SMART GREENHOUSE SYSTEM



SONARGAON UNIVERSITY (SU)

Supervised By:

Rudaina Tasnuva

Lecturer

Department of EEE

Sonargaon University (SU)

Submitted By:

Md. Emon Hossen **ID: EEE2103024041**

Md. Tanvir Wahid **ID: EEE2103024042**

Md. Sakib Hasan **ID: EEE2103024043**

Md. Atikur Rahman **ID: EEE2103024044**

Md. Ratul Hossain **ID: EEE2103024045**

Department of Electrical & Electronic Engineering (EEE)

Sonargaon University (SU)

147/I, Panthapath,

Dhaka-1215, Bangladesh

Date of Submission: May, 2025

Declaration

It is declared hereby that this thesis paper or any part of it has not been submitted to anywhere else for the award of any degree.

……..	……
Md. Emon Hossen	**Md. Tanvir Wahid**
……..	……..
Md. Sakib Hasan	**Md. Atikur Rahman**
…….	
Md. Ratul Hossain	

Under Supervision of

.....
Rudaina Tasnuva
Lecturer
Department of EEE
Sonargaon University (SU) Sonargaon University, Dhaka

Certification

This is to certify that this project entitled **"Smart Greenhouse System"** is done by the following students under my direct supervision. This project work has been carried out by them in the laboratories of the Department of Electrical and Electronic Engineering under the Faculty of Engineering, Sonargaon University (SU) in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering.

Supervisor

.....

Rudaina Tasnuva
Lecturer
Department of Electrical and Electronic Engineer (EEE)
Sonargaon University (SU)

Acknowledgement

The Report titled as on "**Smart Greenhouse System"** has been prepared to fulfill the requirement of our practicum program. In the process of doing and preparing our practicum report, we would like to pay our gratitude to some persons for their enormous help and vast co-operation.

At first, we would like to show our gratitude to the university authority to permit us to do our practicum. Specially, we would like to thank to our honorable teacher **Rudaina Tasnuva** , Lecturer, Department of Electrical & electronics engineering, SU- Sonargaon university, Dhaka for his valuable and patient advice, sympathetic assistance, co-operation, contribution of new idea. Deep theoretical and hardware knowledge & keen interest of our supervisor in this field influenced us to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

Finally, we would like to thanks again to the respected **Vice-Chancellor of SU, Professor Shamim Ara Hassan** ,also thanks to **Head of Department of SU, Electrical & Electronics Engineering, Md. Rais Uddin Mollah** because they are designated such an environment for learning through which we got the opportunity to acquire knowledge under Bsc in EEE program, and that will be very helpful for our prospective career.

We are, indeed, grateful to all those from whom we got sincere co-operation and help for the preparation of this report.

Abstract

We are living in an age of technology. Lifestyle became speedier due to technology. On the daily life, caring one's garden to refresh the mind became nearly impossible. And after a few days or weeks, watching the favorite plants dehydrated to death hurts very much. Without being the cause of relieving the daily stress, it makes one sad who's hobby is gardening. This project will become the lifesaver for the plants. It will make the gardener or the hobbyist feel lot better looking their plants alive and well hydrated and also looking the new flowers which would not be there if they were not cared properly.

Table of Contents

ABSTRACT ..... **II**

CERTIFICATION ..... **I**

TABLE OF CONTENTS ..... **IV**

LIST OF FIGURES ..... **VII**

LIST OF TABLES ..... **VIII**

ACKNOWLEDGEMENTS ..... ERROR! BOOKMARK NOT DEFINED.

CHAPTER 1 INTRODUCTION **1**

1.1 Background 1

1.2 Objectives 2

1.3 Methodology 2

CHAPTER 2 SYSTEM ARCHITECTURE **3**

2.1 Block Diagram 3

2.2 Circuit Diagram 4

2.3 Working Principle 4

CHAPTER 3 HARDWARE ANALYSIS **6**

3.1 NodeMCU 6

3.1.1 Technical Specification 7

3.1.2 Pin Configuration 8

3.1.3 Pin Description Table 9
3.2 DHT22 12
3.2.1 Technical Specification 13
3.2.2 Pin Configuration 14
3.3 Soil Moisture Sensor 14
3.3.1 Technical Specification 15
3.3.2 Pin Configuration 16
3.4 DC 5V Submersible Pump 16
3.4.1 Technical Specification 16
3.5 2 Channel Relay Module 17
3.5.1 Pin Configuration 18
3.6 Incandescent Bulb 19
CHAPTER 4 ACTIVITIES 20
4.1 Project board testing 20
4.2 Placing Components on Project Stand 21
4.3 Apps Developed 22
4.4 Realtime Database 23
CHAPTER 5 CONCLUSION 24
5.1 Result 24
5.2 Application 24

5.3 Costing of the project 25

5.4 Conclusion..... 25

APPENDIX A PROGRAMING OF NODEMCU..... 26

APPENDIX B SOFTWARE USED..... 32

REFERENCES..... 34

LIST OF FIGURES

Figure 2.1: Block diagram of Smart Greenhouse System. 3

Figure 2.2: Circuit diagram of Smart Greenhouse System. 4

Figure 3.1: NodeMCU [2]..... 7

Figure 3.2: Pin configuration [3] 8

Figure 3.3: DHT22 [5] 13

Figure 3.4: Soil moisture sensor [7]..... 15

Figure 3.5: 5V DC mini submersible water pump.[8] 16

Figure 3.6: 2-channel relay module [10]..... 18

Figure 3.7: Incandescent bulb [11] 19

Figure 4.1: Project tested on project board. 20

Figure 4.2: After placing the components in the Project Stand. 21

Figure 4.3: Android apps to show real-time information. 22

Figure 4.4: Realtime database of google firebase. 23

LIST OF TABLES

Table 3.1: Pin description of NodeMCU. 11

Table 5.1: Total cost for this project. 25

CHAPTER 1

Introduction

1.1 Background

Every people have hobbies. Some have one and some have more than one. Some enjoys meeting with new people, some enjoys collecting stamps, and some people just make new decorative designs. There are lots and lots of hobbies. Everyone refreshes their mind through spending time with their loved ones or hobbies. It is important for a sound health and mind. But, in this busy world, everyone does not have that extra time to spend for their hobbies.

