LED INTENSITY CONTROLLER USING EPS8266

A PROJECT REPORT SUBMITTED TO

SRM INSTITUTE OF SCIENCE & TECHNOLOGY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

AWARD OF THE DEGREE OF

MASTER OF COMPUTER APPLICATIONS

BY

SUMANTH REDDY B (REG NO. RA2332241010252)

HEMANATHAN J (REG NO. RA2332241010271)

GOPINATH R (REG NO. RA2332241010298)

UNDER THE GUIDANCE

OF

Ms. M. RAMLA M.C.A., M.Phil.,

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

DEPARTMENT OF COMPUTER APPLICATIONS

FACULTY OF SCIENCE AND HUMANITIES

SRM INSTITUTE OF SCIENCE & TECHNOLOGY

Kattankulathur – 603 203

Chennai, Tamil Nadu

OCTOBER – 2024

BONAFIDE CERTIFICATE

This is to certify that the project report titled **"LED INTENSITY CONTROLLER USING EPS8266"** is a bonafide work carried out by SUMANTH REDDY B (RA2332241010252), HEMANATHAN J (RA2332241010271), GOPINATH R (RA2332241010298), under my supervision for the award of the Degree of Master of Computer Applications. To my knowledge the work reported herein is the original work doneby these students.

Ms. M. RAMLA	**Dr. R. JAYASHREE**
Assistant Professor,	Associate Professor & Head,
Department of Computer Applications	Department of Computer Applications
(GUIDE)	

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

With profound gratitude to the ALMIGHTY, I take this chance to thank the people who helped me to complete this project.

We take this as a right opportunity to say THANKS to my parents who are there to stand with me always with the words “YOU CAN”.

We are thankful to **Dr. T. R. Paarivendhar**, Chancellor, and **Prof. A. Vinay Kumar**, Pro Vice- Chancellor (SBL), SRM Institute of Science & Technology, who gave us the platform to establish us to reach greater heights.

We earnestly thank **Dr. A. Duraisamy**, Dean, Faculty of Science and Humanities, SRM Institute of Science & Technology, who always encourage us to do novel things.

A great note of gratitude to **Dr. S. Albert Antony Raj**, Deputy Dean, Faculty of Science andHumanities for his valuable guidance and constant Support to do this Project.

We express our sincere thanks to **Dr. R. Jayashree**, Associate Professor & Head, for her support to execute all incline in learning.

It is our delight to thank our project guide **Ms. M. RAMLA M.C.A., M.Phil.,** Assistant Professor, Department of Computer Applications, for her help, support, encouragement, suggestions, and guidance throughout the development phases of the project.

We convey our gratitude to all the faculty members of the department who extended their support through valuable comments and suggestions during the reviews.

Our gratitude to friends and people who are known and unknown to us who helped in carrying out this project work a successful one.

SUMANTH REDDY B

HEMANATHAN J

GOPINATH R

Ramla Sha

IOT Project-LED

MCA-2024

MCA

SRM Institute of Science & Technology

Document Details

Submission ID

trn:oid:::1:3057829040

Submission Date

Oct 28, 2024, 11:58 AM GMT+5:30

Download Date

Oct 28, 2024, 11:59 AM GMT+5:30

File Name

MCA_E_IOT.pdf

File Size

7.2 KB

1 Page

202 Words

1,389 Characters

0% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

- **0** Not Cited or Quoted 0%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

0% Internet sources

0% Publications

0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **0** Not Cited or Quoted 0%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

0% Internet sources
0% Publications
0% Submitted works (Student Papers)

ABSTRACT

This project focuses on "LED INTENSITY CONTROLLER USING EPS8266" development and implementation of Energy-Efficient LED Controllers designed to optimize LED lighting systems for enhanced energy efficiency and illumination quality. As global sustainability efforts intensify, these controllers play a crucial role in transitioning from traditional lighting solutions to advanced, eco-friendly alternatives that align with energy conservation goals. The study explores key features of LED controllers, including dimming capabilities, smart scheduling, and integration with renewable energy sources. By intelligently managing power consumption, these controllers not only reduce operational costs but also contribute to significant energy savings across residential, commercial, and industrial applications. Furthermore, the project highlights the economic advantages of adopting LED controllers, such as lower electricity bills and extended product lifespan, leading to reduced maintenance costs.Beyond financial benefits, the impact of these controllers on improving lighting quality is emphasized, showcasing their ability to create healthier and more productive environments. In- depth analysis of design principles, operational benefits, and real-world case studies will illustrate the transformative potential of energy-efficient LED controllers. By embracing these innovative technologies, this project aims to contribute to a greener, more energy-conscious future, underscoring the vital role of smart lighting solutions in promoting sustainability and energy independence.

TABLE OF CONTENTS

CHAPTER 1

1. INTRODUCTION

An Energy-efficient LED controllers are designed to optimize LED lighting systems enhancing energy efficiency while ensuring high-quality illumination. These controllers intelligently manage power consumption, leading to reduced operational costs and significant energy savings. Features such as dimming, smart scheduling, and integration with renewable energy sources make them suitable for various applications in residential, commercial, and industrial settings. In a sustainability-focused, these controllers facilitate the transition to advanced, eco- friendly lighting solutions that conserve energy and align with global to reduce carbon footprints. They offer economic benefits through lower electricity bills, extended lifespan, and reduced maintenance costs. Beyond cost savings, energy LED controllers improve quality of life by providing optimal lighting conditions that enhance visibility and safety. They contribute healthier and more productive environments across different settings. This introduction sets the stage for a deeper exploration of their design principles, benefits, and real-world applications, highlighting their role in creating a greener, energy-conscious future. By embracing these technologies, we move toward a more sustainable world.

CHAPTER 2

2. WORKING ENVIRONMENT

2.1 Software Requirements

COMPONENT	SPECIFICATION
IDE	Arduino IDE
Library for Arduino	Install the Blynk library in your Arduino IDE
Blynk App	Install on your smartphone

2.2 Hardware Requirements

COMPONENT	SPECIFICATION
Operating System	Windows 10 or newer
Processor	Dual-core 2.0 GHz or higher
RAM	4 GB or more
Hard Disk	64 GB or more
Network	Ethernet connection (LAN) or Mobile internet

2.3 COMPONENT LISTS:

Name	Quantity	Component
U1	1	ESP8266
R1	1	100 kΩ Resistor
R2	1	510 Ω Resistor
R3	1	Photoresistor
T1	1	NPN Transistor (BJT)
C1	1	0.1 uF Capacitor
L1	1	Light bulb

Figure.1 Components of LED Intensity Controller

CHAPTER 3

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system for LED control was primarily manual, relying on direct control where users turned LEDs on or off without any energy-saving features. This approach led to inefficient energy usage, particularly since dimming could significantly optimize power consumption. With no automatic adjustments, LEDs were either fully on or off, resulting in wasted energy during periods when lower intensity would suffice. Furthermore, these systemslacked connectivity options, restricting users to physical controls and hindering any form of remote operation or monitoring. This disconnection limited the implementation of effective power management strategies, resulting in significant energy waste over time. As technologyhas evolved, the demand for smarter lighting solutions has grown. Systems like the LED intensity controller using the ESP8266 effectively address these limitations. By incorporatingautomation, users can now adjust brightness levels based on ambient light conditions or personal preferences, greatly enhancing energy efficiency and user comfort. This capability allows users to create tailored lighting environments for various activities, such as reading, working, or relaxing, ensuring that lighting is always optimal. The integration of remote- control capabilities further revolutionize the user experience, allowing management of lighting systems from anywhere. Whether users are at home, at work, or on vacation, they caneasily monitor and adjust their lighting, optimizing power usage even when they are away. Additionally, features such as scheduling and timers enable users to automate lighting patterns, further reducing energy consumption during off-peak hours. These advancements not only improve convenience but also promote sustainable energy practices by minimizing unnecessary energy consumption. They contribute to the growing trend of smart home technologies, which prioritize energy efficiency and user engagement. This underscores the urgent need for modern solutions in LED control systems that align with today's expectationsfor convenience, efficiency, and environmental responsibility. Ultimately, the shift from manual to automated systems marks a significant step forward in how we interact with and manage our lighting environments, paving the way for more sustainable living.

3.2 DRAWBACKS OF THE EXISTING SYSTEM

The LED intensity controller using ESP8266 faces several challenges. Firstly, it has limited connectivity and control, lacking remote access and IoT features that hinder efficient monitoring. Additionally, there is higher power consumption, with inefficient usage during both standby and operational modes. The system also shows limited sensor integration, making it unable to adjust LED brightness based on environmental factors like light or motion detection. Furthermore, it requires manual or limited automation, as it demands manual adjustments and lacks smart automation for energy-saving features. The complex setup and maintenance can be daunting, especially for non-technical users. Moreover, it presents limitedscalability, making it difficult to expand or add additional devices without complicated rewiring. Finally, there are security issues, with potential vulnerabilities arising from the absence of modern security protocols for connected systems.

These challenges can significantly hinder the overall user experience and functionality of theLED intensity controller. Addressing these issues is crucial for enhancing efficiency and ensuring ease of use. Improved connectivity and automation features could facilitate better monitoring and control, while enhanced scalability would allow for a more flexible and adaptable system. Additionally, implementing robust security measures is essential to protectagainst potential cyber threats, ensuring the system remains reliable and secure in an increasingly connected environment. As technology evolves, incorporating these improvements will be vital for the future development of smart LED systems.

3.3 PROPOSED SYSTEM

The LED intensity controller using the ESP8266 and Blynk app marks a major leap forward from older manual LED control systems. By overcoming the limitations of traditional setups, this system introduces automation and connectivity that significantly enhance user experience and energy efficiency. With advanced technology, users can now automate brightness adjustments based on ambient light levels or individual preferences, ensuring that lighting conditions are always optimal for any situation, whether for work, relaxation, or entertaining guests. This feature not only improves comfort but also minimizes energy consumption during times when full brightness is unnecessary, leading to substantial energy savings over time. The system's dimming capabilities allow for precise control over LED intensity, enabling users tofine-tune their lighting to match specific activities or moods. This level of customization conserves energy and extends the lifespan of the LEDs, ultimately reducing replacement costs. Additionally, the integration of the Blynk app facilitates remote control of the lighting systemfrom anywhere, allowing users to adjust settings even when they are away from home. This convenience empowers users to ensure their spaces are always well-lit or to turn off lights that were inadvertently left on, contributing to further energy savings. Users can manage LED settings conveniently via their smartphones, offering flexibility that was previously lacking inmanual systems. The app also provides real-time monitoring, enabling users to track energy usage, set schedules, and receive alerts about their lighting consumption. This feature empowers users to make informed decisions about power management, encouraging more responsible energy use. Beyond improving user interaction, the proposed system promotes sustainable energy practices by reducing unnecessary consumption and aligning with environmentally conscious initiatives. This innovative solution not only meets the growing demand for smart lighting but also supports global efforts toward energy efficiency, highlighting the critical role of technology in transforming modern LED control systems intosmarter, more responsive solutions for today’s energy-conscious consumers. Ultimately, the combination of automation, connectivity, and user empowerment sets a new standard in lighting technology, paving the way for future advancements in smart home integration.

3.4 ADVANTAGES OF THE PROPOSED SYSTEM

- The LED intensity controller using ESP8266 offers several significant advantages. Firstly, it promotes energy savings by reducing electricity consumption through the adjustment of LEDbrightness based on real-time environmental conditions.
- This capability not only lowers energy costs but also contributes to sustainability efforts, making it an environmentally friendly choice. Additionally, the system provides remote control, allowing users to manage lighting via a mobile app or web interface, which enhances convenience and accessibility.
- This feature isparticularly useful for users who want to control their lighting from anywhere, ensuring that they can adjust settings even when they are away from home.
- Moreover, the controller easily integrates with IoT systems and other smart home devices, facilitating enhanced functionality and seamless data sharing.
- This integration allows for advanced automation scenarios, such as syncing lights with motion sensors or environmental data. Its cost-effectiveness is another key benefit, as it utilizes low-cost components and open- source software, making it an affordable solution for energy management without sacrificing quality.
- Finally, the system's scalability allows for the expansion to control multiple LED fixtures or integration into larger smart home systems, providing flexibility for future enhancements. Users can start with a single unit and gradually build a more extensive system as needed, ensuring that their investment remains relevant over time.
- These features make the LED intensity controller a compelling option for modern lighting solutions, catering to the needs of both tech-savvy users and those looking for simple, effective energy management. The combination of energy efficiency, user convenience, and adaptability positions this controller as a forward-thinking solution in the realm of smart lighting.

