

# Thief Buzzer Away

**Smart Locker With Auto Detection of robbery, Sending alert!!** through SMS and turning on the light of the room with a high Buzzer sound.

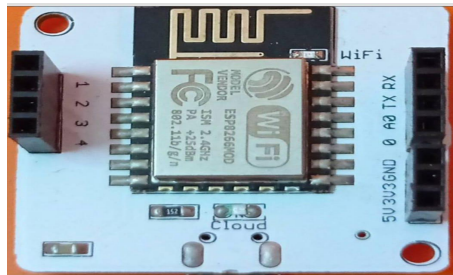
Thief Nowadays is Smart enough to break the lock of a safe easily, so to surprise them, I have built these **smart vaults** with **automatic detection** of robbery and **switching on the light** of the room with a **high buzzer sound** and a **message to the owner of the vault**.

Are you **excited!!** to Built it, let us explore it.

## Things used in This Project

### Hardware Components

- Bolt WiFi Module x1



- Light Sensor (LDR) x1



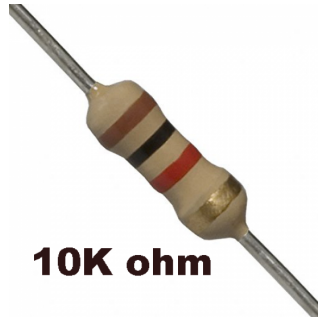
- LED x1



- Buzzer x1



- 
