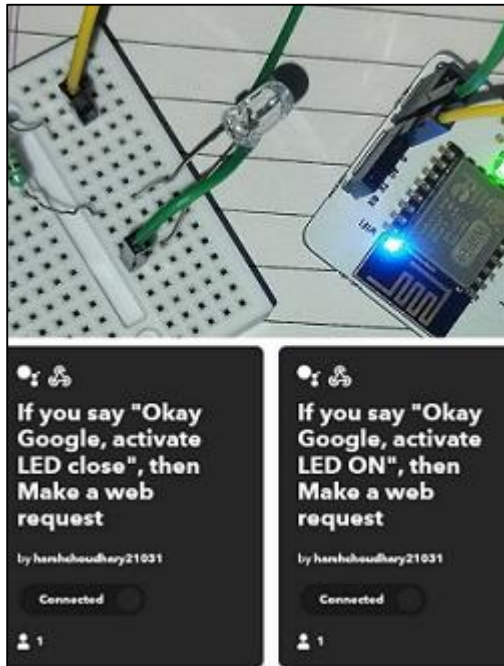


Assistant Automated Light and Sound

You can also view this tutorial at:-

<https://projectsubmission.bolttiot.com/?p=20717&preview=true>



Say to Google Assistant to switch the LED "on" or "off"

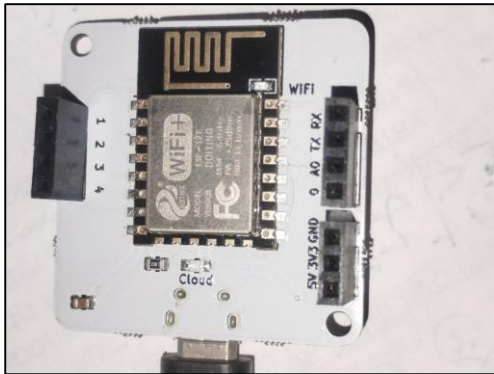
Problem Instance

Home automation is easy and effective. We want to control the Lights of our home with our voice, and play music with our voice. There are many devices out there that offers us this facility to control the lights, fans, air conditioners etc. in our home with voice commands. We sought to build a simplified version of such devices. Here, LED represents the light and buzzer represents the music system. We have used a very basic workflow to achieve this, which we shall describe in this tutorial.

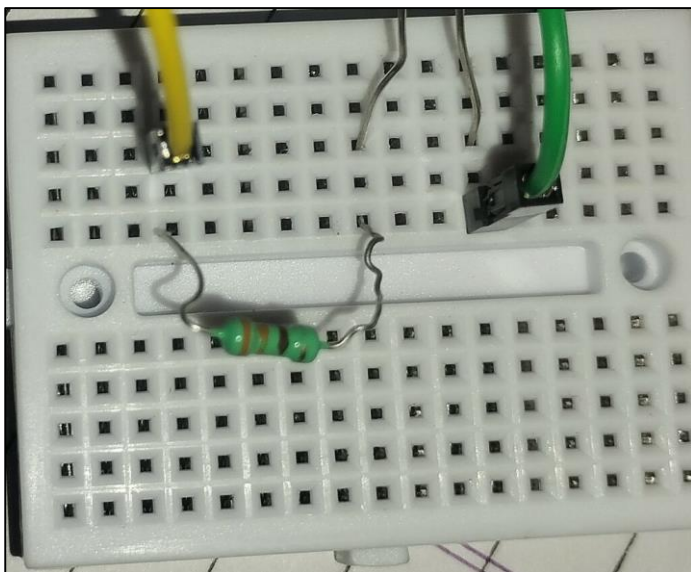
Things used in the Project

Hardware components

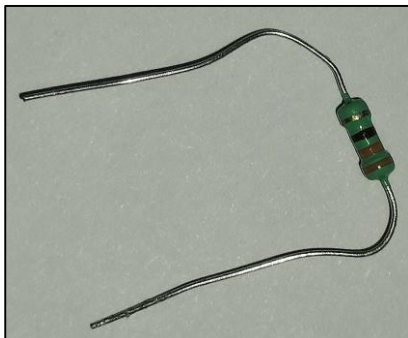
- BOLT IOT WiFi Module



- Mini Breadboard



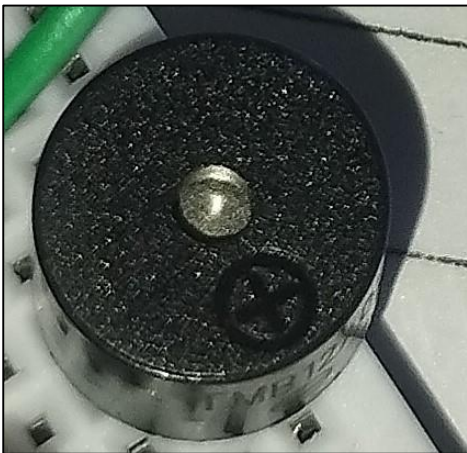
- Resistor



- LED



- Buzzer



- Jumper Wires
- Adapter wire (to power on BOLT module)

All of the above components are included in BOLT IOT/ML training starter kit.

Software apps and components

- [Bolt cloud](#) (for API key and Device Id)



- [IFTTT](#) (for making automation workflow)

IFTTT

- Google Assistant service (to input trigger)



- Google Home mobile app



Overview and Objective

We try to build a simplified version of voice controlled home appliances and devices. To achieve this, we assume the following:-

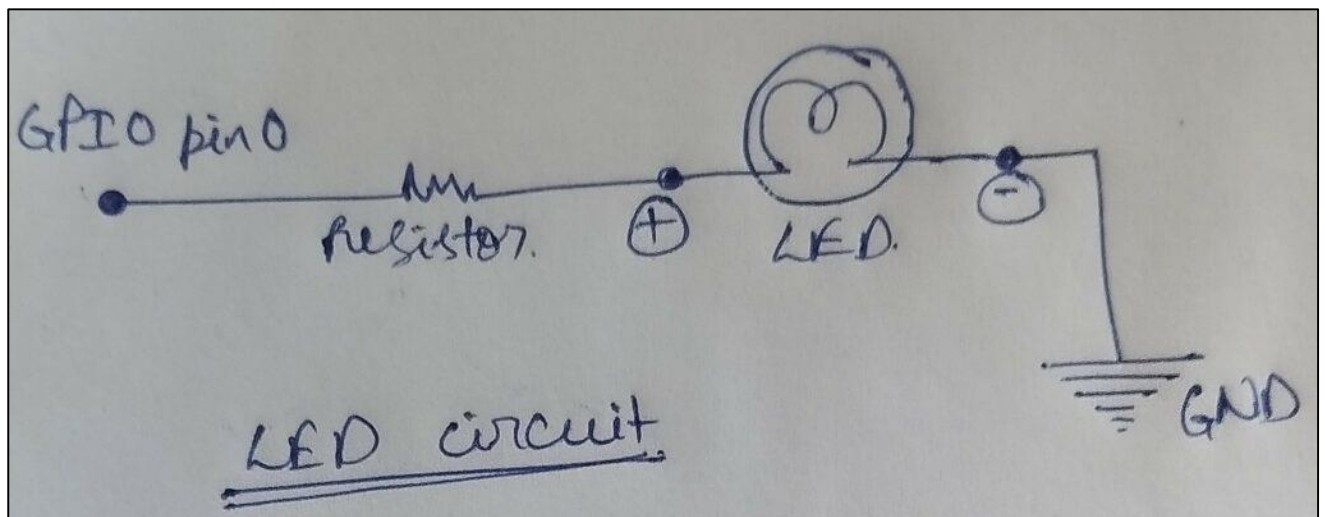
1. LED is used to simulate the lighting of the home.
2. Buzzer is used to simulate the music system or speaker.
3. Google Assistant integrated with Google Home is used to take our speech commands and the workflow in background executes the task we want.

We will use IFTTT free service to build this project, so we are allowed to create 2 workflows with 2 nodes each, at maximum. Hence we will split the tutorial into 2 parts, one for LED and other for buzzer. At the end we shall provide the circuit connection for the combined LED and Buzzer system but executing that is possible by a paid subscription to IFTTT, and hence that system will not be implemented by us. Readers are however encouraged to try it on their own, if they already have the IFTTT paid subscription.