CHAPTER 4

4. SYSTEM DESIGN

USE CASE DIAGRAM

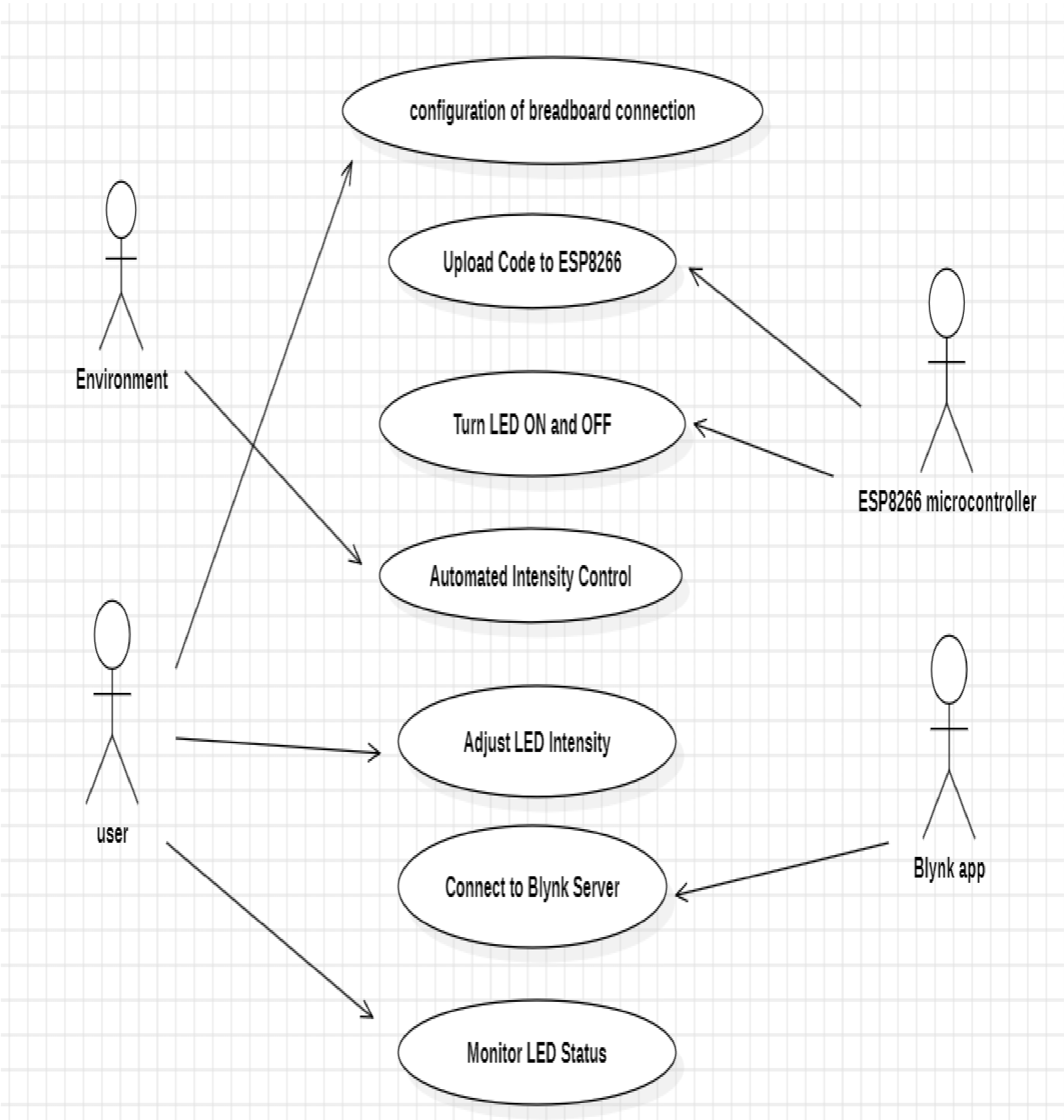


Figure.2 Use case of LED-Intensity Controller

CLASS DIAGRAM

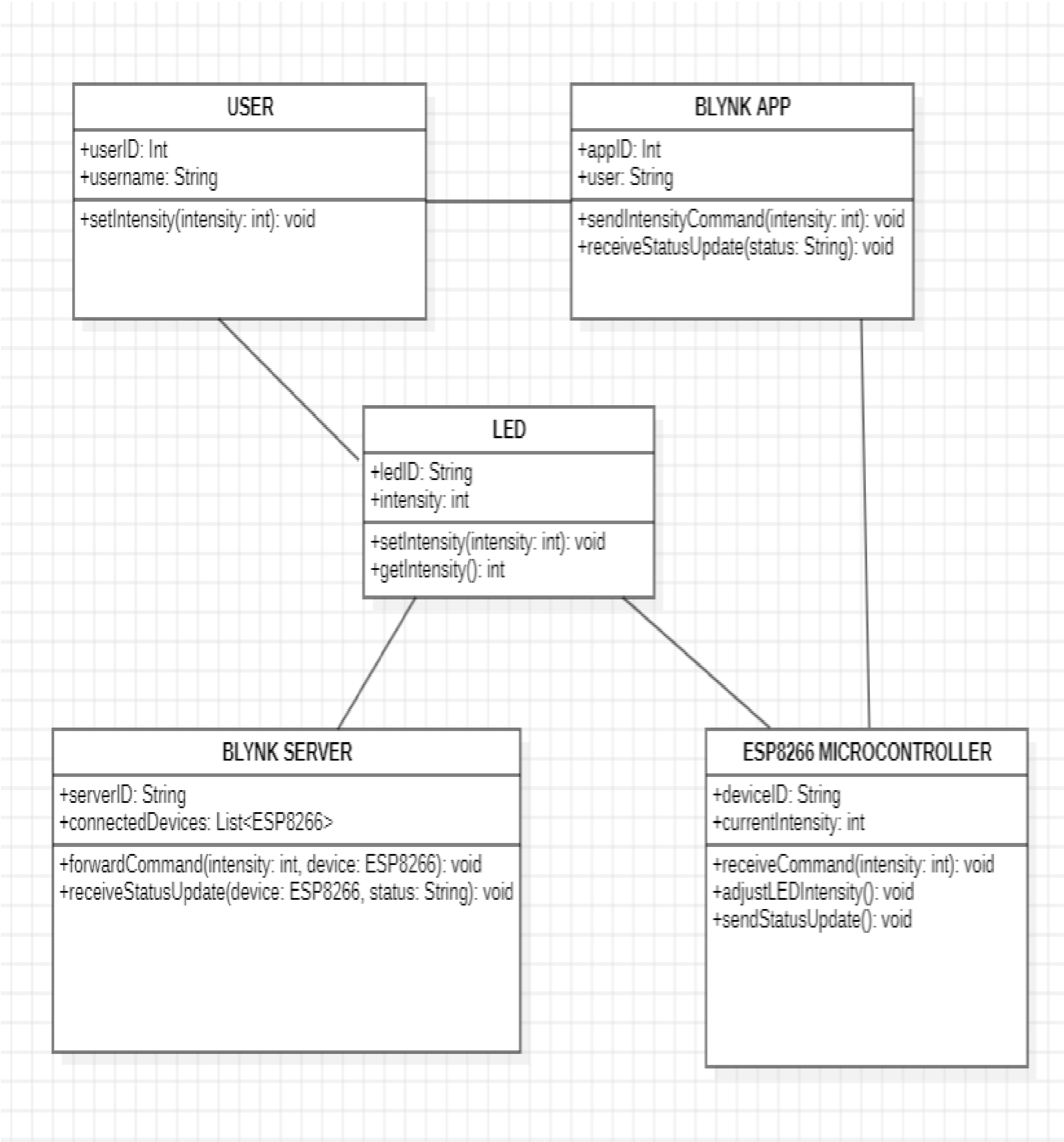


Figure.3 Class Diagram of Led Intensity Controller

SEQUENCE DIAGRAM

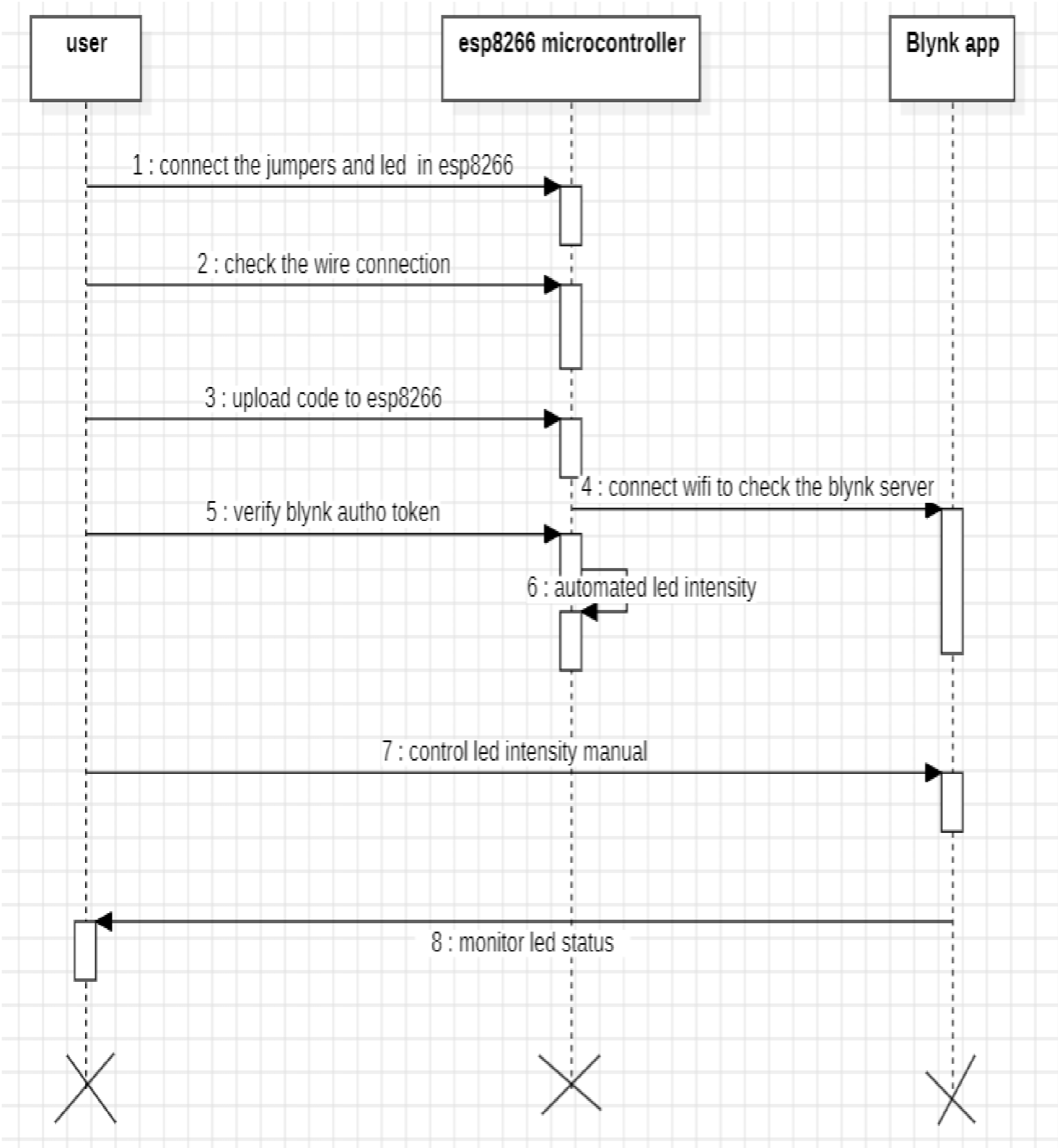


Figure.4 Sequence diagram for LED Intensity controller

FLOWCHART DIAGRAM

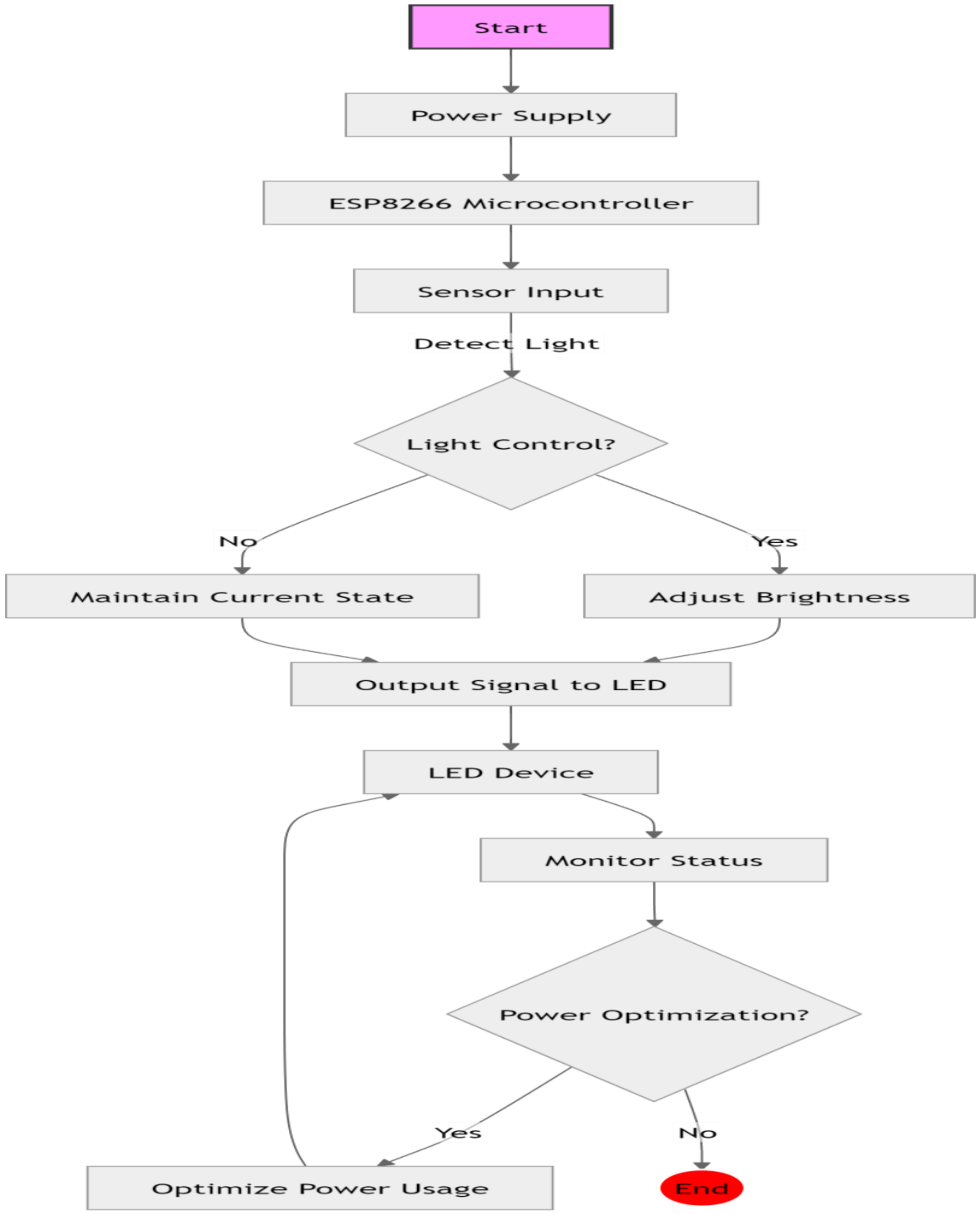


Figure.5 Flow chart of LED Intensity controller

CIRCUIT DIAGRAM

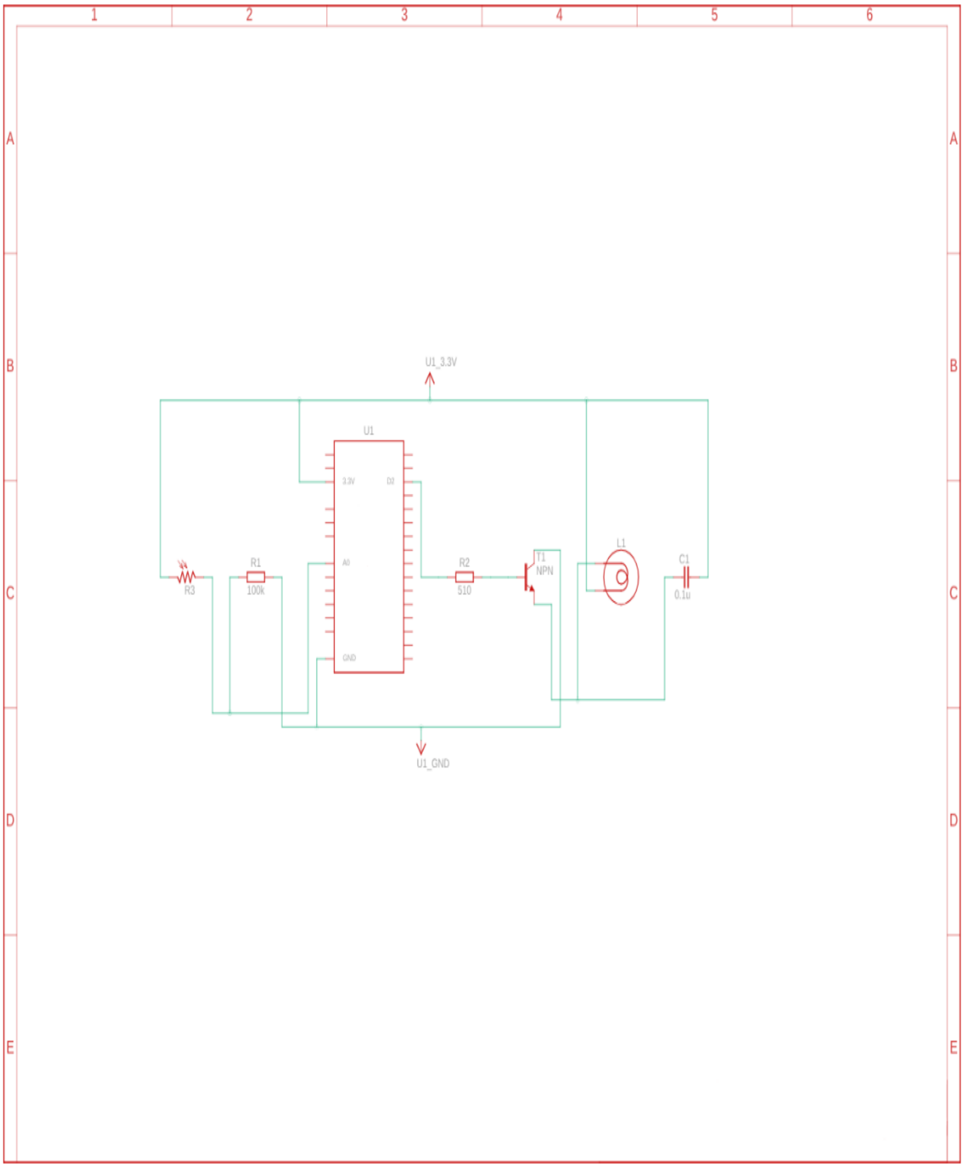


Figure .6 Circuit diagram of LED Intensity controller

ACTIVITY DIAGRAM

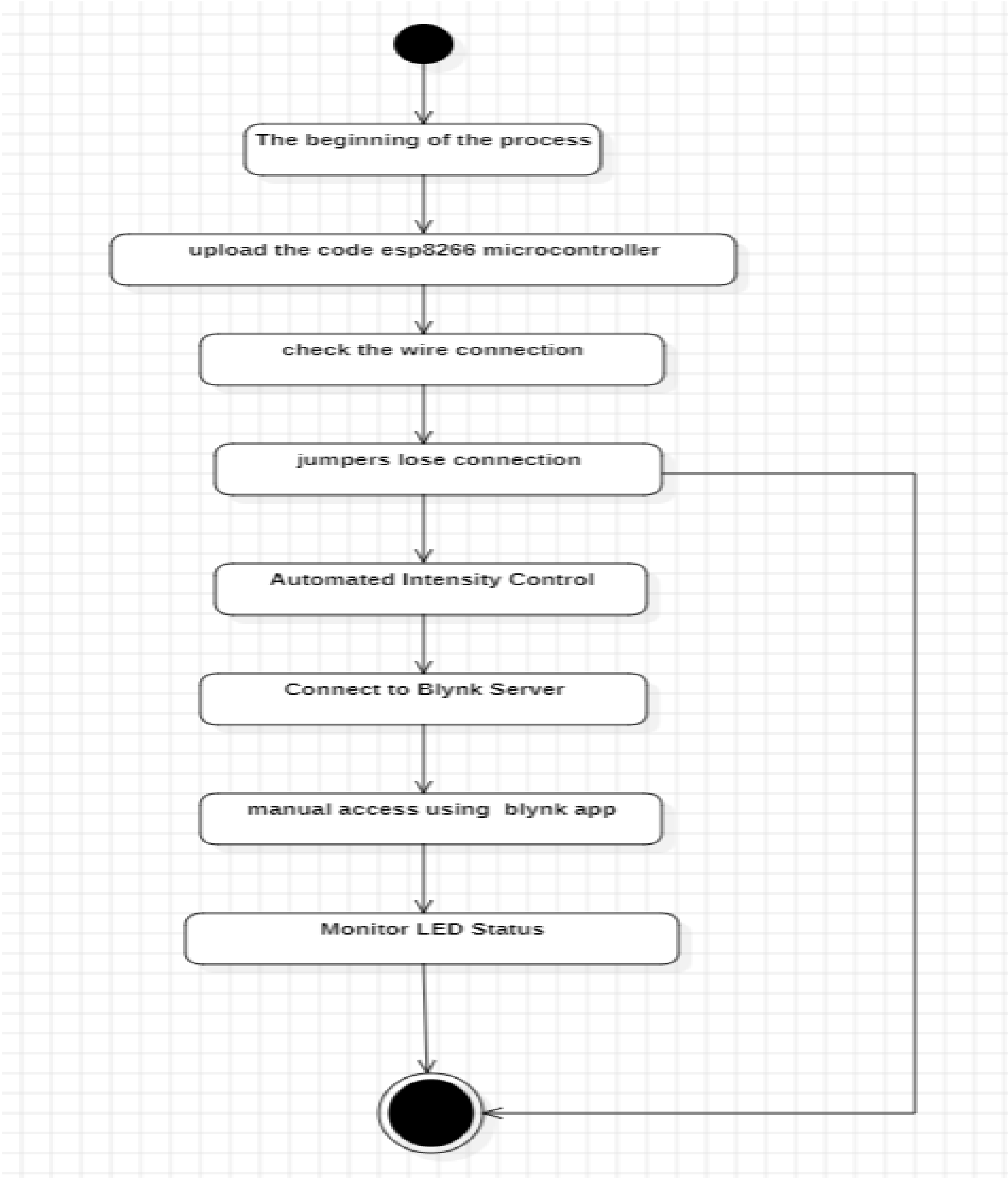


Figure .7 Activity diagram of LED Intensity controller

CHAPTER 5

5. PROJECT METHODOLOGY

5.1 BLYNK CONCEPT

A notable application of Blynk's platform is a **LED light intensity controller** using theESP8266 microcontroller. This project showcases how the Blynk platform can be used tocontrol and adjust LED brightness remotely, leveraging real-time control and data insightsthat Blynk provides. In this setup, the ESP8266 connects to the Blynk app via Wi-Fi,allowing users to control the intensity of an LED directly from their smartphone or webdashboard. With the Blynk app, users can implement a slider widget that adjusts thebrightness level of the LED based on pulse-width modulation (PWM) signals sent from theESP8266. This enables smooth, fine-grained control of the LED's intensity, making it idealfor creating ambient lighting solutions, energy-efficient lighting systems, or mood lighting. The project is highly customizable and can be expanded to control multiple LEDs or othertypes of devices, all through the same interface. Users can also monitor real-time data, log usage patterns, and set up automated triggers or notifications based on specific lightingconditions, such as turning off the lights when ambient light is detected or when certain hours are reached. The ESP8266, combined with Blynk's platform, makes this project highlyaccessible and scalable, whether it's for home automation, decorative lighting, or energymanagement. Blynk's API and compatibility with numerous hardware types make it adaptable across diverse sectors, including smart homes, industrial automation, healthcare, energy management, and more. Its cloud-based infrastructure supports scalable growth, offering reliable performance for projects of any size, while Blynk's community resources and documentation ensure developers have the support they need to bring innovative IoT solutions to life.

