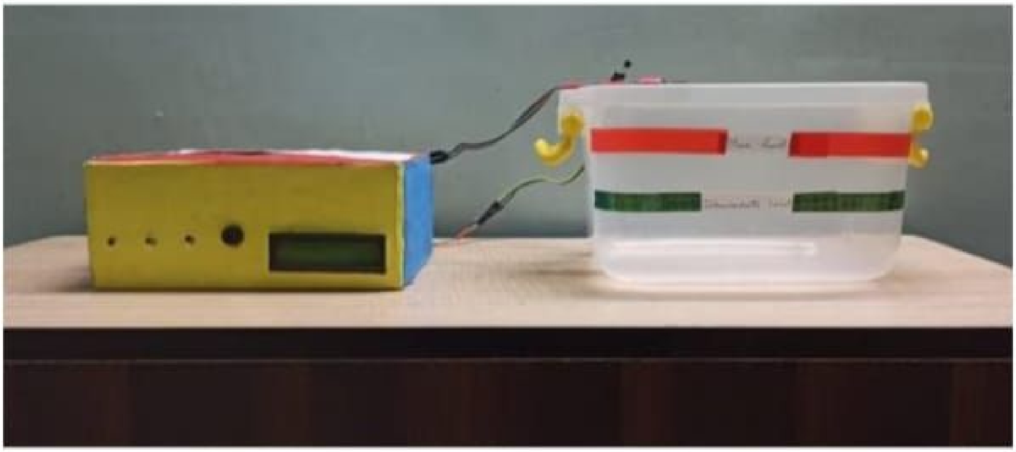
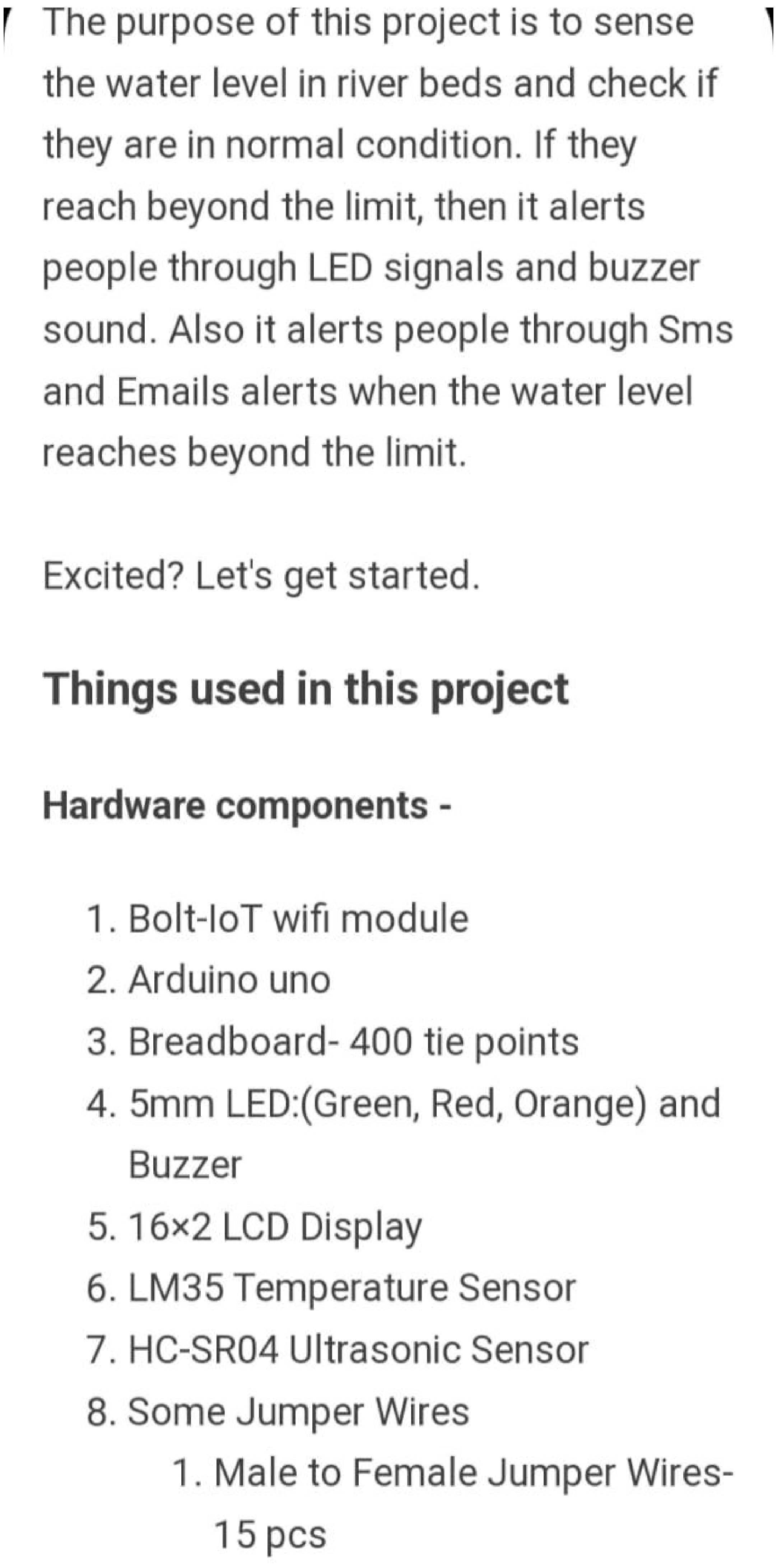
IoT Based Flood Monitoring And Alerting System

Written by Vaibhav Shanbhag | Feb 23, 2021 4:19:26 PM



As we all know that Flood is one of the major well known Natural Disasters. When water level suddenly rises in dams, river beds etc. Alot of Destruction happens at surrounding places. It causes a huge amount of loss to our environment and living beings as well. So in these case, it is very important to get emergency alerts of the water level situation in different conditions in the river bed.