Part 1: Assistant Controlled LED

Hardware Setup

Setup a basic LED circuit using the Bolt WiFi module, one resistor, one LED and jumper wire, on the mini breadboard. The circuit we are going to setup is shown:-

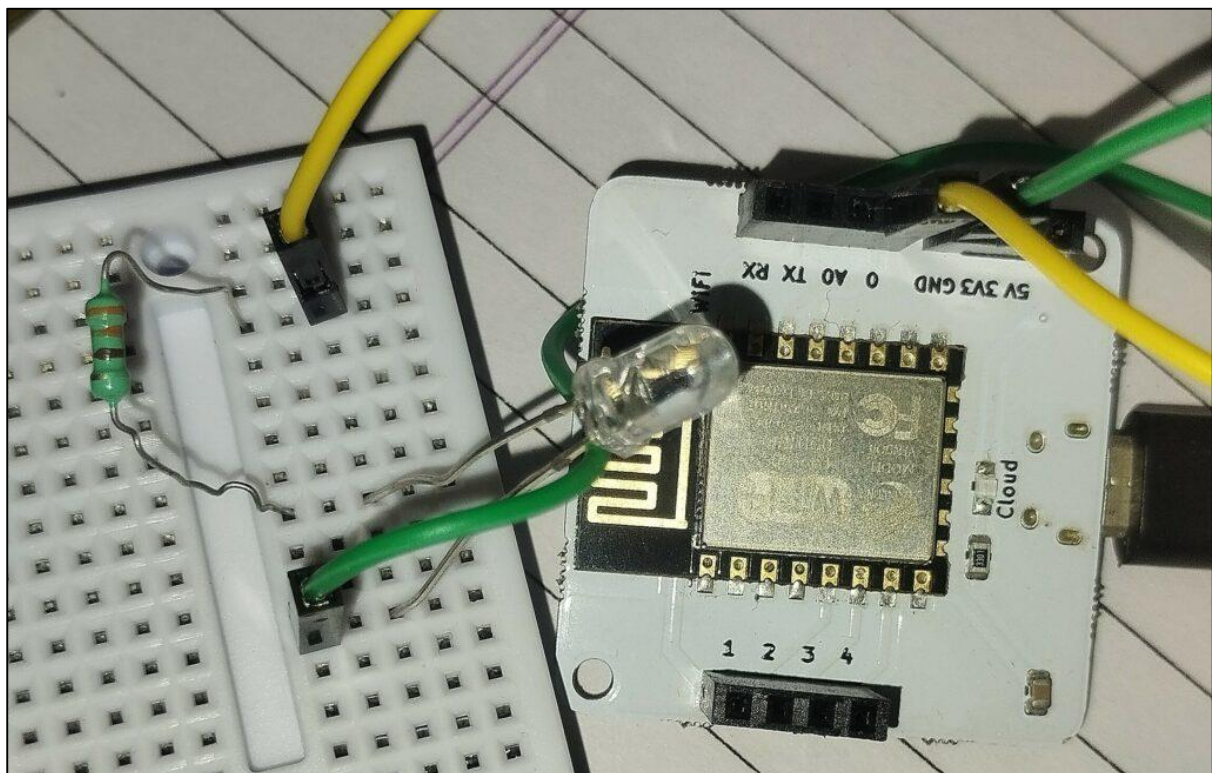


LED circuit as taught in the course

Follow these steps:-

1. The column on same side of the divider line in a mini breadboard are at same potential, i.e., that are electrically equivalent.

2. Connect the longer leg (+ve pin) of LED into a column of mini breadboard, call this column c1. Connect the shorter leg of LED (-ve pin) to another column of breadboard, and call this column c2.
3. Use a jumper wire to connect column c2 to the “GND” pin of the BOLT WiFi module. Thus, we have connected the negative LED terminal to the ground successfully.
4. Now, fix one leg of resistor (call it R) into column c1 and other leg of resistor into another column, say c3.
5. Connect c3 to GPIO pin 0 of the BOLT WiFi module using a jumper wire. Thus, the positive terminal of LED is connected to GPIO pin 0 with a resistance R.



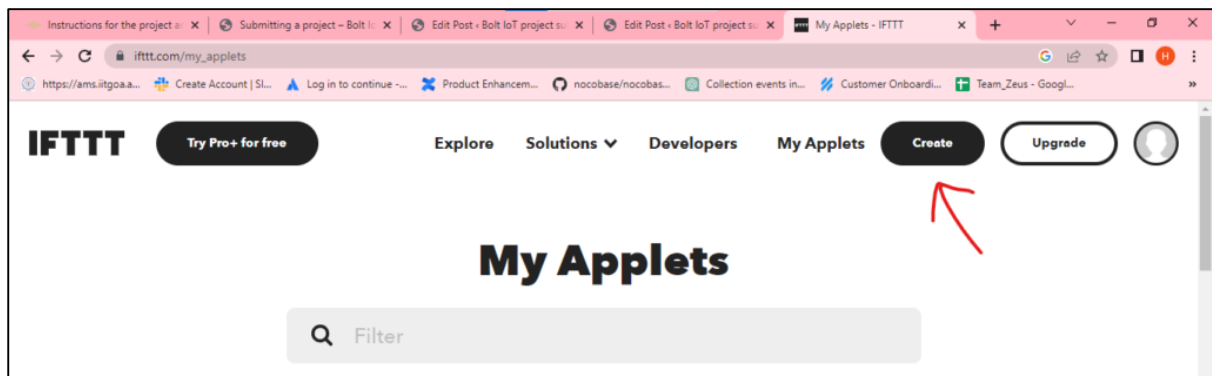
Notice the color of wires, the yellow wire connects positive LED terminal to GPIO pin 0 via a resistance, and the green wire connects negative LED terminal to the GND pin.

Software Setup

We need to create an account on [IFTTT](https://IFTTT.com), as we shall be using it to create the workflow for the setup. Click on the link above and

complete your account setup. Once the account is made, follow the below steps:-

- IFTTT in its free version provides us capability to create a “IF This Then That” kind of logic based applet, which in other terms is also called workflow.
- Click on the “Create” button.



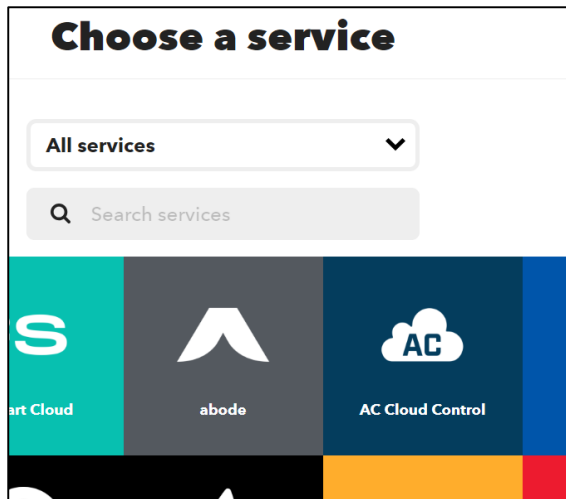
The screen may look somewhat different depending on the updates of the website. Find the “create” button to create your applet (workflow)

- Click on “Add” button to add the trigger, that is basically the condition when this applet runs.