BLYNK CONSOLE

Blynk Console is a feature-rich web application catering to different types of users.

1. Configuration of connected devices on the platform, including application settings.
2. Device, data, user, organization, and location management.
3. Remote monitoring and control of devices

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

There are three major components in the platform:

- Blynk App - allows to you create amazing interfaces for your projects using variouswidgets we provide.
- Blynk Server - responsible for all the communications between the smartphone andhardware.
- Blynk Libraries - for all the popular hardware platforms - enable communication withthe server and process all the incoming and outcoming commands.

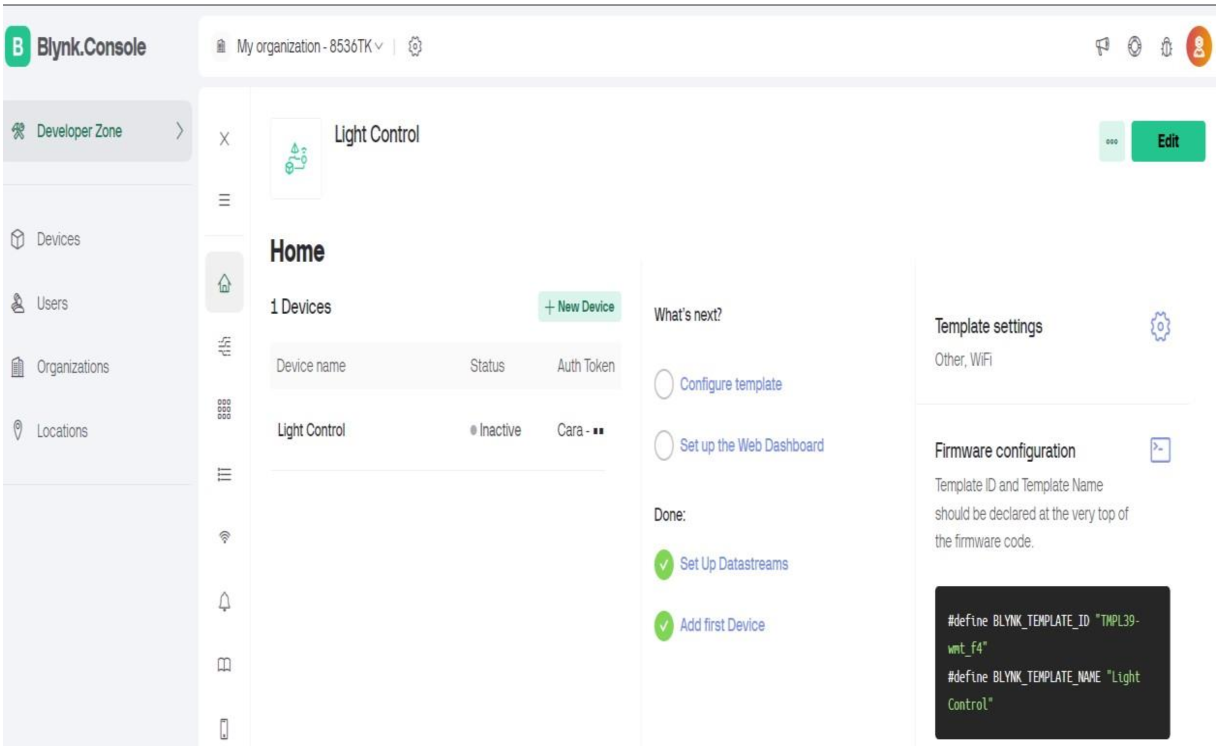


Figure.8 console of Blynk application

Blynk Requirements

1. Hardware.

Blynk works over the Internet. This means that the hardware you choose should be able to connect to the internet. Some of the boards, like Arduino Uno will need an Ethernet or Wi-Fi Shield to communicate, others are already Internet-enabled: like the ESP8266, Raspberry Pi with WIFI dongle, Particle Photon or Spark Fun Blynk Board. But even if you don't havea shield, you can connect it over USB to your laptop or desktop (it's a bit more complicatedfor newbies, but we got you covered). that the list of hardware that works with Blynk is huge and will keep on growing.

2. A Smartphone.

The Blynk App is a well-designed interface builder. It works on both iOS and Android

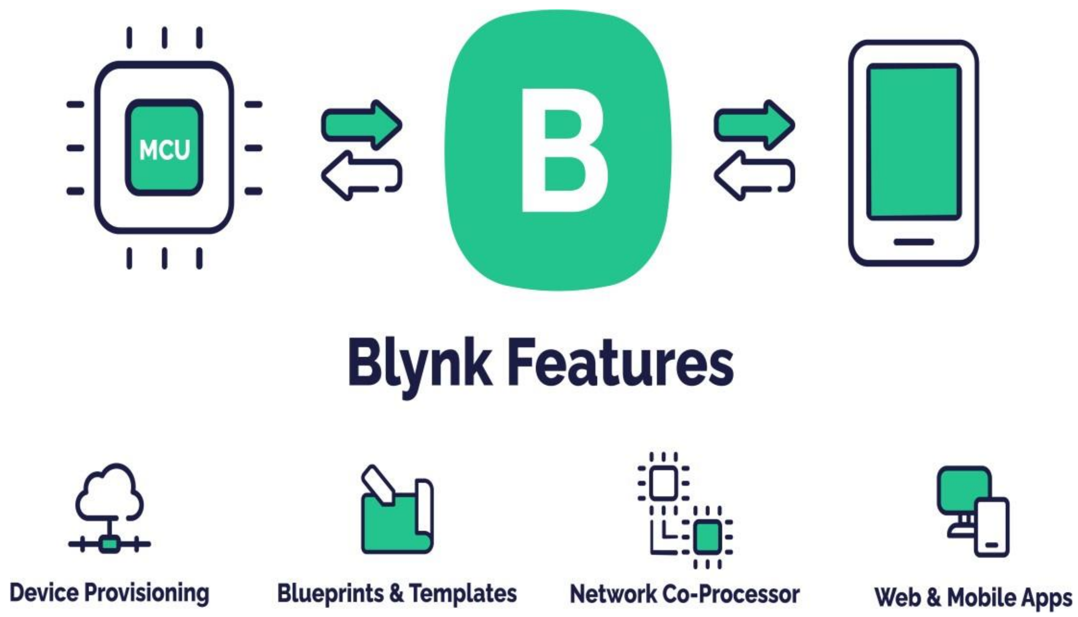


Figure.9 Features of Blynk application

Blynk Application

Blynk App is a versatile native iOS and Android mobile application that serves these majorfunctions

1. Remote monitoring and control of connected devices that work with Blynk platform.
2. Configuration of mobile UI during prototyping and production stages.
3. Automation of connected device operations.

Applications made with Blynk are ready for the end-users. Whether they are family members, employees, or product purchasers, they can easily download the app, connect their devices, and start using them.

Blynk also offers a white-label solution as part of the Business Plan, allowing you to customize the app with your company logo, app icon, theme, colours, and publish it on AppStore and Google Play under your company's name. These customized apps will work seamlessly with your devices.

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = "YourAuthToken";
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

