Faculty of Electrical and Electronic Engineering Technology

SMART STREET LIGHTING SYSTEM USING NODEMCU

MUHAMAD FARIS BIN CE UTEH

Bachelor of Electronics Engineering Technology with Honours

2021

SMART STREET LIGHTING SYSTEM USING NODEMCU

MUHAMAD FARIS BIN CE UTEH

A project report submitted
in partial fulfilment of the requirements for the degree of
Bachelor of Electronics Engineering Technology with Honours

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I declare that this project report entitled "Smart Street Lighting System using NodeMCU" is the result of my research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature	:	
Student Name	:	MUHAMAD FARIS BIN CE UTEH
Date	:	19/12/2021

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.

Signature :

Supervisor Name : TS AHMAD FAIRUZ BIN MUHAMAD AMIN

Date : 11th JANUARY 2022

Signature :

Co-Supervisor Name (if any) :

Date :

DEDICATION

I dedicate my dissertation study project to my parents, Puan Nor Aishah Binti Hashim and Ce Uteh bin Mohd Lani who have always supported me financially and morally and advised me to never give up. For meeting all of my demands and teaching me that even the most difficult tasks can be achieved if handled in little increments and steadily.

I'd also like to thank and dedicate my project Supervisor, Sir Ahmad Fairuz bin Mohamad Amin who was always willing to help and advise me with beneficial ideas, suggestions, and support till I finished it.

Last but not least, I'd want to dedicate my thesis to a friend who has been a fantastic supporter of mine during the development of my project, motivating me with her undivided attention to finishing my work with real respect.

ABSTRACT

In the era of globalization, An IoT ecosystem is made up of web-enabled smart devices that gather, send, and act on data from their surroundings using embedded systems such as CPUs, sensors, and communication hardware. By connecting to an IoT gateway or other edge device, IoT devices may exchange sensor data that is either routed to the cloud for analysis or examined locally. These gadgets may occasionally interact with one another, and act on the information they receive. Although individuals may engage with the devices to set them up, give them instructions, or retrieve data, the gadgets conduct the majority of the work without human participation. This is because the community uses the internet daily. In this paper, the internet is used as an intermediary between the user and the streetlight. It helps society especially road users to stay safe and practice the saving of electrical energy. Dimming the lights during off-peak hours is a simple and practical solution to this problem. The lights around it will shine in the usual (bright) mode whenever presence is sensed. This would save a significant amount of energy while also lowering the cost of operating the lamps. The timer feature was used in the former streetlight system. It is upgraded to another system using a sensor. However, it is still less effective, especially for highway streetlights. They use a lot of electricity to operate, and their heat outputs are also extremely substantial. To monitor and access the system, this system employs an IoT base connected to the Microcontroller wifi. Technology implement is more detailed and safe than the previous system [1] of a streetlight. We can use IoT to check the status of street lights on the internet in real-time from anywhere and resolve any difficulties that arise throughout the process.

ABSTRAK

Di era globalisasi internet pelbagai benda (IoT), ekosistem IoT terdiri dari peranti pintar berkemampuan tinggi yang mengumpulkan, mengirim, dan bertindak berdasarkan data dari persekitarannya menggunakan sistem tertanam seperti CPU, sensor, dan perkakasan komunikasi. Dengan menyambung ke gerbang IoT atau peranti tepi lain, peranti IoT dapat menukar data sensor yang disalurkan ke Cloud utama untuk dianalisis atau diperiksa secara tempatan. Peranti ini mampu saling berinteraksi dan bertindak berdasarkan maklumat yang mereka terima. Meskipun manusia mungkin terlibat dengan peranti untuk, memberi arahan, atau mengambil data, peranti ini mampu melakukan sebahagian besar tanpa adanya manusia. Hal ini demikian kerana setiap hari internet digunakan oleh seluruh lapisan masyarakat. Oleh itu, internet digunakan sebagai perantara antara pengguna jalan raya dan lampu jalan. Ia membantu masyarakat terutama pengguna jalan raya untuk tetap selamat dan mengamalkan penjimatan tenaga tenaga elektrik khususnya. Penyelesaian yang mudah dan berkesan untuk ini adalah dengan meredupkan seketika lampu pada waktu puncak. Setiap kali pergerakan dikesan, lampu di sekeliling akan menyala pada mod normal (terang). Hal ini sedikit sebanyak akan menjimatkan banyak tenaga dan juga mengurangkan kos operasi lampu jalan. Sistem lampu jalan sebelumnya menggunakan fungsi pemasa. Ia dinaik taraf ke sistem lain menggunakan sensor. Walau bagaimanapun, ia dilihat masih kurang berkesan terutamanya untuk lampu jalan raya. Hal ini kerana banyak tenaga elektrik untuk berfungsi dan pelepasan haba yang cukup tinggi. Sistem ini menggunakan pangkalan IoT yang disambungkan ke Wifi Microcontroller untuk memantau dan mengakses sistem. Peranti teknologi alaf baru ini lebih terperinci dan selamat berbanding sistem lampu jalan sebelumnya. Individu dapat memeriksa status lampu jalan di internet menggunakan (IoT) dari mana sahaja dalam masa nyata dan mampu menyelesaikan masalah jika berlaku semasa pemprosesan.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Ts Ahmad Fairuz bin Muhammad Amin for their precious guidance, words of wisdom, patient throughout this project, endless support, and encouragement toward the completion of this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) and the Faculty of Electrical and Electronic Engineering Technology for the financial support which enables me to accomplish the project. Not forgetting my colleague for the willingness of sharing his thoughts and ideas regarding the project.

As a result, I'd like to express my gratitude to my parents for their unwavering support and encouragement in a variety of ways. I'd want to thank all of the laboratory employees who collaborate with me and are prepared to share their knowledge of the equipment. Finally, I'd want to thank my friends and colleagues that assisted me in the creation of my project and report. Please accept my sincere thanks for their unwavering support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

V	-	Voltage
I	-	Current
R	-	Resistance
lx	-	Lux
$\circ$	-	Degree
%	-	Per cent
δ	-	Voltage angle

LIST OF ABBREVIATIONS

V	-	Voltage

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

The main focus of this project is to wisely operate streetlights. For example, by placing this system on streetlight poles, it detects vehicle and pedestrian movement and turns the light on and off accordingly. It may also adjust the brightness of street lights in response to the detection to prevent accidents. There is a risk of thefts and crimes when lamps are turned off at night and there is no detection. Based on the previous system in Malaysia, the streetlight operates when timer at 7 pm. The lights will turn on whether there is vehicle movement or not. Then, it will turn off at 7 pm. It consumes a lot of energy to generate electricity.

Cities are being transformed into smart cities in every way thanks to modern technologies. It's necessary to find ways to manage the underlying infrastructure for physical sensing and actuation resources. The Internet of Things contains lots of such solutions, mostly at the lower (communication) layer (IoT). Street lighting is the most expensive energy expense in a city. Municipal street lighting might be cut in half with the smart street lighting system (SSLS). The smart street lighting system turns on and off the lights automatically depending on the circumstance. It detects the movement of the object within a certain range automatically.

1.2 Problem Statement

In today's world, humans and technology are inextricably linked since they are both essential to achieving any goal. This includes the amount of energy we use in our daily lives. Nowadays, we can observe that every company, particularly the electrical business, by employing new technology to construct their systems to function at a high level.

It is to minimize power usage, especially between the hours of 12:00 A.M. and 5:00 A.M. when there are no people or vehicles on the road, parking lots, gardens, motorways, playgrounds, housing societies, and private/public places are all examples of public/private places. The stars may be obscured by light pollution in urban areas, hindering astronomy and the migration of many bird species.

1.3 Project Objective

Mainly, the objective will clarify the results that need to be accomplished at the end of the project. The goal of the objectives is to preserve the project accurately and well specified. The key to this objective is recorded as below:

a) To develop the new system for street lights in Malaysia cities.

b) To conduct green technology of energy saving.

c) To develop an effective new technology using the Internet of Things (IoT).

1.4 Scope of Project

This project intends to create and implement innovative embedded technologies for energy savings in street lighting, particularly in smart cities. The Street Light Lamp is equipped with Internet of Things (IoT) sensors such as PIR, LDR sensor, Microcontroller, and WIFI module. The sensors detect motion and light and provide data to the microcontroller, which operates the lamp and communicates data to the user application over

Wi-Fi. As a result, the system is intended to be controlled remotely and to operate in AUTO mode on behalf of the users' application.

1.5 Organisation

This project focuses on improving the streetlight system by applying IoT applications to control the lamp function. There are five chapters in this report. The project's introduction, objectives, issue descriptions, and scope are all outlined in the first chapter. In the second chapter, it is significant to develop literary studies that have been applied from past sources for reference purposes. In chapter three, you will find a description of the component, a flowchart for the project, and an explanation of the technique used. All of the analyses done during the project development process are tallied in the fourth chapter, and the results are presented.

1.6 Structure of Report

This report contains three chapters that precisely explain the idea and theory of the project where it requires all action to accomplish the final product. The description of each chapter will be in a paragraph as follows.

Chapter 1 briefly defines the background and basic information about the system of streetlight itself. In addition, problem statements, objectives, and the scope of this project are also proposed in this chapter.

Chapter 2 describes a literature review that approaches the previous projects that are related to this project. In addition, this chapter contains the concept of software and hardware requirements for the system that will be developed.

Chapter 3 alerts more on the methodology of the project from starting point until the end. The methodology consists of the software and hardware development of this project.

The software development consists of the application of streetlights in the city. The hardware development consists of certain components supported by the Blynk application.

Chapter 4 aims at analysing the result of software and hardware development. The results will be in form of figures and discussion.

Chapter 5 will be the summary of this project along with the discussion and recommendation for future improvements.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The goal of the literature review in this chapter is to assess and investigate previously executed projects about the Smart Street Lighting system using the Internet of Things (IoT). Every piece of research is based on a study completed no later than 5 years after the project's implementation date, thus it may be used as a reference. In addition, the research's applicability for the project in question will be examined, assessed, and evaluated. A comparison of the hardware used is also carried out to make the selection possible of appropriate hardware that will be utilized, as well as the benefits and drawbacks of hardware and systems, to produce a more accurate hardware selection. The results and techniques of previous studies will be scrutinized and assessed to verify that they are the best and most accurate source of information for this project.

