Faculty of Electrical and Electronic Engineering Technology

DEVELOPMENT OF SMART HOME SYSTEM USING ARDUINO

DHARSHANADEWI A/P CHANDRASEGAR

Bachelor of Computer Engineering Technology (Computer Systems) with Honours

2021

DEVELOPMENT OF SMART HOME SYSTEM USING ARDUINO

DHARSHANADEWI A/P CHANDRASEGAR

A project report which is submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Computer Engineering Technology (Computer Systems) with Honours

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I declare that this project report entitled “DEVELOPMENT OF SMART HOME SYSTEM USING ARDUINO” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Student Name : DHARSHANADEWI A/P CHANDRASEGAR

Date : 10/01/2022

APPROVAL

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

DR. JAMIL ABEDALRAHIM JAMIL ALSAYAYDEH
Pensyarah Kanan
Jabatan Teknologi Kejuruteraan Elektronik dan Komputer
Fakulti Teknologi Kejuruteraan Elektrik Dan Elektronik
Universiti Teknikal Malaysia Melaka (UTeM)

Signature : ....

Supervisor Name : DR JAMIL ABEDALRAHIM JAMIL ALSAYAYDEH

Date : ....

Signature : ....

Co-Supervisor Name (if any) : DR ADAM WONG

Date : ....

DEDICATION

This desertion is dedication to my late aunt who have always wanted to see me graduating and also my parents who are the reason I am here today. Thankful for my supervisor who has guided me throughout this journey. Last but not least all these would not have been possible without the blessings from the man above. Beyond thankful to everyone who was with me this far to complete this thesis.

ABSTRACT

This project presents the design and prototype implementation of an Arduino based real-time home control system that can be used to monitor home appliances remotely via the Internet. In this system, sensor controlling monitoring system is made for fan and light by using Wi-Fi module which are developed from scratch. All the devices will be turned on using motion sensors. Besides that, this system also has fire alarm system where it detects smoke and alerts the owner via the application. Arduino UNO is used as the main microcontroller as a brain for this project. Results show that this proposed system allows a variety of devices to be controlled in a simple and interactive way, provides power consumption monitoring of devices to ensure the controlling process, and stores real-time data to the cloud in order to be accessed anywhere and anytime.

ABSTRAK

Laporan ini merupakan cadangan projek sistem pemantauan rumah menggunakan arduino dengan kos rendah serta memudahkan pengguna. Projek ini akan membantu mengawal kos elektrik serta merasai teknologi yang canggih menggunakan IoT. Sistem ini menggunakan cara kawalan melalui aplikasi yang menyambung kepada Node MCU yang juga disambung kepada Arduino UNO. Projek ini menghasilkan sebuah sistem yang menarik dengan cara kawalan menggunakan "sensor" yang menafsirkan gerakan serta kebocoran gas. Pengguna akan diberitahu jika ada kebocoran gas di rumah melalui aplikasi tersebut. Manakala lampu serta kipas pula akan dihidupkan jika ada kehadiran manusia. Sistem ini amat canggih serta mudah digunakan bila-bila masa. Hasil kajian menunjukkan bahawa sistem yang dicadangkan ini memungkinkan pelbagai perangkat dikendalikan dengan cara yang sederhana dan interaktif, menyediakan pemantauan penggunaan tenaga untuk memastikan proses pengendalian, dan menyimpan data masa nyata ke awan agar dapat diakses di mana saja dan kapan saja .

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Dr. Jamil and co-supervisor, Dr Adam Wong for their precious guidance, words of wisdom and patient throughout this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for giving me an opportunity to get my Degree in Computer Engineering. Not forgetting my fellow colleagues for always sharing their knowledge and being a part of this journey.

My highest appreciation goes to my late aunt Rajeswari, parents and my close friends for supporting me financially and also making sure that I am mentally strong as well to complete this project. They have always been my support system since day one. I am definitely beyond blessed to have their love and support.

Finally, I would like to thank all the staffs at FTKEE and also my classmates, the faculty members, as well as other individuals who are not listed here for being co-operative and helpful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

This Smart Home System is developed in order to assist users, convenient, time saving and simple. In many homes there are no safety measures taken as this technology costs a lot for many. Thus this project is developed in order to make it convenient for user to turn ON/OFF the switch this is utmost to reach the elderly people who are physically incapable of doing things on their own. Besides that, a smoke detector attached to this system in order to prevent the home from being caught by fire. In this modern world everyone is busy with the hectic lifestyle this causes chances for users to be forgetful. This system will be able to prevent the house from burning. The idea of this smart home is to build a system which is ideal for user inside and outside the home. In this project Arduino is used as the main microcontroller to connect to other peripherals such as Wi-Fi module, MQ-2 smoke sensor and also PIR sensor. These will be connected to the user's smartphone and from there the user will get to control the electrical appliances, alert message of smoke detection and also intruder alert.

1.2 Background

The fast technological advances have revolutionized modern society's living standards. Electricity, radio, TV, and telephones are now readily available in most homes. Household duties that were formerly a time-consuming procedure have now been greatly improved thanks to customized machinery. As example the normal household appliances such as washing and drying machines minimizes the time spend of a labor. Water heaters, on the other hand, have rendered bathing more convenient at any time, while freezers have taken over all the work of food preservation entirely. Heating, Ventilation and Air Conditioning (HVAC) appliances which supply additional enhanced satisfaction. With the rising development of various gadgets in the home, a remote, centralized, integrated smart monitoring system has become an incredibly effective and demanded aspect. Every safety aspect is very important a home thus a smoke sensor is also necessary for a home. A smart home system connects household appliances, permitting for more accessibility, energy efficiency, and security. Designers are allocating a solution to make it more convenient in this project by implementing an automation system that controls the appliances.

.

1.3 Problem Statement

For years, science has pushed the possibilities or notion of the first smart house. The first home automation was a notion rather than actual building. This was initiated due to the trouble or inconvenience caused to citizens. Therefore, many alternatives and solutions were created to ease them. As the time goes by and technology develops far more advanced gadgets is needed to make sure safety and conveniences are fulfilled.

Primarily it started in television and radio among citizens but as time passes technology happens to be in fingertips and eases in all way. This project majorly focuses on safety and convenience of users. To many of the citizens their house is the main asset and due to a hectic schedule and lifestyle most of them tend to forget to check on their stove and gas leakage in the kitchen. Thus many fire accidents happen due to unforeseen circumstances and people tend to lose their home. By having a smoke detector in their home which is directly connected to their mobile phones it will ease the user to check on their home easily.

Next electricity bill happens to be the most expensive expenditure in every household. This happens due to the electricity turned on for a long time thus by turning it on when necessary would help to reduce the electricity bill and also the global warming. To overcome this the motion sensor is used to detect the appearance of people and turn on the electrical appliances accordingly and the most commonly used such as fan and light.

Present life safety devices lack wireless connectivity, making it unable to communicate with the outside world. The potential of smart house automation, would be similar to what is telecasted in the animated series, which is all in digital.

1.4 Project Objective

The objectives are as below:

a) To develop and build a microcontroller-based system capable of monitoring devices in the home system.

b) To create a sensor-controlled system in the home that allows data transfer via a wireless transfer medium.

c) To connect the system to a web browser, allowing for cross-platform control.

1.5 Scope of Project

The work scopes of this project include the study on how to design and develop a smart home system. This project has initiated that there will be motion sensor controlled electric appliances such as light, fan and adding on with smart fire alarm system in the home. To detect the movement in order to switch on the appliances a PIR sensor will be used in this project connected to the Arduino. This PIR sensor is sensitive to the motion or movement and will then be converted to the switch and turn ON the fan & light. A microcontroller and relay is used to control the switching of light. The fan is switched on and off like as light but it also depends on the room temperature. Besides that, different types of approaches have been made towards home automation. The Wi-Fi module technology available in phones to communicate with a microcontroller which acts as the main control for access to home appliances. An alarm system can be developed output values of the smoke detector. The MQ-2 Gas/Smoke sensor was utilised in this design. It is vulnerable to LPG, Hydrogen, Smoke, Methane, Propane, Alcohol, Butane, and other chemical fumes used in industry. The non-inverting terminal of LM358 is connected with output of smoke sensor. When the air is pure, the conduction between the electrodes is lower at first because the resistance seems to be on the order of 50K. The inverting terminal input of comparator is higher than the non-inverting terminal input. When the sensor is loaded with smoke during a fire, the resistance drops to 5K and the conductivity between the electrodes rises.This provides a higher input at the non-inverting terminal of comparator than the inverting terminal and the output of comparator is high. The notification is sent to the user via the Wi-Fi module as an indication of presence of smoke.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter contains several past project theses, articles, journals, and conferences, as well as a brief summary and those related to the project *"Development of Smart Home System Using Arduino"*.

This study shows how Wi-Fi modules play a huge role in today's technology to ease the user. Every thesis has its own highlights on how the project works. With referring to this project's the scope of this project may enhance.

Besides that, this projects are all unique in own way thus what is compatible with this project will be highlighted below. It contains more details on the hardware and software components which will be used in this project.

2.2 Overview of Smart Home System

In today's modern technology users need every peripheral device to be controlled at their fingertips. Thus this project contributes to it and also eases the user to interact with their appliances at home. From basic turning ON/OFF lights to detecting smoke in the kitchen and also security system will be focused in this project. To make this a success studies have been made to ensure the best quality is provided to users. Home automation is also made simple, efficient and budget friendly.

2.3 Past Project Summary

According to numerous Authors' based on the studies have tried Home Automation System in various way using Bluetooth, Voice-Recognition, Wi-Fi Module, GSM and many more. Here are few past projects that was studied:

2.3.1 Development of Home Automation using various methods

Bluetooth-based home automation has several limitations; IoT-based automation is becoming more common. Currently any device, be it television sets, speakers, or some other electronic system, is becoming remote controlled. A consumer had to hold separate remote controls for various equipment in traditional control systems. To address this, a user-friendly automation system with smart switching that can be managed using Android smartphones was developed. The user would be able to monitor all of the appliances from his or her smartphone as a result of this. The consumer would be able to monitor and change the volume and speed of the devices in addition to turning them on and off. This is accomplished by connecting the GSM and Wi-Fi module to Arduino board by programming an Android app to control it. Since their intended users are farmers, this device is made easy and cost-effective as possible. It's also no secret that farmers must often travel to their farms just to start a motor or other system. Furthermore, internet access may not be accessible in some remote areas. So, with all of this in mind, two different versions of the methods were created. One is designed specifically for farms, while the other is designed for general use [1]. Using Wi-Fi module, the control actions can be performed by entering the cloud's IP address. The benefit of putting a webserver on the cloud is that the user can access home from any laptop that has WIFI 802.1 and a web browser. Control actions can be performed by entering the cloud's IP address. Wi-Fi module attached to 9 Arduino UNO can accept data sent from PC over Wi-Fi. The Arduino UNO reads the data and uses Relays to control the switching of electrical devices attached to it [2].

Method 1: A SIM card is included in the GSM module. When a user needs to switch on or off different equipment such as a fan, light, pump, or motor, all there is need to be done is send an SMS to a PRE-registered number, which is the number of the SIM card in the GSM module. Sending an SMS to that number will cause the GSM module to receive the message and send a signal to the Arduino UNO, which will then provide a digital signal to the control circuit. The control circuit will send a signal to the driver circuit, which will then drive the relay, turning on or off the selected unit. Since it need different power supplies, the GSM module and the microcontroller Arduino UNO are operated separately from the single source using ICs [1]. In this project it has merged home surveillance and automation into a single project and installed a support button for an elderly user. This will save electricity and provide protection to the device so if anyone enters without authorization or if gas leaks, a response will be transmitted by GSM modem. Energy savings are achieved by remotely monitoring home appliances based on the desired conditions [2]. The block diagram of Home Automation using GSM modem is as shown in Figure 2.1 below.

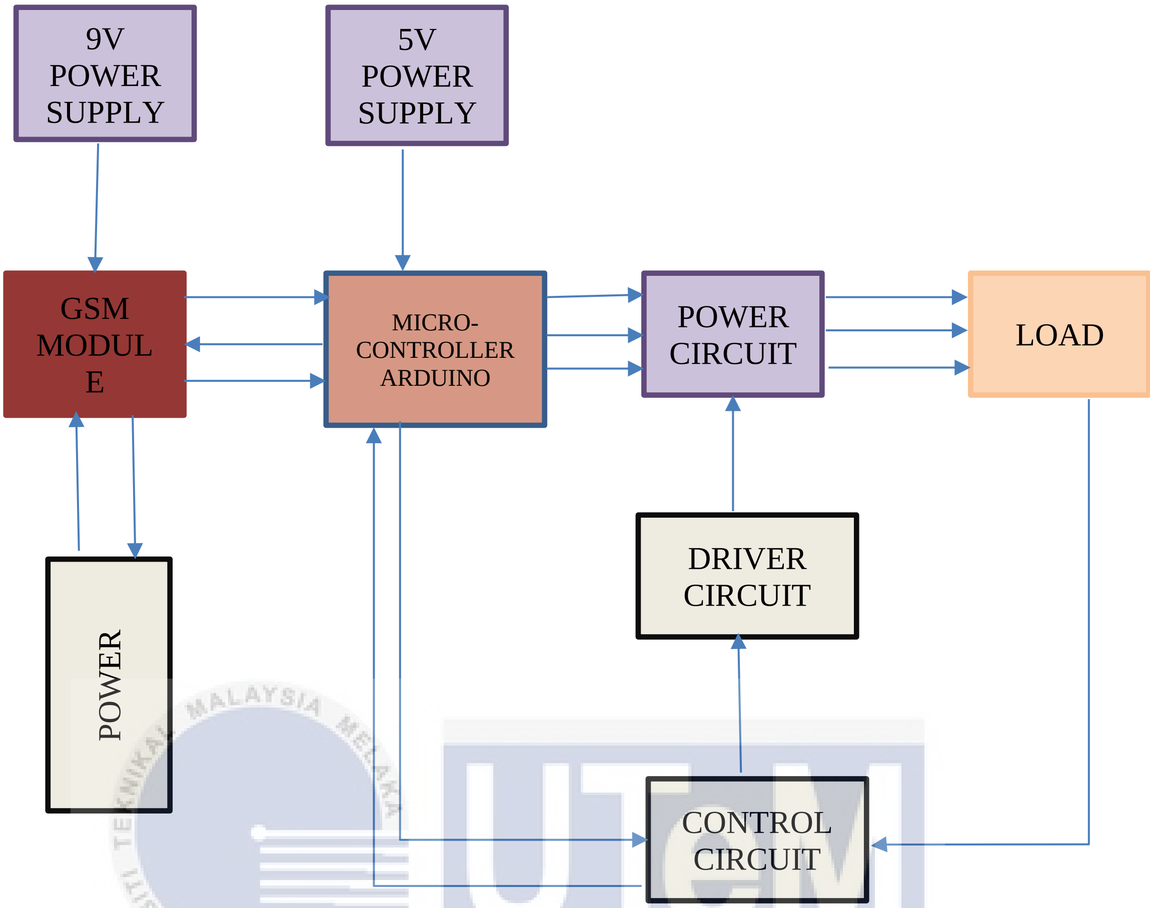

Figure 2.1: Block Diagram of GSM module based automation[2]

Method 2: The operation of this system is somewhat close to that of a GSM-based system; the only change is that instead of using SMS to switch on and off computers, a Web Page can be used. A Wi-Fi module, 3.3V dc, and other circuits are powered by a separate power supply from the same source. When the Wi-Fi module is turned on, it will immediately connect to the Wi-Fi router that is already set up in the home or office. A user can conveniently monitor their computers or equipment by accessing the website and pressing the ON/OFF button in the programme [1]. Each Wi-Fi module has a unique IP address. The sensors can be read, and the machines can be turned on and off using a Personal Computer (PC) and Wi-Fi. These may be used to repair existing switches in the home that cause sparks and, in some cases, fires. Considering the benefits of Wi-Fi and advanced automation. To monitor the appliances in the house, a system was developed. Control actions can be performed by entering the cloud's IP address [3]. The block diagram of Home Automation using Wi-Fi module is as shown in Figure 2.2 below.