void setup()
{
  Blynk.begin(auth, ssid, pass);
}

void loop() {
  Blynk.run();
}
```

Figure.10 Code for WIFI connection

5.2 PROJECT OVERVIEW

Step 1: Node MCU Board

The NodeMCU (Node Microcontroller Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds. The NodeMCU offers a variety of development environments, including compatibility with the Arduino IDE (Integrated Development Environment). The NodeMCU/ESP8266 community took the IDE selection a step further by creating an Arduino add-on. If you’re just getting started programming the ESP8266 or even an established developer, this is the highly recommended environment. Visit our dedicated page on setting up and configuring the Arduino IDE for a NodeMCU ESP8266.

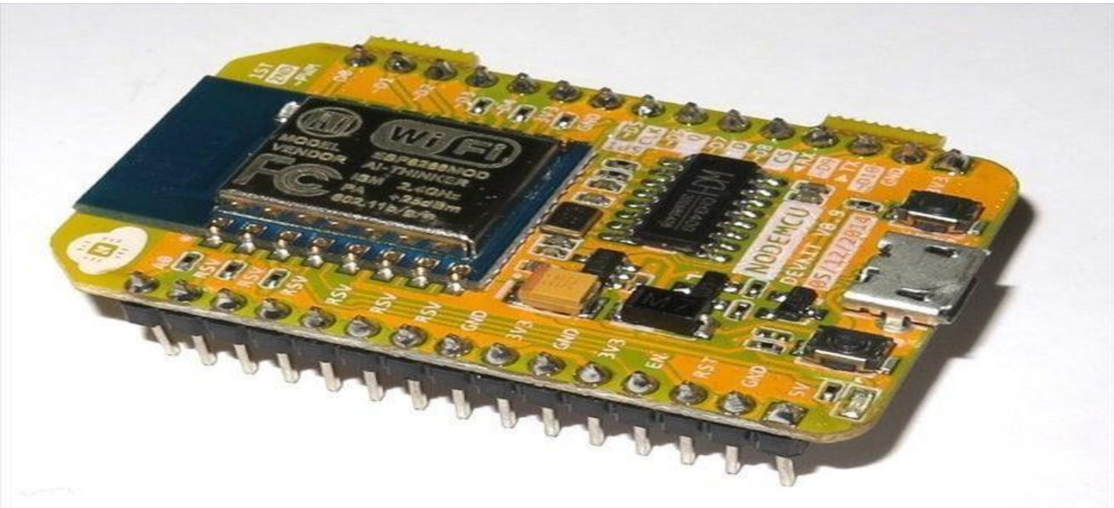


Figure.11 Picture of ESP8266

The ESP8266-based Development Kit is a versatile and accessible choice for IoT enthusiasts, providing an open-source, interactive platform that simplifies the creation of connected devices. With built-in Wi-Fi capability, this board enables seamless internet connectivity, making it a go- to option for projects requiring remote monitoring, control, and data collection. The board's low cost and ease of use make it ideal for students, hobbyists, and professionals alike who seek to prototype IoT solutions quickly. Equipped with NodeMCU firmware, this development kit supports Lua and Arduino programming environments, offering flexibility in coding and compatibility with a vast range of sensors and modules. Its integration of multiple interfaces, such as GPIO (General Purpose Input Output), PWM (Pulse Width Modulation), IIC (Inter- Integrated Circuit), 1-Wire, and ADC (Analog to Digital Converter), makes it adaptable to a wide variety of applications, including environmental monitoring, automation, smart home devices, and wearable technology.

The projects involving components operating on 5V logic, the board's 3V logic system requires careful consideration, often necessitating voltage level shifters to ensure compatibility and protect the board from potential damage. The 30-pin layout offers an organized and efficient setup, with each pin serving dedicated functions like communication, power management, andsensor connectivity. Additionally, the ESP8266's compact size and low power consumption make it well-suited for battery-operated projects, allowing for deployment in remote areas or compact enclosures. With access to a wide community of developers and extensive documentation, the ESP8266 Development Kit provides a robust foundation for building scalable IoT projects, supporting experimentation and innovation in fields like smart agriculture, healthcare, industry automation, and more. This board not only speeds up prototyping but also offers a reliable platform for transitioning from concept to production- ready solutions.

