

IOT Base Smart Dustbin

Aim:

Learn about the Nodemcu (esp8266), Ultrasonic sensor, IR sensor etc... and with the help of these components how we can make smart dustbin and operate it from anywhere by using blynk IOT app.



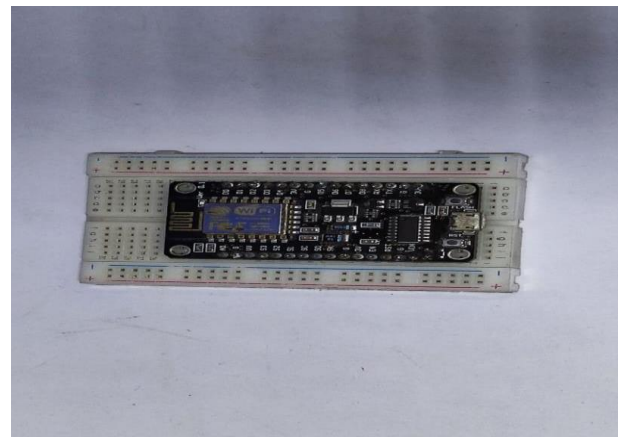
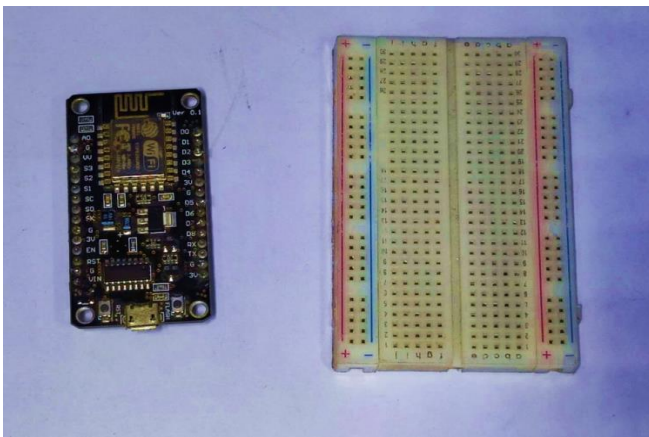
Components Required:

SI no.	Components	Quantity
1	Nodemcu (esp8266)	1
2	Ultrasonic sensor	1
3	IR sensor	1
4	Servo motor	1
5	Breadboard	1
6	Battery (12v)	1
7	Jumper wire	few



Hardware Assembling:

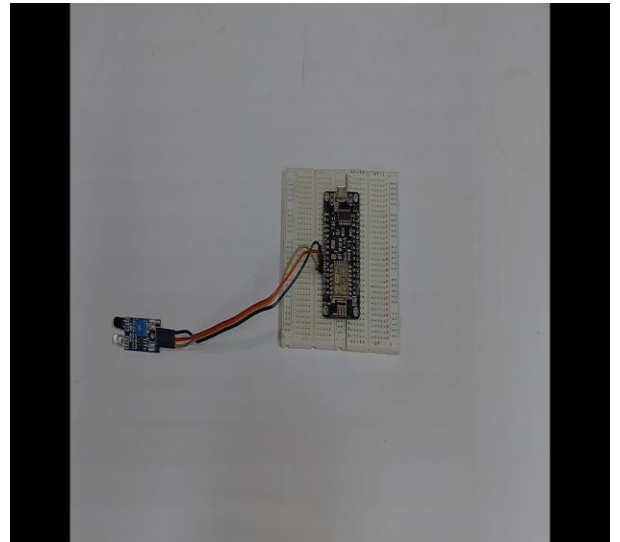
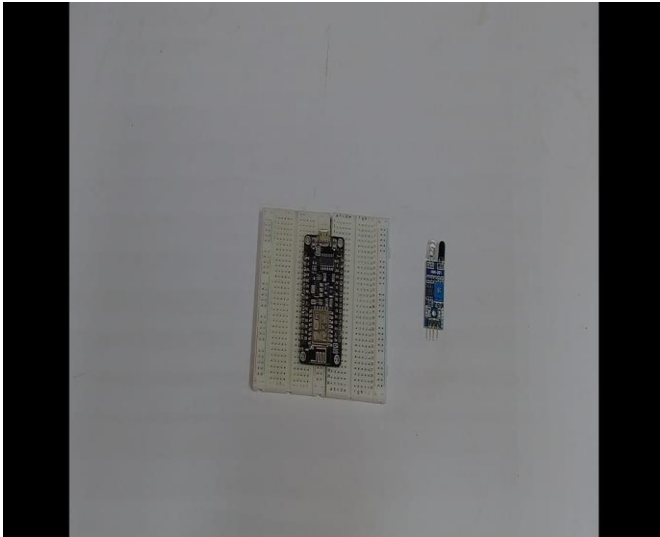
STEP 1- Take the Nodemcu board and put on the breadboard.



STEP 2- Take the IR Sensor and connect with Nodemcu.

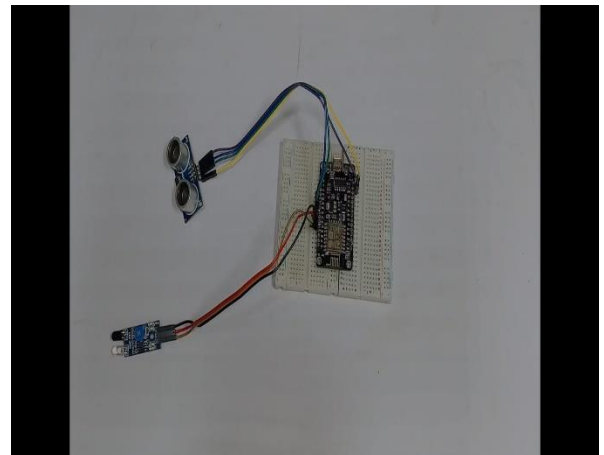
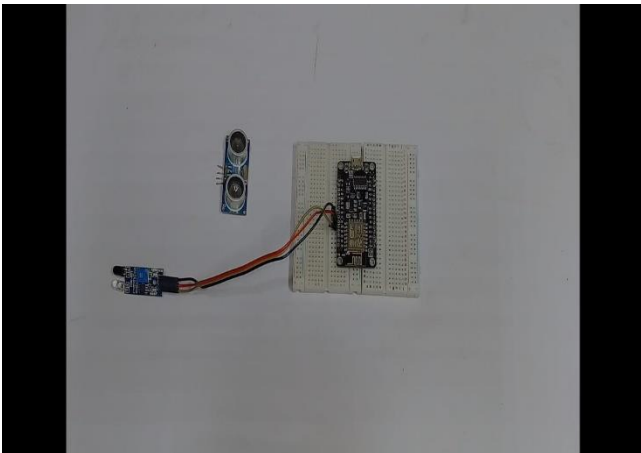