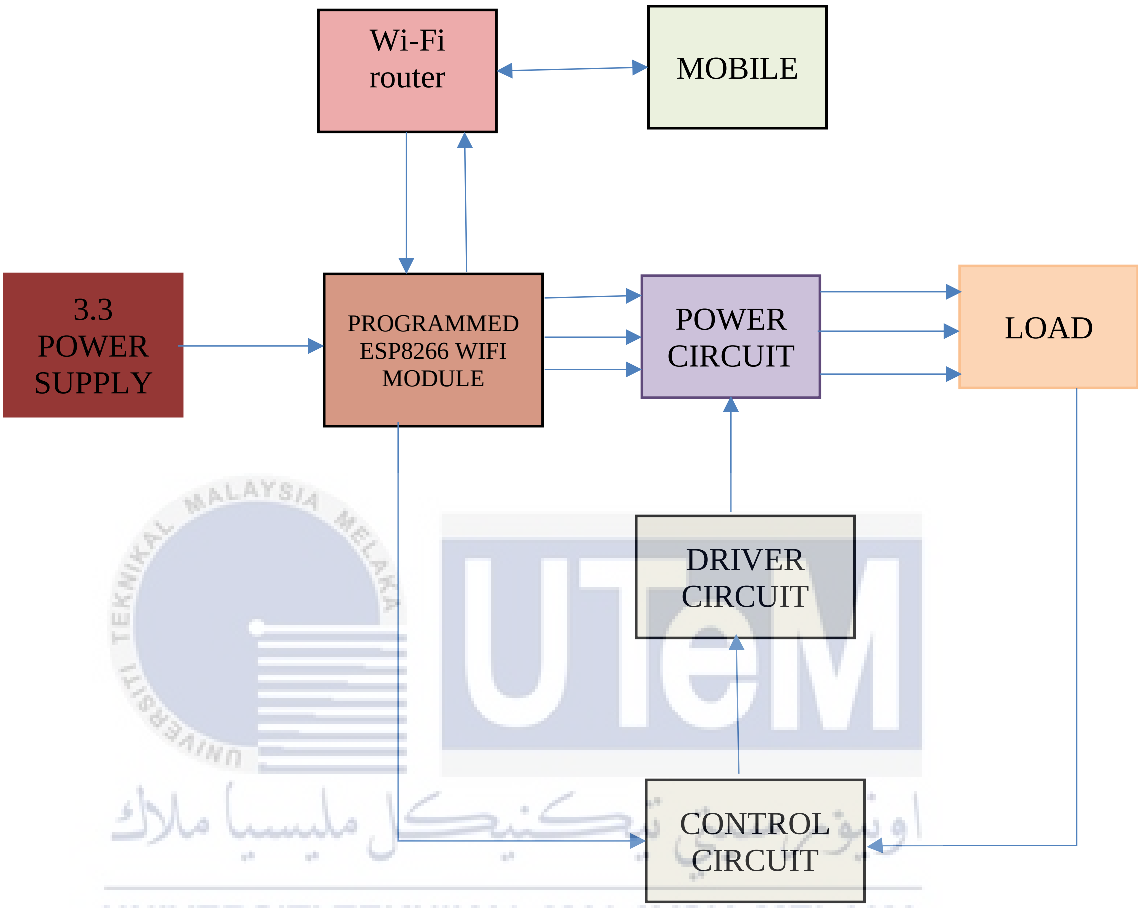

Figure 2.2: Block Diagram of Wi-Fi module based automation[3]

2.3.2 Home Automation System Using Sensors

IoT systems are used extensively in automation. That is the management of energy use and environmental protection in homes, hospitals, classrooms, museums, and offices through the use of different sensors and actuators that control lighting, temperature, and humidity [4]. Though there was no motion present, the sensor sensed motion and remained high for a period of time [17]. All of the home control automation systems use wireless technologies, as shown by the above-mentioned surveyed documents. In both of these schemes, the smart phone is extremely important. GSM technology is found in two different schemes. As shown in this home automation method, a controller such as an ARM7 (ARM7 LPC2148 monitor, PIC16F877 (40 pin IC), ARM9, etc. is used. Among almost every device, the ULN2003 is used to drive the relays. App inventor, embedded C, Keil Compiler, VB.NET, and other computer tools are included, as well as Bluetooth modules LM400 with a span of 100 metres, frequencies of 2400Hz, and a pace of 3 Mbps [4]. In general, the concept of smart homes has sparked the interest of academia, lifestyle experts, and purchasers for the usage of new technology. The crucial part of this dissertation is the Arduino microcontroller. They've also used a PIR (Passive infrared) sensor to identify the presence of a human within a certain range. This sensor, rather than transmitting light like LEDs, measures the amount of difference in infrared rays that happens as a person whose temperature is different from the ambient temperature approaches. The aim of this dissertation is to demonstrate how the Arduino microcontroller can be used to conserve electricity to some degree [5].

2.3.2.1 ZigBee RF communication

The communication protocol used in this framework is the ZigBee protocol. Although ZigBee's highest baud rate is 250 kbps, 115200 bps has been used for transmitting and receiving because it was the fastest speed that perhaps the microcontroller's UART could be programmed to work at. The wireless design of ZigBee helps to solve the previously mentioned noisy implementation issue for current home automation systems. The ZigBee protocol technically supports a data rate of 250 kbps, but since 40 kbps is adequate for most control systems, it is sufficient for controlling most home automation equipment. The reduced installation and operating costs are appealing [5].

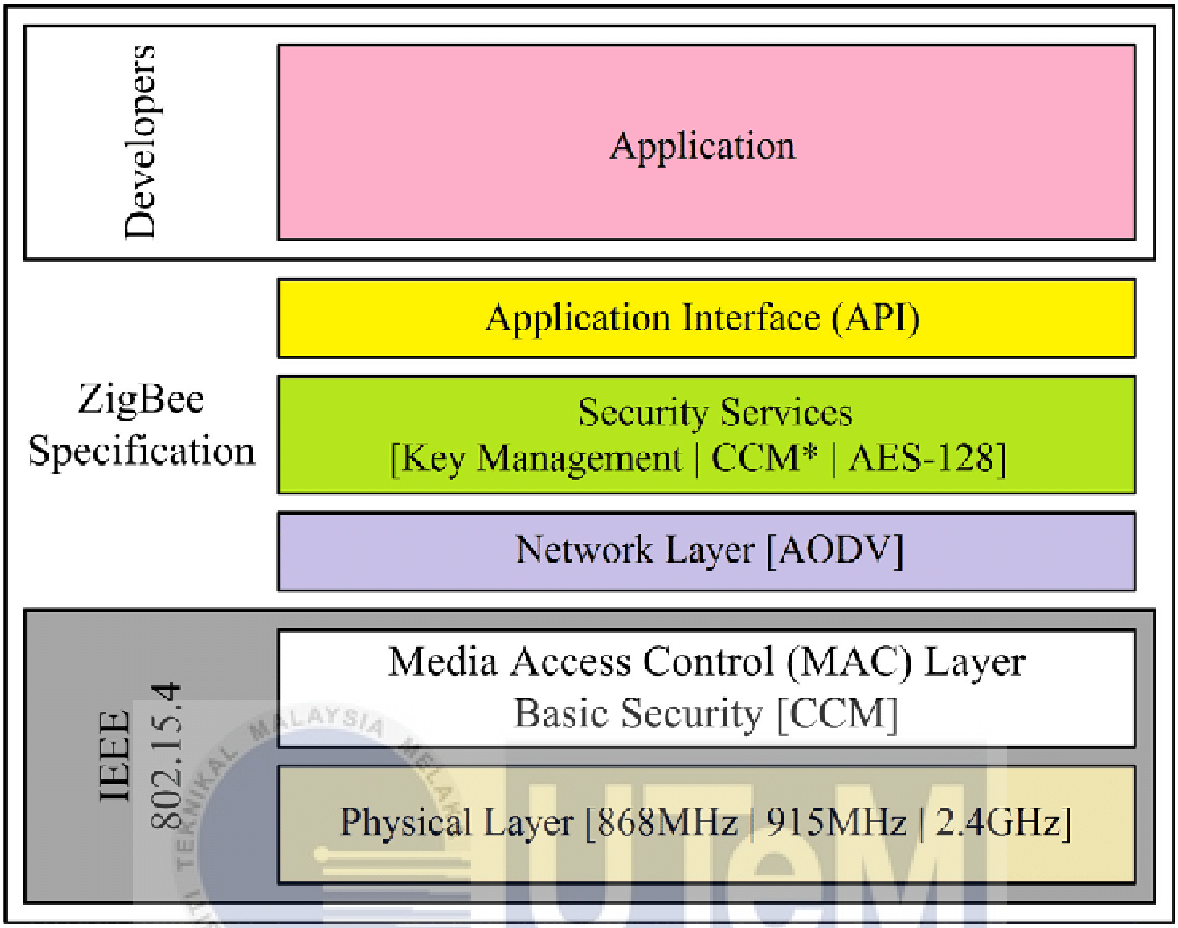

Figure 2.3: ZigBee RF communication[5]

2.3.2.2 Voice Recognition Control

Microsoft Speech API is used in the voice recognition programme. The programme compares incoming speech to a predefined dictionary that is easily accessible. Automatic Speech Recognition (ASR engine) and Text To Speech are the two major engines used in the Microsoft speech API operating system (TTS engine). To determine the spectrum of the fingerprint data, ASR uses the Fast Fourier Transform (FFT) [5].

Figure 2.4: Voice Recognition Control

2.3.2.3 Appliance Control Module

Control signals will be sent to the required appliance address using the ZigBee communication protocol once the speech commands have been recognised. A relay controlling circuit is installed in each appliance which needs to be operated. For voice recognition, the speech recognition technology employs a single-chip solution. The LD3320 is a speech recognition voice chip built on SI-ASR (speaker-independent automatic speech recognition) technology [5]. The home automation system with smart job scheduling is created by utilising wireless ZigBee to link appliances [13]. As shown in the Figure 2.5 the appliance control module breakdown. The LD3320 includes a powerful speaker-independent speech recognition search engine module as well as a comprehensive speaker-independent speech recognition feature library in order [5].

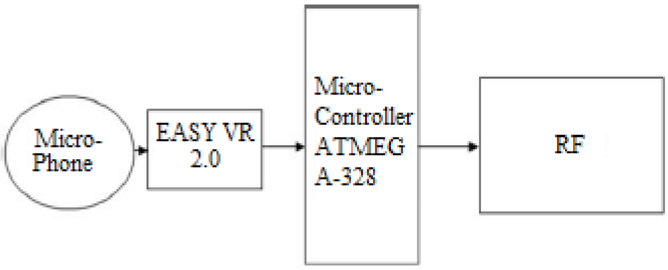

Figure 2.5: Appliance Control Module[5]

2.3.3 Using Arduino in Home Automation

Web Services provide a transparent and interoperable means of delivering remote support services or allowing programmes to connect with one another. Arduino is an open electronics prototyping framework built on a modular and user-friendly hardware and software platform. The microcontroller on the Arduino I board has 54 optical input/output pins. The serial peripheral interface (SPI) pins in Arduino are used to implement the Wi-Fi interface. Lower transmission relays were being used to integrate devices with Arduino, and the switching features will be demonstrated [6]. The Arduino UNO microcontroller, that also serves as a micro web server and interface for all hardware components, is at the core of this device. Environmental tracking using temperature, humidity, gas, and smoke sensors is one of the features of the smart home system. It also has switching capabilities for controlling lights, fans/air conditioners, and other relay-connected home appliances. Another function of this device is the intrusion detection it provides through the motion sensor, which can all be operated remotely [7].

2.3.3.1 Main Components

a) Arduino UNO

Arduino is a one-of-a-kind Arduino board with a WIZ net Wi-Fi port, a Wi-Fi connector, a nRF24L01+ module interface, and an ATMega328 processor. This board would have wireless capabilities. The development process is controlled by an ESP8266 Wi-Fi shield and has internet access [14]. In Italian, the word "uno" means "one," and it was selected to commemorate the release of Arduino Software (IDE) 1.0 [15]. Via jumper wires, the Arduino board is attached to the relay and the sensors. The Arduino's ground pin is attached to the ground pin on the various sensors. The temperature sensor, smoke sensor, flame sensor, and PIR sensor are all attached to the Arduino board's (A0, A1, A2) pins and provide data. The Arduino's digital pin gives feedback to the relay, which then executes the desired action as shown in the figure below [8]. It holds the ESP8266 Wi-Fi module, which is linked to the web server through an IP address [18]. Arduino offers a cost-effective and efficient platform for implementing a smart home automation system [14].