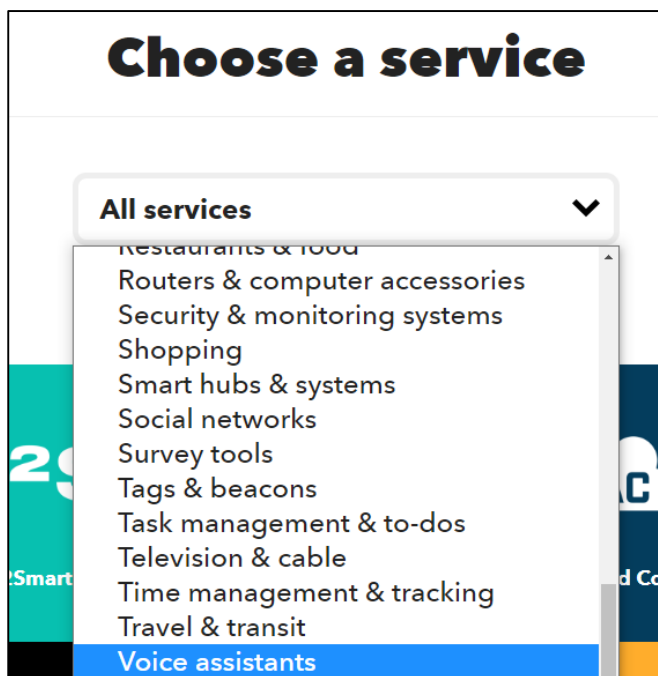


Click on “Add” button

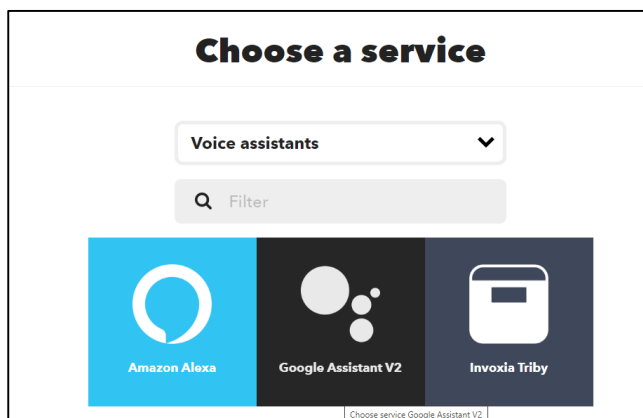
- Click on “All Services” to open the drop down list, and select “Voice Assistants”. Then select “Google Assistant v2” as we shall be using it to give voice commands.



Click on "All Services" to open the drop down

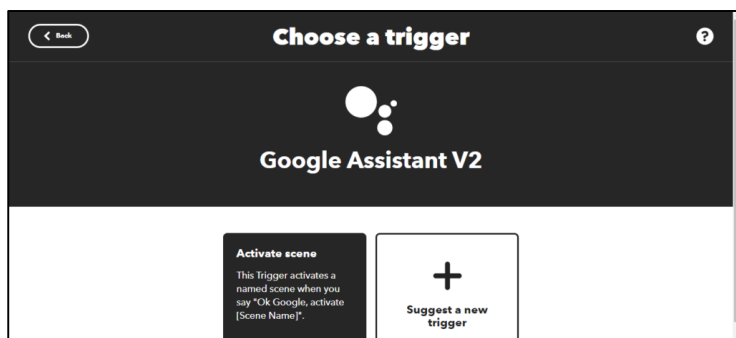


Choose "Voice assistants"



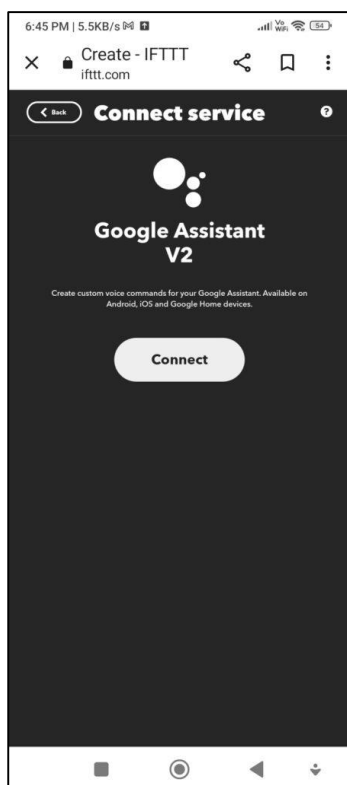
Choose "Google Assistant V2"

- Then it will open the window as shown below, wherein you will have to provide the appropriate trigger which will cause the applet to run. Click on “Activate Scene” to configure it according to what “voice command” shall you give to google assistant when it shall run the applet.



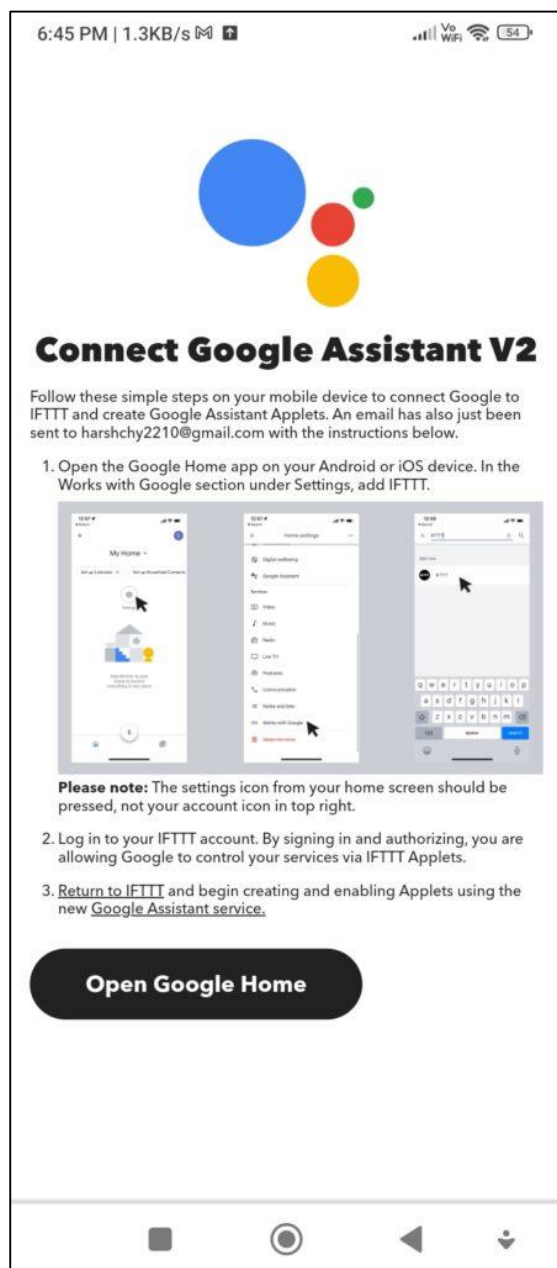
Click on “Activate Scene” to configure the scene name, that will trigger the applet.

- When you set it up for the first time, you will get some guidelines to link IFTTT to your Google Home. Follow the steps below from your smartphone to set it up. Make sure Google Home is installed on your smartphone.



