DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM

BY

KONG WAI KIN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

Jan 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: **DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM**

Academic Session: Jan 2024

I **KONG WAI KIN**

(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:
2252 Jalan batu sinar, taman _
bandar baru, Kampar, Perak

Miss Oh Zi Xin
Supervisor's name

Date: 26/04/2024

Date: 26/04/2024

FACULTYOF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 26/04/2024

SUBMISSION OF FINAL YEAR PROJECT

It is hereby certified that ______***Kong Wai Kin***____________________ (ID No: _***20ACB06669***__) has completed this final year project entitled "*Development Of IOT-Based Smart Laboratory System*" under the supervision of Miss Oh Zi Xin from the Department of FICT, 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,

____
(*KONG WAI KIN*)

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM**" 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 : ______KONG WAI KIN_____

Date : 26/04/2024

ACKNOWLEDGEMENTS

My profound gratitude and admiration go out to my supervisor, Miss Oh Zi Xin, for providing me with this wonderful chance to work on an IoT based project. This marks the beginning of my career as an Electronics and Computer Engineer. Many thanks for it.

To a very precious person in my life, my mother, for her love, tolerance, and unwavering support. She has also supported me through difficult times, helped me with some of my hardware development, and given me constant encouragement along the way.

ABSTRACT

The COVID-19 pandemic has shown how crucial it is to avoid physical contact, particularly for healthcare professionals performing research and diagnosis. This project suggests creating an IoT Lab to reduce the risk of virus exposure during laboratory work. A key feature is integrating a robotic arm for contactless laboratory operations, managed wirelessly via an ESP WROOM 32 microcontroller. The IoT Lab uses a variety of sensors to increase safety and accuracy. The MQ-2 gas sensor identifies potentially dangerous compounds or smoke early. The DHT22 temperature and humidity sensor provides real-time environmental data for maintaining regulated experimental settings. An LDR sensor is also used to control the lab's illumination. The fan may be intelligently regulated using sensor data to control the airflow, and the relay module makes it possible to activate or deactivate the lamp remotely. The logic level converter is also used to convert the signal from 3.3V to 5V to avoid damaging the circuit board. The BLYNK software enables smooth remote monitoring and management of the system through a user-friendly smartphone application. Researchers and healthcare professionals may conveniently and safely use the robotic arm while monitoring sensor readings, starting tests, and initiating investigations from their mobile devices. The Fritzing program is used to design and prototype the circuit board during development, ensuring every component's best connection and operation. This iterative design approach guarantees the IoT Lab's dependability and sturdiness with a limited budget. Establishing an IoT lab that considerably reduces the danger of physical contact for medical professionals during experiments and diagnostics is the main point of this project. This project aims to revolutionize laboratory practices in COVID-19 or upcoming deadly virus outbreaks by integrating cutting-edge technologies, such as a robotic arm and a network of sensors, and by offering remote control through the BLYNK program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Ω	Ohm (resistance)
µl	microliter
<	Less than

LIST OF ABBREVIATIONS

ppm	Parts-per-million
LDR	Light Dependent Resistor
LED	Light-emitting diode
IoT	Internet of Things
DC	Direct Current
V	Voltage
mA	Milli-ampere
cm	centimeter
Mm	millimeter
k	thousand
GPIO	General Purpose input/output
Mhz	Megahertz
VDC	Volts Direct Current

Chapter 1

Introduction

The year 2019 presented unparalleled challenges worldwide due to the COVID-19 pandemic. Surveillance data supplied to the World Health Organisation (WHO) between January 2020 and May 2021 indicated 3.45 million fatalities attributable to COVID-19. From January 2020 until May 2021, approximately 80,000 to 180,000 health and care workers might have perished from COVID-19, according to the World Health Organisation (WHO) estimate, with the prediction of 115,500 fatalities as the middle range [1]. There are a few factors that contribute to the high death rate among healthcare professionals, but the working circumstances play a crucial role. The absence of personal protective equipment, the COVID-19 vaccine shortage, the labour strike, stress, burnout, and other mental illnesses all added to the death toll [2].

The use of IoT in the medical field has increased drastically since the COVID-19 outbreak. It is crucial to protect professional medical care workers from being infected by the deadly coronavirus or being a carrier and infecting their family members or friends. Many new inventions using IoT have emerged since then. Mohammed et al. [3] created a sophisticated helmet equipped with thermal imaging technologies for spotting sick individuals in a crowd. Additionally, the facial recognition technology was installed.

The development of surgical robots allowed surgery performance without physically touching the patient [4]. The Arthrobot which utilised voice instructions to help with patient placement during an orthopaedic surgery case [5], and the Unimation Puma 200, used to orient a needle for brain biopsy [6], were the first surgical robots. Automated Endoscopic System for Optimal Positioning (AESOP) is the first robotic surgical system to get FDA approval. In 1996, researchers developed a complete robotic surgical system with seven degrees of freedom, tremor reduction, and motion scaling [7]. A French patient had the first long-distance tele-surgical procedure, a laparoscopic cholecystectomy in Strasbourg with the surgeon in New York, which was performed with the aid of ZEUS [8]. Another innovation was the Intuitive Surgery Da Vinci robotic system, which is utilised in several surgical specialities for various surgical operations and can carry out technically difficult procedures [9, 10].

The increase in demand for IoT-based technology creates chances for engineering and software engineering students to help develop IoT-based technology that will help the nation's economy and better meet the expectations of Industrial Revolution 4.0.

To effectively address the diverse learning demands of students in the twenty-first century and align with the demands of the present industrial and technological period, contemporary resources like digital smart laboratories and smart gadgets are essential [11].

A paradigm change in teaching and learning has also been brought about by IoT-based smart classrooms and laboratories in higher education, particularly in engineering and software engineering [12].

1.1 Problem Statement and Motivation

The most effective ways of COVID-19 sample collection are through oral swabs, nasal swabs and saliva samples. So, this requires the trained medical care worker to equip with proper personal protective equipment or commonly referred to as "PPE", to minimize exposure to the deadly virus when collecting a sample from the potential patient. All the steps in detecting COVID-19, which involve sample collection, and pipetting the sample for RNA extraction until loading the sample into RT-PCR, require a trained healthcare worker to operate. It might be a laborious job and requires few properly trained healthcare personnel to work; however, due to the high infectious property of the Coronavirus, which contribute to the high dead rate of healthcare worker, the workforce is becoming a massive problem for many countries quickly diagnose and prevent further outbreak of Coronavirus. Many healthcare workers work overtime due to a lack of workers, which burdens their psychological and physical health. The procedure for sample collection and Coronavirus detection using RT-PCR, which requires human interference to complete the task [13].

In the traditional laboratory, most equipment or instruments, such as an incubator, water bath, lab refrigerator, and others, require human adjustment on temperature, humidity, light intensity and carbon dioxide concentration. The traditional lab might need to be friendly to some experiments that require continuous monitoring of environmental factors. For example, when culturing, some thermosensitive bacteria require continuous monitoring. On the other hand, vaccines are also sensitive to temperature changes. Freezers must maintain temperatures between -50 degrees Celsius and -15 degrees Celsius. So, it is crucial to have an IoT temperature monitoring sensor to ensure the freezer is working in the best condition and can trace back the records [14].

The thesis aims to propose a project on an approach to convert conventional laboratories to smart-lab. The designed smart laboratory will help University students manage different systems like light, temperature, humidity, and carbon dioxide concentration using mobile application technology and IoT concept. Besides, this will allow the students to explore the Internet of Things concept and contribute to more IoT inventions and innovations.

Next, the design will monitor critical parameters such as temperature and humidity. The system includes sensors and components like fans to adjust and maintain optimal growth conditions of bacteria. The robotic arm is programmed to hold the surgical pipette and do a simple experiment using a user-friendly interface. The mobile application will be developed to enable the researcher to control temperature, humidity, and robotic arm movement interactively. The system will log temperature, humidity, and robotic arm actions over time. Researchers will analyze the collected data to optimize the experimental protocols and ensure reproducibility. Besides, this project will improve the torque of the robotic arm by replacing SG90 with a higher torque MG996R using ESP WROOM 32 with the help of an external power supply and logic level converter. Finally, the system is expected to enhance efficiency and safety under limited human manual interruption.

1.2 Objectives

The project aims to improve the grab ability of the robotic arm by modifying the number of grippers to hold the laboratory pipette by controlling the robotic arm via Wi-Fi to do the pipetting in the laboratory. This project can reduce the risk of healthcare workers touching the deadly virus.

This project also aims to reduce the dependence on the power source provided by the ESP WROOM 32 to supply power to all the hardware. The external power source can reduce the burden of ESP WROOM 32 by supplying stable and constant 3.3V and 5V power to all the sensors and hardware. It can supply enough power to six high-torque servo motors to lift heavy-weight loads without burning out the ESP WROOM 32. Besides, the temperature sensor and Light Dependent Resistor (LDR) sensor will be added in this project to monitor the temperature of the laboratory and reduce the temperature by activating the DC fan and buzzer. Meanwhile, depending on the light intensity, the LDR photo sensor will detect the light intensity inside the laboratory and control light intensity automatically; this step is crucial so that medical personnel do not need to touch the switch to turn off the light.

Another objective of this project is to develop a user-friendly BLYNK application to remote monitoring and enable real-time data visualization and analysis through the BLYNK app. Users can control the robotic arm to hold a heavier load like a pipette using the BLYNK application without physical contact using MG996R servo motors.

1.3 Project Scope and Direction

This project describes the creation of a sophisticated Internet of Things (IoT) lab prototype meant to improve the capabilities of the current study, which focuses on integrating a robotic arm with an upgraded set of three grippers, temperature and humidity sensors, gas sensors, DC fan, light intensity sensors, and ESP WROOM 32.

Integrating the robotic arm enables more complex and accurate activities, requiring three grippers to grasp the pipette by enhancing and optimizing the robotic arm's functionality, control, and range of motion. A higher torque servo motor MG996R enables the system to provide better support to the robotic arm to perform more complex operations. An external power supply will support the high torque servo motor with the logic level converter to convert the signal from 3.3V to 5V to protect the microcontroller from electrical damage. A special BLYNK application that enables users to interact with the IoT lab, monitor sensor data, and remotely control the robotic arm will be created to provide seamless and user-friendly remote access and control.

1.4 Contributions

The project had changed how a traditional laboratory operates. With the robotic arm, healthcare workers will no longer need to manually do pipetting and handling the specimen which contains deadly viruses, risking their lives and reducing the possibility of carrying the virus and infecting their family. This project will contribute to developing and underdeveloped countries to prepare for the coming more infectious and deadly virus. The current market robotic arm is too expensive and might only be available for some countries. This project provides an alternative to a cheaper, aim-focused robotic arm. This robotic arm can reduce the stress of healthcare workers, especially when lacking staff and PPE. With the Internet of Thing technology, during a critical situation or the next deadly virus outbreak, first-world countries can assist other countries through the Internet or online to perform remote research.

By transforming traditional laboratories to smart laboratories, University students gain exposure to the Internet of Thing technology which can trigger their invention of more advanced IoT devices or equipment. This exposure effectively prepare them for the Industrial Revolution 4.0.

1.5 Report Organization

This report is organized into 7 Chapters. Chapter 1 is the introduction. The introduction includes the problem statement and motivation, the objectives of this project, the project scope, the contribution upon completion, and the report organization, which explains how this project's chapters are organized.

A literature review is the second chapter. This chapter reviews the development done previously and makes a comparison of existing systems.

The third chapter is the methodology. In this chapter, a system architecture diagram is added. This diagram shows how all the system components interact with each other. Use case diagram is the diagram to show the interaction between the system and users; this includes showing all actions that users can perform and how the system responds to the actions. Lastly, an activity diagram is added in this chapter; it is the overview of the workflow of the project.

The fourth chapter is the system design. This includes the system block diagram. The system block diagram focuses on the functional blocks of the project and the relationship between them. The system components specification explains the specification of all the components used in this project. The circuit and components design shows the breadboard diagram, wiring, and connection of the elements. Lastly, a detailed explanation of the system development. This includes how to build the robotic arm, connect to the server and internet, upload code to the microprocessor, set a threshold for the sensor, set output to respond system action, set up the virtual pin, set up Blynk, and modify to the servo motor to become a water pump. This is a significant chapter as users are to rebuild the whole system by referring to the instructions in this chapter.

The fifth chapter is system implementation. This chapter includes hardware and software setup. The setting and configuration of system software and hardware. This chapter also provides system operation, implementation issues and challenges, and concluding remarks.

The sixth chapter is about system evaluation and discussion. This is the chapter on testing the system. The setting up of testing and the result of testing are included in this chapter. The challenges of the project and the evaluation of objectives set earlier are discussed in this chapter. Finally, it is the conclusion remark for this whole chapter.

Chapter 7 is about conclusion and recommendation. It concludes the whole project and provides appropriate future improvement recommendations.

Chapter 2

Literature Review

2.1 Previous Projects on IoT laboratory

2.1.1 IoT based Robotic Arm for Student Online Learning

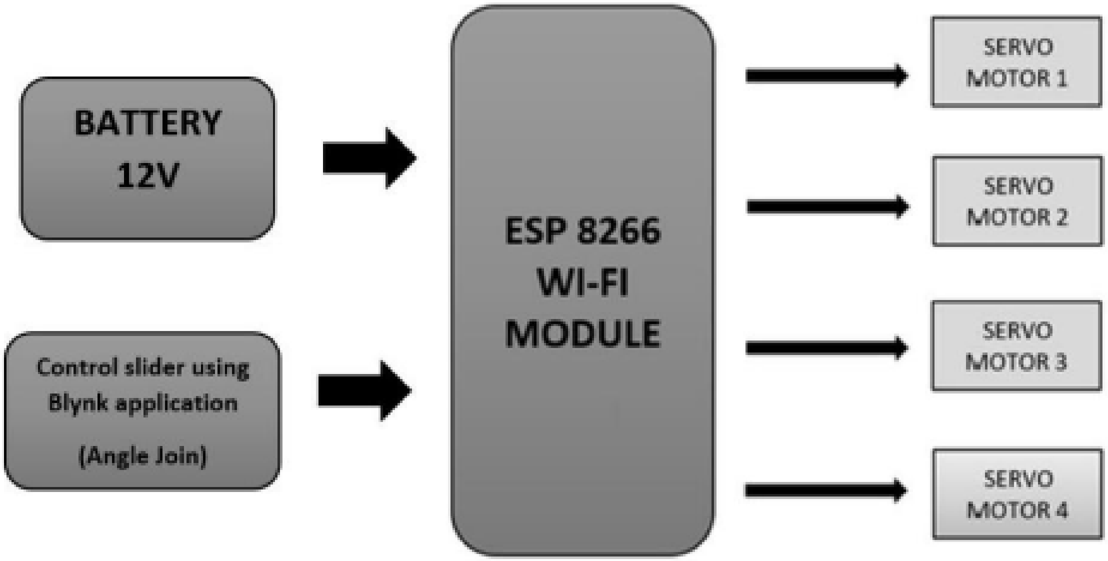


Figure 2.1 Block Diagram of Robotic Arm using IoT [15]

The block diagram of the robotic arm controller for robotic education is shown in Figure 2.1 [15]. The study used an ESP 8266 Wi-Fi module as a microcontroller to operate the robotic arm over the Internet of Things (IoT) using the Blynk application. The robotic arm had four DC servomotors, designated as servomotors 1, 2, 3, and 4. The robotic arm's four joints, Joint 1, Joint 2, Joint 3, and Joint 4 are together referred to as the servomotor. The slider on the Blynk application serves as a control interface, allowing the user to regulate the angle movement of the robotic arm utilizing IoT. The research also established a connection between the IoT and the robotic arm using the ESP 8266 Wi-Fi module. Four degrees of freedom on the robotic arm translates to four joints and four links on the robot. Joint 1 is characterized as a fundamental motion capable of a full 360-degree rotation and the capacity to shift right and left. The robotic arm can move forward and backward from 0 to 180 using joints 2, 3, and 4. The currently operational slider from the Blynk application will adjust the joint angle. A fast internet connection, a medium-speed internet connection, and a slow internet connection were used to evaluate three distinct internet connections. With a reliable internet connection, controlling the robot arm moved more quickly and more efficiently [15].

