4.3.4 BLYNK SERVER

**Blynk** is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets. **Blynk** is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device. After downloading the **Blynk** app, you can create a project dashboard and arrange buttons, sliders, graphs, and other widgets onto the screen.

Supported Hardware

4.3.1 Blynk App

Blynk application can be found from the following links –

[*1. Android Blynk App*](https://play.google.com/store/apps/details?id=cc.blynk)

[*2. IOS Blynk App*](https://itunes.apple.com/us/app/blynk-control-arduino-raspberry/id808760481?ls=1&mt=8)

*After downloading the app, create an account and log in. (If possible than log in with your real mail id for better connectivity later.)*

You’ll also need to install the **Blynk Arduino Library**, which helps generate the firmware running on your ESP8266. Download the latest release from <https://github.com/blynkkk/blynk-library/releases> , and follow along with the directions there to install the required libraries.

4.3.2 Create a Blynk Project

Click the “Create New Project” in the app to create a new Blynk app. Give it any name.

Blynk works with hundreds of hardware models and connection types. Select the Hardware type. After this, select connection type. In this project we have select WiFi connectivity.

The ***Auth Token*** is very important – you’ll need to stick it into your ESP8266’s firmware. For now, copy it down or use the “E-mail” button to send it to yourself.

4.3. 3 Add Widgets To The Project

Then you’ll be presented with a blank new project. To open the widget box, click in the project window to open.

We are selecting a button to control Led connected with NodeMCU.

1. Click on Button.
2. Give name to Button say led.
3. Under OUTPUT tab- Click pin and select the pin to which led is connected to NodeMCU, here it is digital pin 2, hence select digital and under pin D2. And Click continue.

Under MODE tab- Select whether you want this button as “push button” or “Switch”.

*You have successfully created a GUI for Arduino.*

4.3.4 Upload The Firmware

Now that your Blynk project is set-up, open Arduino and navigate to the ESP8266\_Standalone example in the File > Examples > Blynk > Boards\_WiFi> ESP8266\_Standalone menu.

4.3.5 Stand Alone Programming Code:

*Before uploading, make sure to paste your authorization token into the auth [] variable. Also make sure to load your Wifi network settings into the Blynk.begin(auth, “ssid”, “pass”) function.*

#define BLYNK\_PRINT **Serial**

#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);

}

void loop()

{

**Blynk**.run();

}

4.3.6. Execution

After the app has uploaded, open the serial monitor, setting the baud rate to 9600. Wait for the “Ready” message.

Then click the “Run” button in the top right corner of the Blynk app. Press the button and watch the LED

Then add more widgets to the project. They should immediately work on the ESP8266 without uploading any new firmware.

4.3.7 Output

