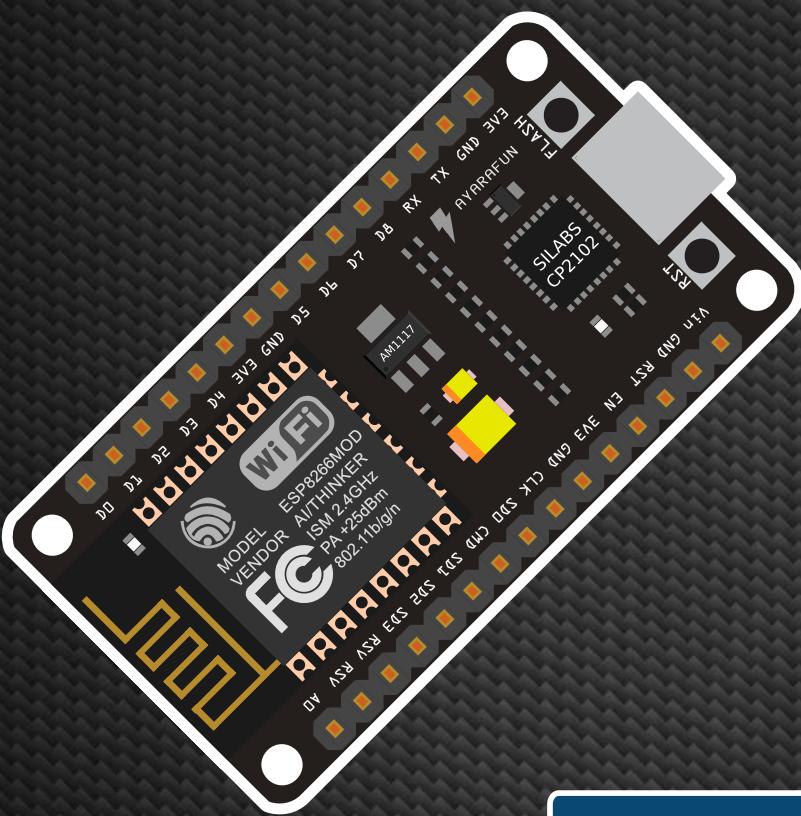




## Getting Started with NodeMCU IOT Project Kit



robotistan





Welcome to the world of Electronics and Coding. Now that you have opened this book, you are eager to swim in the sea of wonder and learn new things.

Although learning new things in such matters is thought to be difficult, if you proceed step by step and with the right practices, you will realize that it is very simple. As long as the applications are made in the early stages, there will be places that do not sit down. You will overcome this problem as you practice.

It just takes a little patience, so you can learn Arduino programming with an easy and correct roadmap, starting from the easy and moving towards more complex. If you want to watch more detailed video explanations of the applications, you can go to our YouTube channel by scanning the QR code at the back of the book. You can access the codes written in the booklet both from the description section of the related videos and from our blog page.

This book was written by Robotistan Electronics™ the purpose of writing is to guide those who want to start Arduino in an easy and correct way. Our hope is that these contents will be beneficial to everyone and facilitate your learning process and make projects quickly.

You can send us your set contents, applications, videos and any suggestions and questions you have in mind at [info@robotistan.com](mailto:info@robotistan.com)

**Robotistan Team**



## **Contents**

|   |    |
|---|----|
| Required Downloads .....                              | 5  |
| Introducing The Set .....                             | 6  |
| Web Based Led Controlling .....                       | 9  |
| Flower, Plant Watering .....                          | 16 |
| Temperature and Humidity Measurement with DHT11 ..... | 23 |
| Remote Relay Controlling .....                        | 29 |
| Blynk Rgb Led Controlling .....                       | 36 |
| Smoke and Gas Detection.....                          | 41 |
| Servo Motor Controlling .....                         | 47 |

## Requirements Download



Android

To download the Blynk application, you can download the appropriate version for your phone by scanning the QR codes on the sides.



iOS

You can download official version of Arduino IDE from [arduino.cc](https://arduino.cc).

To access all the codes in the book, you need to type the links in the url part of your computer. Or, you can reach them thanks to the QR code on the sides.



Web based led controlling : <https://bit.ly/2W9SjuM> ---->



<---- Flower, Plant Watering: <https://bit.ly/3gJlUVo>



Temp. and Humidity Measurement with DHT11: <https://bit.ly/3gbcy4> ---->



<---- Remote Relay Controlling: <https://bit.ly/3nc2f2N>



Blynk Rgb Led Kontrolü: <https://bit.ly/3me4SQ4> ---->

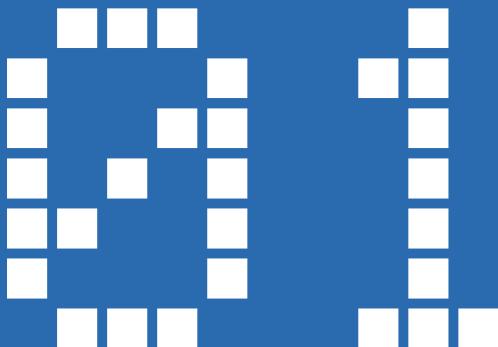


<---- Smoke and Gas Detection: <https://bit.ly/2Kd6Zq9>



Servo Motor Controlling: <https://bit.ly/37c2wgr> ---->





# Introducing The Kit

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)



### NodeMCU LoLin ESP8266 IoT Development Board

It is a development board with NodeMCU firmware loaded ESP8266 WiFi module. Since it was developed using the ESP8266 SDK, it supports GPIO, PWM, IIC, 1-Wire and ADC connections without the need for an extra microcontroller.



### Breadboard

Breadboard is a tool that you can use for testing your projects before soldering. It includes a lot of inputs, so all you have to do is providing connections without wiring.



### 2 Channel Relay Control Board

It is a relay board that can control contacts with 5V and can be used with Arduino or other microcontrollers. It draws a current of 20mA during the trigger signal from the microcontroller. It is frequently used in various hobby, industrial and robotic projects. It can switch current up to 10A at 30VDC or 220VAC voltage. There are control LEDs for each role.



### DHT11 Temperature and Humidity Sensor

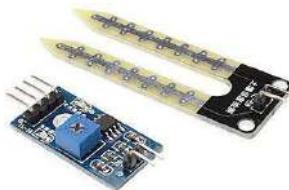
DHT11 temperature and humidity sensor is an advanced sensor unit that outputs a calibrated digital signal. It is highly reliable and stable over long term runs. It measures temperature between 0 and 50 ° C with a margin of 2 ° C error.

## Introducing The Set



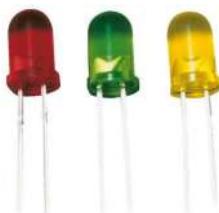
### Mini Submersible Pump

You can transfer liquids such as oil and water up to 120 liters per hour with the water motor that runs very quietly and has low power consumption. This product, which has the IP68 standard, is water and dust proof.



### Soil Moisture Sensor

It is a sensor that you can use to measure the moisture content of the soil or the level of a small-scale liquid. Moisture meter probes are used by immersing them in the medium to be measured. Due to the resistance caused by the soil or the immersed liquid, a voltage difference occurs between the probe tips.



### Diode LED

The light emitting diode (LED), as its name suggests, emits a visible light when energized. They are generally made in three different colors: red, yellow and green. Operating currents are between 5 mA and 50 mA.



### Motor Driver Board

There is L9110 motor driver IC on the board. Two separate DC motors or 4-wire 2-phase stepper motors can be controlled in both directions with the driver card operating at an input voltage between 2.5-12V. Driver output voltage is 7.6V. In this way, it is very easy to use with 6V motors. It can provide continuous current up to 800mA per channel and up to 2A instantaneously.

## Introducing The Set



### RGB LED Diode

It contains three different colored LEDs as Red, Green and Blue. You can get all the colors of the rainbow by lighting these LEDs individually or together.



### Flammable Gas and Smoke Sensor Card

This semiconductor gas sensor, which detects flammable and explosive gas and / or smoke in the environment with a concentration ranging from 300 to 10,000 ppm, can operate between -20 and 50 ° C and draws only 150mA at 5V. Thanks to its analog output, the detected gas concentration can be easily read.



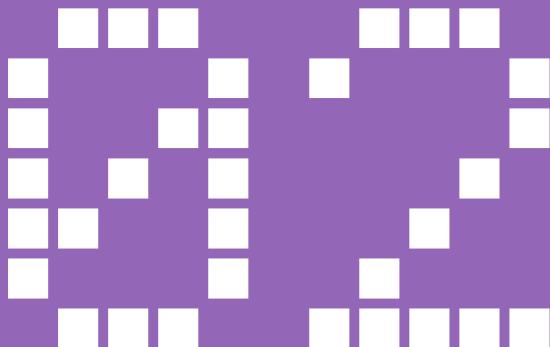
### Sg90 Servo Motor

Tower Pro SG90 is an ideal servo motor for your small mechanisms. It is fully compatible with all brands of remote control receivers and you can use it in your RC vehicles. In addition, you can easily use it in your own robot projects with the PWM signal you can get from many microcontrollers.



### Buzzer

Buzzer; It is a type of auditory warning device that works on mechanical, electromechanical or piezoelectric principles. Buzzers, which have a lot of usage areas, work with the general principle of piezoelectricity. Buzzers can be used in functions such as alarm, timer, confirmation response alert, depending on their area of use.



# Web Based LED Controlling

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

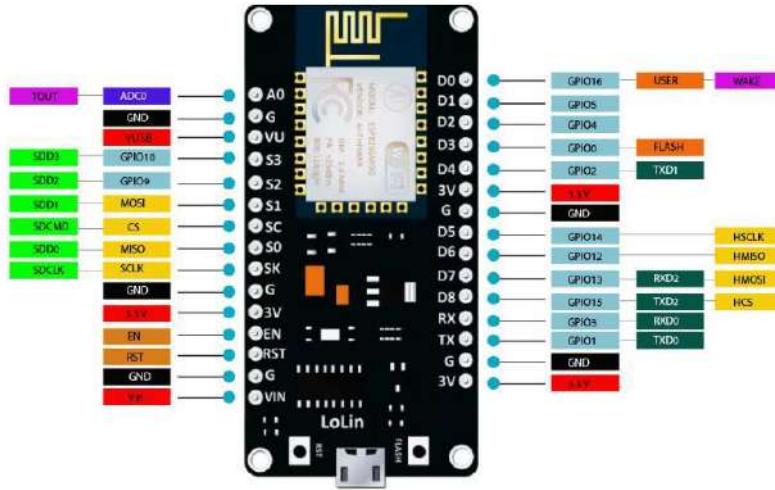
## Web Based Led Controlling

We start our first project with the sentence that the makers use a lot; "Every project starts with lighting". The first project we will do will be to control the LED on the browser, it is a very interesting project for someone who is new to IOT projects.

While doing projects with the NodeMCU development board, we will use the Arduino Ide program. If Arduino Ide is not installed on your computer, you can download it by entering the arduino.cc site. Or you can download it from the download page.

### Necessary materials

- NodeMCU module
- 1 LED
- Breadboard
- A few jumper cables



According to pinout (pin distribution) table, we need to know how to use the NodeMCU pins in this interface, since we will write the software with Arduino IDE. As we will use in our example; For example, GPIO13 pin = corresponds to D7 pin. So when we use the 13th pin in the software, if we look at the pinout table, we also use the D7 pin of the NodeMCU module.