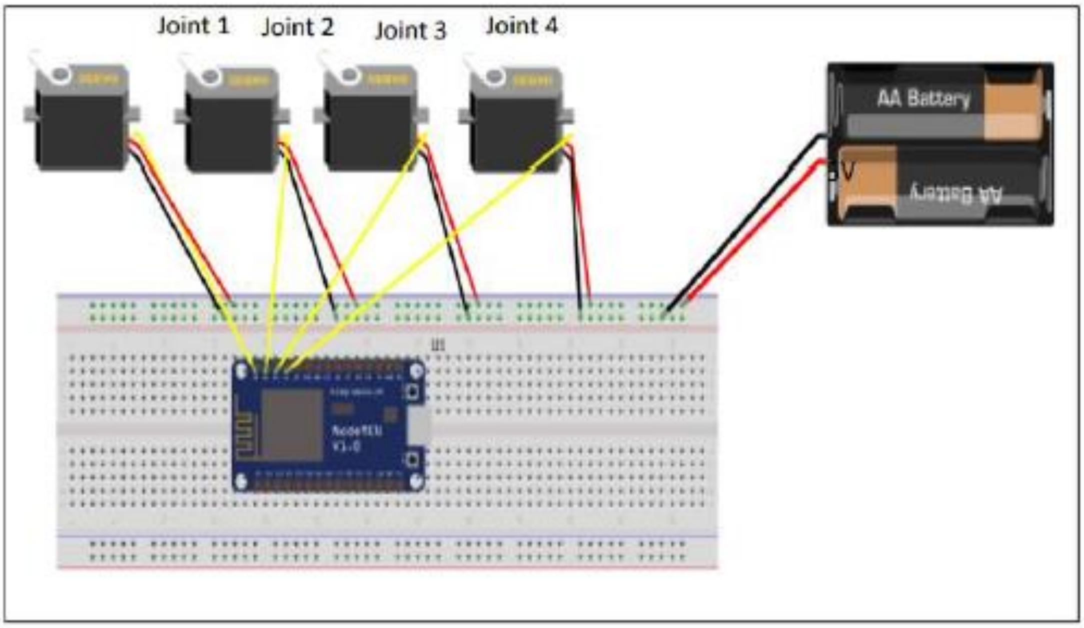


Figure 2.2 Schematic Circuit Diagram [15]

The Schematic Circuit diagram was displayed in [15, Figure 2.2]. It contained four servo motor connected to the breadboard, Node MCU and power supply. In the previous project, the robotic arm was tested with a simple rotation movement; however, to convert the traditional laboratory to an intelligent lab, the robotic arm must hold the pipette and do simple pipetting.

2.1.2 IoT based Temperature and Humidity Controlling using Arduino and Raspberry Pi

Asad [16] had utilized a Raspberry Pi and Arduino UNO together for monitoring, data saving, and data collection. DHT -11 Temperature Sensor was used to measure humidity and temperature [16]. Then, the results were graphically analyzed by one ThingSpeak platform software and ESP8266 Wi-Fi module. Figure 2.3 shows the block diagram setup for the project.

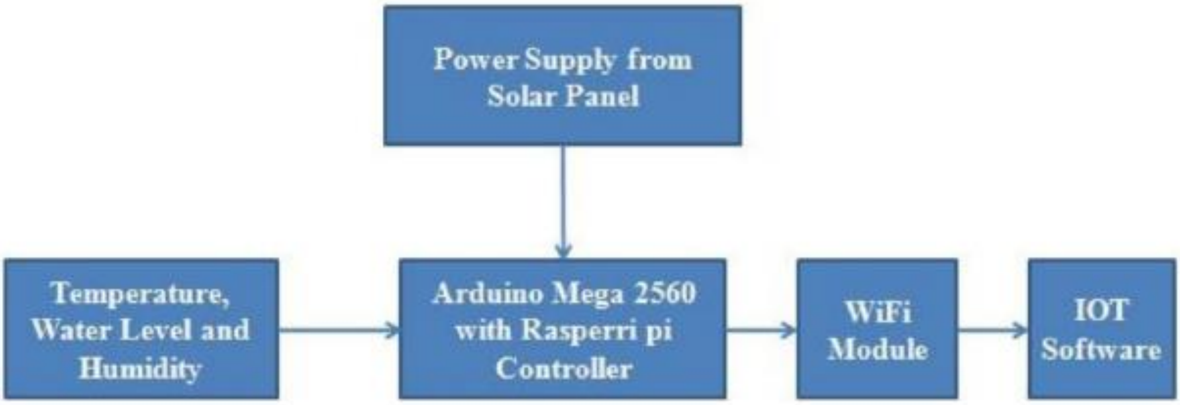


Figure 2.3 Block diagram for IoT based Temperature and Humidity Control [16]

HTU-211D sensor was used to calculate the temperature and humidity value of the particular surface area. Arduino UNO with Raspberry Pi microcontroller was used to control the sensor data. The module was worked on each saver and server and connected to local area network (LAN) to facilitate data transmission over the internet. The ESP8266 LAN module acted as a protocol shopper and sent data to ThingSpeak Server. ThingSpeak was used for the IoT platform for data collection and storage purposes. The collected data was sent to the farmers live via GSM to their cell phones. In this project, the sensor successfully senses the water level and transfers the data to the farmer through GSM, which can control the motors to pump water to their plants. The values collected from the sensors were uploaded within the stipulated period through the ESP-8266 Wi-Fi system. The temperature and standards are measured from the cloud using the ThingSpeak platform. The HTU 211D sensors were offered live data of wetness and temperature to the microcontroller, which was uploaded to the ThingSpeak cloud [16].

2.1.3 The Real-Time Monitoring of Air Quality Using IOT-Based Environment System

The IoT-based system designed by Hashim [17] incorporates of a gas sensor (MQ135), temperature sensor (LM35), and humidity sensor (DHTII) which were connected to Arduino UNO R3, which send data to the cloud using a Wi-Fi module (ESP8266). Smartphones were used to monitor the uploaded data on the server, and it is connected to the microcontroller through a Wi-Fi module. The LM35 was selected since it does not require any external calibration and the DHTI I was preferred for its minimal power consumption. The temperature sensor, LM35, captured the environment's temperature and revealed that the temperature in the city was higher than in the suburbs. Meanwhile, the humidity sensor (DHTI I) also managed to capture the humidity in the city with higher humidity than in the suburb. Lastly, the air quality was measured by a gas sensor (MQ135). Any readings that are more than 300 PPM will be considered unhealthy. Based on the result, the reading showed more than 300 PPM in the city, indicating unhealthy air quality compared to the suburb.

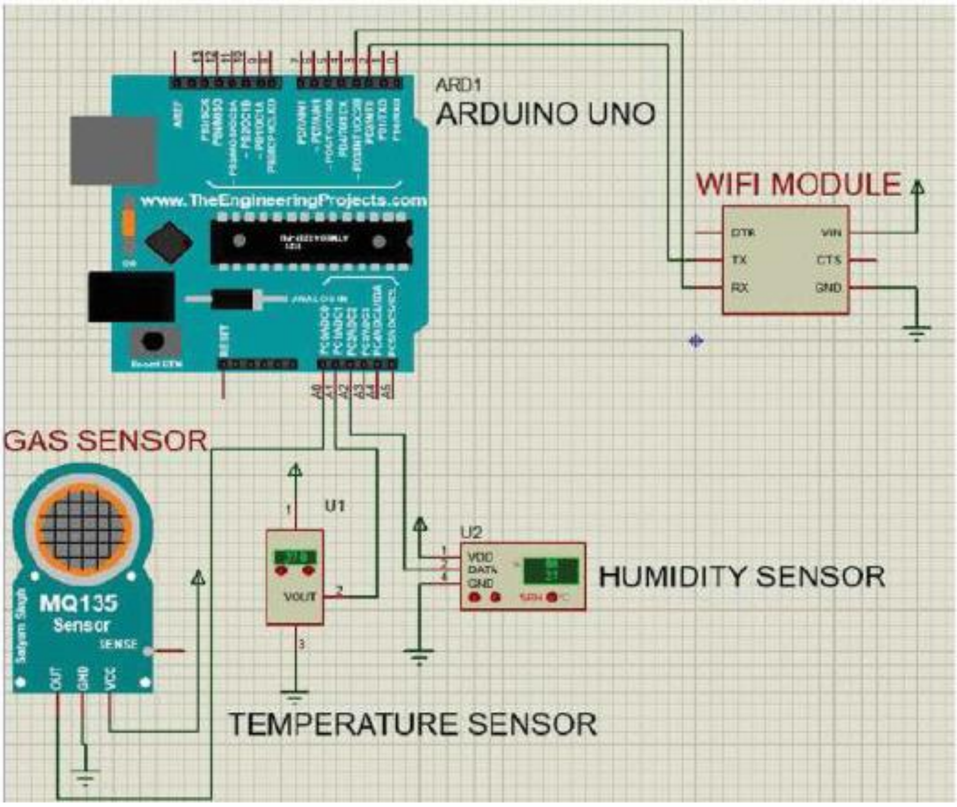


Figure 2.4 Schematic diagram of Arduino UNO R3 [17]

Figure 2.4 shows the schematic diagram of the project. The microcontroller used was Arduino UNO R3. The array of sensors included, the humidity sensor, temperature sensor and gas sensor. The input data from the sensor was processed by Arduino UNO R3 and displayed using the Blynk application. The ESP8266 was connected to Arduino UNO R3 to display the output data to the smartphone using Wi-Fi.

2.1.4 A robotic arm for safe and accurate control of biomedical equipment during COVID-19

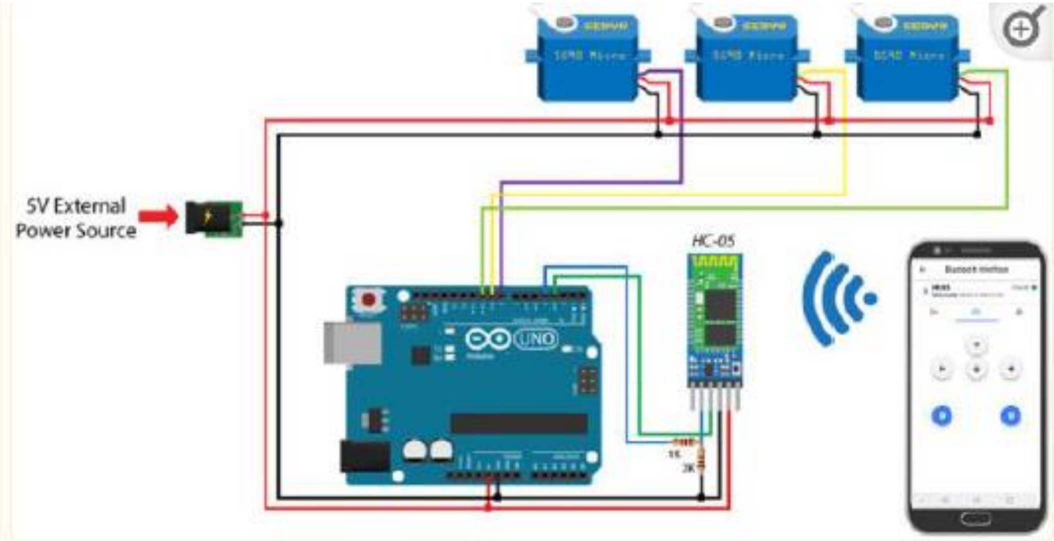


Figure 2.5 Hardware configuration sketch [18]

This project developed a robotic system that is automatically controlled and powered by a smartphone app that connects to the microcontroller remotely. The technologies and tools involved in the design were Unified Modeling Language (UML), Computer-Aided Design (CAD), Additive Manufacturing (AM), Rapid Prototyping (RP), and Integrated Development Environment (IDE). The Arduino Mega 2560 microcontroller was used, based on the ATmega1280 microprocessor. Arduino IDE is an open-source IDE utilized for programming the Arduino to control the four SG90 micro servo motors using the HC-05 Bluetooth module. The servo motor provides 180 degrees and 90 in each direction. The mobile application was developed using Android Studio to remote control and communicate with the Arduino board. The system contained a camera module to provide users with remote visual feedback, so an ESP32-CAM module was employed, which provides internal Bluetooth and Wi-Fi modules. The robotic arm was used to test the ability to press on a calculator panel. From the result obtained, the robotic arm can achieve a 95% success rate in pressing the buttons with and without a mounted pen. A touch-pen stylus was mounted on the robotic arm to improve the button's capability. This achievement contributed to the success of avoiding direct contact between users and the medical equipment or interfaces of civilian use. The module also successfully manages the robotic arm from a far distance via a wireless connection using Wi-Fi or Bluetooth [18].

2.1.5 Cloud-based COVID-19 Patient Monitoring using Arduino

This paper involved using Arduino with several sensors to monitor patients during COVID-19 to prevent the spread of the virus. The project used sensors to collect data and transmit the data to the cloud server to notify the doctors.

The sensors used in this project included MAX30100 to measure heart rate and blood oxygen level, and the MLX90164 sensor was used to detect body temperature, which was connected to the ATmega 2560 microprocessor. The ESP8266-01 was connected to the ATmega 2560 sensor to access the internet via Wi-Fi technology using TCP protocol. Healthcare workers like doctors and nurses do not need to collect the biological measurements of COVID-19 patients physically. OLED Display was also used to display the result. The database system phpMyAdmin was used to store the records. The MAX30100 sensor's precision was compared to an Oximeter with an Industry-standard measuring device. The MLX90614 body temperature sensor was also tested using a digital thermometer. The result showed a high accuracy of the sensors for MAX30100 (Oxygen Saturation and Heart Rate) and MLX90614 (Body temperature sensor) [19].

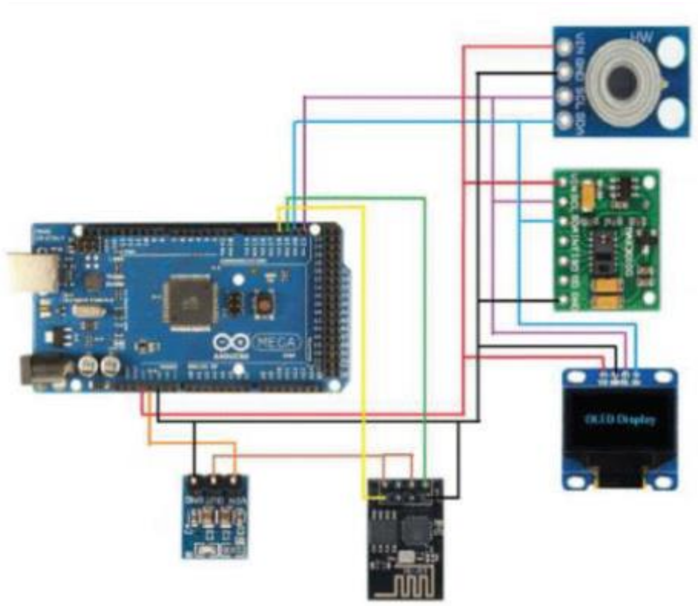


Figure 2.6 Experimental Setup for the Device [19]

Figure 2.6 shows the IoT system prototype used in this project. The Arduino will start to read the pulse rate and ambient temperature from the MLX90614 sensor. Phototransistors and infrared LEDs were used to monitor the heart rate at the fingertip. The reading saved will be transmitted serially using an Inter-integrated circuit to the Wi-Fi module [19].