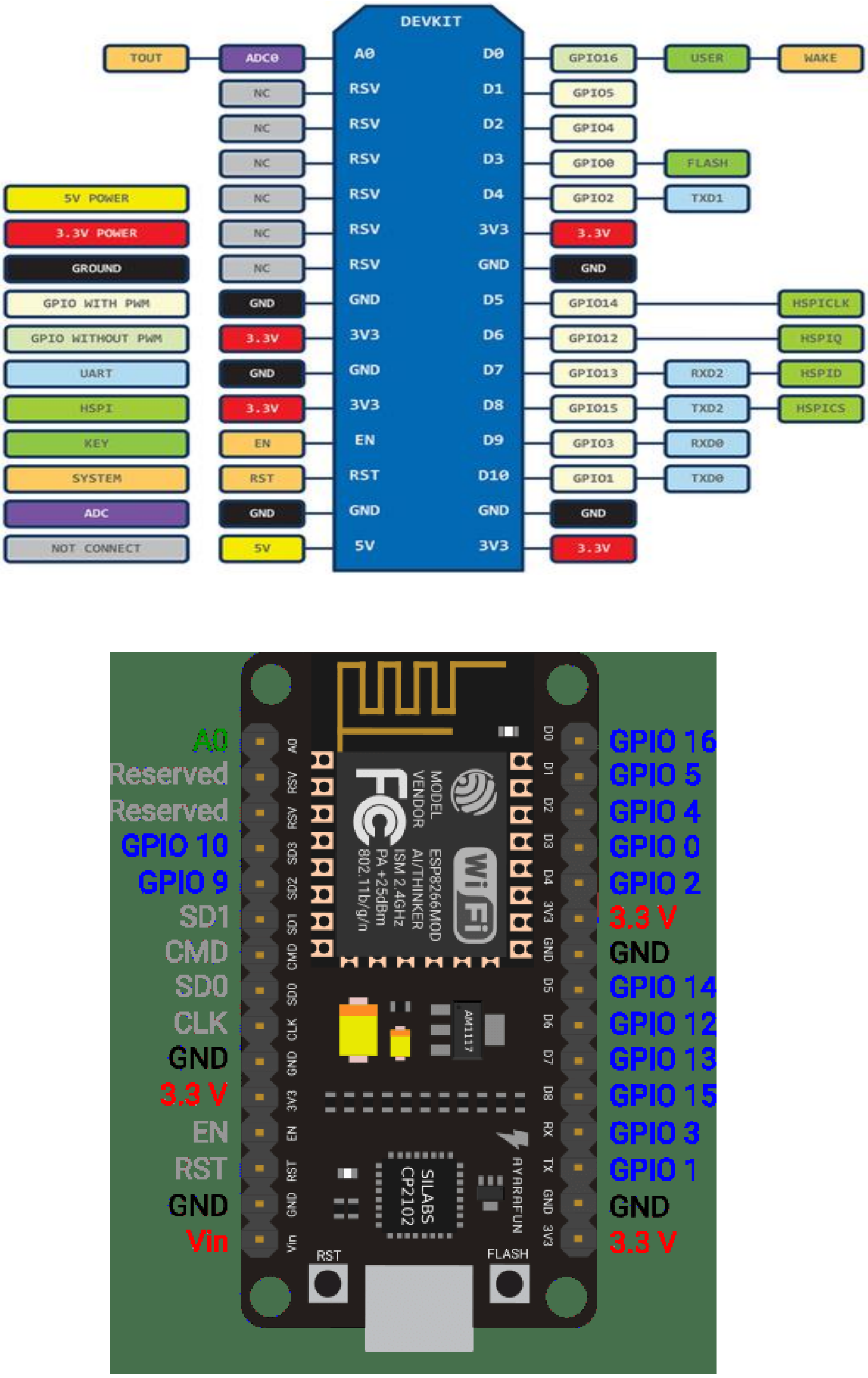


Figure.12 Pin diagram of ESP8266

Step 2: Blynk platform

Blynk is an innovative platform designed for controlling hardware like Arduino and ESP8266over the Internet, making it a key player in the Internet of Things (IoT) landscape. With Blynk, users can remotely manage their devices, display sensor data, and visualize or store information through a user-friendly digital dashboard. The platform allows for rapid development, enabling users to create custom interfaces by simply dragging and dropping various widgets. Importantly, Blynk is hardware-agnostic, meaning it supports a wide range of boards and canconnect to the Internet via Wi-Fi, Ethernet, or other means.

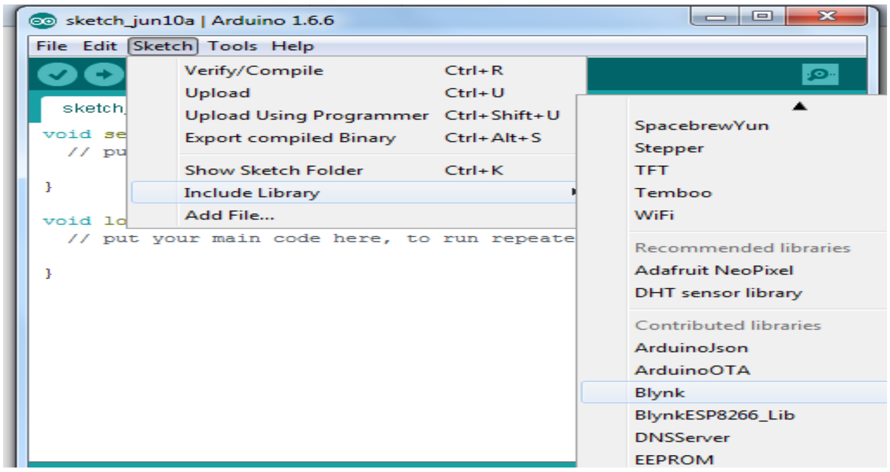
The Blynk ecosystem consists of three main components: the Blynk App, which facilitates thecreation of interfaces; the Blynk Server, responsible for communication between the app and hardware—available as a cloud service or a locally hosted solution; and the Blynk Libraries, which enable seamless interaction with the server across various platforms. With features suchas intuitive design, support for multiple development boards, and direct pin manipulation without extensive coding, Blynk stands out as an accessible and flexible tool for anyone looking to bring their IoT projects to life. It empowers users to easily integrate and expand functionality, making it a popular choice for both beginners and experienced developers in theIoT space.

Figure.13 Platform of Blynk Application

Step 3: Materials Required

1. NodeMCU ESP8266 12E Development Board
2. Smartphone with Blynk App (Make sure it's charged for uninterrupted use)
3. 1-Watt LED with 330-ohm Resistor
4. Additional Resistors
5. Transistor
6. Breadboard
7. Arduino IDE v1.6.6

Step 4: Setting Up Blynk with Arduino IDE



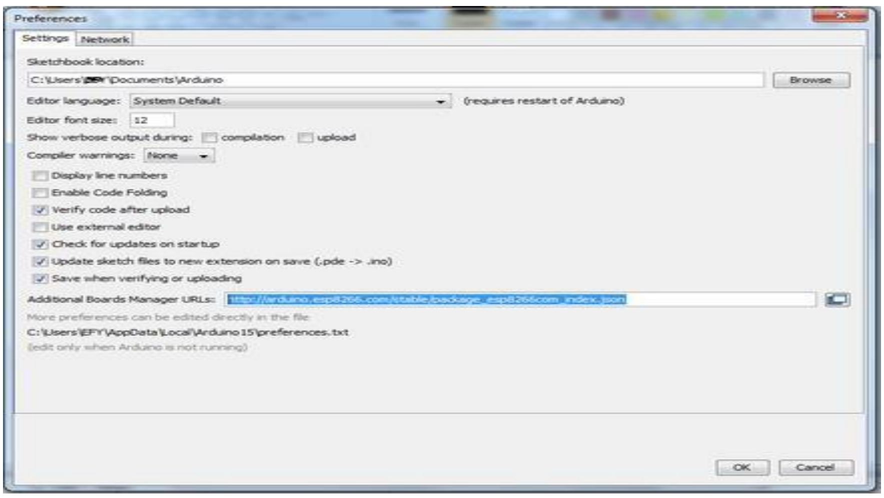


Figure .14 Interface setup of Arduino IDE

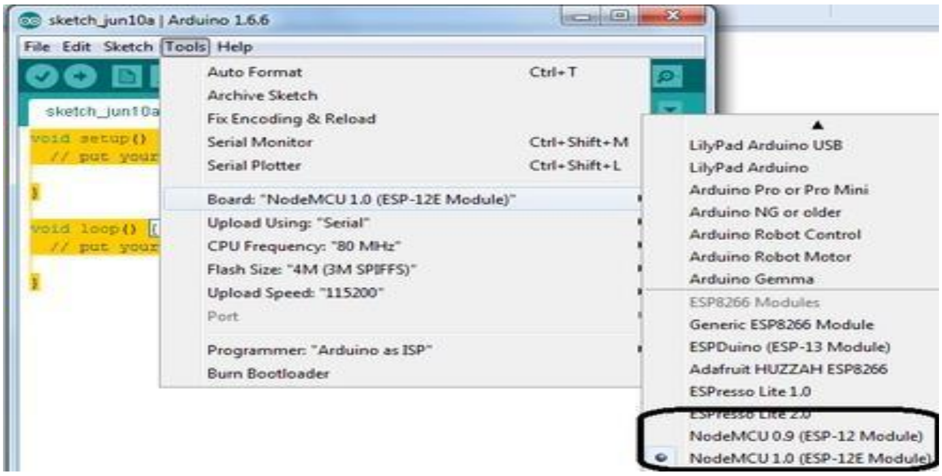


Figure.15 Setting up of ESP8266

Configure the NodeMCU Board

- Open Arduino IDE, go to **File > Preferences**.
- Copy this URL and paste it into **Additional Boards Manager URLs**:http://arduino.esp8266.com/stable/package_esp8266com_index.jn
- Restart Arduino IDE.
- Go to **Tools > Board** and select **NodeMCU 0.9** (or your board version).

Step 5: Setting Up Blynk

1. First install the Blynk app from google play store and then sign in.

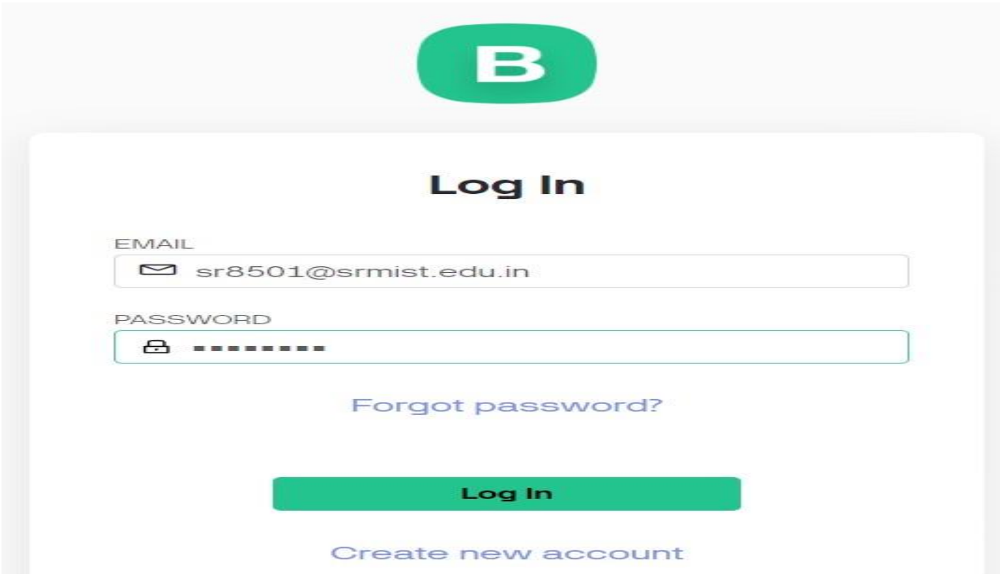