- Resistor x1



- 
- Breadboard x1



- 
- Jumper wire x5



- 
- USB Cable x1

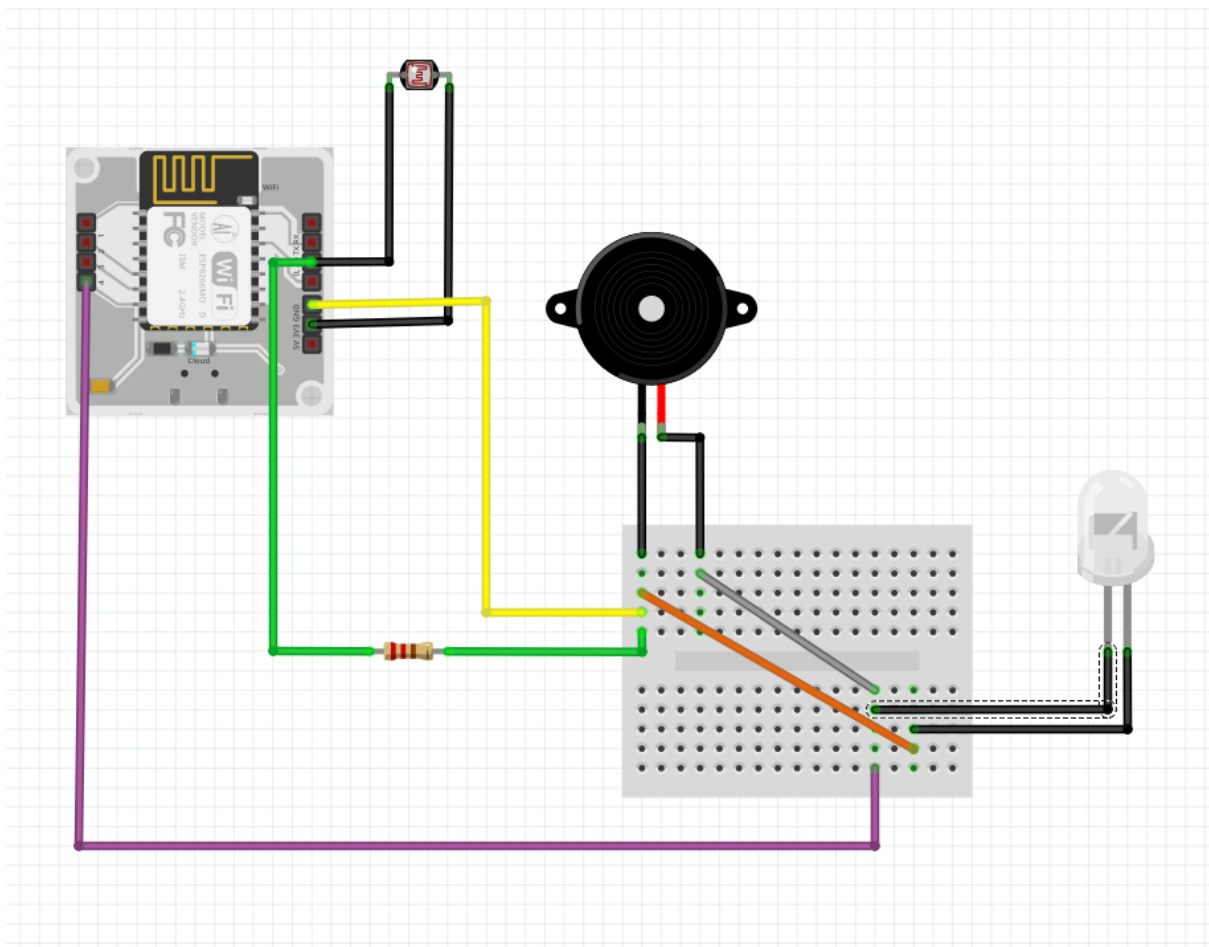


○

## Software, App and Online services

- Boltiot Cloud
- Twilio(Messaging Service)
- Bolt mobile app

## Hardware Setup



Circuit Setup

Youtube link

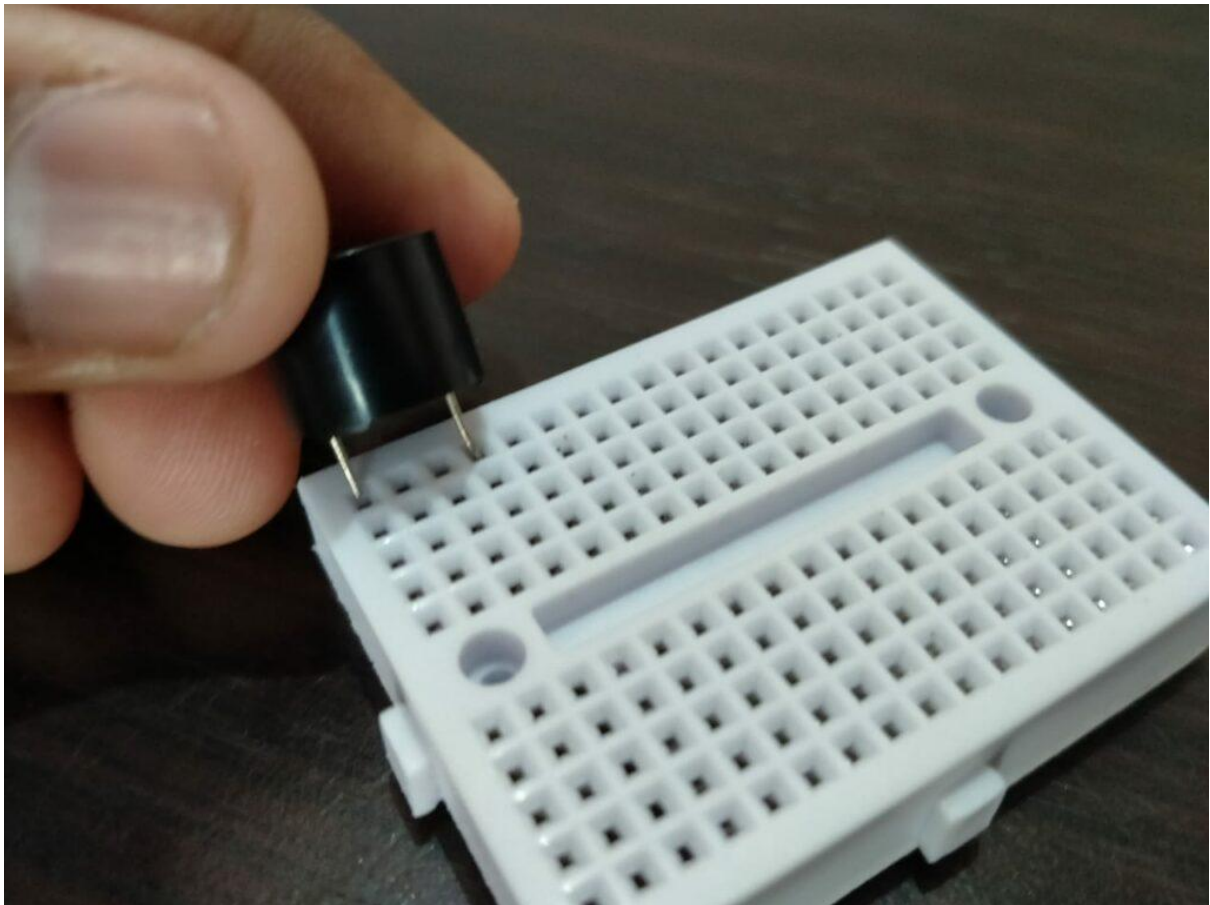
<https://youtu.be/SVq9SFLgue8>

## Process

This whole project is based on the detection of changes in light intensity inside the locker and sending alerts through Buzzer and Sms.

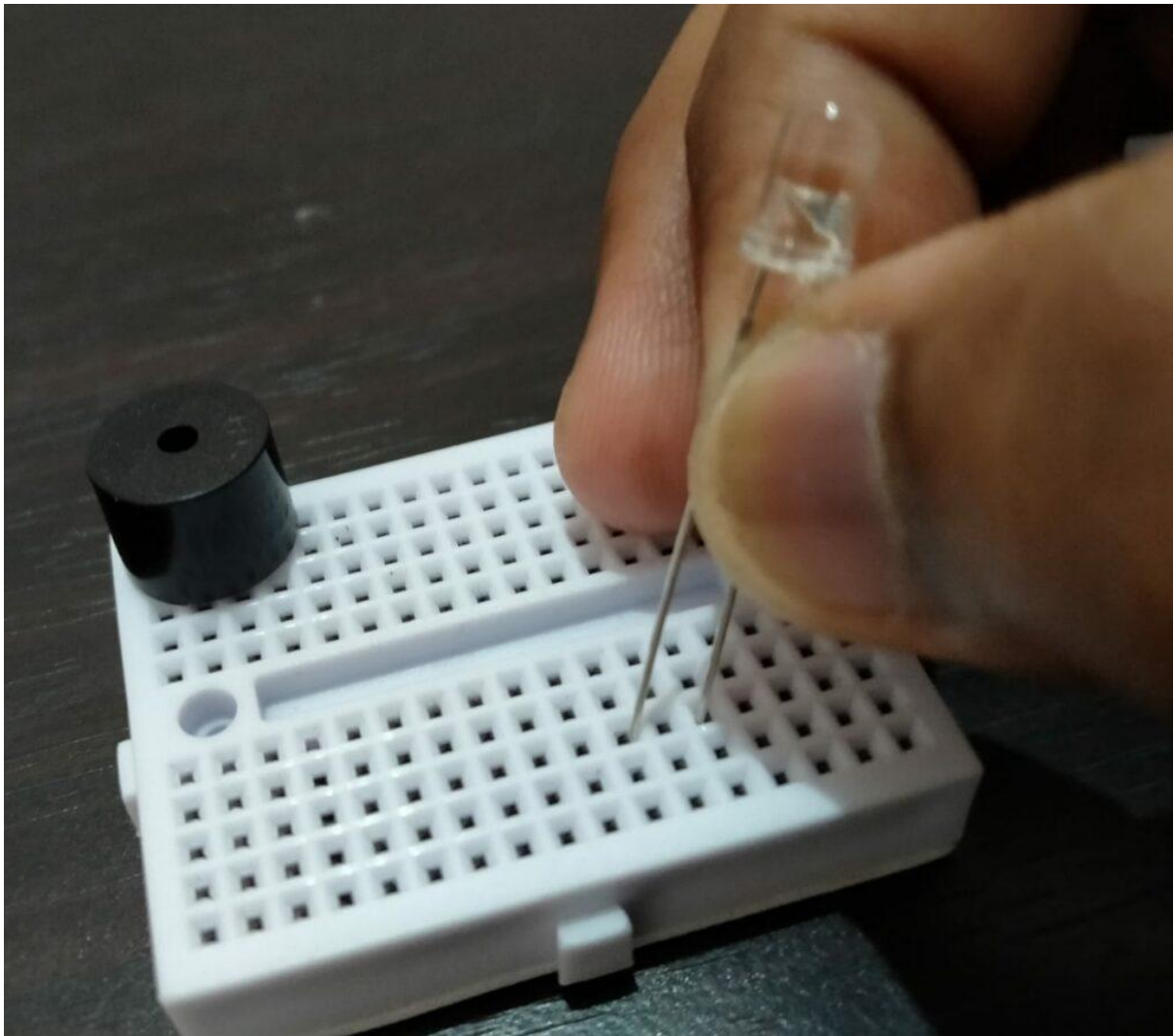
We are using a machine learning algorithm of determining the pattern of intensity by **z score analysis** using variance and mean. it will send an alert if it detects an **anomaly**.