2.1.6 Internet of Things-based smart helmet to detect possible COVID-19 infections

In this project, a smart helmet was developed to detect possible COVID-19 infections. This project used a thermal camera, optical camera, Arduino Integrated Development Environment, proteus software and Google Location History. Thermal cameras were used to detect high-temperature bodies, which will generate infrared spectra of high intensity. The code was written in JAVA in Arduino Integrated Development Environment (IDE). Proteus software was used to provide real-time simulation, schematics and circuit design. Google Located History stores every location that the user visited. The optical camera will take a picture of suspected people found by thermal cameras to have a temperature higher than 37 ˚C. Blynk was used as an external server [20].

2.1.7 Smart-Lab: Design and Implementation of an IoT-based Laboratory Platform

This project used Application Programming Interface (API) as the middle layer. API allows the interaction of different communities to share a common platform, where they store data directly from the applications in real time. The Representational State Transfer (REST) API was implemented using Python with Flask framework. The components were communicated using API in JSON format, which provides a lightweight data-interchange format. MySQL was selected for data storage. Hypertext Transfer Protocol (HTTP) was used to make requests to the web server and give responses to the user.

In this project design, users could store, monitor and control the data remotely using the web application or Android. ESP32 Thing module and DHT11 sensor were utilized because the chip is equipped with a TCP/IP stack with a Wi-Fi compatible microcontroller, and it is a low-energy consumed chip. The DHT11 sensor was used to detect temperature and humidity, and a thermistor was used to measure surrounding air. The client-side code was then used to store the sensed data in the database. The data was successfully obtained, held in the database and displayed on the web application. Temperature and humidity were managed to be plotted [21].

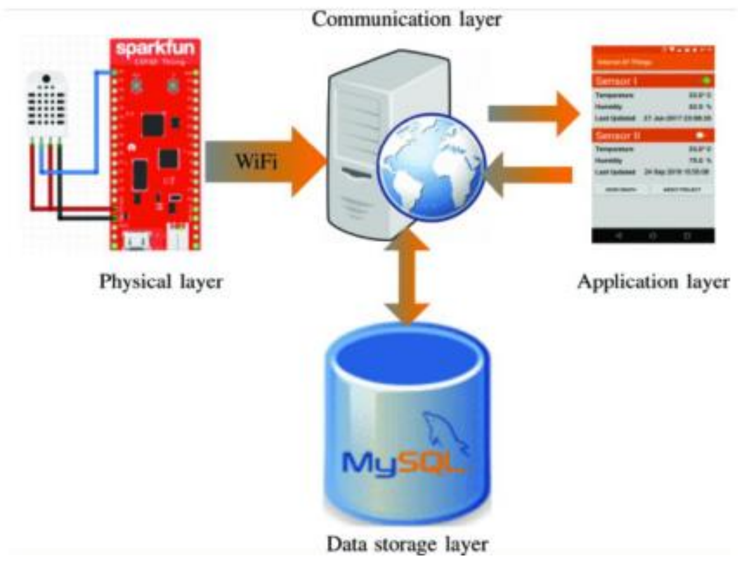


Figure 2.7 Architecture implementation of the IoT platform [21]

In Figure 2.7, this project utilized low energy consumed chip ESP32 and humidity and temperature sensor DHT11 sensor in the physical layer. Hypertext Transfer Protocol was used as the communication layer, MySQL as the data storage layer and Android application as the application layer [21].

2.1.8 Ventilator Using Arduino with Blood Oxygen Sensor for Covid-19

This project used Arduino Uno, a pressure sensor, a 16x2 LCD Module, a Stepper Motor, an Ambu Bag, an Arduino IDE, and a Blood Oxygen Sensor. The circuit diagram was shown in Figure 2.8. Arduino Uno is an ATmega328-based microcontroller. A pressure sensor was used to measure the pressure of liquids and gases. LCD screen displayed 16 characters per line with two lines. A stepper motor was used to convert electrical power into mechanical power. A bag valve mask (BVM) provides positive pressure ventilation to patients who are not breathing adequately. Arduino IDE was used to write and upload programs into Arduino Compatible Boards. The components were written in JavaScript. Blood Oxygen Sensor was used to measure the amount of oxygen via SpO2 measurement. The pressure was supplied in 200 mbar. The Positive end-expiratory pressure (PEEP) was set so that pressure in the lungs had the minimum PEEP pressure, and if that pressure were not achieved, the SIMV mode would allow the patient to take spontaneous breaths. The system was working perfectly and efficiently [22].

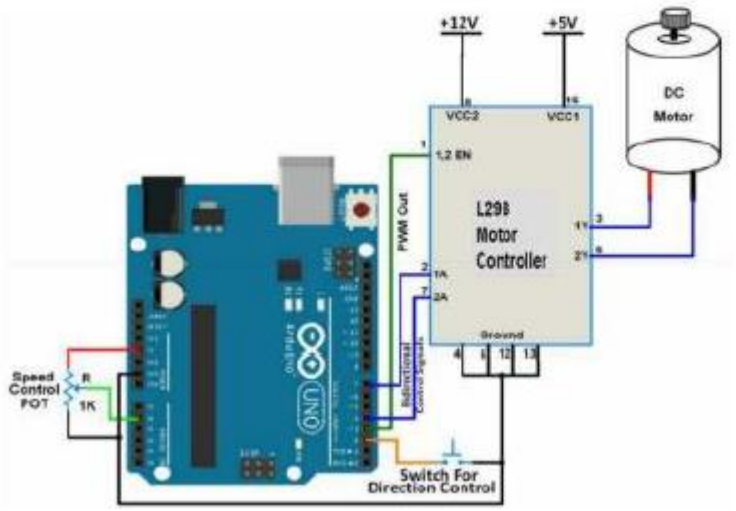


Figure 2.8 Circuit diagram of Blood oxygen sensor [22]

2.1.9 COVID-19 SOP Compliance and Monitoring Electronic System for Business and Public Places Using Arduino Uno

The suggested system comprises a webcam-equipped laptop or personal computer, an RFID reader, an IR sensor, a temperature sensor, and a buzzer. Upon admission, a person's body temperature was measured, and they wore a face mask before washing their hands. If the individual is below 37.6 degrees or below the threshold for admittance, then MATLAB is used for mask detection. Then sanitizer is sprayed, which is followed by the door opening automatically.

The RFID (Radio Frequency Identification) reader collected staff details. Arduino Uno was used as a microcontroller board. This system will automatically identify the person's temperature and face mask to control the coronavirus spread [23].

2.1.10 IoT based Smart Laboratory System

In this project, the system utilized a PIR sensor to detect motion. Then, the DHT11 temperature sensor and TSL2561 luminosity sensor will be activated to detect the temperature and light intensity. The fans of the laboratory will switched on when the temperature is beyond the fixed threshold. Tube lights in the laboratory will be switched off when the luminosity value is beyond the specified threshold. Raspberry Pi version 3B microcontroller was used to process the data, and actions will be taken. The data from the microcontroller was sent to the cloud platform for constant monitoring and analysis. The components include Raspberry Pi 3B, a channel Relay Board, TSL 2561, DHT11 and a passive infrared (PIR) Motion sensor used in this project. This project successfully reduced power consumption by 30 % for smart laboratory over the traditional laboratory [24].

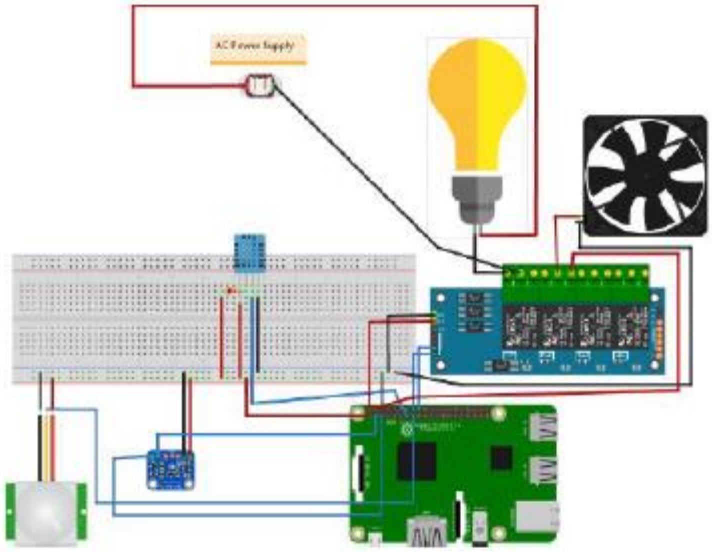


Figure 2.9 Circuit Diagram of the System [24]

Figure 2.9 depicts the connection of the components of the Smart Laboratory Model. The components were Raspberry Pi 3B (Microcontroller), 8-channel Relay Board, TSL 2561 (Luminosity sensor), DHT11 (Temperature & Humidity Sensor), and Passive Infrared (PIR) Motion Sensor [24].

2.2 Limitation of Previous Projects

Based on the project by Wan Faizura [12] and Poonia [20], the Blynk Application was used to create a customized mobile app interface to control the movement of the robotic arm. Blynk Application provides seamless communication between the Blynk cloud server and IoT device by providing firmware and library for various hardware platforms, for example, Arduino, which is the advantage. Besides, the project shows that high internet connectivity was important in controlling the robotic arm.

However, the disadvantage was Wan Faizura [12] only tested a robotic arm in the project, and no other sensors were involved in this project. Additional sensors are highly recommended to become a smart laboratory. Besides, the robotic arm utilized by Wan Faizura [12] only has two grippers, which is insufficient to hold the laboratory pipette. Similar limitations were encountered by Ernesto Ladanza [18] during the design of the robotic system, with only two grippers in the robotic arm. However, the robotic arm can be controlled using Bluetooth via a multi-platform mobile phone app.

Besides, the Arduino used by [19] does not have an inbuilt Wi-Fi module and requires connecting to an external Wi-Fi microprocessor, which reduces the number of pins available to link to other hardware and increase the cost of purchasing additional Wi-Fi module to connect to the internet.

In the project done by L.Barik [16], ThingSpeak was used as a cloud platform to collect and store data from humidity and temperature sensors. The data collected also were transferred to farmer's live via GSM to their phones. A DC fan, buzzer, servo motor, and water pumps were set as output devices to maintain the temperature and water level on the plantation. However, this project did not involve the robotic arm required to hold the pipette. The comparison of limitations between the projects is presented in Table 2.1 below:

Table 2.1 Comparison of limitation across all the projects

	Temperature Sensor	**Humidity Sensor**	**Gas Sensor**	**Mobile Application**	**Wi-Fi / Bluetooth Module**	**Robotic Arm**
Wan Faizura et al[15]	No	No	No	Yes	Yes	Yes
Asad et al[16]	Yes	Yes	No	No	Yes	No
Hashim et al[17]	Yes	Yes	Yes	Yes	Yes	No
Ladanza et al[18]	No	No	No	Yes	Yes	Yes
Hanoon and Aal-Nouman[19]	No	No	No	Yes	Yes	No
Ahlawat and Krishnamurthi[20]	No	No	No	Yes	Yes	No
Khriji et al[21]	Yes	Yes	No	Yes	Yes	No
Devi et al[22]	No	No	No	Yes	Yes	No
Mahendrakuma et al[23]	No	No	No	Yes	Yes	No
Banagar and Khattar[24]	Yes	No	No	No	No	No

2.3 Proposed Solutions for Previous Studies

Due to the Arduino Board used in the previous projects requiring additional installation of a Wi-Fi microprocessor, an ESP WROOM 32 will be used. Espressif Systems created the well-liked and potent Wi-Fi and Bluetooth-enabled module known as the ESP-WROOM-32.

The MQ2 will be used in this project to detect various gases in the air, including carbon dioxide, carbon monoxide, and other gases. The sensor DHT22 will also be involve in this project to detect the temperature and humidity inside the laboratory. Once the temperature sensor and gas sensor detect unusual temperature and carbon monoxide levels, a DC fan will be activated to lower the interior temperature. The buzzer will be activated if smoke is detected. Light Dependent Resistor (LDR) sensor will added in this project to detect the amount of light intensity. LED will light up if the LDR detects low light intensity in the environment. The comparison of the application and hardware used in the previous projects and current project was presented in Table 2.2 below:

Table 2.2 Summary of comparison on the application and hardware used in previous projects and proposed project

	Previous Project	**Current**
BLYNK Application	Yes	Yes
3 Grippers robotic arm	No	Yes
Temperature Sensor	Yes	Yes
Humidity Sensor	Yes	Yes
Gas Sensor	Yes	Yes
MAX30100 sensor	Yes	No
Thermal Camera	Yes	No
GPS location	Yes	No
LDR Photo sensor	No	Yes
ESP WROOM 32 controlling MGR996R servo motor	No	Yes
Use of logic level converter on ESP WROOM 32 for servo motor	No	Yes

Chapter 3

System Methodology

3.1 System Design Diagram

3.1.1 System Architecture Diagram

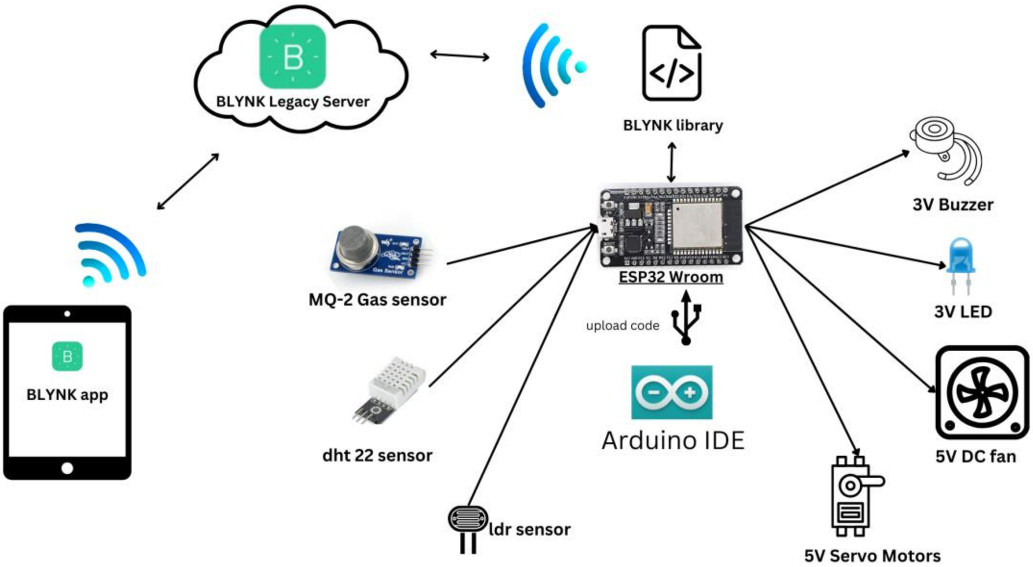


Figure 3.1 System Architecture Diagram

3.1.2 Use Case Diagram and Description

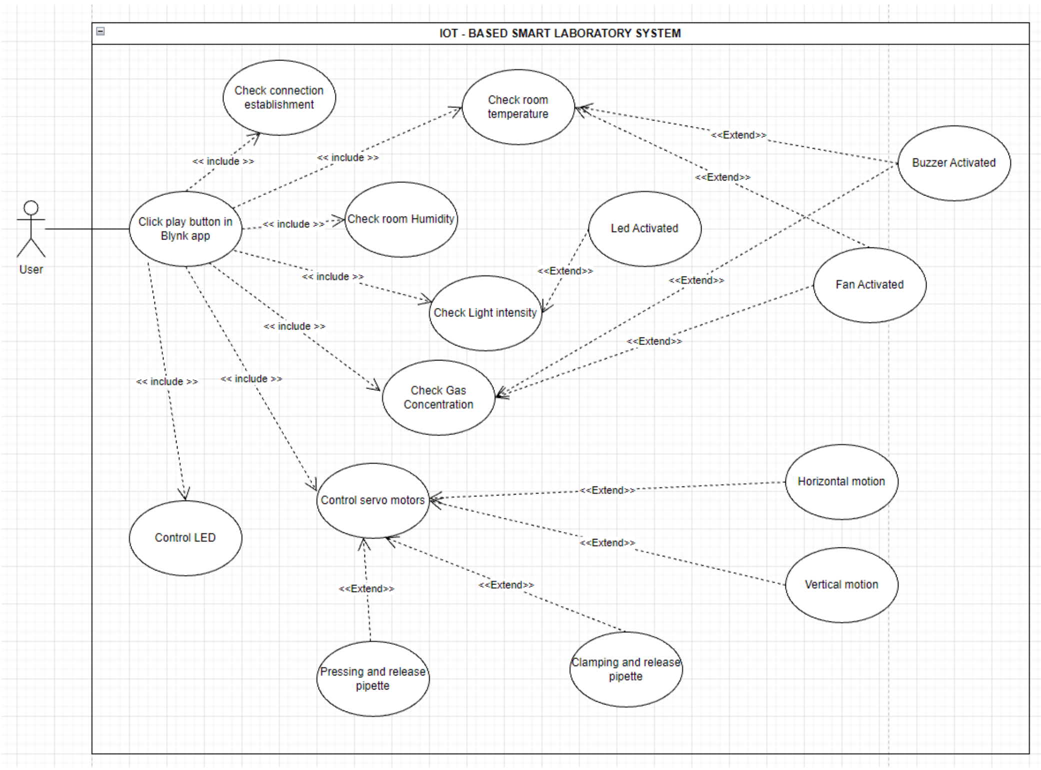


Figure 3.2 Use Case diagram for IoT based smart laboratory system

Table 3.1 Use Case Description for Click play button

Use Case Name:	Click play button	ID: 100	Importance Level: High
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check connection establishment, check room temperature, humidity, light intensity, gas concentration and control servo motors		
Brief Description:	This use case describes how user can make sure the internet and BLYNK server connection established before proceed. This use case also allow user to observe the temperature and humidity.		
Trigger:	User wanted to start the program		
Type:	External		
Relationships:			
	Association:	User	