Figure.16 Login page of Blynk Application

2. After that Press on click on New Project and you will get a screen.

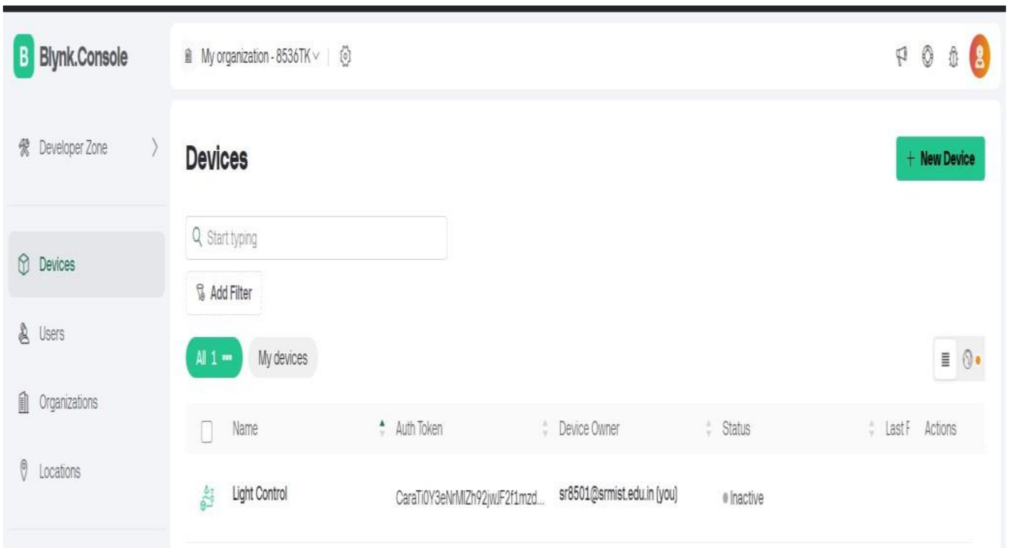


Figure.17 interface of blynk

3. Click the right-side corner New Temple

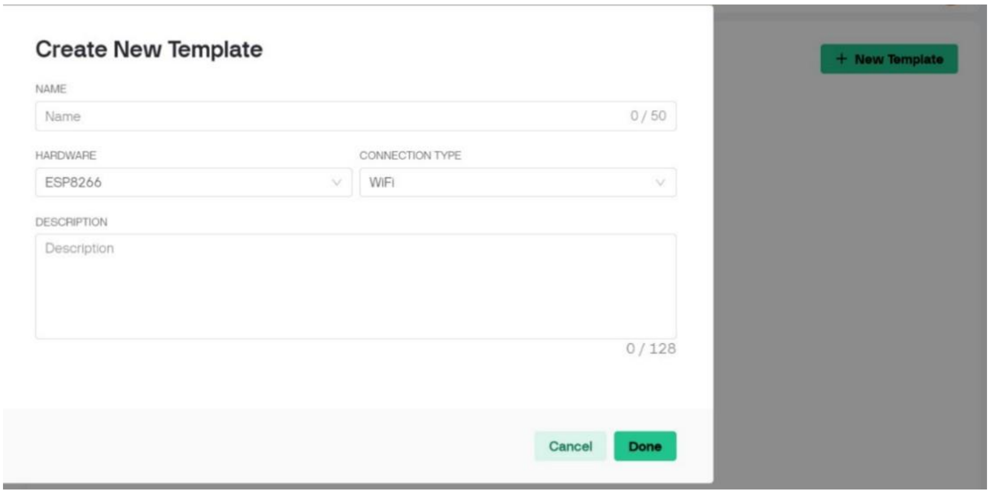
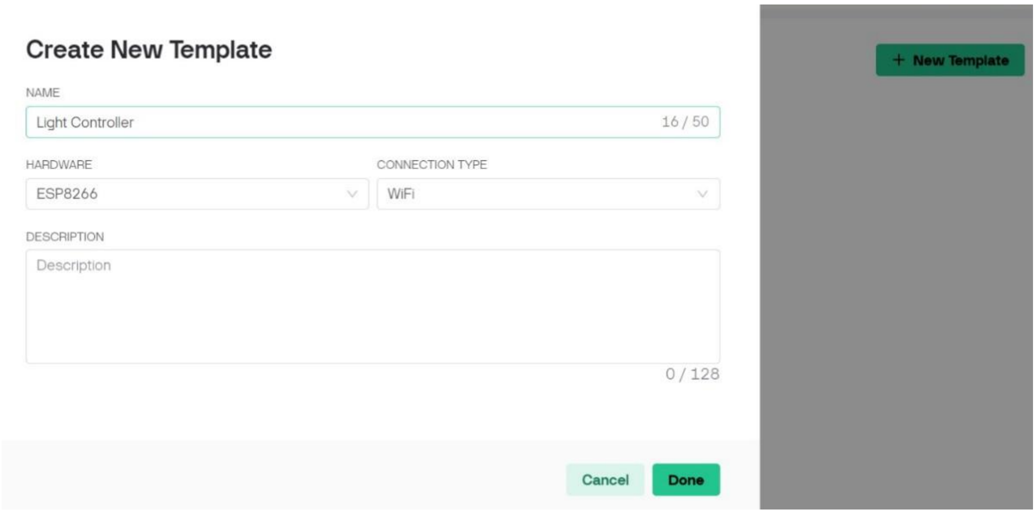


Figure.18 Creation of Template

4. After that give the name for new temple and done.



5. Add new data streams that connect to your device's virtual pins or other sources of data.

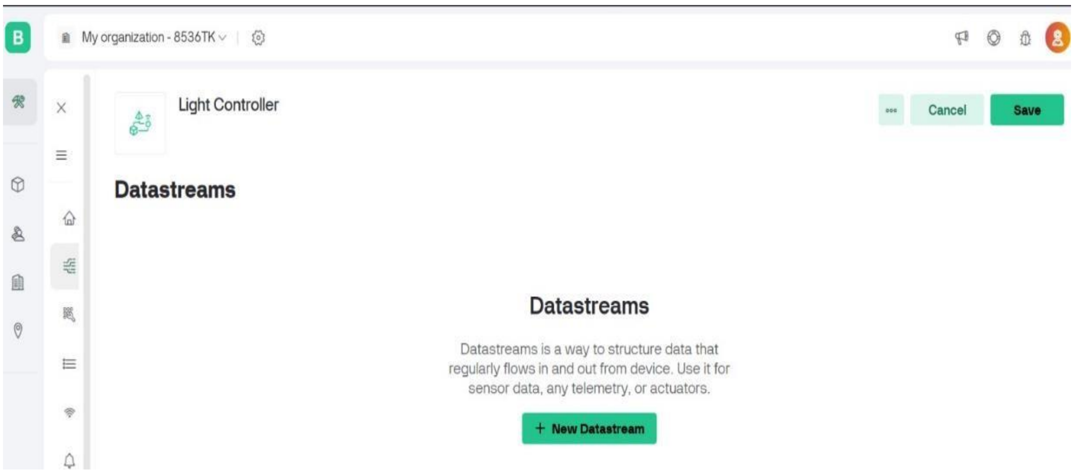


Figure.19 DataStream for connection

6. Choose the **Type** of data stream, such as "Virtual Pin," "Digital Pin," or "Analog Pin.". Setthe **Pin** you want to associate with this data stream.

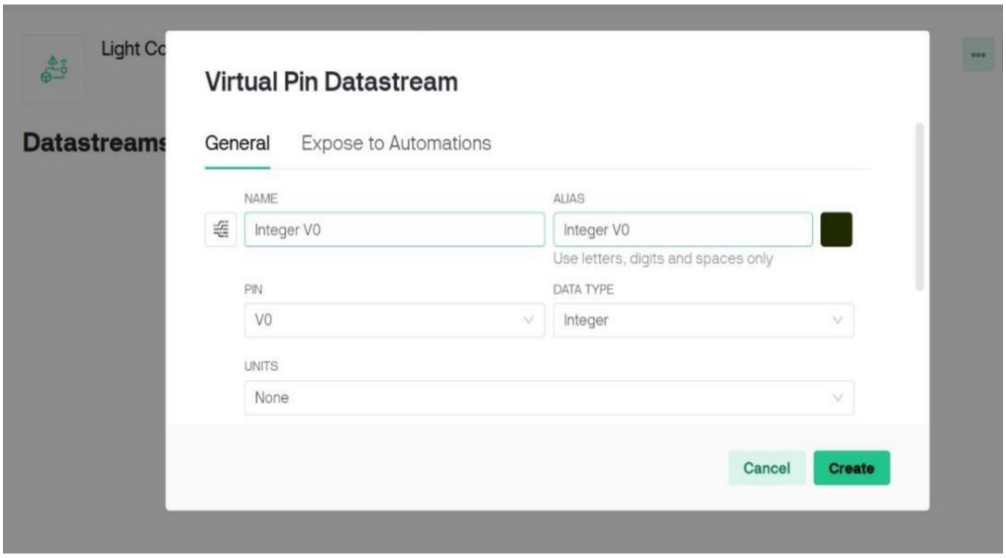


Figure.20 selection of pins

8. Confirm that data is updating as configured and that the **Name**, **Color**, and **Data Type displayed** correctly on the widget.

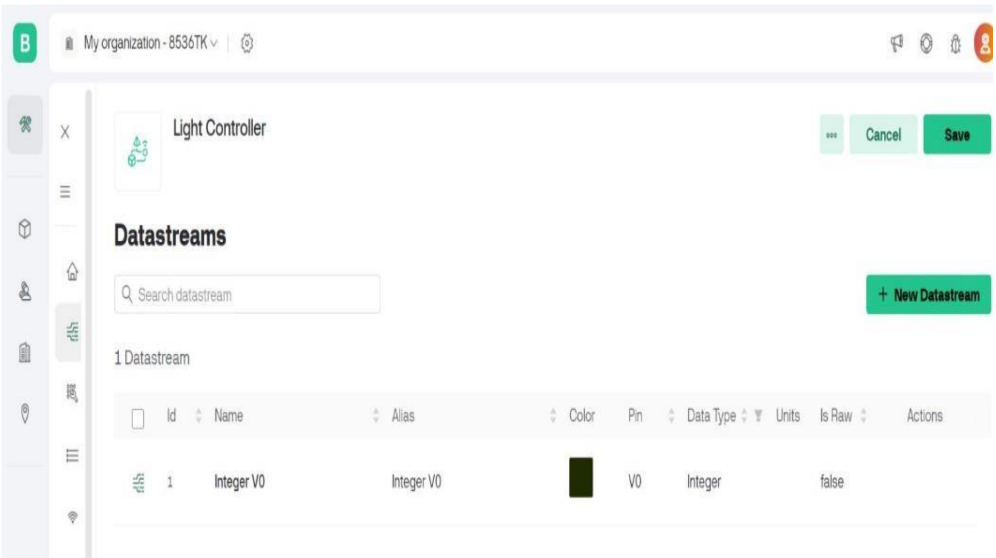


Figure.21 Testing of the data

9. Go back to your project dashboard and add the widget. click the **Power Button** at thetop of the app interface or in the console

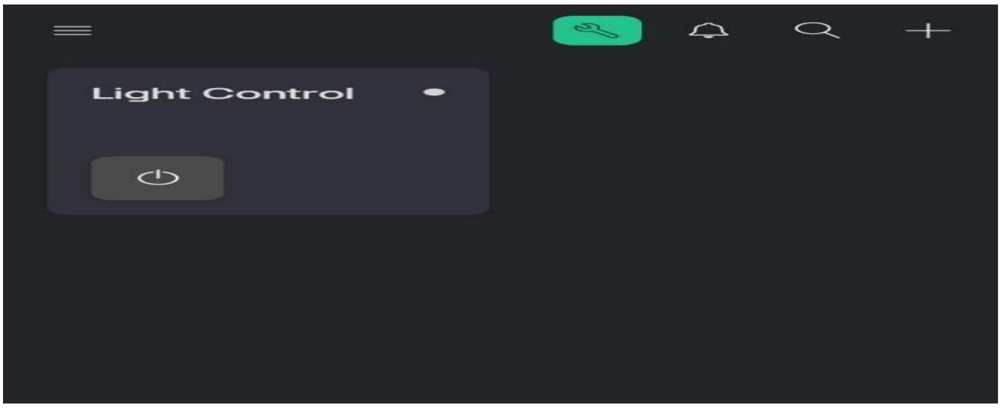


Figure.22 Power button to ON/OFF LED through Blynk

10. Connecting it to your hardware and starting data streaming to access the option.

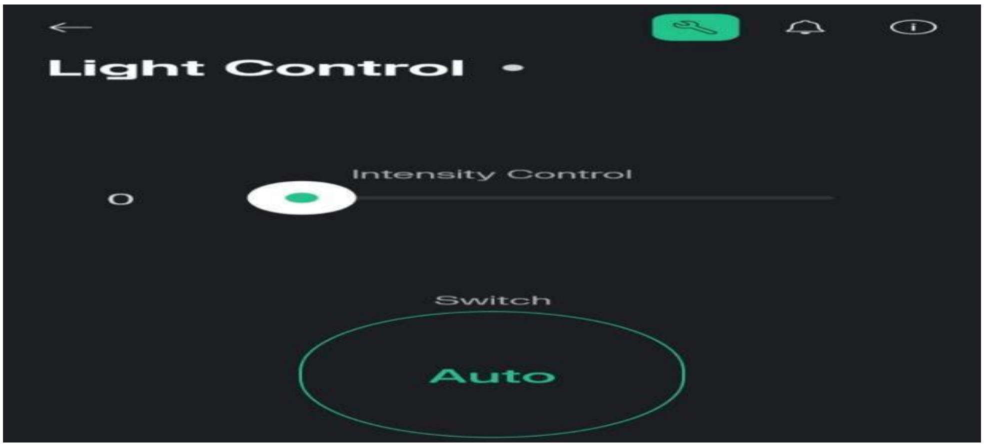


Figure.22 Automatic button for intensity control via Blynk

11. Change the option auto into manual and control the intensity of LED light.

Figure.23 Manual button for intensity control via Blynk

Step 6: Uploading the Code on blynk

1. **Install Blynk Library**: Open the Arduino IDE, go to **Tools > Manage Libraries**,search for **Blynk**, and install it.
2. **Get Your Auth Token**: In the Blynk app, go to **Device Info** and copy your **AuthToken**.

Step-by-Step Code Upload

1. **Open Arduino IDE** and create a new sketch.
2. **Write Your Code**: ESP8266 to Blynk and control an LED.

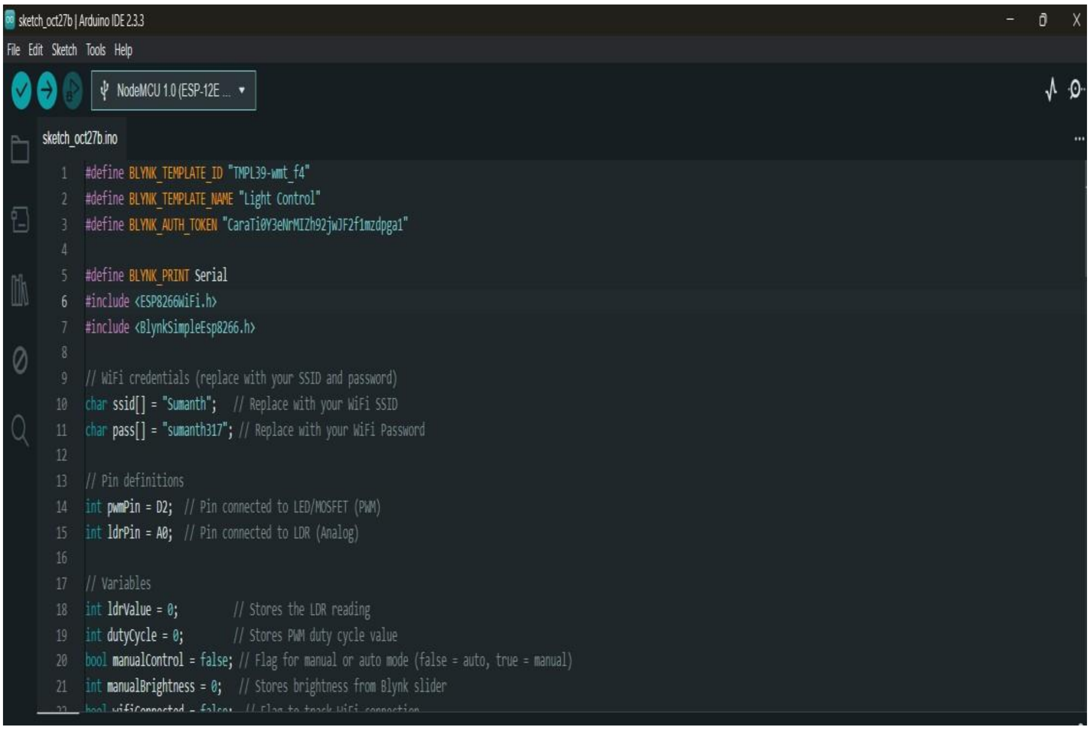


Figure.24 Code for connection of node and Blynk application

Step 8: Final Execution

After successfully executing the code, the LED turns on, and you can now use the Blynk appto control its intensity, adjusting brightness directly from your smartphone. This setup also allows the LED to respond to ambient light conditions, enabling it to adjust automatically based on the surrounding light levels. This integration provides a practical way to interact with the hardware and adapt to the environment in real-time.

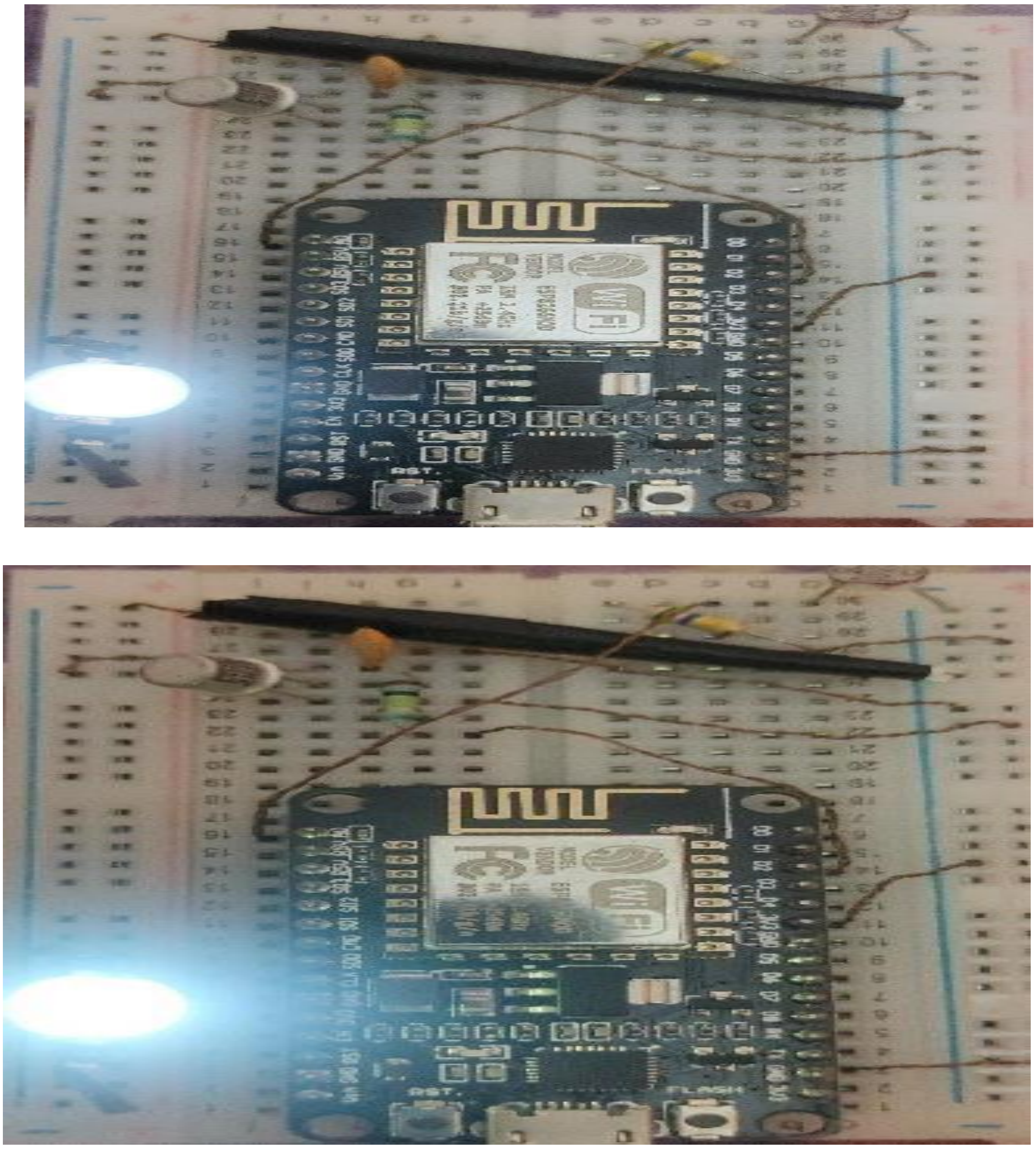


Figure.25 final output

CHAPTER 6

6. ARDIUNO SCRIPT

6.1 SCRIPT IMPLEMENTATION