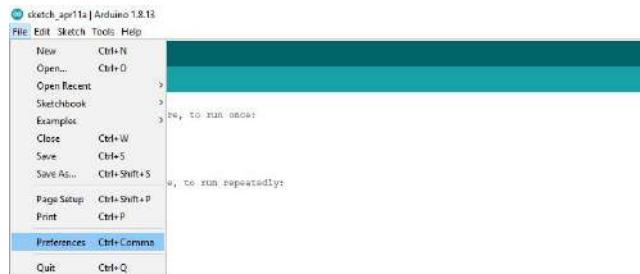
DigitalWrite (13, HIGH); When we use the command, we send logic 1 information to the D7 pin of the NodeMCU module.

I connected the led to the D7 pin, if you want, you can connect it to other pins as long as you specify it in the software.

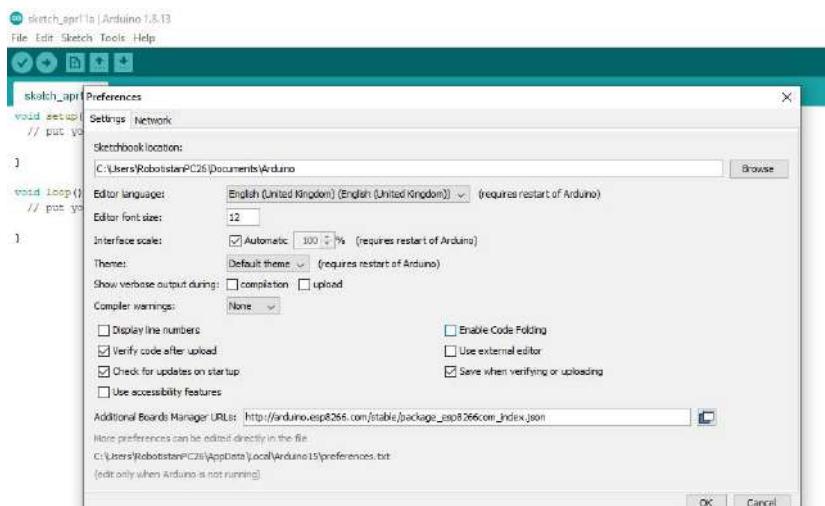
## Web Based Led Controlling

Before writing the software, we need to make a few settings on Arduino Ide. There is no NodeMCU module in the Arduino interface by default.

To integrate the module, we enter the "preferences section from the file menu" located at the top left.

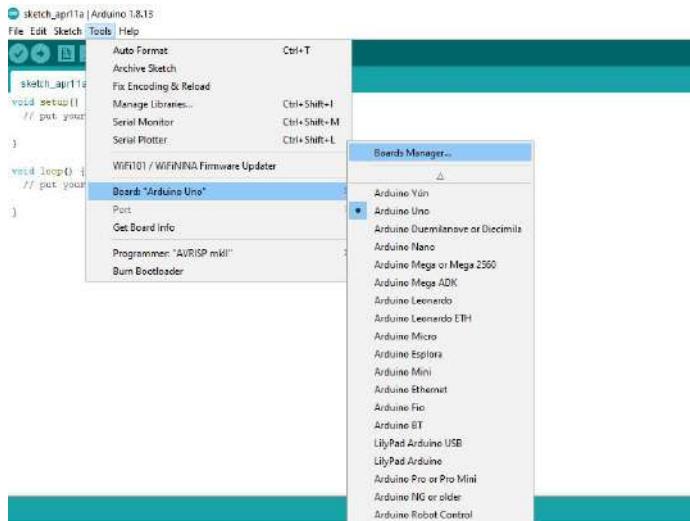


Let's enter this link manually in the section of additional circuit boards manager in the opened tab. Let's say "[http://arduino.esp8266.com/stable/package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable/package_esp8266com_index.json)" and then ok.

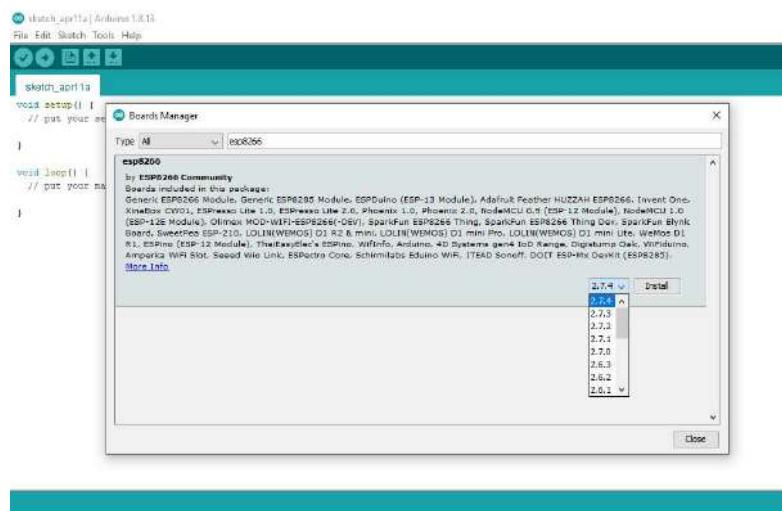


## Web Based Led Controlling

Then we open the card manager from the tools menu.

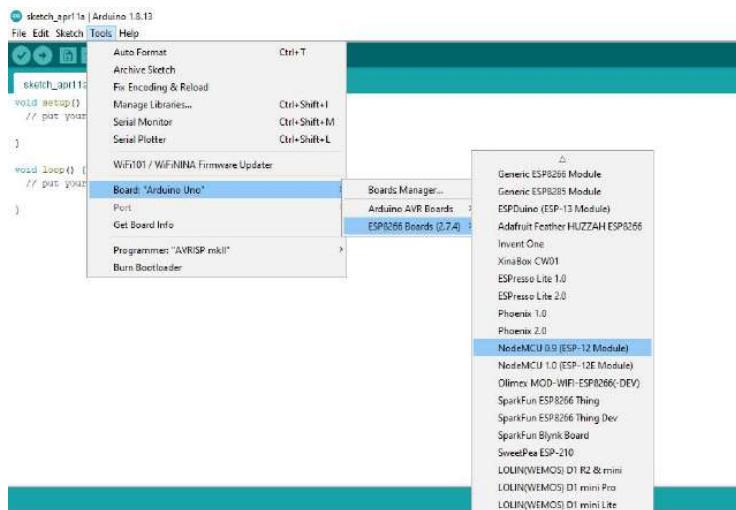


In the window that opens, we search by typing ESP8266 in the search section. Then we select the current version and install it.

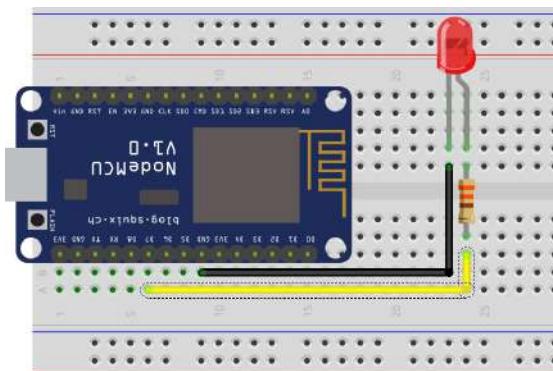


## Web Based Led Controlling

Then we can choose NodeMCU 1.0 (ESP-12E module) from the board section. After making these adjustments, we can now install software on our card.



Now that we have set the Arduino IDE program, we can set up our circuit. Connect the led as shown in the figure to the NodeMCU.



If NodeMCU is clone, we need to install Ch340g driver. There is a detailed explanation on this subject on our Maker blog page.

## Web Based Led Controlling

If everything is ok so far, we can proceed to upload our code.



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** The title bar displays "Arduino" next to a close button.
- Code Area:** The main area contains the C++ code for a WiFi-controlled LED project. The code includes setup and loop functions for connecting to WiFi, printing connection status to Serial, and handling incoming connections from a client to control the LED.
- Line Numbers:** Numerical line numbers are displayed on the left side of the code area, ranging from 1 to 54.

```
1 #include <ESP8266WiFi.h>
2
3 const char* ssid = "wifi-name";
4 const char* password = "wifi-password";
5
6 int led = 13; //
7 WiFiServer server(80);
8
9 void setup() {
10 Serial.begin(115200);
11 delay(10);
12
13 pinMode(led, OUTPUT);
14 digitalWrite(led, LOW);
15
16
17 Serial.println();
18 Serial.println();
19 Serial.print("Connecting to ");
20 Serial.println(ssid);
21
22 WiFi.begin(ssid, password);
23
24 while (WiFi.status() != WL_CONNECTED) {
25 delay(500);
26 Serial.print(".");
27 }
28 Serial.println("");
29 Serial.println("WiFi connected");
30
31
32 server.begin();
33 Serial.println("Server started");
34
35
36 Serial.print("Use this URL to connect: ");
37 Serial.print("http://");
38 Serial.print(WiFi.localIP());
39 Serial.println("/");
40
41 }
42
43 void loop() {
44
45 WiFiClient client = server.available();
46 if (!client) {
47 return;
48 }
49
50
51 Serial.println("new client");
52 while(!client.available()) {
53 delay(1);
54 }
```

## Web Based Led Controlling

```
55
56
57 String request = client.readStringUntil('\r');
58 Serial.println(request);
59 client.flush();
60
61 int value = LOW;
62 if (request.indexOf("/LED-ON") != -1) {
63 digitalWrite(ledPin, HIGH);
64 value = HIGH;
65 }
66 if (request.indexOf("/LED-OFF") != -1) {
67 digitalWrite(ledPin, LOW);
68 value = LOW;
69 }
70
71
72
73 client.println("HTTP/1.1 200 OK");
74 client.println("Content-Type: text/html");
75 client.println("");
76 client.println("<!DOCTYPE HTML>");
77 client.println("<html>");
78
79 client.print("Led Status: ");
80
81 if(value == HIGH) {
82 client.print("ON");
83 } else {
84 client.print("OFF");
85 }
86 client.println("");
87 client.println("<a href=\"/LED-ON\"><button>LED On </button></a>");
88 client.println("<a href=\"/LED-OFF\"><button>LED Off </button></a>");
89 client.println("</html>");
90
91 delay(1);
92 Serial.println("Client disconnected");
93 Serial.println("");
94
95 }
```

After installing the program on our card, let's follow it from the serial monitor. Yes, as you can see, our snow wifi connection was established and he showed us the IP number he received.

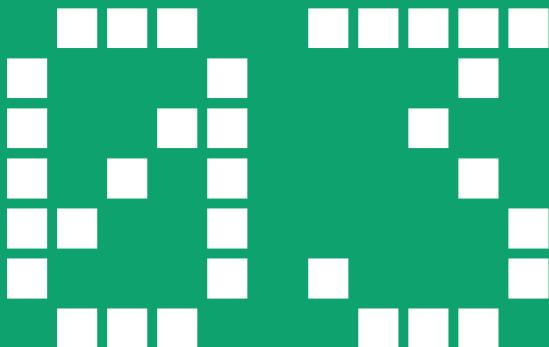
```
Connecting to blue...
...
WiFi connected
Server started
Use this URL to connect: http://192.168.2.184/
```

## Web Based Led Controlling

You should type the IP address on the Serial monitor to our internet browser. You can turn the led on and off thanks to the buttons on the screen that comes across.