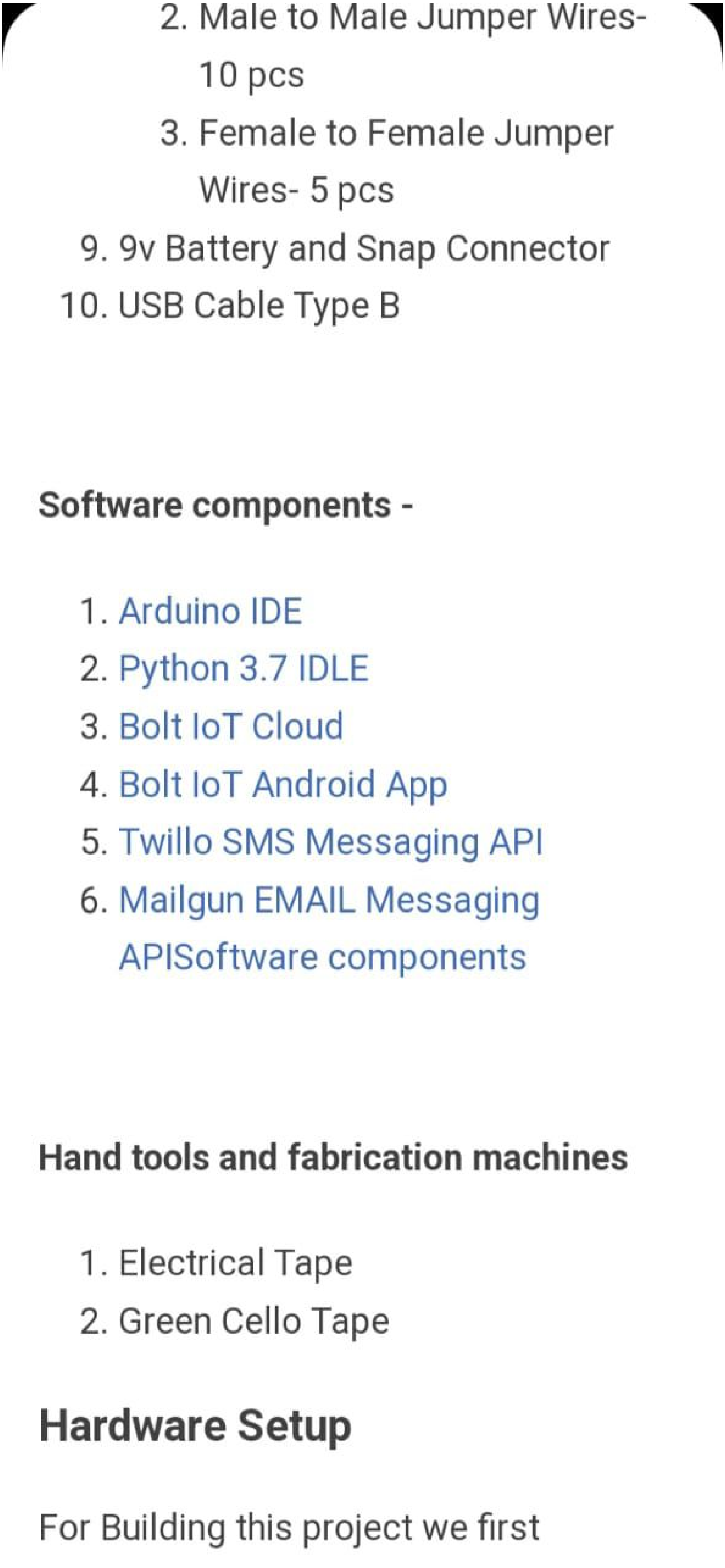
The purpose of this project is to sense the water level in river beds and check if they are in normal condition. If they reach beyond the limit, then it alerts people through LED signals and buzzer sound. Also it alerts people through Sms and Emails alerts when the water level reaches beyond the limit.

Excited? Let's get started.

Things used in this project

Hardware components -

1. Bolt-IoT wifi module
2. Arduino uno
3. Breadboard- 400 tie points
4. 5mm LED:(Green, Red, Orange) and Buzzer
5. 16×2 LCD Display
6. LM35 Temperature Sensor
7. HC-SR04 Ultrasonic Sensor
8. Some Jumper Wires
    1. Male to Female Jumper Wires- 15 pcs



   2. Male to Male Jumper Wires- 10 pcs
   3. Female to Female Jumper Wires- 5 pcs
9. 9v Battery and Snap Connector
10. USB Cable Type B

Software components -

1. Arduino IDE
2. Python 3.7 IDLE
3. Bolt IoT Cloud
4. Bolt IoT Android App
5. Twillo SMS Messaging API
6. Mailgun EMAIL Messaging APISoftware components