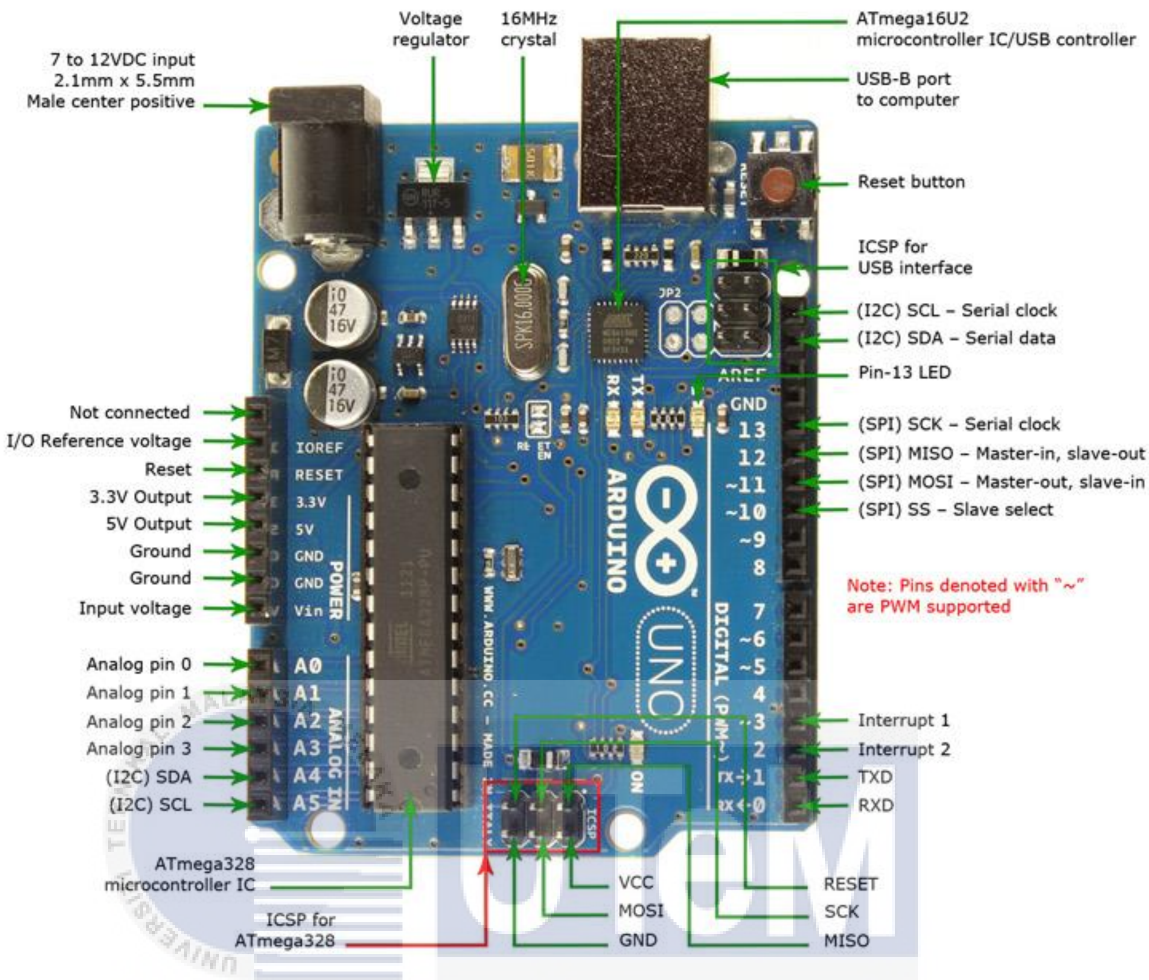

Figure 2.6: Arduino UNO Pin Diagram

b) Arduino program

To interface with the programme, the Arduino-Uno board should be updated with *javascript*. Arduino offers an adaptable stage that creates a code that can be executed by the Arduino and transferred to the surface. The all-inclusive Synchronous Asynchronous Receiver Transmitter (USART) convention is utilized to link the Atmega 328 with Electrically Erasable Programmable Read Only Memory (EEPROM). Atmel studio 6.0 was used to write the code in Embedded C. After that, the code is rearranged and converted to HEX code. The HEX code is then scorched to the Atmega 328 microcontroller a short time later [9].

2.3.4 Home Automation Using IOT

Home robotization structures have become unavoidable in recent years, coinciding with advancements in the Internet of Things' potential. Regardless of the fact that mechanisation for workplace systems is a huge achievement, technology and automation solutions for households are a relatively new invention that customers are embracing. Home automation incorporates the monitoring and management of activities such as lighting, heating, ventilation, and air conditioning (HVAC), electrical mechanical meetings, sound frameworks, vision cameras, passage shocks, and warnings. Comfort, increased stability, and critical viability are only a few of the benefits of home robotization [9]. The Internet of Things (IoT) has enabled the prospect of bringing all items on the earth's surface under one roof, allowing for inter-connectivity and interaction for data exchange [12]. Wi-Fi is a worldwide open standard based on the IEEE 802.15.4 MAC/PHY standard. To endorse enhanced mesh routing features, Wi-Fi establishes a network layer above the 802.15.4 layers. The Wi-fi Alliance, a growing group of organisations, is responsible for developing the Wi-fi specification. Over 300 participants make up the Alliance, which includes semiconductor, module, stack, and software developers [6]. A communicating module called ESP8266-01 is used to link these devices to local Wi-Fi [10].

2.3.4.1 Node MCU

This microcontroller was designated because of its inexpensive cost, small size, and the availability of an integrated Wi-Fi module [19]. Via an application, the Node MCU is configured to monitor various types of appliances. The Node MCU's digital pins (D1, D2, D3, D4) are attached to the relay's input socket. The application is intended to be hooked to the same IP address as the Wi-Fi module in order for signals to be traded. When signal is generated, the user selects an appropriate command from the application, which transmits signals to the Node MCU, which then transfers the signal to the relay, which is programmed to do such tasks as monitoring the utilities. The user selects which commands to send to the Node MCU from a menu of options in the programme. The programme sends a signal to the Wi-Fi module (Node MCU), and when the user launches the software, he or she is given a field to enter an IP address, allowing the user to access the equipment from anywhere as long as it is linked to the Network [8]. This technology allows the user to connect to various devices wirelessly via the Wi-Fi module and provide commands to the server [20].

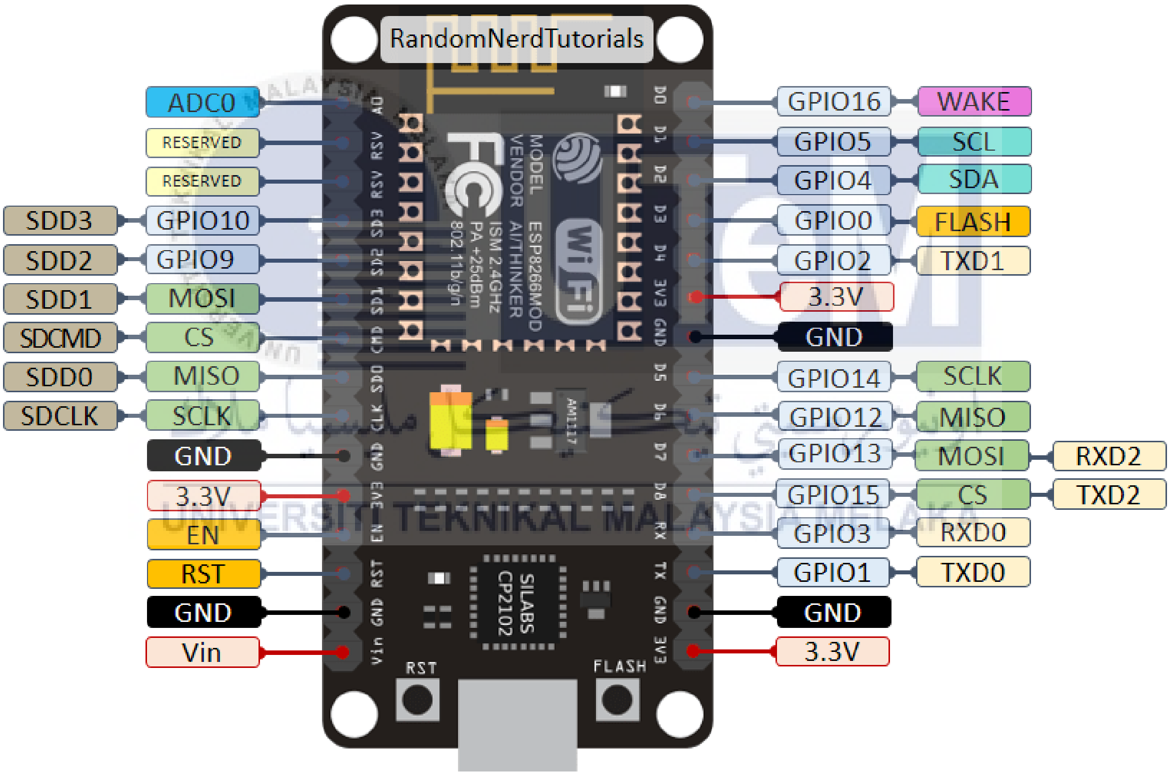

Figure 2.7: Node MCU Pin Diagram

2.3.5 Hardware Implementation in Home Automation

For the home automation scheme, a model house is constructed [7]. Once the sensor detects a human, it transmits a data to the microcontroller [16]. The light sensor senses darkness, Light 1 will immediately turn on. When the room temperature rises above the fixed threshold, a cooler or fan may activate, lowering the temperature. The MQ-6 gas sensor is installed in the kitchen to detect any gas leaks; if a leak is found, the hall alarm is activated [7]. Based on the Figure 2.8 it is well described about the block diagram of this design. The four different appliances, including the fan, lamp, room heater, and television, are controlled remotely via Wi-Fi and an app for Android or iPhone. These devices are connected using the Arduino Uno's digital input/output pins [10]. In the Figure 2.9 shown the hardware connection using Arduino to the sensors.

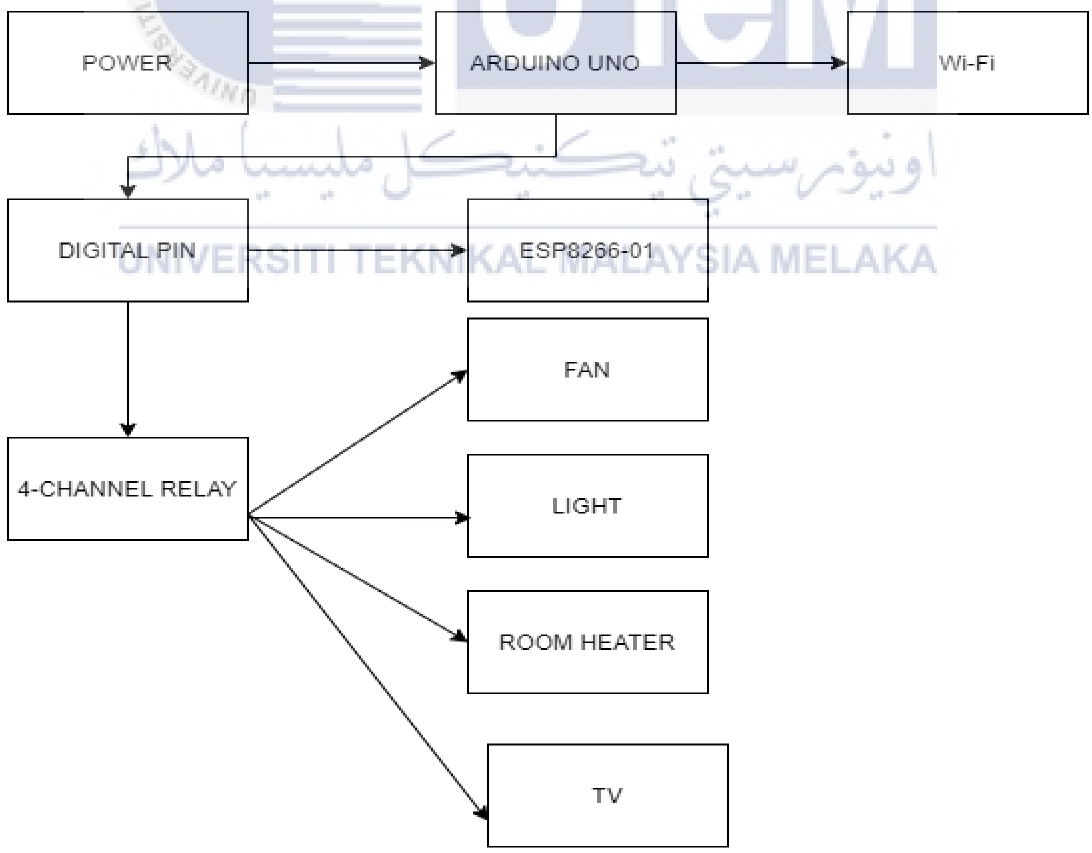

Figure 2.8: Block Diagram of the Home Automation[10]

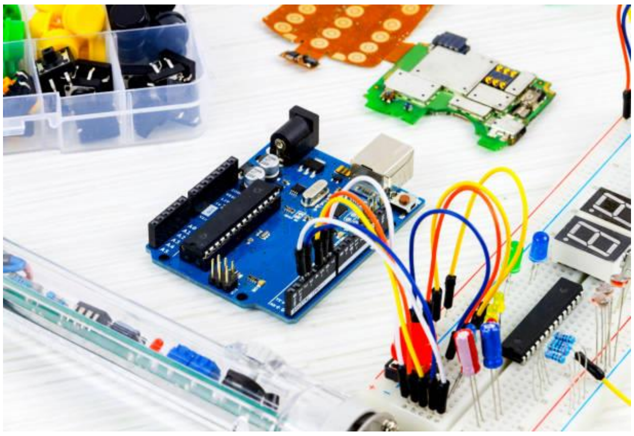

Figure 2.9: Hardware connection

2.3.6 Software Implementation in Home Automation

The demonstration below shows how to automate two machines using the Internet of Things in home automation system (IOT). The microcontroller is an Arduino UNO R3 [10]. This endeavour involves two types of communications: wired and remote correspondence. Internet of Things is used to control the computer through remote communication [9]. The Eclipse SDK includes Java and CVS support, compatibility with other modules supported by third-party plug-ins. Anything in Eclipse is plug-in, except in particular case of a minimal run-time kernel. As a result, every plug-in built interacts with Eclipse in the same manner that other plug-ins do; all features are "made equal" in this regard. The Eclipse SDK contains the Eclipse Java development tools (JDT), which provide an integrated development environment (IDE) with a built-in Java incremental compiler and a complete model of Java source data. Advanced refactoring methods and data interpretation are now possible [7].

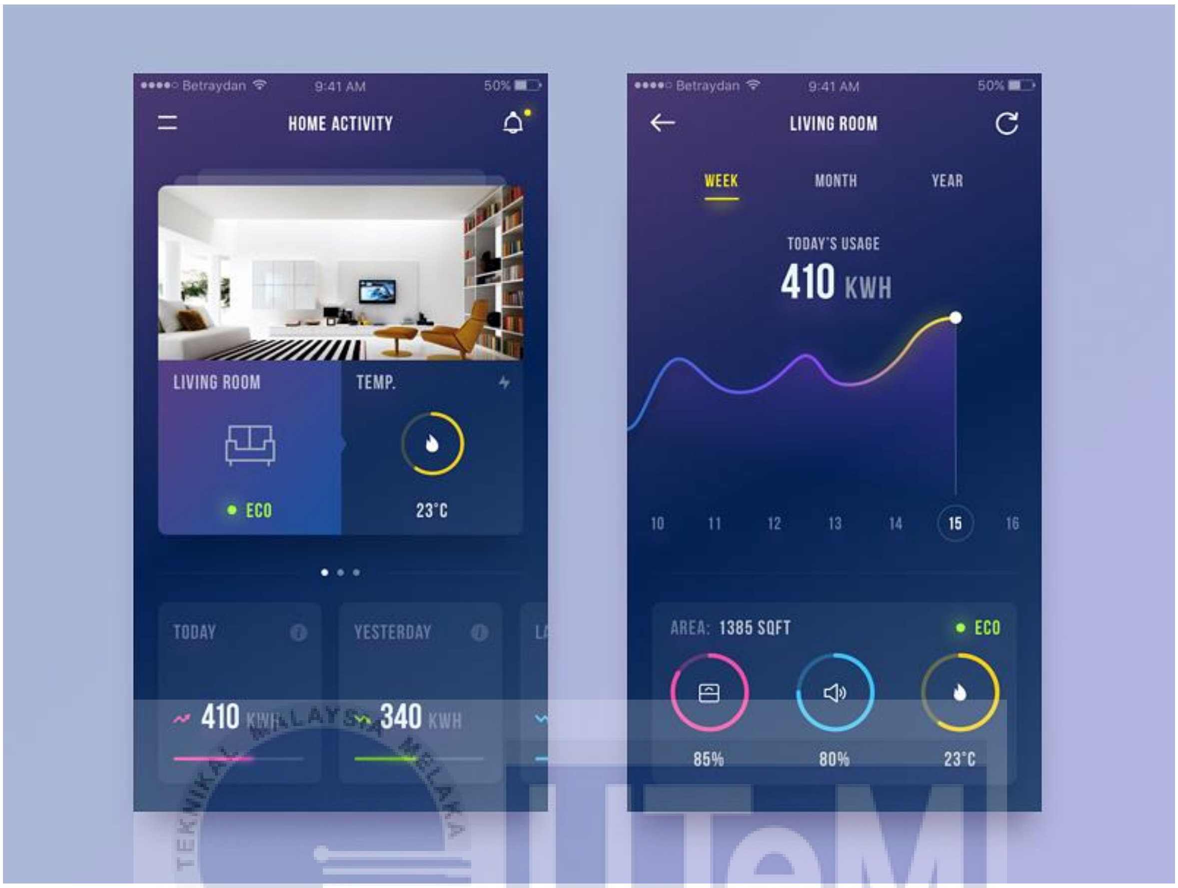

Figure 2.10: User Interface[7]

This project proposes and implements a novel framework for a low-cost and scalable home management and surveillance framework using Android-based smart phones. To access and supervise home devices, any Android-based mobile phone with built-in Wi-Fi capability may be utilized. Where Wi-Fi is unavailable, mobile cellular networks such as 3G or 4G may be used instead, without the requirement for an additional speech recognition module [7].