In this project, we have learned how to control a led with the Nodemcu module over the local network.



# Flower, Plant Watering

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

## Flower, Plant Watering

Things to do before starting the installation and the project:

First, we need to add our library, which is required for the ESP8266 module. At the same time, since we will use the blynk application during the project, we need to add the blynk application and libraries. You can access libraries and applications from the must-download page in the book.

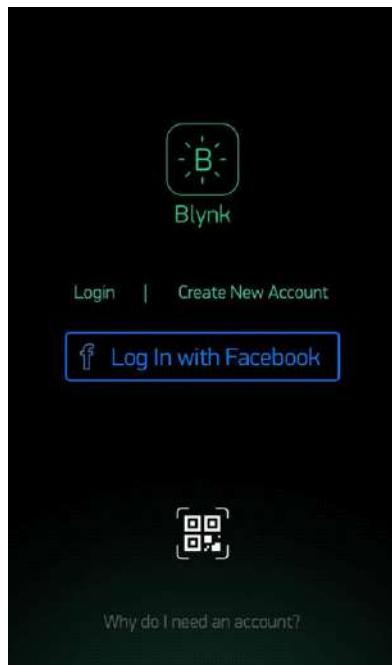
If you do the next two things you need to do in the first project, you do not need to do it again.

You can skip these steps.

Then you need to write [http://arduino.esp8266.com/stable\\_package\\_esp8266com\\_index.json](http://arduino.esp8266.com/stable_package_esp8266com_index.json) in the Additional Circuit Boards Manager field in File >> Preferences.

Go to Tools >> Card >> Card Manager field and write EPS8266 in the search field and load the resulting package card.

I explained this visually in the first project, if you want, you can also get help from there.

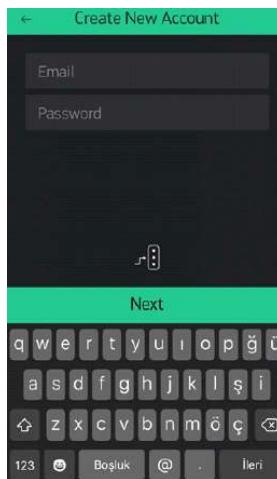


1-) After downloading the application to your phone, this page appears first.

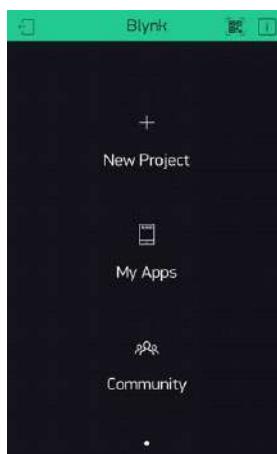
Here you can quickly open an account or log in with your Facebook account.

I recommend opening an account because it is a very easy process.

## Flower, Plant Watering



2-) Create an account and tap the next button.



3-) After opening your account, you will see this page where you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

Press "New Project" to start our project.

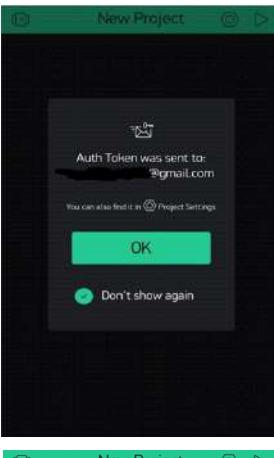


4-) After pressing the New Project section, this page will appear. Here, we give a name to our project and choose which module to use.

Since we are going to use ESP8266, we used WiFi because we wanted it to be remote control by selecting it. If you want, you can choose a different card and choose different transactions.

As other projects, you can control your Arduino over Blynk by selecting Arduino for your home, and it can provide great benefits in your projects using Bluetooth. Anyway, let's go back to our project. Select ESP8266 and select WiFi and then press Create Project.

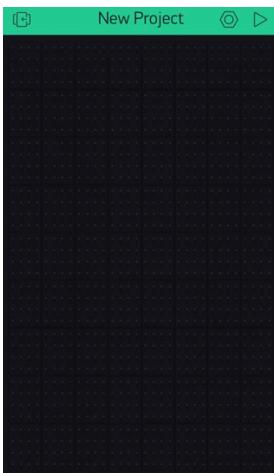
## Flower, Plant Watering



5-) When you click on "Create Project", we will see such a screen. Here, the application sends us the Token, which we will use as a kind of password of our now, to our e-mail address.

We can also see the Token in the settings section of the project.

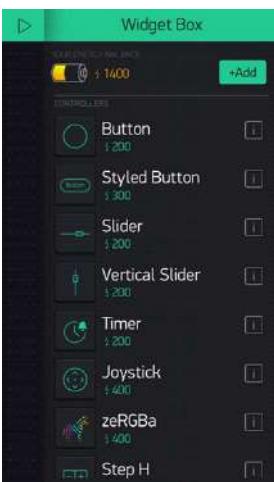
An important point to remember is that the application will give us different Token numbers for each project.



6-) This page is the interface where we will control our projects. This is how Blynk has given us a kind of plan. We can design and size it ourselves.

After this page appears, we have the button to start our project in the upper right corner and the settings section next to it.

You can change the project name and learn your Token in the Settings section. We will put the Wingets we need here on our platform. For this, it will be enough to press the black area.

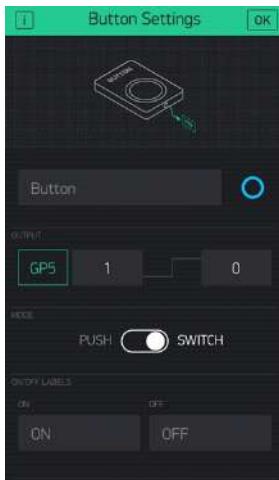


7-) 6. When you press the black area in option, you can see the Control panels, Wingets etc. that are required in a page that moves down from the side. you can see. Blynk is giving us an energy for the projects, the room above is what you'll see in the form of batteries.

You see that each element has a cost below and your budget is set as a default of 1400. If this will not be enough for you, you can start from there by pressing the battery section, but of course this is also paid.

In our flower irrigation project, you need to add a Button to control the water motor and a Gauge to read the soil moisture values.

## Flower, Plant Watering



8-) When you press the button element, the following page will appear. Here we can name and color the button.

The important part for us here is OUTPUT. In the place shown in green, we need to choose the correct Pin that we connected in our circuit.

About this topic, I will give the datasheet of the ESP8266 module while setting up our circuit. Whatever connection you have made, it is very important that you choose the appropriate one here.

For example, I got an output from the D1 pin on the module and because I use this pin as GPIO5, I chose GP5.

The humidity sensor will normally give a value between 0-1023. When we add the Gauge, you will see this range of numbers. Since we do not want to see this range as a humidity range, we will convert it to the value range we want in the software.

We will make the relationship between 0-1023 and 0-100 in the software part and we will read on the V1 pin. Since we will do this analog reading, we need to connect the humidity sensor to the A0 pin.



10-) After adding our elements and determining their pins, what we need to do is go to the circuit and software part.

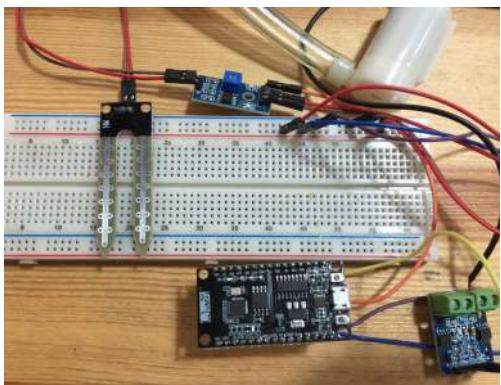
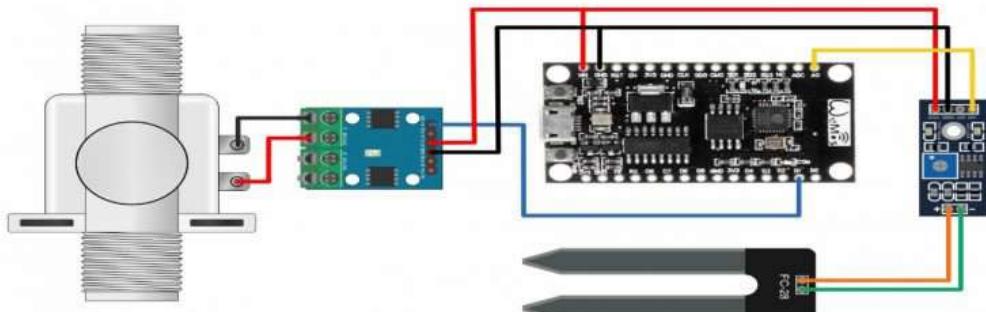
## Flower, Plant Watering

Now that we have installed our application and completed the installation process, we can start to set up our circuit.

### Necessary materials:

- Submersible pump
- ESP8266 WIFI Development board
- Soil moisture sensor
- Jumper cable (10 D-D)
- L9110 Dual motor driver
- Breadboard

After completing the connections, our circuit will be like this.



1. First, you should make the connections that come out of the humidity sensor. After connecting the 2 female side to the female of the sensor, place the GND (-), VCC (+) pin on the 4 female part of the sensor on the breadboard and the A0 pin on the A0 pin on the board for analog reading.

## Flower, Plant Watering

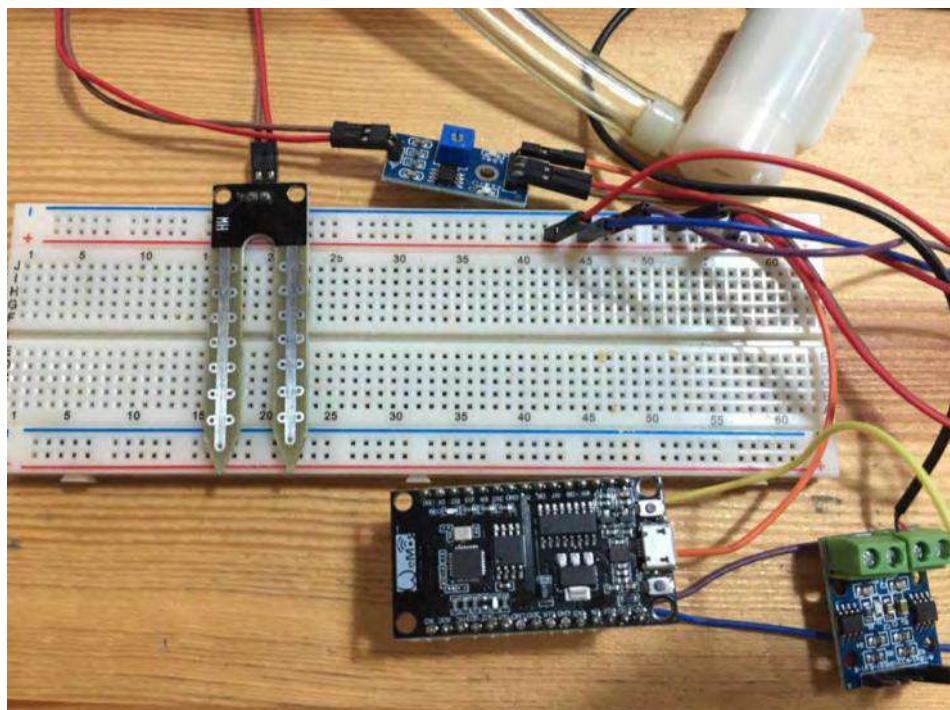
Secondly, you need to complete our motor and motor driver connections. While doing these, place and tighten the cables on the motor in the places located in the motor A part on the motor driver. Then, we place your cables on VCC (+), GND (-) and A1-B pin ends and put them aside to make motor driver energy and use.