2.2 Related Work

2.2.1 NodeMCU ESP32 Microcontroller

The NodeMCU [1] ESP32 is the newest product of Ai-Thinker in the global market. Equipped with an L106 32-bit RISC microprocessor core with running speed at 160 MHz, which is faster than the ESP8266 model (80MHz). It also comes with Bluetooth 4.2 and HT40 on Wi-Fi. This is including the temperature sensor and touch sensor.

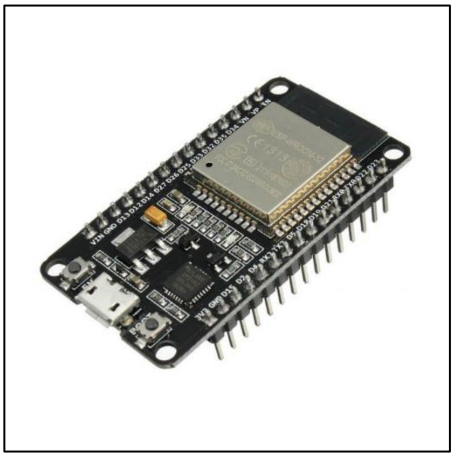

Figure 2.1 NodeMCU ESP32

2.2.2 GPIO

The general-purpose input-output device is one sort of input-output device (GPIO). GPIO [2] pins can be set to act as a general-purpose input, a general-purpose output, or one of up to six pin-dependent functionalities. The ESP32 has its own set of GPIO banks. It has RX for data reception and TX for data transmission. A 3.3V VCC supply powers all GPIO banks. The ESP32's row of GPIO pins along the top edge of the board is a standout feature. On all current ESP32 boards, there is a 36-pin GPIO header.

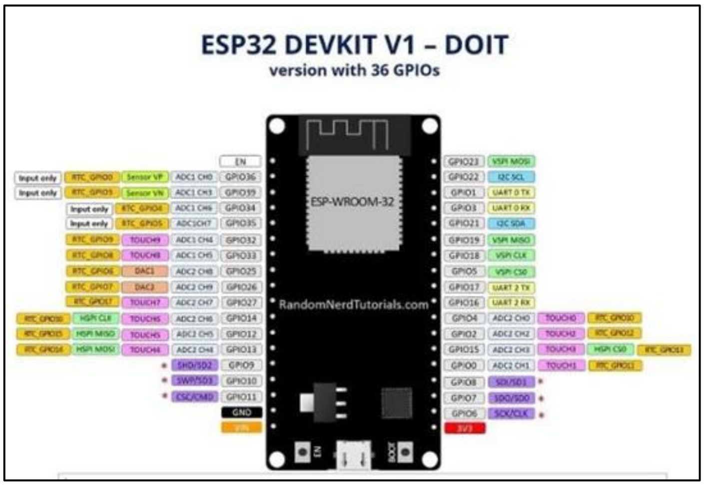

Figure 2.2 GPIO Pinout

2.2.3 Smart LED Systems Using Esp32 IoT Based Wi-Fi

Based on a proposal for a project by [3], this project aims for a Smart LED system based on ESP32 technology. The current sensor, voltage sensor, PIR sensor, and ZCD circuit are all included. This project works when the voltage and current sensors are connected to the ESP32's ADC pins, and the passive actinic radiation sensor is connected to the ESP32's GPIO pin. Power usage is detected and communicated to the cloud [3] using the current sensor, voltage sensor, and ZCD data.

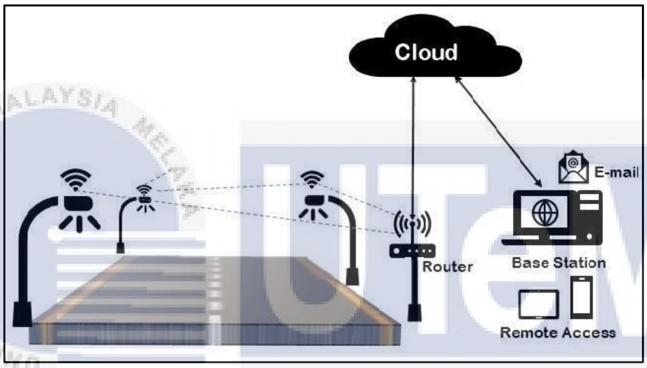

Figure 2.3 Schematic Diagram of ESP32 Streetlight Based Wi-Fi

Network assets boards are used to create smart links between existing networks and context-aware computing. are two important divisions of IoT as a state in the study by [4]. The conceptual scheme of the proposed system is presented in Figure 2.1.3. It comprises streetlights and a central base station. An adjacent structure houses the central base station. This system is simple to extend. The cloud is used to communicate between the central base station and the streetlights.

2.2.4 Electrical Energy Conservation using a Smart Streetlight Management System

Based on a project proposed by [5], This project is an example of a smart streetlight control system, in which the system detects the movement of automobiles and people and turns the light on and off appropriately after being put on the streetlight poles. This system not only adjusts the status of street lights but also manages their intensity in response to theft and criminal detections, with the lamps shut off during the hours of darkness when no detection is made. When a car or person is detected on the roadway, the lamps on the next five poles are activated, and for street lights in towns, cities, public gardens, and parks, a device is put for each pole so that the lights may be more efficiently managed.

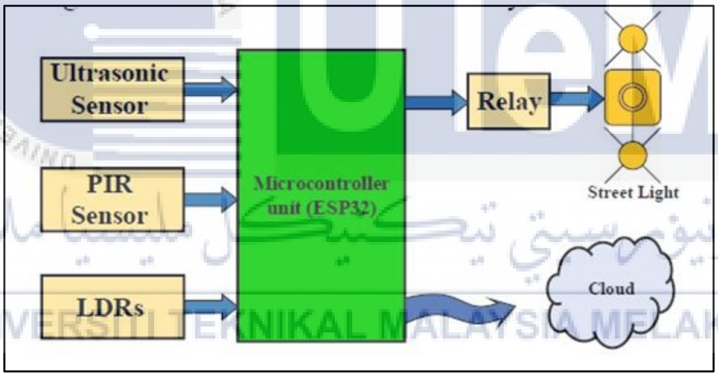

Figure 2.4 Block Diagram of Smart Street Light System with IoT

2.2.5 Street Light Automation for Smart Cities: An Energy-Efficient Approach Based on IoT

Based on a project proposed by [6], a sensor is used to measure sunlight intensity, day /night conditions, and track on a road. Several factors influence the brightness of LED streetlights. The motion sensor and the LDR (light-dependent resistor) sensor are two types of sensors. This is accomplished through a sensor known as a Light Dependent Resistor

(LDR), which detects light in the same way that human eyes do. The system automatically switches OFF lights whenever the sunlight comes, observable to our eyes [7]. When there is no motion of an object on the street at midnight, a motion sensor is utilized to modify the intensity of LED light, and all of the street lights are reduced to save energy.

The key characteristics of the Internet of Things are all-around perception, dependable transmission, and intellectual functioning. Using perception capture and measurement techniques, “all-out perception” refers to collecting and acquiring information about things wherever and at any time. Computer, ESP32 microcontroller, LED, LDR, and motion sensor. The quantity of sunshine available determines the system's switching activity.

LDR is in charge of it.; for example, when there is enough sunlight in the atmosphere, LDR resistance is lowest, and when there is darkness, LDR resistance is highest. The resistance threshold value can be set to any value desired by the user. This value is sent to the microcontroller, which turns on and off the LED lights. Because there is relatively little track on the road at night, LED lights with full intensity light are not required. We utilized a motion sensor to detect the movements of an item on the road. It can detect the motion of an item within a seven-meter range. When an object on a road does not move for a user-defined length of time, the light intensity is decreased to the absolute minimum using a PWM (Pulse width Modulation) pulse. When a car moves on the road, the microcontroller receives a signal, which triggers the microcontroller to produce a signal.

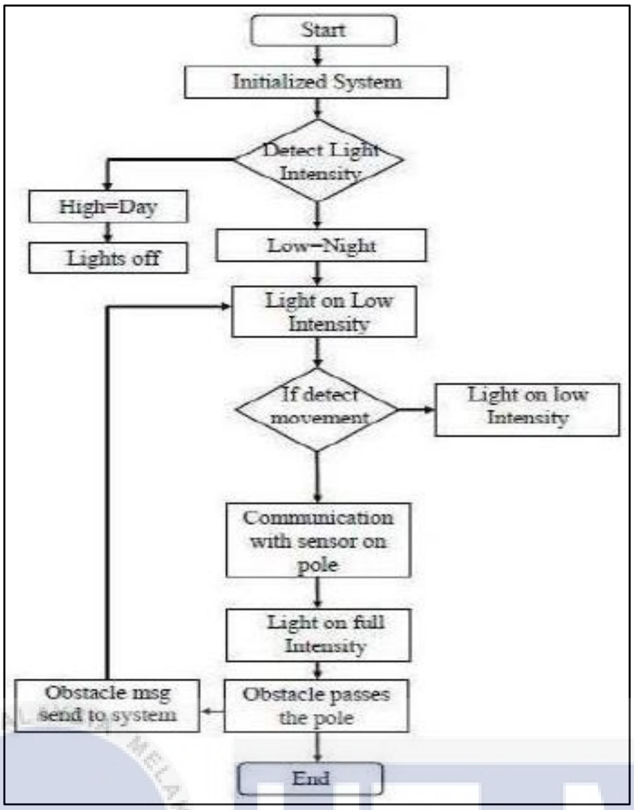

Figure 2.5 Flowchart of Streetlight Automation System Based on IoT

2.2.6 Internet of Things (IoT)

Based on [8], by exploiting the increased ubiquity of RFID, wireless, mobile, and sensor devices, IoT has presented a potential chance to construct sophisticated industrial applications and systems. Industrial IoT applications have been created in a variety of ways and implemented in recent years. To better comprehend the growth of IoT in industries, this article reviews current IoT research, essential supporting technologies, primary IoT Industry uses, as well as research trends and issues. Based on [9], the loT will be the true starting point of industry and other information industries, given the government push to integrate industrialization and information. Following the huge success of loT, many more little things will require the installation of clever sensors. IoT is gaining traction in a variety of businesses. As wireless communication, smartphone, and sensor network technologies advance, more networked or smart devices are getting involved in IoT. As a result, IoT technologies have had a significant influence on new ICT and business system technologies.