	Include:	Check Connection Establishment, Check Room Temperature, Check Room Humidity, Check Light Intensity, Check Gas Concentration, Control Servo Motors, Control LED
	Extend:	
	Generalization:	
Normal Flow of Events:	1. User click BLYNK application 2. User click the play button 3. System validate internet connection 4. System validate BLYNK server 5. System display connection status 6. System read input from dht11, MQ2 sensor, and LDR 7. System display the humidity and temperature 8. System will response if threshold exceeded for temperature, light intensity, and gas concentration 9. System display the status on LED, Buzzer, Fan 10. User can control the servo motor	
SubFlows:	Not applicable	
Alternate/Exception Flows:	3a. The system display message “ Internet connection lost” 4a. The system display message “ BLYNK server disconnected”	

Table 3.2 Use Case Description for Control Servo Motor

Use Case Name:	Control servo motor	ID: 101	Importance Level: High
Primary Actor:	User	Use Case Type: Detail, Essential	

Stakeholders and Interests:	User – want to move robotic arm in vertical, horizontal direction, clamping or release pipette and pressing or release pipette	
Brief Description:	This use case describes how user can control the servo motor of the robotic arm to turn left or right, move upward or downward and also holding or pressing the pipette	
Trigger:	User wanted to control the robotic arm	
Type:	External	
Relationships:		
	Association:	User
	Include:	
	Extend:	Horizontal motion, Vertical motion, Clamping and release pipette, Pressing and release pipette
	Generalization:	
Normal Flow of Events:	1. User control the joy stick on BLYNK app for vertical and horizontal motion 2. User press on button to Clamp and release the pipette 3. User press on button to Press and release the pipette	
SubFlows:	Not applicable	
Alternate/Exception Flows:		

Table 3.3 Use Case Description for Check Buzzer and Fan

Use Case Name:	Check buzzer and fan	ID: 101	Importance Level: Low
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check on response of buzzer and fan		
Brief Description:	This use case describes how user can check whether fan and buzzer are activated when temperature and gas concentration is high		

Trigger:	User wanted to control the robotic arm	
Type:	External	
Relationships:		
	Association:	User
	Include:	
	Extend:	Horizontal motion, Vertical motion, Clamping and release pipette, Pressing and release pipette
	Generalization:	
Normal Flow of Events:	1. User control the joy stick on BLYNK app for vertical and horizontal motion 2. User press on button to Clamp and release the pipette 3. User press on button to Press and release the pipette	
SubFlows:	Not applicable	
Alternate/Exception Flows:		

Table 3.4 Use Case Description for Control LED

Use Case Name:	Control LED	ID: 102	Importance Level: Middle
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to switch on or off the LED manually		
Brief Description:	User can control the LED manually or LED will be controlled automatically by LDR		
Trigger:	User wanted to switch on LED manually		
Type:	External		
Relationships:			
	Association:	User	
	Include:		
	Extend:		
	Generalization:		

Normal Flow of Events:	1. User click on LED button to switch on the LED 2. User click on LED button again to switch off the LED
SubFlows:	Not applicable
Alternate/Exception Flows:	1a. The system will switch on the LED when LDR detected light intensity is below the threshold

Table 3.5 Use Description for Check temperature and humidity

Use Case Name:	Check temperature and Humidity	ID: 103	Importance Level: Middle
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check room temperature and humidity		
Brief Description:	This use case describes how user can observe room temperature and humidity and how the system response if threshold exceeded		
Trigger:	User wanted to observe the room temperature and humidity		
Type:	External		
Relationships:			
	Association:	User	
	Include:		
	Extend:	Buzzer activated, Fan Activated	
	Generalization:		
Normal Flow of Events:	1. User click the play button 2. The dht11 sensor will detect temperature and humidity 3. The system will display the result 4. The system will react if threshold exceeded 5. The system will continue to loop and read the reading from sensor		
SubFlows:	Not applicable		

Alternate/Exception Flows:	4a. The system will turn on the buzzer and fan if limit exceeded

Table 3.6 Use Case Description for Check gas concentration

Use Case Name:	Check gas concentration	ID: 104	Importance Level: Middle
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check gas concentration		
Brief Description:	This use case describes how user can observe room temperature and humidity and how the system response if threshold exceeded		
Trigger:	User wanted to observe the room temperature and humidity		
Type:	External		
Relationships:			
	Association:	User	
	Include:		
	Extend:	Fan Activated	
	Generalization:		
Normal Flow of Events:	1. User click the play button 2. The dht11 sensor will detect temperature and humidity 3. The system will display the result 4. The system will react if threshold exceeded 5. The system will continue to loop and read the reading from sensor		
SubFlows:	Not applicable		
Alternate/Exception Flows:	4a. The system will turn on the fan if limit exceeded		

Table 3.7 Use Case Description for Check Light Intensity

Use Case Name:	Check Light intensity	ID: 10	Importance Level: Middle
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check light intensity		
Brief Description:	This use case describes how user can observe indoor light intensity if threshold exceeded		
Trigger:	User wanted to observe the indoor light intensity		
Type:	External		
Relationships:			
	Association:	User	
	Include:		
	Extend:	Led Activated	
	Generalization:		
Normal Flow of Events:	1. User click the play button 2. The LDR will detect the light intensity 3. The system will react when indoor is dimmer 4. The system will activate the LED		
SubFlows:	Not applicable		
Alternate/Exception Flows:	3a. The system will turn off the led if indoor is brighter		

Table 3.8 Use Case Description for Check Connection Establishment

Use Case Name:	Check Connection Establishment	ID: 106	Importance Level: Middle
Primary Actor:	User	Use Case Type: Detail, Essential	
Stakeholders and Interests:	User – want to check connection establishment for ESP WROOM 32 and Blynk app with the server and internet		
Brief Description:	This use case describes how user can observe connection establishment on ESP WROOM 32, Blynk app and server		

Trigger:	User wanted to observe if the Blynk app and ESP WROOM 32 is connected to internet and server	
Type:	External	
Relationships:		
	Association:	User
	Include:	
	Extend:	
	Generalization:	
Normal Flow of Events:	1. User click the play button 2. The system will show the connection establishment	
SubFlows:	Not applicable	
Alternate/Exception Flows:	2a. The system will show "system disconnected" if not connected to internet and server	

3.1.3 Activity Diagram

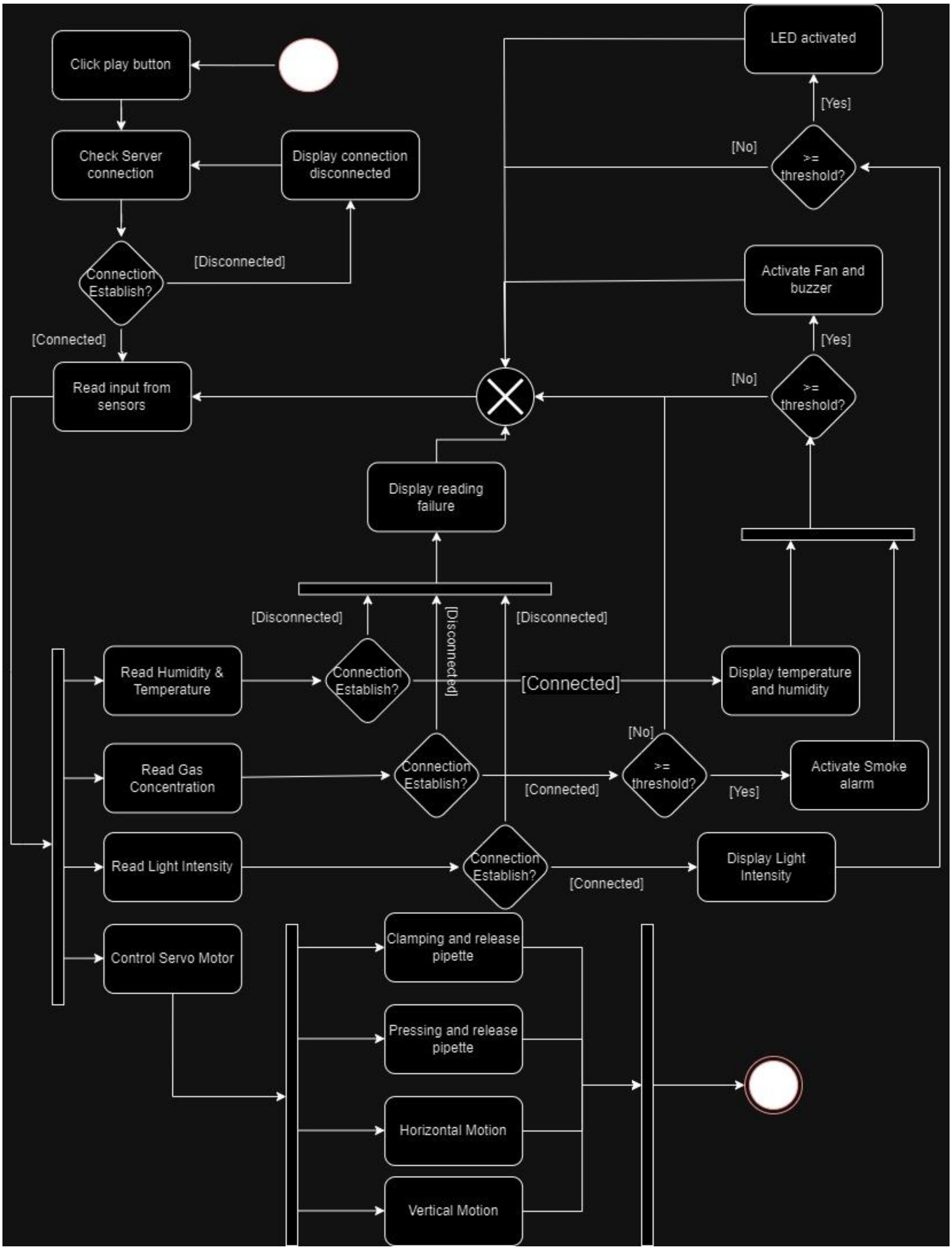


Figure 3.3 Activity Diagram

The flow of events will be explained in Chapter 4.4, system components interaction operation.

Chapter 4

System Design

4.1 System Block Diagram

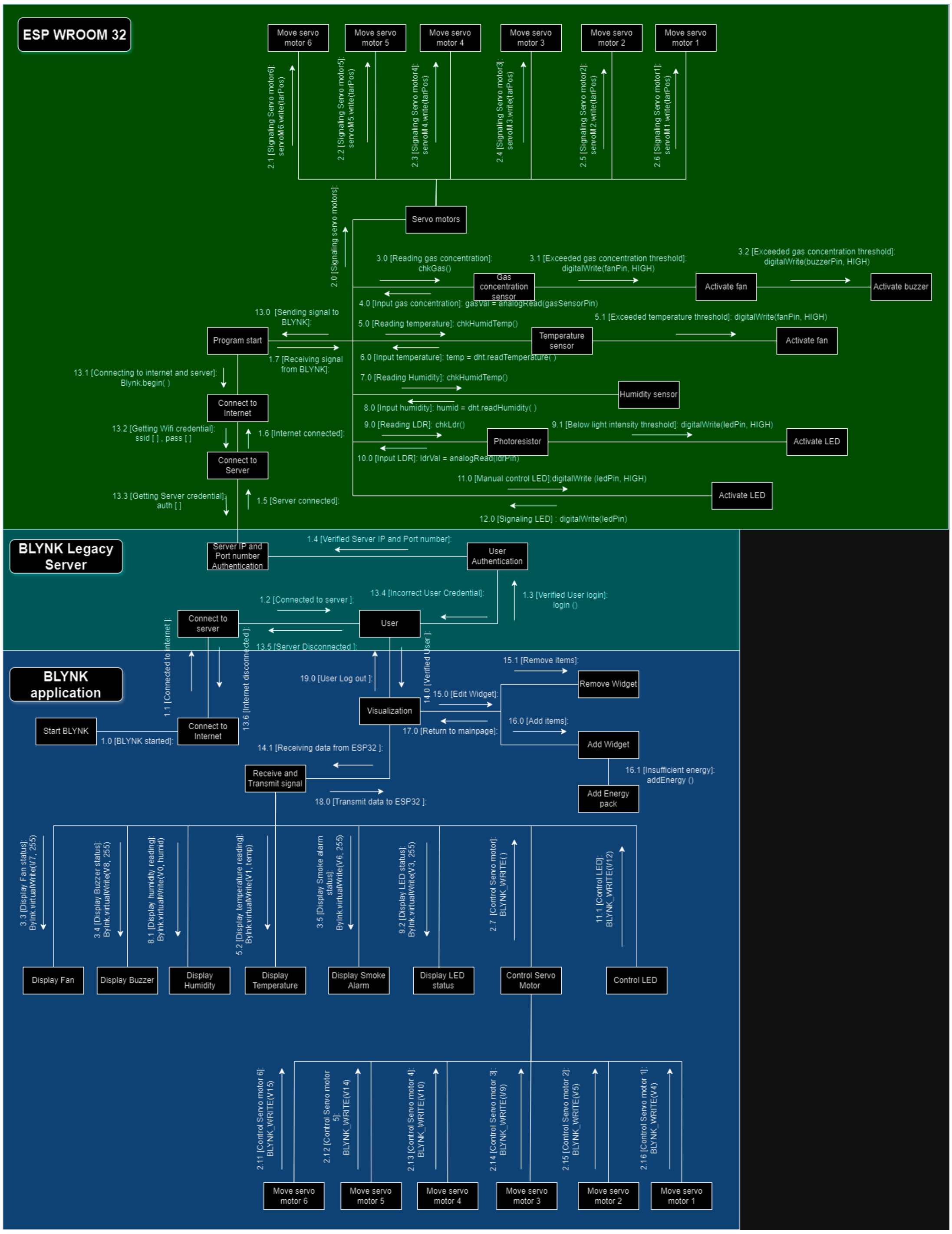


Figure 4.1 System Block Diagram

4.2 System Components Specifications

4.2.1: Robotic and Servo motor specification

Table 4.1 Specifications of Robotic Arm

Description	Features	Conditions
Rotation angle (DOF)	180˚	
Radius of gyration	355mm	
Height	460cm	Holder closed
Clamp Max Opening	55mm	
Holder of widest distance	98mm	
Items	**Amount**	
Multi-functional servo bracket	5	
Long U-type servo bracket	4	
L-type servo bracket	1	
U-type robot waist bracket	3	
Miniature Ball Radial Bearing	4	
Aluminium Clamp Claw	1	
Screw and screw cap set	1	

