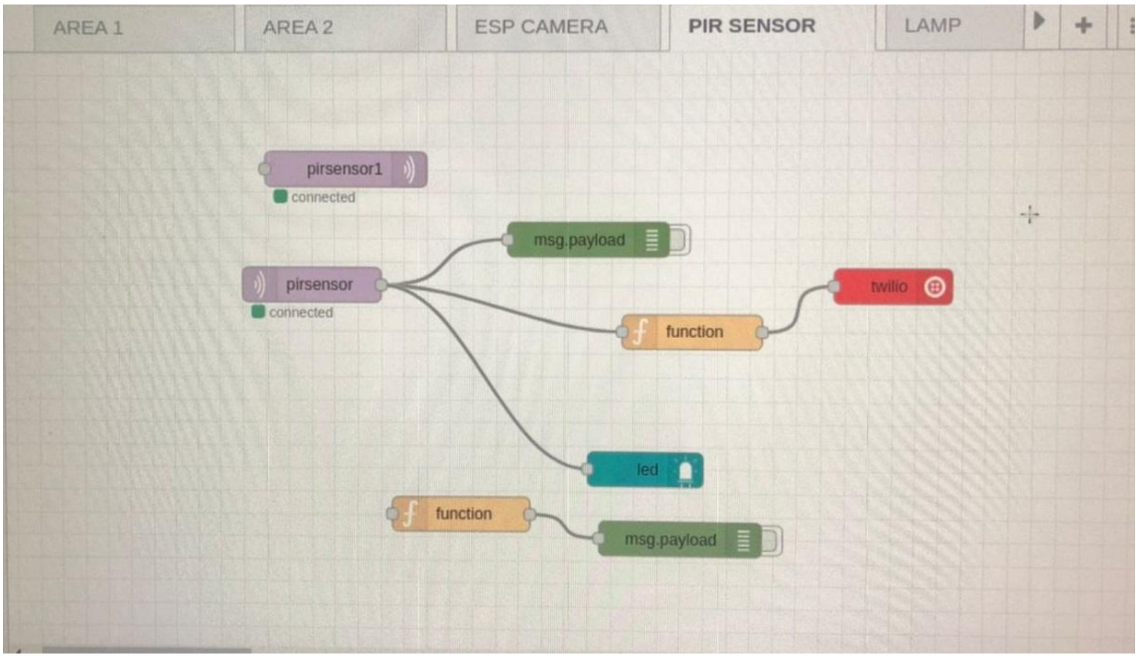

Figure 4.9: Flow of PIR Sensor with twilio in Node-RED

Figure 4.8 depicts the dashboard display of a PIR sensor or motion sensor. The indication light will turn green if motion is sensed in the laboratory. This PIR sensor was successfully constructed with the Node-RED function. As seen in Figure 4.9, once all nodes have been connected to MQTT, the PIR sensor will continue to read any motion in the laboratory. Twilio for sent messages is one of the new features added to this node. It works as follows: when the PIR sensor detects movement in the laboratory, the system immediately sends a message to the laboratory's owner or user.

4.6 Twilio setting with PIR sensor in Node-RED

In advanced, the system was upgraded with twilio system where the system automatically sent message to the owner of the system.