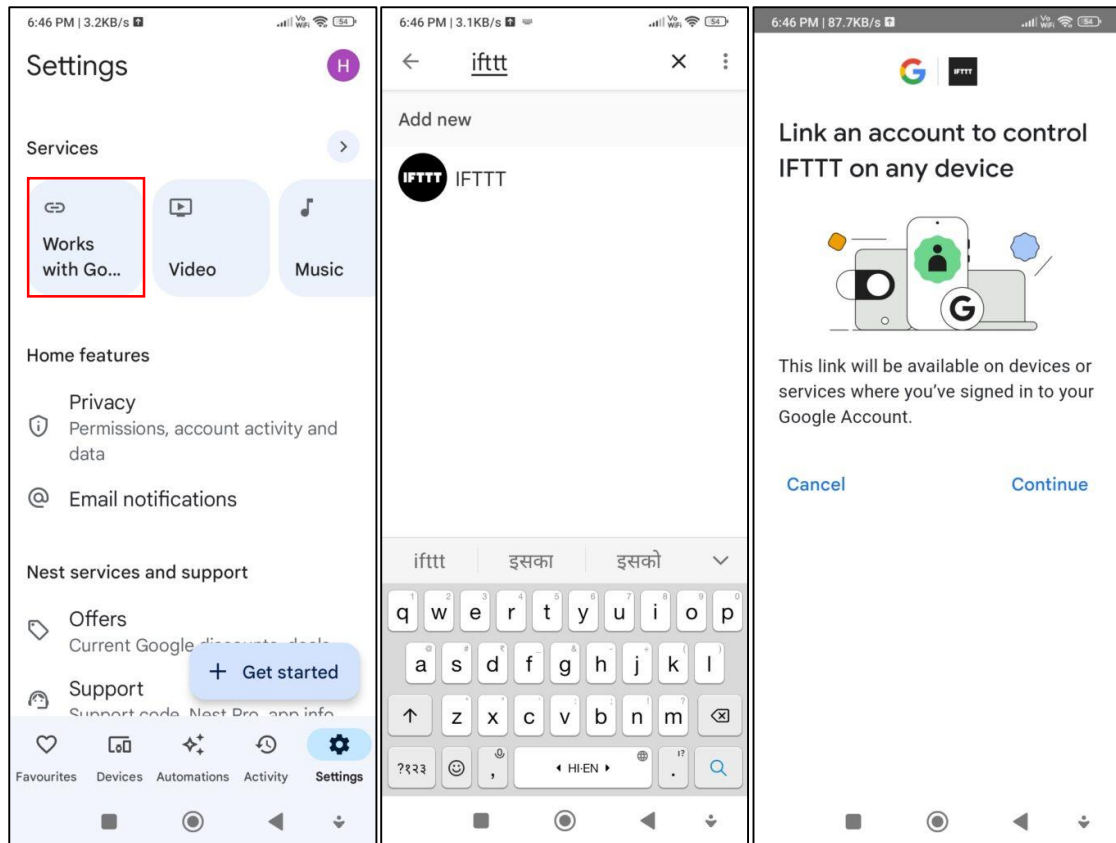
Click on “Connect” button

- On clicking the connect button, the below screen shows up:-



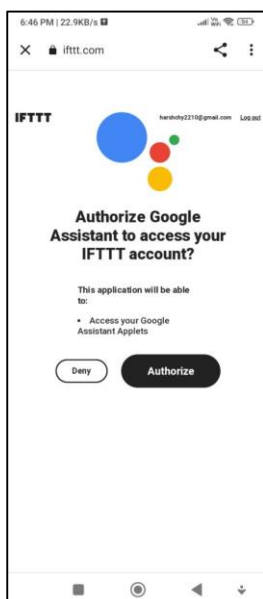
This shows the steps you need to perform in order to link IFTTT to Google Home and then we can use Google Assistant service.

- Open Google Home and follow these steps to link your Google Home to IFTTT.

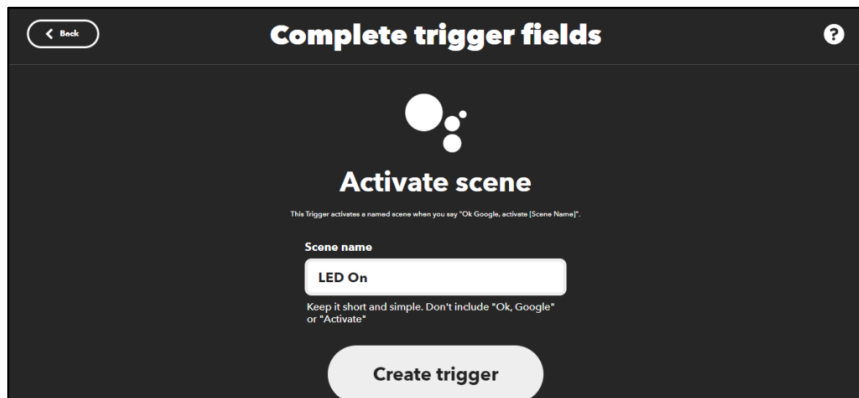


1. Open Google Home and go to "Settings" tab
2. Select "Works with Google" and then search for IFTTT.
3. Select IFTTT and then the screen asks you to "continue". Press "Continue"

- Then finally click "Authorize" to complete the connection. You will get some warning dialog boxes, but close them and open IFTTT again in your browser and now you will be able to follow from next step.



- Now when you click on “Activate Scene” you see the below screen. Give the scene name as “LED on”. Then click on create trigger.



Give the scene name as “LED On”

- Thus, you have finished setting up the trigger that will cause your applet to run, which forms the very first node (If condition of your workflow). When you will say “Ok Google, Activate ‘LED On’”, the applet shall run. You can edit this trigger by clicking on “edit” button whenever you want.

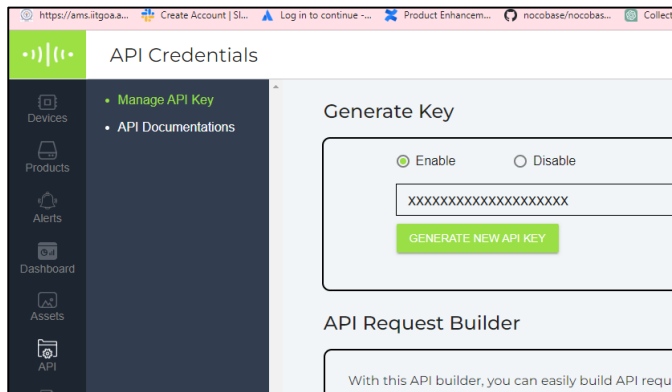


“If” part is configured, and before moving to the “then” part we have to obtain our API credentials from the BOLT cloud

Now, we want that when you say “Ok Google, Activate LED On” to your Google assistant, the LED set up in hardware setup section actually glows up. For this we shall make an API request using Bolt cloud to your Bolt Wifi Module’s GPIO pin 0, as that is the pin we connected the positive terminal of LED to. Follow these steps for getting credentials to make API request using BOLT cloud:-

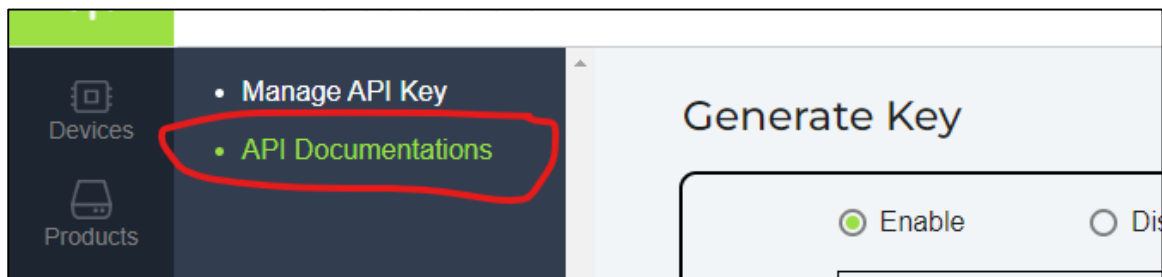
- Login to the [BOLT](#) cloud. (click on the link)

- Go to API tab and generate your API key, and copy it somewhere.



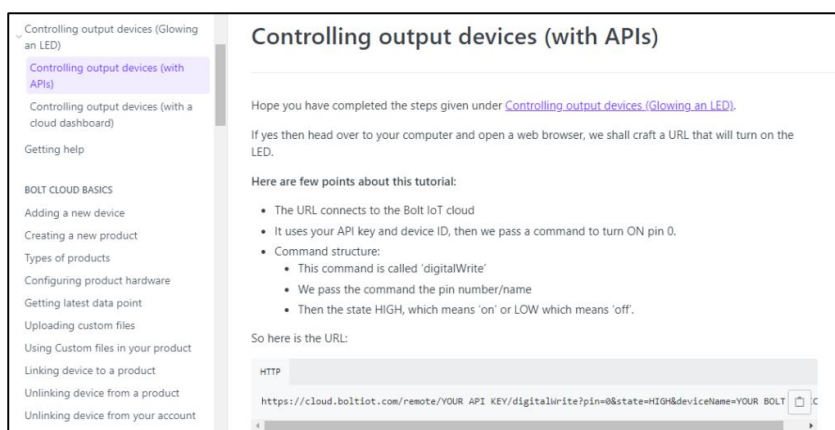
Enable API key and copy it somewhere as we shall use it later.

- Click on the API documentation to get the URL which will be used to make API request to BOLT module.



We shall find all required information in API documentation

- This is the page where we find all instructions to make API requests.

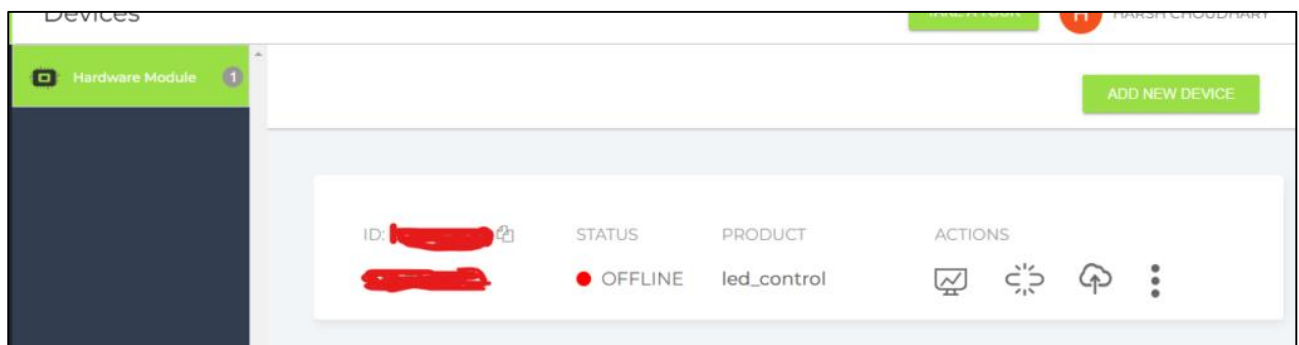


Click on the [link](#) to visit the page

- Let us try to understand the URL.