Now that we have prepared the soil moisture sensor and motor connection cables, let's continue our circuit.

Take the breadboard in front of you and draw your VCC (+), GND (-) lines from your ESP8266 board to the + and - lines on the board.

Then we connect the cable that you connected to A1-B pin of your motor driver to D1 pin on the board. From here, you will be able to control your engine with your stop-start commands. To complete your motor driver connection, we connect the VCC and GND lines of your motor to the + and - lines respectively on the board.

Then we connect the cable you get from the A0 pin of your soil moisture sensor to the analog input A0 on the card.



## Flower, Plant Watering

This is the code you will upload to our project. In my opinion, I suggest you to look and type the code by hand, so that you will see your mistakes and reinforce yourself.

```
1 #define BLYNK_PRINT Serial
2 #include <ESP8266WiFi.h>
3 #include <BlynkSimpleEsp8266.h>
4
5 char token[] = "blynktoken";
6 char ssid[] = "name";
7 char password[] = "password";
8 int value ;
9
10 #define sensor A0
11 void setup() {
12     Serial.begin(9600);
13
14     Blynk.begin(token, ssid, password);
15
16 }
17 void loop() {
18     value = analogRead(A0);
19     map(value ,0,1023,0,100);
20     Blynk.run();
21 }
```

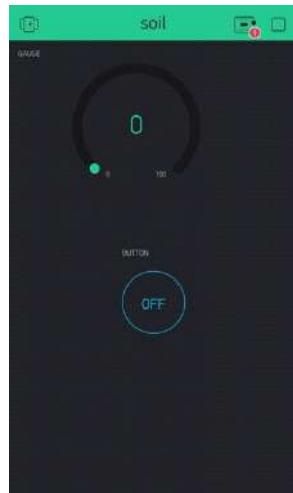
If all the things are ready, let's test the project.

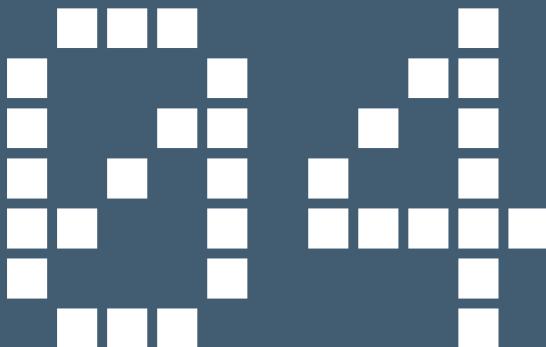


Open the app on your phone and navigate to the part of your project called the platform.

After coming to this part, we press the triangle shaped phrase in the upper right corner.

In this way, our ESP8266 card and phone will be prepared for communication.





# Temperature and Humidity Measurement with DHT11

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

## Temperature and Humidity Measurement with DHT11

With this project, you will be able to follow the temperature and humidity values of any environment through the blynk application.

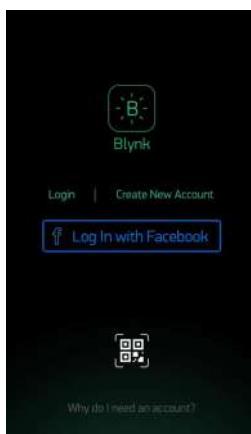
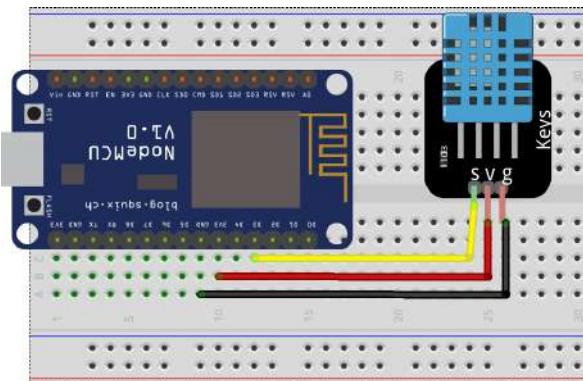
NodeMCU module is not included in Arduino by default. In order to integrate the NodeMCU module into arduino IDE, we explained it in the first project of the book. You can learn how to integrate NodeMCU by looking there.

First you download the blynk application, there are different versions for ios and android. You can download the appropriate version for your phone thanks to the QR code on the source page of the book.

### Required Materials:

- Nodemcu wifi development profit
- Dht11 temperature humidity sensor
- Breadboard
- Enough jumper cables

As can be seen in the connection diagram, we connect the sensor's data pin to D3.

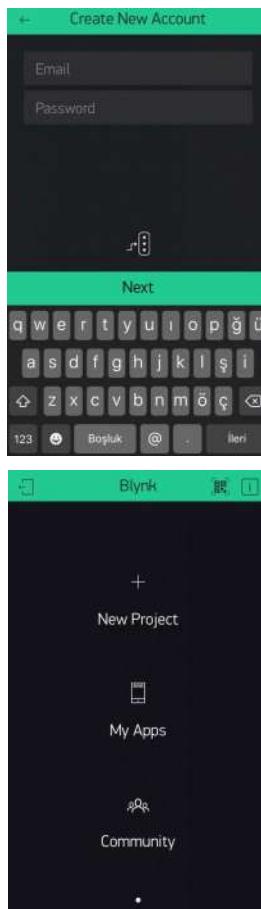


1-) After downloading the application to your phone, this page appears first.

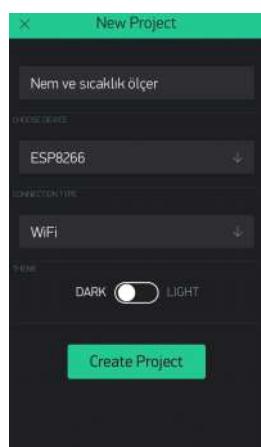
Here you can quickly open an account or log in with your Facebook account.

I recommend opening an account because it is a very easy process.

## Temperature and Humidity Measurement with DHT11



2-) Create an account and tap the next button.

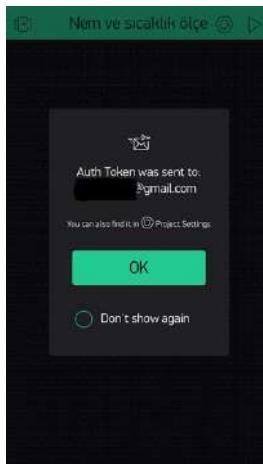


3-) After opening your account, you will see this page where you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

Press "New Project" to start our project.

4-) After pressing the New Project section, this page will appear. Here, we give a name to our project and choose which module to use.

Since we are going to use ESP8266, we used WiFi because we wanted it to be remote control by selecting it. If you want, you can choose a different card and choose different transactions.



5-) When you click on "Create Project", we will see such a screen. Here, the application sends us the Token, which we will use as a kind of password of our now, to our e-mail address.

We can also see the Token in the settings section of the project.

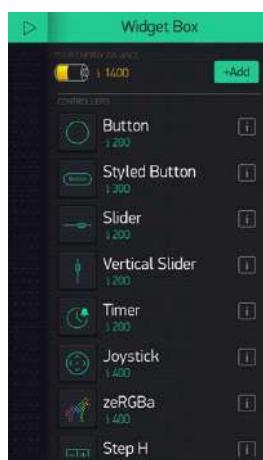
An important point to remember is that the application will give us different Token numbers for each project.



6-) This page is the interface where we will control our projects. This is how Blynk has given us a kind of plan. We can design and size it ourselves.

After this page appears, we have the button to start our project in the upper right corner and the settings section next to it.

You can change the project name and learn your Token in the Settings section. We will put the Wingets we need here on our platform. For this, it will be enough to press the black area.



7-) 6. When you press the black area in option, you can see the Control panels, Wingets etc. that are required in a page that moves down from the side. you can see. Blynk is giving us an energy for the projects, the room above is what you'll see in the form of batteries.

You see that each element has a cost below and your budget is set as a default of 1400. If this will not be enough for you, you can start from there by pressing the battery section, but of course this is also paid.

In this project, you need to add Gauge to read the values.

## Temperature and Humidity Measurement with DHT11



8 -) When we press the gauge element, this page will appear. Here, we can call humidity as the name of gauge.

Whatever connection you have made, it is very important that you choose the appropriate one here.

For example, I output from the D6 pin on the module and because I use this pin as GPIO, I chose GP14.



Our humidity sensor will normally give us a value between 0-1023. When we add the Gauge, we will see this range of numbers. Since we do not want to see this range as a humidity range, we will convert it to the value range we want in the software part.

You should create the relationship between 0-1023 and 0-50 in the software part and read from V6 pin.



Do the same within the temperature gauge. Just don't forget to set your pin here as v5.

After adding your elements and determining the pins, what you need to do now is to go to the software section:

## Temperature and Humidity Measurement with DHT11

If it's all done, you can upload the code :

```
1 #define BLYNK_PRINT Serial
2 #include <ESP8266WiFi.h>
3 #include <BlynkSimpleEsp8266.h>
4 #include <DHT.h>
5
6 char auth[] = "Your Blynk Auth Code";
7 char ssid[] = "SSID";
8 char pass[] = "Password";
9
10#define DHTPIN 0           // D3
11#define DHTTYPE DHT11      // DHT 11
12
13DHT dht(DHTPIN, DHTTYPE);
14BlynkTimer timer;
15
16void sendSensor()
17{
18    float h = dht.readHumidity();
19    float t = dht.readTemperature(); // or dht.readTemperature(true) for Fahrenheit
20
21    if (isnan(h) || isnan(t)) {
22        Serial.println("Failed to read from DHT sensor!");
23        return;
24    }
25    Blynk.virtualWrite(V5, t);
26    Blynk.virtualWrite(V6, h);
27}
28void setup()
29{
30    Serial.begin(9600);
31    Blynk.begin(auth, ssid, pass);
32    dht.begin();
33
34    timer.setInterval(1000L, sendSensor);
35}
36void loop()
37{
38    Blynk.run();
39    timer.run();
40}
```

## Temperature and Humidity Measurement with DHT11

After you've finished this project, wouldn't you like to put it in a nice box?

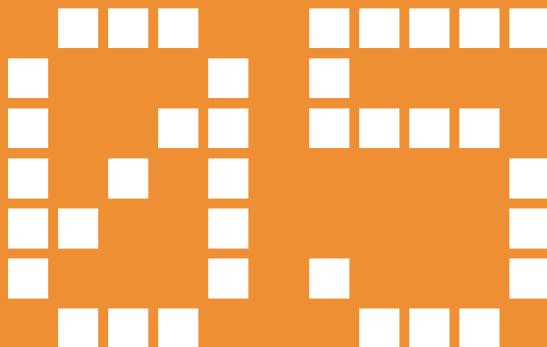
That's why we shared with you a link to a stl file that can be printed with a 3d printer. With a 3d printer, you can print this box and keep your project beautifully.