OUT <————> **D4**
GND <————> **GND**
VCC <————> **3V, VIN**



STEP 3- Take the Ultrasonic Sensor and connect with Nodemcu.

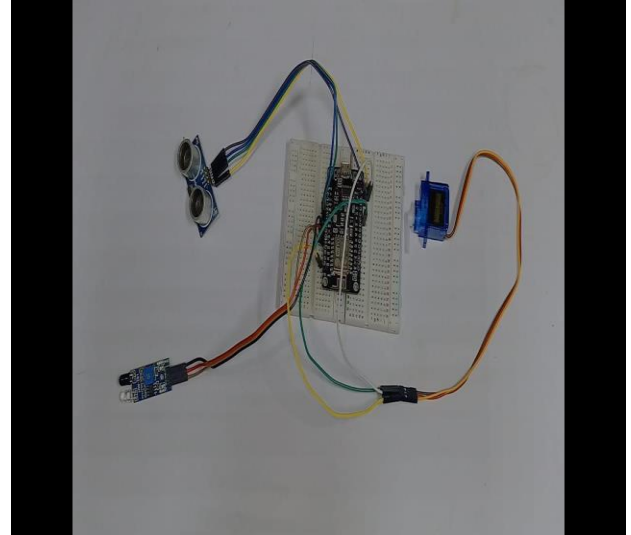
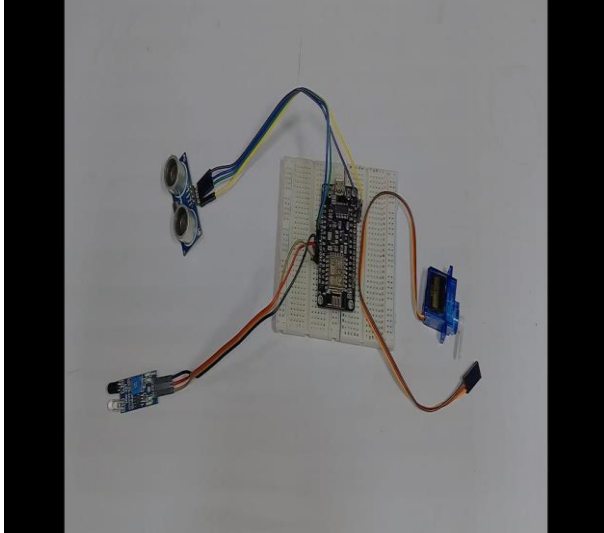
TRIG <————> **D7**
ECHO <————> **D8**
GND <————> **GND**
VCC <————> **VIN**



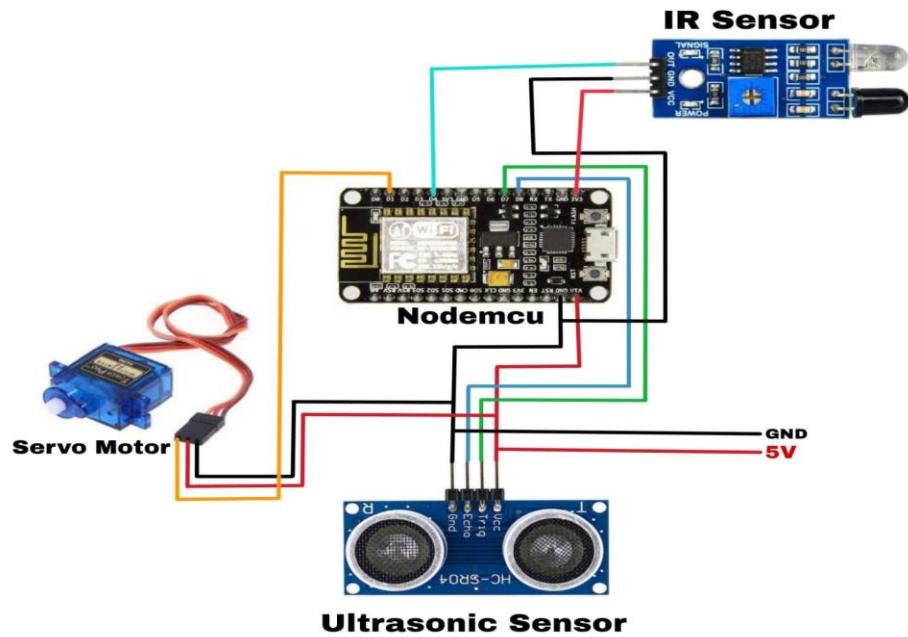
STEP 4- Take the Servo Motor and connect with Nodemcu.



CTRL <-----> **D1**
GND <-----> **GND**
VCC <-----> **VIN**



STEP 5- This is the circuit diagram of the project.