2.4 Summary

The primary goal of this technology is to automate the control of household appliances such as lights, fans, doors, and air conditioners. Savings, security, ease, and power are only a couple of the advantages of home automation. Certain clients frequently acquire home automation for the reasons of convenience and peace of mind. Home automation techniques based on Arduino, GSM, and Android have been applied to make it easier for people to monitor their appliances. Different home automation strategies using Arduino, GSM, and Android are presented along with their architecture, implementation, and flowcharts, allowing for a clear understanding of their advantages and disadvantages. This allows consumers to automate their homes without having to purchase costly smart appliances. Many improvements can be made to this initiative, such as installing a motor to power window drapes, a fire sensor to avoid mishaps, and so on.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discusses the methodology used and the factors taken into consideration. The development of the main controller circuit, which includes hardware devices and software settings, provides the basis for the whole system design, is part of the intelligent home automation system development process. It started with a description of the project process, then moved on to the methods, approaches, and tools used in system development. Both hardware and software will be used in the creation of this system. The overall workflow is as shown in the Figure 3.1 below.

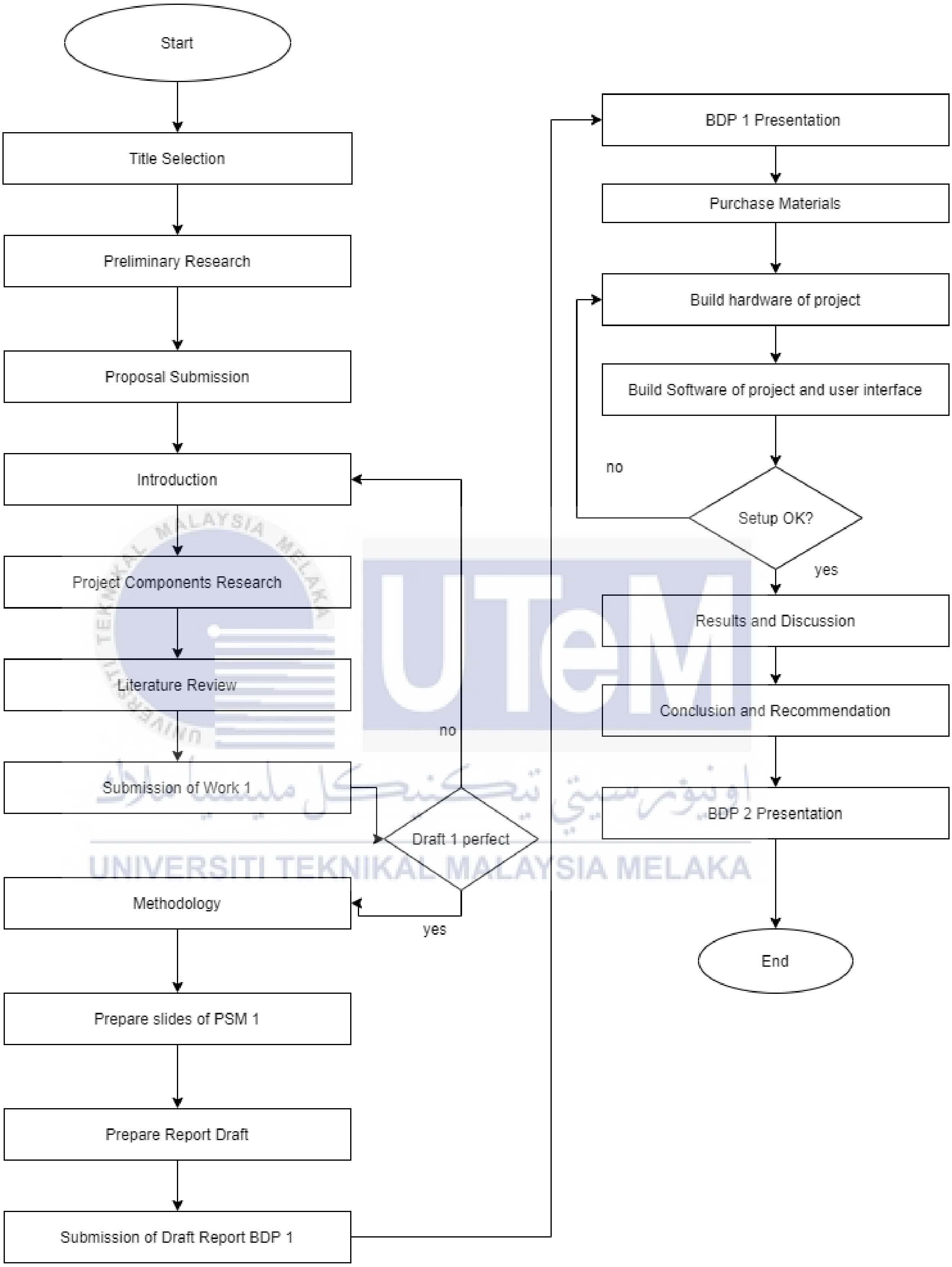

Figure 3.1 Overall Project Flowchart

3.2 Methodology

In this project, Arduino UNO is the main microcontroller followed by Node MCU which acts as an open source IoT that connects to Wi-Fi module. The ground pin is connected to the ground pin on the various sensors. The smoke sensor and PIR sensor are all connected to the Arduino board's pins as well as provide input. The Arduino's digital pin provides input to the relay, which then conducts the desired action. The Node MCU is the connected to the Arduino UNO and to control the appliances. The relay then functions according to the motion sensor as ON or OFF. These appliances are connected via Relay to the microcontroller. When the signal is passed to the Node MCU from the application, user should be connected to internet to control this system. The below image that represents the flow of the proposed system. The figure 3.3 represents the block diagram of the project.

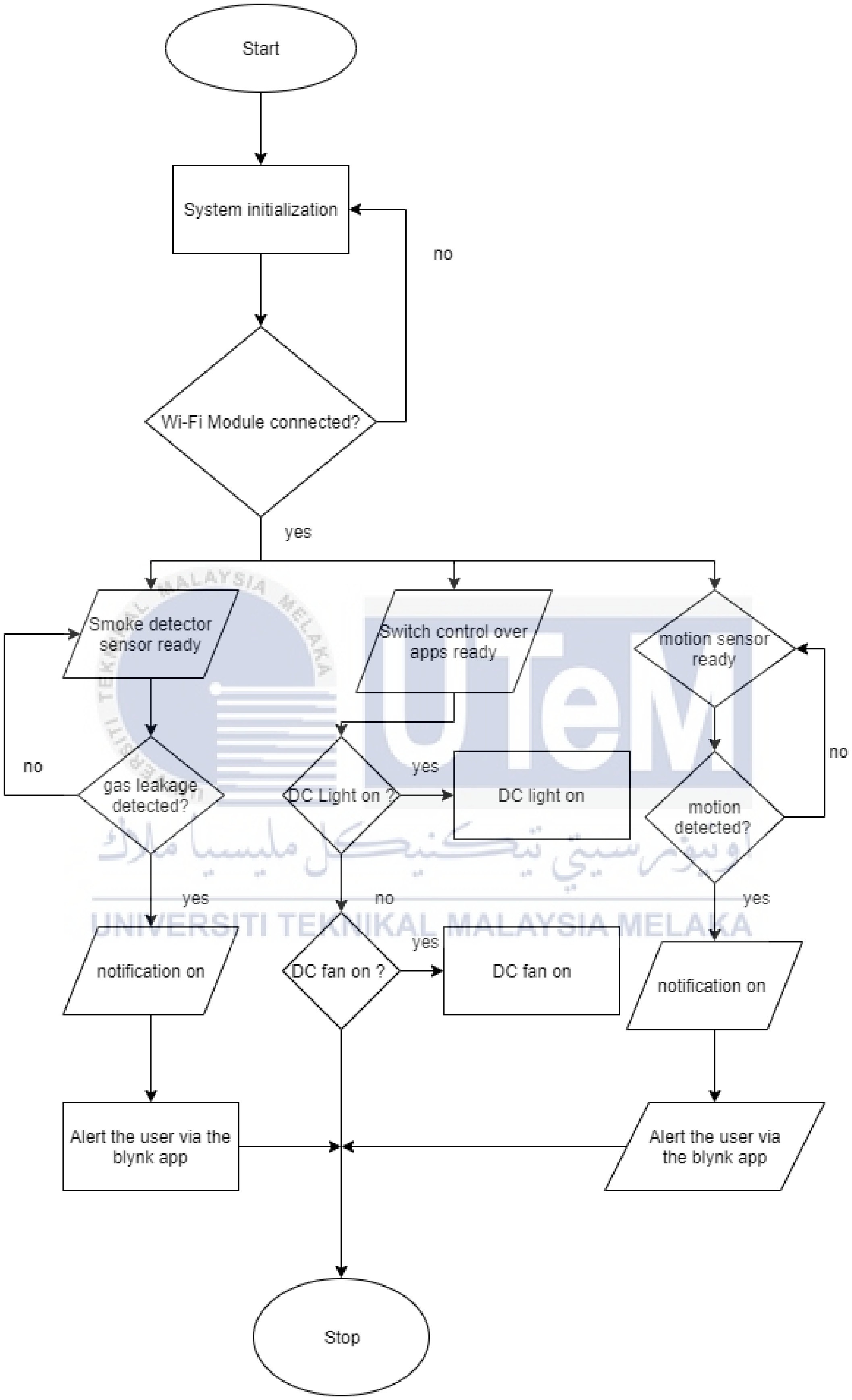

Figure 3.2 Flowchart of the Smart Home System

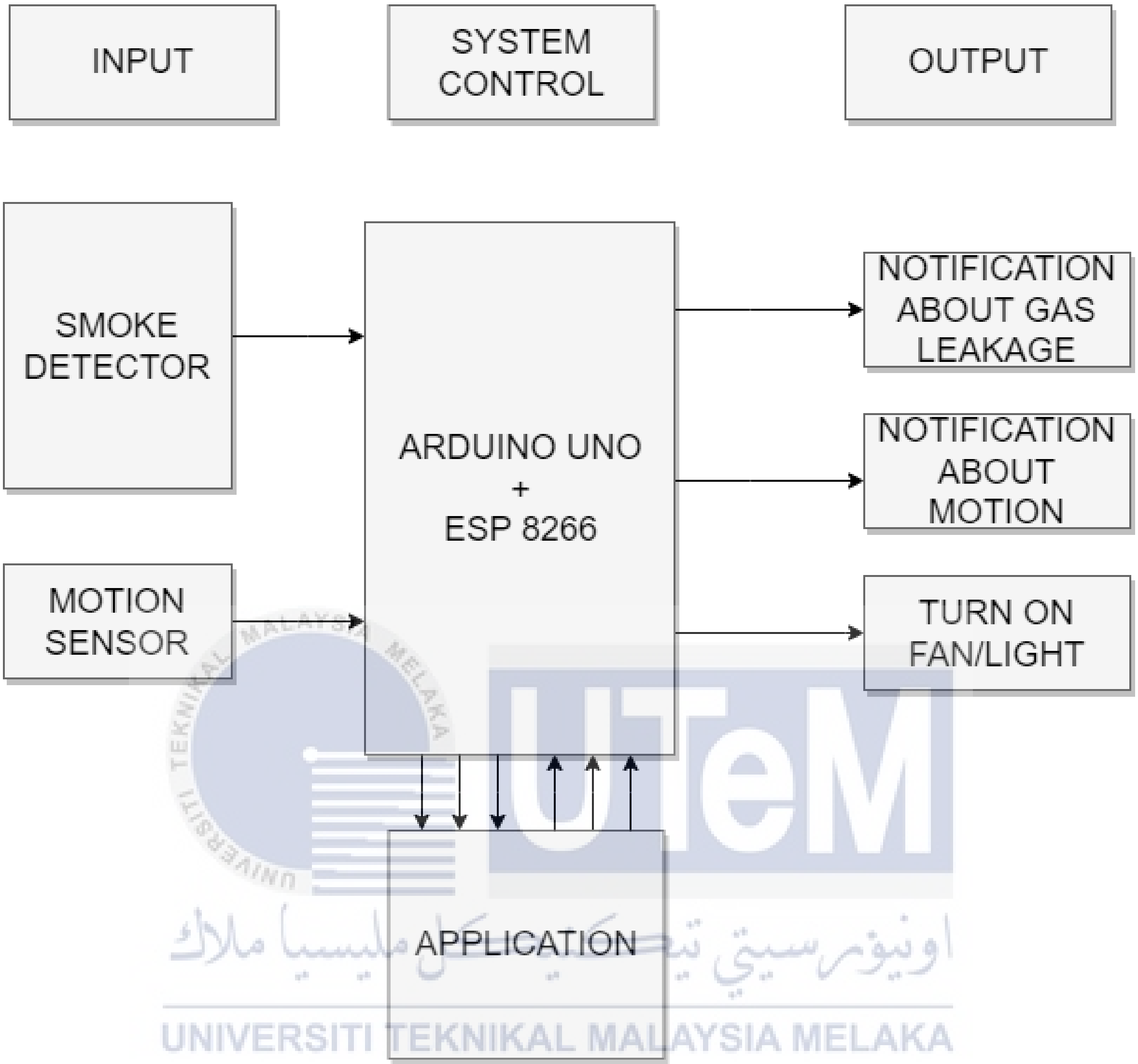

Figure 3.3 Block Diagram of the Smart Home System

3.3 Hardware Specification

There are combination of hardware and software used in this project. As for hardware relays, Arduino UNO the main microcontroller, Node MCU which is the Wi-Fi module, lights, fan, relay, PIR sensor and MQ-2 smoke sensor. Then, for software the Arduino IDE and ESP 8266 module is used to make the connection between the hardware components to the interface via IoT.