<https://cloud.boltiot.com/remote/YOUR API KEY/digitalWrite?pin=0&state=HIGH&deviceName=YOUR BOLT DEVICE ID>

Replace “YOUR API KEY” with your API key you copied from BOLT cloud. The “YOUR BOLT DEVICE ID” should be replaced with your Bolt’s device ID that you see on the BOLT cloud.



Check this in your BOLT cloud. Copy it from there

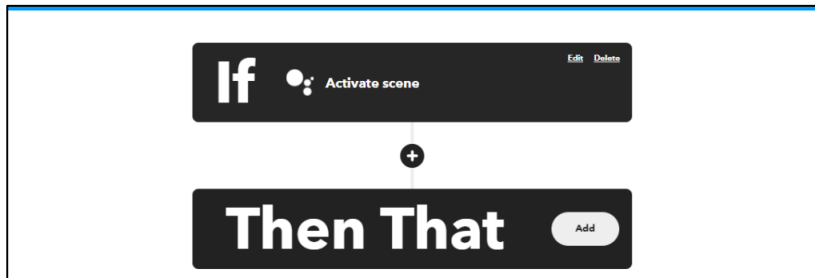
Now, we see the query parameters that are being passed into this URL. The part that follows the ‘?’ symbol in URL are the query parameters and the values passed determines what services we are delivered.

- Here, “pin” is a query parameter and the value passed is ‘0’ as we have connected the positive terminal of LED to GPIO pin ‘0’. If some other pin is used, simply put the value as that pin number.
- The “state” parameter commands the module to raise/drop the potential of the specified pin as per the value it has. Here the “state” parameter is having value ‘HIGH’ which means it increases the potential of specified pin (i.e. 0) and due to the potential difference being applied, the LED shall glow. If we want to switch off the LED, the value should be passed as ‘LOW’.
- We have already looked at “deviceName” parameter.

Now, we shall proceed on building the action node of the workflow, that can also be called the “then” part of this applet. (Basically, given

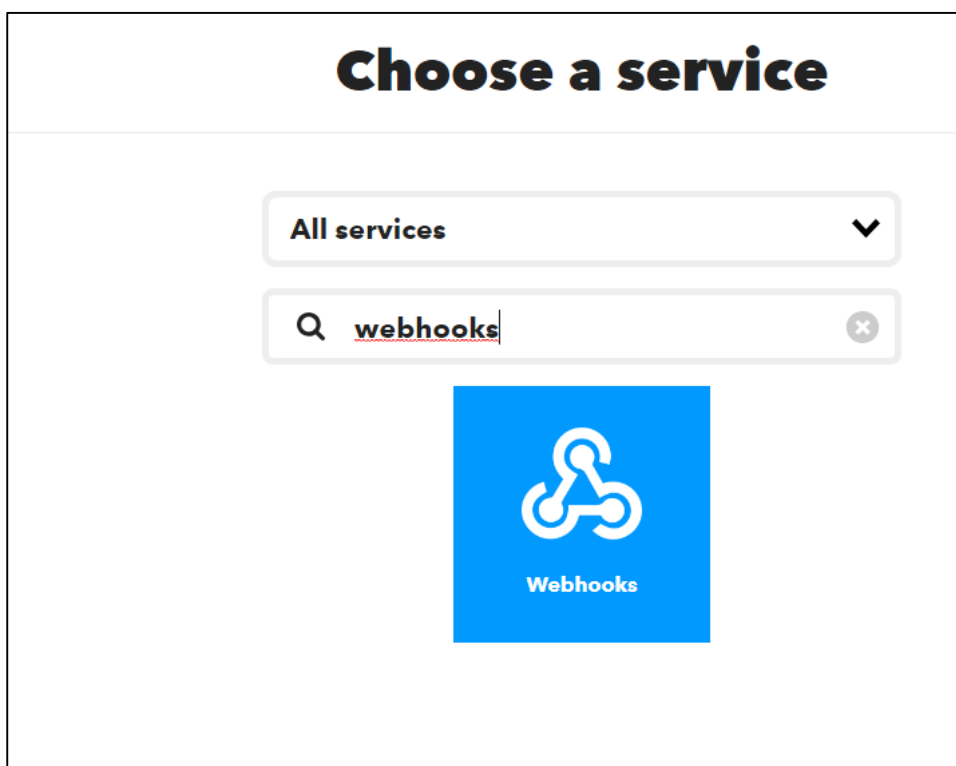
a trigger we perform an action, and on IFTTT we refer to this as given an 'IF' condition we perform some 'THEN' action).

- Click on the “Add” button. This will allow us to configure the web API call settings.



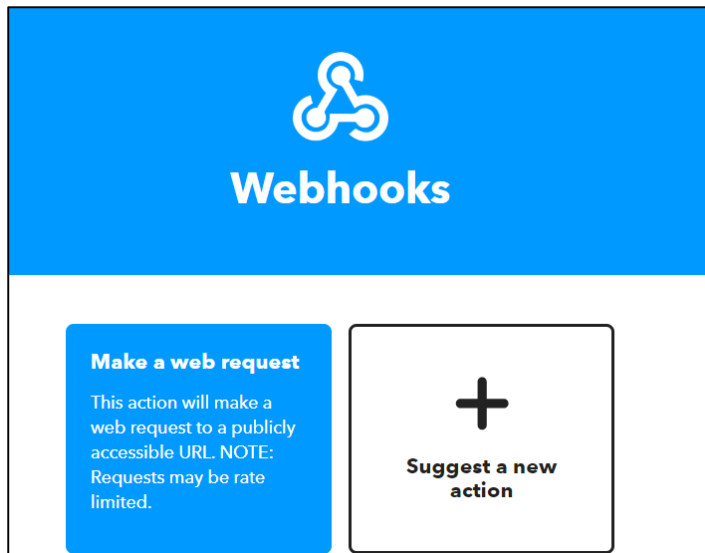
Click on the “Add” button in the then part, where we shall make the web API call

- For selecting the service, let the “All services” be selected and type “webhooks” in the “Filter” textbox.



we will use Webhooks as they allow us to make API call automatically in response to some event

- Click on “Make a Web Request” blue button to configure the URL and the request method to make HTTP API request.



Click on “Make a Web request”

- Enter the URL for API request here. This is the same URL which we explained above. Make sure all the parameters and API key are entered correctly.

Make a web request

This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

URL

`https://cloud.boltiot.com/remote/ba543d63-3c48-407f-a793-ba0ad98e77c0/digitalWrite?pin=0&state=HIGH&deviceName=BOLT987342`


Surround any text with <<< and >>> to escape the content. See [FAQ](#) if using an IPv6 URL.

Add ingredient

Enter the URL for API request here. This is the same URL which we explained above

- Enter the request type as “GET” and content type as “application/json”. The final configuration page looks like as shown below. Click on “create action” button.

Complete action fields



Make a web request

This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

URL

```
https://cloud.bolttiot.com/remota/
ba543d63-3c48-407f-a793-
ba0ad98a77c0/digitalWrite?
pin=0&state=HIGH&deviceName=
BOLT987342
```

Surround any text with <<< and >>> to escape the content. See [FAQ](#) if using an IPv6 URL.