CODE OF THE PROJECT:

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>

char auth[] = "-----";//Enter your Auth token
char ssid[] = "-----";//Enter your WIFI name
char pass[] = "-----";//Enter your WIFI password

BlynkTimer timer;

#define trig D7
#define echo D8

int value; // variable to store value read on digital pin

Servo servo;

void setup() {
  Serial.begin(9600);
  pinMode(2, INPUT); // initialize digital pin 2 as an input
  servo.attach(5); //D1
  servo.write(0);
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  delay(2000);
  Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
  timer.setInterval(100L, ultrasonic);
}
```



```
void ultrasonic() {  
  
  digitalWrite(trig, LOW);  
  
  delayMicroseconds(4);  
  
  digitalWrite(trig, HIGH);  
  
  delayMicroseconds(10);  
  
  digitalWrite(trig, LOW);  
  
  long t = pulseIn(echo, HIGH);  
  
  long cm = t / 29 / 2;  
  
  Blynk.virtualWrite(V0, cm);  
  
}  
  
void loop()  
{  
  value = digitalRead(2); // read value on pin 2  
  
  if(value==1) // if value is 1  
  {  
    servo.write(90);  
  
    delay(1000);  
  }  
}
```



```
if(value==0)
```

```
{
```

```
  servo.write(0);
```

```
  delay(1000);
```

```
}
```

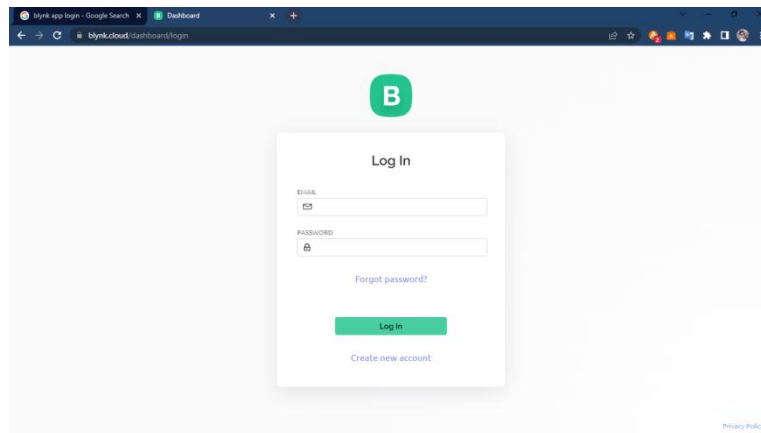
```
Blynk.run();//Run the Blynk library
```