Also, you can revise this drawing according to yourself, maybe you want to print your own brand on this box.

You can access the drawing files by typing **2201956** in the search section on

[www.thingiverse.com](http://www.thingiverse.com)





# Remote Relay Controlling

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

## Remote Relay Controlling

In this project, with the "Blynk" application, we will implement the application of turning on and off two roles even in different networks.

In other words, whatever you connect to the ends of this role (air conditioner, lamp, motor), you will be able to control it from your home or workplace.

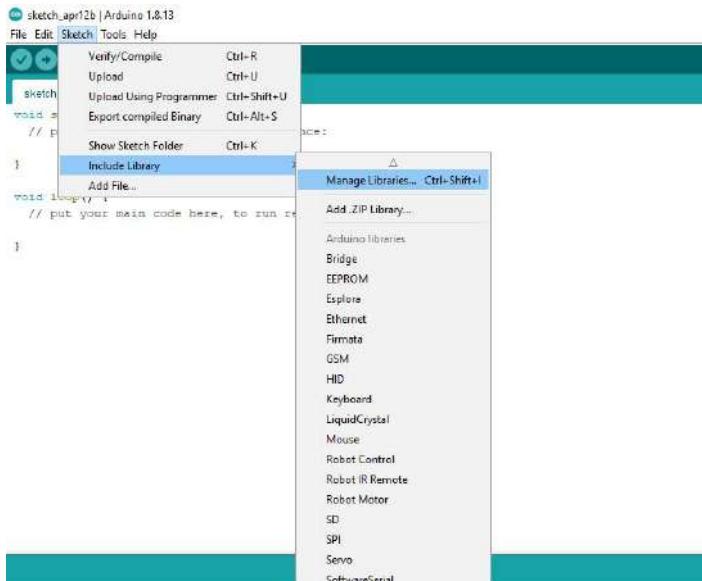
While doing this project, be careful while connecting the high voltage devices that you will connect to the relay.

### Necessary materials:

- Nodemcu wifi development card
- 2 channel relay module
- Breadboard
- Enough jumper cables

First you download the blynk application, it has different versions for iOS and Android. You can download the appropriate version for your phone thanks to the QR code on the source page of the book.

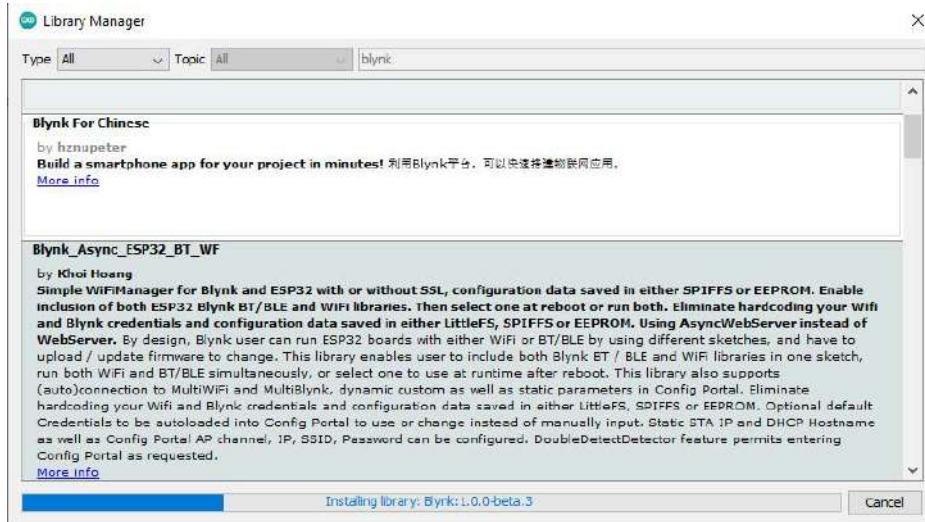
Then the Blynk library for Arduino IDE needs to be installed. To install, we follow the **Draft> Add Library> Manage Libraries** menu.



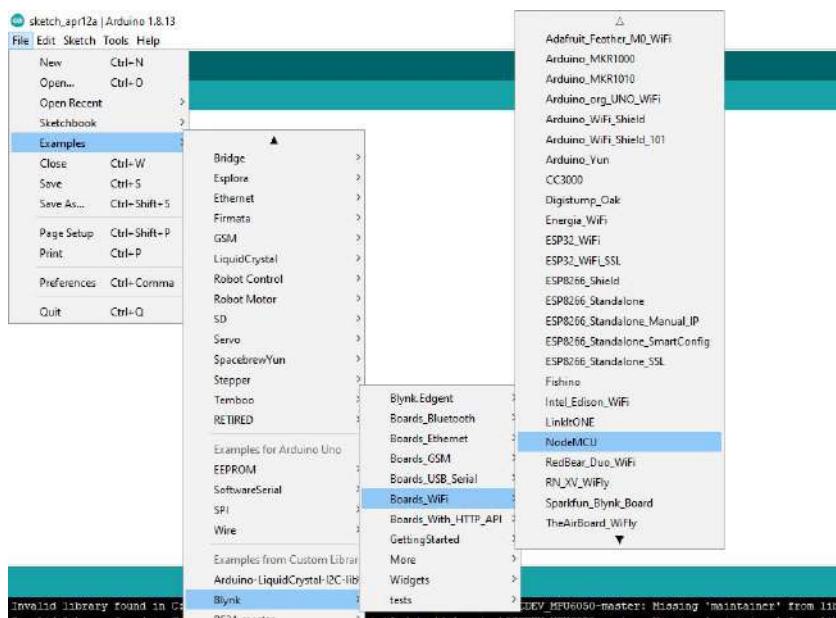
## Remote Relay Controlling

Then you should install the first package by typing blynk in the search section in the window that opens.

After the installation is finished, we close this screen.



After installing the package, we open the code given as an example by following the File > Examples > Blynk > Boards\_wifi > Nodemcu menu.



## Remote Relay Controlling

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

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

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

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

    Blynk.begin(auth, ssid, pass);
    // You can also specify server:
    //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
    //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}