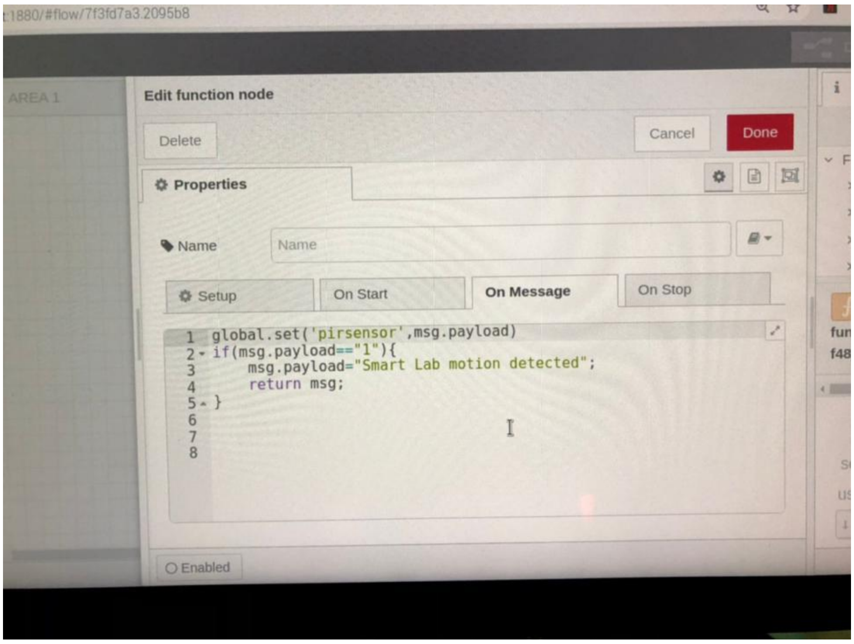

Figure 4.10: Twilio Node

The twilio has been configured with a PIR sensor, as seen in Figure 4.10 above. This means that whenever motion is detected, the twilio or SMS will immediately turn on and transmit messages to the owner. The PIR sensor in Node-RED was set to global, which implies that global variables can be accessed from anywhere within Node-RED, including other flows. After the global variable is set, any variable with the same name in other flows will work asynchronously. We select "1" as the message payload because the system will send the message if the PIR sensor returns a false result, as shown in Figure 4.11.