```
timer.run();//Run the Blynk timer
```

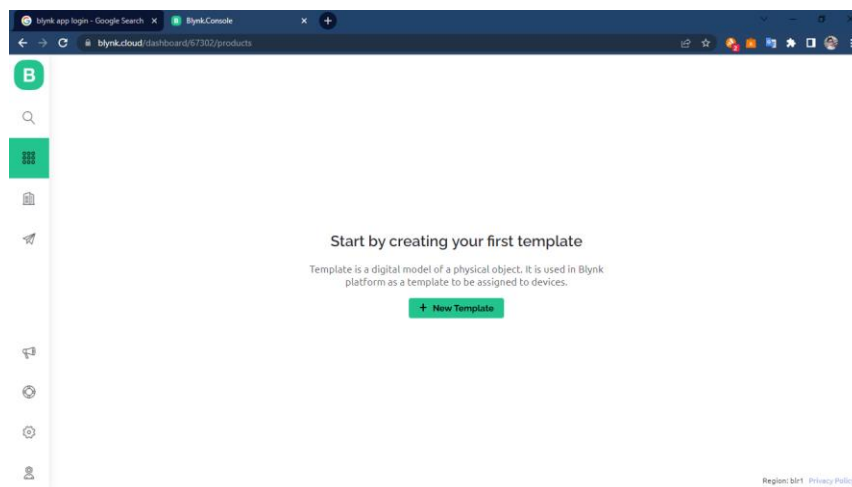
```
}
```

Setup the blynk:

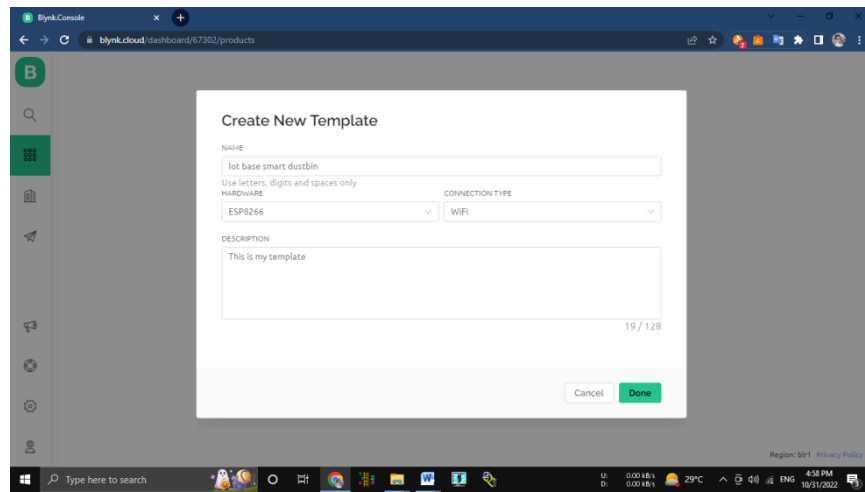
STEP 1– Open the blynk in the chrome and login the blynk page.



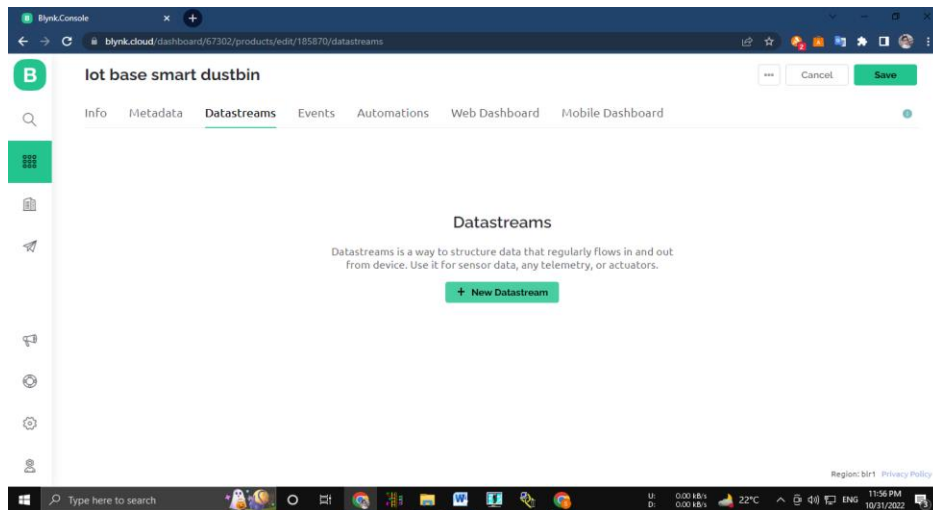
STEP 2 - After that click the new template.



STEP 3– Give the project name and select the hardware esp8266 board.



STEP 4 – select the datastreams and go to the new datastreams.



STEP 5- Give the name and select pin V0 and hit the option create.



Virtual Pin Datastream

NAME: dustbin level ALIAS: dustbin level

PIN: V0 DATA TYPE: Integer

UNITS: None

MIN: 0 MAX: 20 DEFAULT VALUE: 0

ADVANCED SETTINGS

Cancel Create

STEP 6-Then show this type of window.

lot base smart dustbin

Info Metadata **Datastreams** Events Automations Web Dashboard Mobile Dashboard

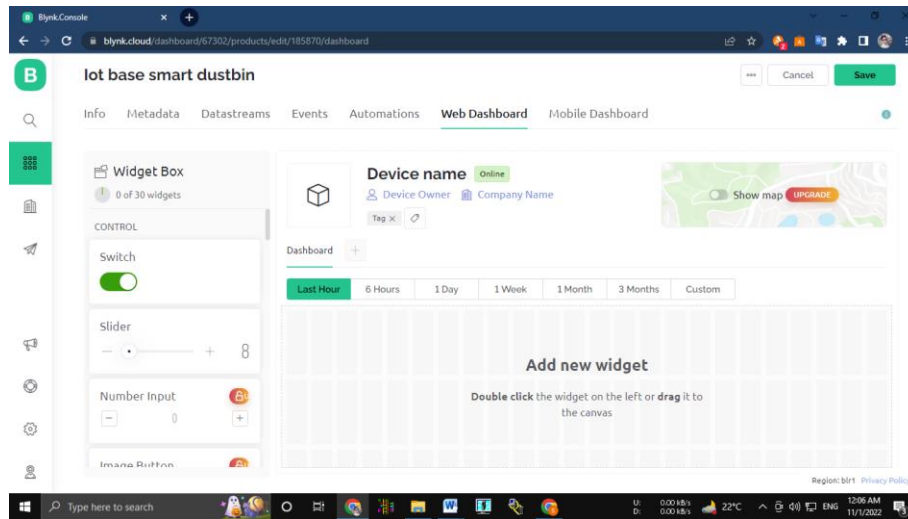
Search datastream + New Datastream

1 Datastream

Id	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min	Actions
1	dustbin level	dustbin level		V0	Integer		false	0	



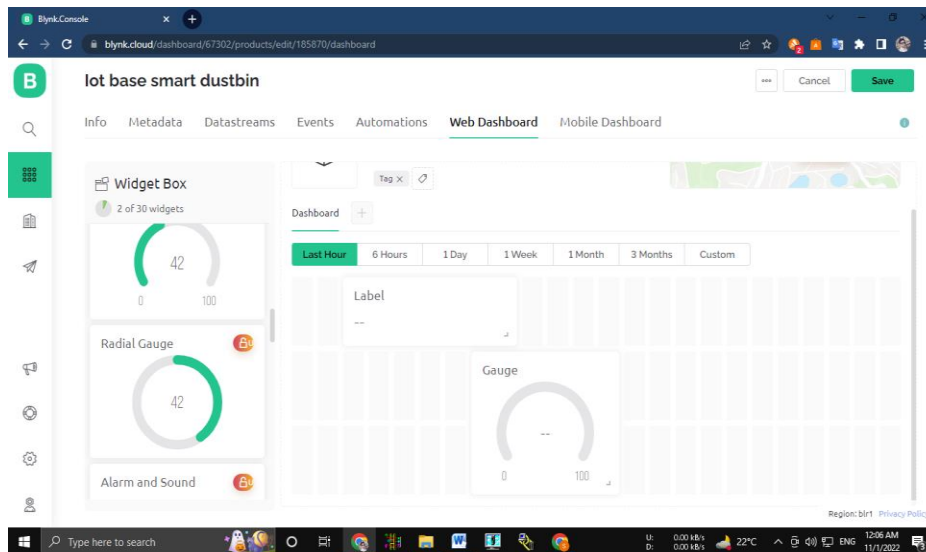
STEP 7– Select the web dashboard option.