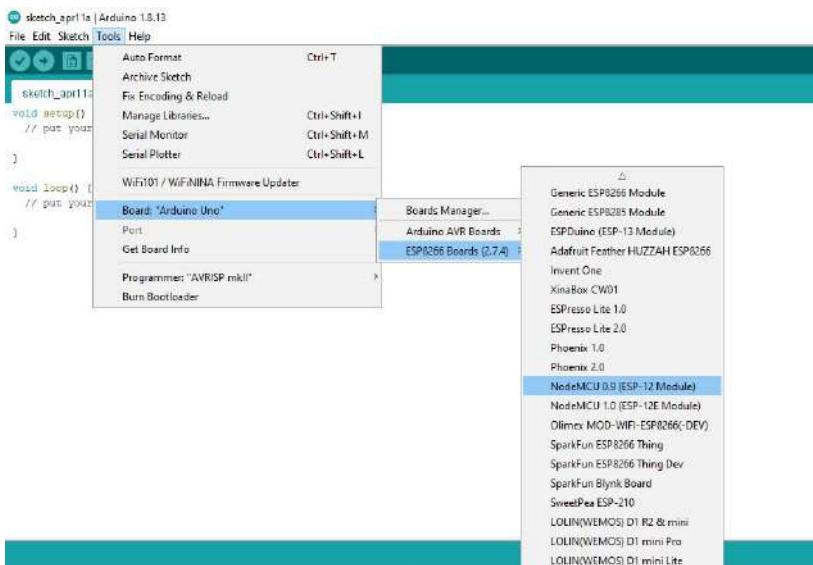
void loop()
```

There are three parts we need to change in this sample NodeMCU code that opens.

You need to paste the token code that the blynk application sends to you as an e-mail to the box marked first.

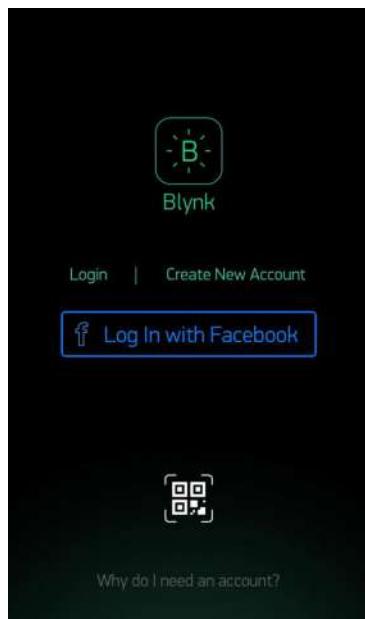
In the second box, you need to enter your wifi name and wifi password..

Then select your card and port to upload the code and upload the code to NodeMCU.

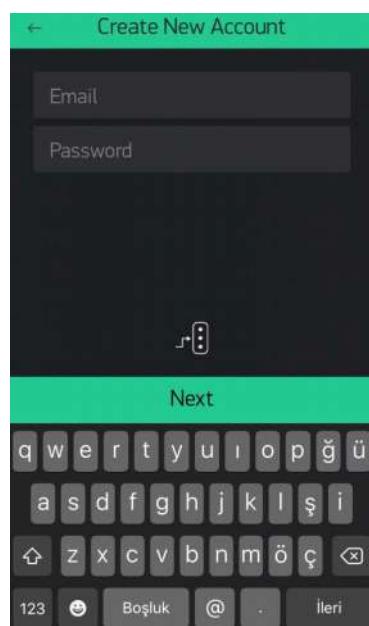


## Remote Relay Controlling

If you do not encounter any problems so far, you have settled the processes we can do from the computer. Now, in the blynk application, there are settings you need to do.

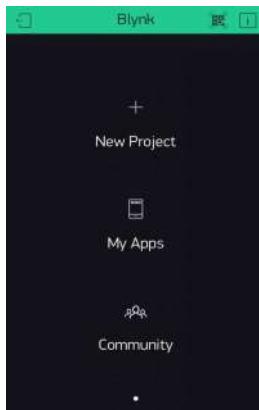


1-) After downloading the application to your phone, this page appears first. Here you can quickly open an account or log in with your Facebook account. I recommend opening an account because it is a very easy process.



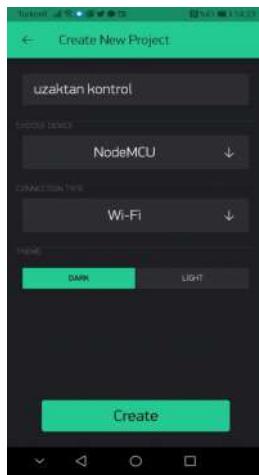
2-) Create an account now and click the next button.

## Remote Relay Controlling



3-) After opening your account, you will see this page where you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

We press "New Project" to start our project.

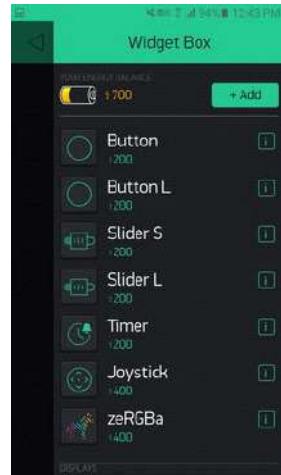


4) Enter the name of your project, select the tool you will use and show the link path. Continue with Create.

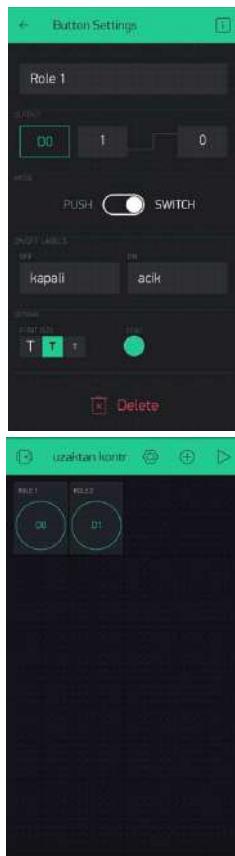


Using the interface is very simple. We click the + sign and add 2 buttons there.

By clicking on the buttons, we can select both the terminals and change their names etc.



## Remote Relay Controlling



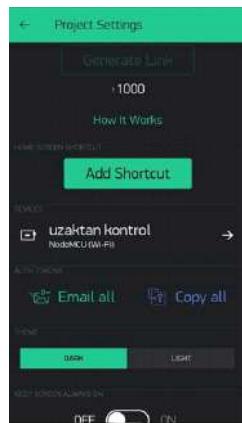
This is the final version of the code. I deleted some of the comments so that you don't get confused.

Note: Since our relay outputs when it is in the 0 position, we wrote the 1 and 0 states in reverse. We wrote "open" to On state, "closed" to Off state. We wanted it to be in the form of a switch to make it easier.



Our interface will look like this and you have one last action left to do. Come to the Settings section and send the code we need to write in the "YourAuthToken" section that you have specified above, to your e-mail address by clicking the "email all" section.

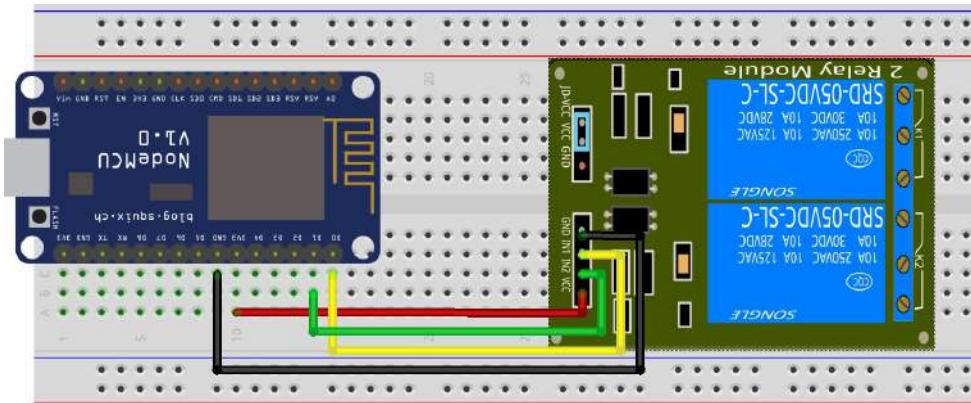
```
Arduino
1 #define BLYNK_PRINT Serial
2 #include <ESP8266WiFi.h>
3 #include <BlynkSimpleEsp8266.h>
4
5 char auth[] = "YourAuthToken";
6
7 char ssid[] = "YourNetworkName";
8 char pass[] = "YourPassword";
9
10 void setup()
11 {
12     Serial.begin(9600);
13
14     Blynk.begin(auth, ssid, pass);
15 }
16
17 void loop()
18 {
19     Blynk.run();
20 }
```



"YourAuthToken";  
"YourNetworkName";  
"YourPassword"; Do  
not forget to change  
parts of it.

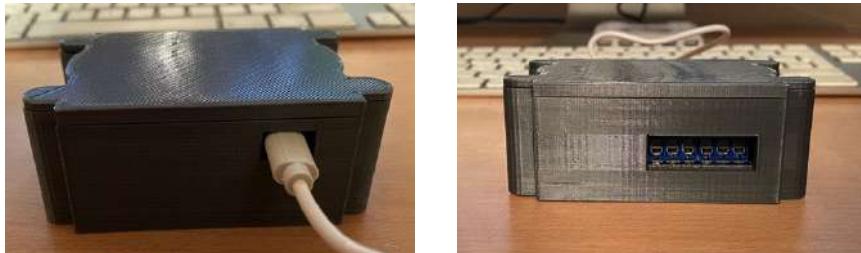
## Remote Relay Controlling

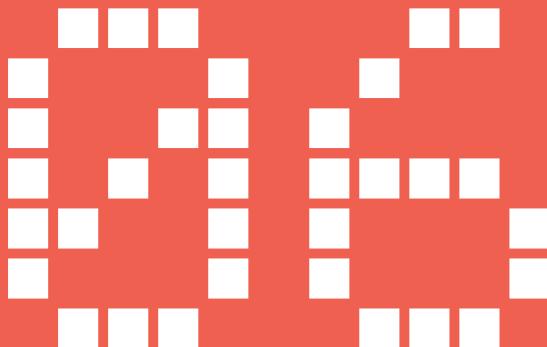
As in our connection scheme; Connect the D0 pin to the In1 end of our 2-channel relay, the D1 pin to the In2 end, and connect the 3V GNDs to the VCC part. In this way, we set up our circuit.



We shared with you a link to a stl file that can be printed with a 3D printer so that you can put the project in a nice box. With a 3D printer, you can print this box and keep your project beautifully. Also, you can revise this drawing according to yourself, maybe you want to print your own brand on this box.

You can access the drawing files by typing the code 4093462 in the search section on [www.thingiverse.com](http://www.thingiverse.com).





# Blynk Rgb LED Controlling

robotistan

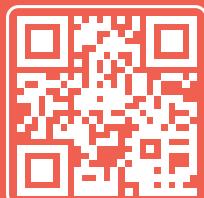


BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)

YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

## Blynk Rgb Led Controlling

In this project, we will do rgb diode led control with blynk application. By turning it into a project such as a night light or accessory, you can create a beautiful environment in your home.

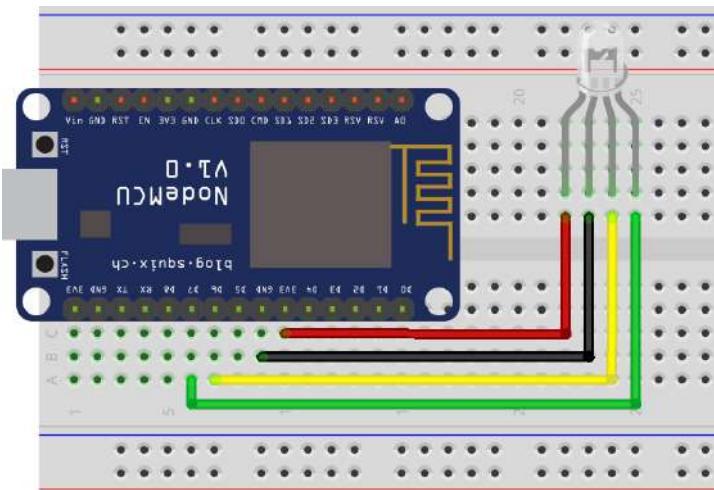
Necessary materials:

- Nodemcu wifi development card
- rgb led diode
- Breadboard
- Enough jumper cables

Since we will use Arduino Ide while doing the project, NodeMCU package is not loaded on Arduino Ide. That's why you need to install Nodemcu and blynk packages, you can see how the package is installed from previous projects.

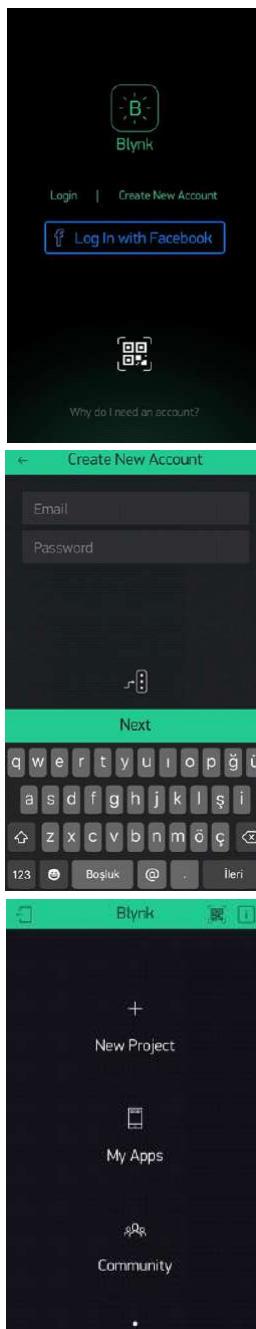
Likewise, you can download the blynk program we use in the project from the download page in the book.

First we will make the connections of the circuit. The important thing here will be that you do not confuse the negative end of the LED.



You will connect the red color pin of the LED to d6, green d7, blue d8 pins.

## Blynk Rgb Led Controlling



1-) After downloading the application to your phone, this page appears first.

Here you can quickly open an account or log in with your Facebook account.

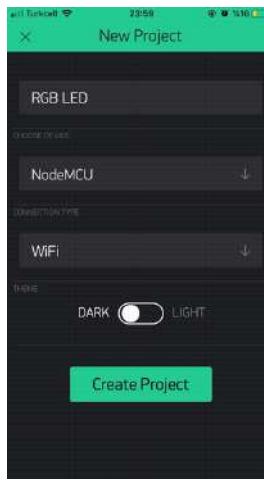
I recommend opening an account because it is a very easy process.

2-) Create an account now and tap the next button.

3-) After opening your account, you will see this page where you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

We press New Project to start our project.

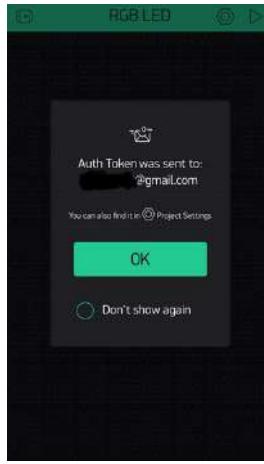
## Blynk Rgb Led Controlling



4-) After pressing the New Project section, this page will appear. Here, we give a name to our project and choose which module to use.

Since we are going to use NodeMCU, we used WiFi because we wanted it to be controlled remotely by selecting it.

After selecting ESP8266 and selecting WIFI, we click Create Project.



5-) When you press Create Project, you will see a screen like this. Here, the application sends us the Token, which you will use as a kind of password for our communication, to your e-mail address.

You can also see the Token in the settings section of the project.

An important point to remember is that the application will give us different Token numbers for each project.

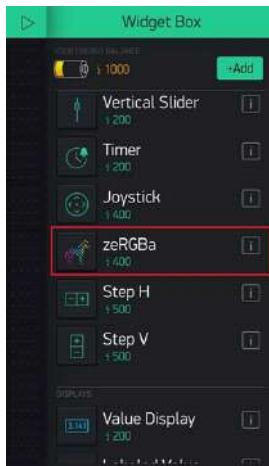


6-) This page is the interface where you can control our projects. Blynk has given us some sort of platform in this way. You can design and size it ourselves.

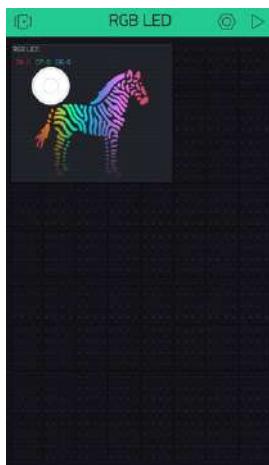
After you see this page, there is the button to start our project in the upper right corner and the settings section next to it.

You can change the project name and learn your Token in the settings section. We will put the zeRGBa you need here on our platform. For this, it will be enough to press the black area.

## Blynk Rgb Led Controlling



7) Press on it to add the zeRGBa you need in the project.

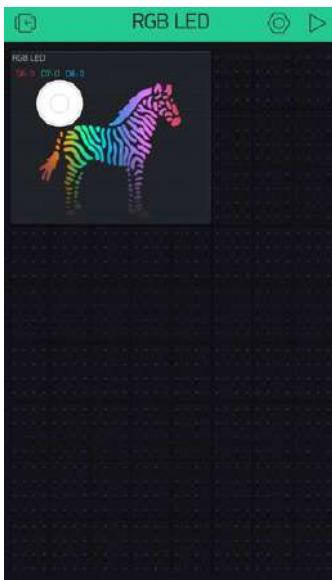


8) Open the setting menu by tapping the zeRGBa added to the interface.



9) Set the D6, D7, D8 digital pins to which the LED will be connected and the values 0 to 255 from the settings menu that opens and press the ok button.

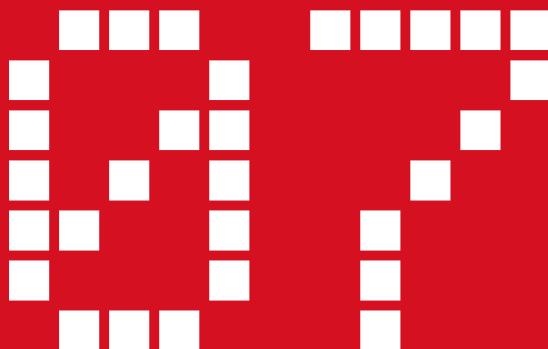
## Blynk Rgb Led Controlling



10) After making all the settings, your screen will look like this. You can start to adjust the color of the led by pressing the play button in the upper right corner.

Now that the adjustments are done on the Blynk application, you can upload the code to the NodeMCU card. Before uploading the code, do not forget to paste the token code sent to your e-mail address, and write the wifi name and password where necessary. After uploading the code, your project will be completed.