3.3.1 Arduino UNO

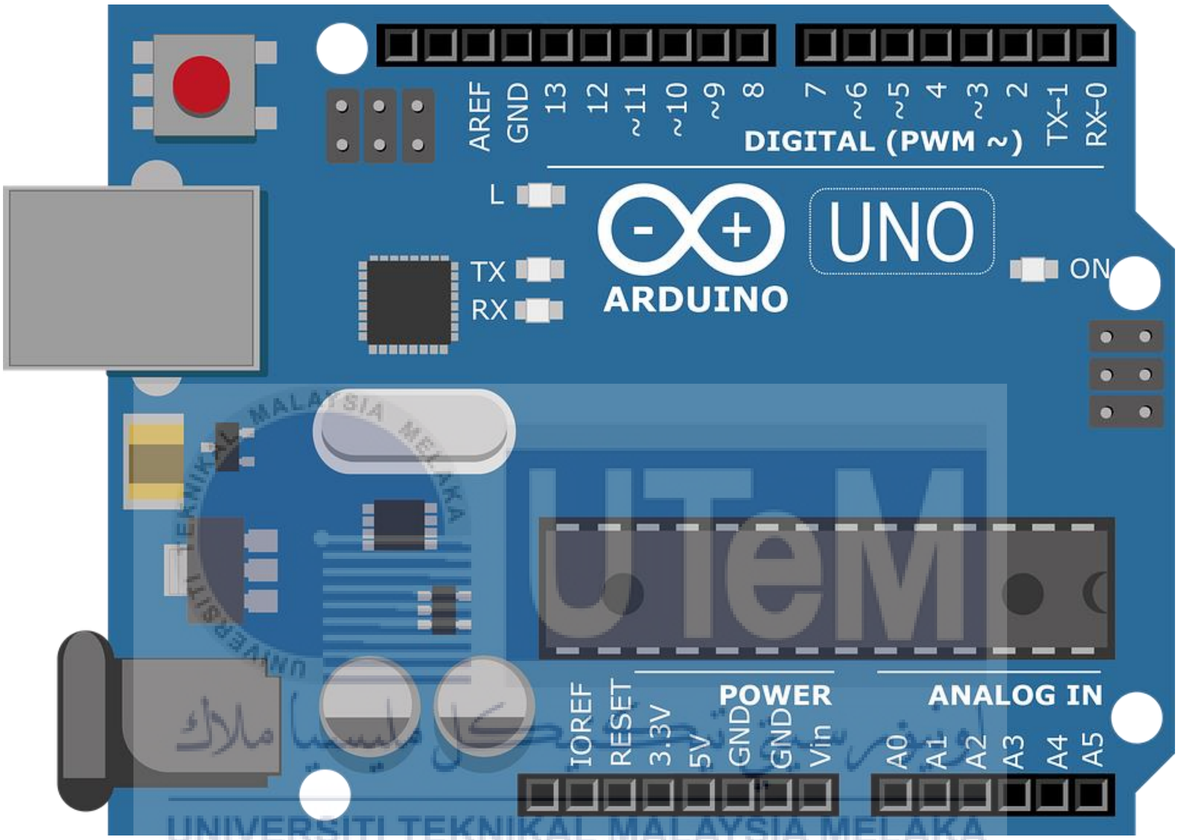

Figure 3.4 Arduino UNO board

Arduino can also be used to create a natural interface, receive commitments from a variety of switches and sensors, and control the output from a variety of physical contraptions such as leds' and various mechanical components. For this project Arduino UNO board with the listed specification was chosen as it is works well with all types of Operating system such as Linux, Windows, Apple and many others. Besides that, it also uses a basic coding which is easy to be understood and make amendments right away using Arduino IDE. Also has variety of sample coding's which can be used. The specifications of Arduino UNO board are as shown in the below figure.

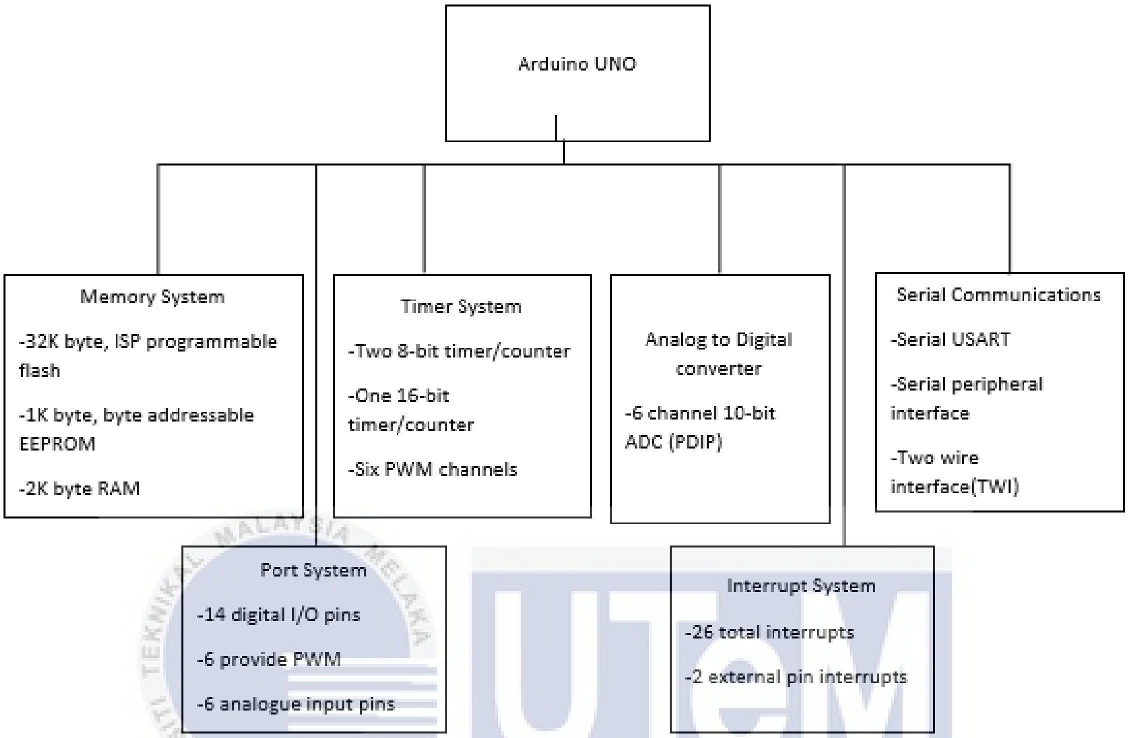

Figure 3.5 Features in Arduino UNO

3.3.2 Node MCU

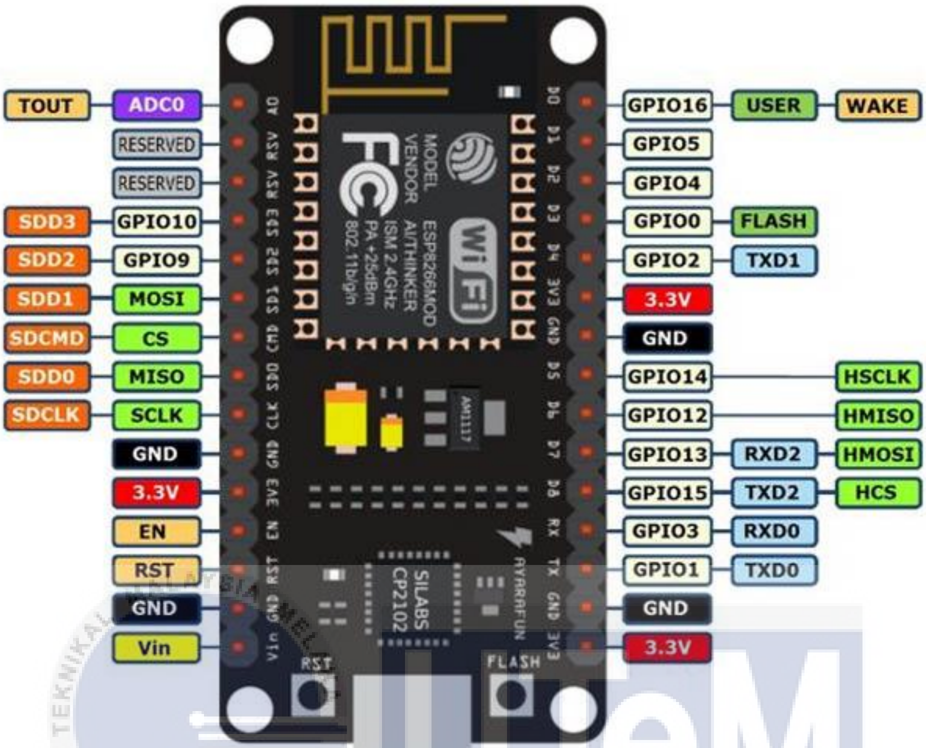

Figure 3.6 Node MCU board

Table 3.1 : ESP8266 / Node MCU Pin Configuration

Pin Category	Name	Description
Power	Micro-USB, 3.3V, Vin, GND	**Micro-USB**: NodeMCU can be powered through the USB port **3.3V**: Regulated 3.3V can be supplied to this pin to power the board **GND**: Ground pins **Vin**: External Power Supply
Control	EN, RST	The pin and the button resets the microcontroller

Analog	A0	Used to measure the analog voltage in the range of 0-3.3V
GPIO	GPIO1-GPIO6	NodeMCU which has 16 general uses input – output pins on this board
SPI	SD1, CMD, SD0, CLK	NodeMCU has 4 pins which is used for SPI communication
UART	TXD0, RXD0, TXD2, RXD2	NodeMCU has 2 UART interfaces which are UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 to upload the progam
12C pins		Functionality support but due to the internal functionality of these pins, the pin 12C can be self-detected

Node MCU is an open source IoT platform. It is a microcontroller used as Wi-Fi module for user interface to be connected to the hardware peripherals. This module has a very powerful processor and also storage capability which enables it to be connected with all the components thus connects to user interface. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existence interfaces thus it also has a self-calibrated Radio Frequency that allows it to function in any environment, and demands no extra RF components.

3.3.3 Sensors

Sensors is a device that detects senses such as heat, moisture, pressure, motion, smoke and light. The output of this sensor is transferred to the via an interface as human readable display. It helps to indicate when there is no presence of human to alert on the hazard. Thus, in this project 2 types of sensors' are used which are MQ-2 and PIR sensor.

A PIR sensor captures variations in the quantity of infrared radiation intruding on it, which varies based on the temperature and surface properties of the objects in front of it. The temperature at that location in the sensor's field of vision will rise from room temperature to body temperature when an item, such as a human, pass in front of the backdrop, such as a walls. Objects with matching temperatures but varying surface properties may emit infrared light in various patterns, triggering the detector when moved in relation to the backdrop. In this project, it is used to detect motion for presence of humans when there is the lights and fans turn on automatically. It turns off when there is no presence of human.

Next, the MQ-2 sensor identify the level of gases such as methane, butane, LPG, and smoke in the air, but it is unable to discriminate among them. As a nutshell, there is no way of knowing which gas it is. This sensor is often used in hospitals to analyze air quality, detect gas leaks, and manage environmental regulations. These are used in enterprises to monitor the leaking of dangerous gases. In this project it used to detect gas leakage and alert the user.

Figure 3.7 PIR sensor

Figure 3.9 MQ-2 sensor

3.4 Software Specification

The software specifications used to connect the hardware peripherals to the user interface. In this project Arduino IDE is used to be connected to ESP 8266 Node MCU (Wi-Fi Module) to be connected to the user interface. The coding language used in Arduino IDE is C++ to program the hardware peripherals which are connected to the Arduino UNO board. Upon programmed the hardware peripherals will be able to function. For ESP8266 Node MCU which is connected to the Arduino UNO board will also be programmed by Arduino IDE software. Besides that, for this project Android Studio Application is used to create user interface application. The home control network for the internet of things uses the Blynk program to build apps that can track the device. Blynk is an iot device platform that allows users to operate electronic equipment wirelessly via iOS and Android apps. It features a screen from which the user can develop graphical interfaces using various widgets. Sensor data can also be stored and displayed by Blynk. If the library is linked, it simply needs to create a sketch in Arduino UNO [22].

3.5 System Development

The “Smart Home System” project planning is done accordingly upon the theory studies and coding implementation which is done. The two most important peripherals in the development of this project are Arduino UNO board which acts as the main microcontroller followed by Node MCU which will be connecting the hardware peripherals to the user interface. With the collaboration of all these peripherals and the sensors this project will succeed.

3.6 Testing

Once the project is done, the creator will inspect it for defects and drawbacks. When there are such faults detected it has to go through troubleshooting process in order to make the system work. Each component has its own function and plays an important role in this system thus the errors and faults will be detected accurately. By doing this the quality of this system is promised. This technique is very important, and it will be efficient for end user to use it with no errors.

3.7 Maintenance

It is very important to meet the requirements of the project and to achieve this system maintenance needs to be verified. So that it can perform effectively, the maintenance element is the utmost component of the project's development. Thus, the support must be offered on a regular basis, as the problem may not manifest itself quickly and plainly.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter discuss about the expected outcome and the results of this "Development of Smart Home System Using Arduino". This proposed system was to ease the user to control the electrical appliances at home and alert on the gas leakage. The results and discussion of throughout the project is recorded as below.

4.2 Circuit Analysis

4.2.1 Arduino UNO

The hardware design of the microcontroller which is also known as the brain of the system used in this design is Arduino UNO. The other hardware peripherals are then connected to the respective pins. MQ-2 is used as the smoke detector in this project to detect the gas leaking. The main electrical appliances such as light and fan are connected to the power source using relay. Then the sensors and ESP8266 are connected to the microcontroller. The circuit was designed in Proteus and recorded as below in Figure 4.1.