I have a hobby and it is gardening. I enjoy spending time with my plants and flowers very much. It helps me a lot to release any stress I have. But sometimes being busy I cannot take care of my plants regularly. As a result, my plants die or become weak due to dehydration. And also, due to very cold or very hot weather, the growth of the plans decreases a lot.

To extend the life of my plants and to hydrate them regularly, the idea of automatic watering system came in. Using a microcontroller and a sensor, automatic watering system is possible. But it will not give me any update when I am out of my home. To get information online and live update along with automatic watering and temperature control, the project idea took place in my mind.

1.2 Objectives

- It will water the plants when the water level becomes below the low level.
- It will turn an incandescent bulb on to raise the temperature if the temperature of the garden becomes below desired temperature.
- It will update current temperature, humidity and water level to an online database.
- The database will update an Android apps in real-time.

1.3 Methodology

- Creating an idea do design a circuit for this project.
- Collecting all required components as necessary.
- Placing everything in a project board and test for results.
- Designing an Android apps.

CHAPTER 2

System Architecture

2.1 Block Diagram

The first step to design a project is to lay down a block diagram. The block diagram for this project

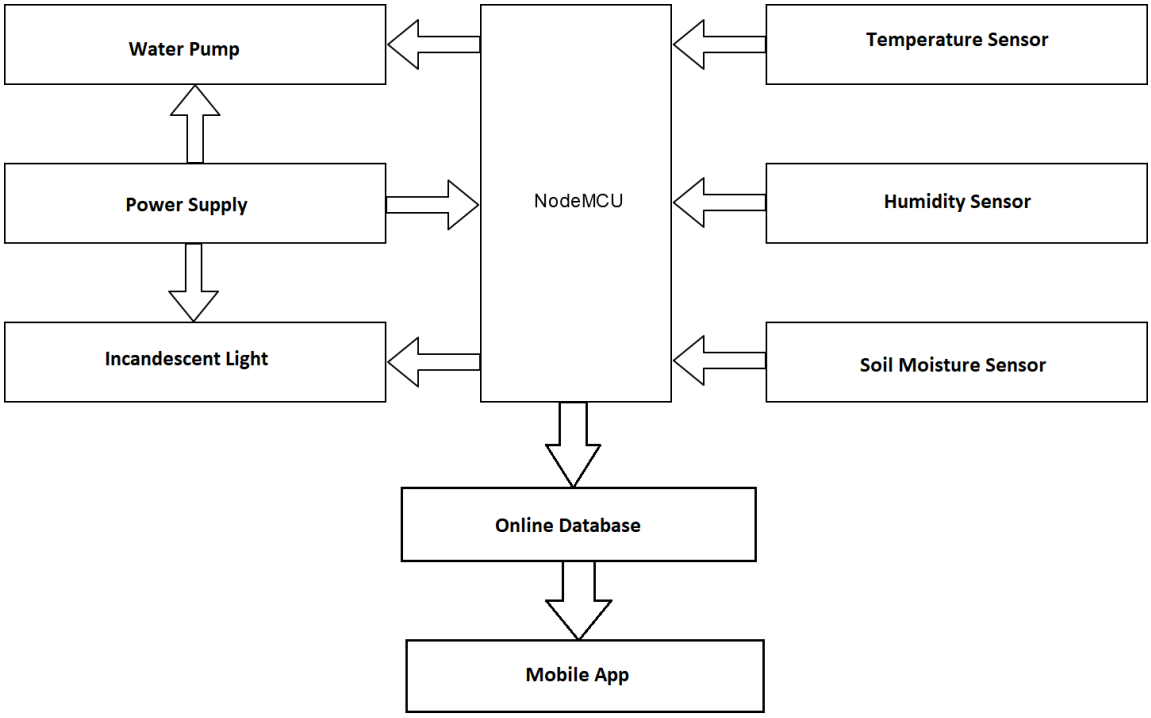


Figure 2.1: Block diagram of Smart Greenhouse System.

2.2 Circuit Diagram

The circuit diagram of the project

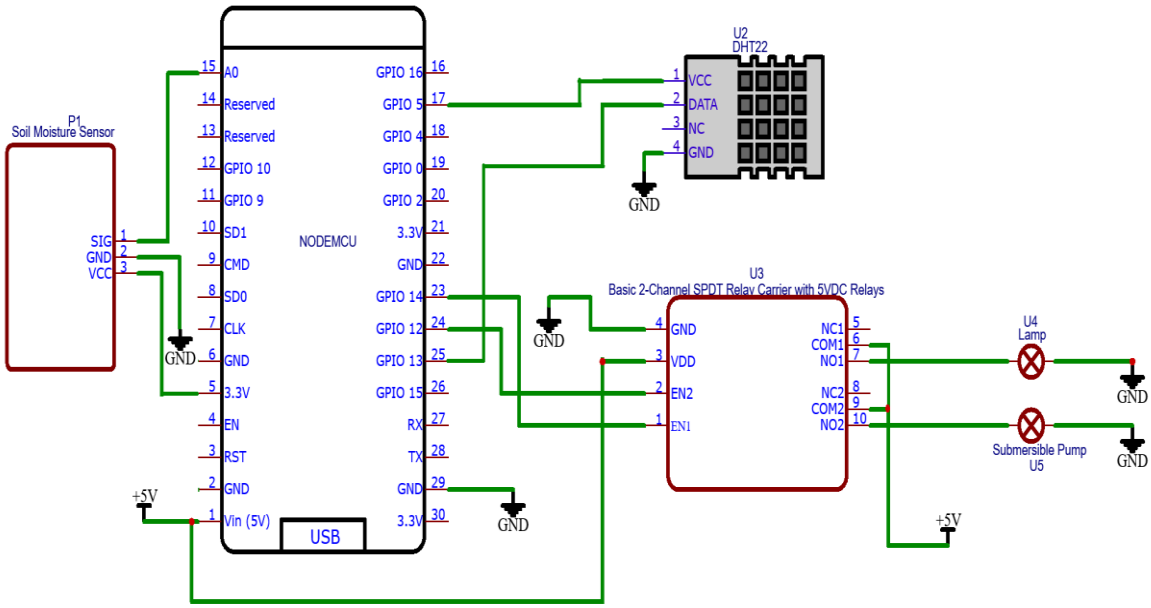


Figure 2.2: Circuit diagram of Smart Greenhouse System.