```
1 #define BLYNK_PRINT Serial
2 #include <ESP8266WiFi.h>
3 #include <BlynkSimpleEsp8266.h>
4
5 char auth[] = "-----";
6
7 void setup()
8 {
9     Serial.begin(9600);
10    Blynk.begin(auth, "-----", "-----");
11 }
12
13 void loop()
14 {
15     Blynk.run();
16 }
```



# Smoke and Gas Detection

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)



YouTube



[youtube.com/robotistan](http://youtube.com/robotistan)

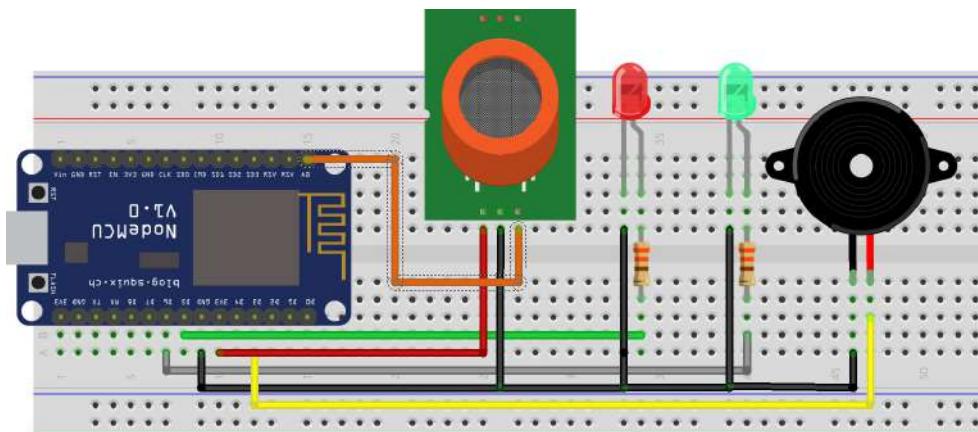
## Smoke and Gas Detection

In this project, by using the blynk program from your phone, you will be able to measure the gas and smoke level of the environment thanks to the data from the gas sensor thanks to the NodeMCU snow over the wifi network, a notification will be sent to our phone above a certain level and the buzzer will work.

### Necessary materials:

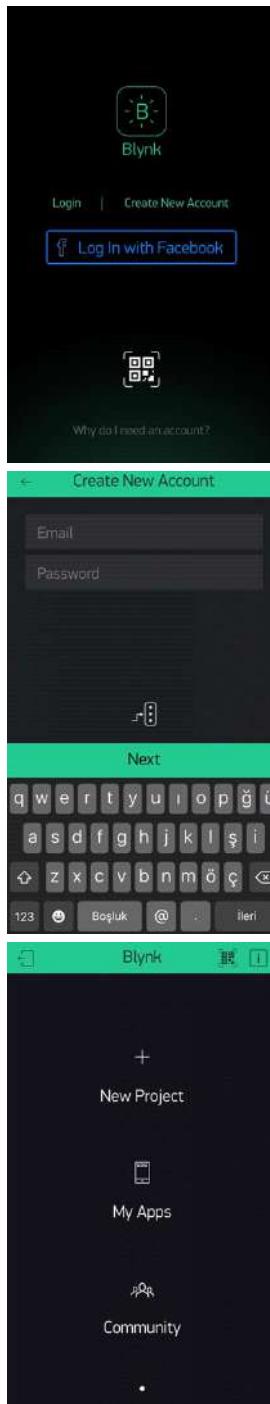
- NodeMcu
- 2 LEDs
- 2 150 ohm Resistors
- MQ2 Gas sensor
- buzzer
- Breadboard
- Jumper Cables

Make the connections as shown in the circuit diagram. Connect the leds to the d5 and d6 pins and the buzzer to the d3 pin. Connect the gas sensor to analog A0 pin.



Next is the installation of the blynk application and the necessary adjustments. You can download the Blynk application thanks to the QR code on the need to download page.

## Smoke and Gas Detection



1-) After downloading the application to your phone, this page appears first.

Here you can quickly open an account or log in with your Facebook account.

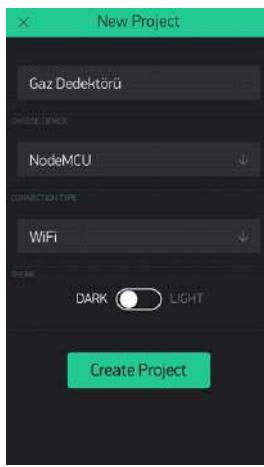
I recommend opening an account because it is a very easy process.

2-) Create an account now and tap the next button.

3-) After opening your account, you will see this page where you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

We press New Project to start our project.

## Smoke and Gas Detection

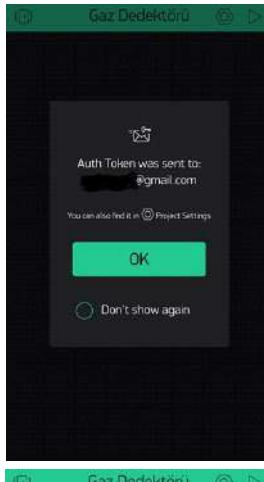


4-) After pressing the New Project section, this page will appear. Here, we give a name to our project and choose which module to use.

We chose it because we will use the nodemcu, and we used WiFi because we wanted it to be remote control as the communication.

If you want, you can choose a different card here and choose different communication types.

After selecting ESP8266 and selecting my WiFi, we press Create Project.



5-) When you press Create Project, you will see a screen like this. Here, the application sends us the Token, which we will use as a password for communication, to our mail address.

We can also see the Token in the settings section of the project.

An important point to remember is that the application will give us different Token numbers for each project.

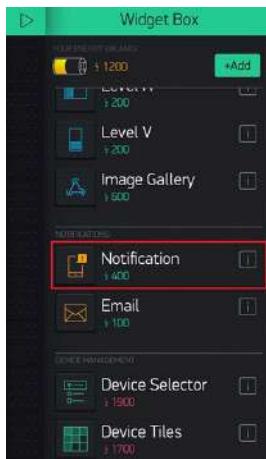


6-) This page is the interface where we will control our projects. Blynk gives us a platform like this. We can design and size it ourselves.

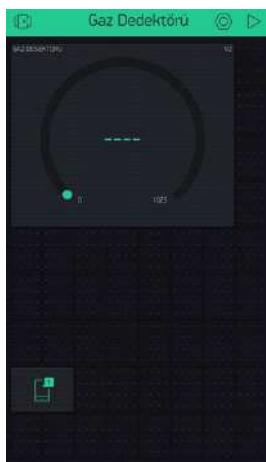
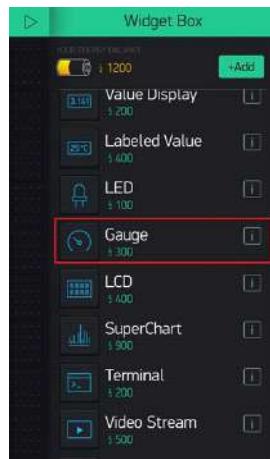
After this page appears, we have the button to start our project in the upper right corner and the settings section next to it.

You can change the project name and learn your Token in the Settings section.

## Smoke and Gas Detection



7) We add the "notification" feature for the Blynk application to send us notifications. Likewise, we add the gauge to be able to see the data we receive from the sensor.

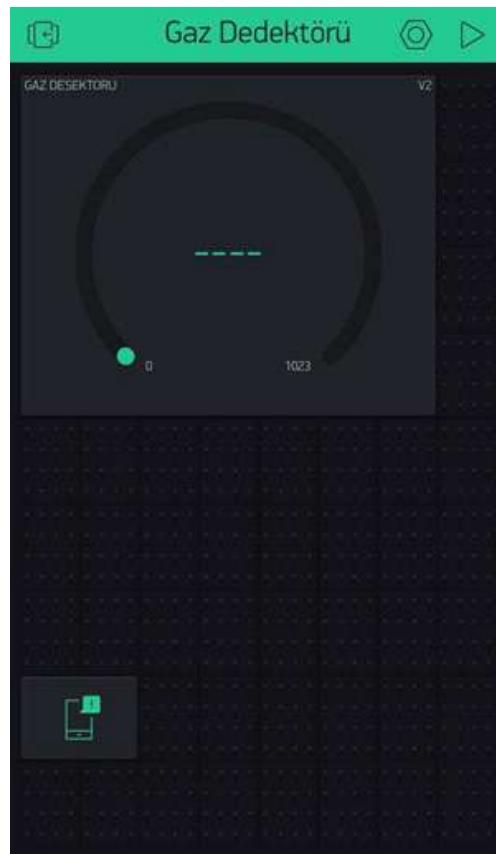


8) Open the setup menu by touching the gauge added to the interface.



9) From the settings menu that opens, select the v2 pin to which the gas sensor is connected and set the refresh interval as 1 second.

## Smoke and Gas Detection



10) After making all the settings, your screen will look like this. After the project is finished, you can start getting information from the gas sensor by pressing the play button in the upper right corner.

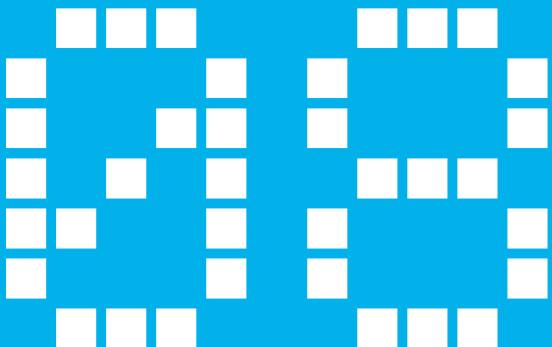
Now that the adjustments are done on the Blynk application, you can upload the code to the NodeMCU card. Before uploading the code, do not forget to paste the token code sent to your e-mail address, and write the wifi name and password where necessary.

Before running the codes, make sure that the blynk library and Esp8266 boards are loaded into the arduino IDE. We talked about how to install libraries in projects 1 and 4, and you can get help from there.

## Smoke and Gas Detection

And the project codes as follows :