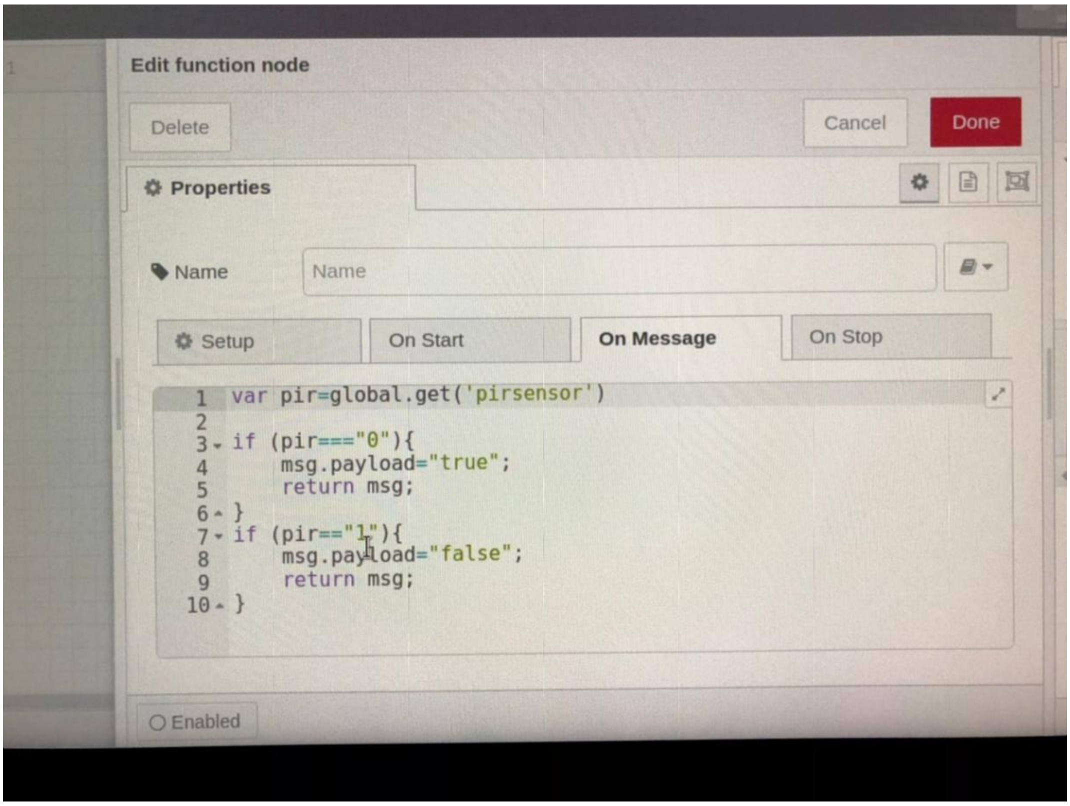

Figure 4.11: Twilio function node

4.7 Light Bulb

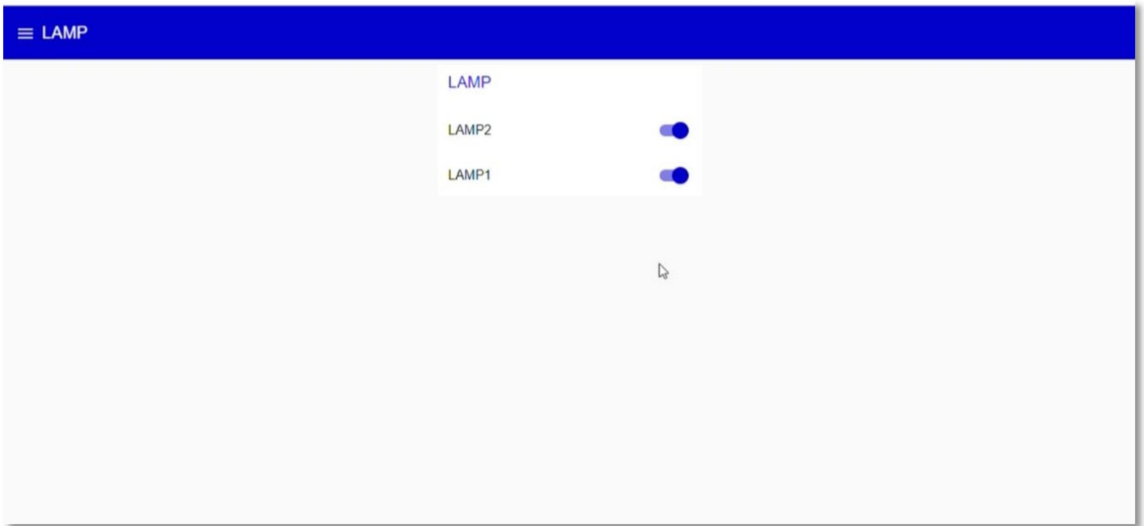

Figure 4.12: LAMP

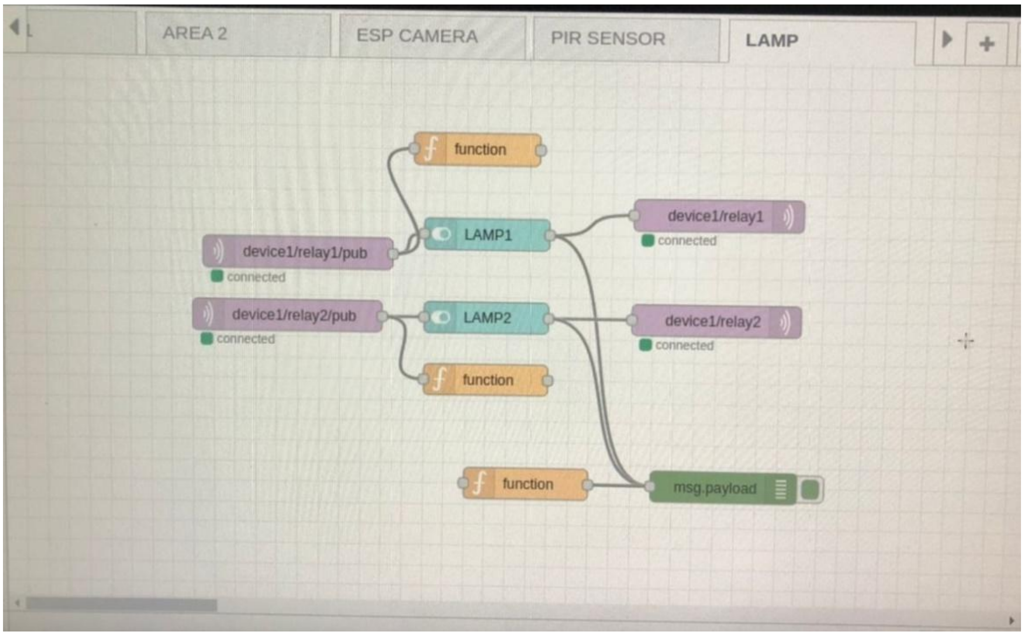

Figure 4.13: Flow of light bulb in Node-RED

The light system successfully install as can been seen in Figure 4.12 and flow of light bulb in Node-RED as in Figure 4.13. The publish and subscribe was needed as the Node-RED will publish a topic into ESP and it will subscribe then finally light system can be control by using dashboard as in Figure 4.12.

4.8 MQTT Node-RED Integration

Each flow has their integration process, set the MQTT broker host with any MQTT Node-RED. There is a few instructions to set up the MQTT broker with Node-RED nodes.