Hand tools and fabrication machines

1. Electrical Tape
2. Green Cello Tape

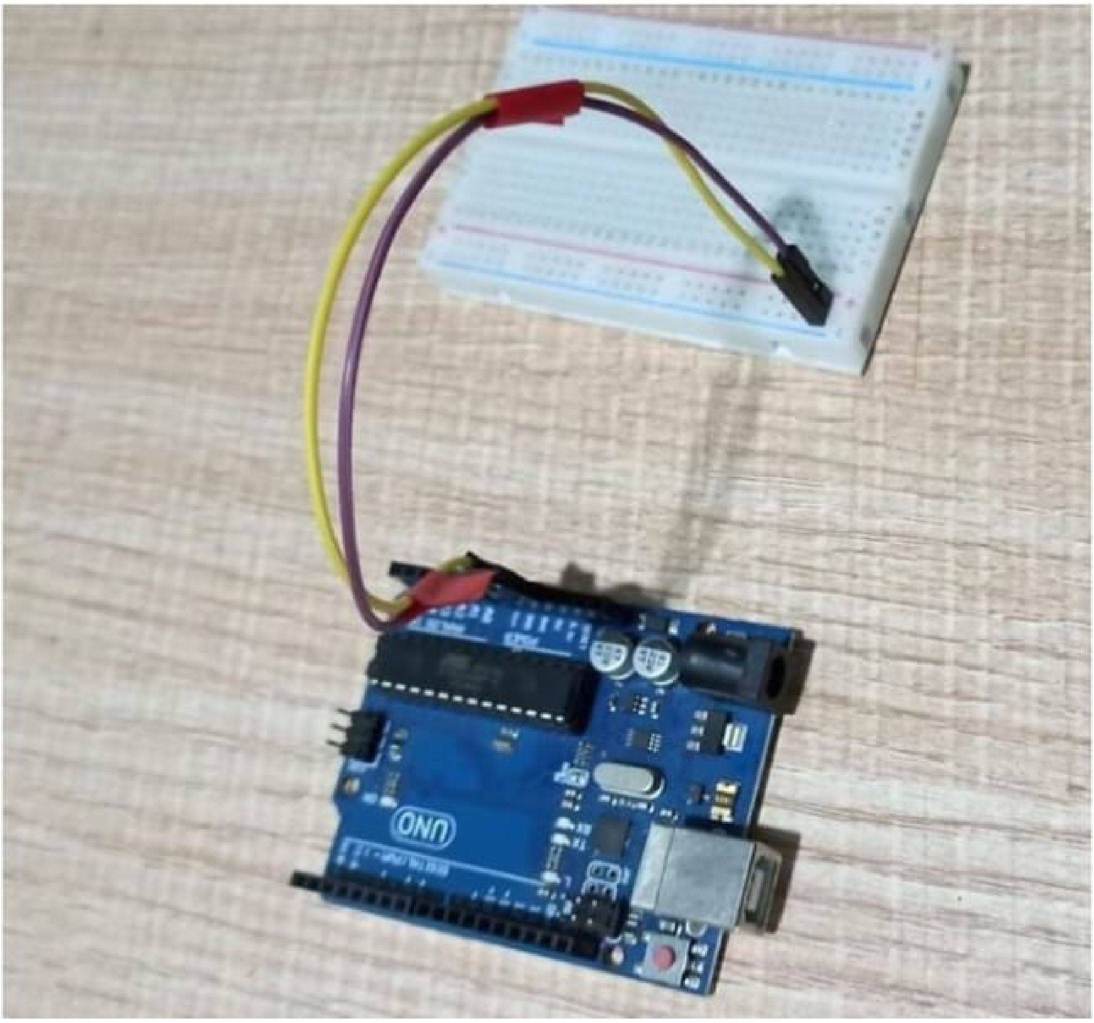
Hardware Setup

For Building this project we first

Hardware Setup

For Building this project we first configure the hardware connections. Then later on moving to the software part.

Step 1: Connecting 5v and GND of Arduino to the Breadboard for power connection to other components.



Step 2: Connecting LED’s

For Green LED:

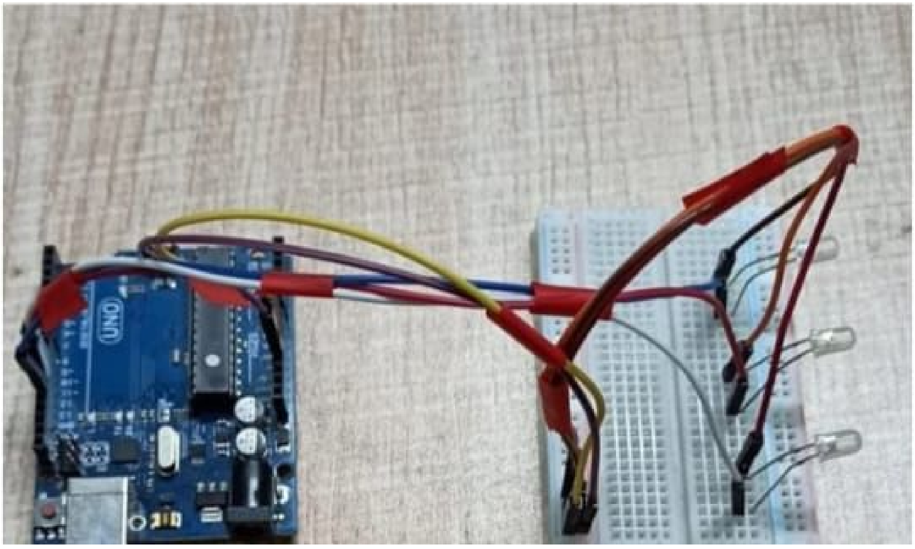
- VCC of Green Colour LED to Digital Pin ‘10’ of the Arduino.
- GND of Green Colour LED to the GND of Arduino.

For Orange LED:

- VCC of Orange Colour LED to Digital Pin ‘11’ of the Arduino.
- GND of Orange Colour LED to the GND of Arduino.

For Red LED:

- VCC of Red Colour LED to Digital Pin ‘12’ of the Arduino.
- GND of Red Colour LED to the GND of Arduino.

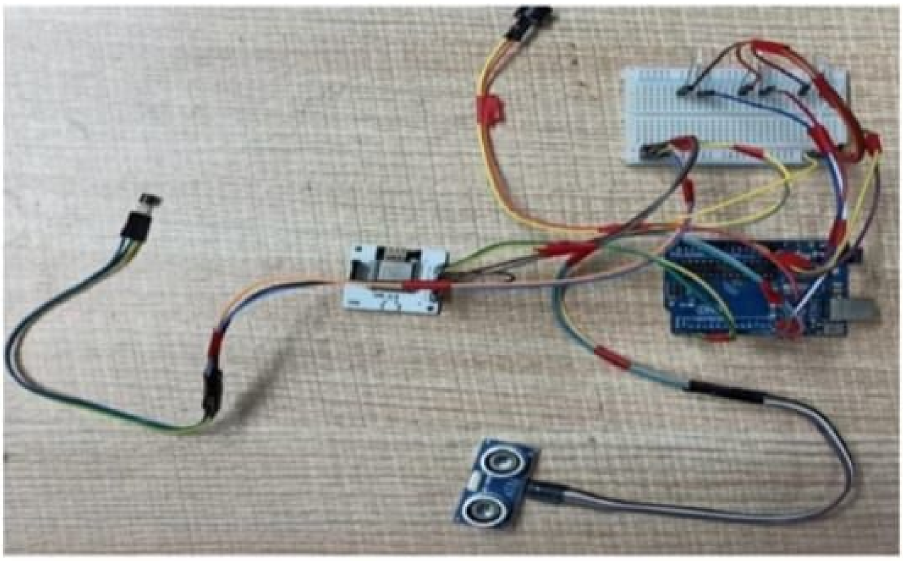


Step 5: Connecting Bolt WiFi Module

- 5v of Bolt WiFi Module to 5v of Arduino.
- GND of Bolt WiFi Module to GND of Arduino.
- TX of Bolt WiFi Module to RX of Arduino.
- RX of Bolt WiFi Module to TX of Arduino.

