

Smart Home Using NodeMCU ESP8266 and Blynk 2.0 With Real-time Feedback | IoT - 2021



by techstudycell

In this IoT project, I have shown how to make IoT-based Smart Home Automation using the new **Blynk 2.0** & NodeMCU ESP8266 to control a 4-channel relay module from the manual switch & smartphone using the **Blynk IoT App**.

During the article, I have shown all the steps to make this Blynk home automation system.

This **Blynk ESP8266** control smart relay has the following features:

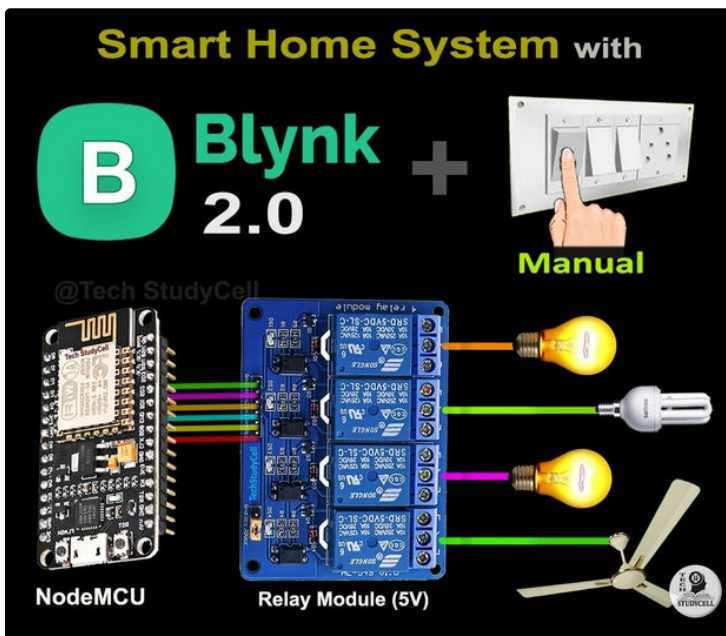
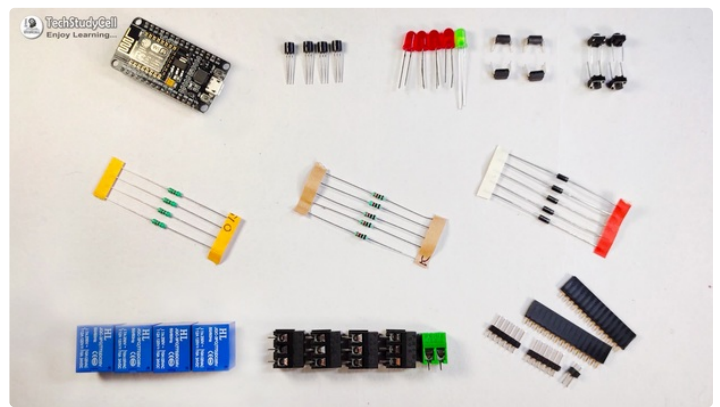
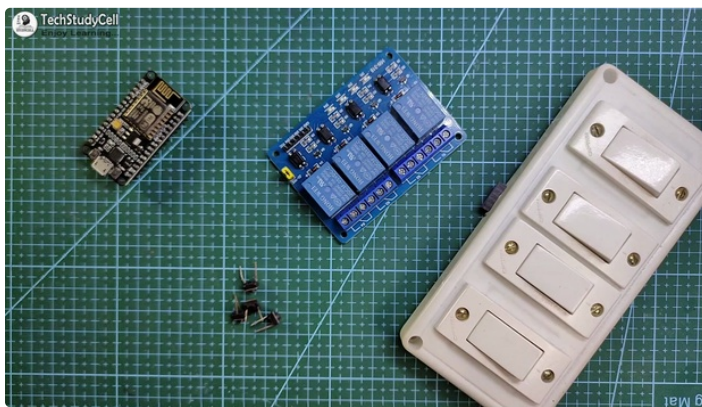
1. Control home appliances with WiFi (**Blynk IoT App**).
2. Control home appliances with **Blynk web dashboard**.
3. Control home appliances with **manual switches** or push buttons.
4. Monitor **real-time feedback** in the Blynk IoT App.

So, you can easily make this home automation project at home just by using a NodeMCU and relay module. Or you can also use a custom-designed PCB for this project.

Supplies:

You can make this project just by using NodeMCU and 4-channel relay module. But if you use PCB then you need the following components.

- NodeMCU
- Relays 5v (SPDT) (4 no)
- BC547 Transistors (4 no)
- PC817 Optocouplers (4 no)
- 510-ohm 0.25-watt Resistor (4 no) (R1 - R4)
- 1k 0.25-watt Resistors (5 no) (R5 - R9)
- LED 5-mm (5 no)
- 1N4007 Diodes (5 no) (D1 - D5)
- Push Buttons (4 no)
- Terminal Connectors
- 5V DC supply



https://youtu.be/CpUVssHPm_s

Step 1: Circuit Diagram of the NodeMCU Home Automation Project

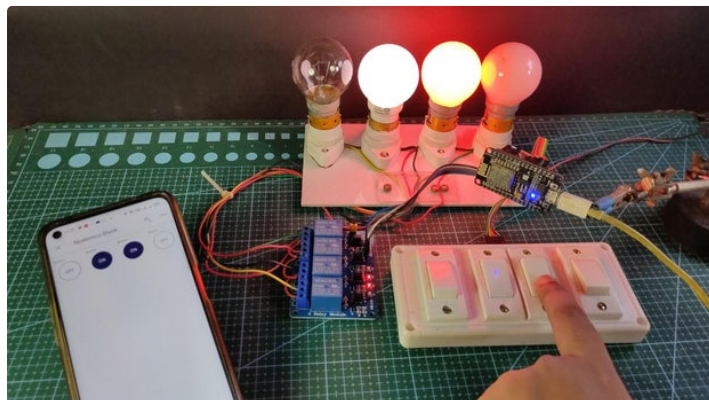
The D3 pin should not be connected with GND during

the booting process of NodeMCU.