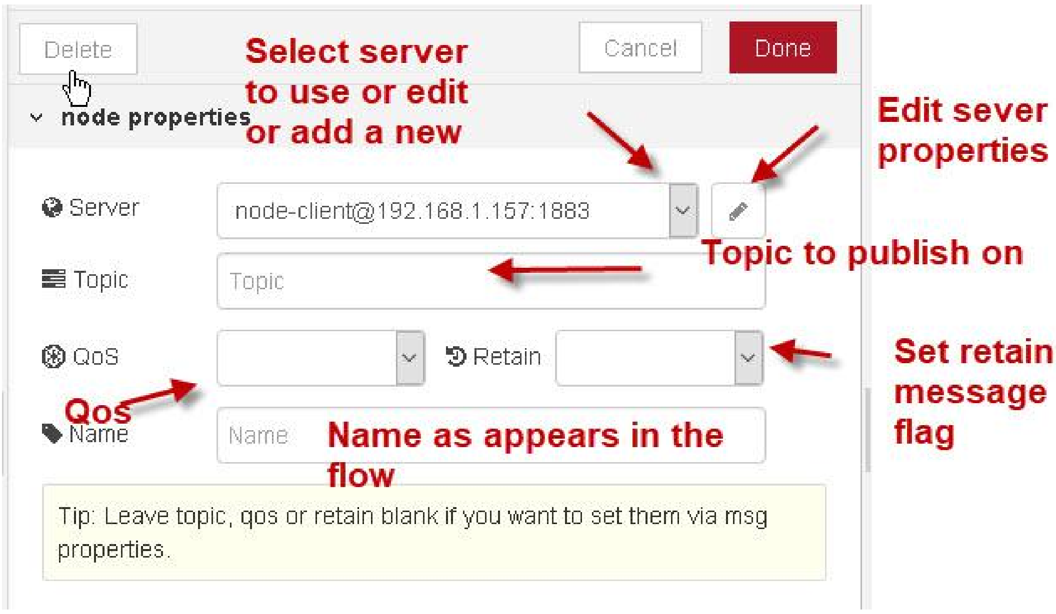

Figure 4.14: MQTT Node-RED Integration
Source: (Industrial & Results, 2021)

As shown in Figure 4.14, the first step is to drag the MQTT in and MQTT out nodes from Node-left RED sidebar to the dashboard and double-click them. In the connection tab, enter the name of your choice, as well as the IP address of the server where the MQTT broker is installed, the port is usually 1883. Finally, go to the security page and input your MQTT password and username.

Figure 4.15: Security tab

As demonstrated in Figure 4.15, we can configure our system's security there. Anyone who does not have a password and a username will be unable to access the MQTT as usual. Security is critical to preventing hackers into our system. The username can be substituted with the message broker password that we specified during the message broker installation.

CHAPTER 5

CONCLUSION

5.1 Introduction

This chapter summarised the research's key findings. The study's findings provided an overall description of the case study. The limitations or challenges encountered while doing this research were also reported, along with recommendations for future research.

5.2 Conclusion

In conclusion, the Node-RED platform is particularly useful since it provides a cost-efficient, user-friendly, and dependable solution for any system, including smart laboratory systems. As a result, by controlling and monitoring the system wirelessly, it is possible to improve products and services. It is more efficient when Node-RED collaborates with a cloud system because we can access the Node-RED dashboard from anywhere. Next, the Node-RED platform is a superior solution for controlling and monitoring appliances, particularly in industries. With this approach, all tasks may be completed in a more productive and effective manner.

5.3 Recommendation

Overall, the system will be more effective if it is upgraded and modified with various components such as hardware, access, and website. There is advice to utilise actual equipment such as a door lock and a bulb for hardware. For example, we can monitor electricity usage in the laboratory and manage real-world equipment from there via a website. Next is the entry part, perhaps the system can be improved to incorporate technology, such as an RFID card to enter the laboratory or face recognition to access the laboratory. This technology can increase laboratory security by limiting access to persons who have a card ID or facial detection. Finally, in the future, we will be able to develop the laboratory by accessing the website wherever we go. There is a plan to create a separate website with a unique URL to access the laboratory system. This means that anyone can access the website to view the real-time temperature and humidity, but only the administrator or user can control the appliances or security in the laboratory.