The screenshot shows the Blynk Console interface for configuring a 'lot base smart dustbin' dashboard. The 'Web Dashboard' tab is selected. On the left, a 'Widget Box' contains several control widgets: a Switch (turned on), a Slider, a Number Input (set to 0), and an Alarm Button. The main dashboard area shows a 'Device name' field with 'Online' status, a 'Show map' button with an 'UPGRADE' label, and a time filter set to 'Last Hour'. A large grid in the center is labeled 'Add new widget' with instructions: 'Double click the widget on the left or drag it to the canvas'. The bottom of the screen shows a Windows taskbar with the date 11/1/2022 and time 12:06 AM.



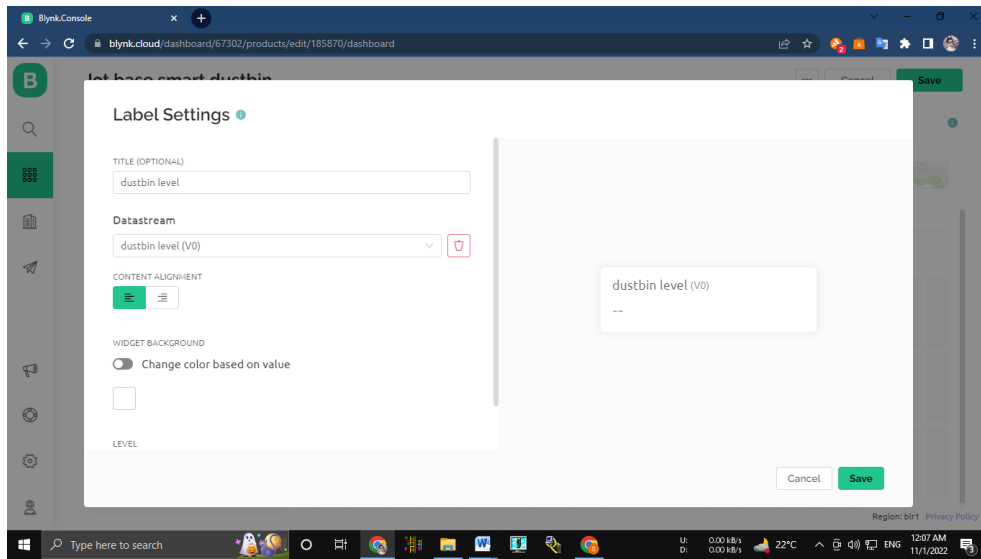
STEP 8 – Drag the label and Gauge option.



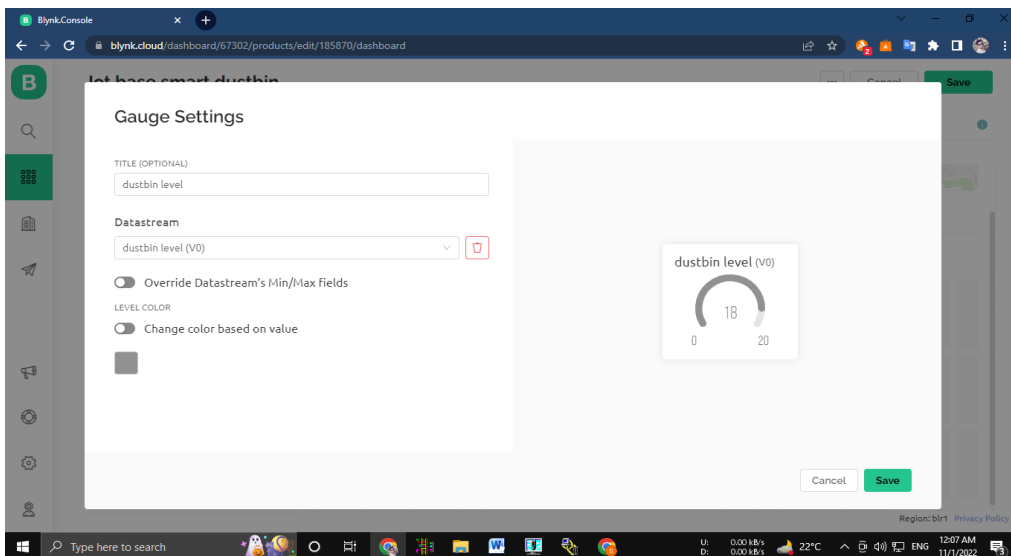
This screenshot shows the same Blynk Console interface as in Step 7, but with two additional widgets added to the dashboard. The 'Widget Box' now contains four widgets: the Switch, Slider, Number Input, and Alarm Button, plus a 'Radial Gauge' widget. The main dashboard area now has a 'Label' widget and a 'Gauge' widget. The 'Gauge' widget is a radial gauge showing a value of 42. The time filter remains 'Last Hour'. The Windows taskbar at the bottom shows the same date and time as in Step 7.