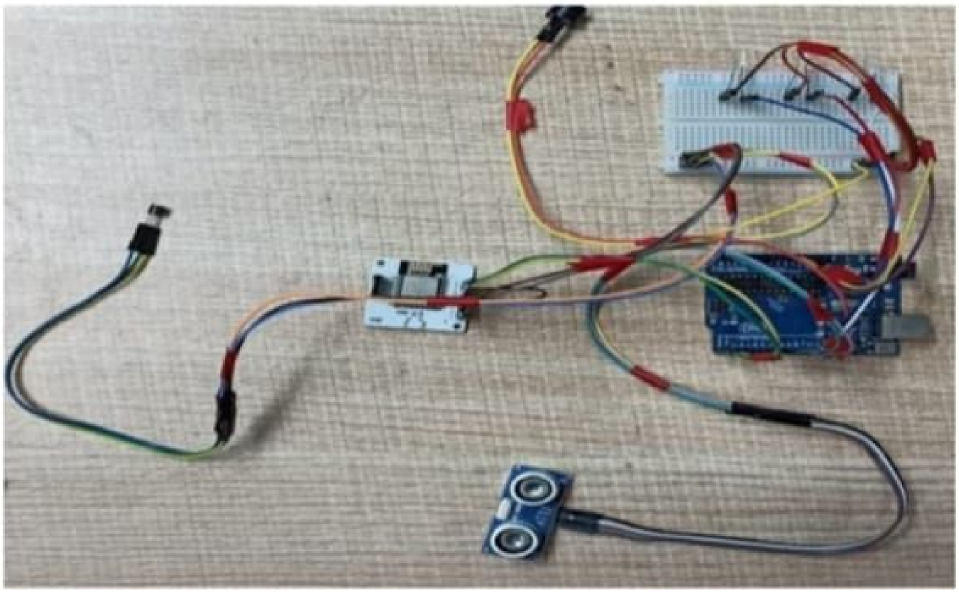
Step 6: Connecting LM35 Temperature Sensor

- VCC of LM35 to 5v of Bolt WiFi Module.
- Output Pin of LM35 to Pin ‘A0’ of Bolt WiFi Module.
- GND of LM35 to GND of Bolt WiFi Module.