REFERENCES

Amruta N. Banagar, & Rajshankar Khattar. (2020). IoT based Smart Laboratory System. *International Journal of Engineering Research And*, *V9*(01), 315–318. https://doi.org/10.17577/ijertv9is010178

Balaji, R., Mayuri, A. V. R., Ramadevi, N., & Anirudh Reddy, R. (2020). Advanced implementation patterns of internet of things with MQTT providers in the cutting edge communications. *Materials Today: Proceedings*, *xxxx*. https://doi.org/10.1016/j.matpr.2020.11.090

Cards, G., Deals, T., Service, C., Valentine, S., Items, S. S., Type, L., & Color, L. (n.d.). *3 Claps ON / Off Remote Sound Control LED Light Bulb Perfect for Bedroom Table Lamp Upgrade , Wireless Sound Activated ON / Off Solution , Clap Detection , Smart Home , As Seen On TV , Soft White , 8-Watt Have a question ?* 2–9.

Chooruang, K., & Mangkalakeeree, P. (2016). Wireless Heart Rate Monitoring System Using MQTT. *Procedia Computer Science*, *86*, 160–163. https://doi.org/10.1016/j.procs.2016.05.045

Domínguez, M., González-Herbón, R., Rodríguez-Ossorio, J. R., Fuertes, J. J., Prada, M. A., & Morán, A. (2020). Development of a Remote Industrial Laboratory for Automatic Control based on Node-RED. *IFAC-PapersOnLine*, *53*(2), 17210–17215. https://doi.org/10.1016/j.ifacol.2020.12.1741

Industrial, A., & Results, V. (2021). *Configuring the Node-Red MQTT Publish and Subscribe Nodes*. 1–30.

Lekić, M., & Gardašević, G. (2018). IoT sensor integration to Node-RED platform. *2018 17th International Symposium on INFOTEH-JAHORINA, INFOTEH 2018 - Proceedings*, *2018-Janua*(March), 1–5. https://doi.org/10.1109/INFOTEH.2018.8345544

Pi, R., Bulk, M. B., Pi, R., & Ram, M. B. (n.d.). *Product details Article Information*. *12*, 4–5.

Poongothai, M., L., A., & Priyadharshini, R. (2018). Implementation of IoT based Smart Laboratory. *International Journal of Computer Applications*, *182*(15), 31–34. https://doi.org/10.5120/ijca2018917853

Revelivan, C., Wiratama, H., & Hanafiah, N. (2019). ScienceDirect ScienceDirect Door Security System for Home Monitoring Based on ESP32. *Procedia Computer Science*, *157*, 673–682. https://doi.org/10.1016/j.procs.2019.08.218

Security, R., Wireless, K., & Sensor, M. (2022). *Rm32.80 - rm39.00 100%*. 2–6.

Specifications, P. (n.d.). *Rm69.00 - rm99.00 81%*. 2–5.

Tabaa, M., Chouri, B., Saadaoui, S., & Alami, K. (2018). Industrial Communication based on Modbus and Node-RED. *Procedia Computer Science*, *130*, 583–588. https://doi.org/10.1016/j.procs.2018.04.107

Tsao, Y., Cheng, F., Li, Y., & Liao, L. (2022). *An IoT-Based Smart System with an MQTT Broker for Individual Patient Vital Sign Monitoring in Potential Emergency or Prehospital Applications*. *2022*.

WolkAbout. (2019). *Connect Your Device to WolkAbout IoT Platform Using Node-RED*. 2–5. https://wolkabout.com/blog/wolkconnect-node-red/

Yang, A., Zhang, C., Chen, Y., Zhuansun, Y., & Liu, H. (2020). Security and Privacy of Smart Home Systems Based on the Internet of Things and Stereo Matching Algorithms. *IEEE Internet of Things Journal*, *7*(4), 2521–2530. https://doi.org/10.1109/JIOT.2019.2946214