2.3 Working Principle

The temperature and humidity sensor continuously sense the temperature and humidity of the area. NodeMCU receive this data and process it as the program. The data get updated every 10-30 seconds and sent to the online database every 5 minutes.

The soil moisture sensor also senses the moisture in the soil and send the data to NodeMCU every 10-30 seconds. This data is updated to the online database.

If the temperature gets below a given value, the switching relay of the incandescent bulb (in this project an LED is used) is turned on. If the temperature gets up to a certain level after switching on the bulb, the switching relay is turned off.

If the soil moisture gets below a given value, the submersible pump is turned on for few seconds and then turned off by the switching relay.

CHAPTER 3

Hardware Analysis

3.1 NodeMCU

The NodeMCU (Node Microcontroller Unit) is an open-source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains all crucial elements of the modern computer: CPU, RAM, networking (WIFI), and even a modern operating system and SDK.[1]

NodeMCU is an open-source firmware for which open-source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits. Both the firmware and prototyping board designs are open source. The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266.

The design was initially based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications (see related projects).

Through its pins we can read inputs - light on a sensor, a finger on a button, or a Twitter message -and turn them into an output - activating a motor, turning on an LED,

publishing something online. It has also Wi-Fi capabilities, so we can control it wirelessly and make it work on a remote installation easily!



Figure 3.1: NodeMCU [2]

3.1.1 Technical Specification

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB

- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna

3.1.2 Pin Configuration

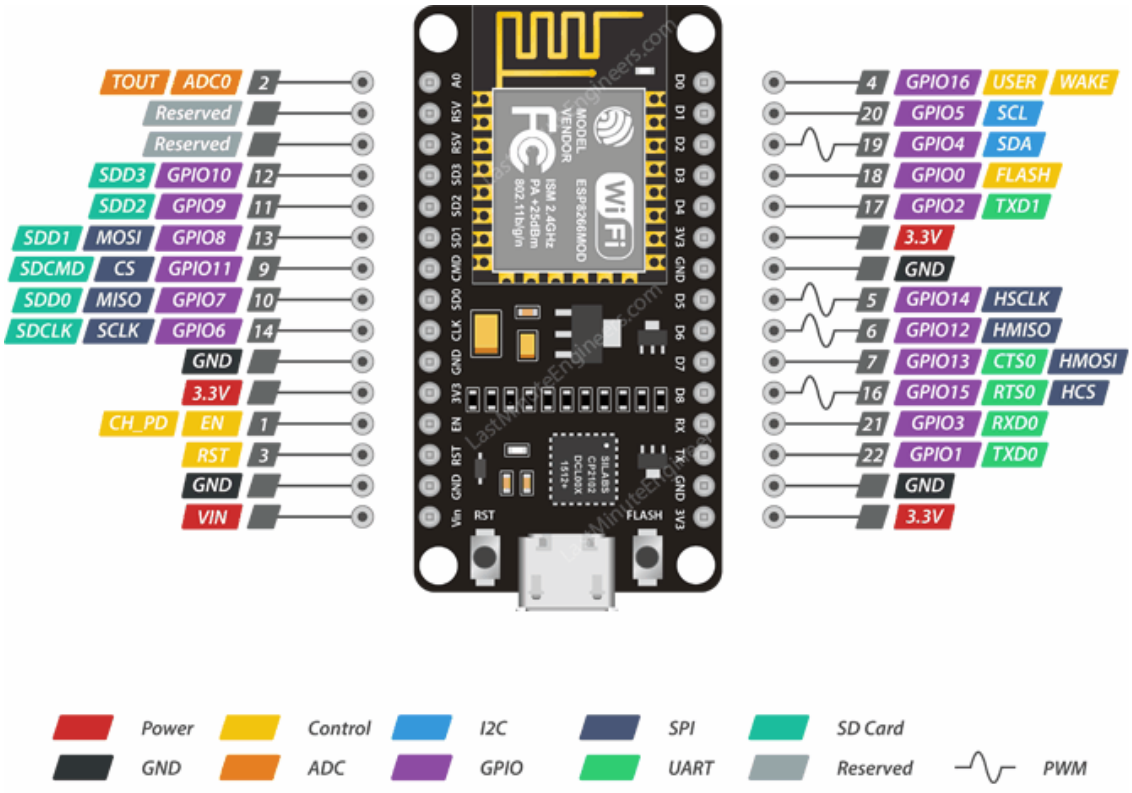


Figure 3.2: Pin configuration [3]

3.1.3 Pin Description Table

Power Pins: There are four power pins viz. one VIN pin & three 3.3V pins. The VIN pin can be used to directly supply the ESP8266 and its peripherals, if you have a regulated 5V voltage source. The 3.3V pins are the output of an on-board voltage regulator. These pins can be used to supply power to external components.

GND: is a ground pin of ESP8266 NodeMCU development board.

I2C Pins: are used to hook up all sorts of I2C sensors and peripherals in your project. Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.

GPIO Pins: ESP8266 NodeMCU has 17 GPIO pins which can be assigned to various functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.

ADC Channel: The NodeMCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC viz. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.