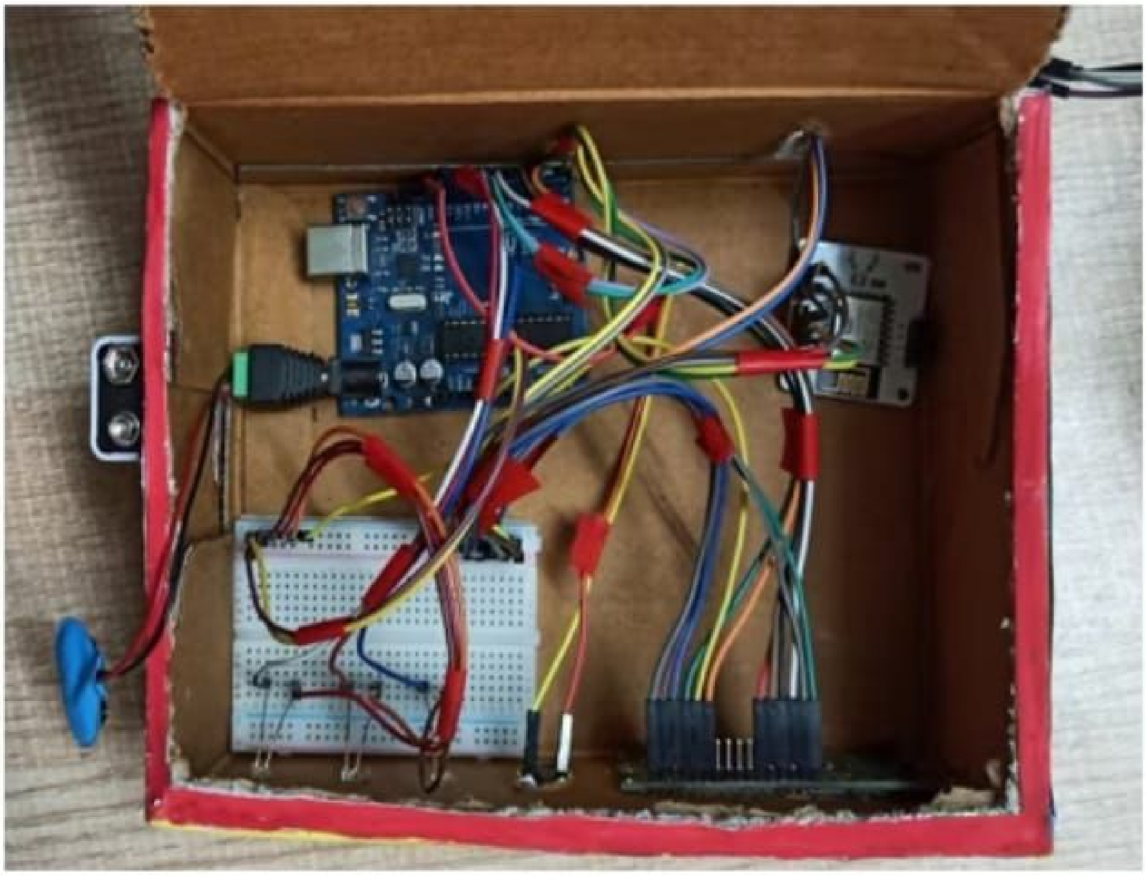


Step 7:Connecting 16×2 LCD Display

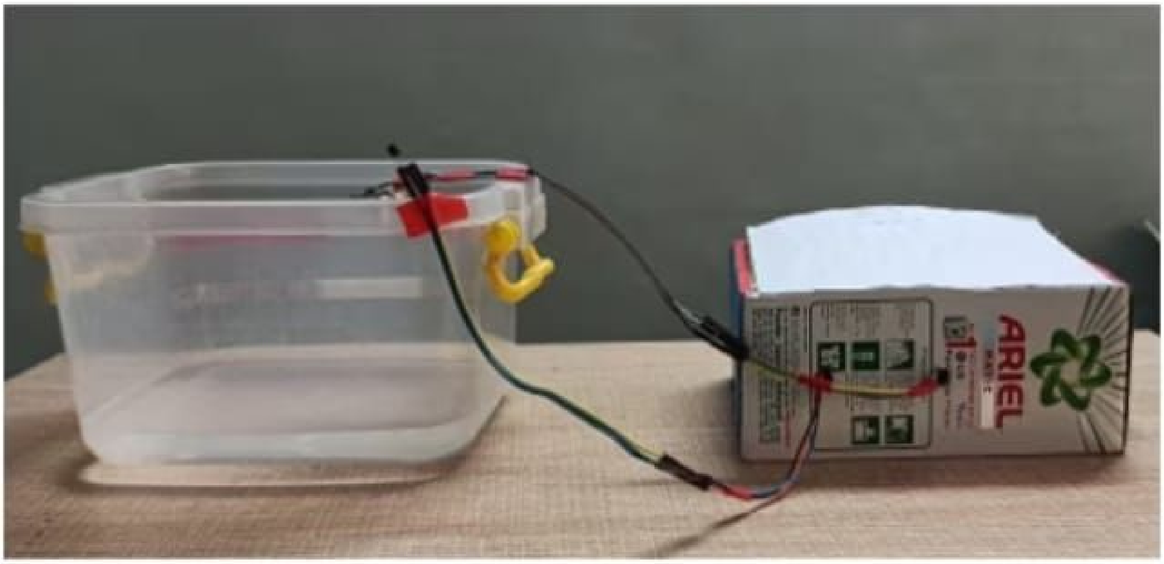
- Pin 1,3,5,16 of 16×2 LCD to GND of Arduino.
- Pin 2,15 of 16×2 LCD to 5v of Arduino.
- Pin 4 of 16×2 LCD to Digital Pin ‘2’ of Arduino.
- Pin 6 of 16×2 LCD to Digital Pin ‘3’ of Arduino.
- Pin 11 of 16×2 LCD to Digital Pin ‘4’ of Arduino.
- Pin 12 of 16×2 LCD to Digital Pin ‘5’ of Arduino.
- Pin 13 of 16×2 LCD to Digital Pin ‘6’ of Arduino.
- Pin 14 of 16×2 LCD to Digital Pin ‘7’ of Arduino.

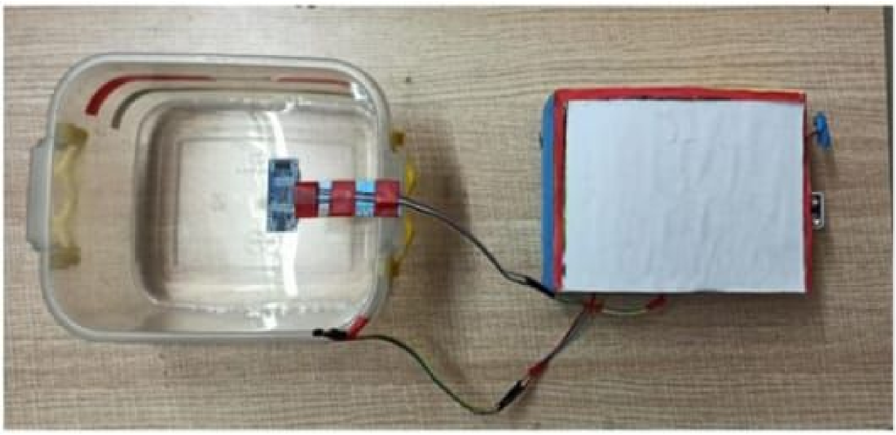


After doing the hardware connection put all the hardware components in one box.



Also attach LM35 Temperature Sensor on the side of the container.





Software Programming

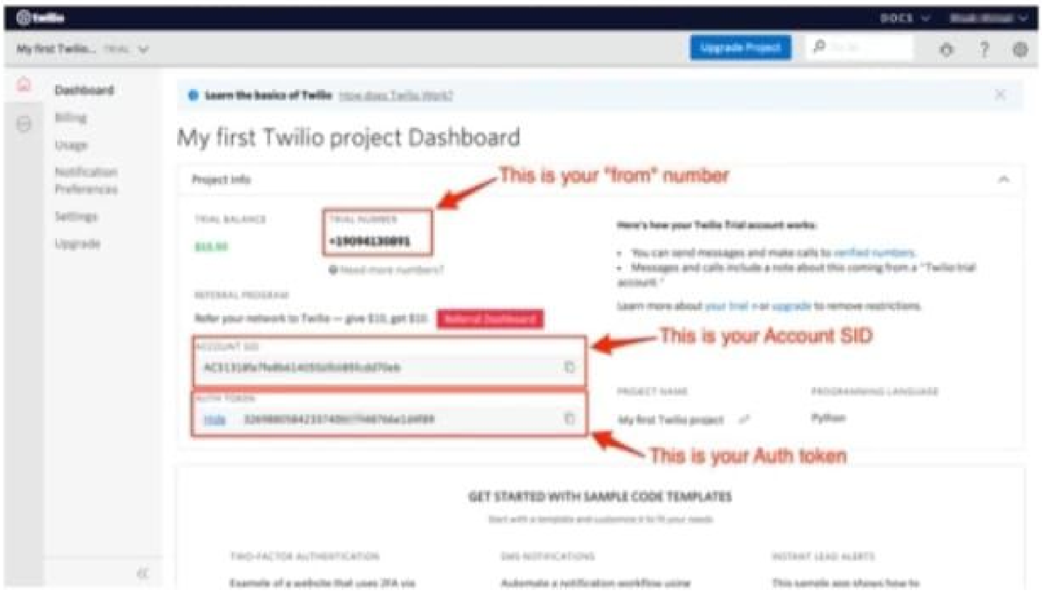
After the successful completion of hardware setup. Now it's the time to do software setup for the project. For that you have to first Download and Install Arduino IDE and Python IDE from the link given above in the software apps and online services section. Also Creating account on various online app services and noting down the important keys and id's. Below all the steps given to create account on online app services and noting down the keys.

Step 1:Creating an account on Twillo and setting up Twillo for sending Sms alerts.

- Visit https://www.twilio.com/ .