APPENDICES

Appendix A: Overall Actual Product

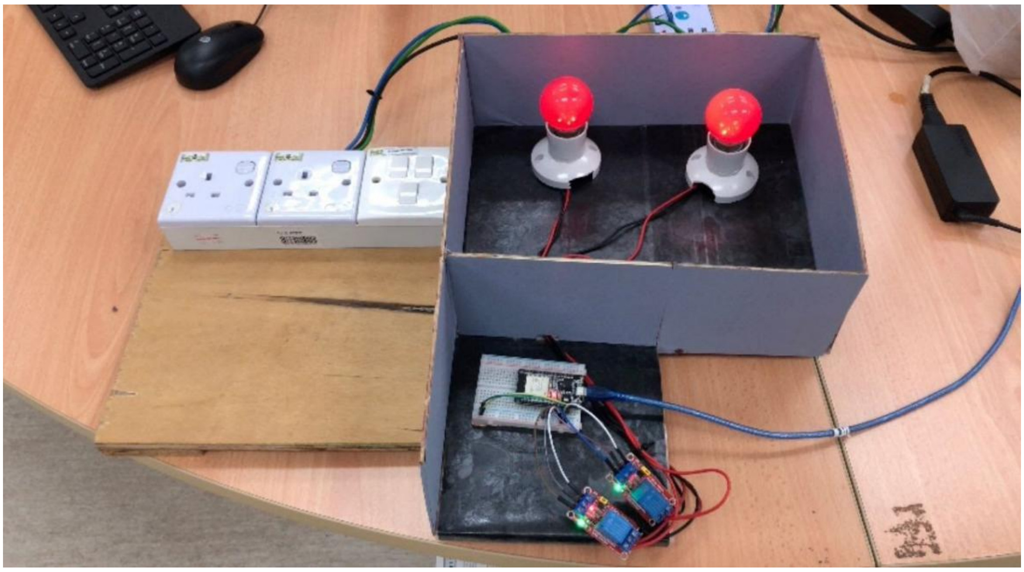

Figure 5.1: Hardware of Lamp system

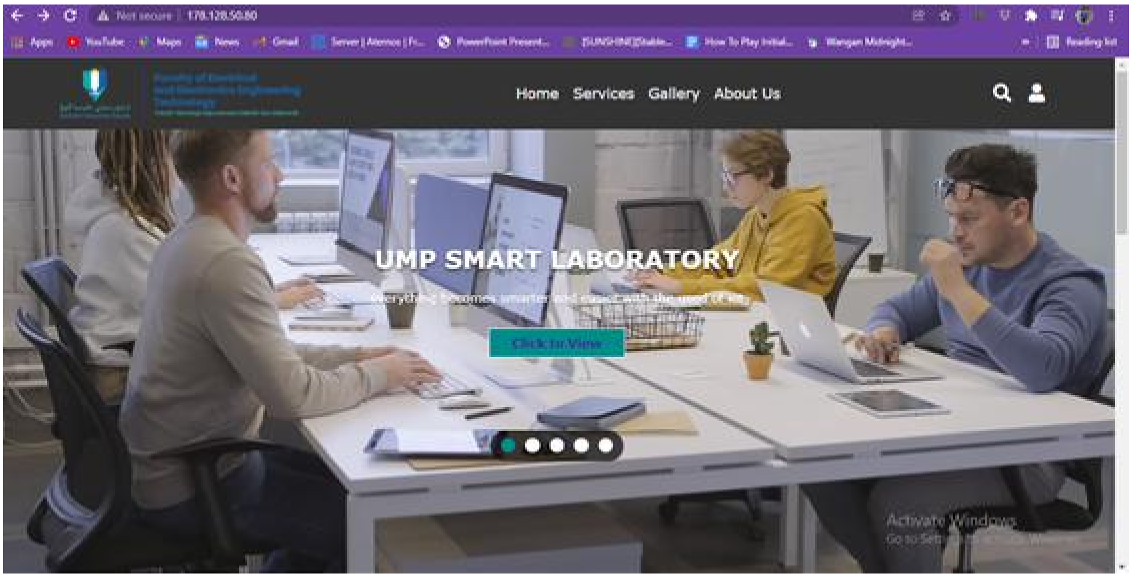

Figure 5.2: Website of Smart Laboratory

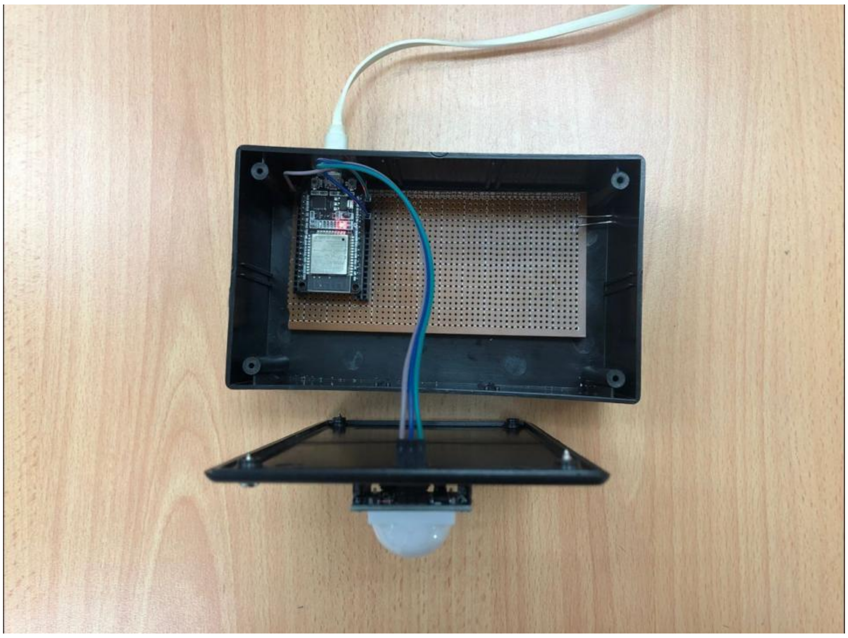

Figure 5.3: Prototype of PIR Sensor

Figure 5.4: Prototype of ESP Camera

Figure 5.5: Prototype of Temperature and Humidity Sensor

Appendix B: Twilio

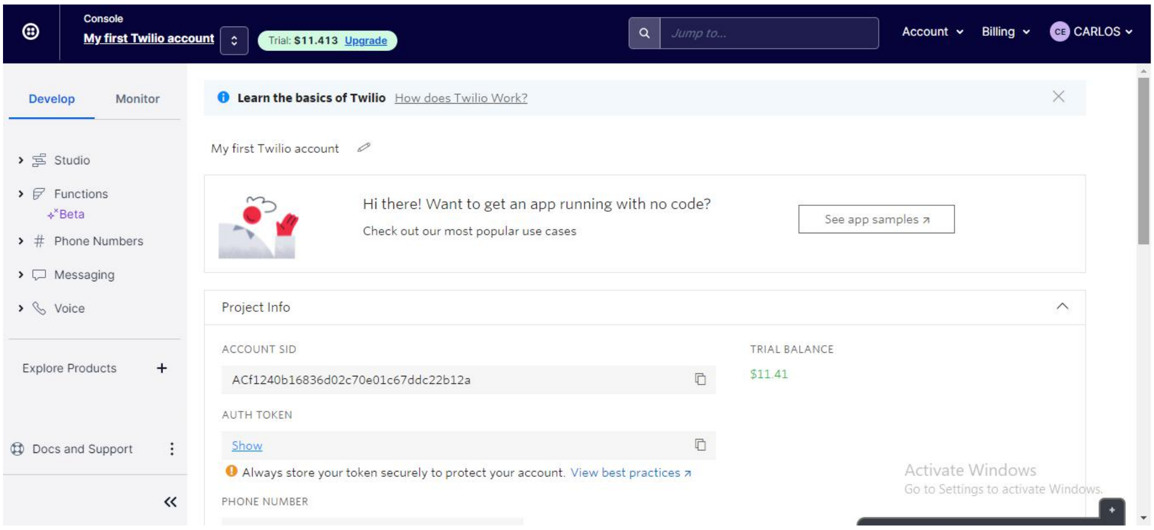

Figure 5.6: Account of Twilio

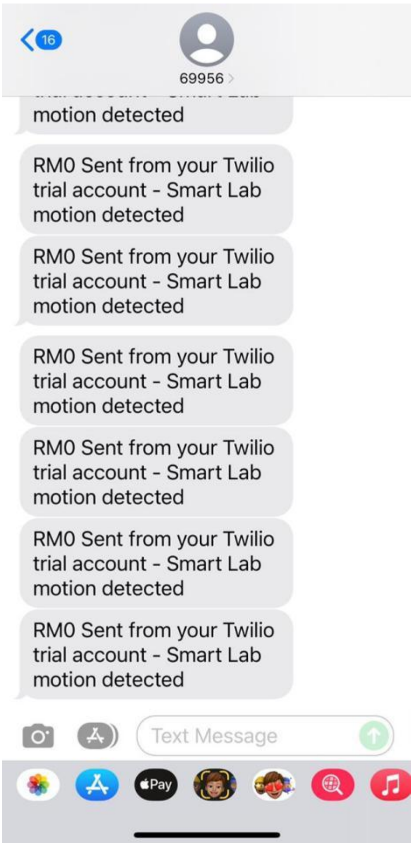

Figure 5.7: SMS from Twilio

Appendix C: Phpmyadmin

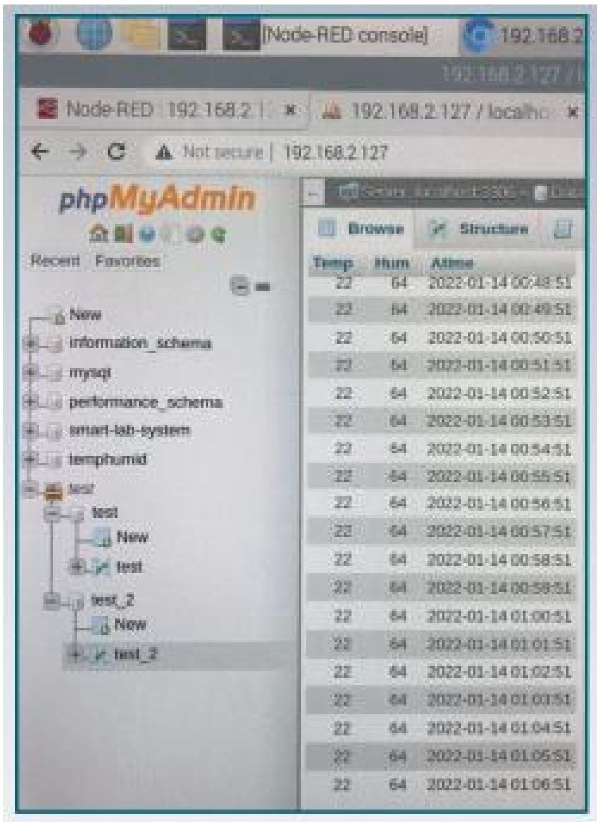

Figure 5.8: Reading of Temperature and Humidity sensor in database

Appendix D: Gantt Chart Project Planning for SDP2

Table 5.1: Project Gannt Chart

SMART LABORATORY PREPARATION PLANT																				
	OCT 21				**NOV 21**				**DEC 21**				**JAN 22**				**FEB 22**			
TASK	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
SDP2 briefing	■																			
Project Meeting		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
Develop prototype													■	■	■	■				
Specify detail requirement			■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
First draft Of thesis																■	■			