Table 4.2 Specifications of Servo Motor

Description	Features	
Operating Voltage	4.8V to 6V	
Operating Speed	4.8V 6V	0.17 sec/60° 0.14 sec/60°
Stall Torque	4.8V 6V	9.4 kg-cm 11 kg-cm
Operating Temperature	-30°C to +60°C	
Dead Band Width	5 microseconds	
Operating Angle	360 degrees continuously	
Dimensions	40.7mm x 19.7mm x 42.9mm	

Weight	55 grams	
Gear Type	metal gears	
Control Interface	3-wire interface for control	power, ground, and signal
Compatibility	Generating PWM signals	

4.2.2: Specification of all Sensor

Table 4.3 Specification of dht22 Sensor

Description	Features
Operating Voltage	3.3V - 6V DC
Temperature Measurement Range	-40°C to 80°C
Temperature Accuracy	±0.5°C
Humidity Measurement Range	0% to 100% (Relative Humidity)
Humidity Accuracy	±2% (Relative Humidity)
Resolution	0.1°C for temperature, 1% RH for humidity
Output Signal	Digital signal
Sampling Rate	2s
Dimensions	15.1mm x 25mm x 7.7mm
Operating Humidity	0% to 99%

Table 4.4 Specification of MQ-2 Sensor

Description	Features
Operating Voltage	5V DC
Power Consumption	Less than 800mW
Heater Voltage	5V ±0.2V
Heater Resistance	33Ω ±5%

Heating Current	Less than 180mA
Operating Temperature	-10°C to 50°C
Humidity Range	20% to 95%(Relative Humidity)
Gas Detection Range	LPG, i-butane, propane, methane, alcohol, hydrogen, smoke
Response Time	<10 seconds
Recovery Time	<30 seconds

Table 4.5 Specification of LDR-5528

Description	Features
Operating Voltage	3V to 5V DC
Photo Resistance	8 ~ 20 KΩ
Dark Resistance	1 MΩ
Pmax	100mW
Operating Temperature	-30°C to 70°C

Table 4.6 Specification of 2N2222 transistor

Description	Features
Type	NPN
Maximum Collector-Base Voltage	75V
Maximum Collector-Emitter Voltage	40V
Maximum Emitter-Base Voltage	6V
Maximum Collector Current	800mA
Transition Frequency (ft)	250MHz

Table 4.7 Specification of 1N5819 Schottky Diode

Description	Feature
Average forward current	1A
Forward Surge Current	25A
Forward voltage drop	600mV at 1A
Peak reverse voltage	40V
RMS Reverse Voltage	28V

Table 4.8 Specification of TXS0108E

Description	Feature
Number of Channels	8 bidirectional channels
Voltage Translation	1.2V and 3.6V on the A-side 1.65V to 5.5V on the B-side
Operating Voltage Range	1.2V to 3.6V on the A-side 1.65V to 5.5V on the B-side

Table 4.9 Specification of 220 ohms resistor

Description	Feature
Resistance	220ohms
Power Rating	0.25 watts
Tolerance	±5%
Temperature Coefficient	200-500 ppm/°C
Maximum Voltage	200 volts

4.2.3: Microcontroller used for controlling the robotic arm

Table 4.10 Specification of ESP WROOM 32

Description	Feature	Conditions
Microcontroller	ESP32-D0WDQ6 chip	Run at 240 MHz
Wireless Connectivity	Wi-Fi	802.11 b/g/n/e/i (2.4 GHz) and 802.11 n/ac (5 GHz)
	Bluetooth	Bluetooth v4.2 and BLE (Bluetooth Low Energy)
Memory	RAM	520 KB SRAM
	Flash	4 MB SPI flash memory
Peripherals	34 GPIO pins	UART, SPI, I2C, I2S, PWM

	Analog-to-Digital Converter	12-bit SAR ADC with up to 18 channels
	Touch Sensor	Capacitive touch sensor
Interfaces	UART, SPI, I2C, I2S, PWM, SDIO, and CAN interfaces	
	USB 2.0 OTG	
Operating Voltage	3.3V DC	
Operating Temperature	-40°C to 125°C	
Power Consumption	5 µA	
Security	AES, SHA-2, RSA, ECC	
Programming	UART bootloader, OTA (Over-The-Air) updates	
Dimensions	18mm x 25.5mm	

4.2.4: Power Supply

The power supply unit responsible for providing power to the robotic arm system.

Table 4.11 Specification of MB-102 Breadboard

Description	Features
Input Voltage	6.5V – 12V
DC barrel connector	2.1mm
Output Voltage	3.3V, 5V
Maximum Output Current	<700 ma
Size	5.3cm x 3.5cm

4.2.5: Communication Module

To establish connectivity between the robotic arm system and other networks or devices, a communication module is utilised.

Table 4.12 Specification of control software and User Interface

Control software	Version	Description
BLYNK legacy server	Custom server	IP server: 128.199.159.242 Port: 9443
BLYNK application	2.27.34	
Arduino IDE	2.2.1	

4.2.6: Connectivity

Table 4.13 Connection for tablet and ESP WROOM 32

Device	Connection
BLYNK application (tablet)	Home Wi-Fi
ESP WROOM 32	Mobile hotspot
Code upload to ESP WROOM 32	USB 3.0

4.3 Circuits and Components Design

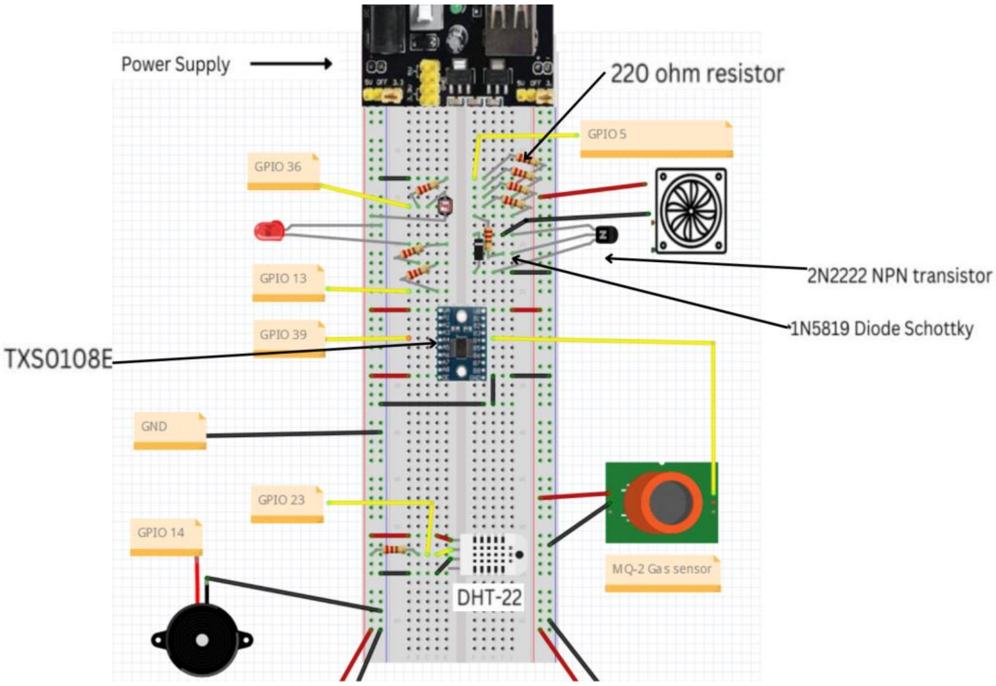


Figure 4.2 Breadboard diagram for all components except servo motor

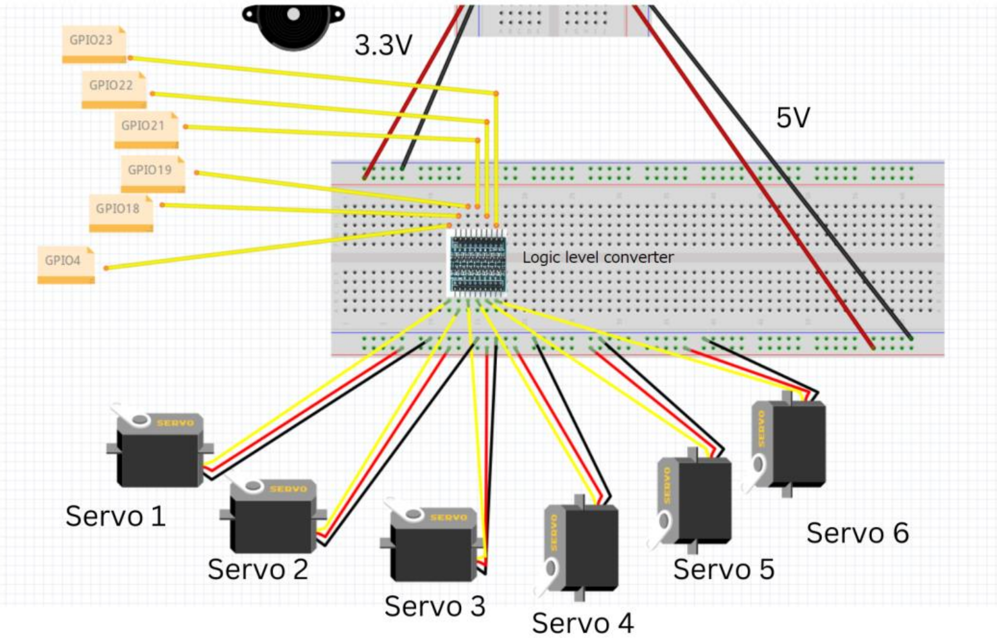


Figure 4.3 Breadboard diagram for servo motor only

4.4 System Components Interaction Operations

4.4.1 BLYNK legacy_2.27.34 installation

The BLYNK application with the version of BLYNK legacy_2.27.34 was installed on the Samsung tablet so that users could interact with ESP WROOM 32. The BLYNK legacy's package installer was installed. Then, users must log in to the BLYNK application and change the server settings to custom. Next, users must enter the IP server "128.199.159.242" and port number "8080".

4.4.2 Connectivity of ESP WROOM 32

The ESP WROOM 32 could connect to the internet by declaring WiFi credential, which is ssid [ ] and pass [ ] for the network and could connect to the server by declaring auth [ ] = "lg0cavZ9BAXEHQjSVduMYhMbjDcWpnSF". The ESP WROOM 32 will connect to the server and internet when Blynk.begin(auth, ssid, pass, IPAddress(128,199,159,242), 8080) was called. With all these settings, the ESP WROOM 32 was successfully connected to the internet and the server. On the other hand, users can connect to ESP WROOM 32 using the BLYNK application that was installed on their device and start to receive and transmit signal interchange.

4.4.3 Power supply and Connectivity

The MB-102 steps down 12VDC into 3.3VDC and 5VDC, supplying power to the breadboard. A 12VDC 1A power supply powered it. The components are separated into two parts that require different amounts of voltage. The components that require the 3.3 VDC were LDR, LED, buzzer and dht-22 sensor. Those components connected to 5 VDC were MQ-2 gas sensors, fans, and six servo motors.

4.4.4 Fan

The fan was activated when the temperature exceeded 37˚C and 400 ppm. The collector pin of the 2N2222 transistor was connected to the cathode of the 1N5819 Diode and then connected to the negative terminal of the fan. The base will connect to the five 110-ohms resistors and the GPIO5 of ESP WROOM 32. Meanwhile, the emitter and anode of the diode were connected to the ground. Lastly, the positive terminal was connected to the positive terminal of the power supply. The transistor is used like a switch to control the fan and the transistor conducts when the GPIO pin is HIGH and will allow current flow to the fan, and the fan stops when the GPIO pin is LOW. The diode protects the circuit when the fan is turned off to prevent voltage spikes and acts as circuit protection.

4.4.5 Light Dependent Resistor and LED

LED was activated by LDR or by the user manual switch. One leg of LDR was connected to a 3.3VDC positive terminal of the power supply. Another leg of LDR was connected to a GPIO 36 connected to a 10k ohm resistor. Another leg of the resistor was attached to the ground. The anode of the LED was connected to two 100 ohm resistors before being connected to the GPIO 13. The cathode was connected to the ground. LED can be activated manually by pressing the Blynk application button with the Virtual pin of V12.

4.4.6 DHT22 sensor

The positive terminal of the dht22 sensor was connected to a positive terminal of 3.3VDC. The signal GPIO 12 was connected to a 10k ohm resistor before being connected to the positive terminal power supply. The negative terminal was connected to the ground.

4.4.7 MQ-2 Sensor

The MQ-2 gas sensor will connect its positive and negative cables to the 5V power supply, and the analogue line will connect to the logic level converter, which converts the 5V signal to a 3.3V signal, which sends a signal to the GPIO 39.

4.4.8 Robotic Arm development and Wiring

The first degree of freedom, or the base servo motor, allows the robotic arm to rotate horizontally from 0 to 180 degrees. The second degree of freedom, or a shoulder servo motor, allows the robotic arm to move forward and backward. The third degree of freedom did not require the installation of a servo motor as it is unnecessary. The fourth degree of freedom installed servo motor, which acts as a wrist servo motor that allows bending like a wrist. The fifth degree of freedom acted like a wrist turning, and the sixth degree of freedom acted like a clamp to hold the pipette. Lastly, the pipette pressing was performed using a 5ml syringe attached to the top of the clamp and connected using a plastic hose. Another end of the hose was connected to a 5ml syringe, which was attached to an M6x12cm bolt using an epoxy hardener. An epoxy hardener connected another end of the bolt to the sixth servo motor. The brown cable from the servo motors was connected to the ground of the 5VDC power supply. In contrast, the red cable was connected to the positive terminal, and the yellow signal cable was connected to the logic level converter to receive a signal from 3.3V from ESP WROOM 32 to convert to a 5V servo motor. Besides, additional effort was made by adding two 0.3cm X 8.2cm and one 0.7cm X 6.4 cm spring from the second degree of freedom servo motor until the fifth degree of freedom servo motor. This step is very crucial, as when the clamp is holding the pipette, the extra weight will push against the fourth degree of freedom servo motor and cause the whole clamp to fall due to extra weight. The three springs act like a backbone to the robotic arm and hold the fifth servo motor when clamping the pipette. The GPIO is displayed in the table below:

Table 4.14 Respective GPIO pin with logic level converter and servo motors

ESP WROOM 32	Logic Level Converter		
GPIO	3.3V	5V	
Negative terminal	GND	GND	Negative terminal
Positive terminal	VCCB	VCCA	Positive terminal
4	B2	A2	Servo 1
18	B3	A3	Servo 2
19	B4	A4	Servo 3
21	B5	A5	Servo 4
22	B6	A6	Servo 5
23	B7	A7	Servo 6

4.4.9 BLYNK mobile application user interface

The servo motors were controlled by adding a joystick widget in the BLYNK application. Each servo motor was assigned a virtual pin that must be similar to the function that handles the incoming data from the Blynk app. The virtual PIN in the ESP WROOM 32 must also be identical to the one used to communicate between the app and the hardware. The user can control the robotic arm movement by simply controlling the joystick on the Blynk application. The respective virtual pin that controls the servo motor is shown below:

Table 4.15 Virtual pin that used to communicate between Blynk app and servo motors

Virtual pin	Number of servo motor
V4	Servo motor 1
V5	Servo motor 2
V9	Servo motor 3
V10	Servo motor 4
V14	Servo motor 5
V15	Servo motor 6

The user must add a temperature and humidity gauge in the Blynk app and assign the correct virtual pin to receive readings from the dht22 sensor. Next, an LED, smoke alarm, buzzer, and fan display were added to show the user how ESP WROOM 32 responds to display the result on the Blynk app when it detects that the results obtained from the sensors exceed the designed limit. The table below shows the threshold for each sensor and their respective virtual pin:

Table 4.16 Virtual pin that communicate with sensors and alarming if threshold exceeded

Gauge	**Virtual pin**	**Condition**
Temperature Gauge	V1	Received input from dht22
Humidity Gauge	V0	Received input from dht22
LED	V3	LDR > 200
Smoke Alarm	V6	Gas concentration > 400
Buzzer	V8	Gas concentration > 400 and Temperature > 37˚C
Fan	V7	Gas concentration > 400 and Temperature > 37˚C

The threshold for temperature, light intensity value, and gas concentration were initialized in the Arduino IDE. The code was then uploaded to ESP WROOM 32 so that when the sensor detected any reading exceeding the threshold, the output would be activated and displayed on the user interface to show the user. The parts per million (ppm) set for MQ-2 at 400ppm was to detect Carbon Monoxide which ranges from around 100ppm to 1,000ppm, methane with a detection range of 300ppm to 10,000ppm, propane which is between 100ppm to 10,000ppm and smoke around 200ppm to 500ppm. These are the common gases that are used in common laboratories for experiments. The LDR threshold was set to 200 lux due to general indoor lighting the illumination level inside a laboratory ranges around 300 lux to 500 lux.