Step 1



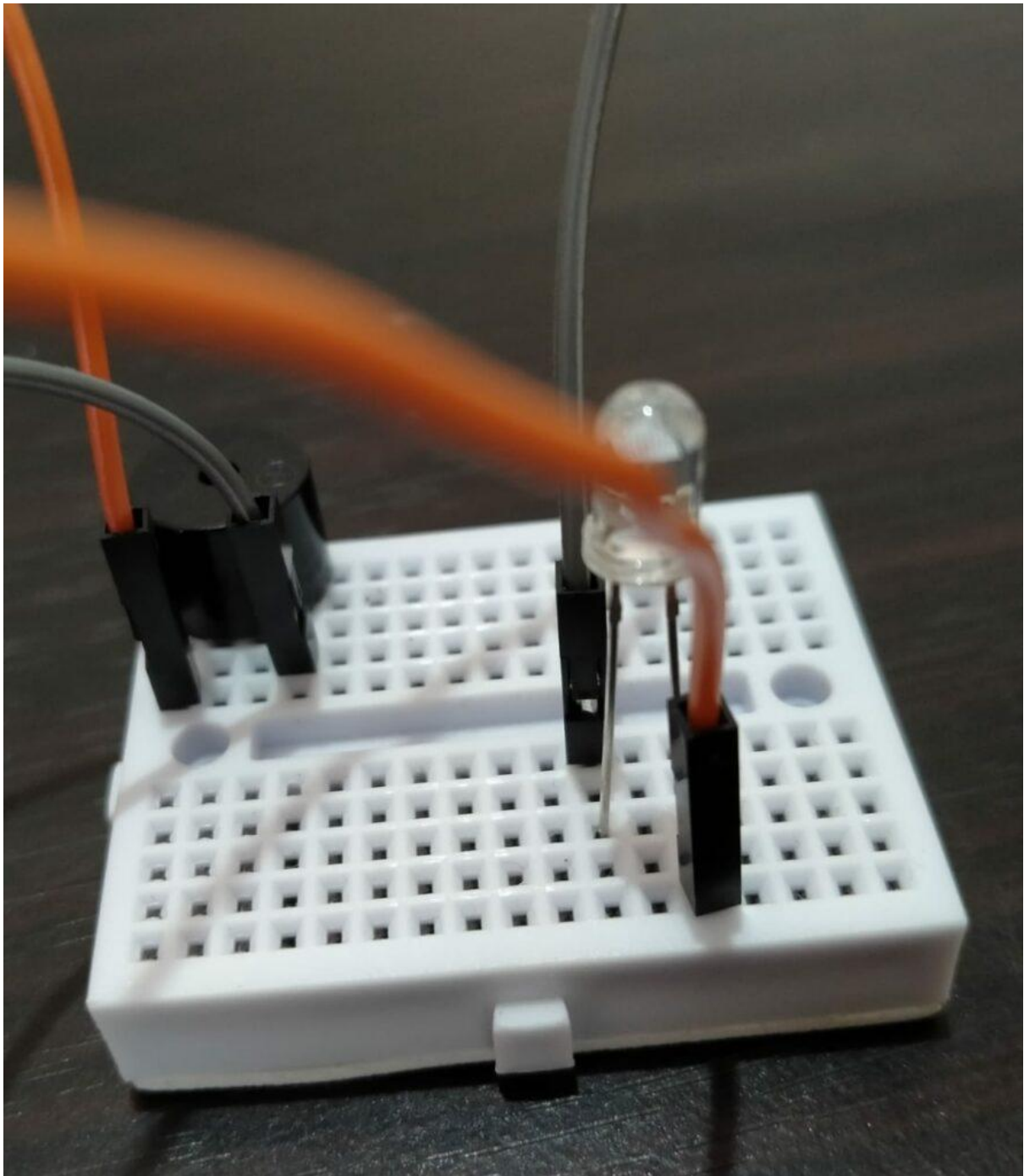
Let Start With the circuit of the breadboard, Now Take Buzzer and fix it on the breadboard knowing which end is positive and which negative, larger is positive and smaller negative.