You can also control the relays from the switches or pushbuttons.

You can monitor the real-time feedback in the Blynk IoT App.

Please refer to the circuit diagram to connect the pushbuttons or switches.

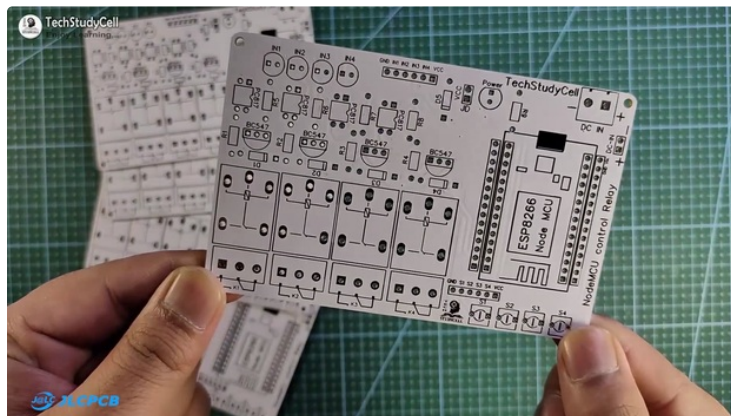
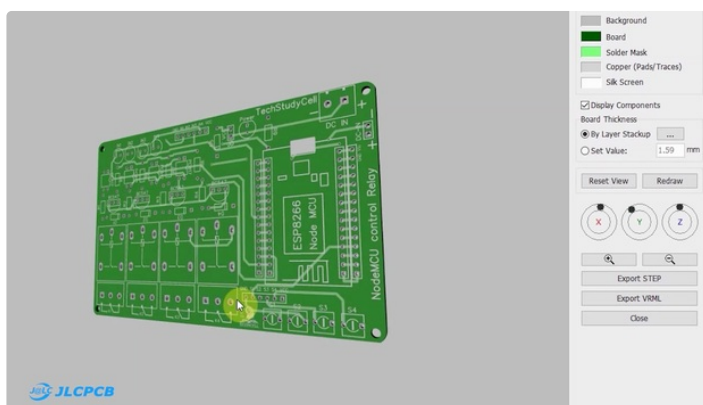
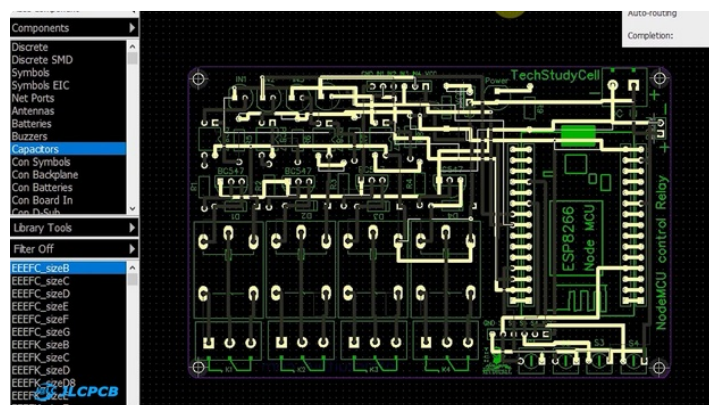


Step 4: Design the PCB for This Smart Home System

To make the circuit compact and give a professional look, I have designed the PCB after testing all the features of the smart relay module.

You can download the PCB Gerber file of this home automation project from the following link:

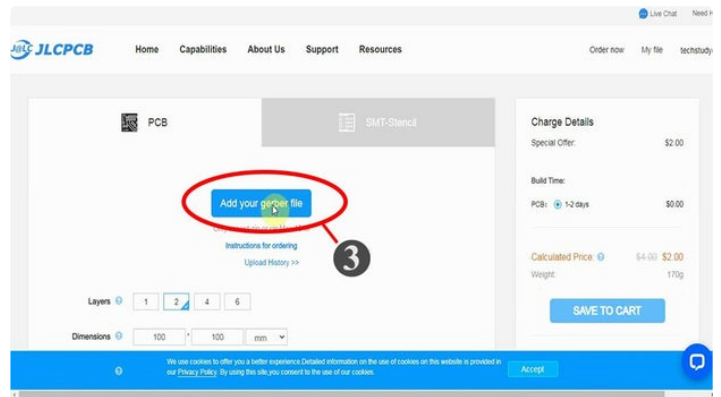
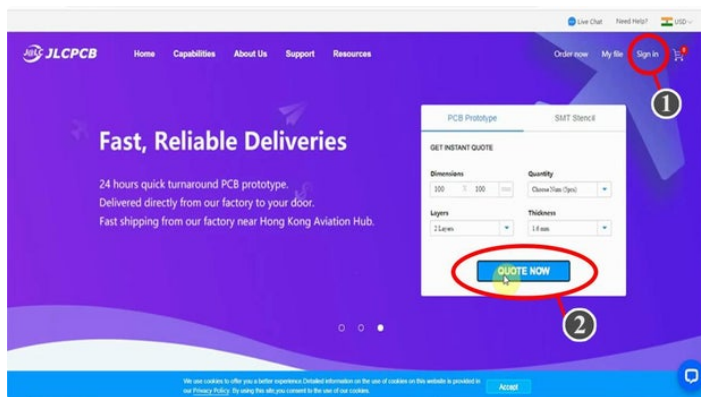
<https://drive.google.com/uc?export=download&id=1J...>