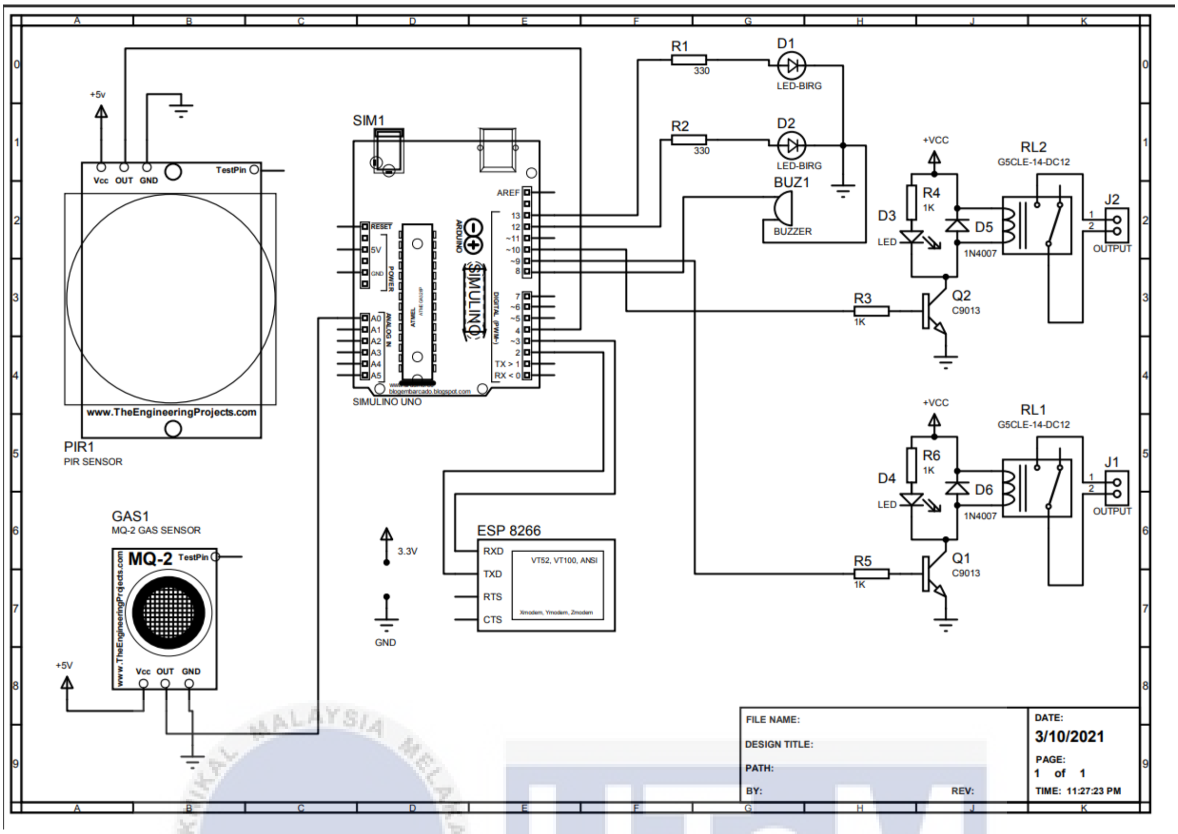

Figure 4.1 Schematic Diagram of the project

4.2.2 ESP8266 Wi-Fi Module

The ESP8266 Wifi Module, which features WIFI capability, analogue pins, digital pins, and serial communication protocols, is the core component of this project. This module's maximum supply voltage is 5V. The below diagram 4.2 shows the Wi-Fi Module which connected to the microcontroller Arduino UNO.

Figure 4.2 ESP8266 Wi-Fi Module

4.2.3 Circuit Connection

In this project the circuit was designed first upon the schematic diagram completed. To create the circuit the inputs were connected first to the microcontroller which is the Arduino UNO board. The connections were done according to the designated pins for each sensors. Relay is usually used to do the automatic adjustment, safety protection, and conversion circuit in the circuit. In this project it used to convert the alternating current from the direct power supply to the light and fan which uses direct current. The circuit build for the connection of input is as shown in the Figure 4.3 below. The MQ-2 sensor is used as the smoke sensor and PIR sensor is to check the movement around which is then observed through the application developed. The outputs were connected through the relay which are the light bulb and fan which is used to create the prototype as shown in the below Figure 4.4.

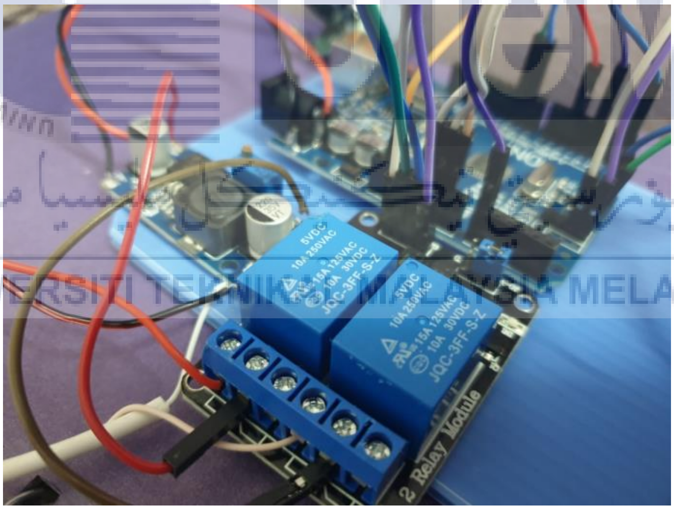

Figure 4.3 The circuit connection of the project

Figure 4.4 Connection of the output to the Microcontroller

4.3 Software Analysis

The Arduino IDE is the software that was used in this project to program the components with the appropriate code that was produced to ensure the system ran properly as well. The Arduino IDE is an open-source platform that may be used to create a diversified combination of both hardware and software for this project. The Arduino IDE allows the user to write code and then upload it to the microcontroller, which will run it while interacting with inputs and outputs such as the ESP8266, smoke detector, and motion detector. To execute the code successfully, libraries such as ESP8266, Blynk, and software serial must be installed. The libraries and board are essential for running the project coding without error messages and moving on to the next procedure. Sensor data may be seen in real time. IoT insights can be done automatically in response to certain schedules or events. [20]

4.3.1 ESP-8266 WiFi Module

The ESP8266 is linked to a local WiFi hotspot which in this case to the laptop’s hotspot, allowing it to connect to the internet and send data to the Blynk server, along with the authentication code. This component has robust enough on-board computing and storage to allow it to be coupled with sensors and other application-specific devices via its general-purpose input/outputs (GPIOs) with little programming and loading during runtime. Because of its great degree of on-chip integration, it requires very little external circuitry [21]. An authentication code is then transmitted to the app, which has the same authentication code, and the data is received by the app to establish a secure connection between the app and the

ESP8266. The hotspot name and password is saved in the coding created to connect to blynk application in the Arduino IDE as shown in the Figure 4.4 below.

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleStream.h>

// Your WiFi credentials.
// Set password to "" for open networks.
const char* ssid = "smarthome";
const char* pass = "12345678";

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "sZCld4jogUt1FhgLLhBr-tOgQR_vlH0Z";
char server[] = "139.59.206.133";
```

Figure 4.5 WiFi setup code

```
WiFiClient wifiClient;

// This function tries to connect to the cloud using TCP
bool connectBlynk()
{
  wifiClient.stop();
 // return wifiClient.connect(BLYNK_DEFAULT_DOMAIN, BLYNK_DEFAULT_PORT);
  return wifiClient.connect(server, BLYNK_DEFAULT_PORT);
}

// This function tries to connect to your WiFi network
void connectWiFi()
{
  Serial.print("Connecting to ");
  Serial.println(ssid);

  if (pass && strlen(pass)) {
    WiFi.begin((char*)ssid, (char*)pass);
  } else {
    WiFi.begin((char*)ssid);
  }

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

Figure 4.6 Coding for the WiFi connection with Blynk Application

4.3.2 Connection of Smoke Sensor

The function of smoke sensor is controlled using this code as below to record the reading of the smoke value whereby it is set to more than 200 the alarm will be rang and notify the user. When it exceeds over 150 it will send notification to the user. The code is used as below in Figure 4.6.

```
TimerWifi++;
if (TimerWifi>4){
  TimerWifi=0;
   ss.print("*");
 ss.print(Smoke,1);
  ss.print("*");
 ss.print(Motion);
 ss.print("#");
 Serial.println("Update Wifi..");
}


if (Smoke>200){
  if (ALARM==0){
  ALARM=1;
      ss.println("Y");
      digitalWrite(ledR,HIGH);
      digitalWrite(ledG,LOW);
  }
}
if (Smoke<150){
  ALARM=0;
}

if (ALARM==1){
  digitalWrite(BUZZ,HIGH);
  delay(200);
    digitalWrite(BUZZ,LOW);
  delay(20);

}
```


Figure 4.7 Coding for smoke sensor

4.3.3 Connection of Motion Sensor

The function of motion sensor is controlled using this code as below to record the reading of the motion value whereby it is set to PH > 7 which is the ph of the human body. Thus when it is detects motion it will notify the user. The PIR sensor is connected to the Pin 4 of the Arduino UNO board as declared in the coding. The code is used as below in Figure 4.7.

```
#define FAN 9
#define LIGHT 10
#define ledG 13
#define ledR 12
#define BUZZ 8
#define PIR 4


SoftwareSerial ss(2, 3); //(RX,TX)
int ALM1=0;
int PIRStat=0;
float Smoke;
float Sens1;
String Motion="NORMAL";


 if (digitalRead(PIR)==1){
 PIRStat=100;
 Motion="DETECTED!";
 if (ALM1==0){
  ALM1=1;
  ss.println("X");
  delay(1000);
 }
 }
  if (digitalRead(PIR)==0){
    Motion="NORMAL";
 PIRStat=0;
 ALM1=0;
 }
```

Figure 4.8 Coding for motion sensor

4.3.4 Connection of Light and Fan

The dc light and fan is connected to the microcontroller using the relay, in this code it shows that the light is connected to pin 10 and the fan is connected to pin 9 of the Arduino UNO. It also shows that the blynk input value is set to V10 and V11 respectively in the Wi-Module coding as shown in below Figure 4.8.

```
#include <dummy.h>


#include <SoftwareSerial.h>


#define FAN 9
#define LIGHT 10
#define ledG 13
#define ledR 12
#define BUZZ 8
#define PIR 4


 while (Serial.available()) {
  // get the new byte:
  char inChar = (char)Serial.read();


  // if the incoming character is a newline, set a flag
  // so the main loop can do something about it:

   if (inChar == '1') {
      digitalWrite(FAN,LOW);
  }
  if (inChar == '2') {
      digitalWrite(FAN,HIGH);
  }
       if (inChar == '3') {
      digitalWrite(LIGHT,LOW);
  }
  if (inChar == '4') {
      digitalWrite(LIGHT,HIGH);
  }


}
BLYNK_WRITE(V10)
{
  int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable
  Rly1=pinValue;
  // process received value
}
BLYNK_WRITE(V11)
{
  int pinValue1 = param.asInt(); // assigning incoming value from pin V1 to a variable
  Rly2=pinValue1;
  // process received value
}
```


Figure 4.9 Coding to control the switch of light and fan

4.3.5 Blynk Application

This Blynk Application is used as cloud in this project to connect the hardware peripherals to the software through WiFi using ESP8266. The data from the Blynk app is transmitted to the Blynk server along with the authentication code, and because the ESP8266 already has the same authentication code as the Blynk server, the server recognises the matching code and data is transferred to the associated ESP8266 as shown in Figure 4.10. The user interface is as shown in figure 4.11.

```
void setup()
{

  Serial.begin(9600);

  connectWiFi();

  connectBlynk();

  Blynk.begin(wifiClient, auth);

}

void loop()
{
WidgetLED led1(V1);

  // Reconnect WiFi
  if (WiFi.status() != WL_CONNECTED) {
    connectWiFi();
    return;
  }

  // Reconnect to Blynk Cloud
  if (!wifiClient.connected()) {
    connectBlynk();
    return;
  }
```


Figure 4.10 Coding to setup Blynk Application in Wi-Fi module

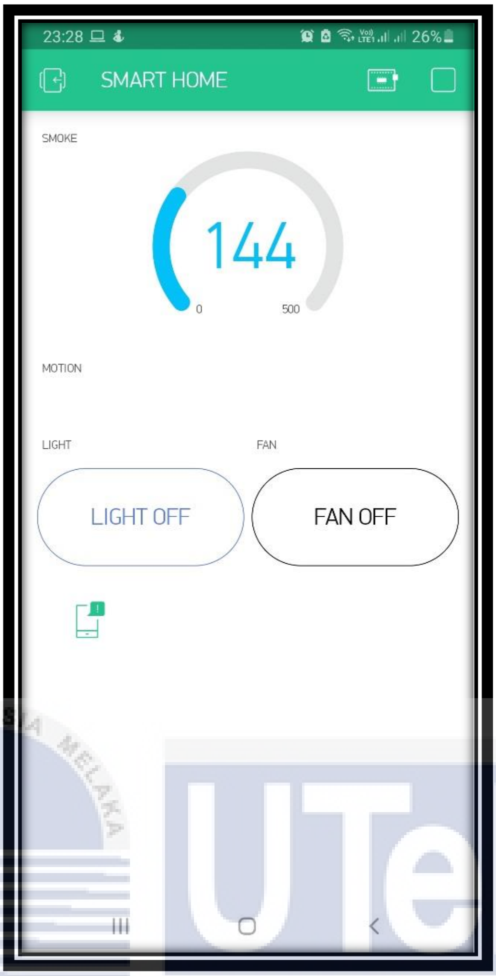

Figure 4.11 The blynk application User Interface

4.4 Hardware Analysis

4.4.1 Protoype Results

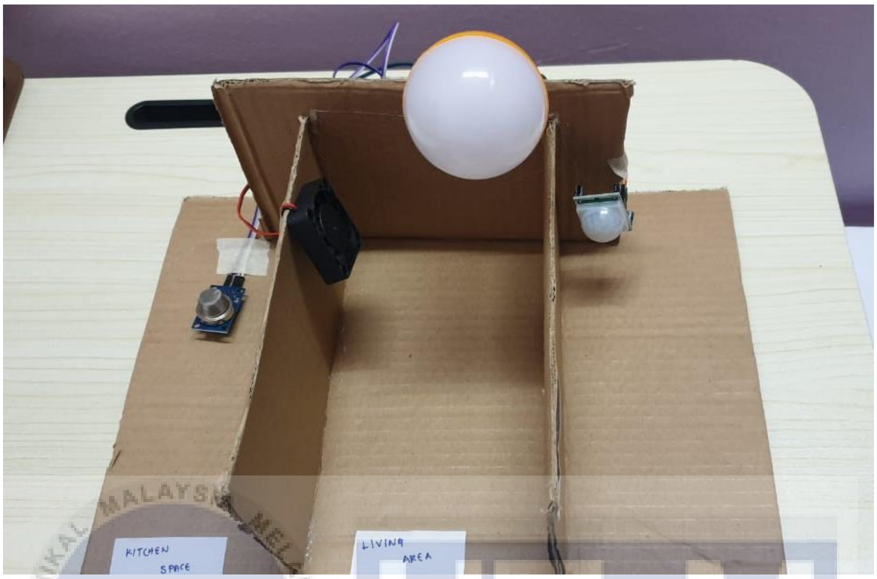

Figure 4.12 Top view of the prototype

Figure 4.13 Side view of the prototype

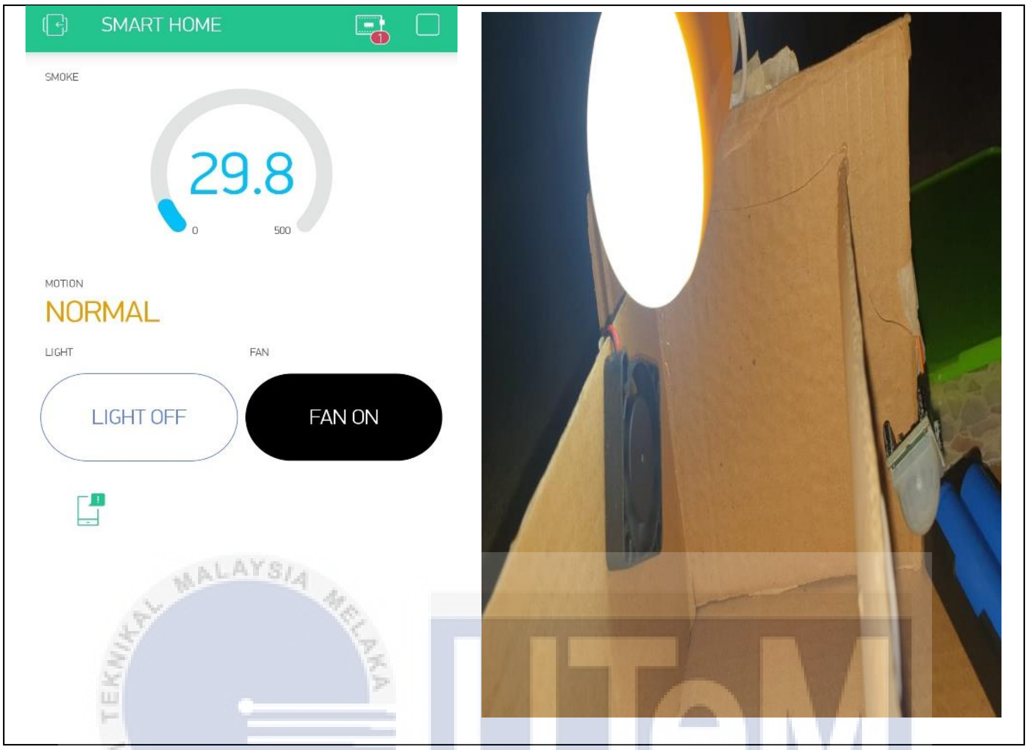

Figure 4.14 Blynk interface used to turn on DC light

Figure 4.14 shows that the condition of DC Bulb turns ON / OFF when the user press the button features on the Blynk application to control the home appliances.