Step 2



Now, the second step fix the LED at a distance away from the buzzer for ease of handle, noting its positive and negative end similar to the buzzer

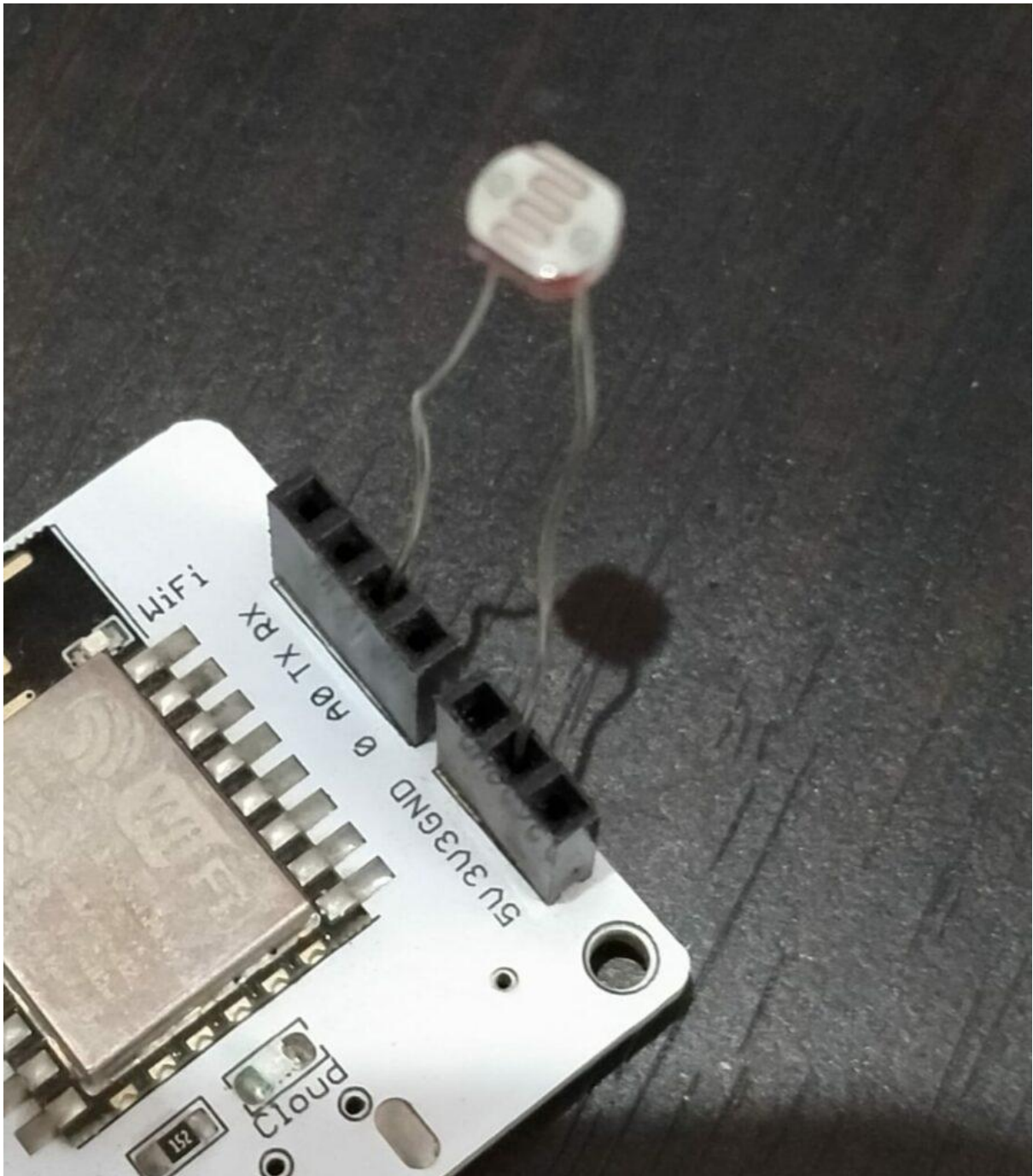
Step 3



Now, Take two jumper wires and connect the positive end of a buzzer with the positive end of the LED as shown in the circuit diagram above, DO a similar connection with the negative end of the Buzzer and LED(The orange wire connecting the negative end and gray positive end.)

