A PROJECT REPORT ON

## HOME AUTOMATION USING NODE MCU ,GOOGLE ASSISTANT AND BLYNK APP



DEPARTMENT OF COMPUTER SCIENCE AND

ENGINEERING

**SUBJECT : ANALOG & DIGITAL ELECTRONICS .**

**Report-2**

**SUBMITTED BY : SUBMITTED T0:**

ROHIT KUMAR DAS : 1MV18CS081 Mrs. REKHA.B.N

SAURAV KUMAR : 1MV18CS091 (Professor)

SHAIKH ABDUL THOUHID: 1MV18CS093 (Dept. of ECE)

SHAURYA AGARWAL : 1MV18CS096

SRESTH SINGH : 1MV18CS102

## ACKNOWLEDGEMENT

I express my profound gratitude to

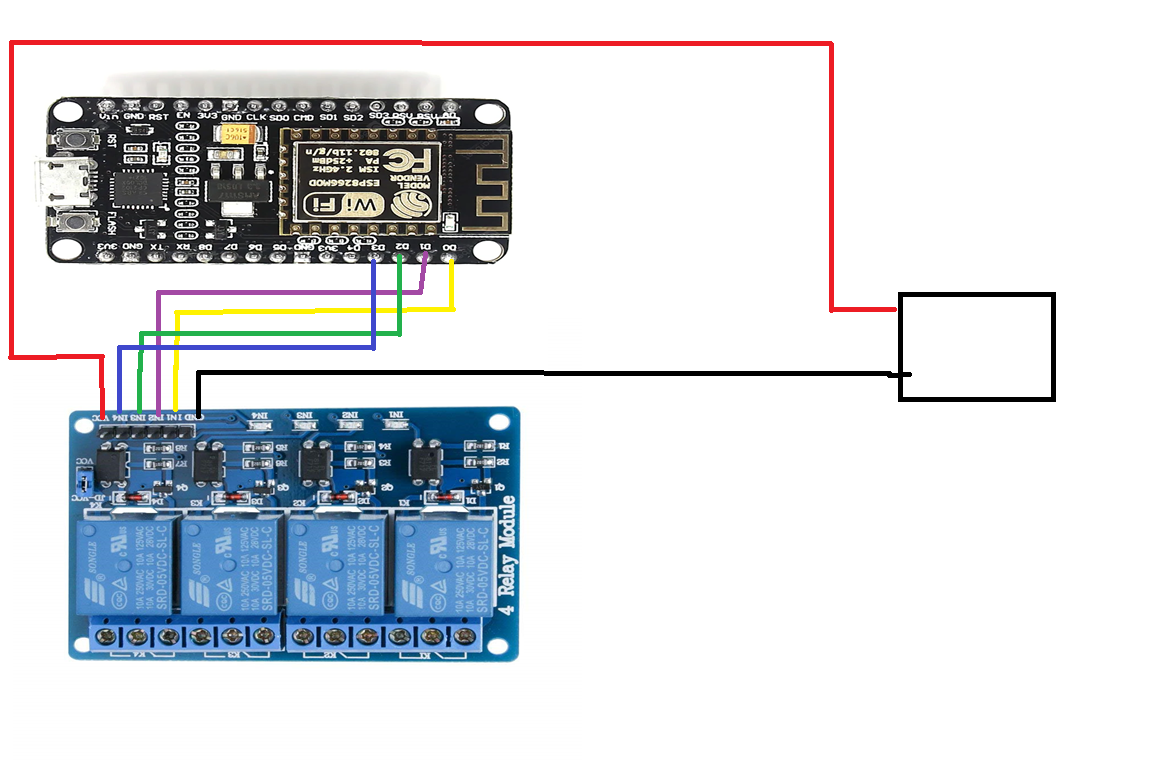
Mrs. Rekha.B.N (Professor), Department of Electronics & Communication Engineering for the valuable help and guidance in the preparation of this project on Home Automation using node mcu, google assistant and blynk app.

I would like to extend my sincere thanks to Professor and all other staff members of Electronics & Communication Engineering. We are also thankful to everyone who all supported us ,for that we have completed our report effectively and moreover on time. I sincerely thank everyone for their valuable suggestions, motivation and encouragement.