```
#define BLYNK_TEMPLATE_ID "TMPL39-wmt_f4"
#define BLYNK_TEMPLATE_NAME "Light Control"
#define BLYNK_AUTH_TOKEN "CaraTi0Y3eNrMIZh92jwJF2f1mzdpga1"
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
// WIFI credentials (replace with your SSID and password)
char ssid [] = "SSID"; // Replace with your WIFI SSID
char pass [] = "PASSWORD"; // Replace with your WIFI Password
// Pin definitions
int pwmPin = D2; // Pin connected to LED/MOSFET (PWM)
int ldrPin = A0; // Pin connected to LDR (Analog)
// Variables
int ldrValue = 0;        // Stores the LDR reading
int dutyCycle = 0;       // Stores PWM duty cycle value
bool manualControl = false; // Flag for manual or auto mode (false = auto, true = manual)
int manualBrightness = 0;  // Stores brightness from Blynk slider
bool wifiConnected = false;  // Flag to track WIFI connection
// Blynk virtual pins
#define V1 1 // Slider for manual brightness control
#define V2 2 // Button for switching between auto and manual mode


// Slider widget to manually control brightness
BLYNK_WRITE(V1) {
  if (manualControl) {// Only adjust brightness in manual mode
    manualBrightness = param.asInt(); // Get value from Blynk slider
    analogWrite(pwmPin, manualBrightness);
  }
```

```
}

// Button widget to toggle between manual and auto modes
BLYNK_WRITE(V2) {
  manualControl = param.asInt(); // Get the state of the button (0 = auto, 1 = manual)
  if (manualControl) {
    Serial.println("Manual mode activated");
  } else {
    Serial.println("Automatic mode activated");
  }
}

void controlByLDR() {
  // Adjust brightness based on LDR reading in auto mode
  ldrValue = analogRead(ldrPin);
  Serial.print("LDR Value: ");
  Serial.println(ldrValue);
  // Map LDR value to PWM duty cycle
  dutyCycle = map(ldrValue, 0, 1023, 1023, 0);
  // Set the brightness based on LDR
  analogWrite(pwmPin, dutyCycle);
}
void setup() {
  // Set up serial monitor and pins
  Serial.begin(9600);
  pinMode(pwmPin, OUTPUT);
  pinMode(ldrPin, INPUT);
  // Immediately start controlling light based on LDR
  Serial.println("Starting in automatic mode (LDR control)...");
  controlByLDR();  // Control light right after startup