2.2.7 Previous Project Comparison

Table 2.1 Previous Project Comparison

Project	Name	Method	Description	Advantages
2.1.2	Smart LED Systems Using Esp32 IoT Based Wi-Fi	• An adjacent structure houses the central base station. • The cloud is used to communicate between the central base station and the streetlights. • Seasonal data is used to switch the lamps on and off. • Seasonal data is kept at a central base station.	• This system requires proper monitoring and energy management techniques to reduce energy wastage.	• LEDs have a lifespan of 5 times that of CFLs and 10 times that of Halogen lights. • Save up to 80% on electricity by replacing CFL and halogen lighting with LEDs.
2.1.3	Electrical Energy Conservation using a Smart Streetlight Management System	• The entire system can be monitored and controlled by a central system through a web interface. • A central database is created to fetch data from all individual systems.	• Mounting this system on streetlight poles.	• Increased current luminous efficiency. • Lower maintenance costs. • Excellent colour rendering index.
2.1.4	Street Light Automation for Smart Cities: An Energy-Efficient Approach Based on IoT	• The LDR sensor is used to regulate the switching of LED street lights based on the amount of sunshine.	• Utilized a sensor to detect the strength of the sun's light, the day/night cycle, and the track on a road.	• LED bulbs have a life expectancy than traditional lights. • LEDs are both low-cost and low-energy.

2.3 Summary

In terms of hardware and software implementation, the chosen research has certain differences and similarities. To access the light condition and monitor energy use, for example. As a microcontroller, the Arduino and Raspberry Pi are used in the majority of the projects. The ESP32 will be utilized as the main microcontroller in this project since it includes Wi-Fi and Bluetooth, which the conventional Arduino boards do not. Furthermore, it is easier to build a system feature that can regulate automatically and remotely when developing an IoT system. To overcome this constraint, smartphone apps that can operate the equipment are being developed.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The approach for project implementation will be outlined in this chapter as the development of a smart streetlight system using ESP32. It is including the project block diagram and flow chart to visualize the main ideas on how the system will be implemented. This chapter also will cover the whole method that will be used in the scheduled Gantt chart.

3.2 Project Workflow

To assess a project's performance, it is critical to have a well-organized and efficient workflow. A well-thought-out strategy is the foundation of any successful undertaking. Following the planning stage, research is required to aid in the project's implementation. Every probable problem may be discovered thanks to a thorough investigation to avoid complications during project execution. As a result, the project's design is built, followed by the project's execution. The project was then examined when it was completed to determine its efficacy.

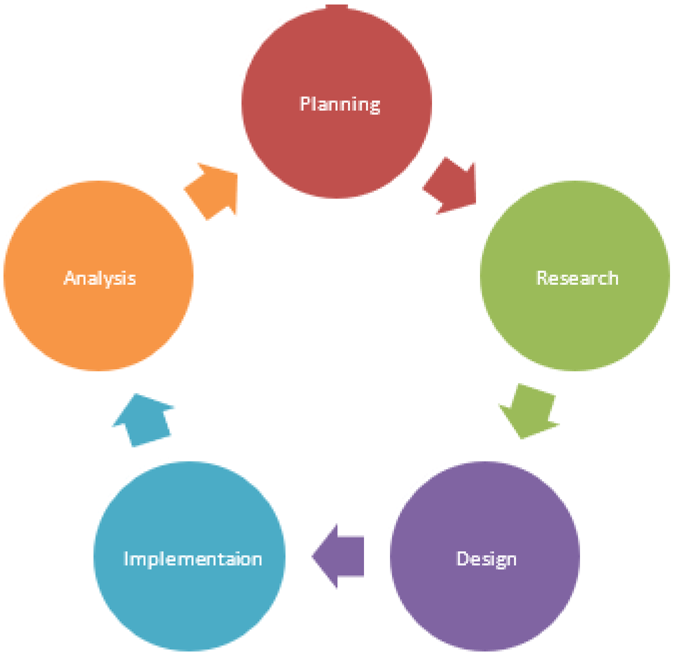

Figure 3.1 Project Work Flow

3.2.1 Planning

The purpose of project planning is used to help a student to manage their activities on PSM 1 and PSM 2. For PSM 1, students start to search for a suitable project title with the help of a supervisor. Then, students need to do some research about the topic based on the related topic of their project through articles, journals, and newspapers. In week 4, students need to submit their project title with some description of the project such as project summary of objective, problem statement, project method and expected outcomes of the project to the supervisor to get approval for the student to perform the project. In week 5, student needs to make research and analysis about other projects that are related to their project for comparison and determine what is the best solution and task that student needs to do for completing their project. Then, the student needs to write a report such as chapter 1 is for the introduction of the project that includes a background of the study, problem statement, objective, and scope of the project. Meanwhile, chapter 2 includes a summary of the research paper of the related project as well as the hardware and software requirements that are suitable to be used to complete the project. In chapter 3 include a summary of student project specifications and planning such as project block diagram, flowchart, and project development. In week 14, a presentation for PSM 1 is conducted where two panels will review the student project. These panels asked some questions about the related topic and will give some suggestions to the student to improvise the project.

Students need to complete their project for hardware and software design during PSM 2 in the next semester. Students need to configure hardware and determine what is the best solution to complete the task and make sure that all the components can simulate accordingly with their functions. Then, students need to develop software and make sure that both hardware and software can integrate well. After that, the student needs to monitor the simulation to make sure that it can operate properly and if there is an error that occurs, the student needs to configure the error for the project to work successfully. The final task is for students need to prepare for a final presentation to the panels and hand in a complete final report of PSM including chapter 4 and chapter 5 to the supervisor and panel.

3.2.2 Project Flowchart

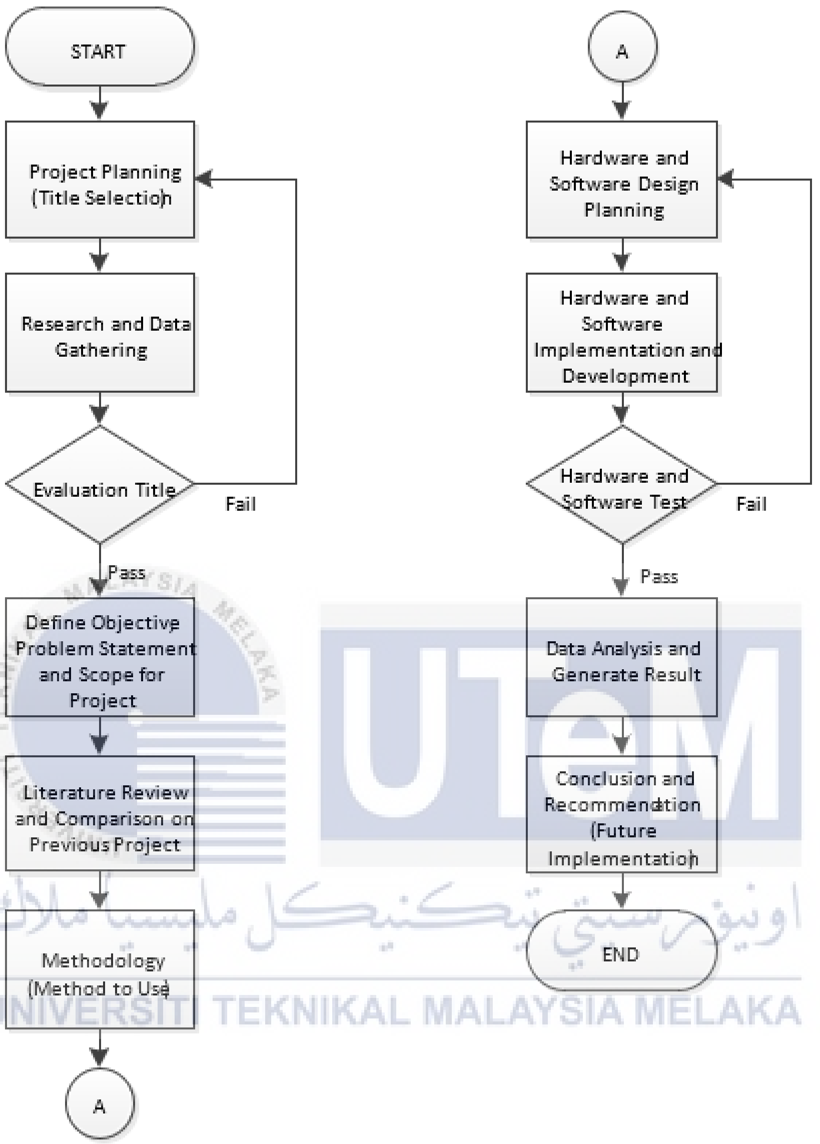

Figure 3.2 Project Flowchart

3.2.3 Research and Data Gathering

The major goal of this investigation is to identify the source and issues that contribute to the neglect and incompetence of the associated issue of a streetlight. The research process is one of the most important elements of designing a smart streetlight system with ESP32. This is because, at any point during this phase, the source and problem produced by the preceding system's negligence and ineptitude might be discovered. As a result, many concerns have been investigated to discover a solution. Some of the earlier projects were investigated and compared to help in the creation of a solution and the enhancement of the system that would be developed. In addition, a project is being developed, and the appropriate hardware and software will be chosen based on the information and data gathered from the previous journal. As a result, several options for finding further solutions in this project have been considered, such as performing research through websites.