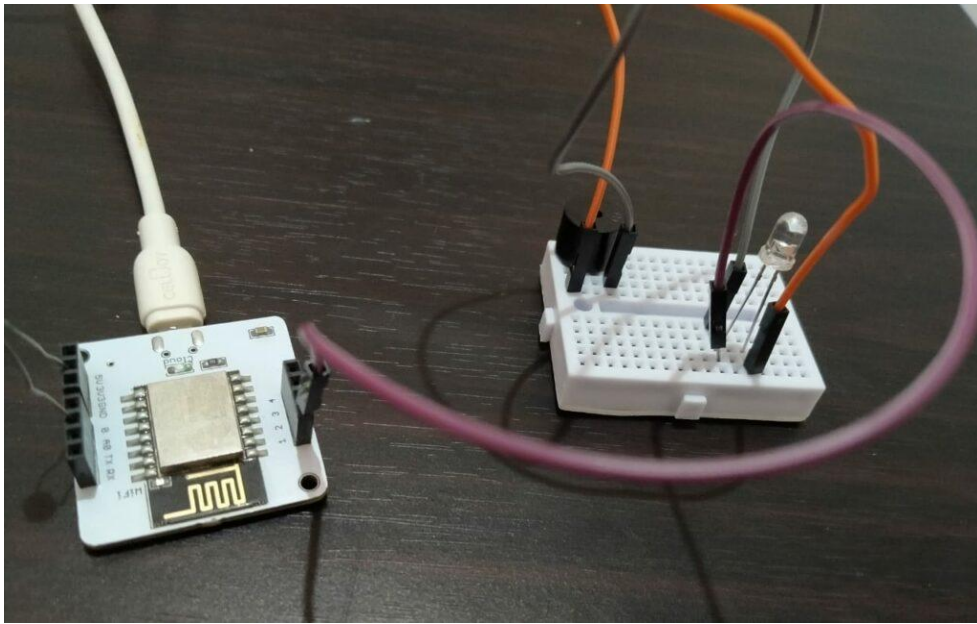
Step 4



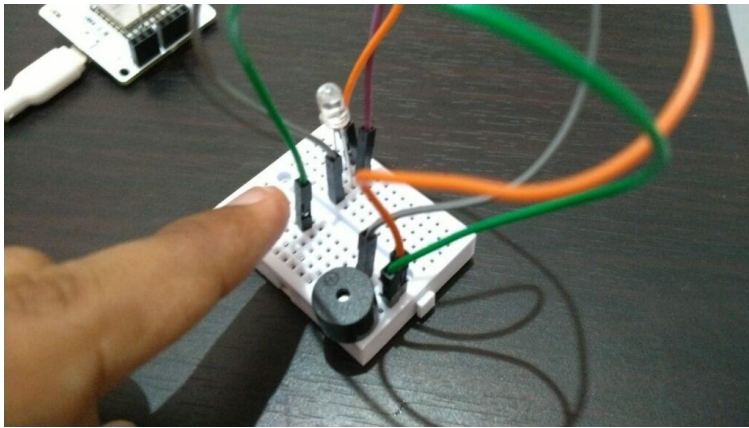


Now, Take the Boltiot wifi module and connect the LDR sensor to it, one leg of the sensor in '3v3' and the other to 'A0'.