- Create account by clicking sign up, fill required details.
- Confirm your email.
- You will need to authenticate your phone number on which the sms alerts will be notified.
- Enter the code sent to your phone
- When prompted " Do you write code?" Click yes
- Select python as your programming language
- When prompted "What is your target today? "Choose" Twilio as a project.
- When prompted "What do you want to do first? "Choose" Send or receive a message.
- My First Twilio Project Dashboard page will open. Now you can Edit your Project as "My Project".
- Get a trial number and save it somewhere and then choose to use this number.
- You will see the ACCOUNT SID and AUTH TOKEN.

- We will need Account Sid, Auth Token and Trial Number of these so save them somewhere.



My first Twilio project Dashboard

Project Info

This is your "from" number

TRIAL BALANCE

TRIAL NUMBER

+19094130891

Need more numbers?

REFERRAL PROGRAM

Refer your network to Twilio — give $10, get $10 Referral Dashboard

Here's how your Twilio Trial account works:

- You can send messages and make calls to verified numbers.
- Messages and calls include a note about this coming from a "Twilio trial account."

Learn more about your trial or upgrade to remove restrictions.

ACCOUNT SID

This is your Account SID

AUTH TOKEN

Hide

This is your Auth token

PROJECT NAME

My first Twilio project

PROGRAMMING LANGUAGE

Python

GET STARTED WITH SAMPLE CODE TEMPLATES

Start with a template and customize it to fit your needs

TWO-FACTOR AUTHENTICATION

Example of a website that uses 2FA via

SMS NOTIFICATIONS

Automate a notification workflow using

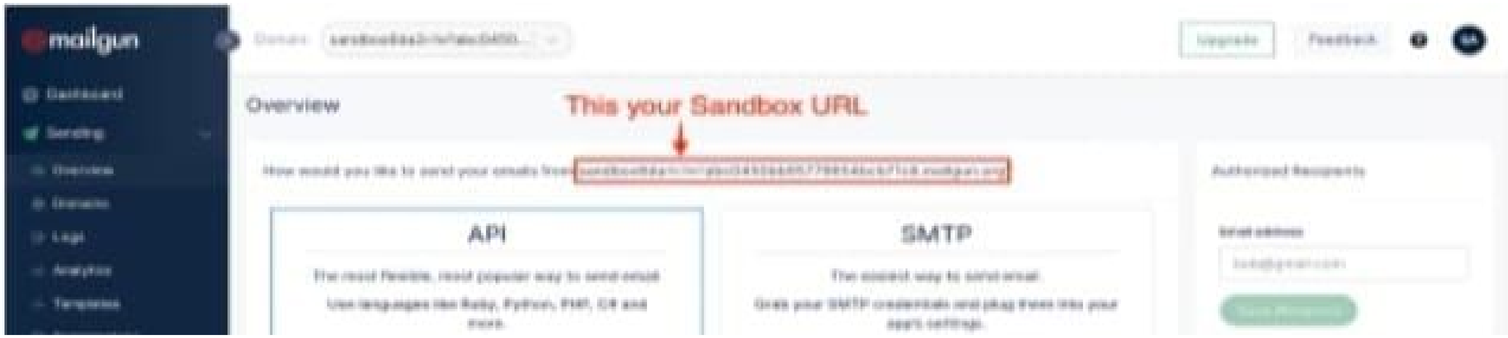
INSTANT LEAD ALERTS

This sample app shows how to

Step 2:Creating an account on Mailgun and setting up Mailgun for sending Email alerts.

- Visit https://www.mailgun.com/.
- Create an account by clicking on the start sending option and by filling up details.
- Verifying your Account.
- Once you have verified your Email after that you have add your phone number.

- After Creating account on Mailgun go to the overview option. Click on API and Click on Python.
- After doing this so you will receive API Key and Sandbox URL. Save this both credentials somewhere you will be further using in this project.



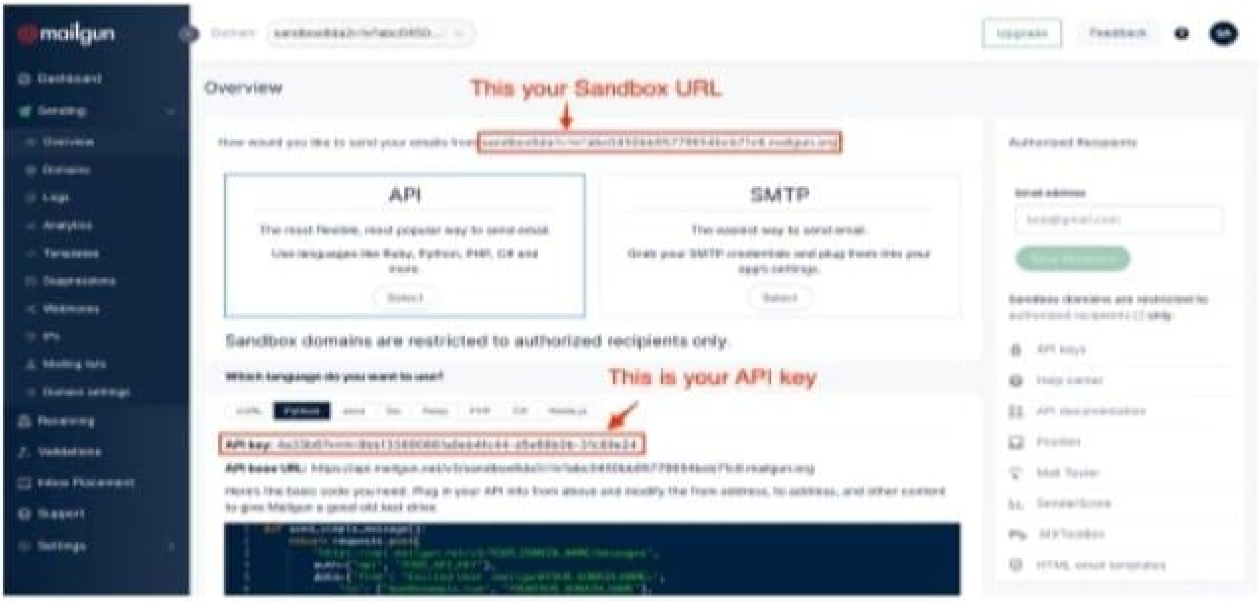
Overview

This your Sandbox URL

How would you like to send your emails from [illegible]

API: The most flexible, most popular way to send email. Use languages like Ruby, Python, PHP, C# and more.