3.2.4 Design the Smart Street Lighting System

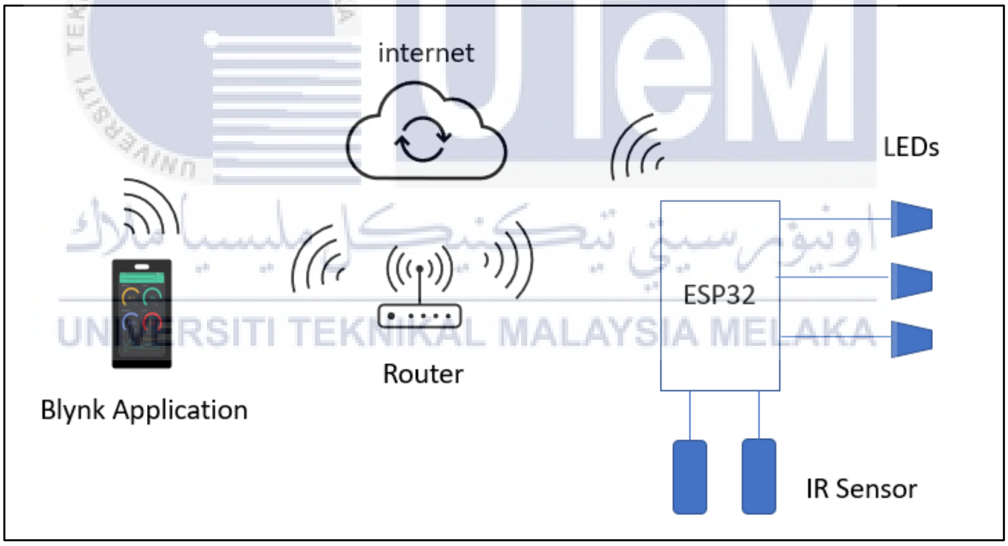

Figure 3.3 Wireless Connection between ESP32 and Internet

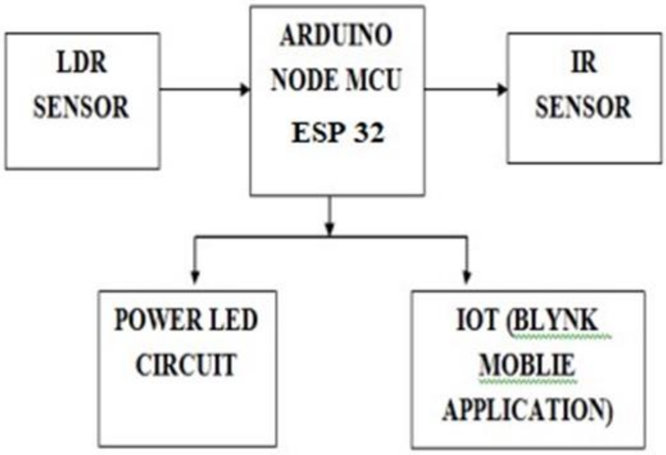

Figure 3.4 Smart Streetlight Block Diagram

This graphic depicts the overall operation of the Streetlight System. Users can use a mobile device or a personal computer to manually access the control system. This system will monitor the active and non-active lamps. Here, a single microcontroller is used as the brain of the system. All the sensors such as a resistor, IR sensor, LDR, and LED will connect to the microcontroller. LED reacts as an output of the project, which is emitting light. The IR sensor detects the motion of any movement. As the IR sensor gets the information, it will send the data to the microcontroller. Later, the microcontroller instructs the LED whether to turn OFF or ON the lamp. In addition, the data from sensors will send to NodeMCU and can be accessed through mobile apps which Blynk.

3.2.5 System Access Using Mobile Device Flowchart

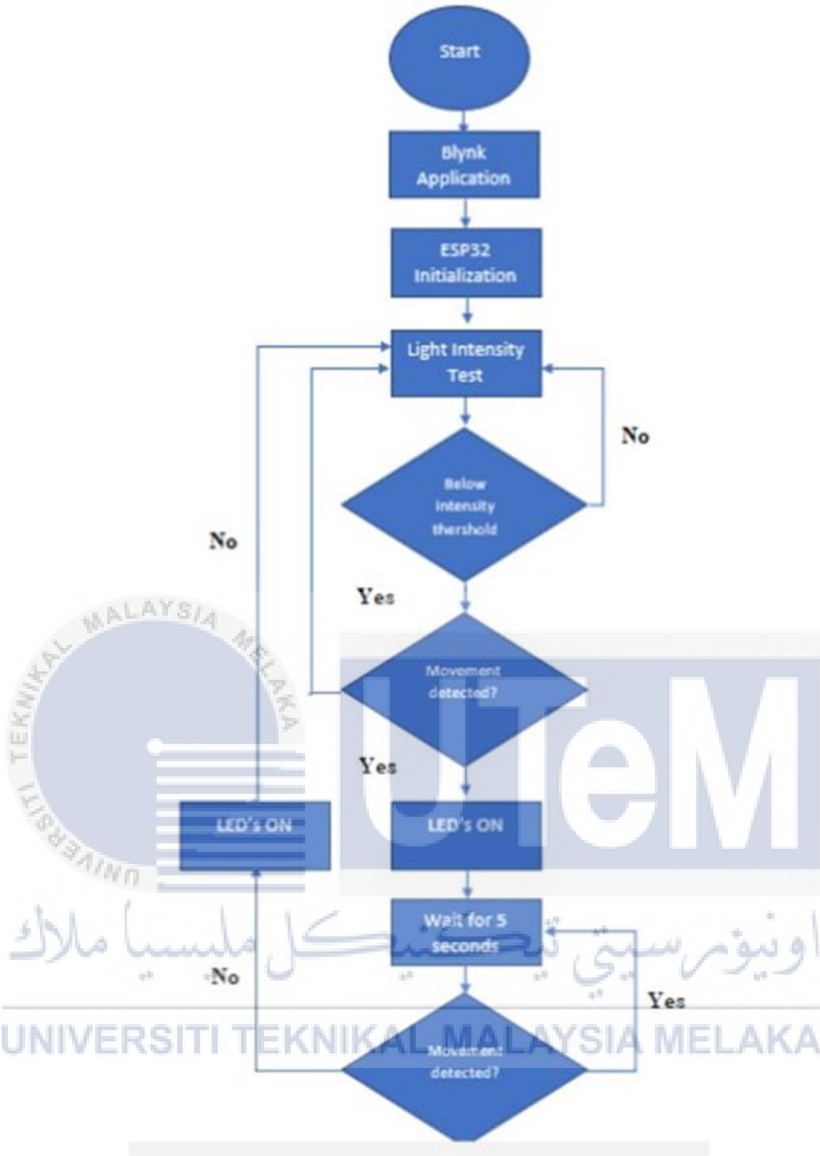

Figure 3.5 Flowchart of Software on ESP32

3.3 Hardware Implementation and Circuit Connection

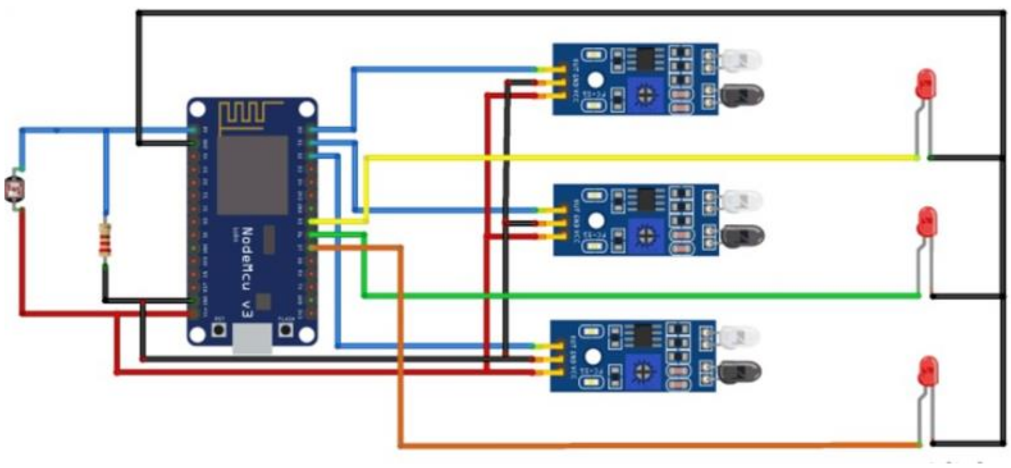

Figure 3.6 Hardware Design on Fritzing

3.3.1 Resistor with ESP32

In an electronic circuit, a resistor is an electrical component that controls or regulates the passage of electrical current. Resistors can also be used to supply a set voltage to an active device like a transistor. In a direct-current (DC) circuit, the current through a resistor is inversely proportional to its resistance and directly proportional to the voltage across it, all other parameters being equal. To avoid an excess supply, it connects before the LED and sensor connection.

3.3.2 LED with ESP32

A single LED with a power greater than 0.5W is referred to be an LED light source. Many manufacturers now utilize low-power LEDs, however, this necessitates the use of a large number of LEDs, as well as lower-power LEDs with higher light decay. As a result, high-power LED sources are becoming increasingly popular in commercial lighting. It is connected as output for the project, which emits light. This 4 LED is connected to pins 18, 19, 26, and 27 of the microcontrollers.

3.3.3 IR Sensor with ESP32

An infrared sensor is an electronic equipment that emits and/or detects infrared radiation to perceive particular features of its surroundings. It's also capable of sensing motion and monitoring an object's heat. Infrared wavelengths can't be seen with the naked eye. Infrared radiation is the part of the electromagnetic spectrum with wavelengths greater than visible light but shorter than microwaves. The data pins are linked to the ESP32 pins 21, 25, 32, 34, and 39.

3.3.4 Power Supply for ESP32

As the needed power source to switch on the ESP32 is 3.3V, the power supply is obtained from the power bank for testing reasons. As a result, the power bank's output is adequate to power this gadget. The 5V power supply from the power adapter is extremely ideal for usage if the system operates continuously for the system's dependability. The microcontroller is connected to the power bank through a USB connection.