**WORK PROGRESS**

1. *Project layout prepared with actual diagrams.*
2. *All the components Required are procured.*
3. *All the components are arranged As per the schematics.*
4. *Coding is done for the hardware.*
5. *Third party software are configured according to the needs.*
6. *Component testing is done.*
7. *System testing is pending for the whole system and it is to be completed soon.*

**Circuit Diagram**

****

5V 2A

Power Supply

**Components and Cost**

|  |  |  |
| --- | --- | --- |
| Items | Quantity | Cost(in Rs.) |
| Node MCU | 1 | 251 |
| Relay Module 4 Channel | 1 | 285 |
| Power Supply |  |  |
| Wires |  |  |
| Bread Board |  | 90 |

**Coding for Node MCU**

*/\**

*\* Coding is done in Arduino IDE*

*\* Node MCU library are added*

*\* Virtual pins are used for data transfer over Wi-Fi*

*\**

*\*/*

**#define BLYNK\_PRINT Serial**

**#include <ESP8266WiFi.h>**

**#include <BlynkSimpleEsp8266.h>**

**int a,b,c,d;**

*//GPIO Pin outs for Node MCU.*

**int sw1=16;**

**int sw2=5;**

**int sw3=4;**

**int sw4=0;**

//functions to accept data from Blynk App and Google assistant.

**BLYNK\_WRITE(V1){a=param.asInt();}**

**BLYNK\_WRITE(V2){b=param.asInt();}**

**BLYNK\_WRITE(V3){c=param.asInt();}**

**BLYNK\_WRITE(V4){d=param.asInt();}**

*//Assignment of Indicators to virtual pin*

**WidgetLED ST1(V20);**

**WidgetLED ST2(V21);**

**WidgetLED ST3(V22);**

**WidgetLED ST4(V23);**

*// You should get Auth Token in the Blynk App.*

*// Go to the Project Settings (nut icon).*

**char auth[] = "jLWP2HTNC4kcQvNw037DJURV\_cvWcJSl";**

**char ssid[] = "Redmi";**

**char pass[] = "Rohit321";**

**void setup()**

**{**

**Serial.begin(9600);**

**pinMode(sw1,OUTPUT);**

**pinMode(sw2,OUTPUT);**

**pinMode(sw3,OUTPUT);**

**pinMode(sw4,OUTPUT);**

**Blynk.begin(auth, ssid, pass);**

**}**

**void loop()**

**{**

**Blynk.run();**

**BLYNK\_CONNECTED();**

**BLYNK\_WRITE(V1);**

**BLYNK\_WRITE(V2);**

**BLYNK\_WRITE(V3);**

**BLYNK\_WRITE(V4);**

**digitalWrite(sw1,a);**

**if(a>0)**

**ST1.off();**

**else**

**ST1.on();**

**digitalWrite(sw2,b);**

**if(b>0)**

**ST2.off();**

**else**

**ST2.on();**

**digitalWrite(sw3,c);**

**if(c>0)**

**ST3.off();**

**else**

**ST3.on();**

**digitalWrite(sw4,d);**

**if(d>0)**

**ST4.off();**

**else**

**ST4.on();**

**}**

**Third party Application Configuration**

1. Blynk APP

To configure BLYNK go to their official website.

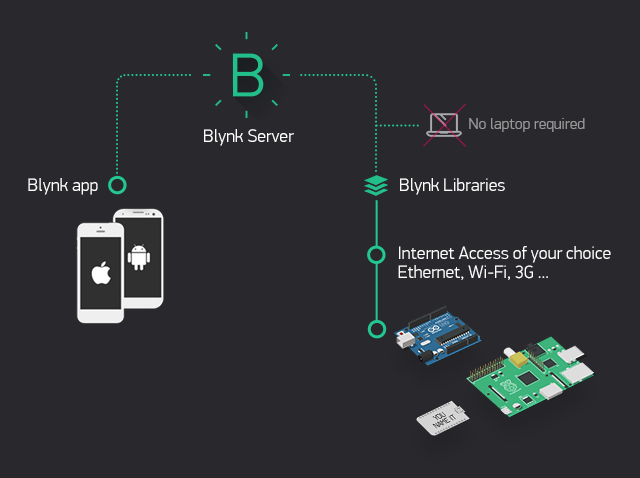
<https://blynk.io/>

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

There are three major components in the platform:

* **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
* **Blynk Server** - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your [private Blynk server](https://docs.blynk.cc/#blynk-server) locally. It’s open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
* **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