UART Pins: ESP8266 NodeMCU has 2 UART interfaces, i.e. UART0 and UART1, which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication. It supports fluid control. However, UART1 (TXD1 pin) features only data transmit signal so, it is usually used for printing log.

SPI Pins: ESP8266 features two SPIs (SPI and HSPI) in slave and master modes. These SPIs also support the following general-purpose SPI features:

- 4 timing modes of the SPI format transfer
- Up to 80 MHz and the divided clocks of 80 MHz
- Up to 64-Byte FIFO

SDIO Pins:ESP8266 features Secure Digital Input/Output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.

PWM Pins: The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 μs to 10000 μs, i.e., between 100 Hz and 1 kHz.

Control Pins: are used to control ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.

- EN pin – The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
- RST pin – RST pin is used to reset the ESP8266 chip.
- WAKE pin – Wake pin is used to wake the chip from deep-sleep.

Table 3.1: Pin description of NodeMCU.

Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	3.3V: Regulated 3.3V can be supplied to this pin to power the board GND: Ground pins Vin: External Power Supply
Control Pins	EN, RST	The pin and the button reset the microcontroller
Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	Four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	Two UART interfaces.
I2C Pins	GPIO5, GPIO4	I2C functionality at GPIO5 and GPIO4 pins

3.2 DHT22

The DHT22 is a basic, low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It's fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds, so when using our library, sensor readings can be up to 2 seconds old.[4]

DHT22 output calibrated digital signal. It utilizes exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability. Its sensing elements is connected with 8-bit single-chip computer. Every sensor of this model is temperature compensated and calibrated in accurate calibration chamber and the calibration-coefficient is saved in type of program in OTP memory, when the sensor is detecting, it will cite coefficient from memory. Small size & low consumption & long transmission distance(20m) enables DHT22 to be suited in all kinds of harsh application occasions. Single-row packaged with four pins, making the connection very convenient.

This sensor was chosen for this project to get an accurate temperature and humidity value. In different types of application for this project, it would be an ideal choice to use DHT22 module.

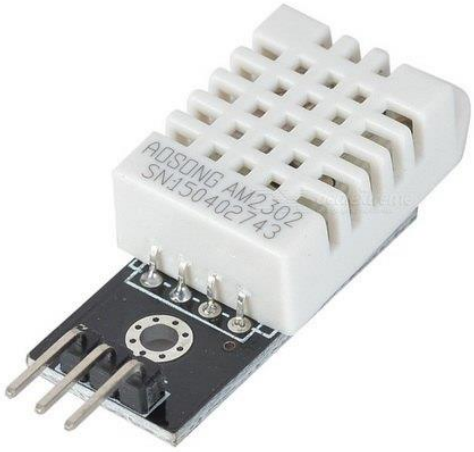


Figure 3.3: DHT22 [5]

3.2.1 Technical Specification

- High precision
- Capacitive type
- Full range temperature compensated
- Relative humidity and temperature measurement
- Low power consumption
- Measuring range: temperature -40℃-120 ℃
- Measuring range: humidity 0; 99.9%RH
- Measurement accuracy: temperature: + 0.5; humidity: + 2%RH (10; 90%RH)
- Response time: temperature: 5S; humidity: 5S 1/e (63%)
- Power source: DC3.3-5.5V

3.2.2 Pin Configuration

1. Vcc pin: Power supply 3.3V to 5.5V
2. Data pin: Outputs both Temperature and Humidity through serial Data
3. Ground pin: Connected to the ground of the circuit

3.3 Soil Moisture Sensor

The Soil Moisture Sensor is a simple breakout for measuring the moisture in soil and similar materials. The soil moisture sensor is pretty straight forward to use. The two large exposed pads function as probes for the sensor, together acting as a variable resistor. The more water that is in the soil means the better the conductivity between the pads will be and will result in a lower resistance, and a higher SIG out. [6]

Most soil moisture sensors are designed to estimate soil volumetric water content based on the dielectric constant (soil bulk permittivity) of the soil. The dielectric constant can be thought of as the soil's ability to transmit electricity. The dielectric constant of soil increases as the water content of the soil increases. This response is due to the fact that the dielectric constant of water is much larger than the other soil components, including air. Thus, measurement of the dielectric constant gives a predictable estimation of water content.

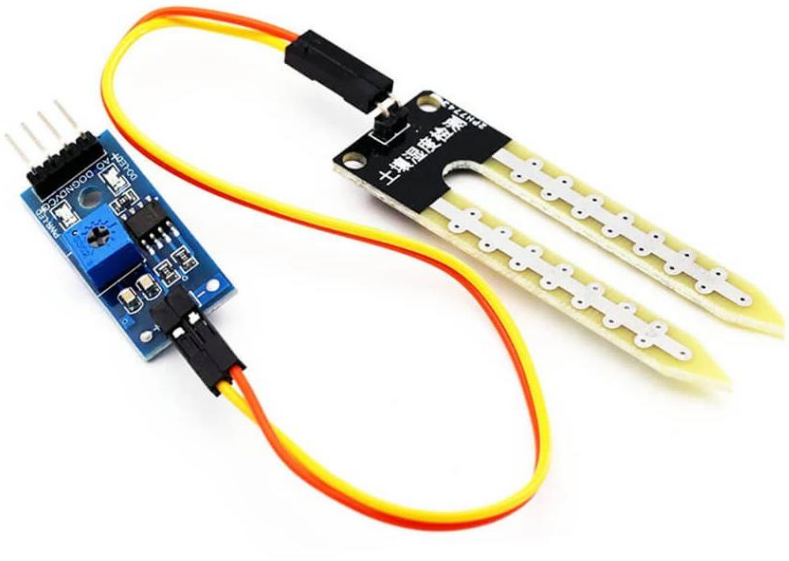


Figure 3.4: Soil moisture sensor [7]

3.3.1 Technical Specification

- Operating Voltage: 3.3V to 5V DC
- Operating Current: 15mA
- Output Digital - 0V to 5V, Adjustable trigger level from preset
- Output Analog - 0V to 5V based on the soil moisture reading and wet-soil falling on the sensor
- LEDs indicating output and power
- PCB Size: 3.2cm x 1.4cm
- LM393 based design
- Easy to use with Microcontrollers or even with normal Digital/Analog IC
- Small, cheap and easily available

3.3.2 Pin Configuration

1. VCC pin: The Vcc pin powers the module, typically with +5V
2. GND pin: Power Supply Ground
3. D0 pin: Digital out pin
4. A0 pin: Analog out pin

3.4 DC 5V Submersible Pump

A small sized DC submersible water pump. Operated at 5V. Used in this project for watering the plant.