3.4 Software Implementation

A set of data instructions or programs that instructs a computer on how to do a certain activity is known as software. To function, almost all automated devices or machines require software integration. System software and application software are the two types of software available. System software is software that allows users to interact with the actual computer system by providing fundamental capabilities. Linux, Windows, and IOS are examples of system software.

3.4.1 Blynk Application

Blynk allows us to create applications and then use them to control an Arduino board connected to a PC with an internet connection using a smartphone from anywhere in the world. Operate LEDs, servos, and many more. It is made for the Internet of Things (IoT) [10], and it can show sensor data, store it, and visualize it.

Figure 3.7 Blynk Mobile Application

3.4.2 Arduino IDE

The Arduino [11] Integrated Development Environment (IDE) is an open-source electronics platform that allows users to develop projects with basic hardware and software. The programming languages C and C++ are supported. With the help of third-party cores, it is used to create and upload applications to Arduino-compatible boards and other vendor development boards.

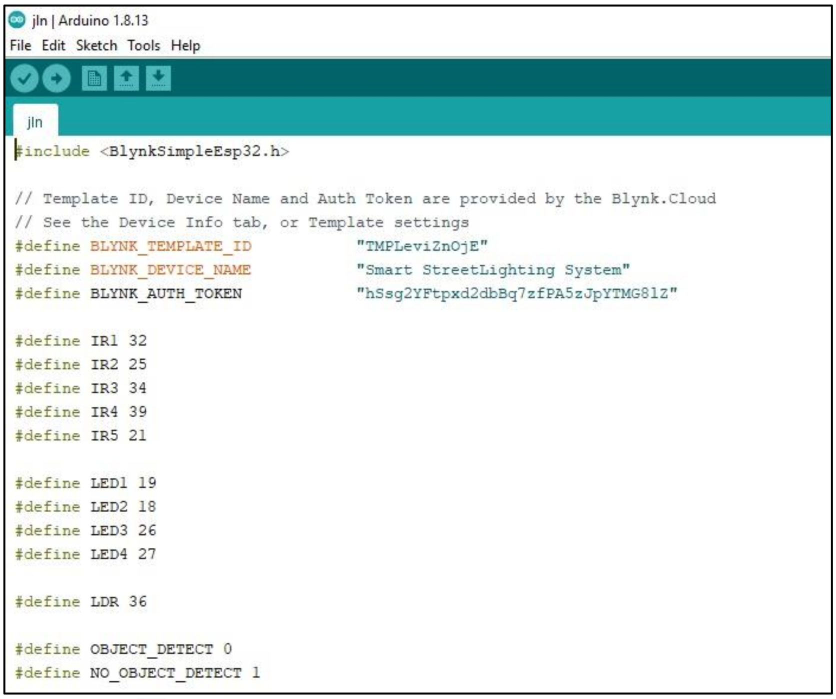

Figure 3.8 Arduino IDE Software

3.5 Prototype Installation

The last task to complete this project is to make a prototype installation. A prototype is the actual representation of the finished product. All the components need to be assembled before making the prototype to ensure that the process of installation will run smoothly. The components are placed inside the casing and a power bank is used as the power source of this project.

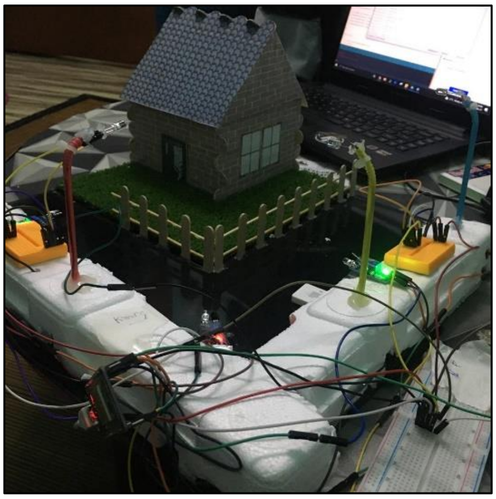

Figure 3.9 Prototype Installation

3.6 Summary

For summarization, Chapter 3 involves on how the structure of the project flow. This chapter explains the details of the hardware and software that are going to be used for the project. In addition, it also includes a piece of information by using a table, flowcharts, and block diagram that is easier to understand.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter will discuss the result obtained throughout this project from the beginning until after completion of this project as well as the performances with a problem that occurred. The findings and analysis from creating a smart streetlight system based on the ESP32 microcontroller are summarised in this chapter. Both hardware and software will be examined in this section. As a result, the data will be evaluated independently for LED and LDR intensity to meet the objectives. The complete result, including the serial monitor output from the ESP32 and sensor data from the Blynk mobile application, will be presented.

4.2 Data Analysis

Data analysis was managed to obtain from the results of the components on the smart streetlight project. The analysis was done to get accurate data from the sensor and component. In this section, the result of the resulting product of our project will be shown. The system includes an IR sensor, LED, and LDR. IR sensor is used to detect the movement of the object, which is the vehicle. Before setting up the parameters, some data must be collected and examined, such as the degree of brightness and LED power consumption in different duty cycles. Table 4.1 tabulates the average reading taken at Seremban in good weather using the analogue value of the LDR, which runs from 0 to 1024 lux.

Table 4.1 Average Reading of LDR

Time	Average Reading(Lux)
5.00 PM	650
6.00 PM	550
7.00 PM	400
8.00 PM	63
9.00 PM	57
10.00 PM	45
11.00 PM	23
12.00 AM	23

In addition, for LED brightness, a study of voltage consumption throughout multiple life cycles was carried out. The data is presented in Table 4.2.

Table 4.2 LED Consumption in Duty Cycle

Duty Cycle %	**LED**		
	Consumed Voltage (V)	**Current (A)**	**Consumed Energy (mW)**
7	0.08	0.02	1.6
10	0.09		1.8
20	0.19		3.8
30	0.29		5.8
40	0.39		7.8
50	0.48		9.6
60	0.58		11.6
70	0.68		13.6
80	0.78		15.6
90	0.88		17.6
100	0.97		19.4

This project decided to employ a 70% duty cycle of PWM based on the measurements reported above. The LED consumes 70% less energy and provides a brightness that is adequate for human eyes when the PWM duty cycle is set to 70%, as shown in Figure 4.1.

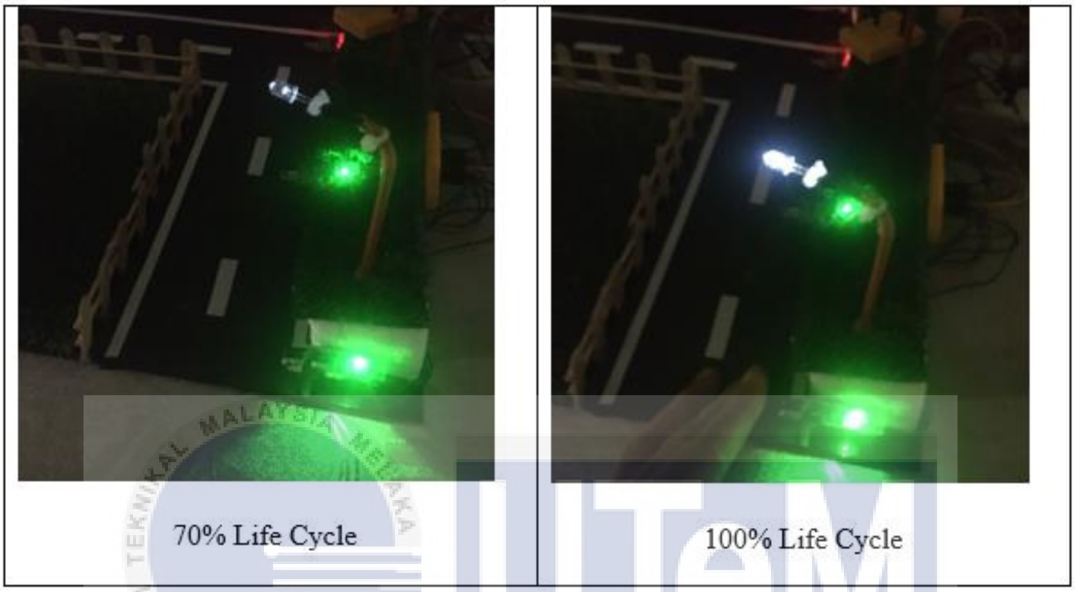

Figure 4.1 LED Brightness at 70% and 100% Life Cycle

4.3 Results

Based on data obtained, the life cycle from 100% fully bright to 70% life cycle situations based on the data from Table 4. The operating hours are from 5 pm to 12 am, as shown in Figure 4.2 and Figure 4.3. Assuming all led are the same as it places in the same sequence.