Apply final correction												■	■	■	■	■	■			
Implementation Final design												■	■	■	■	■	■			
SDP2 presentation																■				
Thesis Evaluation																	■			

Appendix E: Coding of PIR Sensor

```
#include <WiFi.h>

#include <PubSubClient.h>




#define wifi_ssid "Innovation_02_2.4G"

#define wifi_password "ftkee1000"


#define mqtt_server "192.168.2.127"

#define mqtt_user "gaurav"

#define mqtt_password "rasp1234"


#define pir_sensor "pirsensor"




int ledPin = 12;              // choose the pin for the LED

int inputPin = 14;            // choose the input pin (for PIR sensor)

int pirState = LOW;           // we start, assuming no motion detected

int val = 0;                  // variable for reading the pin status
```

```
WiFiClient espClient;
PubSubClient client(espClient);


void setup() {
  pinMode(ledPin, OUTPUT);      // declare LED as output
  pinMode(inputPin, INPUT);     // declare sensor as input


  Serial.begin(115200);
  setup_wifi();
  client.setServer(mqtt_server, 1883);
}


void setup_wifi() {
  delay(10);
  // We start by connecting to a WiFi network
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(wifi_ssid);
```

```
  WiFi.begin(wifi_ssid, wifi_password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void reconnect() {
  // Loop until we're reconnected
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    // Attempt to connect
    // If you do not want to use a username and password, change next line to
    // if (client.connect("ESP8266Client")) {
```

```
    if (client.connect("ESP8266Client", mqtt_user, mqtt_password)) {
      Serial.println("connected");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // Wait 5 seconds before retrying
      delay(5000);
    }
  }
}



void loop() {
  val = digitalRead(inputPin);// read input value

  if (val == HIGH) {            // check if the input is HIGH
    digitalWrite(ledPin, HIGH);  // turn LED ON
    if (pirState == LOW) {
```

```
      // we have just turned on
       char pirState[8];
     dtostrf(val, 1, 2,  pirState);
      Serial.println("Motion detected!");


      // We only want to print on the output change, not state
      pirState = HIGH;
    }
  } else {
    digitalWrite(ledPin, LOW); // turn LED OFF
    if (pirState == HIGH) {
      // we have just turned of
      Serial.println("Motion ended!");



      // We only want to print on the output change, not state
      pirState = LOW;
    }
  }
}
```