Figure 3.5: 5V DC mini submersible water pump.[8]

3.4.1 Technical Specification

- Input Voltage: DC 3V-5V
- Flow Rate: 1.2-1.6 L/min

- Operation Temperature: 80 Deg.C
- Operating Current: 0.1-0.2A
- Suction Distance: 0.8 meter (Max)
- Outside diameter of water outlet: 7.5mm
- Inside diameter of water outlet: 5.0 mm
- Diameter of water Inlet : 5.0 mm
- Wire Length: 200 mm
- Size: 45 x 30 x 25 mm
- Weight: 30g

3.5 2 Channel Relay Module

2-Channel 5V Relay Module is a relay interface board, it can be controlled directly by a wide range of microcontrollers such as Arduino, AVR, PIC, ARM and so on. It uses a low-level triggered control signal (3.3-5VDC) to control the relay. Triggering the relay operates the normally open or normally closed contacts. It is frequently used in an automatic control circuit.

To put it simply, it is an automatic switch to control a high-current circuit with a low-current signal.5V relay signal input voltage range, 0-5V. VCC power to the system. JD-VCC relay in the power supply. JD-VCC and VCC can be a shorted.[9]



Figure 3.6: 2-channel relay module [10]

3.5.1 Pin Configuration

1. Vcc pin: Power supply pin for module.
2. GND pin: Connected to ground.
3. IN1 pin: Switching signal for relay 1.
4. IN2 pin: Switching signal for relay 2.

3.6 Incandescent Bulb

Incandescent bulbs are tungsten filament bulb which emits light by heating the filament. It creates heat and glow at the same time. In many cases, incandescent bulb is used to heat a surface or area.

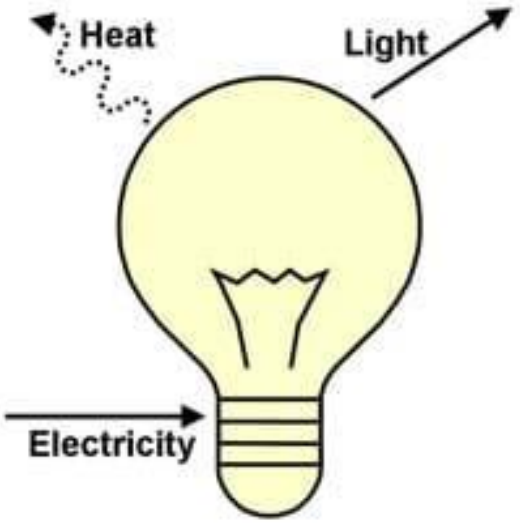


Figure 3.7: Incandescent bulb [11]

CHAPTER 4

Activities

4.1 Project board testing

The project was firstly tested on a project board to debug if any flaws are in the program.