Chapter 5

System Implementation

5.1 Hardware Setup

In this project, several pieces of hardware were involved in order to detect humidity, temperature, gas ppm, and brightness, which acted as signal input. Besides, there were a few pieces of output hardware used to respond to the change in the configuration detected by the sensors. Table 5.1 lists all the hardware components required to be used in this project. Chapter 4.4 System component interaction operations explained the hardware setup for all the components. Meanwhile, Chapter 4.3 visualized the circuits and components design of the whole project. The modification of the servo motor into a hydraulic actuator will only be discussed in this chapter. The head of the M6 X 12cm bolt was attached to the servo motor using an epoxy glue hardener, and the nut was attached to the syringe using an epoxy glue hardener. The syringe will push the water from one side to another when the servo motor spins clockwise, and the thread will start turning on the fixed nut, and vice versa. Figure 5.1 displays the complete prototype for the modification. Users can insert the pipette into a pipette holder and put it on top of the clamp before performing the pipetting task.

Table 5.1 List of hardware components

Items	Description
Microprocessor	ESP WROOM 32
Temperature and Humidity Sensor	DHT-22
Gas Sensor	MQ-2 gas detector
8 Channel Logic Level Converter	TXS0108E (2 pieces)
Buzzer	3.3V supported
Power supply (5V and 3.3V)	MB-102 Breadboard Power supply
Servo Motor	MG996r (6 pieces)
Robotic Arm	ROT3U 6DOF aluminium robotic arm
Breadboard	Solderless 830 holes Breadboard
LDR	5528 GL 5528 5mm 10k-20k Photosensor

LED	1.8-2.4 VDC LED
Resistor	220ohm resistor
Transistor	2N2222 NPN transistor
Diode	1N5819 Diode Schottky
Jumper Wires	-
Power supplies	12v, 1A
5V Fan	-
Syringe	Medical syringe (2 pieces)
Hose	1M (0.5cm)
Spring	0.3 cm X 8.2cm (2 pieces) 0.7cm X 6.4cm (1 piece)

Figure 5.1 Modification of servo motor into hydraulic actuator

5.2 Software Setup

Before starting to develop the code, there are three software needed to be installed and downloaded in the laptop:

- Arduino IDE 2.2.1
- BLYNK Legacy
- Fritzing

5.2.1 Fritzing

Fritzing will be required to be installed in the computer in order to draw the breadboard diagram. This step is important before starting to wire the components and connect the hardware.

5.2.2 Arduino IDE 2.2.1

The Arduino Integrated Development Environment (IDE) is used to program Arduino or other microcontrollers like ESP WROOM 32. It gives code writers, compilers, and Arduino board uploaders an easy-to-use interface. Arduino Sketch allows users to write code with the text editor that comes with the IDE. The setup() and loop() procedures are usually the two primary functions of a sketch. When the Arduino first boots up, the setup() method will be called once, whereas the loop() function runs continually. The Arduino IDE makes library management simple. By enabling data transmission and reception over the USB connection between the computer and the Arduino board, the Serial Monitor facilitates troubleshooting. All the code is translated into machine language that the Arduino can comprehend by the IDE.

5.2.3 BLYNK application

A flexible and easy-to-use smartphone application for Internet of Things (IoT) projects is called BLYNK. Its main purpose is to give developers a platform to make unique mobile apps that let customers manage and keep an eye on the hardware devices they have linked. BLYNK makes advantage of the notion of virtual pins to make communication easier between the hardware device and the mobile application. Virtual pins are a useful tool that users may utilise to convey data from an app to a device or the other way around. The cloud services offered by BLYNK serve as a link between the physical device and the mobile application. Because of this, users may access and operate their devices from a distance without requiring complicated networking setups. Real-time monitoring of sensor data and other parameters from linked devices is made possible by BLYNK. Users may get real-time information from their IoT projects on temperature, humidity, sensor readings, and other data streams.

5.3 Setting and Configuration

After the code is successfully built, connect a USB connection to upload the code to the ESP WROOM 32. There are a few libraries that need to be installed in the Arduino IDE in order to connect to the hardware. The libraries and board support package are as follows:

- DHT sensor library by Adafruit
- https://dl.espressif.com/dl/package_esp32_index.json (ESP 32 board support package)
- BLYNK legacy 0.6.1 by Volodymyr Shymanskyy
- Wifi.h
- ESP32Servo.h

The authentication token is a unique identifier that is used to connect ESP WROOM 32 to the BLYNK cloud server. The ESP WROOM 32 are required to connect to the Wi-Fi connection by inserting Wi-Fi credentials. The IP address 128.199.159.242 is the IP address of the BLYNK cloud server to which ESP WROOM 32 is connected. This is the server where this project is hosted. Port 8080 is used for the web server.

The user needs to log in to the Blynk application with a registered user and insert the IP address of the server and port number. Then, the user can add any widget to receive any reading from ESP WROOM 32. In this project, two gauge displays were used, one for temperature and another for humidity. Four LED indications were used for LED, smoke alarm, buzzer and fan. One button widget was used for manual control the LED. Six joysticks widget were used for controlling six servo motors.

5.4 System Operation

Figure 5.2, Figure 5.3, and Figure 5.4 show the user interface display for Blynk and Arduino IDE when ESP WROOM 32 executes the code and responds to the input.

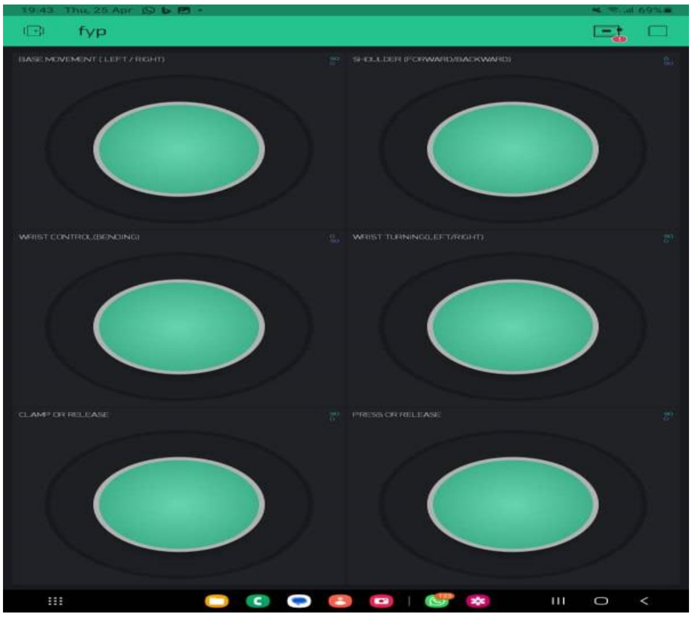


Figure 5.2 Six joystick to control six servo motor in Blynk application

Figure 5.3 Display of Blynk's system operation

```
19:49:39.960 ->     ___  __          __
19:49:39.960 ->    / _ )/ /_ _____  / /__
19:49:39.980 ->   / _  / / // / _ \/  '_/
19:49:40.012 ->  /____/_/\_, /_//_/_/\_\
19:49:40.044 ->         /___/ v0.6.1 on ESP32
19:49:40.076 ->
19:49:40.076 -> [21302] Connecting to 128.199.159.242
19:49:40.323 -> [21617] Ready (ping: 202ms).
19:49:42.394 -> LDR value: 1246
19:49:44.397 -> LDR value: 1325
19:49:46.399 -> LDR value: 1226
19:49:48.403 -> LDR value: 1219
19:49:50.396 -> Gas ppm: 1153
19:49:50.396 -> Smoke detected!
19:49:50.611 -> LDR value: 1339
19:49:52.382 -> LDR value: 1278
19:49:52.382 -> Humidity: 80.90 %
19:49:52.414 -> Temperature: 30 °C      LDR value: 1244
```

Figure 5.4 System operation output in Arduino IDE

5.5 Implementation Issues and Challenges

The robotic arm cannot be adjusted to record and replay the recorded movement accurately and automatically. Although it can respond to user input, accuracy is still a problem that needs to be solved. The ESP WROOM 32 microprocessor was facing heat when connected to all the servo motors and another sensor together, and an additional microprocessor might be considered to share the load. The robotic arms are having delayed response issues due to slow internet.

5.6 Concluding Remark

To sum up, the system's implementation required a sophisticated hardware configuration that included sensors to identify different ambient elements and output devices to react to these inputs. One important component of the project was the conversion of the servo motor into a hydraulic actuator, which improved its functionality. The Arduino IDE and Fritzing were two important software tools used in the system's design and programming. The incorporation of the BLYNK software offered an intuitive interface for remotely monitoring and managing the devices.

Nonetheless, a number of difficulties and problems with implementation were found. These included challenges in getting the robotic arm to operate precisely, problems with the ESP WROOM 32 microprocessor's heat management, and sluggish response times because of slow internet. To overcome these obstacles, further optimisation will be needed, and additional hardware might need to be added in order to reduce the workload.

Notwithstanding these difficulties, the system shows promise in terms of environmental monitoring and control, and future iterations may see even more development and enhancement.

Chapter 6

System Evaluation And Discussion

6.1 System Testing and Performance Metrics

In this chapter, there were several tests done on the project. This includes functional testing, integration testing, usability testing, and performance testing.

The functional testing for the robotic arm was done to verify that the robotic arm can perform certain movements and is able to pipette and transfer water from one beaker to another beaker and release the contents inside the pipette tip into the beaker. Meanwhile, the functional testing for ESP WROOM 32 was done on data collection from the sensors, processed it, and reacted to the results.

Integration Testing was done to ensure seamless integration between sensors, server, ESP WROOM 32, output components, Blynk application, and robotic arm.
Usability testing was used to check the Blynk application's interaction with the robotic arm, server, and ESP WROOM 32.

Performance testing on the response time for executing commands of the ESP WROOM 32 when input from sensors is detected.

The performance metrics for the sensors, ESP WROOM 32, and the robotic arm are stated in the table below:

Table 6.1 Performance metric for robotic arm

Servo motor	Performance Metric	Degree
Base servo motor	Rotate horizontally	0˚ - 180˚
Shoulder servo motor	Forward and Backward	0˚ - 180˚
Elbow motor	-	-
Wrist servo motor	Upward and downward	0˚ - 180˚
Wrist tuning servo motor	Rotate vertically	0˚ - 180˚
Gripper servo motor	Clamp and release	-

Hydraulic actuator	Press and release	Clockwise or anti-clockwise

Table 6.2 Performance metric for ESP WROOM 32, sensors and Blynk application

	Performance Metric	Response	Blynk
Dht22 sensor	Apply heat which more than 37 ˚C	Buzzer and Fan will be activated	Buzzer and Fan will be on, temperature reading will be displayed
MQ-2 sensor	Apply smoke which more than 400ppm	Buzzer and fan will be activated	Buzzer, smoke alarm and Fan will be on
LDR	Lower indoor light intensity to less than 200 lux	LED will be activated	LED display will be on

6.2 Testing Setup and Result

6.2.1 Testing setup for robotic arm and result

The robotic arm was tested for performing the testing performance metric before it was tested for pipetting. The robotic arm was successfully able to perform all the required testing. Then, the testing on pipetting was executed in the Faculty of Science in UTAR. Firstly, the 10-microliter Thermo Fisher Scientific pipette was attached to a pipette holder, pipette and the pipette holder was put on top of the clamp. Then, the pipette was attached with a pipette tip to hold the water after the suction of the pipette. After all the steps were completed. The robotic arm was ready to do pipetting.

The robotic arm was required to pipette the water from one beaker to another beaker and release the water on the other beaker. Figure 6.2 below shows the robotic arm was pressed on the plunger of the pipette to remove the air inside the pipette tip and release the plunger to perform suction. The step on performing pressing was done when the sixth servo motor rotates clockwise and the release step by rotating anti-clockwise. The sixth servo motor was acting like a hydraulic actuator. After the required water amount was achieved, the wrist servo motor was moved upward, followed by the shoulder servo motor moved backward to remove the pipette from the first beaker. Next, the base servo motor rotated to 30 degrees to the left, and shoulder servo motor moved forward and the wrist servo motor moved downward to insert the pipette into the second beaker. The sixth servo motor turned clockwise to push pressure from one syringe to another syringe, which was pushed against the plunger of the pipette to release the water from the pipette tip. The testing was completed.