Step 5: Order the PCB

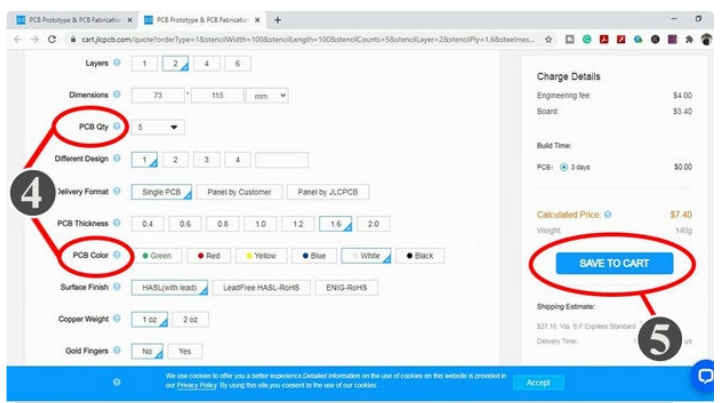
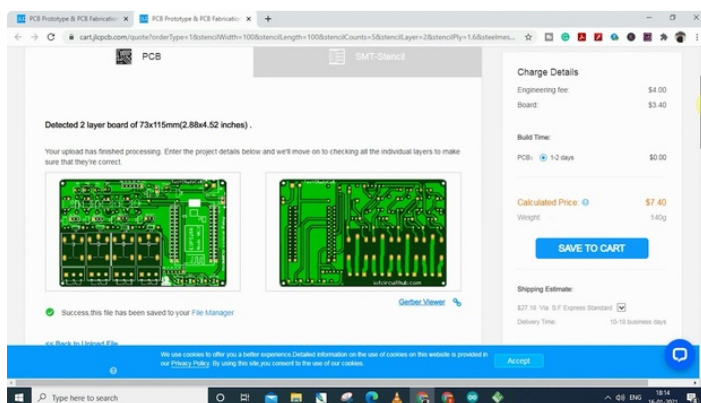
After downloading the Gerber file you can easily order the PCB

1. Visit <https://jlcpcb.com/RHS> and **Sign in / Sign up**
2. Click on the **QUOTE NOW** button.
3. Click on the **"Add your Gerber file"** button. Then browse and select the Gerber file you have downloaded.



Step 6: Uploading the Gerber File and Set the Parameters

4. Set the required parameter like **Quantity**, **PCB masking color**, etc
5. After selecting all the Parameters for PCB click on **SAVE TO CART** button.



Step 7: Select Shipping Address and Payment Mode

6. Type the **Shipping Address**.

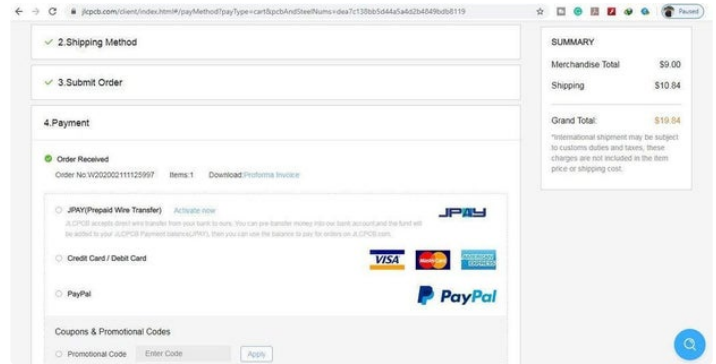
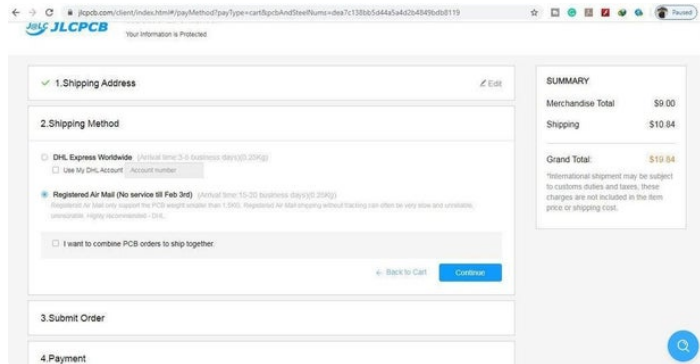
7. Select the **Shipping Method** suitable for you.

8. **Submit the order** and proceed with the **payment**.

You can also track your order from JLCPCB.com.

My PCBs took 2 days to get manufactured and arrived within a week using the DHL delivery option.

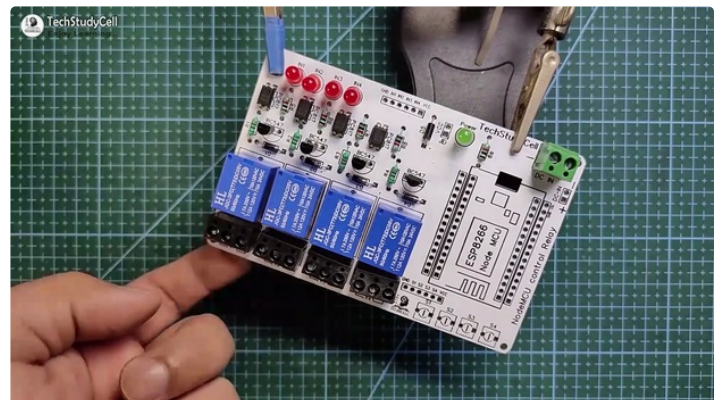
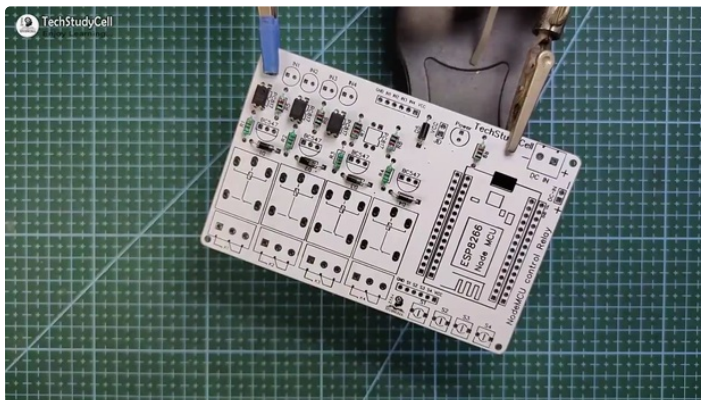
PCBs were well packed and the quality was really good at this affordable price.