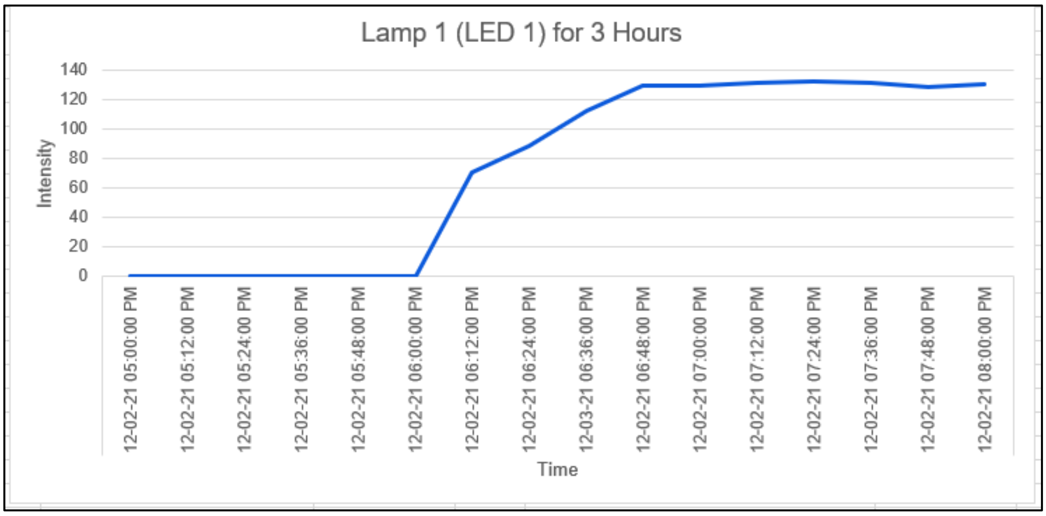

Figure 4.2 Data Recorded for Lamp Intensity from 5 PM to 8 PM

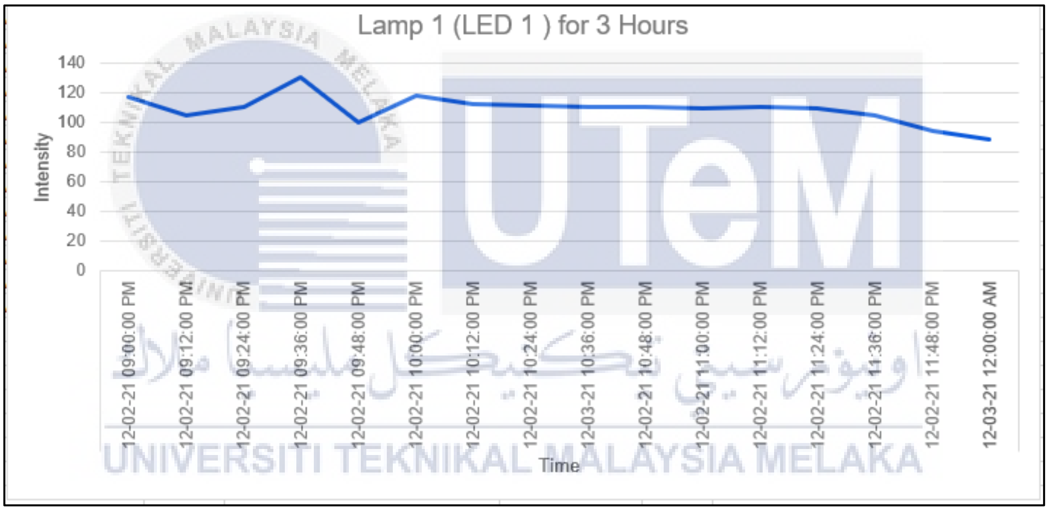

Figure 4.3 Data Recorded for Lamp Intensity from 9 PM to 12 AM

Figure 4.2 and Figure 4.3 show the input and output obtained from the hardware used in this project. LDR will detect the level of light. When there is sunshine, LDR will read a high value in lux in second (lx/s). Commonly, it indicates daytime. When the LDR value is between 250 and above, LED will light up at 70% intensity. Hence, it shows that the surroundings getting darker and nighttime will come. As there is vehicle movement in the city, the lamp will glow from 70% to 100% intensity. It will light up for 3 seconds until the vehicle passes through it.

4.3.1 Software Part

The interface was designed using Blynk Legacy Application. The app's interface was intended simply to ensure users monitor every component and sensor used. This application consists of monitoring and data uploading to the cloud.

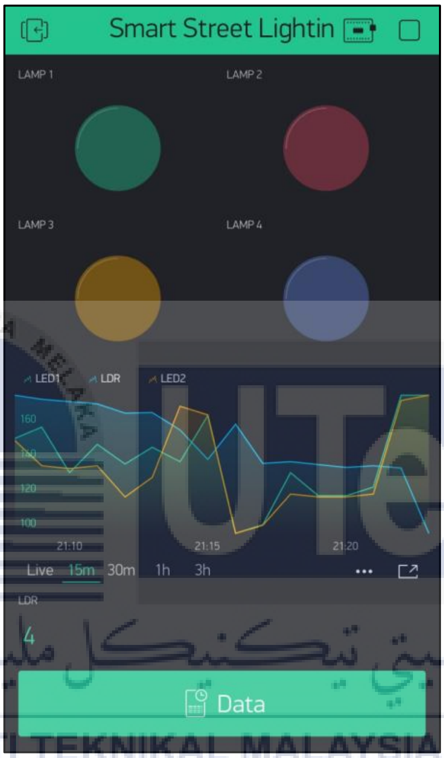

Figure 4.4 Interface of Blynk Application

Figure 4.4 shows the interface of the application developed using in Blynk Legacy Application. This app consists of a few icons or buttons that can operate the apps. All these functions were picked from the widget box, which provided the accessories we need. The first 4 icon upward is for LED. It indicates each of Lamp 1, Lamp 2, Lamp 3, and Lamp 4. The big box in middle represents data of each sensor and component, called the Super Chart. It shows live data and previous data. Also, the data can be downloaded from the Blynk cloud database. The file will be in .csv format. Lastly, it is for LDR reading. It is functioned to display the current LDR light level. To connect to the system, the user must connect to the Wi-Fi, between ESP32 and mobile phone hotspot. An error message will pop up in Figure 4.5 if there is some issue while connecting to the smart street lighting system. This error will occur if the Wi-Fi or the device is turned off or the device is out of range.

Figure 4.5 Error Connection Notification

Figure 4.6 shows data that import from Blynk Cloud. By sending from the cloud, the database will be entered to email registered in Blynk. The format of data will be in Microsoft Excel (.csv).

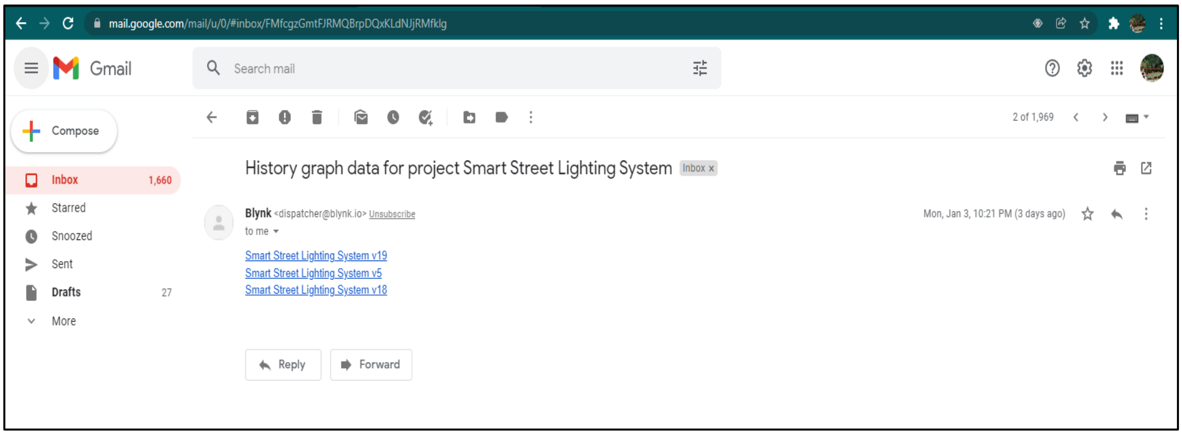

Figure 4.6 Import Data from Blynk Cloud

Figure 4.7 shows the uploaded data from the Blynk Application. The database shows the data collected from Lamp 1 (Sensor V19). Certain values display in the sheet. For example, the intensity of the sensor and timestamp. Timestamp data was shown in Unix timestamp. However, the timestamp was not displayed incorrect format. Hence, a formula has been added to convert to Excel Date, which was Real-Time date. ‘(B2/1000)/86400+DATE(1970,1,1)+time(8,0,0)’ was the conversion formula to convert to real date and time.