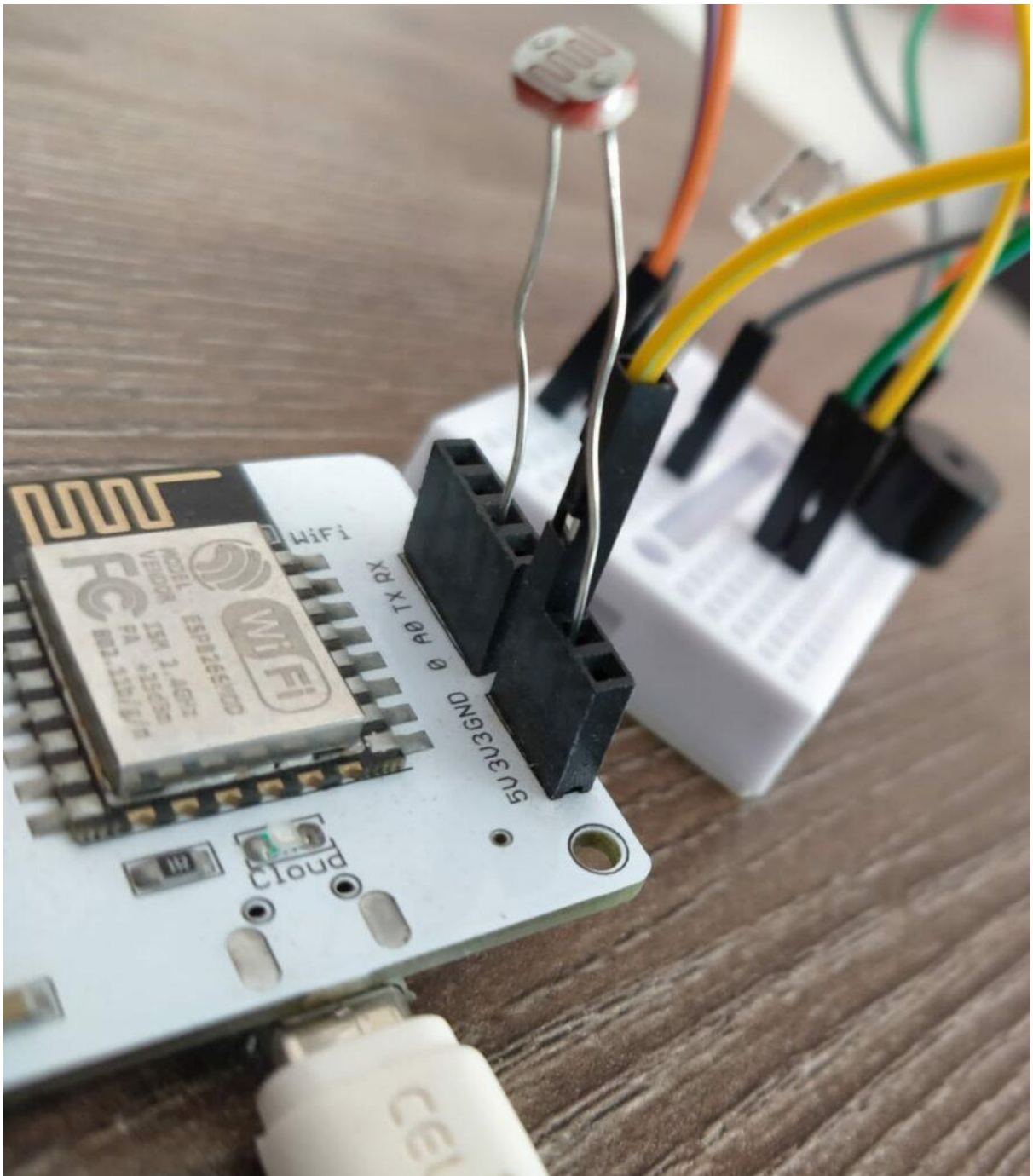
Step 5





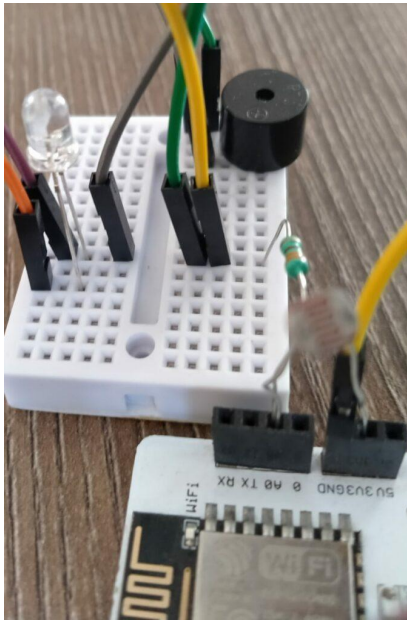


2



Now, Take jumper wires to connect the breadboard circuit to the wifi module, use the first wire(Purple) to connect the negative end of a buzzer with the 'GND' ground of the wifi module and other wire(Green + yellow) to connect the positive end of LED with pin '1' of wifi module. follow the above diagram as number 1,2,3.

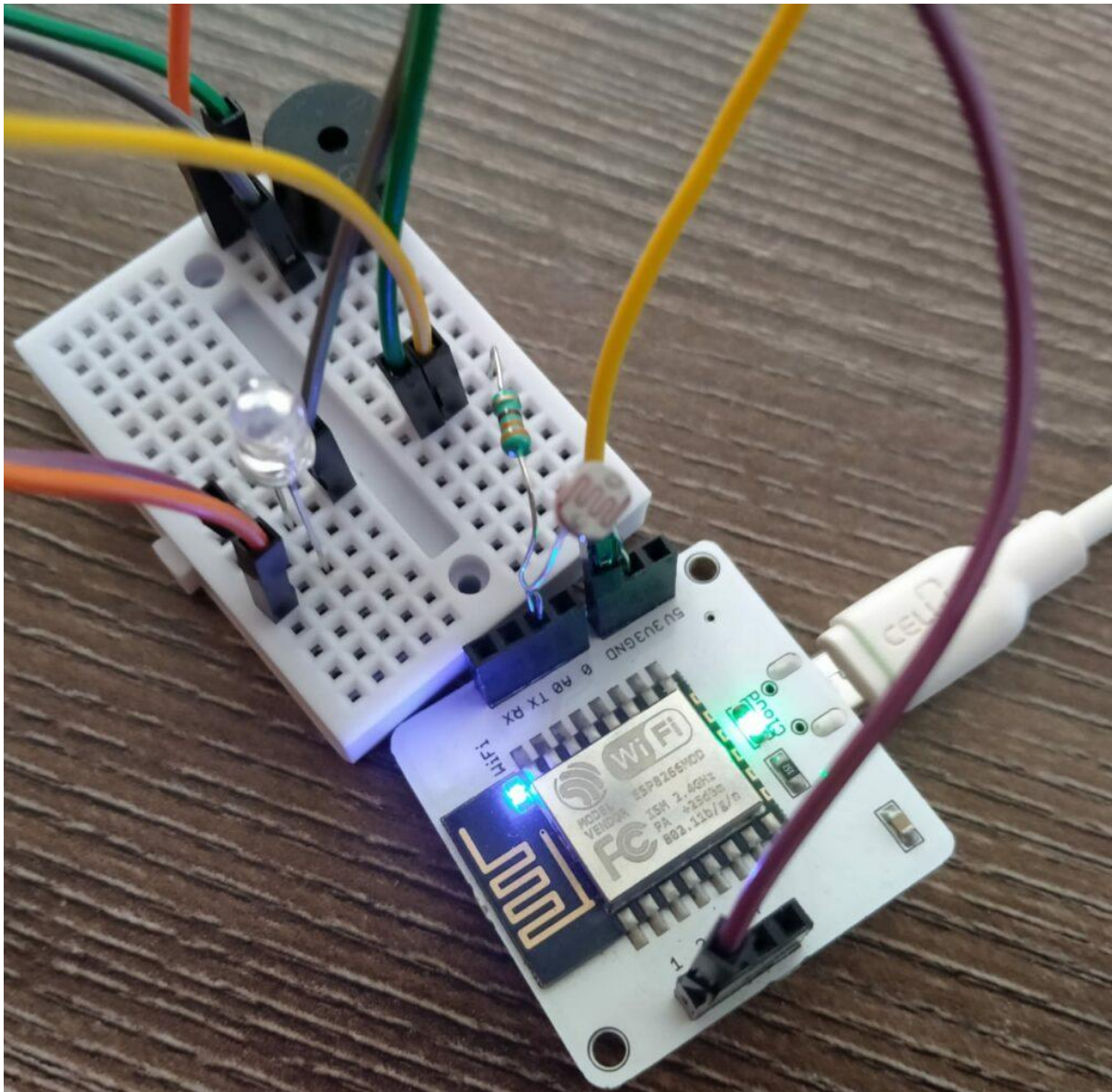
#### Step 6



Now, take a 10 k ohm resistor and connect one end in 'A0' of wifi and the second to the negative end of the buzzer

#### Step 7

finally, connect the boltiot wifi module to power with the help of a USB cable.



## Step 8

Now, for SMS alert we need to create a Twilio account.

## TWILIO SERVICE

Twilio is an American cloud communications platform as a service company based in San Francisco, California. Twilio allows software developers to programmatically make and receive phone calls, send and receive text messages, and perform other communication functions using its web service APIs.