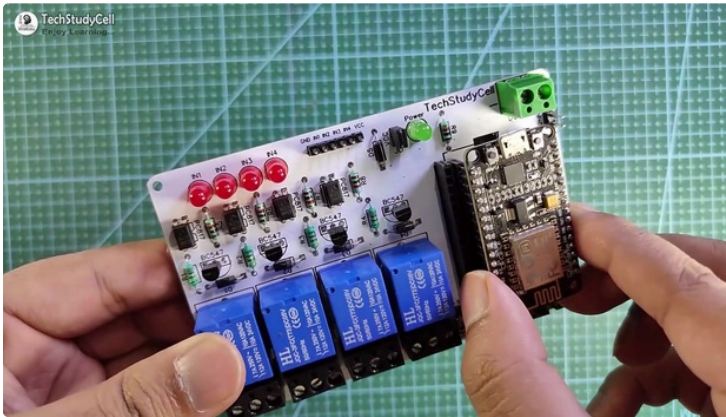


Step 8: Solder All the Components on PCB

After that, I have soldered all the components as per the circuit diagram.

Then connect the NodeMCU board with the PCB.





Step 9: Create Blynk Cloud FREE Account

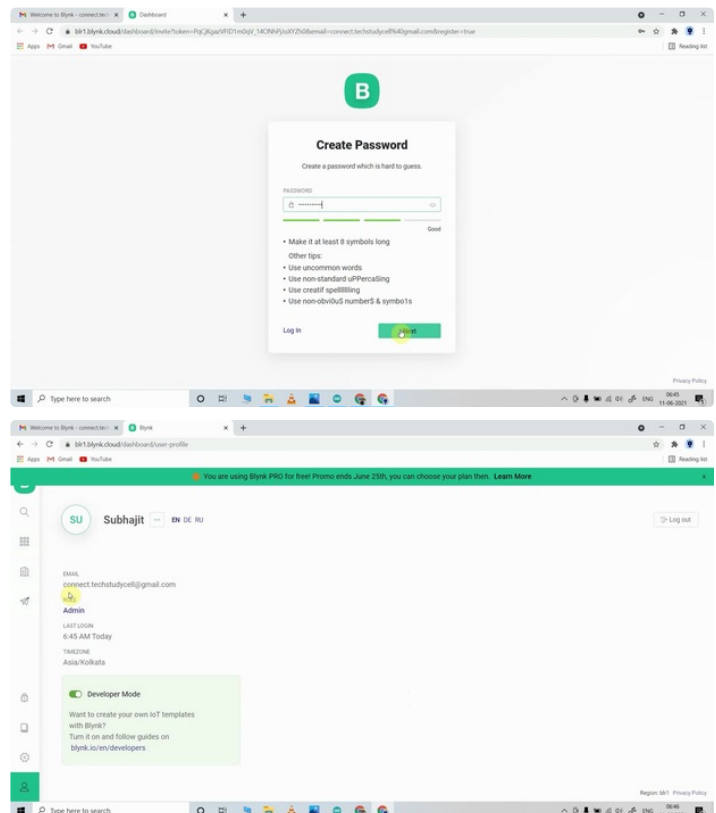
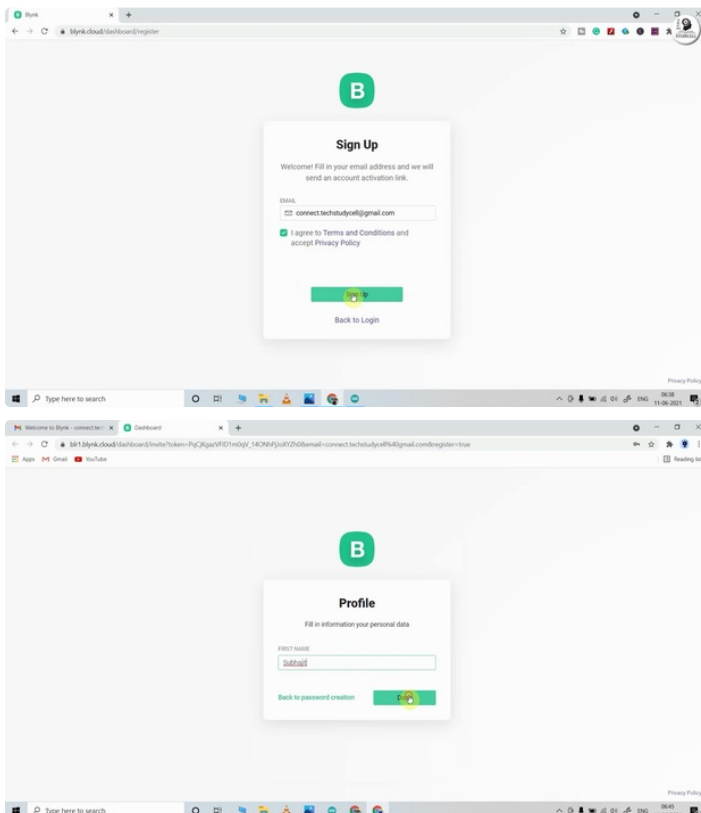
For this smart house project, I have used the **Blynk IoT Cloud Free plan**.