Method

GET

The method of the request e.g. GET, POST, DELETE

Content Type

application/json

Optional

Additional Headers

Each header should be on a new line formatted as Some-Header: Some-Value

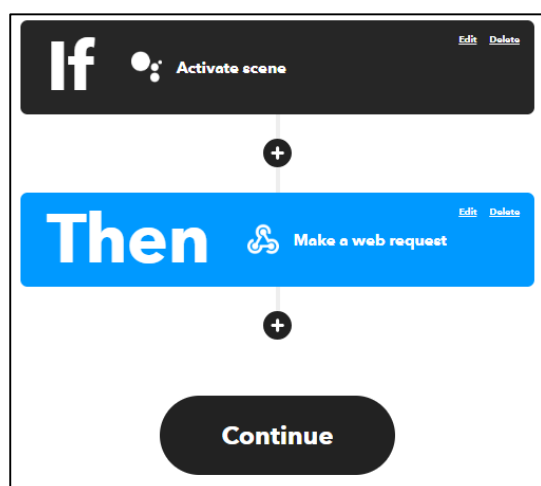
Body

Surround any text with <<< and >>> to escape the content (ex. to preserve line breaks). Learn more [here](#).

Create action

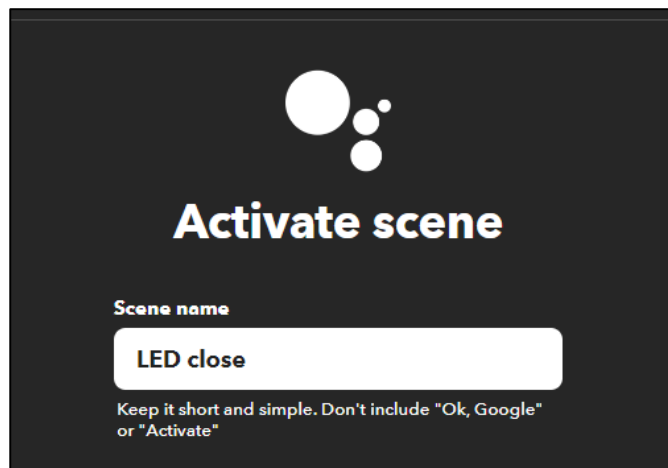
Click on “Create action” button

With this we have successfully configured our applet. Click on “Continue” to proceed.

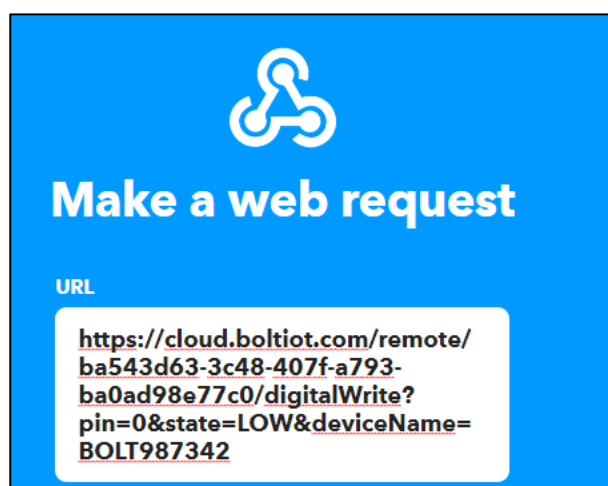


Till now we have completed the applet to switch the LED On. Now we shall make a similar applet to switch the LED Off.

- Follow exactly the same steps as above, but when you give the scene name, the name should be “LED Close”. (You can use phrase “LED Off” but the assistant may recognize your voice as ‘of’ instead of ‘off’ and the applet won’t be triggered then.)

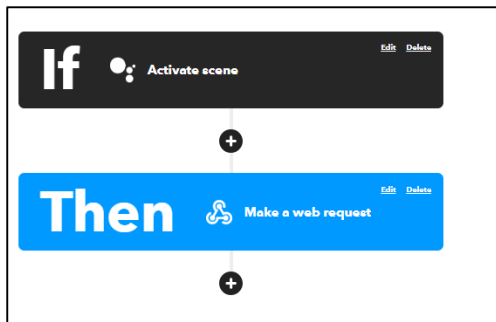


- Also, for configuring the “then” part of the applet, we have to pass the API request to switch Off the LED. For this, we have to pass the parameter value ‘state’ to ‘LOW’ instead of ‘HIGH’.

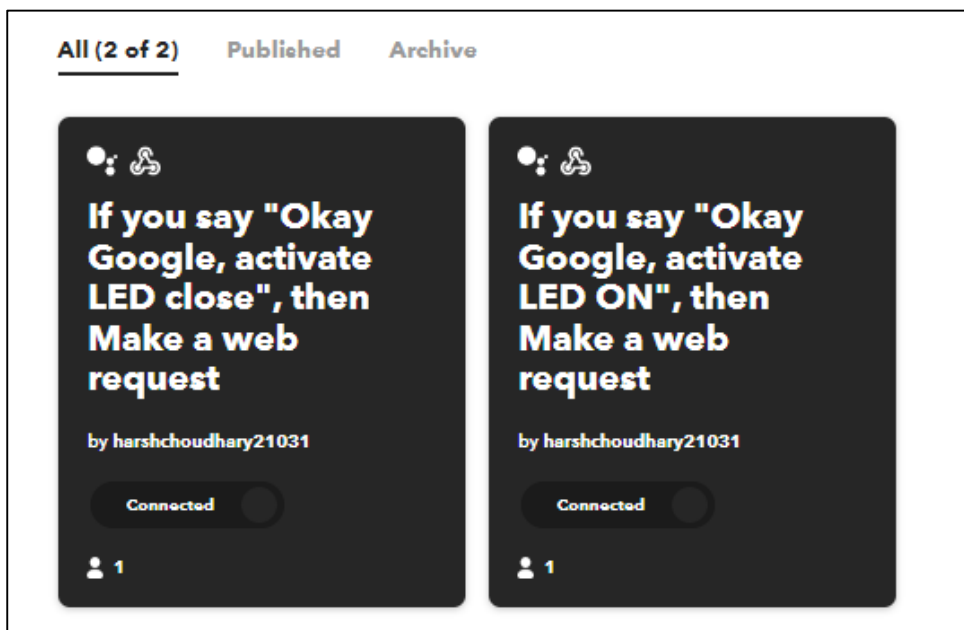


The value of “state” parameter is ‘LOW’, so when it runs the LED is switched off

The final applet to switch the LED off is shown below:-



Finally, with both these applets running and connected, we can control the LED using voice with the help of Google Assistant.



Power ON the BOLT WiFi module and then you should be able to do the following:-

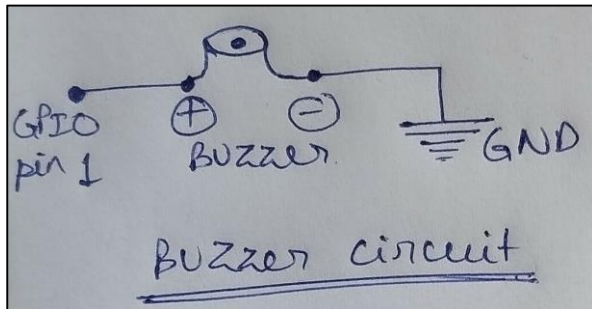
- Open Google Assistant on your smartphone (by long pressing the “Home” button) and say “Ok Google, Activate LED On”. This will switch the LED on.
- On the Google Assistant only, say “Ok Google, Activate LED Close” and this switches the LED off.

You can view the video on [YouTube](#) or from Google Drive folder.