SMTP: The easiest way to send email. Grab your SMTP credentials and plug them into your app's settings.



Overview

This your Sandbox URL

How would you like to send your emails from [illegible]

API

The most flexible, most popular way to send email.

Use languages like Ruby, Python, PHP, C# and more.

SMTP

The easiest way to send email.

Grab your SMTP credentials and plug them into your app's settings.

Sandbox domains are restricted to authorized recipients only

Which language do you want to use?

This is your API key

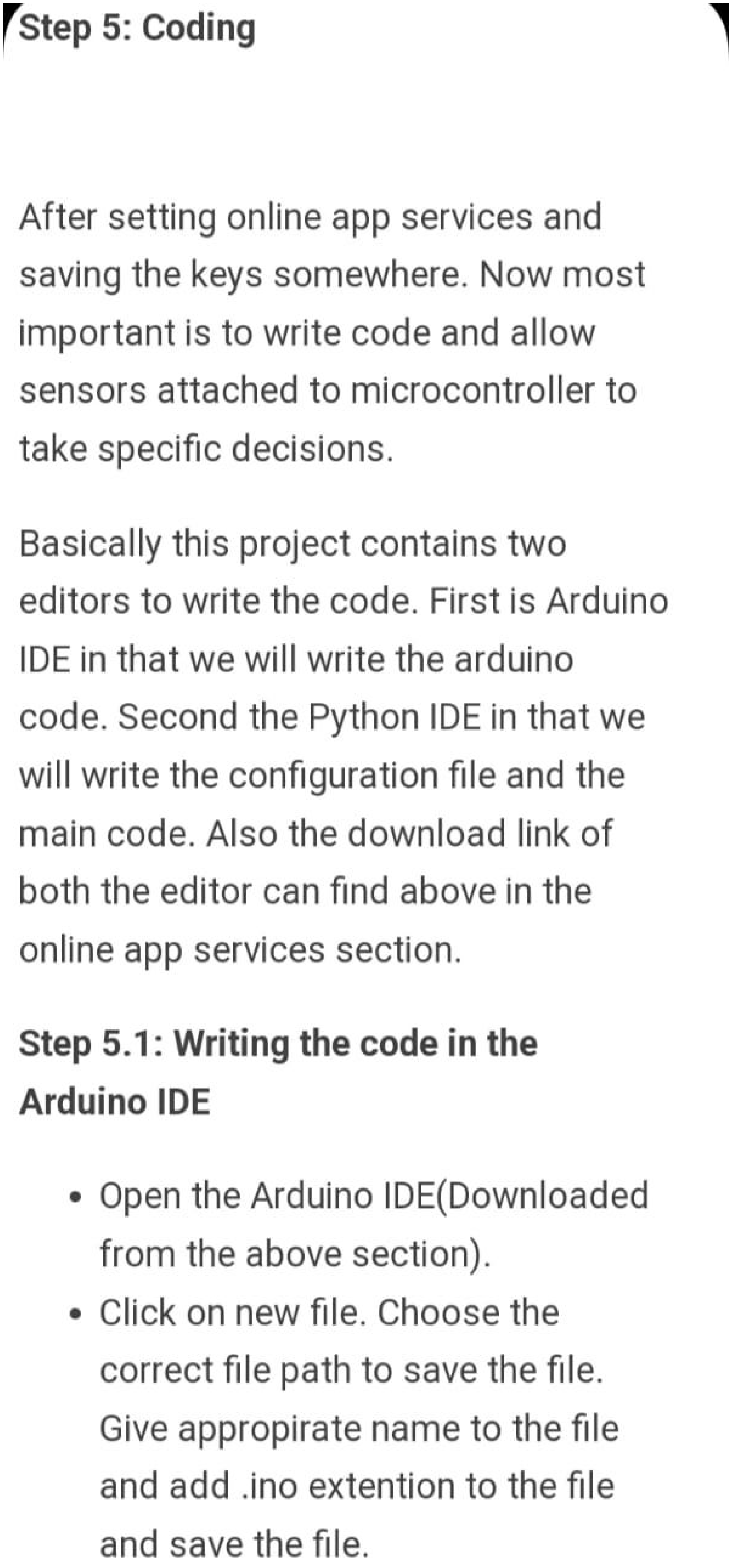
API key: [illegible]

API base URL: [illegible]

Authorized Recipients

Step 4:Creating an account on Bolt Cloud and Bolt Android App and Link the Bolt Module to Cloud.

- Visit https://cloud.boltiot.com.
- Create account using Email-Id and password.(Use the same email which was used to order hardware kit also use same email for app for linking the hardware to cloud.)
- After creating account on cloud. Then Download Bolt Android App from playstore.