Click on the following link to create a Blynk Cloud account.

<https://blynk.cloud/dashboard/register>

1. Enter email ID, then click on "**Sign Up**". You will receive a verification email.
2. Click on **Create Password** in the email, Then set the **password**, click on **Next**.
3. Enter your **first name**, click on **Done**.

After that Blynk cloud dashboard will open.



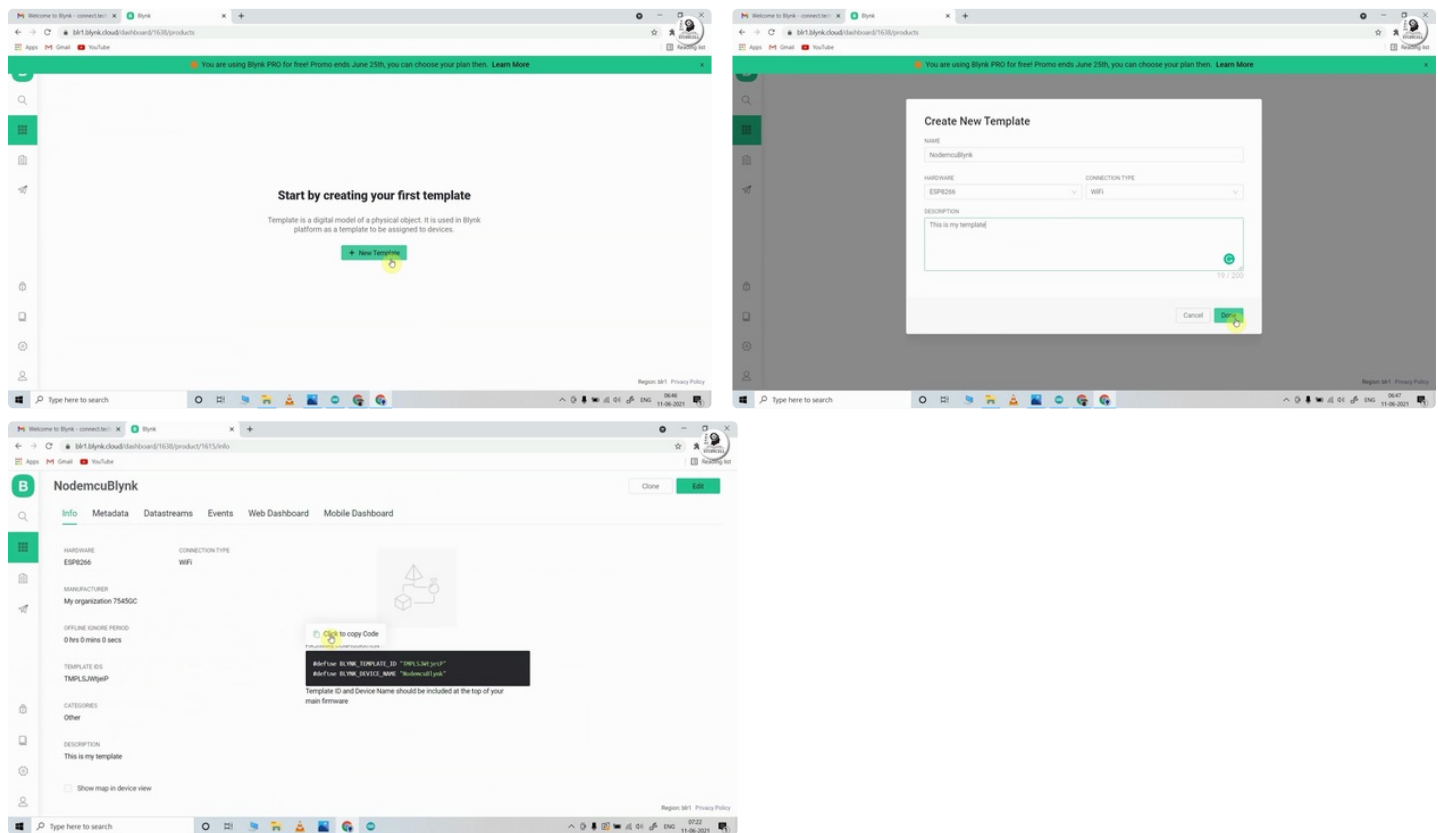
Step 10: Create a New Template in Blynk Cloud

First, you have to create a template in the Blynk cloud.

1. Click on **New Template**.
2. Enter a template **name**, select the hardware as **ESP8266**, and the connection type will **WiFi**.
3. Then click on **DONE**.

You will get the **BLYNK_TEMPLATE_ID** and **BLYNK_DEVICE_NAME** after creating the temple.

The BLYNK_TEMPLATE_ID and BLYNK_DEVICE_NAME will be required while programming the NodeMCU.