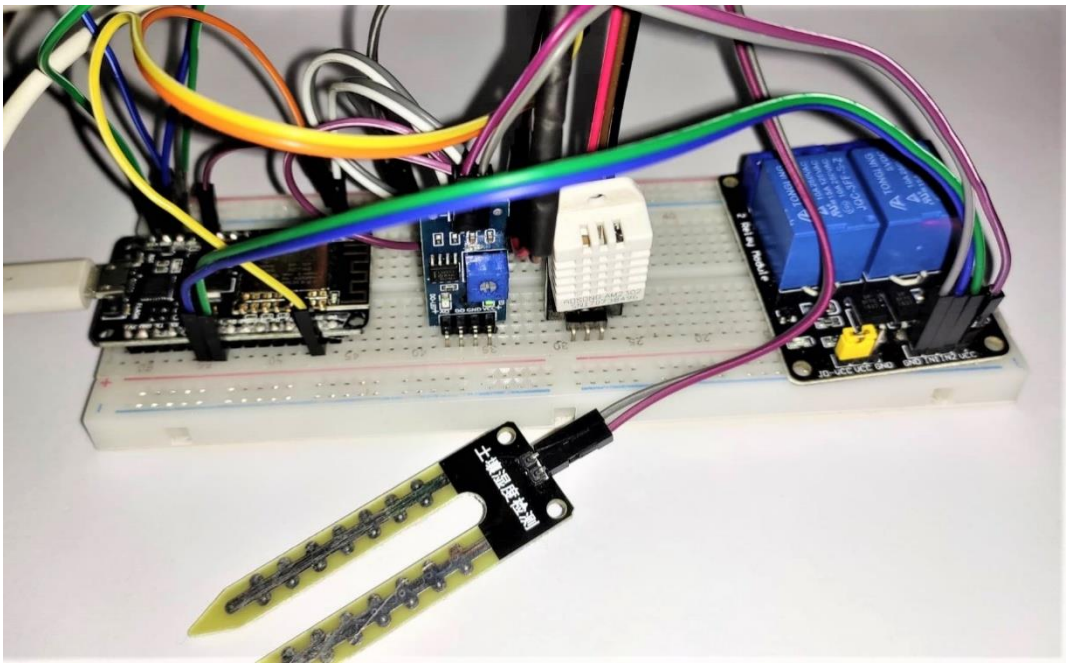
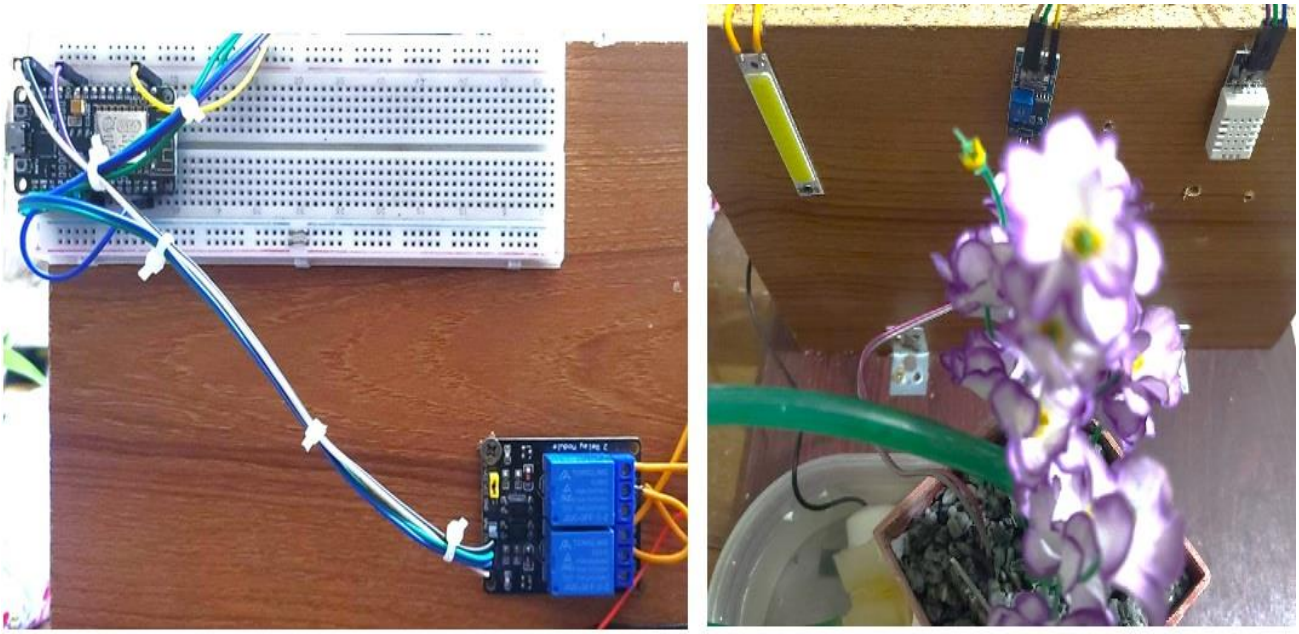


Figure 4.1: Project tested on project board.

The analog sensor device works at a voltage level of 5V. The sensor was limited to a maximum output of 3.3V using resistor by voltage divider method.

4.2 Placing Components on Project Stand

After a successful test, every component was placed on a project stand. Verifying all connection for the final time, the power was given to the circuit.



MicroController and 2 Channel Relay Module

LED, Soil Moisture Sensor, Temperature and Humidity Sensor

Figure 4.2: After placing the components in the Project Stand.

4.3 Apps Developed

An Android apps was developed to always know about the condition of my garden in real-time.

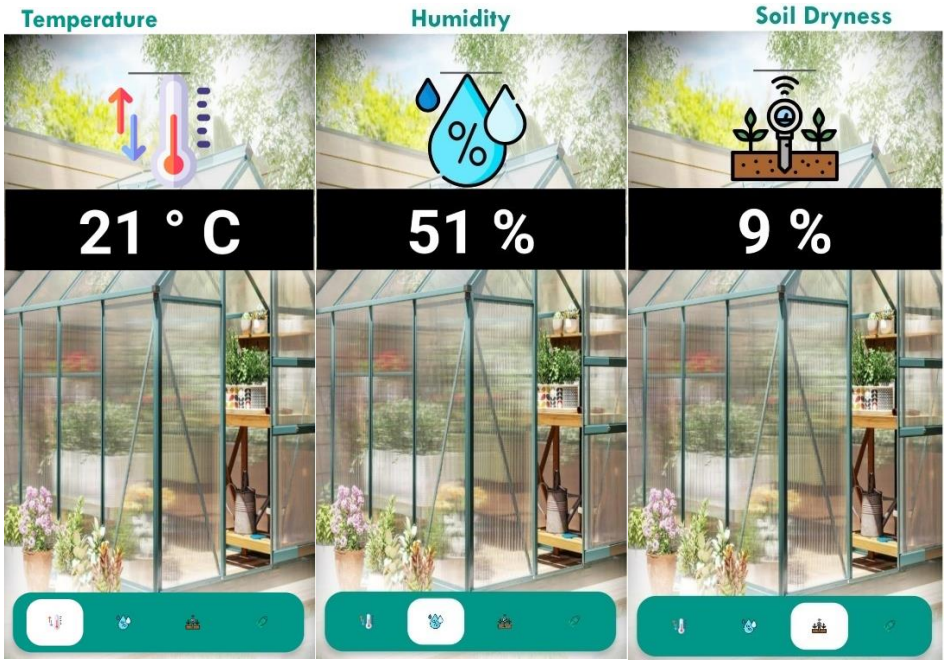


Figure 4.3: Android apps to show real-time information.