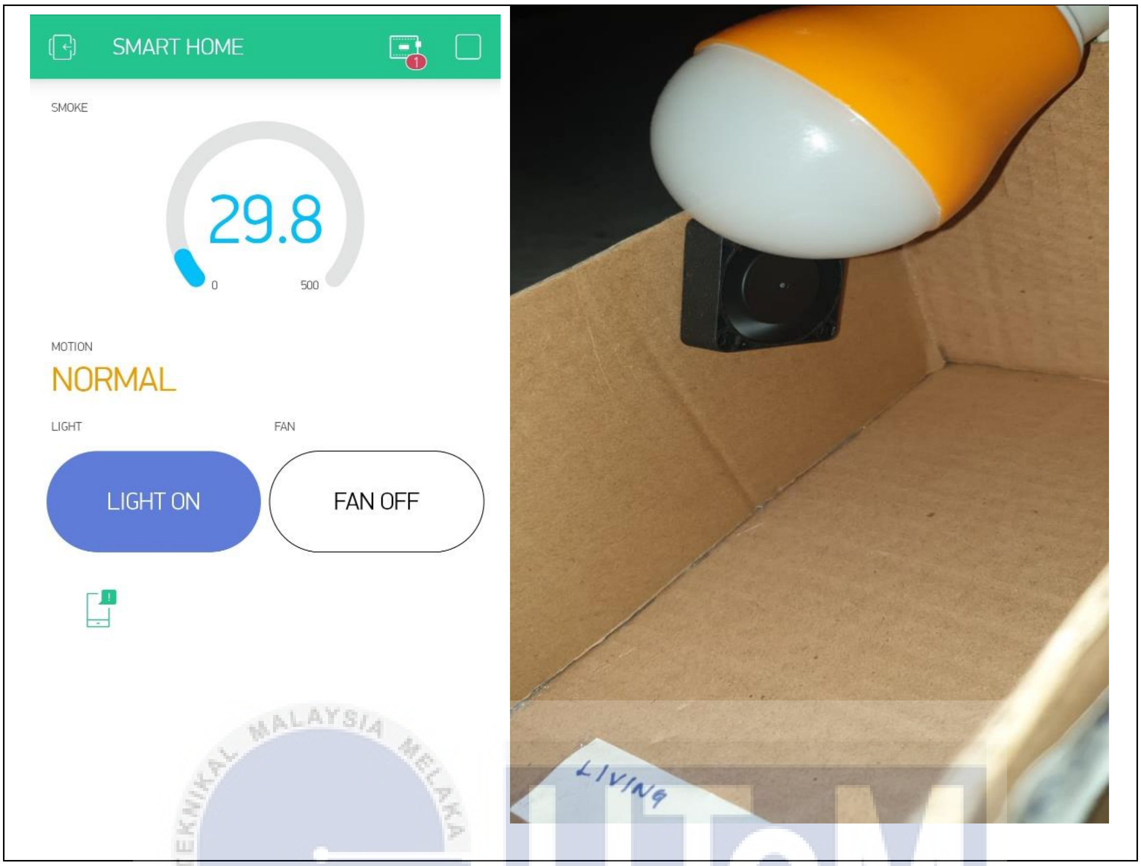

Figure 4.15 Blynk interface used to turn on DC fan

Figure 4.15 shows that the condition of DC fan turn ON / OFF when the user press the button features on the Blynk application to control the home appliances.

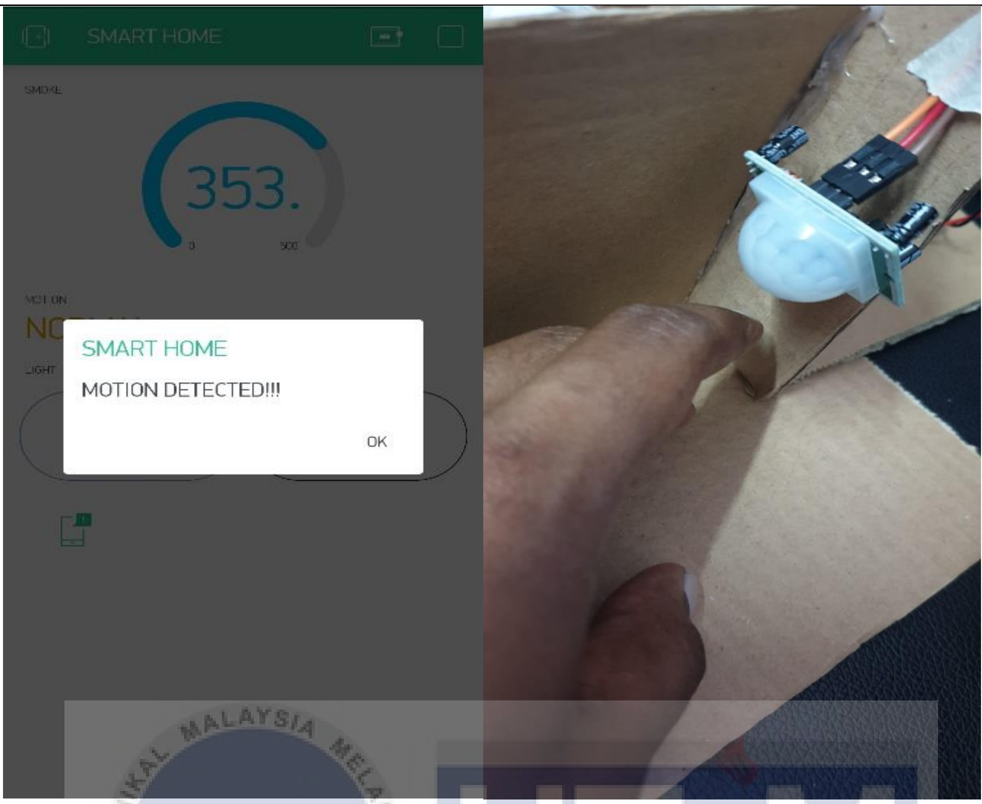

Figure 4.16 Blynk interface and motion sensor

Figure 4.16 shows the condition of PIR Sensor when there is human movement detected. When there is no human presence detected, the user can make sure their home free from intruder.

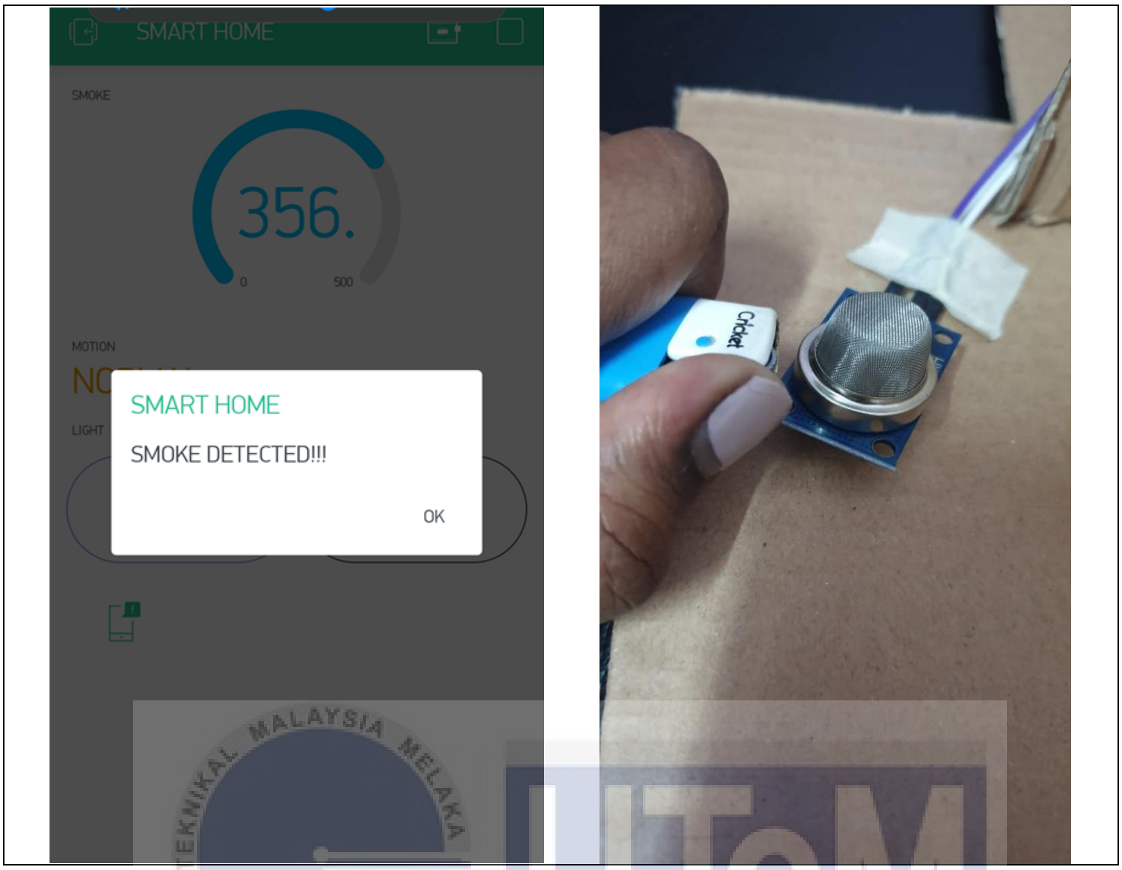

Figure 4.17 Blynk interface and MQ-2 sensor (optimum temperature)

Figure 4.17 shows the condition of MQ-2 Smoke Sensor when in optimum temperature. Since, there was no any smoke detected, the user can make sure their home free from fire or any unexpected incident occur.

4.4.2 ESP8266 Malfunction

The ESP8266 which acts as the Wi-Fi module had fatal errors due to hardware malfunctionality. This happened due to some hardware component failure in the Wi-Fi module. The codes were not able to be uploaded to the Wi-Fi module upon refreshing the authentication code. The figure 4.18 below shows the error which was reflected when uploading the new code to the Wi-Fi module ESP8266. To overcome this many methods were tried firstly started by refreshing the authentication code in the Blynk application and editting the authentication. Next, the hardware wiring were cross checked to ensure there is no cable faulty which connects the ESP8266 to the Wi-Fi Module. Upon checking these two the coding was uploaded to the ESP8266 Wi-Fi module using Arduino UNO. It resulted in error as well as recorded in the Figure 4.18 below.

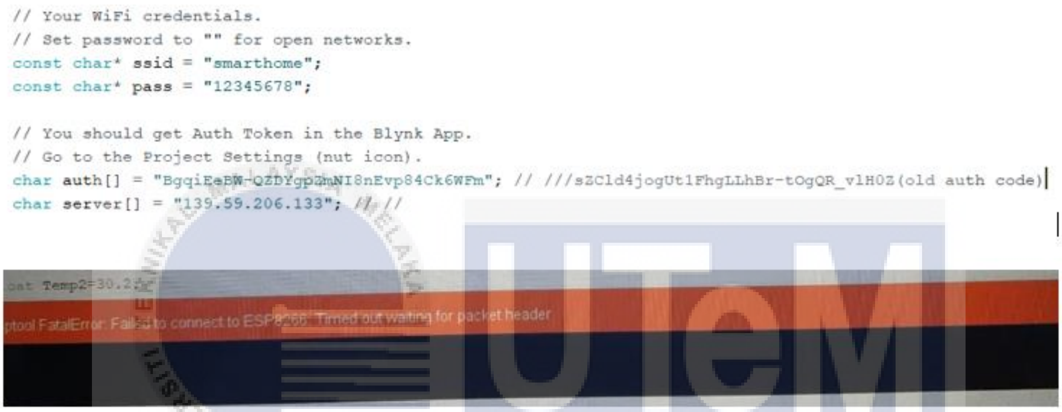

Figure 4.18 Failure to upload code to the Wi-Fi module

CHAPTER 5

CONCLUSION

5.1 Conclusion

In this project the main focus is to develop a technology which provides a safe and also cost saving home system. A home is a living place for many, and it is an important asset that needs to be well preserved. The flexibility in the control of the designed smart phone is widely used all over the world with the control of Internet. A wide range of sensors which are used to capture the motion, humidity and gas leakage results in an efficient safety feature with ease of control of the user. This project uses affordable components and compatible software to be controlled by the user. Additionally, the software application is embedded using Android which is most profound in most of the latest smartphones and has open source which tags the smartphones to be affordable too. The initial design was done based on the studies done theoretically. Lastly, this project can be used prevent accidents at home and cost saving.

5.2 Future Work

This section will go through the project's limitations as well as future recommendations. To begin, my project relied heavily on a 5V rechargeable battery. It is the smallest possible input value. As a result, in order to address this difficulty in the future, this project must include a 24V rechargeable battery as a backup. Next, in this project the Blynk application is setup over a Wi-Fi connection with the username and password which was set. This will not be sufficient if the user has less resources to connect to the particular network. To overcome this, it can be set to work over mobile data as it will be convenient for the user. Besides that, there is no surveillance camera added to this project the future developer may add this feature to enable the user to monitor and control of the guests entering their home. This way it is saver and less troublesome to the owner. Lastly, solar energy is also recommended in this project because it is a totally renewable resource. It also requires relatively minimal maintenance to keep solar cells operating. Solar panels and solar lighting may appear to be expensive when initially purchased, but this may save a lot of money in the long term because it provides 70% of the energy.