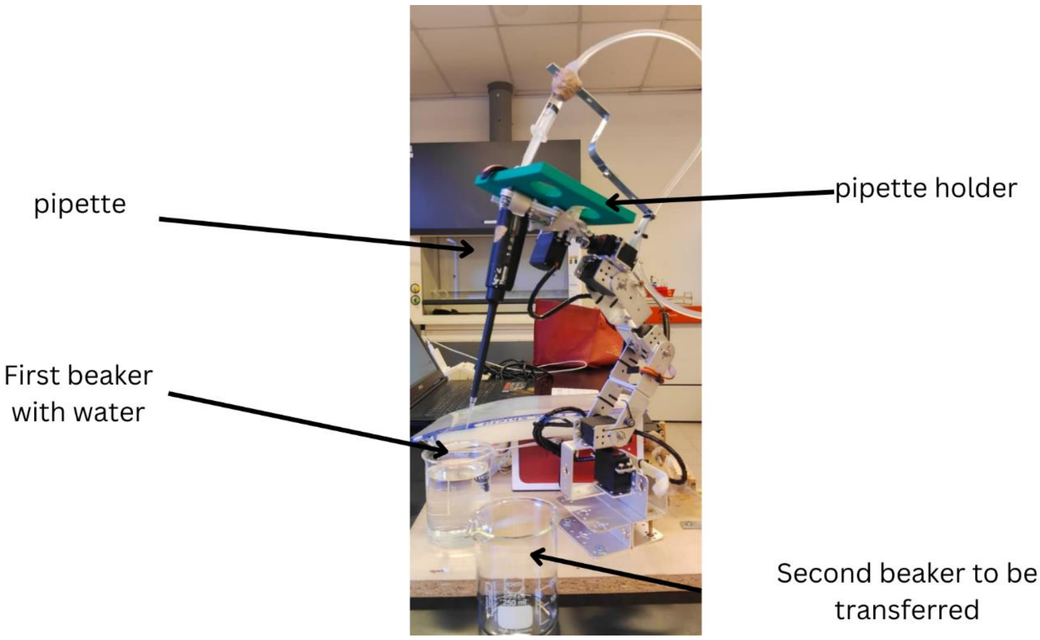


Figure 6.1 Setting up pipette for testing

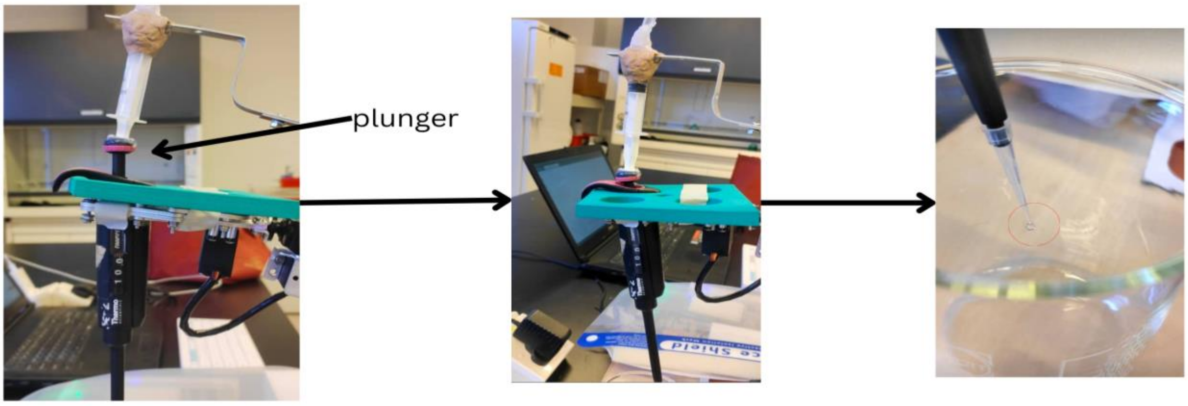


Figure 6.2 The plunger pressed and released by robotic arm to remove water inside pipette

6.6.2 Testing setup for ESP WROOM 32 and result

```
23:33:43.411 -> [23183] Connected to WiFi
23:33:43.411 -> [23183] IP: 192.168.137.49
23:33:43.425 -> [23183]
23:33:43.428 ->     ___  __          __
23:33:43.428 ->    / _ )/ /_ _____  / /__
23:33:43.506 ->   / _  / / // / _ \/  '_/
23:33:43.506 ->  /____/_/\_, /_//_/_/\_\
23:33:43.535 ->         /___/ v0.6.1 on ESP32
23:33:43.567 ->
23:33:43.567 -> [23305] Connecting to 128.199.159.242
23:33:44.011 -> [23845] Ready (ping: 203ms).
```

Figure 6.3 Testing on the internet connectivity and server

```
11:41:33.336 -> Humidity: 75.40 %
11:41:33.336 -> Temperature: 30 °C      LDR value: 1362
11:41:37.307 -> LDR value: 1214
11:41:39.332 -> Gas ppm: 387
```

Figure 6.4 Normal gas concentration, temperature, humidity and light intensity value in Arduino IDE

Figure 6.5 Output displayed on the Blynk application for standard condition

Figure 6.3 shows the ESP WROOM 32 was successfully connected to the internet and server. Figure 6.4 shows the reading obtained from the Arduino IDE, reflected from the ESP WROOM 32. Figure 6.5 shows the result of temperature, humidity reading, LED, smoke alarm, buzzer, and fan status when the test was not executed.

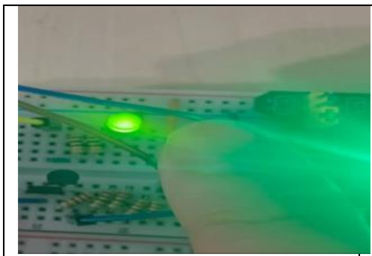
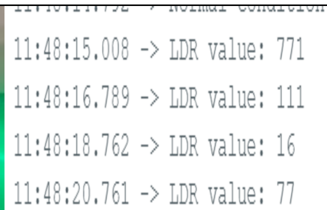
<table>
<tr><td></td><td>
</td><td>
</td></tr>
<tr><td>Figure 6.6 LDR covered manually</td><td>Figure 6.7 LDR value in Arduino IDE</td><td>Figure 6.8 LED widget in Blynk application</td></tr>
</table>

Figure 6.6 to Figure 6.8 showed the testing on LDR in response to dimmer light intensity. The result in Figure 6.7 proves the reading before and after LDR was covered. The LDR value showed a significantly reduced, and at the same time, the LED was triggered immediately. This result is also displayed in the Blynk application.

Gas ppm: 997 Smoke detected! Unable to receive data from DHT Sensor!	 	
Figure 6.9 Arduino IDE reading on gas concentration	Figure 6.10 Blynk displayed the result when smoke is detected	Figure 6.11 Fan activated when smoke is detected

Figure 6.9 to Figure 6.11 shows the testing and result for the MQ-2 gas sensor. In this setting, dht22 was removed from the breadboard as it might interfere with the buzzer and fan result, as high temperature will trigger both. Figure 6.10 shows when gas concentration exceeded and the smoke alarm, buzzer, and fan. Figure 6.11 was activated when smoke was detected.

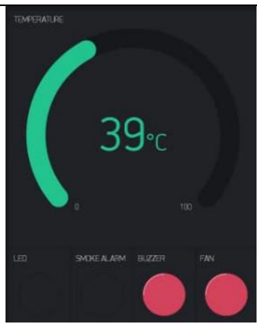
Temperature: 39 °C LDR LDR value: 1408 Gas ppm: 0 High temperature detected!	 	
Figure 6.12 Result shows in the Arduino IDE on testing dht22 at higher temperature	Figure 6.13 Temperature widget shows temperature and responses	Figure 6.14 Fan activated when temperature is more than 37˚C

Figure 6.12 to Figure 6.14 showed the result tested on dht22 using high temperatures. This testing was done by removing the MQ-2 sensor, as it might interfere with the buzzer and fan activation. Figure 6.13 displays the result of temperature in the temperature widget, which had exceeded the threshold. The system then activates the buzzer and fan.

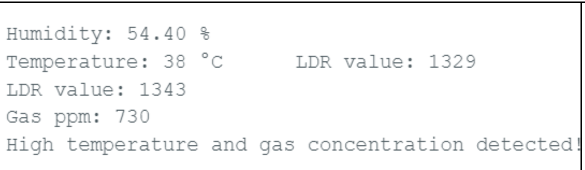


Figure 6.15 Test on both high temperature and gas concentration

Figure 6.16 Result shows high temperature and high gas concentration

Figure 6.15 to Figure 6.16 were tested on high temperature and gas concentration. In this condition, the system had triggered the smoke alarm, buzzer, and fan.

6.3 Project Challenges

The challenges of this project mainly were on the hardware part. The most challenging part was converting a servo motor into a hydraulic actuator. This conversion was due to the inconsistent voltage inconsistency of the hydraulic actuator sold in the market with the power supply in this project. Also, there was an extra servo motor purchased. Secondly, the welding of the electronic components was the second most challenging.

6.4 Objectives Evaluation

The first objective could not be achieved by adding several grippers to hold the pipette. However, the increment of grippers was successfully replaced by modification on the servo motor to become a hydraulic actuator to press and release the plunger of the pipette. Besides, the pipette holder solved the problem gripper of the robotic arm to keep holding the pipette as the pipette holder can fit various sizes of pipette. The pipette holder just leans on top of the gripper.

The second objective was successfully achieved. The external power source provided all servo motors, fans, and MQ-2 sensors with 5V. At the same time, it provided 3.3V to dht22, LDR, LED, and buzzer. This successfully reduced the dependence on the power source provided by ESP WROOM 32. The external power reduced the burden of ESP WROOM 32 without burning out due to the high workload.

The third objective was managed to achieve. The fan was activated successfully when the temperature was higher than 37˚C. The lighting of the room is controlled by a Light Dependent Resistor (LDR). The medical personnel do not need to switch on the light manually. However, it can be controlled manually using Blynk application if required.

The fourth objective was achieved as a user-friendly Blynk application was used for monitoring and enabling real-time visualization and able to control the robotic arm to do pipetting without physical contact.

6.5 Concluding Remark

To sum up, extensive testing of the system's functionality, integration, usability, and performance was conducted to guarantee its dependability and efficacy. Before undertaking pipetting tests, which were carried out successfully at the Faculty of Science in UTAR, the robotic arm cleared performance metrics. The main hardware challenges were welding electrical components and transforming servo motors into hydraulic actuators because of inconsistent voltage. Despite these difficulties, the goals were assessed and, for the most part, met using creative solutions. A redesigned servo motor and pipette holder fixed the gripper's problems, and external power sources reduced the strain on ESP WROOM 32. Automation of lighting and temperature management was one of the goal accomplishments, made possible by the user-friendly Blynk application. All things considered, the project proved to be successfully implemented and accomplished its goals, highlighting its potential for practical uses.

Chapter 7

Conclusion and Recommendation

7.1 Conclusion

In conclusion, contactless laboratory work is successfully achieved by the integration of ESP WROOM 32, BLYNK application, sensors, and output components. All of the objectives as proposed have successfully been accomplished with the addition of the hydraulic actuator. All the sensors were tested and provided an accurate reading to the ESP WROOM 32 based on the result. The system also responded correctly and efficiently by activating the respective output. The robotic arm was able to perform the pipetting task just like healthcare workers. However, the accuracy of servo motor movement requires improvement. Laboratory and healthcare workers will be prepared for the upcoming worldwide virus pandemic with the integrated ESP WROOM 32 model.

7.2 Recommendation

The design for the hydraulic pump could have been more reliable as it only lasted for a short time, although it was cheap to build. The recommendation to improve the reliability and accuracy of the robotic arm is to replace the servo motor with a pneumatic actuator. The shoulder servo motor, elbow servo motor, and wrist servo motor can be replaced by a pneumatic actuator. Pneumatic actuators can operate at a faster speed, which can improve the responsiveness of the robotic arm, which will provide faster movement and reduce overall operation time. Pneumatic actuators generally provide a higher level of torque and force, which allow the robotic arm to hold a heavier load. Pneumatic actuators can provide smooth motion profiles, which is very important to provide precise and more controlled movements of the robotic arm. Smooth motion is very important for tasks like pipetting that require accurate positioning. It is much easier to program a pneumatic system due to simpler control mechanisms compared to servo motors, and it can reduce the complexity of the project. Pneumatic actuators have fewer moving parts and reduce the possibility of wear and tear. Less complexity can provide lower maintenance and a prolonged lifespan for the robotic arm. Lastly, pneumatic systems are more suitable to work with a risk of electrical hazards, which can reduce the risk of injury or damage during a malfunction.

Overall, by offering faster speed, more power, smoother motion, easier control, less maintenance, and more safety, pneumatic actuators—a substitute for servo motors—can greatly increase the stability and accuracy of a robotic arm. Pneumatic actuators are an appealing option for applications demanding accurate and dependable robotic manipulation because of these advantages.

REFERENCES

[1] World Health Organization, "Health and Care Worker Deaths during COVID-19," WHO, Oct. 20, 2021. [Online]. Available: https://www.who.int/news/item/20-10-2021-health-and-care-worker-deaths-during-covid-19. [Accessed: Aug. 28, 2023].

[2] World Health Organization, "Health Workforce Financing: A Working Paper," WHO Health Workforce Series, no. 1, pp. 1-40, Aug. 2021. [Online]. Available: https://apps.who.int/iris/bitstream/handle/10665/345300/WHO-HWF WorkingPaper-2021.1-eng.pdf?sequence=1&isAllowed=y. [Accessed: Aug. 25, 2023].

[3] Assoc. Prof. Dr. Mohammed Abdulrazaq, "NOVEL COVID-19 DETECTION AND DIAGNOSIS SYSTEM USING IOT BASED SMART HELMET," ResearchGate, 2020. [Online]. Available: https://www.researchgate.net/profile/Assoc-Prof-Dr-Mohammed Abdulrazaq/publication/340264439_NOVEL_COVID-19_DETECTION_AND_DIAGNOSIS_SYSTEM_USING_IOT_BASED_SMART_HELMET/links/5e8dba3ba6fdcca789fe053e/NOVEL-COVID-19-DETECTION-AND-DIAGNOSIS-SYSTEM-USING-IOT-BASED-SMART-HELMET.pdf. [Accessed: Aug. 25, 2023].

[4] R. M. Satava, "Surgical robotics: the early chronicles: a personal historical perspective," Surg. Laparosc. Endosc. Percutaneous Tech., vol. 12, no. 1, pp. 6–16, 2002.

[5] Smith, J. A., Jivraj, J., Wong, R. & Yang, V. 30 years of neurosurgical robots: review and trends for manipulators and associated navigational systems. Ann. Biomed. Eng. 44, 836–846 (2016).

[6] Kwoh, Y. S., Hou, J., Jonckheere, E. A. & Hayati, S. A robot with improved absolute positioning accuracy for CT guided stereotactic brain surgery. IEEE Trans. Biomed. Eng. 35, 153–160 (1988).

[7] Ranev, D. & Teixeira, J. History of computer-assisted surgery. Surg. Clin. North Am. 100, 209–218 (2020).

[8] Marescaux, J. & Rubino, F. in Teleophthalmology (eds Yogesan, K. et al.) 261–265 (Springer, 2006).

[9] Marescaux, J. et al. Transatlantic robot-assisted telesurgery. Nature 413, 379–380 (2001).

[10] Troccaz, J., Dagnino, G. & Yang, G.-Z. Frontiers of medical robotics: from concept to systems to clinical translation. Annu. Rev. Biomed. Eng. 21, 193–218 (2019).

[11] Mrabet, E.L., Ait Moussa, H.: Smart classroom environment via IoT in basic and secondary education. Trans. Mach. Learn. Artif. Intell. 5, 4 (2017). https:// doi. org/ 10. 14738/ tmlai. 54. 3191.

[12] M. M. Asad, A. Naz, A. Shaikh, et al., "Investigating the impact of IoT-Based smart laboratories on students' academic performance in higher education," Univ Access Inf Soc, 2022. [Online]. Available: https://doi.org/10.1007/s10209-022-00944-1. [Accessed: Aug. 25, 2023].

[13] S. Chen, et al., "Facile synthesis of Mn3O4/C microspheres as anode materials for high-performance lithium-ion batteries," Nanoscale Research Letters, vol. 17, no. 1, p. 76, Feb 2022. Available: https://www.nature.com/articles/s41378-022-00400-3. DOI: 10.1186/s11671-022-03584-3.

[14] CDC, "Vaccine Storage and Handling Toolkit," Centers for Disease Control and Prevention, [Online]. Available: https://www.cdc.gov/vaccines/hcp/admin/storage/toolkit/storage-handling-toolkit.pdf . [Accessed: September 9, 2023].

[15] W. T. Wan Faizura, M.Z.M. Luqman, A. H. Luqman Hakim, O. Mohd Hafiz, and I. Zulkifli, "IoT based Robotic Arm for Student Online Learning," in Journal of Physics: Conference Series, vol. 2319, International Conference on Robotic Automation System 2021 (ICORAS 2021), Oct. 25-26, 2021. DOI: 10.1088/1742-6596/2319/1/012019, 2022.

[16] M. Asad et al., "IoT-based Temperature and Humidity Controlling," The Scientific Research Institute Journal, vol. 10, pp. 66-71, 2019.

[17] H. Hashim, M. N. Hazwan, P. S. M. Saad and Z. Harun, "The Real-Time Monitoring of Air Quality Using IOT-Based Environment System," 2023 19th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), Kedah, Malaysia, 2023, pp. 54-58, doi: 10.1109/CSPA57446.2023.10087557.

[18] E. Iadanza, G. Pasqua, D. Piaggio, C. Caputo, M. Gherardelli, and L. Pecchia, "A robotic arm for safe and accurate control of biomedical equipment during COVID-19," Health and Technology, vol. 13, no. 2, pp. 285–300, 2023.

[19] I. K. Hanoon and M. I. Aal-Nouman, "Cloud-based COVID-19 Patient Monitoring using Arduino," in 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), Surabaya, Indonesia, 2021, pp. 292-296, doi: 10.1109/EIConCIT50028.2021.9431881.