4.4 Realtime Database

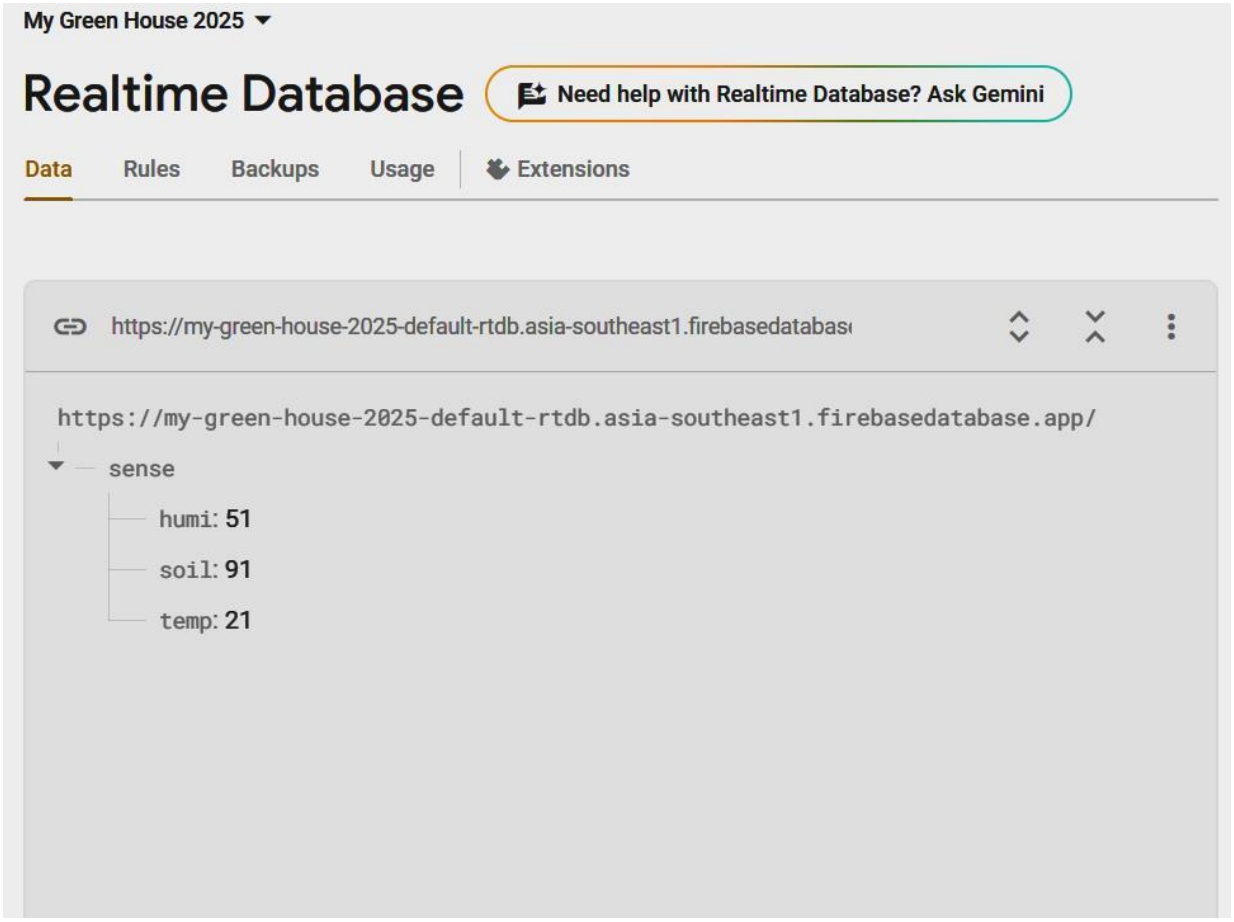


Figure 4.4: Realtime database of google firebase.

CHAPTER 5

Conclusion

5.1 Result

ii) The device was connected to a 5V adapter (Smartphone charger).

iii) On the database, the temperature, humidity and soil moisture data were being updated continuously in every 10-30 seconds.

iv) Simulated a dehydration of soil by moving away the soil moisture sensor from the wet soil, and the pump started pumping water for exactly 10 seconds and waited for few more seconds to make sure the water reached the sensor.

v) Again, the data were started to be updated every 300 seconds or 5 minutes.

5.2 Application

There could be many scopes to apply this project mainly who are gardener and having trouble keeping their plants hydrated every now and then while being too busy. Agricultural fields are much needed this project. Micro-climate control is possible by using this project. Growth rate could be maintained with proper research. Controlling of infection and diseases outbreak could be controlled by implementing smart greenhouse to a bigger level.

5.3 Costing of the project

Table 5.1: Total cost for this project.

SL No.	Item Name	Quantity	Price (Taka)
1	NodeMCU	1	350
2	DHT22 sensor	1	320
3	Soil moisture sensor	1	120
4	2-channel relay module	1	100
5	5V DC submersible pump	1	100
6	Power Adapter	N/A	400
7	Wires, Project board and Misc.	N/A	300
		Total	1690

5.4 Conclusion

There are lots of plants that are very expensive and need a lot of care in my garden. Also, my garden is placed inside my house and the balcony. So, proper care was very much needed to keep them alive because they look so beautiful when cared so. Such a project can take care of plants when needed.