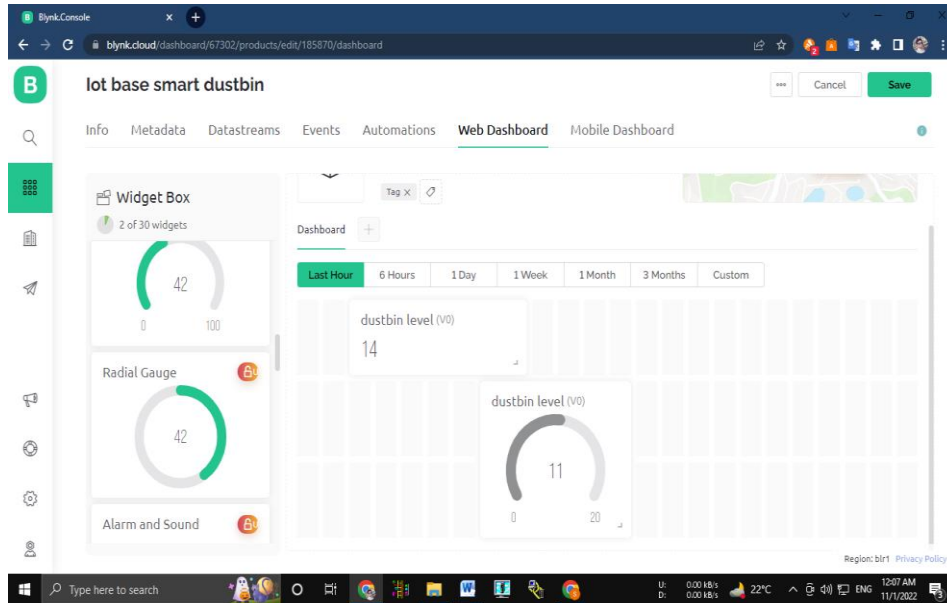
STEP 9 – Go to setting option of label select datastreams V0.



STEP 10– Go to the setting option of gauge and select the datastream V0.



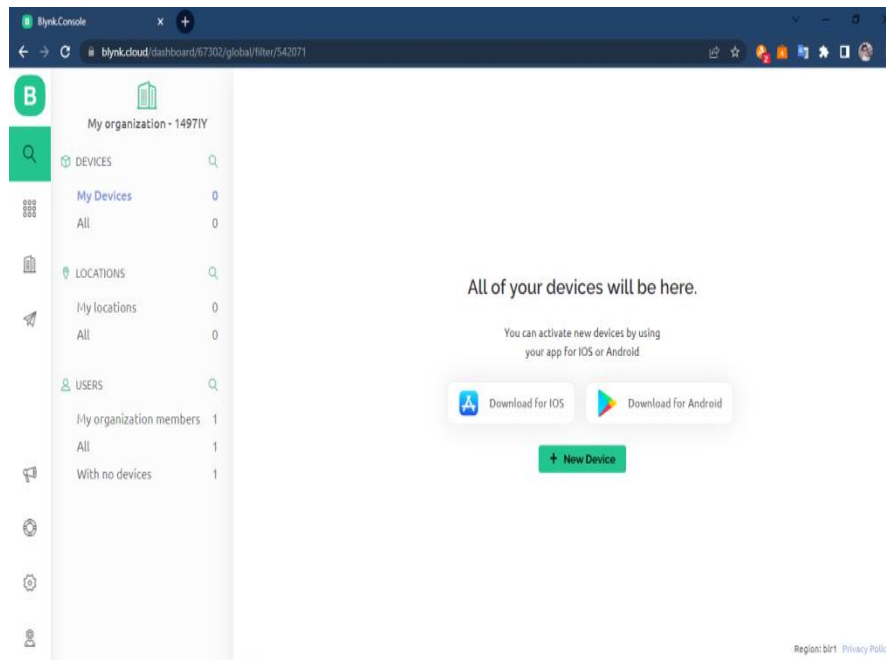
STEP 11 - Click the save button.



The screenshot shows the Blynk Console interface for a device named 'lot base smart dustbin'. The top navigation bar includes 'Info', 'Metadata', 'Datastreams', 'Events', 'Automations', 'Web Dashboard', and 'Mobile Dashboard'. The 'Web Dashboard' is active, displaying a 'Widget Box' with two widgets: a radial gauge showing a value of 42 and an 'Alarm and Sound' icon. Below this is a 'Dashboard' section with a bar chart and a radial gauge showing a value of 11. A 'Save' button is located in the top right corner of the dashboard area.



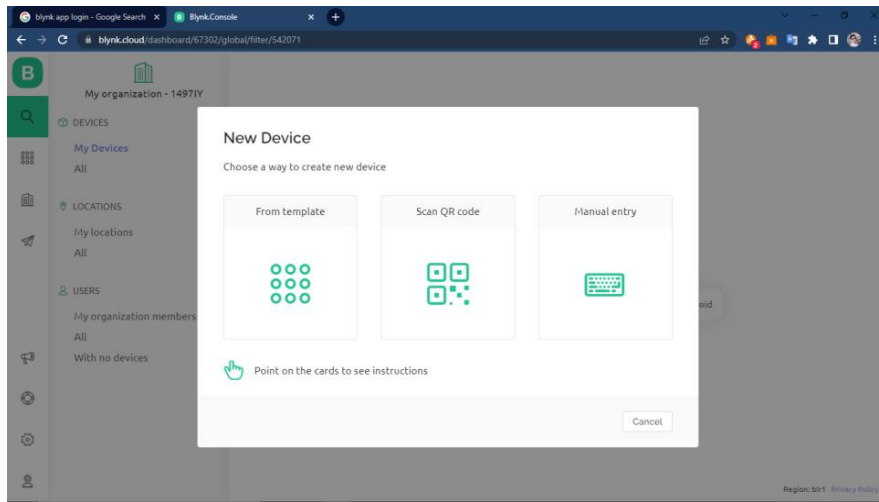
STEP 12 – After that click the search point. And select the new device option.



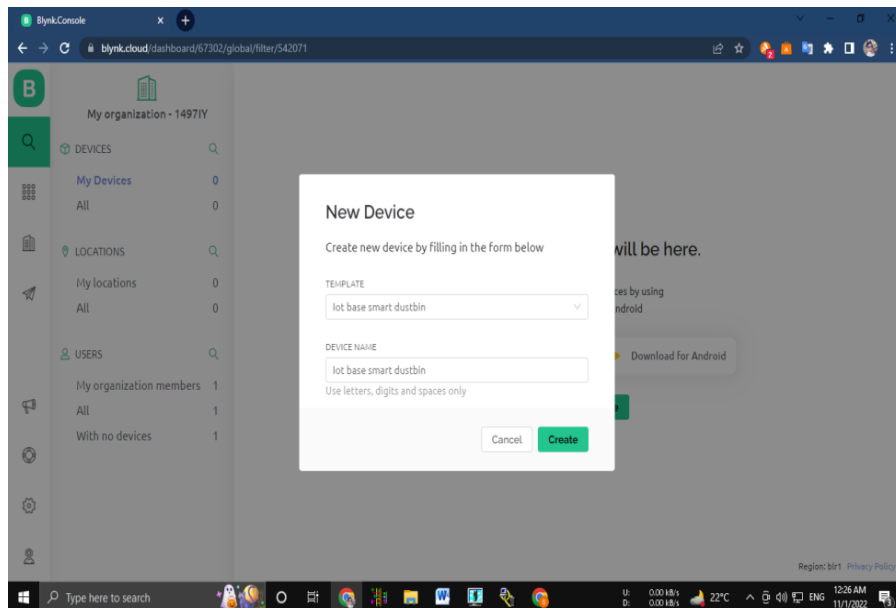