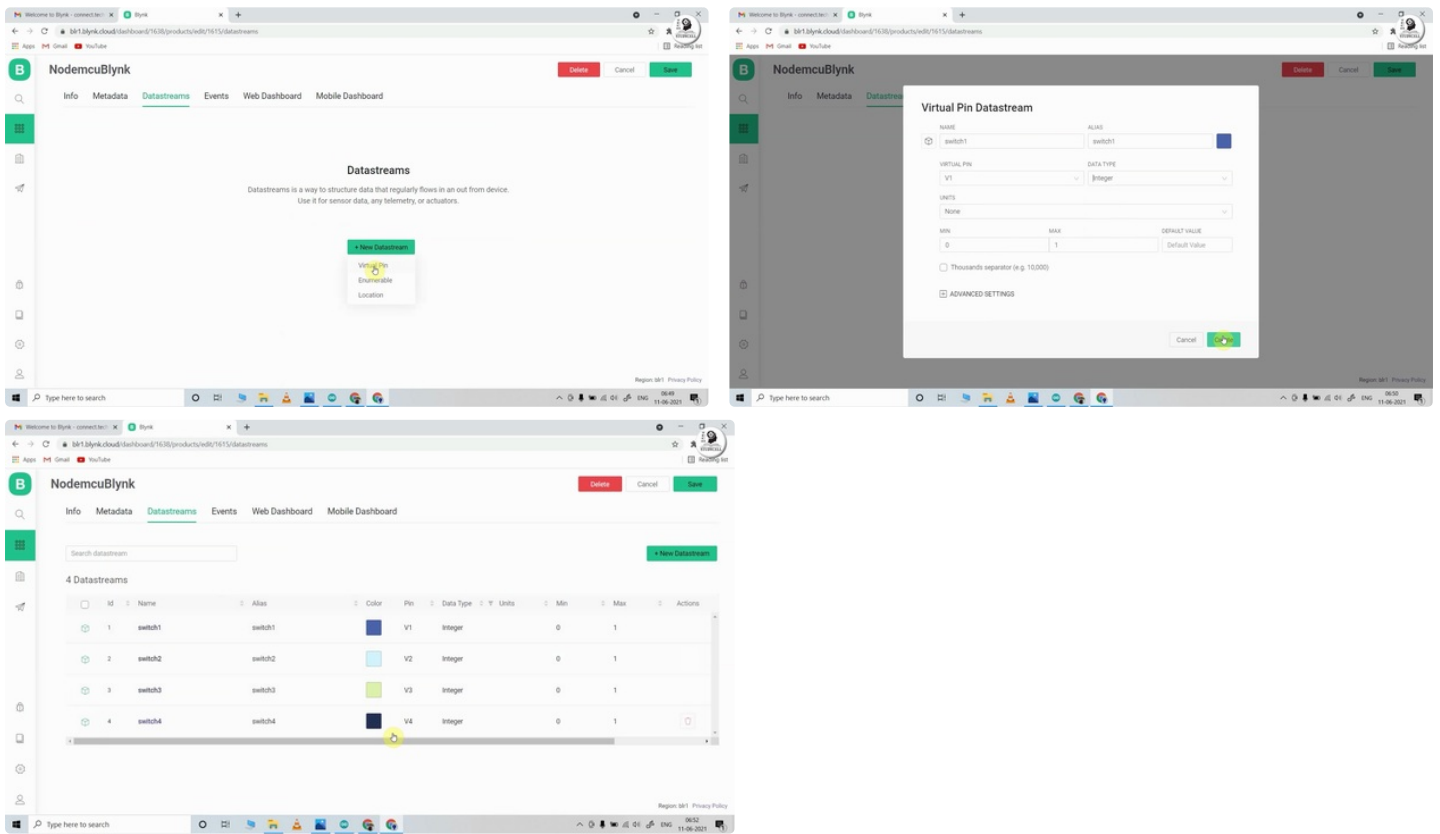


Step 11: Create a Datastream in Blynk Cloud

After that, you have to create Datastreams. Here I will control 4 relays, so I have to create 4 Datastreams.

1. Go to the **Datastreams** tab.
2. Click on **New Datastream** and select **Virtual Pin**.
3. Enter a **name**, select the **virtual pin V1**, and the datatype will be **Integer**.
4. Then click on **Create**.

In a similar way, create 4 datastreams with virtual pin **V1, V2, V3**, and **V4**.

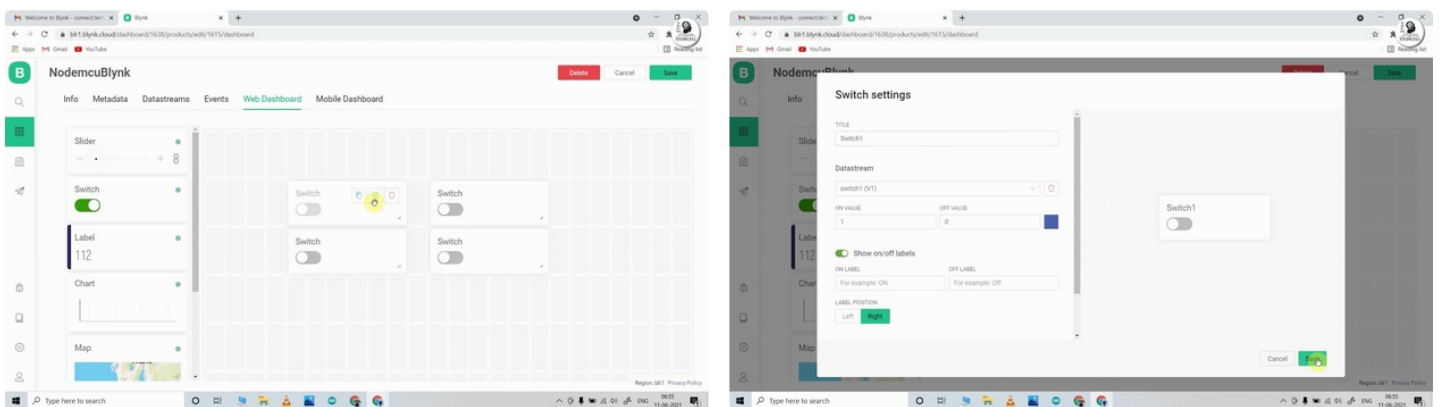


Step 12: Set Up Blynk Cloud Web Dashboard

Now go to the web dashboard tab.