We will be using these as our SMS service provider that will send SMS alerts on unusual activity in the locker.

For the step to create a Twilio Account click on the link below



[https://docs.google.com/document/d/1u5YF4JIOMBdw44fKNnJv42WR\\_uFdDhEJnJlpucLY49o/edit?usp=sharing](https://docs.google.com/document/d/1u5YF4JIOMBdw44fKNnJv42WR_uFdDhEJnJlpucLY49o/edit?usp=sharing)

we will be needing 'SID', 'Auth Token', ' Trial number' that we will get after the account on Twilio.

#### Step 9

Now, we need to write code to detect abnormality in light intensity of locker, we are doing this using the 'LDR' sensor and z score analysis technique, and if an anomaly is detected then it will trigger light of room and SMS service of Twilio.

A **Z-score** is a numerical measurement that describes a value's relationship to the mean of a group of values. **Z-score** is measured in terms of standard deviations from the mean. If a **Z-score** is 0, it indicates that the data point's **score** is identical to the mean **score**.

## Software Programming

we will be writing our code in ubuntu in VMware virtual machine, we need to create two files first conf.py and second anomaly\_detection.py.

```
//Conf file for storing the detail of Twilio account and api_key, id of bolt device //
SSID = 'You can find SSID in your Twilio Dashboard after you          created the account'
AUTH_TOKEN = 'You can find  on your Twilio Dashboard'
FROM_NUMBER = 'This is the no. generated by Twilio. You can find this on your Twilio
Dashboard'
TO_NUMBER = 'This is your number. Make sure you are adding +91 in beginning'
API_KEY = 'This is your Bolt Cloud account API key'
DEVICE_ID = 'This is the ID of your Bolt device'
FRAME_SIZE = 10
MUL_FACTOR = 6
```

### Code of anomaly\_detection.py

```

# code of Smart locker with LED and Buzzer trigger
import conf, json,time,math,statistics # 'math' and 'statistics' library for calculating mean and
from boltiot import Sms,Bolt # 'sms' and 'bolt' from boltiot for sms service and device

def compute_bounds(history_data,frame_size,factor): #function to calculate 'High' and 'Low' bound as
    if len(history_data)<frame_size:
        return None
    if len(history_data)>frame_size:
        del history_data[0:len(history_data)-frame_size] #Deleting access data from history_data[]

    Mn=statistics.mean(history_data) #Taking Mean of history data
    Variance=0
    for data in history_data:
        Variance+=math.pow((data-Mn),2) #Calculating Variance
    Zn=factor*math.sqrt(Variance/frame_size) #Z score
    High_bound =history_data[frame_size-1]+Zn
    Low_bound=history_data[frame_size-1]-Zn
    return [High_bound,Low_bound] # Returning High and Low bound

mybolt=Bolt(conf.API_KEY,conf.DEVICE_ID) # Taking Detail of device from conf file created before
sms= Sms(conf.SSID,conf.AUTH_TOKEN,conf.TO_NUMBER,conf.FROM_NUMBER) # Twilio Detail to sms variable
history_data=[] # Creting history_data Array

while True:
    response=mybolt.analogRead('A0') #Reading LDR reading
    data=json.loads(response)
    if data['success']!=1:
        print("There was an error while retriving the data.")
        print("This is the error:"+data['value'])
        time.sleep(5)
        continue
    print("This is the vallue "+data['value'])

```

```

        return [High_bound,Low_bound] # Returning High and Low bound

mybolt=Bolt(conf.API_KEY,conf.DEVICE_ID) # Taking Detail of device from conf file created before
sms= Sms(conf.SSID,conf.AUTH_TOKEN,conf.TO_NUMBER,conf.FROM_NUMBER) # Twilio Detail to sms variable
history_data=[] # Creting history_data Array

while True:
    response=mybolt.analogRead('A0') #Reading LDR reading
    data=json.loads(response)
    if data['success']!=1:
        print("There was an error while retriving the data.")
        print("This is the error:"+data['value'])
        time.sleep(5)
        continue
    print("This is the vallue "+data['value'])
    sensor_value=0
    try:
        sensor_value=int(data['value'])
    except e:
        print("There was an error while parsing the response:",e)
        continue
    bound=compute_bounds(history_data,conf.FRAME_SIZE,conf.MUL_FACTOR) # Calling compute_bounds fun
    if not bound:
        required_data_count=conf.FRAME_SIZE-len(history_data)
        print("Not enough data to compute Z-score. Need ",required_data_count," more data points")
        history_data.append(int(data['value']))
        time.sleep(10)
        continue
    try:
        if sensor_value>bound[0]:
            response=mybolt.digitalWrite('1', 'HIGH' ) #Triggering GPIO '1' to turn on LED and Buzz
            print(response)

```



```

        sensor_value=int(data['value'])
    except e:
        print("There was an error while parsing the response:",e)
        continue
    bound=compute_bounds(history_data,conf.FRAME_SIZE,conf.MUL_FACTOR) # Calling compute_bounds func
    if not bound:
        required_data_count=conf.FRAME_SIZE-len(history_data)
        print("Not enough data to compute Z-score. Need ",required_data_count," more data points")
        history_data.append(int(data['value']))
        time.sleep(10)
        continue
    try:
        if sensor_value>bound[0]:
            response=mybolt.digitalWrite('1', 'HIGH' ) #Triggering GPIO '1' to turn on LED and Buzz
            print(response)
            print("The light level increased suddenly. Sending an SMS.")
            response=sms.send_sms("Someone unlocked your locker!!!") # Triggering sms Alert through
            print("This is the reponse",reponse)
        elif sensor_value<bound[1]:
            print ("The light level decreased suddenly. Sending an SMS.")
            response=sms.send_sms("Someone locked your locker!!!")
            print("This the the response",response)
        history_data.append(sensor_value);
    except Exception as e:
        print ("Error",e)
    time.sleep(4) # rest in between

```

Before writing this code you need to set up the bolt python library,if you have not done it follow this command in the terminal.

```

sudo apt-get -y update
sudo apt install python3-pip
sudo pip3 install boltiot

```

for running the code type command "sudo python3 yourfilename.py" and it will start taking intensity reading, it will take minimum reading equal to frame\_size before operation and then will send an SMS alert and buzzer sound if it detects an anomaly.