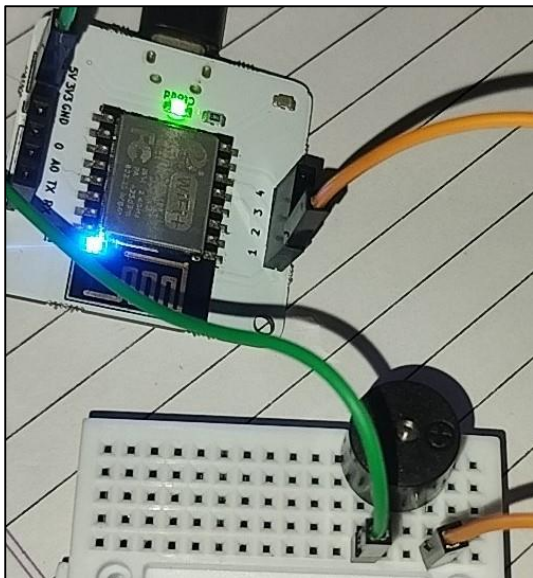
Part 2: Assistant Controlled Sound

Hardware Setup

The circuit connection for the Buzzer is very similar to that of the LED apart from a difference that there is no need of a resistance in case of Buzzer circuit. The circuit we are going to make is shown:-



- Connect the negative end (shorter leg) of the buzzer to a column, say c1 of the mini breadboard and use a jumper wire to connect this column to the “GND” pin of the BOLT module.
- Connect the positive terminal (longer leg) of the buzzer to the another column, say c2 of the breadboard, and using a jumper wire connect it to the GPIO pin 1 of the BOLT module. (You can use any pin of your choice)

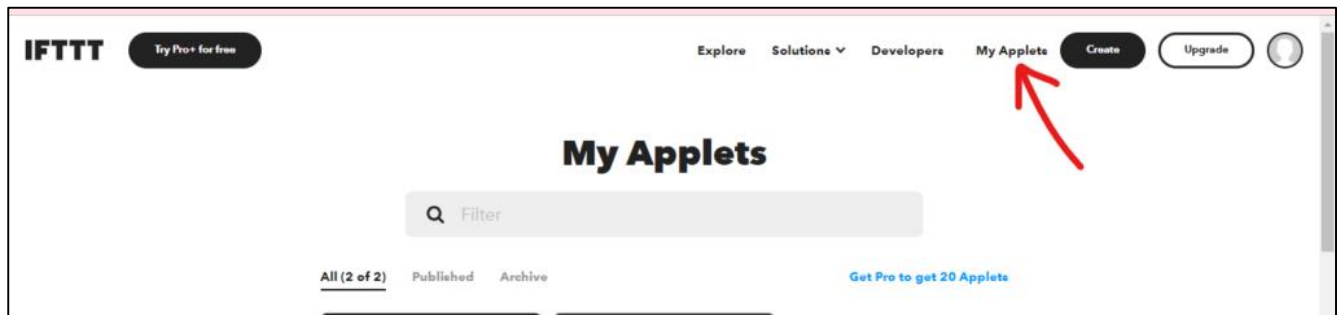


Buzzer circuit connection, green wire connects -ve buzzer end to GND and orange wire connects +ve buzzer end to GPIO pin 1

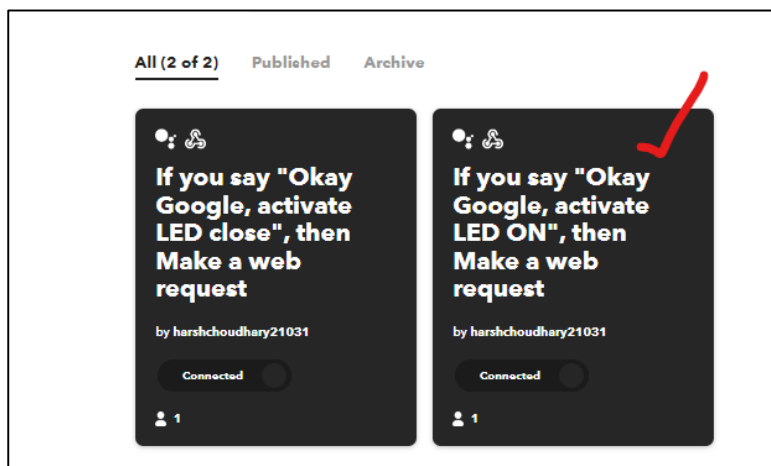
Software setup

With free version of IFTTT, only two applets are allowed, so we have to edit the applets we created for the LED connection itself. Follow these simple steps and things will be fixed as per the buzzer.

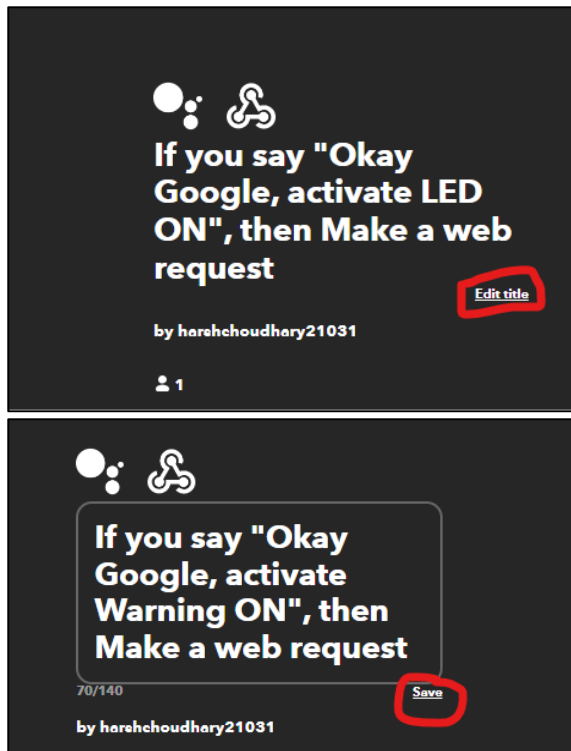
- Click on “My Applets” on the IFTTT screen.



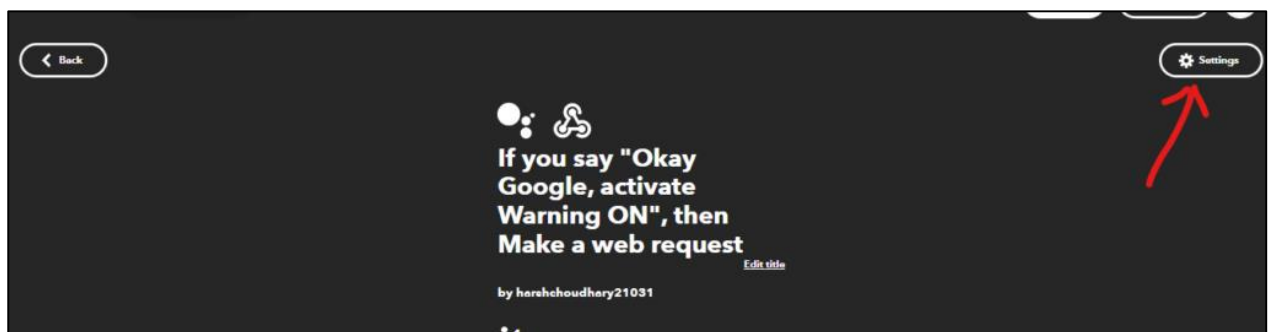
- Select the “LED On” applet, i.e. the one you created to switch on the LED.



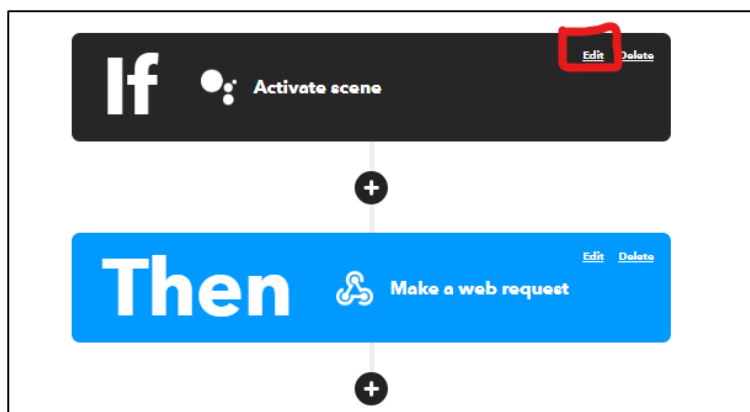
- Now click on “Edit Title” button and replace LED with Warning. Click on “save” button.



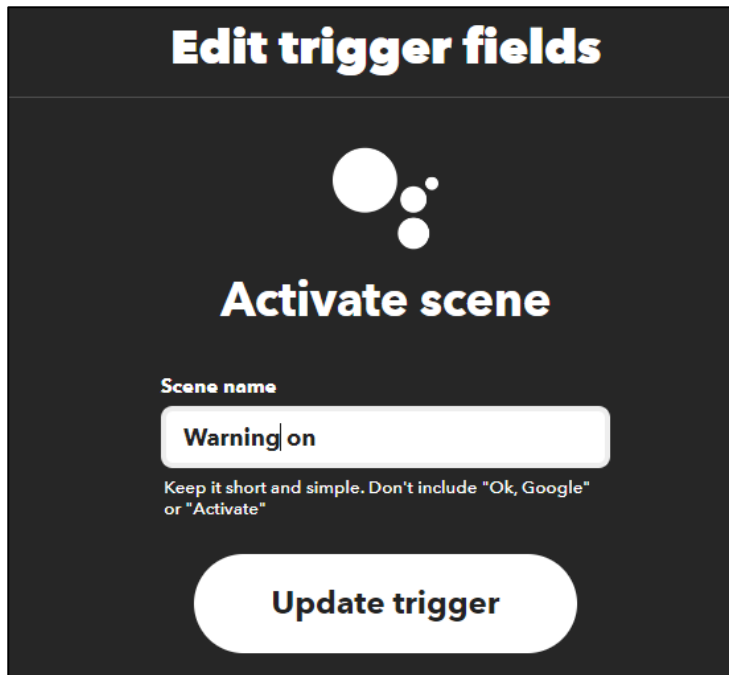
- Click on the “settings” icon to configure the applet.