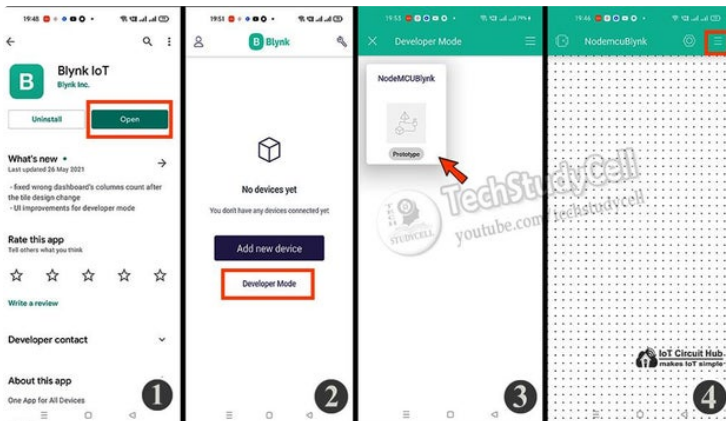
Drag and drop 4 Switch widgets.

Go to the settings of each widget, and select a Datastream.



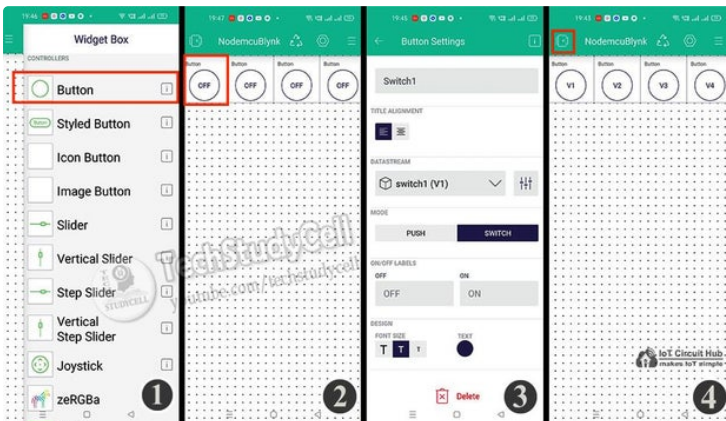
Step 13: Install Blynk IoT App to Configure Mobile Dashboard

1. Install the **Blynk IoT app** from Google Play Store or App Store. Then **log in**.
2. Go to **Developer Mode**.
3. Tap on the **template** that you have already made.
4. Now go to the **Widget box** (on the right) to add widgets.



Step 14: Add Widgets in Blynk IoT App

1. Add **4 Button** widgets from Widget Box.
2. Go to **Button widget settings**.
3. Enter the **name**, select **Datastream**, Mode will be **Switch**. Then exit.
4. After setting all the Buttons tap on **exit**.



Step 15: Program the NodeMCU for This Blynk Project

First, download the code from the following link.

<https://drive.google.com/file/d/1WK6AOzVyS-P6G32mK...>

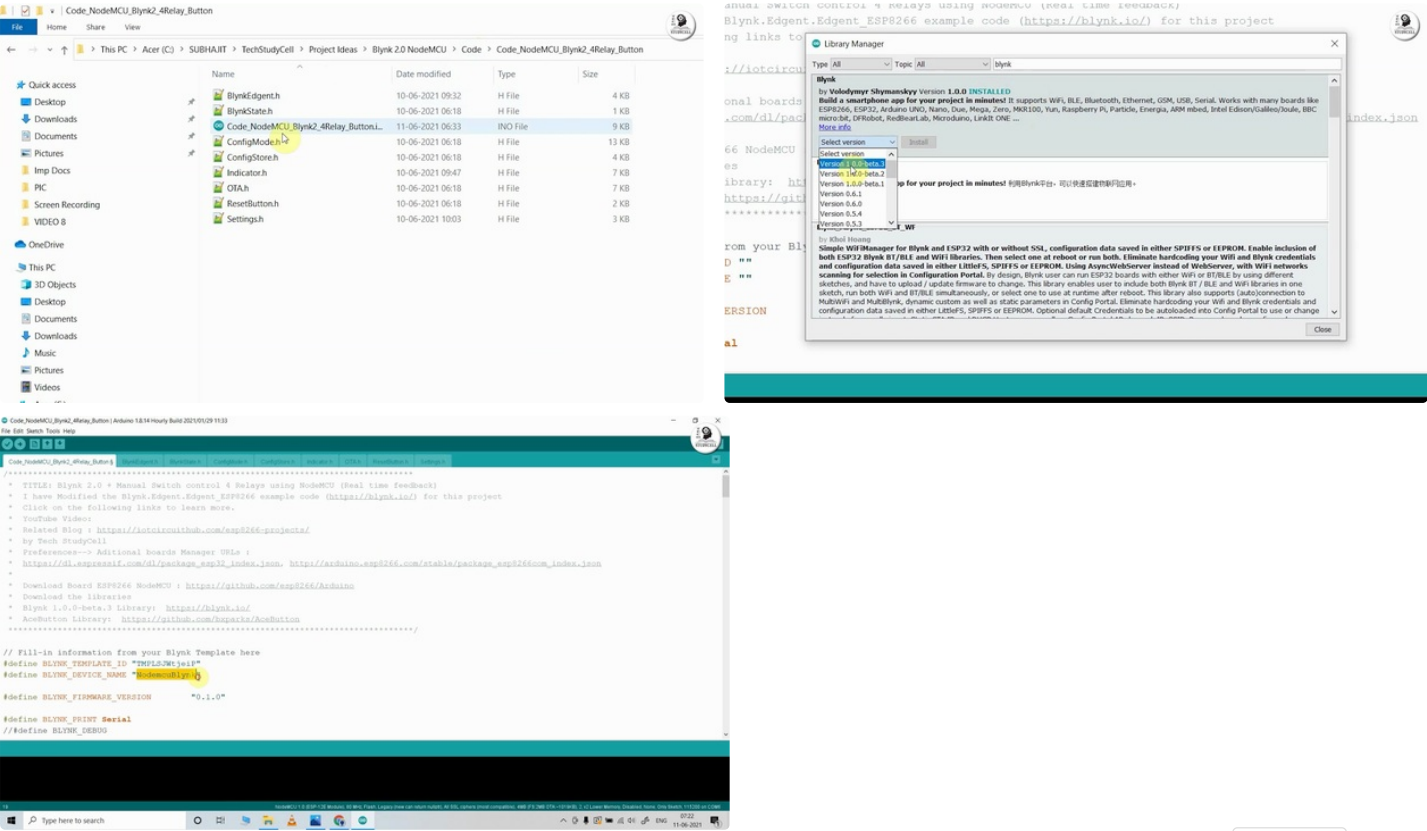
You have to keep all the 9 files in the same folder.