APPENDIX A

Programing of NodeMCU

The code for Smart Greenhouse

```
#include <ESP8266WiFi.h>
#include <Firebase_ESP_Client.h>
#include <DHT.h>

int mati = A0;
#include <addons/TokenHelper.h>
#include <addons/RTDBHelper.h>
#define DHTPIN 14
#define DHTTYPE DHT22
#define WIFI_SSID "Infxted"
#define WIFI_PASSWORD "paxxxxxxxxa"
#define API_KEY "AIzaSyCVhcBxxxxxxxxcF0anty5niWo"
#define DATABASE_URL "https://my-green-house-2025-default-rtdb.asia-southeast1.firebasedatabase.app/"
#define USER_EMAIL "ratul.project@gmail.com"
#define USER_PASSWORD "RAxxxx23"
```

```
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;

DHT dht(DHTPIN, DHTTYPE);

unsigned long sendDataPrevMillis = 0;

void setup()
{

  Serial.begin(9600);
  dht.begin();
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  pinMode(5, OUTPUT);

  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

  Serial.print("Connecting to Wi-Fi");
  while (WiFi.status() != WL_CONNECTED)
```

```
  {
    Serial.print(".");
    delay(300);
  }
  Serial.println();
  Serial.print("Connected with IP: ");
  Serial.println(WiFi.localIP());
  Serial.println();
  config.api_key = API_KEY;
  auth.user.email = USER_EMAIL;
  auth.user.password = USER_PASSWORD;
  config.database_url = DATABASE_URL;
  config.token_status_callback = tokenStatusCallback;
  Firebase.reconnectNetwork(true);
  fbdo.setBSSLBufferSize(4096 /* Rx buffer size in bytes from 512 - 16384 */, 1024 /* Tx buffer size in bytes from 512 - 16384 */);
  fbdo.setResponseSize(2048);
  Firebase.begin(&config, &auth);
  Firebase.setDoubleDigits(5);
  config.timeout.serverResponse = 10 * 1000;
  digitalWrite(5, HIGH);
 }
```

```
void loop()
{
  delay(5000);
  float h = dht.readHumidity();
  // Read Humidity as Percentage
  float t = dht.readTemperature();
  // Read temperature as Celsius (the default)

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }
  delay(2000);

  int m = analogRead(mati);

  Serial.print(F("Humidity: "));
  Serial.println(h);
  Serial.print(F("Temperature: "));
  Serial.println(t);
```

```
Serial.print(F("Soil Moist.: "));
Serial.println(m);
delay(2000);

if (m > 850){
  // If soil dryness is above 85%
  digitalWrite(12, HIGH);
  Serial.println("Pump On !");
  delay(5000);
  digitalWrite(12, LOW);
  Serial.println("Pump Off");
}
  if (t < 30){
  // If temperature is below 30 degree celcius
  digitalWrite(13, HIGH);
  Serial.println("Heater On !");}
  else{
    digitalWrite(13, LOW);
    Serial.println("Heater Off");
  }
```

```
        if (Firebase.ready() && (millis() - sendDataPrevMillis > 60000 ||
sendDataPrevMillis == 0))
        {
          sendDataPrevMillis = millis();

          Serial.printf("Info: %s\n", Firebase.RTDB.setFloat(&fbdo, F("/sense/temp"), t)
? "ok" : fbdo.errorReason().c_str());
          Serial.printf("Info: %s\n", Firebase.RTDB.setFloat(&fbdo, F("/sense/humi"), h)
? "ok" : fbdo.errorReason().c_str());
          Serial.printf("Info: %s\n", Firebase.RTDB.setInt(&fbdo, F("/sense/soil"), m) ?
"ok" : fbdo.errorReason().c_str());
          delay(2000);
        }
        delay(3000);
      }
```

APPENDIX B

Software Used

i) Arduino IDE[12]



Arduino IDE

Software

ii) Google Firebase[13]



iii) EasyEDA circuit designer[14]



iv) Diagram editor [15]



v) Kodular[16]



REFERENCES

[1] Autodesk Instructables, (n.d.). Quick Intro: a Beginner's Guide to ESP8266 NodeMCU Development Kit, (Online). Available: https://www.instructables.com/Quick-Intro-a-Beginners-Guide-to-ESP8266-NodeMCU-D/

[2] ElctronicWings, (n.d.). Introduction to NodeMCU, (Online). Available: https://www.electronicwings.com/nodemcu/introduction-to-nodemcu Accessed: 2025, Mar. 27

[3] LastMinuteEngineers, (n.d.). Installing ESP8266 Board in the Arduino IDE, (Online). Available: https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/ Accessed: 2025, Mar. 27

[4] Adafruit, (n.d.). DHT22 temperature-humidity sensor + extras, [Online]. Available: https://www.adafruit.com/product/385#description Accessed: 2025, Mar. 27

[5] Udvaboni, (n.d.). DHT22 Digital Temperature Humidity Sensor Module (Online). Available: https://udvabony.com/product/dht22-digital-temperature-humidity-sensor/ Accessed: 2025, Mar. 27

[6] SparkFun, (n.d.). SparkFun Soil Moisture Sensor. (Online). Available: https://www.sparkfun.com/products/13322 Accessed: 2025, Mar. 27

[7] Kuongshun, (n.d.). Soil Moisture Detection Humidity Sensor. (Online). Available: https://kuongshun.com/products/soil-moisture-detection-humidity-sensor Accessed: 2025, Mar. 27

[8] Indiamart, (n.d.). DC 5V Mini Submersible Water Pump. (Online). Available: https://www.indiamart.com/proddetail/dc-5v-mini-submersible-water-pump-22317318591.html Accessed: 2025, Mar. 27

[9] Components101, (Jan, 05, 2021,). 5V Dual-Channel Relay Module. (Online). Available: https://components101.com/switches/5v-dual-channel-relay-module-pinout-features-applications-working-datasheetAccessed: 2025, Mar. 27

[10] Potentiallabs, (n.d.) 2-Channel Relay Module (Online). Available: https://potentiallabs.com/cart/buy-2-channel-relay-module-arduino-raspberry-pi-online-hyderabad-india Accessed: 2025, Mar. 27

[11] Teach Engineering. (n.d.) Hands-on Activity: Light vs. Heat Bulbs. (Online). Available: https://www.teachengineering.org/activities/view/cla_activity3_household_light_bulbs Accessed: 2025, Mar. 27

[12] Arduino, (n.d.). Arduino | Software. (Online). Availabe: https://www.arduino.cc/en/software/ Accessed: 2025, Mar. 27

[13] Google, (n.d.). Firebase | Google. (Online). Available: https://firebase.google.com/ Accessed: 2025, Mar. 27

[14] EasyEDA, (n.d.). An Easier and Powerful Online PCB Design Tool, (Online). Available: https://easyeda.com/ Accessed: 2025, Mar. 27

[15] DiagramEditor, (n.d.). Free Online Diagram Editor, (Online). Available: https://www.diagrameditor.com/ Accessed: 2025, Mar. 27

[16] Kodular, (n.d.). Home, (Online). Available: https://www.kodular.io/ Accessed: 2025, Mar. 27