- Click on the “edit” button for the “if” node, and change the scene name to “warning on”. Then click on “Update Trigger” button.



Click on “edit” button



Edit trigger fields

Activate scene

Scene name

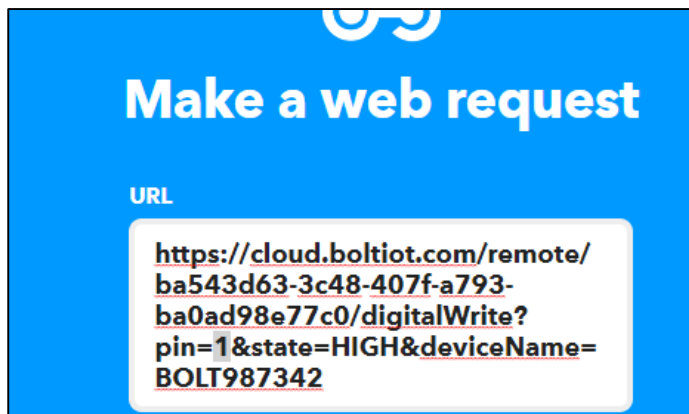
Warning on

Keep it short and simple. Don't include "Ok, Google" or "Activate"

Update trigger

Click on "Update trigger"

- Next "edit" the "then" action webhook node, and change the URL so that pin=1 (as buzzer's positive leg is connected to GPIO pin 1) is entered instead of pin=0. Then press the "Update action" button.




Make a web request

URL

<https://cloud.boltiot.com/remote/ba543d63-3c48-407f-a793-ba0ad98e77c0/digitalWrite?pin=1&state=HIGH&deviceName=BOLT987342>

Change pin=0 to pin=1 and remaining URL is same

Edit action fields



Make a web request

URL

`https://cloud.bolt.io.com/remote/
ba543d63-3c48-407f-a793-
ba0ad98e77c0/digitalWrite?
pin=1&state=HIGH&deviceName=
BOLT987342`

Surround any text with <<< and >>> to escape the content. See [FAQ](#) if using an IPv6 URL.

Add ingredient

Method

GET

The method of the request e.g. GET, POST, DELETE

Content Type

application/json

Optional

Additional Headers

Each header should be on a new line formatted as Some-Header: Some-Value

Add ingredient

Body


Surround any text with <<< and >>> to escape the content (i.e. to preserve line breaks). Learn more [here](#).

Add ingredient

Update action

Click on "Update Action"

- Similarly, edit the other applet to switch off the buzzer. You again only need to edit the title, the trigger and the action.



Activate scene


Scene name

Warning close

Keep it short and simple. Don't include "Ok, Google" or "Activate"

Update trigger

Edit action fields



Make a web request


URL

`https://cloud.bolt.io/remote/
ba543d63-3c48-407f-a793-
ba0ad98e77c0/digitalWrite?
pin=1&state=LOW&deviceName=
BOLT987342`

Surround any text with <<< and >>> to escape the content. See [Add ingredient](#)

- Finally, you should have the following two applets.

All (2 of 2)PublishedArchive




If you say "Okay Google, activate Warning close", then Make a web request

by harshchoudhary21031

Connected

1



If you say "Okay Google, activate Warning ON", then Make a web request

by harshchoudhary21031

Connected

1

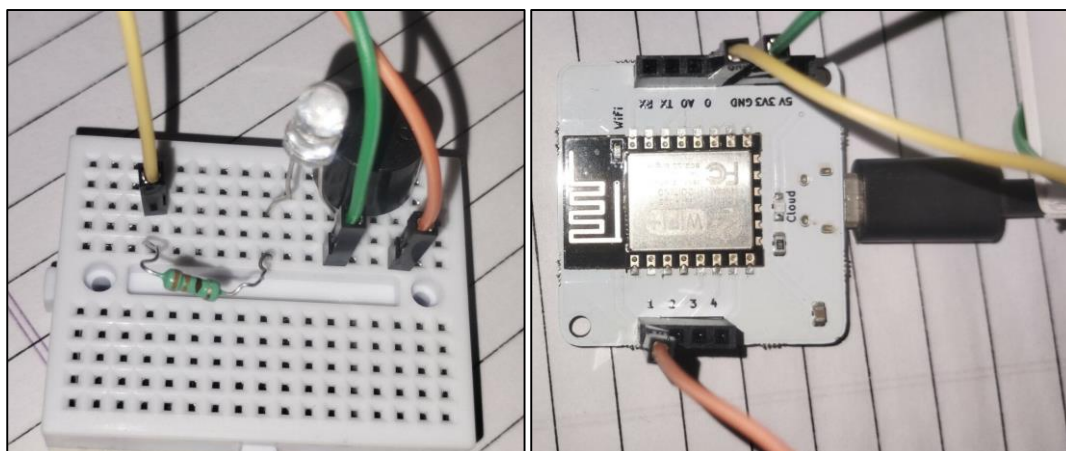
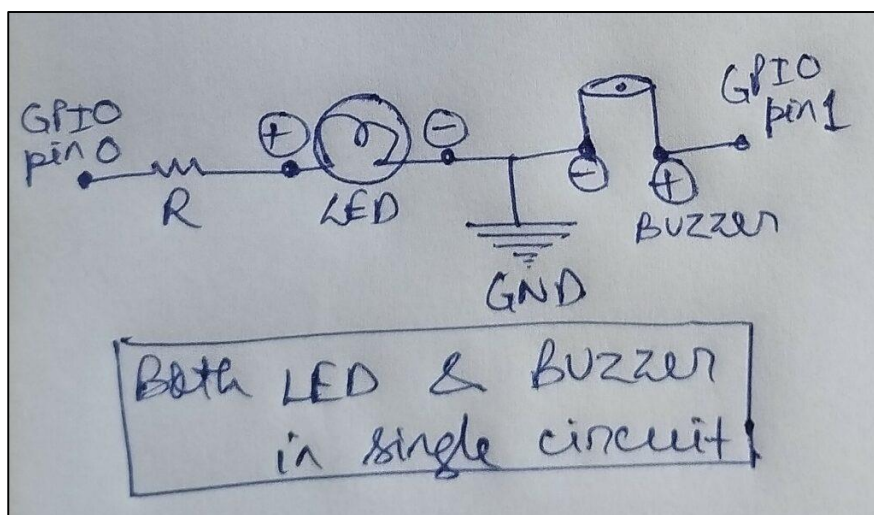
After doing the above changes, you should be able to :-

- Say “Ok Google, Activate warning on” to the google assistant and the buzzer starts sounding.
- Say “Ok Google, activate warning close” and the buzzer stops beeping.

You can view video demo on Google drive folder or on [You Tube](#)

Circuit for combined buzzer and LED control

We can setup the circuit shown below to achieve the controlling of LED and Buzzer simultaneously.



Notice the color of wire to understand the connection. Yellow wire connects the positive end of LED to GPIO pin 0 via a resistor. Orange wire connects positive terminal of Buzzer to GPIO pin 1. The green wire connects the negative terminal of both the LED and Buzzer to the GND pin of BOLT module.

If you have the paid subscription of IFTTT, simply keep up all the 4 applets described above and you will be able to switch on/off the LED or the Buzzer using your voice, thanks to Google Assistant. I do not have a paid subscription, but readers are encouraged to experiment this out, in case they are a pro member of IFTTT.

Conclusion

Thus, with the help of IFTTT applets, we are able to automate the controlling of LED and the Buzzer using Google Assistant and webhooks. The Bolt cloud API enables us to perform digital write on the digital GPIO pins via web requests.