The screenshot shows the 'My organization' page in the Blynk Console. The sidebar on the left lists 'DEVICES', 'LOCATIONS', and 'USERS' with their respective counts. The main content area features a message: 'All of your devices will be here.' Below this message are two buttons: 'Download for IOS' and 'Download for Android'. At the bottom of the main content area is a green button labeled '+ New Device'.



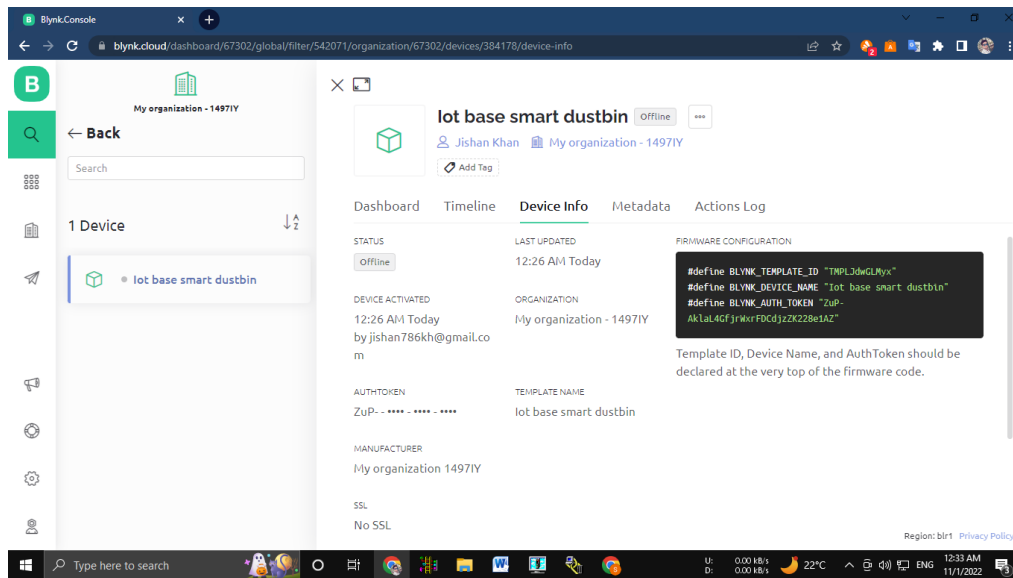
STEP 13 – Select the from template option.



STEP 14 - Give the template name and device name then click create.



STEP 15 – Go to device info option and copy the FIRMWARE CONFIGURATION. This configuration is used in Arduino code.



The screenshot shows the Blynk Console interface for a device named "lot base smart dustbin". The "Device Info" tab is selected, displaying the following details:

- STATUS: Offline
- LAST UPDATED: 12:26 AM Today
- DEVELOPER: Jishan Khan
- ORGANIZATION: My organization - 1497IY
- DEVICE ACTIVATED: 12:26 AM Today by jishan786kh@gmail.com
- AUTH TOKEN: ZuP-... ..
- TEMPLATE NAME: lot base smart dustbin
- MANUFACTURER: My organization 1497IY
- SSL: No SSL

The FIRMWARE CONFIGURATION section contains the following code:

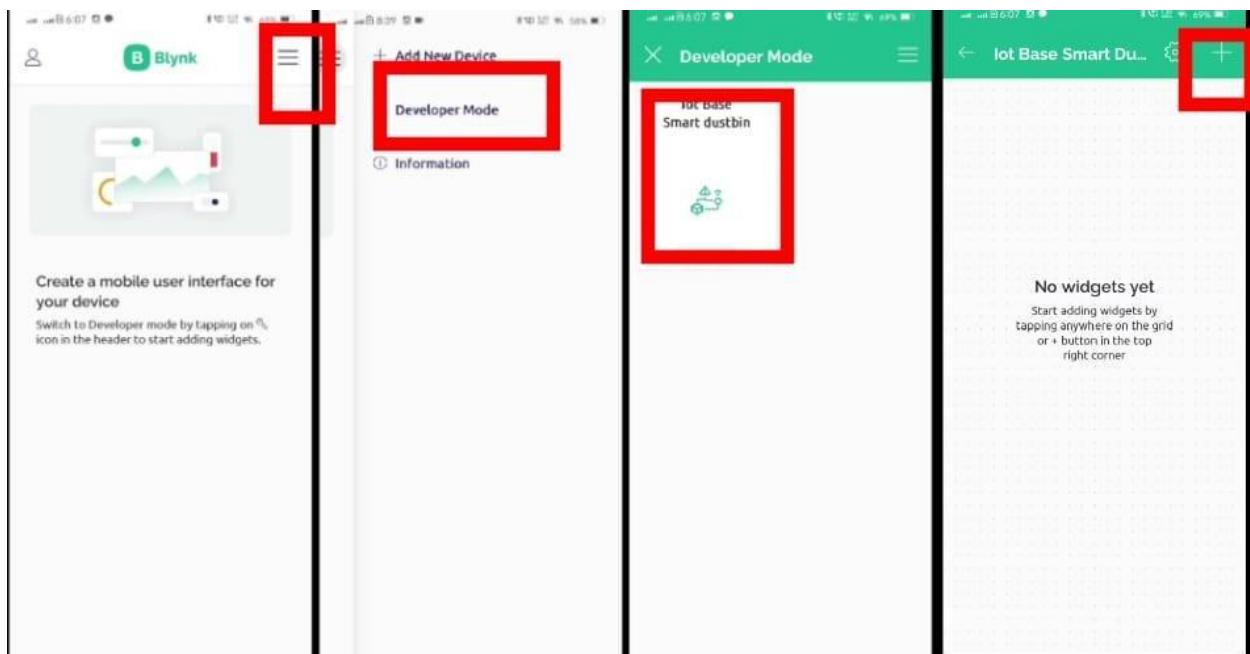
```
#define BLYNK_TEMPLATE_ID "TMPL3dwGLMyx"  
#define BLYNK_DEVICE_NAME "lot base smart dustbin"  