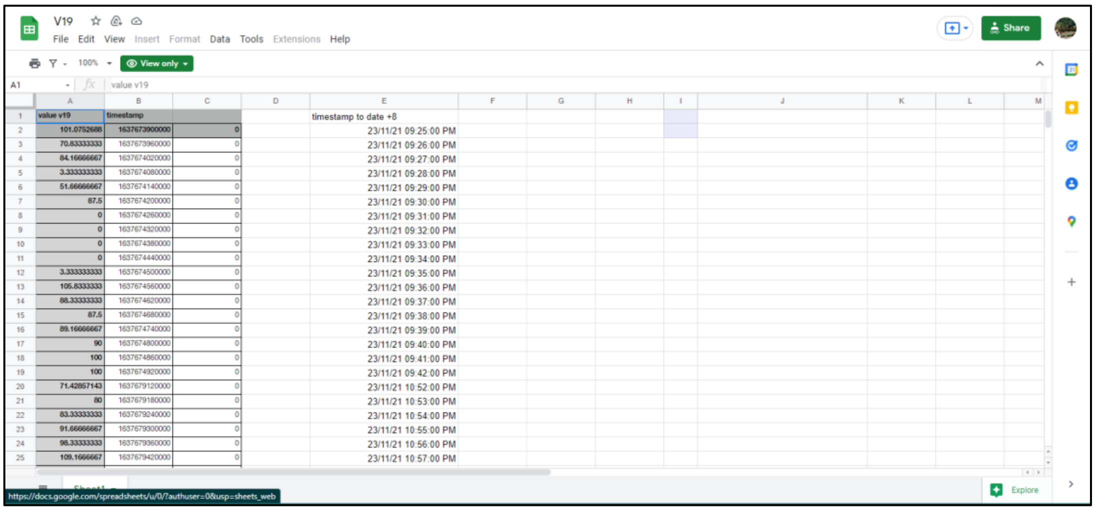

Figure 4.7 Web-based Database/Google Spreadsheets

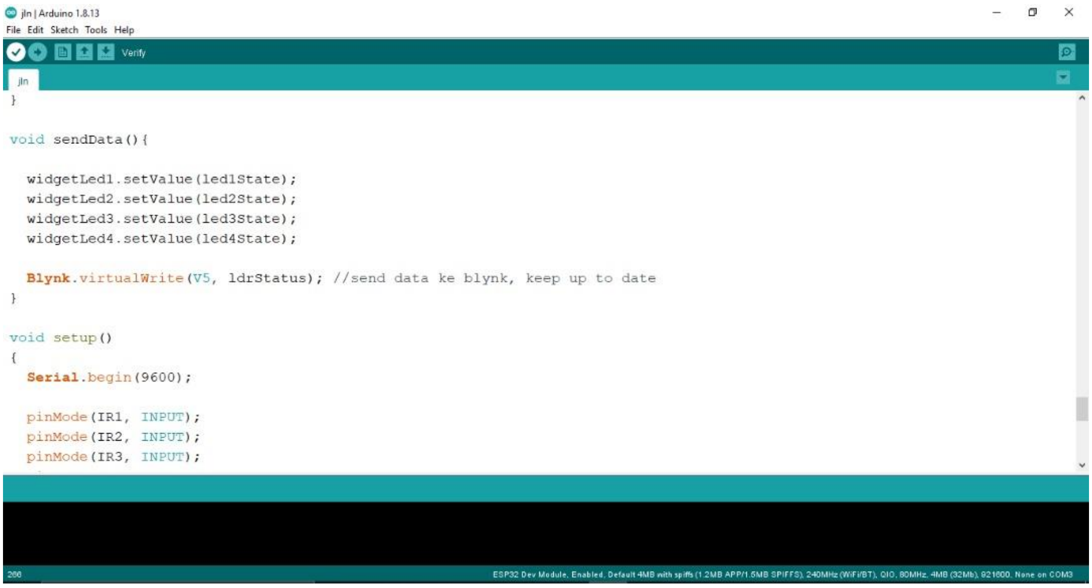

Figure 4.8 Programming Language of Project System

To build a complete project, programming is the main thing in this task. All the entire system runs from the program. Some functions were used to get the flow of the project. For example, voidSendData was used to keep updates between the system and prototype. Also, voidTimer was declared for timing purposes. This includes when the sensor detected an object, street light begin to count timer. Hence, all functions are associated with each other for perfect flow.

4.3.2 Hardware Part

Based on the hardware part, several techniques and configurations are used to ensure that the project is complete and working successfully. This part contains three main sections: the project testing, design of the prototype, and system of a prototype.

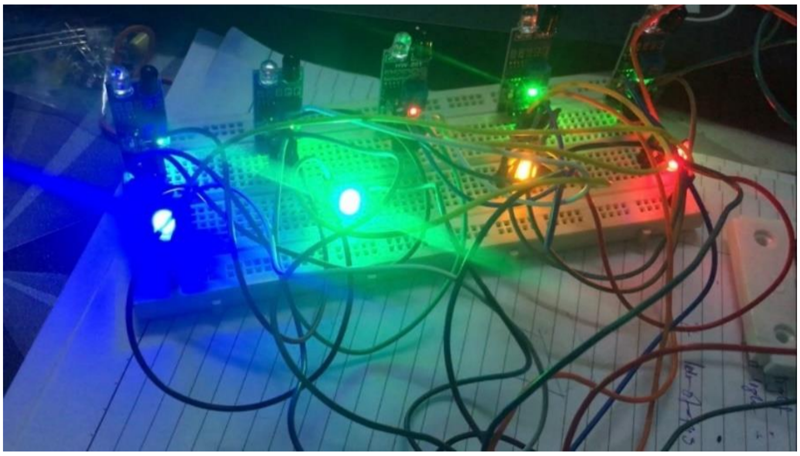

Figure 4.9 Project Testing Development

Figure 4.9 above shows the product testing. Initially, the circuit was developed with PIR Sensor (HC-SR505), LED, LDR, and Resistor and Power Supply. However, after certain research and development, HC-SR505 was changed to IR Sensor. This is because the PIR sensor detects the movement of any object for 360°. It is more relevant to use for the real project but not on a prototype. Also, the PWM method was used for LED intensity. This is to differentiate between 70% and 100% intensity.

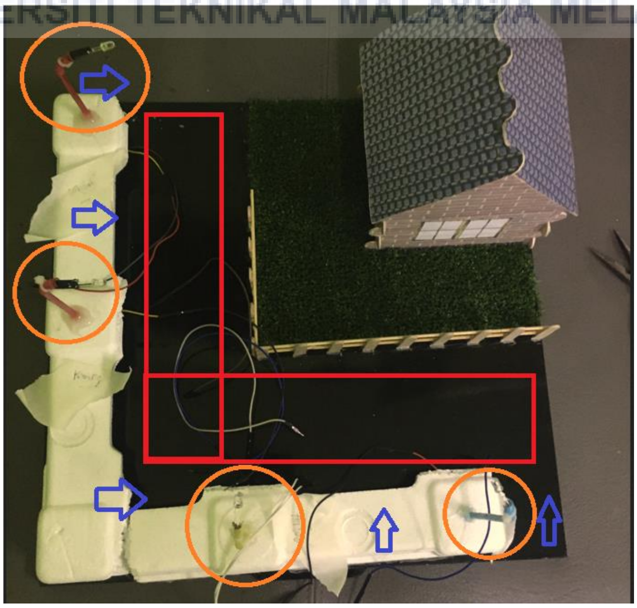

Figure 4.10 Design Prototype

Figure 4.10 above shows the design of the prototype. The prototype was designed for such a small city especially for a home resident. The road was like an L-shape. There were some indicators highlighted in this figure. For example, the red rectangle was for the main road. Blue arrow shape was for sensor place. Lastly, an orange circle was for streetlight placement.

Figure 4.11 System of Prototype

Figure 4.11 above shows the system of a prototype. All the components and sensors were constructed in the project. This was a two-way road. Initially, the streetlight will glow at 70%, suitable for the surrounding area. The flow of the system started when a vehicle met the IR sensor, both streetlights front and back of the vehicle will glow up to 100%. After leaving the IR sensor, both lamps will start to glow at 70% again. This procedure only occurred at night or when the surrounding met the LDR requirement and was not available during the daytime.

4.4 Summary

In summary, the prototype of the Smart Street Lighting System was successfully constructed according to the design in the previous chapter, together with the function of the sensors. This monitoring was managed to function even there were delayed interactions between the transmission module with the developed application. Based on the result that was obtained, the heavy usage of a car detected by the sensor will affect the system and may affect the stability of system flow, especially when run on both ways simultaneously.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter inspects the conclusion for the whole project of the street lighting system and the recommendation for future work and research. The data analysis, procedure, past research, and main component of the system culminate. Future recommendations are being analyzed for more efficient work and functions for the street lighting system project.

5.2 Conclusion

In conclusion, the development of a smart street lighting system is discussed and presented in this paper. Although the advantages of having such a system in place are indisputable and obvious, several issues must be considered before it can be fully implemented. In rural areas with limited activity, the technology will undoubtedly assist the authorities in saving more energy than in cities or towns. However, the energy economy was not the only advantage, as the maintenance schedule can be improved without requiring roving inspection. Finally, aside from ensuring decreased energy usage, the street lighting system will undoubtedly increase the council's efficiency in maintaining the street lights.

5.3 Future Work

There are a few modifications that can be made to this project to increase the accuracy and dependability of the current method. As a result, precise and appropriate component selection can be employed to increase the streetlighting's reliability for future work to keep up with the latest technology. The focus of a future project will be on a complete wireless smart street lighting system with additional security elements to assure secure data flow from sites to the central server.

REFERENCES

[1] I. Allafi and T. Iqbal, "Design and implementation of a low-cost web server using ESP32 for real-time photovoltaic system monitoring," *2017 IEEE Electr. Power Energy Conf. EPEC 2017*, vol. 2017-Octob, pp. 1–5, 2018, DOI: 10.1109/EPEC.2017.8286184.

[2] A. Wibisono and M. Nugraha, "Passive Keyless Entry Locking Door With ESP32," *Ultim. Comput. J. Sist. Komput.*, vol. 12, no. 1, pp. 9–12, 2020, doi: 10.31937/sk.v12i1.1613.

[3] S. S. Priyanka, K. A. Vijay, and M. Chandu, "Using Esp32 IoT Based Wi-Fi Enabled Smart LED Systems," vol. 1, no. April, pp. 1–8, 2021.

[4] N. K. Walia, P. Kalra, and D. Mehrotra, "An IoT by information retrieval approach: Smart lights controlled using WiFi," *Proc. 2016 6th Int. Conf. - Cloud Syst. Big Data Eng. Conflux. 2016*, pp. 708–712, 2016, DOI: 10.1109/CONFLUENCE.2016.7508211.

[5] R. Arabella, P. Anuradha, K. Rajkumar, and S. Yedulapuram, "Smart street light management system for conservation of electrical energy," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 981, no. 3, 2020, DOI: 10.1088/1757-899X/981/3/032041.

[6] M. Saifuzzaman, N. N. Moon, and F. N. Nur, "IoT based street lighting and traffic management system," *5th IEEE Reg. 10 Humanit. Technol. Conf. 2017, R10-HTC 2017*, vol. 2018-Janua, no. December, pp. 121–124, 2018, DOI: 10.1109/R10HTC.2017.8288921.

[7] S. Priya, M. M. Vijayan2, and V. S. Priya, "AUTOMATIC STREET LIGHT CONTROL SYSTEM USING WSN BASED ON VEHICLE MOVEMENT AND ATMOSPHERIC CONDITION." Accessed: Jun. 12, 2021. [Online]. Available: http://www.ijccts.orgwww.ijccts.org.

[8] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Trans. Ind. Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014, DOI: 10.1109/TII.2014.2300753.

[9] L. Powder, "The Application and Development of Multifunctional Particles (II)," vol. 07, no. 3, pp. 91–97, 2010.

[10] "Blynk Program with Multiple WiFi Networks | FactoryForward." https://www.factoryforward.com/blynk-program-multiple-wifi-networks/ (accessed Jun. 18, 2021).

[11] P. Kumar and P. Kumar, "International Journal of Computer Science and Mobile Computing Arduino Based Wireless Intrusion Detection Using IR Sensor and GSM," 2013. Accessed: Jun. 03, 2021. [Online]. Available: www.ijcsmc.com.