[20] C. Ahlawat and R. Krishnamurthi, "Internet of Things-based smart helmet to detect possible COVID-19 infections," in Cyber-Physical Systems, R. C. Poonia, B. Agarwal, S. Kumar, M. S. Khan, G. Marques, and J. Nayak, Eds. Academic Press, 2022, pp. 15-36, ISBN 9780128245576, doi: 10.1016/B978-0-12-824557-6.00004-2.

[21] S. Khriji, D. El Houssaini, R. Barioul, T. Rehman, and O. Kanoun, "Smart-Lab: Design and Implementation of an IoT-based Laboratory Platform," in 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), New Orleans, LA, USA, 2020, pp. 1-5, doi: 10.1109/WF-IoT48130.2020.9221143.

[22] K. Brunda Devi, et al., "Ventilator Using Arduino with Blood Oxygen Sensor for Covid19," IOSR Journal of Engineering (IOSRJEN), vol. 11, no. 07, pp. 20-26, 2021.

[23] S. Mahendrakumar et al., "COVID-19 SOP Compliance And Monitoring Electronic System For Business And Public Places Using Arduino Uno," 2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE), Bengaluru, India, 2023, pp. 478-482, doi: 10.1109/IITCEE57236.2023.10090961.

[24] A. Banagar and R. Khattar, "IoT based Smart Laboratory System," International Journal of Engineering Research and, vol. 9. [Online]. Available: 10.17577/IJERTV9IS010178.

Appendix A: Coding on Arduino IDE

```
#define BLYNK_PRINT Serial
#include <WiFi.h>
//#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <ESP32Servo.h>
#include "DHT.h"

// Define servo motor pin
#define SERVO_PIN_1 4
#define SERVO_PIN_2 18
#define SERVO_PIN_3 19
#define SERVO_PIN_4 21
#define SERVO_PIN_5 22
#define SERVO_PIN_6 23

//Define fan pin
#define fanPin 5

//Define sensor pin
#define DHTTYPE DHT22
#define DHTPIN 12

//Server & Wifi credential
char auth[] = "lg0cavZ9BAXEHQjSVduMYhMbjDcWpnSF";
char ssid[] = "vivo";
char pass[] = "1234567890123";

Servo servoM1;
Servo servoM2;
Servo servoM3;
Servo servoM4;
Servo servoM5;
Servo servoM6;

// Define led and ldr pin
int ldrPin = 36;
int ledPin = 13;

// Define gas sensor and buzzer pin
int gasSensorPin = 39;
int buzzerPin = 14;

//initialize threshold
int gasLmt = 400;
int ldrThreshold = 200;
int temp;
```

```
int gasVal;
int ldrVal;


DHT dht(DHTPIN,DHTTYPE);
BlynkTimer timer;
void chkGas();
void chkLdr();
void chkHumidTemp();

int manualCtrlState = 0;
bool buttonState = false;

// Signal Servo motor 1
BLYNK_WRITE(V4) {
  int tarPos = param.asInt();
  servoM1.write(tarPos);
}
// Signal Servo motor 2
BLYNK_WRITE(V5) {
  int tarPos = param.asInt();
  servoM2.write(tarPos);
}
// Signal Servo motor 3
BLYNK_WRITE(V9) {
  int tarPos = param.asInt();
  servoM3.write(tarPos);
}
// Signal Servo motor 4
BLYNK_WRITE(V10) {
  int tarPos = param.asInt();
  servoM4.write(tarPos);
}
// Signal Servo motor 5
BLYNK_WRITE(V14) {
  int tarPos = param.asInt();
  servoM5.write(tarPos);
}
// Signal Servo motor 6
BLYNK_WRITE(V15) {
  int tarPos = param.asInt();
  servoM6.write(tarPos);
}

//Led manual control
BLYNK_WRITE(V12){
  manualCtrlState = param.asInt();
  buttonState = (manualCtrlState == 1);
```

```
}


void setup()
{
  // Debug console
  Serial.begin(9600);

  //CUSTOM server by BLYNK.CONFIG
  Blynk.begin(auth, ssid, pass, IPAddress(128,199,159,242), 8080);

  digitalWrite(ledPin,LOW);
  pinMode(gasSensorPin, INPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(ldrPin, INPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(fanPin, OUTPUT);
  timer.setInterval(2000, chkLdr);
  timer.setInterval(8000, chkGas);
  timer.setInterval(12000, chkHumidTemp);


  servoM1.attach(SERVO_PIN_1);
  servoM2.attach(SERVO_PIN_2);
  servoM3.attach(SERVO_PIN_3);
  servoM4.attach(SERVO_PIN_4);
  servoM5.attach(SERVO_PIN_5);
  servoM6.attach(SERVO_PIN_6);


  dht.begin();
  while (!Blynk.connected()) {
    Serial.println("Connecting to Blynk server, check your
connection!");
    delay(500);
  }
}

void loop()
{
    Blynk.run();
    timer.run();
    delay(10);
}

//Function to check humidity and temperature
void chkHumidTemp()
{
```

```
  float humid = dht.readHumidity();
  temp = dht.readTemperature();

  if(isnan(humid) || isnan(temp)){
    Serial.println("Unable to receive data from DHT Sensor!");
    return;
  }

  Serial.print("Humidity: ");
  Serial.print(humid);
  Serial.println(" %\t");
  Serial.print("Temperature: ");
  Serial.print(temp);
  Serial.print(" °C\t");

//Write to Blynk app
  Blynk.virtualWrite(V0, humid);
  Blynk.virtualWrite(V1, temp);

  if (temp > 37 && gasVal > 400) {
    digitalWrite(fanPin, HIGH);
    Blynk.virtualWrite(V7, 255);
    Blynk.virtualWrite(V8, 255);
  } else {
    digitalWrite(fanPin, LOW);
    Blynk.virtualWrite(V7, 0);
    Blynk.virtualWrite(V8, 0);
  }

}
//Function to check light intensity
void chkLdr(){
  ldrVal = analogRead(ldrPin);
  if (manualCtrlState == 1) {
    digitalWrite(ledPin, buttonState ? HIGH : LOW);
    Blynk.virtualWrite(V3, buttonState ? 255 : 0);
  } else { // Automatically control LED using LDR
    digitalWrite(ledPin, ldrVal > ldrThreshold ? LOW : HIGH );
    Blynk.virtualWrite(V3, ldrVal > ldrThreshold ? 0 : 255);
  }
  Serial.print("LDR value: ");
  Serial.println(ldrVal);
}

//Function to check gas concentration
void chkGas()
{
  delay(2000);
```

```
  gasVal = analogRead(gasSensorPin);
  Serial.print("Gas ppm: ");
  Serial.println(gasVal);
  digitalWrite(buzzerPin, gasVal > gasLmt ? HIGH : LOW);
  Blynk.virtualWrite(V6, gasVal > gasLmt ? 255 : 0);

  if (temp > 37 && gasVal > 400)
    {
      Serial.println("High temperature and gas concentration 
detected!");
      digitalWrite(fanPin, HIGH);
      Blynk.virtualWrite(V7, 255);
      Blynk.virtualWrite(V8, 255);
    }
    else if (gasVal > gasLmt)
    {
      Serial.println("Smoke detected!");
      digitalWrite(fanPin, HIGH);
      Blynk.virtualWrite(V7, 255);
      Blynk.virtualWrite(V8, 255);
    }
    else if (temp > 37)
    {
      Serial.println("High temperature detected!");
      digitalWrite(fanPin, HIGH);
      Blynk.virtualWrite(V7, 255);
      Blynk.virtualWrite(V8, 255);
    }
    else
    {
      Serial.println("Normal conditions");
      digitalWrite(fanPin, LOW);
      Blynk.virtualWrite(V7, 0);
      Blynk.virtualWrite(V8, 0);
    }
}
```

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.:1
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE

This fortnight will include the development of a robotic arm; the water pump was successfully built using a servo motor, and the robotic arm was successfully built with extra stability and robustness using a spring attached to the robotic arm. The water pump was built to push the pipette with the help of a servo motor.

2. WORK TO BE DONE

Next fortnight, I need to connect and combine all the components and control them using ESP WROOM 32 and BLYNK application.

3. PROBLEMS ENCOUNTERED

The problem encountered was there was a leak in the opening of the water pump, and it took a lot of time to find the correct spring diameter and length in order to fit into the robotic arm.

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Supervisor's signature

kenneth

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT
(Project II)

Trimester, Year: Y3S3	**Study week no.:3**
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE

This fortnight, I combined all the components of my projects, including servo motors, sensors and output devices, into one single project and managed to control all of them using ESP WROOM 32.

2. WORK TO BE DONE

I will finalize my user interface on the BLYNK application and wrap up the connection and wiring for next fortnight.

3. PROBLEMS ENCOUNTERED

The problem encountered was after combining all the components, the ESP WROOM 32 was overheating, and the robotic arm was not so responsive after combining all the components.

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Supervisor's signature

kenneth

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.:5
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE

I finalized my Blynk application user interface and the robotic arm's exterior design.

2. WORK TO BE DONE

For the coming week, I will start writing my report for my FYP. The chapter that I am going to cover includes Chapter 3 on system methodology.

3. PROBLEMS ENCOUNTERED

There was no problem encountered during this week

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.:7
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

I had completed Chapter 3, which was about system methodology. This chapter included a system architecture, a use case, and an activity diagram. All my diagrams were submitted and checked by my FYP supervisor, Miss Oh.

2. WORK TO BE DONE

For the coming week, I am continuing with chapter 4, which is system design. This chapter includes a system block diagram, system component specification, circuits and components interaction operations and system components interaction operations.

3. PROBLEMS ENCOUNTERED

There was no problem encountered during this week

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Supervisor's signature

kenneth

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.:9
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE

I completed Chapter 4, which is system design. This chapter includes a system block diagram, system component specification, circuits and components interaction operations and system components interaction operations. I have described in detail how my project was developed. Someone reading the chapter would want to rebuild my system can rebuild my system quickly.

2. WORK TO BE DONE

For the coming week, I will write Chapter 5 about system implementation.

3. PROBLEMS ENCOUNTERED

There was no problem encountered during this week

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	Study week no.:11
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE
For this week, I completed Chapter 5, including hardware setup, software setup, setting and configuration, system operation, implementation issues and challenges, and concluding remarks.
2. WORK TO BE DONE
For the coming week, I will write Chapter 7 about conclusion and recommendation.
3. PROBLEMS ENCOUNTERED
There was no problem encountered during this week
4. SELF EVALUATION OF THE PROGRESS
I was fully attached to the progress of my planning and completed all the tasks on time.

Xin

Supervisor's signature

kenneth

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT
(Project II)

Trimester, Year: Y3S3	Study week no.:13
Student Name & ID:20ACB06669	
Supervisor: Miss Oh Zi Xin	
Project Title: DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM	

1. WORK DONE
[Please write the details of the work done in the last fortnight.]

For this week, I completed chapter 7 by concluding my project and giving an appropriate recommendation for improving the project. I sent my project for a Turnitin check.

2. WORK TO BE DONE

3. PROBLEMS ENCOUNTERED

There was no problem encountered during this week

4. SELF EVALUATION OF THE PROGRESS

I was fully attached to the progress of my planning and completed all the tasks on time.

Xin
Supervisor's signature

kenneth
Student's signature

POSTER


UTAR
UNIVERSITI TUNKU ABDUL RAHMAN
FACULTY OF INFORMATION COMMUNICATION AND TECHNOLOGY
DEVELOPMENT OF IOT-BASED SMART LABORATORY SYSTEM
Introduction
The impact of COVID-19 has brought significant death to the human species, significantly contributing to approximately 80,000 to 180,000 healthcare workers. The main reason contributing to the high death rate of the workers is the lack of personal protective equipment. IoT can prepare healthcare workers for possible upcoming more deadly virus pandemics.
Method
1. This project uses an ESP WROOM 32 microcontroller to communicate with all sensors, servo motors, and BLYNK applications.
2. This project used a modified robotic arm to perform pipetting using an external power supply. ESP WROOM 32 uses a logic level converter to control the servo motors.
Results
ESP WROOM 32 has communicated with all sensors, six servo motors, and the BLYNK application. All real-time data is displayed in the serial monitor in Arduino IDE and BLYNK application. All outputs like the DC fan, buzzer, and LED have responded automatically when triggered by ESP WROOM 32. The robotic arm was improved by adding spring on the backbone and successfully passed the pipetting test.
Discussion
The reliability and accuracy of the robotic arm require improvement. A pneumatic system can replace the servo motors. Besides, the integration of the robotic system increased the temperature of ESP WROOM 32. Heat sink can be pasted on the ESP WROOM 32. A higher ampere power supply is required to support the robotic arm.
Hardware and Software:
ESP WROOM 32 microcontroller
MQ2 Gas Sensor
DHT22 (Temperature & Humidity sensor)
220 ohm resistor
NPN transistor & Diode
Breadboard
5V DC fan & 3.3V buzzer
Robotic arm Aluminium rack & Six servo motors
LED & LDR
External power supply & Logic Level Converter
Arduino IDE, Fritzing, BLYNK legacy
Conclusion
IoT robotic arm is a technology that will provide a contactless laboratory environment to deal with future deadly virus pandemics and save the lives of medical healthcare workers.
SUPERVISOR: MISS OH ZI XIN
DEVELOPER: KONG WAI KIN

PLAGIARISM CHECK RESULT

FINAL FYP2 DOCUMENT - Copy.docx

ORIGINALITY REPORT

15%	11%	7%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.utm.my Internet Source	2%
2	W. T. Wan Faizura, M.Z.M. Luqman, A. H. Luqman Hakim, O. Mohd Hafiz, I. Zulkifli. "IoT based Robotic Arm for Student Online Learning", Journal of Physics: Conference Series, 2022 Publication	1%
3	Hashimah Hashim, Muhamad Naqiuddin Hazwan, Puteri Sarah Mohamad Saad, Zambri Harun. "The Real-Time Monitoring of Air Quality Using IOT-Based Environment System", 2023 19th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), 2023 Publication	1%
4	Sabrine Khriji, Dhouha El Houssaini, Rim Barioul, Talha Rehman, Olfa Kanoun. "Smart-Lab: Design and Implementation of an IoT-based Laboratory Platform", 2020 IEEE 6th	1%

63	Shamar Ward, Mechelle Gittens. "Locating Optimal Study Areas Using Augmented Reality And Low Cost IoT Devices", 2022 IEEE Games, Entertainment, Media Conference (GEM), 2022 Publication	<1 %
64	Xugui Qiang, Wangde Wen. "Simulation study on the trajectory planning of table tennis robot arm by applying polynomial interpolation algorithm", Journal of Physics: Conference Series, 2023 Publication	<1 %
65	diyprojectslab.com Internet Source	<1 %
66	ebin.pub Internet Source	<1 %
67	thepihut.com Internet Source	<1 %
68	www.irjmets.com Internet Source	<1 %
69	www.lionlaboratories.com Internet Source	<1 %
70	www.uctronics.com Internet Source	<1 %
71	www2.mdpi.com Internet Source	<1 %

Exclude quotes	On	Exclude matches	< 8 words
Exclude bibliography	On		

Universiti Tunku Abdul Rahman			
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

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	KONG WAI KIN
ID Number(s)	20ACB06669
Programme / Course	Bachelor of Computer Science
Title of Final Year Project	Development of IOT-Based Smart Laboratory System

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: ___15___ % **Similarity by source** Internet Sources: ______11_______% Publications: ____7_____ % Student Papers: _____7_____ %	-
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.

______________________	______________________
Signature of Supervisor	Signature of Co-Supervisor
Name: ____Oh Zi Xin_______	Name: __________________________
Date: _____26/4/2024______	Date: ___________________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB06669
Student Name	KONG WAI KIN
Supervisor Name	Miss Oh Zi Xin

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.
√	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 Symbols (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.

kenneth

____

(KONG WAI KIN)
Date:26/04/2024