  // Start non-blocking WIFI connection
```

```
  WiFi.begin(ssid, pass);
  Blynk.config(BLYNK_AUTH_TOKEN); // Configure Blynk, but don't block for
connection
}
void loop() {
  // Non-blocking WIFI connection check
  if (WiFi.status() == WL_CONNECTED) {
    if (!wifiConnected) {
      // If WiFi was previously disconnected and now connected
      wifiConnected = true;
      Serial.println("WiFi connected. Switching to user control.");
      Blynk.connect(); // Attempt to connect Blynk if WiFi is connected
    }
    Blynk.run(); // Blynk only runs if WiFi is connected
  } else {
    if (wifiConnected) {
      // If WiFi was previously connected and now disconnected, switch to auto mode
      wifiConnected = false;
      Serial.println("WiFi disconnected. Switching to automatic mode.");
    }
  }
  // Default control using LDR, whether WiFi is connected or not
  if (!manualControl || !wifiConnected) {
    controlByLDR(); // Control by LDR in automatic mode or when WiFi is not available
  } else if (manualControl) {
    analogWrite(pwmPin, manualBrightness); // In manual mode, brightness is controlled by
the Blynk slider
  }
    // Add a small delay to reduce the rate of readingsdelay(100);
}
```

6.3 CODE DESCRIPTION:

This Arduino sketch is designed for the ESP8266 to control the brightness of an LED based on readings from a Light Dependent Resistor (LDR) and user input via the Blynk app. It includes necessary libraries for WiFi and Blynk functionality, with placeholders for WiFi credentials that the user must replace with their actual SSID and password. The code defines pins for PWM control (D2) and LDR readings (A) and initializes variables for storing LDRvalues, PWM duty cycles, and control flags. Two virtual pins are defined for the Blynk app: one for a slider controlling brightness (V1) and another for toggling between manual and automatic modes (V2). When in manual mode, the LED brightness is adjusted using the Blynk slider, while in automatic mode, the brightness is controlled based on LDR readings.The setup function initializes serial communication, sets pin modes, and connects to WiFi, while the loop function checks the WiFi status, runs Blynk if connected, and controls the LED based on the current mode. The LDR value is mapped to a PWM duty cycle for brightness adjustment, and the code ensures an active connection to Blynk while automatically controlling the LED when disconnected. It provides user feedback through the serial monitor and employs non-blocking calls for WiFi and Blynk to maintain responsiveness, along with a small delay to reduce the frequency of readings and control commands. Overall, this code enables dynamic lighting control based on ambient light conditions and user preferences through the Blynk app.

CHAPTER 7

7. CONCLUSION

The development and implementation of LED controllers mark a significant shift toward more sustainable lighting solutions, focusing on optimizing energy consumption and enhancing illumination quality. As the world moves towards smarter and more eco-friendly technologies, these controllers are essential in modernizing lighting systems across various sectors. By incorporating features like dimming, smart scheduling, and integration with renewable energysources, LED controllers help reduce operational costs, extend product lifespans, and improve lighting quality, ultimately creating healthier and more productive environments. This projectnot only showcases the financial advantages of adopting LED controllers but also emphasizes their broader contribution to global sustainability efforts. Through comprehensive analysis of design principles, operational benefits, and case studies, this project demonstrates the transformative potential of LED controllers in shaping a smarter, more sustainable future for lighting systems worldwide. Through a comprehensive analysis of design principles, operational benefits, and real-world applications, this project highlights both the financial and ecological advantages of adopting LED controllers. By reducing electricity bills, lowering maintenance demands, and decreasing carbon footprints, these controllers offer a path to energy-efficient, sustainable lighting solutions. Additionally, they support global sustainability efforts by aligning with environmental goals and promoting responsible resource usage.

This project underscores the transformative potential of LED controllers, not only in terms ofcost savings and enhanced lighting quality but also in their broader impact on global sustainability initiatives. The adoption of LED controllers exemplifies a commitment to a greener, more energy-conscious future. As communities and industries increasingly integrate these technologies, we move closer to realizing a sustainable world where advanced, eco- friendly lighting solutions contribute to long-term environmental and economic well-being. This journey toward smarter lighting represents a powerful step forward in creating energy- efficient, resilient infrastructure that supports a healthier and more sustainable planet.

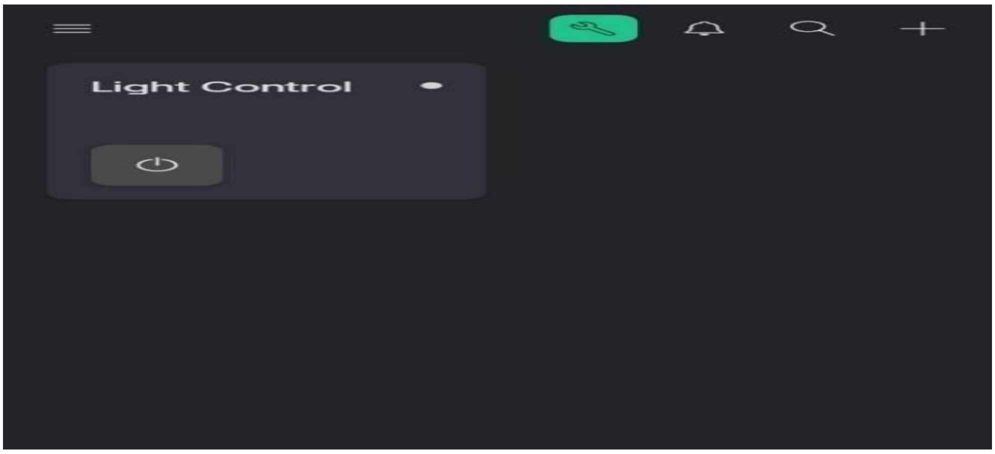
7.1 FUTURE ENHANCEMENT

The next generation of LED controllers will feature advanced IoT integration, enabling seamless connections with smart home and building systems. This connectivity will facilitate real-time monitoring and dynamic adjustments based on factors such as occupancy, ambient light, and user preferences, significantly enhancing energy efficiency and convenience. Leveraging artificial intelligence (AI) and machine learning (ML), these controllers will learn from user behavior and environmental conditions, automatically adjusting lighting settings for optimal efficiency. Predictive analytics will further enhance functionality by allowing proactive maintenance, minimizing unexpected issues. Additionally, future controllers will incorporate self-diagnostic capabilities that autonomously detect and resolve component failures or inefficiencies, improving overall system reliability and reducing maintenance needs. As smart city infrastructure expands, LED controllers will scale to manage extensive lightingnetworks, such as city-wide street lighting systems, ensuring efficient energy use across urban areas. Users will benefit from greater personalization options, allowing them to tailor lightingsettings based on circadian rhythms, tasks, or moods, which will enhance comfort, productivity, and overall well-being. With the increase in connectivity, robust cybersecurity measures, including encryption and secure communication protocols, will become essential to protect these systems from potential cyber threats. Furthermore, by utilizing AI-powered analytics, future LED controllers will anticipate system failures and performance issues before they occur, ensuring proactive maintenance and reducing downtime. This comprehensive approach will transform LED control systems into more efficient, reliable, and user-friendly solutions in our increasingly connected world.

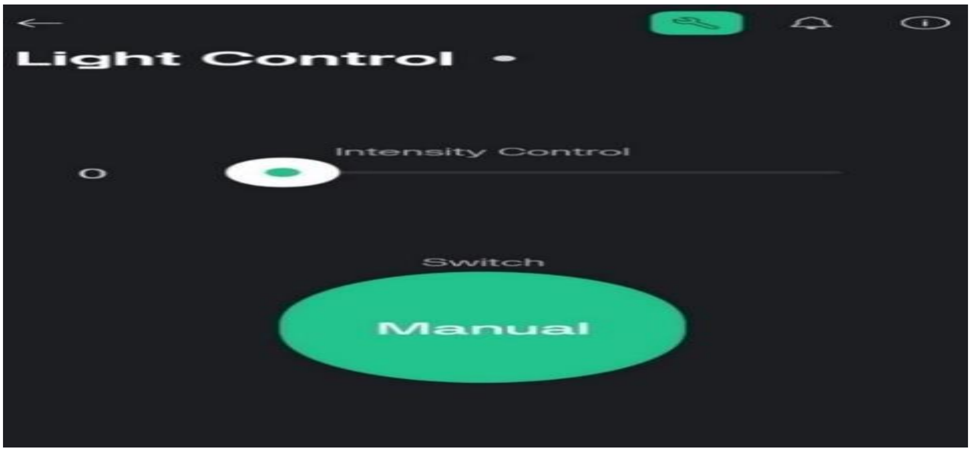
APPENDIX

SCREENSHOTS:

1. Complete all configuration process connect mobile WiFi with Esp8266 NodeMCU issuccessfully connected
2. Open blynk app show our manual controller use control.



3. This interface uses to control led light



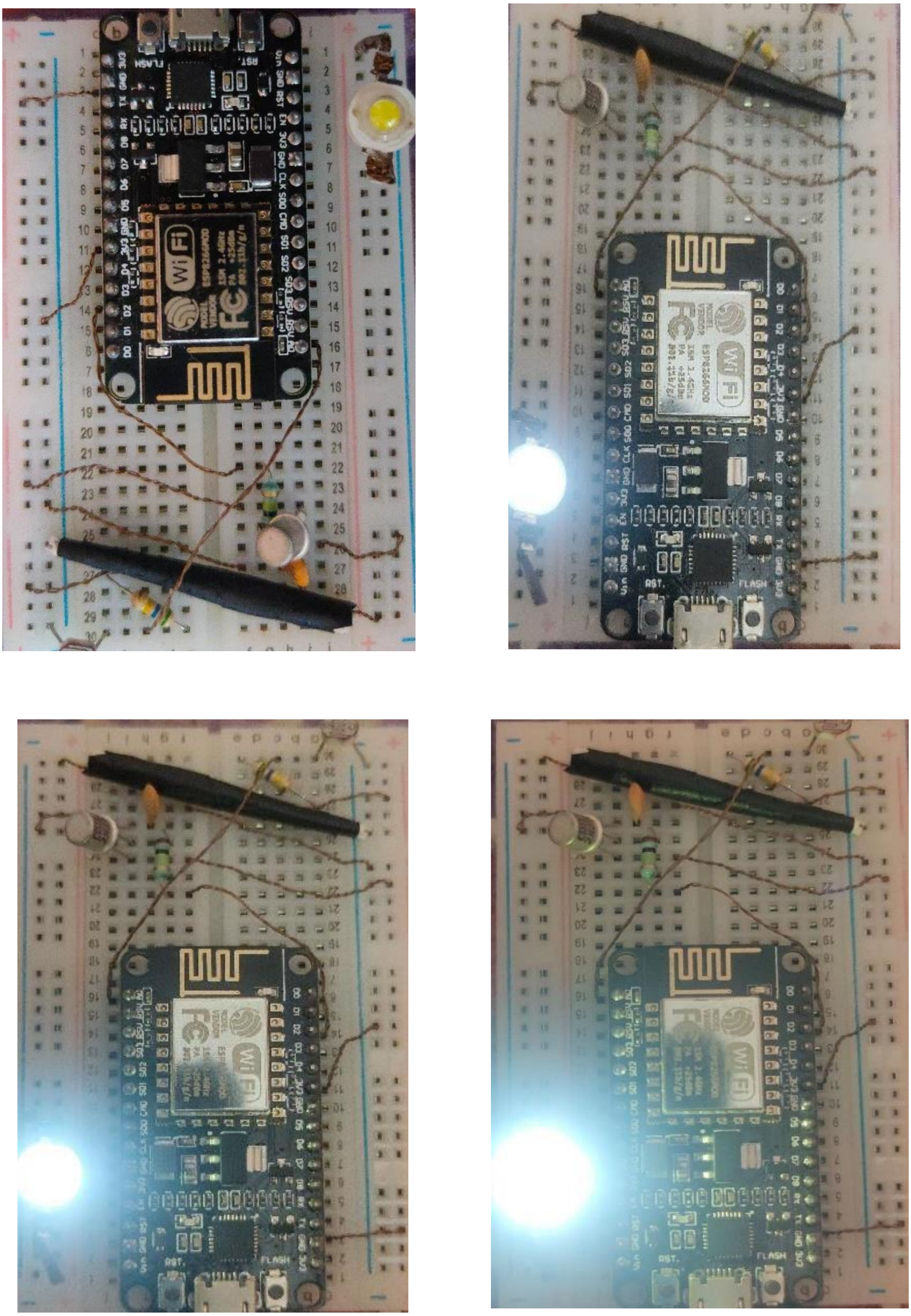


Figure.26 Stages of output according to ambient light

REFERENCES

1. Implementing IoT for Energy-Efficient Smart Street Light Management(October 2024)journal of Innovative Computing and Communications

2. An Electronic Economic Controller for automatically adjusting the intensity of Solar-
Powered LED Street Lightning Systems in rural areas
(July 2024) GUB Journal of Science and Engineering

3. Design and building of a room light control System based on iot esp8266 cloud(July 2024) Jambura Journal of Electrical and Electronics Engineering

4. Monitoring System for Traffic Light Lamp Damage using BLYNK Application Basedon IOT ESP32
(June 2024) Jurnal Teknologi Transportasi dan Logistik

5. IOT controlled led notice board
(April 2024) International Scientific Journal of Engineering and Management