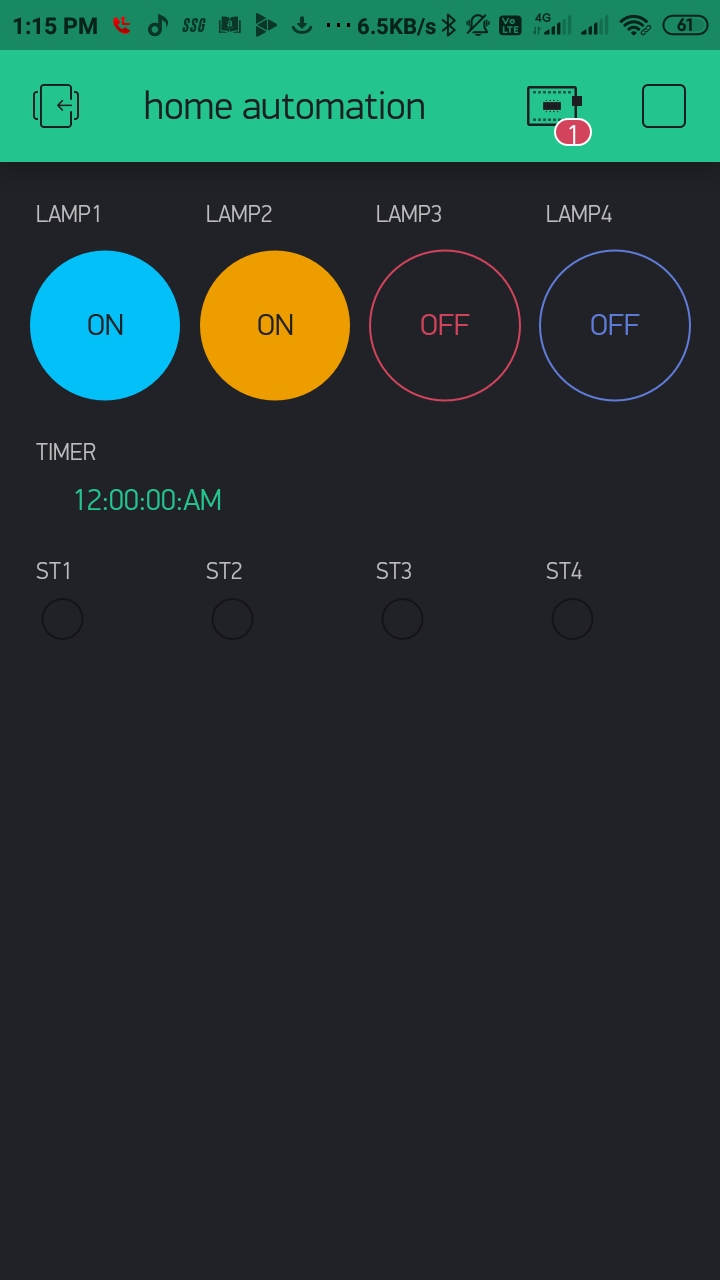
Now imagine: every time you press a Button in the Blynk app, the message travels to  the Blynk Cloud, where it magically finds its way to your hardware. It works the same in the opposite direction and everything happens in a blynk of an eye.