```
 1 #include <ESP8266WiFi.h>
 2 #include <BlynkSimpleEsp8266.h>
 3 #define BLYNK_PRINT Serial
 4
 5 char auth[] = "blynk token";
 6 char ssid[] = "wifi name";
 7 char pass[] = "wifi password";
 8
 9 BlynkTimer timer;
10
11 int Sensor = A0;
12 int data = 0;
13 #define greenLed 14 //D5
14 #define redLed 12 // D6
15 int buzzer = 0; // D3
16
17 void setup()
18 {
19   Serial.begin(115200);
20   Blynk.begin(auth, ssid, pass);
21   pinMode(greenLed, OUTPUT);
22   pinMode(redLed, OUTPUT);
23   timer.setInterval(1000L, getSendData);
24 }
25
26 void loop()
27 {
28   timer.run();
29   Blynk.run();
30 }
31
32 void getSendData()
33 {
34   data = analogRead(Sensor);
35   Blynk.virtualWrite(V2, data);
36
37   if (data > 300)
38   {
39     Blynk.notify("Attention !!! Your Gas Sensor Has Worked");
40     digitalWrite(greenLed, LOW);
41     digitalWrite(redLed, HIGH);
42     tone(buzzer, 1000, 200);
43   }
44
45   else
46   {
47     digitalWrite(greenLed, HIGH);
48     digitalWrite(redLed, LOW);
49     noTone(buzzer);
50   }
51 }
```



# Servo Motor Controlling

robotistan



BLOG



[maker.robotistan.com](http://maker.robotistan.com)

FORUM



[forum.robotistan.com](http://forum.robotistan.com)

YouTube



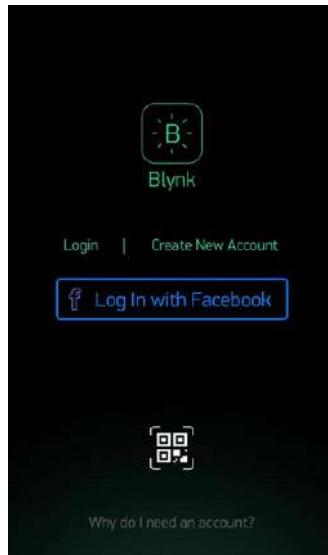
[youtube.com/robotistan](http://youtube.com/robotistan)

## Servo Motor Controlling

In this project, you will make a servo motor control project with blynk application over wifi. You can download the Blynk application from the download must page.

You can also download the blynk library that we will use in the code section from the same page.

We start doing the project by making adjustments in the blynk application at first.

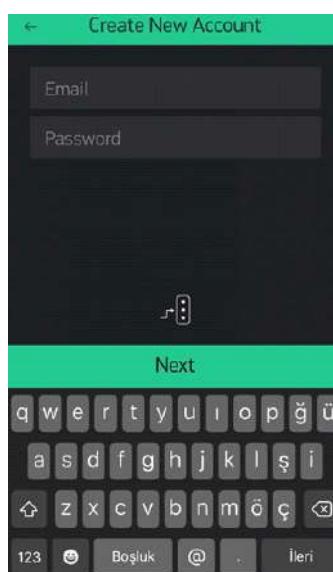


1-) After downloading the application to your phone, this page appears first.

Here you can quickly open an account or log in with your Facebook account.

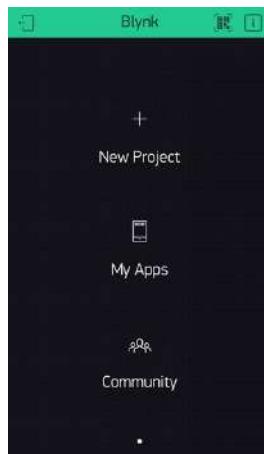
I recommend opening an account because it is a very easy process.

We start doing the project by making adjustments in the blynk application at first.



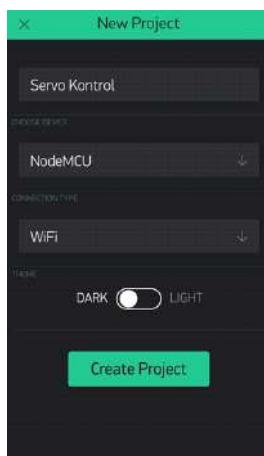
2-) Create an account now and tap the next button.

## Servo Motor Controlling



3-) After opening your account, this page appears here, you can start a new project and get information about the application. Since it has a very simple interface, you will understand everything better if you try it a little.

Press **New Project** to start the project.

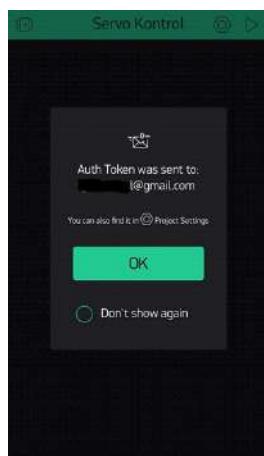


4-) After pressing the New Project section, this page will appear. Here, we give a name to our project and choose which module to use.

We chose it because we will use NodeMCU, we used WiFi as we want remote control as the communication type.

If you want, you can choose a different card here and choose different communication types.

After selecting ESP8266 and selecting WiFi communication, we press Create Project.



5-) When you press Create Project, you will see a screen like this. Here, the application sends us the Token, which we will use as a kind of password of our now, to our e-mail address.

We can also see the Token in the settings section of the project.

An important point to remember is that the application will give us different Token numbers for each project.

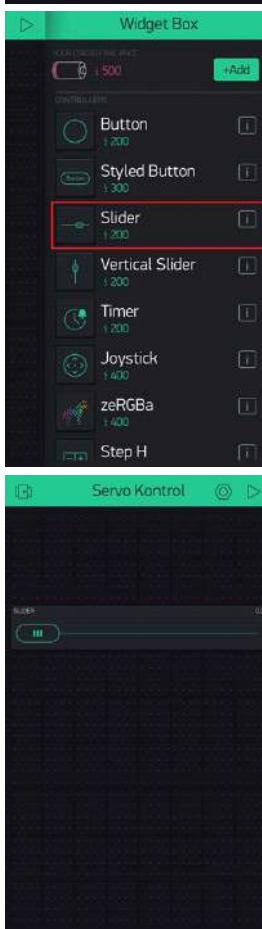
## Servo Motor Controlling



6-) This page you see is the interface where you will control your projects. Blynk gives us a platform like this. We can design and size it ourselves.

After this page appears, we have the button to start our project in the upper right corner and the settings section next to it.

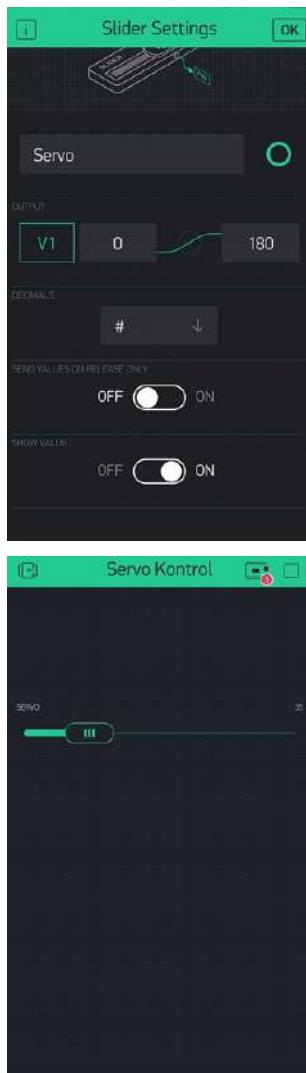
You can change the project name and learn your Token in the Settings section.



7)Add slider to control the angle of the servo motorvia the Blynk app.

8) Open the settings menu by tapping the slideradded to the interface.

## Servo Motor Controlling



9)Select the v1 pin to which the servo motor isconnected from the settings menu that opens.Since the servo motor can rotate between 0 and180 degrees, set the value range from 0 to 180.

10)After making all the settings, your screen willlook like this. After your project is finished, you cancontrol the servo motor by pressing the play buttonin the upper right corner.

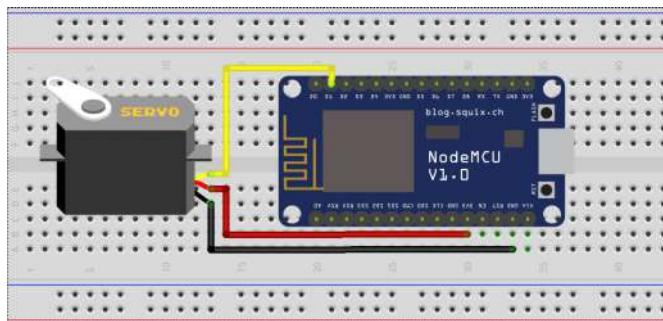
Now that the adjustments are finished on the Blynk application, we can proceed to set up the circuit.

### Necessary materials:

- NodeMcu
- Servo Motor
- Breadboard
- Jumper Cable

## Servo Motor Controlling

Since the servo motor works with 5v, we connect the positive end to the "vin" input of the NodeMCU, this part is important. Connect the signal end to "d1", and the minus end to "gnd".



Now that you have set up the circuit, you can upload the code to the NodeMCU card. Before uploading the code, do not forget to paste the token code sent to your e-mail address where necessary, and write the wifi name and password in the required places.

Before running the codes, make sure that the blynk library and Esp8266 boards are loaded into the arduino IDE. We talked about how to install libraries in projects 1 and 4, you can get help from there. You can download libraries from the must-download page.

You can find the complete code on the download required page.

```
#include <Servo.h>
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
Servo servo;
char auth[] = " ";
char ssid[] = " ";
char pass[] = " ";
BLYNK_WRITE(V0) {
    servo.write(param.asInt());
}
void setup() {
    Serial.begin(9600);
    servo.attach(D1);
    Blynk.begin(auth, ssid, pass);
}
void loop() {
    Blynk.run();
}
```

## Notes

## Notes





# YouTube



[youtube.com/robotistan](https://youtube.com/robotistan)

# FORUM

robotistan



[forum.robotistan.com](https://forum.robotistan.com)

# BLOG

robotistan



[maker.robotistan.com](https://maker.robotistan.com)