REFERENCES

[1] Masi, Faiz & Husain, Kurban. (2019), “Smart Automation Based On IoT and GSM Module.”

[2] Ilyas Baig, Chiktay Muzamil, Salahuddin Dalvi. (2016), “Home automation using Arduino Wi-Fi module ESP 8266”

[3] Nadeem Pasha, Varun Koushik K. M, Shahbaaz Ahmed P, Sufia Begum Shaikh. (2017), “Smart home automation and security system using GSM and ARM7”, International Journal Of Engineering Research & Technology (Ijert) Nconpcs –(Volume 5 – Issue 18)

[4] Ms. Pawar Pallavi Tatyasaheb, Mr. B.E Shinde. (2016), “Review on Home Automation System Using Different Techniques.”

[5] Prof. Vikram M.Kakade. (2017), “Home Automation System using PIR Sensor”.

[6] J.Chandramohan, R.Nagarajan, K.Satheeshkumar, N.Ajithkumar,P.A.Gopinath, S.Ranjithkumar. (2017, March 3), “Smart Home Automation and Security System using Arduino and IOT.”

[7] Gunasekaran, Mahalalakshmi. (2017), “IOT based home automation using Arduino.” International Journal of Engineering and Advanced Research Technology.

[8] J.Selvin Paul Peter, S. Selvakumar, Henum Pandit, Pratul Aggarwal. (2019, May), “Home automation and home security using Arduino and ESP8266 (IOT)”.

[9] Akrit Kapoor (151426) Ujwal Dogra (151408). (2018), “Wireless home automation system.”

[10] Lalit Mohan Satapathy, Samir Kumar Bastia Nihar Mohanty, (2018), “Arduino based home automation using Internet of things (IoT)”

[11] Heetae Yang, Wonji Lee, Hwansoo Lee. (2018), “IoT Smart Home Adoption: The Importance of Proper Level Automation”

[12] Davies, E.I & Anireh, Vincent. (2019), “Design and Implementation of Smart Home System Using Internet of Things.” Advances in Multidisciplinary & Scientific Research Journal Publication 7

[13] Gunge, V. S. & Yalagi P.S. (2016), “Smart Home Automation: A Literature Review, International Journal of Computer Applications”

[14] Arvindhan, R., Ramnathan, M., Sanjayumar, D., & Kishore, R. (2017), “Home Automation Using Wi-Fi Interconnection.”

[15] Mehta, H., Jadhav, K., Mishra, A., & Deshmukh, A. (2017), “IoT Based Home Automation System Using Arduino Board.”

[16] Rasika Hedaoo, P R Morey. (2016), “Auto-Fact Security System. International Journal for Innovative Research in Science & Technology.”

[17] Kunal Kumar, Navneet Sen, Sheikh Azid, Utkal Mehta. (2016), “A Fuzzy Decision in Smart Fire and Home Security System”

[18] N. Vikram, K. S. Harish, M. S. Nihaal, R. Umesh, A. Shetty and A. Kumar. (2017), "A Low Cost Home Automation System Using Wi-Fi Based Wireless Sensor Network Incorporating Internet of Things (IoT),"

[19] H. Singh, V. Pallagani, V. Khandelwal and U. Venkanna. (2018), "IoT based smart home automation system using sensor node,"

[20] Md. Rakib Ahsan , Sheikh Zarif Ahmad , Mohammad Shamsul Arefin , Md.Aminul Bari. (June 2019), “Implementation of IOT based Smart Security and Home Automation System”, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 08, Issue 06

[21] Tae-Gue Oh, Chung-Hyuk Yim, Gyu-Sik Kim. (2017), "ESP8266 Wi-Fi Module For Monitoring System Application".

[22] Homera Durani, Mitul Sheth, Madhuri Vaghasia, Shyam Kotech. (2019, May 8), "Smart Automated Home Application using IoT with Blynk App".

APPENDICES

Appendix A

Codes of the project

```
#include <dummy.h>
#include <SoftwareSerial.h>
#define FAN 9
#define LIGHT 10
#define ledG 13
#define ledR 12
#define BUZZ 8
#define PIR 4

SoftwareSerial ss(2, 3); //(RX,TX)
int ALM1 = 0;
int PIRStat = 0;
float Smoke;
float Sens1;
String Motion = "NORMAL";
int Sens1Pin = 0;
char datarelay[3];
int getdata = 0;
int datacount = 0;
String inputString[10];
int ALARM = 0;
int TimerWifi = 0;
```

```
void setup() {
  Serial.begin(9600);
  ss.begin(9600);
  pinMode(PIR, INPUT);
  pinMode(FAN, OUTPUT);
  pinMode(LIGHT, OUTPUT);
  pinMode(ledG, OUTPUT);
  pinMode(ledR, OUTPUT);

  pinMode(BUZZ, OUTPUT);
  digitalWrite(FAN, HIGH);
  digitalWrite(LIGHT, HIGH);
  digitalWrite(BUZZ, HIGH);
  delay(10);
  digitalWrite(BUZZ, LOW);
  delay(10);
  digitalWrite(BUZZ, HIGH);
  delay(10);
  digitalWrite(BUZZ, LOW);
  delay(10);
  digitalWrite(BUZZ, HIGH);
  delay(10);
  digitalWrite(BUZZ, LOW);
  delay(10);
  digitalWrite(BUZZ, HIGH);
  delay(10);
  digitalWrite(BUZZ, LOW);
  delay(10);
  delay(15000);

  // delay(10000);
}
```

```
void loop() {
  Sens1 = analogRead(A0);            //read the value from the sensor
  Sens1 = (5.0 * Sens1 * 100.0) / 1024.0; //convert the analog data to DC AC VOLTAGE
  Smoke = Sens1;

  if (digitalRead(PIR) == 1) {
    PIRStat = 100;
    Motion = "DETECTED!";
    if (ALM1 == 0) {
      ALM1 = 1;
      ss.println("X");
      delay(1000);
    }
  }
  if (digitalRead(PIR) == 0) {
    Motion = "NORMAL";
    PIRStat = 0;
    ALM1 = 0;
  }

  Serial.print(Smoke);
  Serial.print("\t");
  Serial.println(PIRStat);

  TimerWifi++;
  if (TimerWifi > 4) {
    TimerWifi = 0;
    ss.print("*");
    ss.print(Smoke, 1);
    ss.print("*");
    ss.print(Motion);
    ss.print("#");
    Serial.println("Update Wifi..");
  }
```

```
if (Smoke > 200) {
  if (ALARM == 0) {
    ALARM = 1;
    ss.println("Y");
    digitalWrite(ledR, HIGH);
    digitalWrite(ledG, LOW);
  }
}
if (Smoke < 150) {
  ALARM = 0;
}

if (ALARM == 1) {
  digitalWrite(BUZZ, HIGH);
  delay(200);
  digitalWrite(BUZZ, LOW);
  delay(20);

}
if (ALARM == 0) {
  digitalWrite(BUZZ, LOW);
  digitalWrite(ledR, LOW);
  digitalWrite(ledG, HIGH);

}
delay(500);

while (Serial.available()) {
  // get the new byte:
  char inChar = (char)Serial.read();


  // if the incoming character is a newline, set a flag
  // so the main loop can do something about it:
```

```
    if (inChar == '1') {
      digitalWrite(FAN, LOW); }
    if (inChar == '2') {
      digitalWrite(FAN, HIGH);
    }
    if (inChar == '3') {
      digitalWrite(LIGHT, LOW);
    }
    if (inChar == '4') {
      digitalWrite(LIGHT, HIGH);
    }}}
//-------------------------------------------------------------
void serialEvent() {
  while (Serial.available()) {
    // get the new byte:
    char inChar = (char)Serial.read();


    // if the incoming character is a newline, set a flag
    // so the main loop can do something about it:

    if (inChar == '1') {
      digitalWrite(FAN, LOW);
    }
    if (inChar == '2') {
      digitalWrite(FAN, HIGH);
    }
    if (inChar == '3') {
      digitalWrite(LIGHT, LOW);
    }
    if (inChar == '4') {
      digitalWrite(LIGHT, HIGH); }}}
```

Wi-Fi Module Codes

```
//Working with ESP-01
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleStream.h>

// Your WiFi credentials.
// Set password to "" for open networks.
const char* ssid = "smarthome";
const char* pass = "12345678";

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "sZCld4jogUt1FhgLLhBr-tOgQR_vlH0Z";
char server[] = "139.59.206.133";



int Beat, BPM, SPO2;
int Counter, BeatCycle = 0;
int countsend = 0;
int cycle = 0;  float voltage = 0;
String DATA = "";
int P1 = 0, P2 = 0, P3 = 0, P4 = 0;
int Rly1 = 0, Rly2 = 0, Rly3 = 0, Rly4 = 0, Rly5 = 0;
int led1x = 0, led2x = 0, led3x = 0, led4x = 0;
int TotalUse = 0;
int TotalAvai = 0;
float Temp1 = 30.1423;
float PH = 7;
float Temp2 = 30.2;
String Flat;
String Flon;
String Temp1x = "";
```

```
String PHx = "";
String Temp2x = "";
String Temp1y = "";
String PHy = "";
String Temp2y = "";
String Temp3y = "";
String Temp3x = "";
String Temp4y = "";
String Temp4x = "";
String Temp5y = "";
String Temp5x = "";
String Temp6y = "";
String Temp6x = "";
String Temp7y = "";
String Temp7x = "";
String Temp8y = "";
String Temp8x = "";
String Temp9y = "";
String Temp9x = "";
float latx, lonx;
int Timer = 0;
int Mode = 0;
int DataIn = 0;
int ALERT = 0;
float Sens1 = 0;
```

```
WiFiClient wifiClient;

// This function tries to connect to the cloud using TCP
bool connectBlynk()
{
  wifiClient.stop();
  // return wifiClient.connect(BLYNK_DEFAULT_DOMAIN, BLYNK_DEFAULT_PORT);
  return wifiClient.connect(server, BLYNK_DEFAULT_PORT);
}

// This function tries to connect to your WiFi network
void connectWiFi()
{
  Serial.print("Connecting to ");
  Serial.println(ssid);

  if (pass && strlen(pass)) {
    WiFi.begin((char*)ssid, (char*)pass);
  } else {
    WiFi.begin((char*)ssid);
  }

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

```
//-----------------------Manage Virtual Pin---------------------------
BLYNK_WRITE(V10)
{
  int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable
  Rly1 = pinValue;
  // process received value
}
BLYNK_WRITE(V11)
{
  int pinValue1 = param.asInt(); // assigning incoming value from pin V1 to a variable
  Rly2 = pinValue1;
  // process received value
}

BLYNK_WRITE(V12)
{
  int pinValue3 = param.asInt(); // assigning incoming value from pin V1 to a variable
  Rly3 = pinValue3;
  // process received value
}

BLYNK_WRITE(V13)
{
  int pinValue4 = param.asInt(); // assigning incoming value from pin V1 to a variable

  Rly4 = pinValue4;
}
BLYNK_WRITE(V14)
{
  int pinValue5 = param.asInt(); // assigning incoming value from pin V1 to a variable

  Rly5 = pinValue5;
}
```

```
void setup()
{

  Serial.begin(9600);

  connectWiFi();

  connectBlynk();

  Blynk.begin(wifiClient, auth);

}

void loop()
{
  WidgetLED led1(V1);

  // Reconnect WiFi
  if (WiFi.status() != WL_CONNECTED) {
    connectWiFi();
    return;
  }

  // Reconnect to Blynk Cloud
  if (!wifiClient.connected()) {
    connectBlynk();
    return;
  }

  if (Rly1 == 1) {
    if (P1 == 0) {
      P1 = 1;
      Serial.println("1");
      delay(1000);
    }
```

```
}
if (Rly2 == 1) {
  if (P2 == 0) {
    P2 = 1;
    Serial.println("3");
    delay(1000);
  }
}
if (Rly2 == 0) {
  if (P2 == 1) {
    P2 = 0;
    Serial.println("4");
    delay(1000);
  }
}

Blynk.run();
Timer++;


if (Timer > 1000) {


  Timer = 0;
}


while (Serial.available()) {
  // get the new byte:
  char inChar1 = (char)Serial.read();
  if (inChar1 == '*') {
    DataIn++;
```

```
}
if (inChar != '*' && inChar != '#' && DataIn==2) {
  Temp2x+=inChar;

}
if (inChar != '*' && inChar != '#' && DataIn==3) {
  Temp3x+=inChar;

}
 if (inChar != '*' && inChar != '#' && DataIn==4) {
  Temp4x+=inChar;

}
     if (inChar != '*' && inChar != '#' && DataIn==5) {
  Temp5x+=inChar;

}
     if (inChar != '*' && inChar != '#' && DataIn==6) {
  Temp6x+=inChar;

}
         if (inChar != '*' && inChar != '#' && DataIn==7) {
  Temp7x+=inChar;

}
         if (inChar != '*' && inChar != '#' && DataIn==8) {
  Temp8x+=inChar;

}
         if (inChar != '*' && inChar != '#' && DataIn==9) {
  Temp9x+=inChar;
```

```
}
         if (inChar != '*' && inChar != '#' && DataIn==9) {
  Temp9x+=inChar;

}


if (inChar == '#') {
  DataIn=0;
 Temp1y=Temp1x;    PHy=PHx;       Temp2y=Temp2x;   Temp3y=Temp3x;   Temp4y=Temp4x;
 Temp5y=Temp5x;    Temp6y=Temp6x;   Temp7y=Temp7x;   Temp8y=Temp8x;   Temp9y=Temp9x;
 Temp1x="";
 PHx="";    Temp2x=""; Temp3x="";    Temp4x="";   Temp5x="";   Temp6x="";
 Temp7x="";    Temp8x="";    Temp9x="";
 Blynk.virtualWrite(V0, Temp1y);
 Blynk.virtualWrite(V1, Temp2y);

}
    }

}
```

Appendix B

	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14
TASK 1: Background study: Find for research papers related to the project														
TASK 2: Begin to draft the report														
TASK 3: Research on Literature Review														
TASK 4: Begin report according to the template given														
TASK 5: Complete Chapter 1 and Chapter 2 of Report														
TASK 6: Submitted the draft of Chapter 1 & 2 after correction														
TASK 7: Complete Chapter 3 (Methodology)														
TASK 8: Submission of Draft 1														
TASK 9: Prepare Slide Presentation														
TASK 10: Mock presentation submit video (Evaluation by Supervisor)														
TASK 11: BDP 1 Presentation														

Gantt chart FYP 1

Appendix C

	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14	WEEK 15
TASK 1: Briefing on FYP 2 & e-psm website															
TASK 2: Workshop															
TASK 3: Discussion with Supervisor															
TASK 4: Continue report and project hardware															
TASK 5: Week 6 evaluation on logbook															
TASK 6: Submission of Draft report															
TASK 7: Continue the software of project															
TASK 8: Submission logbook week 12															
TASK 9: Submit Report to SV															
TASK 10: Prepare poster/slide/video for presentation															
TASK 11: BDP 2 Presentation & Evaluation															

Gantt Chart FYP 2