## Features

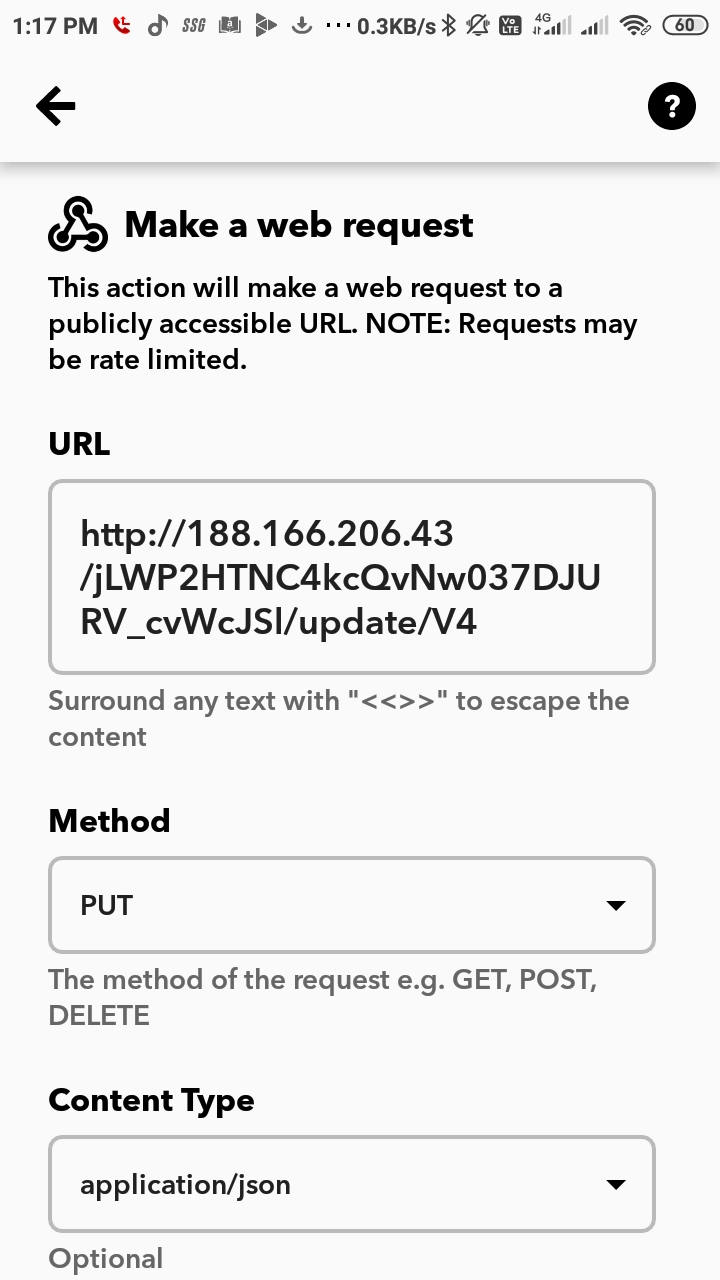
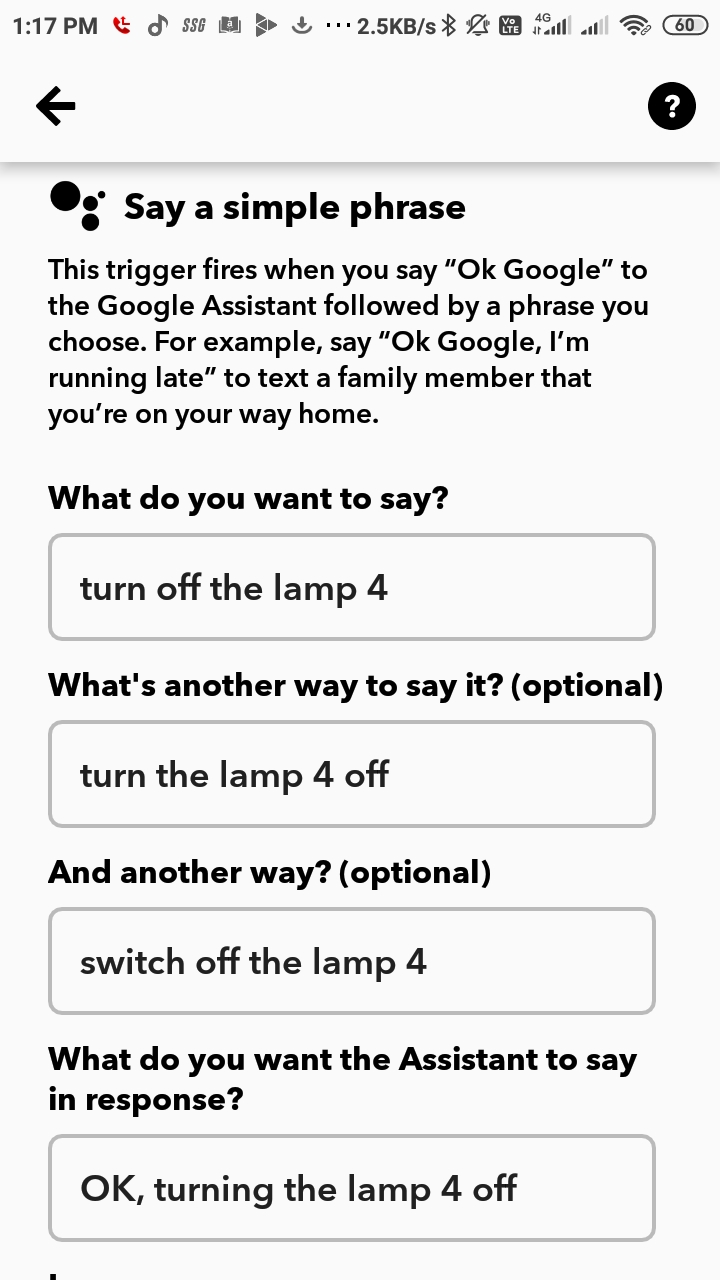
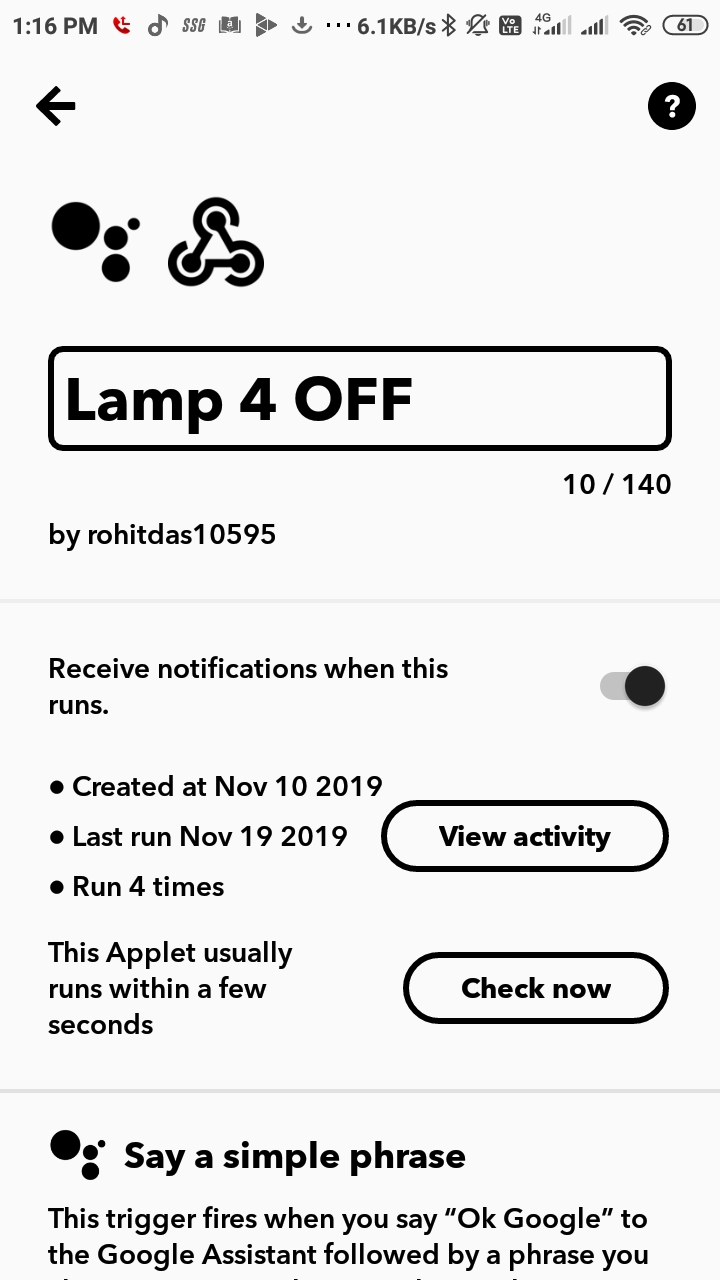
* Similar API & UI for all supported hardware & devices
* Connection to the cloud using:
  + WiFi
  + Bluetooth and BLE
  + Ethernet
  + USB (Serial)
  + GSM
  + …
* Set of easy-to-use Widgets
* Direct pin manipulation with no code writing
* Easy to integrate and add new functionality using virtual pins
* History data monitoring via SuperChart widget
* Device-to-Device communication using Bridge Widget
* Sending emails, tweets, push notifications, etc.
* … new features are constantly added!

You can find [example sketches](https://github.com/blynkkk/blynk-library/tree/master/examples) covering basic Blynk Features. They are included in the library. All the sketches are designed to be easily combined with each other.



IFTTT SERVER Configuration

1. Create a free account.
2. Browse the **IFTTT** website or app to find an Applet that interests you.
3. Click into the Applet and turn it on.
4. Connect the services that are involved in the Applet — this is only so we can use them to run Applets on your behalf. ...
5. Find more Applets, and repeat!



REFERENCES :

1. <https://codeometry.in/home-automation-using-nodemcu-and-google-assistant/>
2. Youtube
3. Hackster.io
4. Toptechboy.com