APPENDICES

Appendix A Coding

```
#include <BlynkSimpleEsp32.h>

// Template ID, Device Name and Auth Token are provided by the Blynk.Cloud
// See the Device Info tab, or Template settings
#define BLYNK_TEMPLATE_ID           "TMPLeviZnOjE"
#define BLYNK_DEVICE_NAME           "Smart StreetLighting System"
#define BLYNK_AUTH_TOKEN            "hSsg2YFtpxd2dbBq7zfPA5zJpYTMG8lZ"

#define IR1 32
#define IR2 25
#define IR3 34
#define IR4 39
#define IR5 21

#define LED1 19
#define LED2 18
#define LED3 26
#define LED4 27

#define LDR 36

#define OBJECT_DETECT 0
#define NO_OBJECT_DETECT 1

const int TIMEOUT_DELAY = 10000; //milli second *timing delay to turn off led

int proxy1 = NO_OBJECT_DETECT; //inital value, takda object detect
int proxy2 = NO_OBJECT_DETECT;
int proxy3 = NO_OBJECT_DETECT;
int proxy4 = NO_OBJECT_DETECT;
int proxy5 = NO_OBJECT_DETECT;

int led1State = 0; //inital value
int led2State = 0;
int led3State = 0;
int led4State = 0;

int ldrStatus = 0; //inital value

// setting PWM properties
const int freq = 5000;
const int ledChannel1 = 0; // pwm fx / pwm channel 0-15 (16 channel)
const int ledChannel2 = 1;
```

```
const int ledChannel3 = 2;
const int ledChannel4 = 3;

const int resolution = 8; //set resolution

const int LOW_LIGHT = 10; //intensity (guna kat board)
const int MED_LIGHT = 100; //(guna blink)
const int FULL_LIGHT = 200;

char auth[] = BLYNK_AUTH_TOKEN;

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "iPhone m";
char pass[] = "farissiraf123";

WidgetLED widgetLed1(V19); //blink widget
WidgetLED widgetLed2(V18);
WidgetLED widgetLed3(V26);
WidgetLED widgetLed4(V27);

BlynkTimer timer;

int timer1id = -1; //timer id -initial value (-1) so timer takjalan lagi, akan jalan bila
timeid jalan
int timer2id = -1;
int timer3id = -1;
int timer4id = -1;

void turnOnOffLed(const int ledChannel){

 switch (ledChannel) {
     case 1:
       if(proxy1 == OBJECT_DETECT || proxy2 == OBJECT_DETECT){
         ledcWrite(ledChannel1, FULL_LIGHT);
         led1State = FULL_LIGHT;
       }else{
         ledcWrite(ledChannel1, LOW_LIGHT);
         led1State = MED_LIGHT;
       }
       timer1id = -1; //bila timeout tadi habis, so jadi initial balik

       break;
     case 2:
       if(proxy2 == OBJECT_DETECT || proxy3 == OBJECT_DETECT){
         ledcWrite(ledChannel2, FULL_LIGHT);
         led2State = FULL_LIGHT;
       }else{
         ledcWrite(ledChannel2, LOW_LIGHT);
         led2State = MED_LIGHT;
```

```
      }
      timer2id = -1;

      break;
    case 3:
      if(proxy3 == OBJECT_DETECT || proxy4 == OBJECT_DETECT){
        ledcWrite(ledChannel3, FULL_LIGHT);
        led3State = FULL_LIGHT;
      }else{
        ledcWrite(ledChannel3, LOW_LIGHT);
        led3State = MED_LIGHT;
      }
      timer3id = -1;

      break;
    case 4:
      if(proxy4 == OBJECT_DETECT || proxy5 == OBJECT_DETECT){
        ledcWrite(ledChannel4, FULL_LIGHT);
        led4State = FULL_LIGHT;
      }else{
        ledcWrite(ledChannel4, LOW_LIGHT);
        led4State = MED_LIGHT;
      }
      timer4id = -1;
      break;
    default: // cam else function
      break;
  }
}

void runStreetLight(){
  ldrStatus = analogRead(LDR);
  //Serial.println("ldrStatus");
  //Serial.println(ldrStatus);

  if (ldrStatus <= 250) //kalau gelap..
  {
    proxy1 = digitalRead(IR1);
    proxy2 = digitalRead(IR2);
    proxy3 = digitalRead(IR3);
    proxy4 = digitalRead(IR4);
    proxy5 = digitalRead(IR5);

    Serial.print("proxy1: ");
    Serial.print(proxy1);
    Serial.print("        proxy2: ");
    Serial.print(proxy2);
    Serial.print("        proxy3: ");
    Serial.print(proxy3);
    Serial.print("        proxy4: ");
```

```
  Serial.print(proxy4);
  Serial.print("        proxy5: ");
  Serial.println(proxy5);


  if(proxy1 == OBJECT_DETECT || proxy2 == OBJECT_DETECT){

    if(led1State != FULL_LIGHT){
      ledcWrite(ledChannel1, FULL_LIGHT);
      led1State = FULL_LIGHT;
    }

    if(timer1id == -1){
      timer1id = timer.setTimeout(TIMEOUT_DELAY,[]{turnOnOffLed(1);});
    }

  }else{
    if(led1State != FULL_LIGHT && led1State != MED_LIGHT){
      ledcWrite(ledChannel1, LOW_LIGHT);
      led1State = MED_LIGHT;
    }
  }

  Serial.print("led1State: ");
  Serial.println(led1State);

  if(proxy2 == OBJECT_DETECT || proxy3 == OBJECT_DETECT){

    if(led2State != FULL_LIGHT){ //kalau dah detect, tapi tak full light
      ledcWrite(ledChannel2, FULL_LIGHT);
      led2State = FULL_LIGHT;
    }

    if(timer2id == -1){
      timer2id = timer.setTimeout(TIMEOUT_DELAY,[]{turnOnOffLed(2);}); // set
timout led1 untuk balik ke lowlight
    }

  }else{
    if(led2State != FULL_LIGHT && led2State != MED_LIGHT){
      ledcWrite(ledChannel2, LOW_LIGHT);
      led2State = MED_LIGHT;
    }
  }

  if(proxy3 == OBJECT_DETECT || proxy4 == OBJECT_DETECT){

    if(led3State != FULL_LIGHT){
      ledcWrite(ledChannel3, FULL_LIGHT);
      led3State = FULL_LIGHT;
```

```
    }

    if(timer3id == -1){
      timer3id = timer.setTimeout(TIMEOUT_DELAY,[]{turnOnOffLed(3);});
    }

  }else{
    if(led3State != FULL_LIGHT && led3State != MED_LIGHT){
      ledcWrite(ledChannel3, LOW_LIGHT);
      led3State = MED_LIGHT;
    }
  }

  if(proxy4 == OBJECT_DETECT || proxy5 == OBJECT_DETECT){

    if(led4State != FULL_LIGHT){
      ledcWrite(ledChannel4, FULL_LIGHT);
      led4State = FULL_LIGHT;
    }

    if(timer4id == -1){
      timer4id = timer.setTimeout(TIMEOUT_DELAY,[]{turnOnOffLed(4);});
    }

  }else{
    if(led4State != FULL_LIGHT && led4State != MED_LIGHT){
      ledcWrite(ledChannel4, LOW_LIGHT);
      led4State = MED_LIGHT;
    }
  }
 }
 else
 {
  //Serial.println("TURN OFF ALL LED");
  ledcWrite(ledChannel1, 0); // write kat board /led off
  ledcWrite(ledChannel2, 0);
  ledcWrite(ledChannel3, 0);
  ledcWrite(ledChannel4, 0);

  led1State = 0; //simpan state ni // value state save sebab nak hantar ke blynk
  led2State = 0;
  led3State = 0;
  led4State = 0;
 }
}

void sendData(){

 widgetLed1.setValue(led1State);
 widgetLed2.setValue(led2State);
```

```
  widgetLed3.setValue(led3State);
  widgetLed4.setValue(led4State);

  Blynk.virtualWrite(V5, ldrStatus); //send data ke blynk, keep up to date
}

void setup()
{
  Serial.begin(9600);

  pinMode(IR1, INPUT);
  pinMode(IR2, INPUT);
  pinMode(IR3, INPUT);
  pinMode(IR4, INPUT);
  pinMode(IR5, INPUT);

  pinMode(LDR, INPUT);

  // configure LED PWM functionalitites
  ledcSetup(ledChannel1, freq, resolution); //ledc cara setup pwm
  ledcSetup(ledChannel2, freq, resolution);
  ledcSetup(ledChannel3, freq, resolution);
  ledcSetup(ledChannel4, freq, resolution);

  // attach the channel to the GPIO to be controlled
  ledcAttachPin(LED1, ledChannel1);
  ledcAttachPin(LED2, ledChannel2);
  ledcAttachPin(LED3, ledChannel3);
  ledcAttachPin(LED4, ledChannel4);

  Blynk.begin(auth, ssid, pass); //blynk config, kalau takda wifi , dia block

  timer.setInterval(5L, runStreetLight); //every 5 mili sec jalan function street light, keep
loop (L = long value)

  // send data to blynk every 1 sec
  timer.setInterval(1000L, sendData); // every 1 sec send data blynk
}

void loop() // bila setup habis jalan, dia akan jalan void loop
{
  Blynk.run(); // handshake/establish connection dengan blynk server
  timer.run(); // start run timer , counting , run streetlight fucntion


}
```

Appendix B   Example of Appendix B

Task / Progress	Week																											
	BDP 1														BDP 2													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Briefing About BDP 1																												
Confirmation Research Project Title																												
Briefing Chapter 1																												
Background Of Study																												
Problem Statement																												
Objective, Project Scope																												
Briefing On Chapter 2 (Literature Review)																												
Research For Previous Related Project																												
Sorting Important from Previous Project																												
Literature Review Writing																												
Submission Draft Chapter 1, Logbook And Project Progress																												
Methodology																												
Expected Result																												
Submission Draft for Chapter 2,3 And 4																												
Correction And Submission of BDP Report 1																												
Preparation And Presentation of BDP 1																												

Task / Progress	Week																											
	BDP 1														BDP 2													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Briefing On BDP 2																												
Configuration For Hardware Implementation																												
Configuration For Software Implementation																												
Result And Analysis																												
Finalized Report for BDP 2																												
Preparation For Presentation of BDP 2																												
Presentation Of BDP 2																												