Open the **.ino** file in Arduino IDE.

In the code, you have to update the **BLYNK_TEMPLATE_ID** and **BLYNK_DEVICE_NAME**.

For this project, you have to install the **Blynk 1.0.0 beta.3** & **AceButton** libraries.

Now select the NodeMCU 1.0 board and proper PORT. Then upload the code to NodeMCU Board.

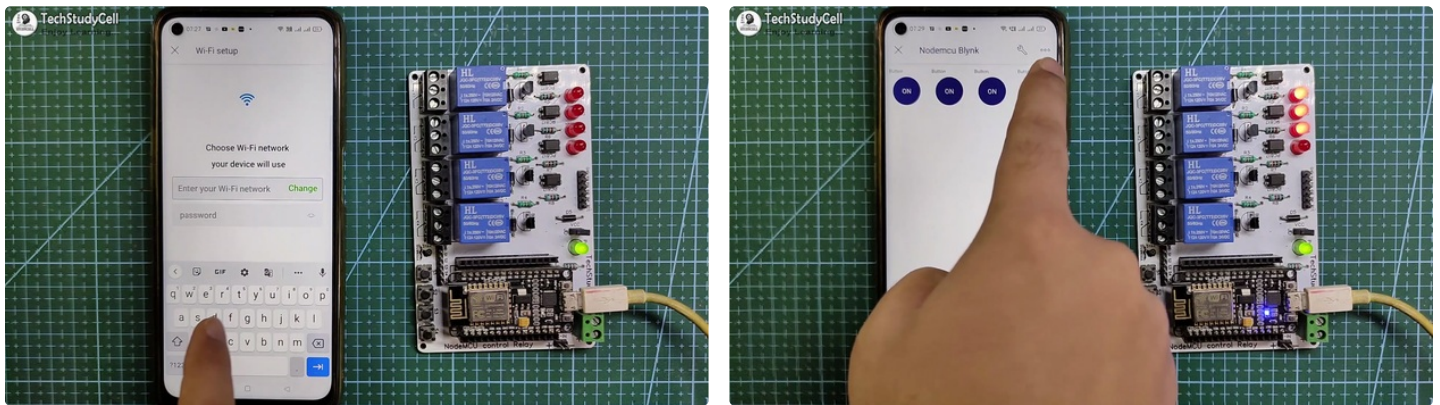


	https://www.instructables.com/ORIG/F1M/KK62/KQF7BLKS/F1MKK62KQF7BLKS.h	Download
	https://www.instructables.com/ORIG/F9F/T2PT/KQF7BLKT/F9FT2PTKQF7BLKT.h	Download
	https://www.instructables.com/ORIG/FEZ/2Y5G/KQF7BLKU/FEZ2Y5GKQF7BLKU.ino	Download
	https://www.instructables.com/ORIG/FUG/6C00/KQF7BLKV/FUG6C00KQF7BLKV.h	Download
	https://www.instructables.com/ORIG/FZE/ZXXA/KQF7BLKW/FZEZXXAKQF7BLKW.h	Download
	https://www.instructables.com/ORIG/FSG/JH01/KQF7BLKX/FSGJH01KQF7BLKX.h	Download
	https://www.instructables.com/ORIG/F8J/CJ8T/KQF7BLKY/F8JCJ8TKQF7BLKY.h	Download
	https://www.instructables.com/ORIG/F3J/FNL2/KQF7BLKZ/F3JFNL2KQF7BLKZ.h	Download

Step 16: Update the WiFi Credentials Through OTA

After programming the NodeMCU, you have to update the WiFi credentials from the Blynk IoT app.

In the tutorial video, I have explained all the steps to update the WiFi credentials to NodeMCU through OTA.



Step 17: Connect the Home Appliances

Connect the **4 home appliances** with the relay module as per the circuit diagram.

Please take proper safety precautions while working with high voltage.

Connect 5-volt DC supply with the PCB. (I have used my old mobile charger 5V 2Amp)

Turn on the **110V/230V** supply and **5V** DC supply.



Step 18: Finally!! the Blynk Smart Home System Is Ready

Now you can control your home appliances in a smart way.

I hope you have liked this new **Blynk home automation project**. I have shared all the required information for this project.

I will really appreciate it if you share your valuable feedbacks. Also if you have any query please write in the comment section.

Thank you & Happy Learning.