## Code To stop Buzzer and led after the operation

This code will be written in bolt cloud so that we can access the "off" button through the mobile app.


let first create a new product and type the name "smart locker" in the bolt cloud.

Secondly, click on configure icon and come to the code section, and type below code with 'js' extension and 'locker' as the file name.

Save it, link it with your Bolt wifi module.



Products: Setup

 Smart\_locker  
Output device • Connected with GPIO

HardwareCode

Write your code in the code window below.

lockerjs

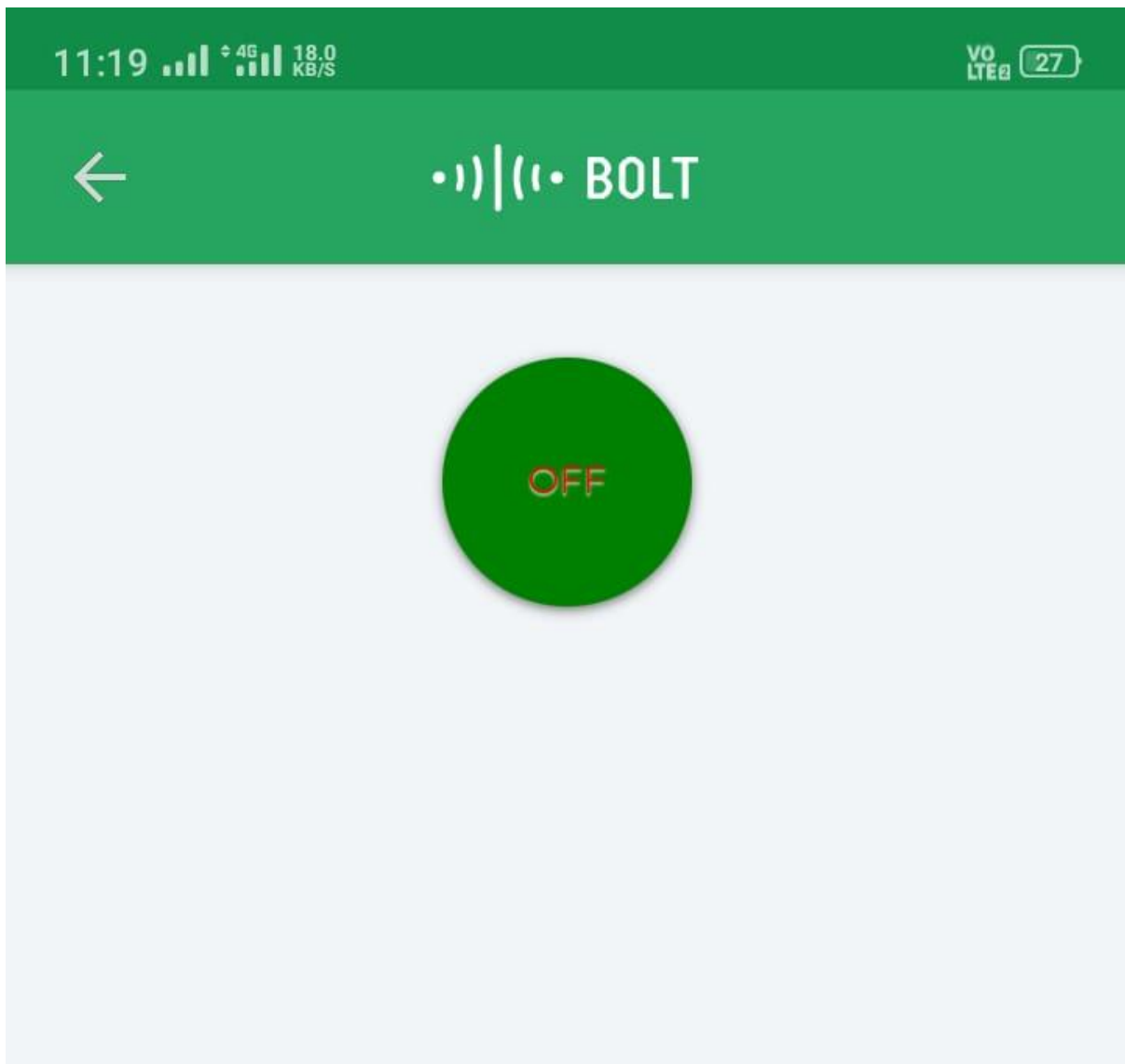
Start typing your code below.

Feedback

```
1 singleButton({name:" OFF ",action:"digitalWrite",pin:"1",value:"LOW",bgcolor:"green",shape:"circle",align:"center",text_color:"red"})
2
```

A setup in bolt cloud

```
singleButton({name:" OFF
",action:"digitalWrite",pin:"1",value:"LOW",bgcolor:"green",shape:"circle",align:"center",text_color:"red"})
```



This is the screenshot of the bolt mobile App

## Conclusion

With a Smart locker in your home, you can sleep carefree as with a Smart locker high pitch buzzer sound and SMS Alert are there to Alert you!!

Thank you reader for going through my project share and apply if you liked it.