#define BLYNK_AUTH_TOKEN "ZuP-... .."  
#define BLYNK_AUTH_TOKEN "Ak1aL4GfJrWxF0C4jz2K228eIAZ"
```

Below the code, it states: "Template ID, Device Name, and AuthToken should be declared at the very top of the firmware code."



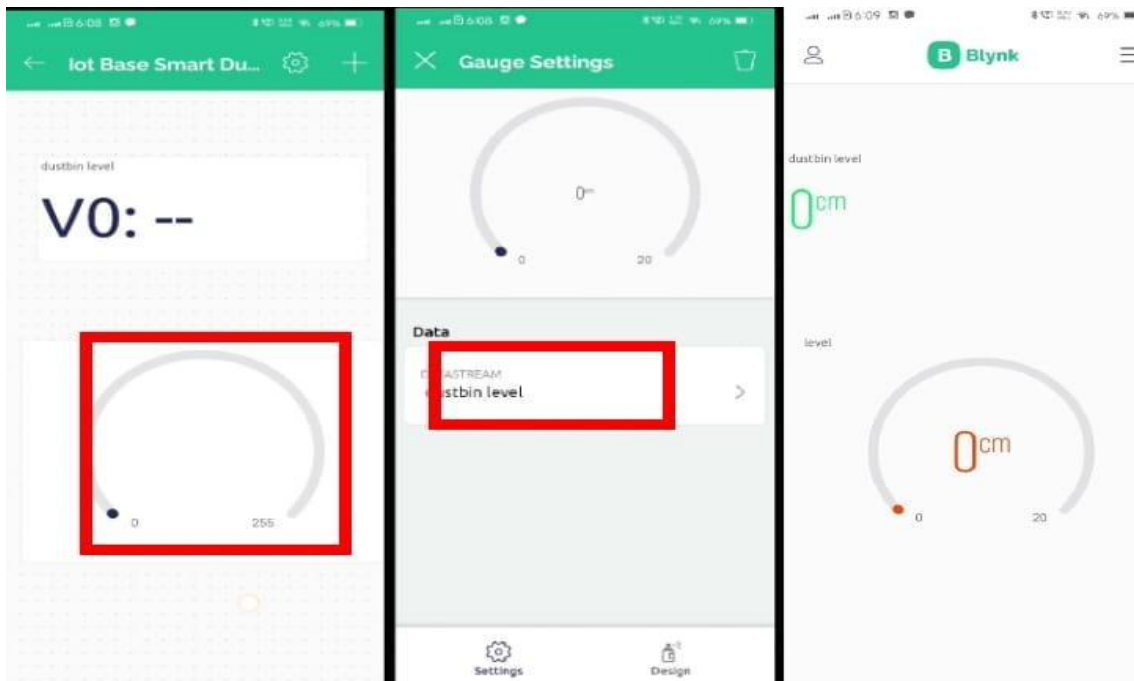
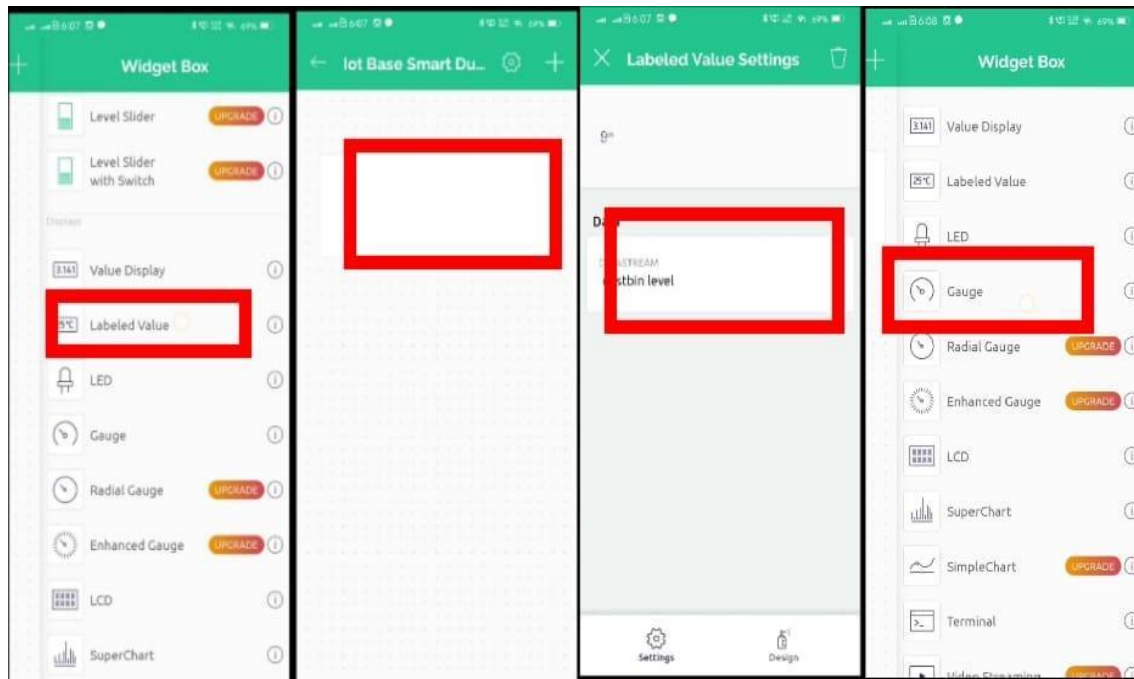
Setup the blynk app:

Install the new blynk app and login. After that follow all the step which is given below in the picture.



The image shows four sequential screenshots of the Blynk mobile app interface, illustrating the steps to enable Developer Mode:

- The first screenshot shows the main Blynk app screen with a hamburger menu icon in the top left corner, which is highlighted with a red box.
- The second screenshot shows the menu with "Developer Mode" selected, also highlighted with a red box.
- The third screenshot shows the "Developer Mode" screen with a "Developer Mode" icon in the top left corner, highlighted with a red box.
- The fourth screenshot shows the "Developer Mode" screen with a "+" button in the top right corner, highlighted with a red box.



Conclusion/Learning Outcomes:

We have learnt the project Smart Dustbin and known the use of Blynk App.