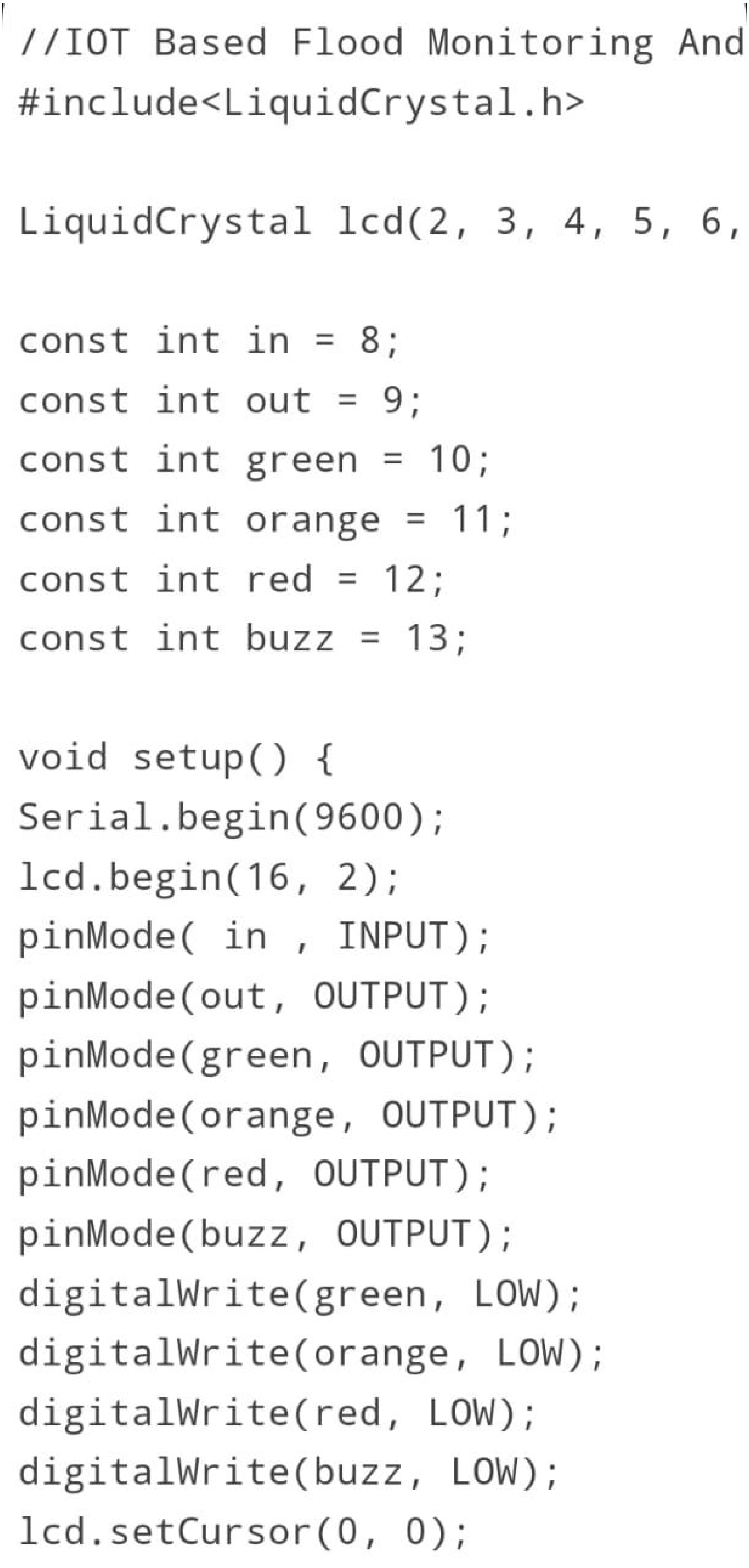
Step 5: Coding

After setting online app services and saving the keys somewhere. Now most important is to write code and allow sensors attached to microcontroller to take specific decisions.

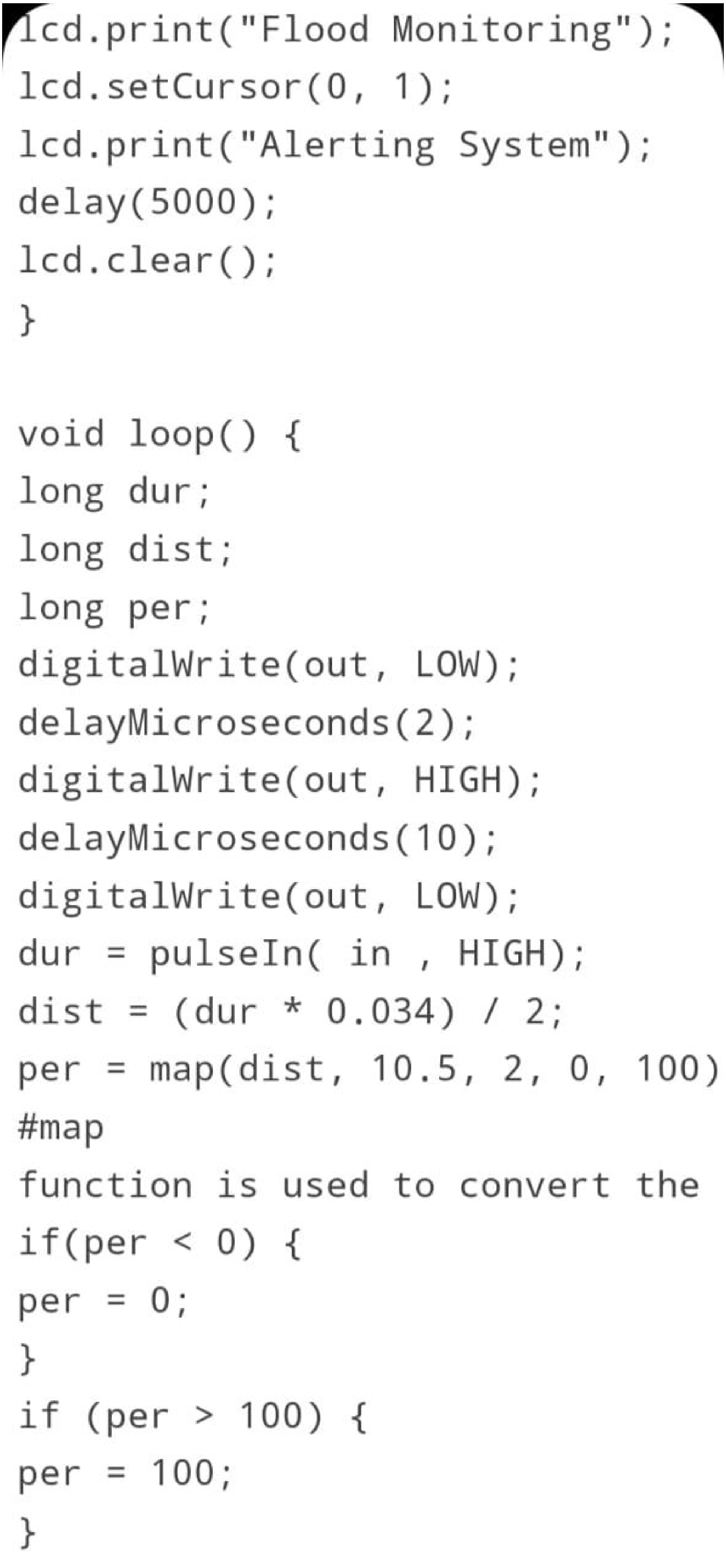
Basically this project contains two editors to write the code. First is Arduino IDE in that we will write the arduino code. Second the Python IDE in that we will write the configuration file and the main code. Also the download link of both the editor can find above in the online app services section.

Step 5.1: Writing the code in the Arduino IDE

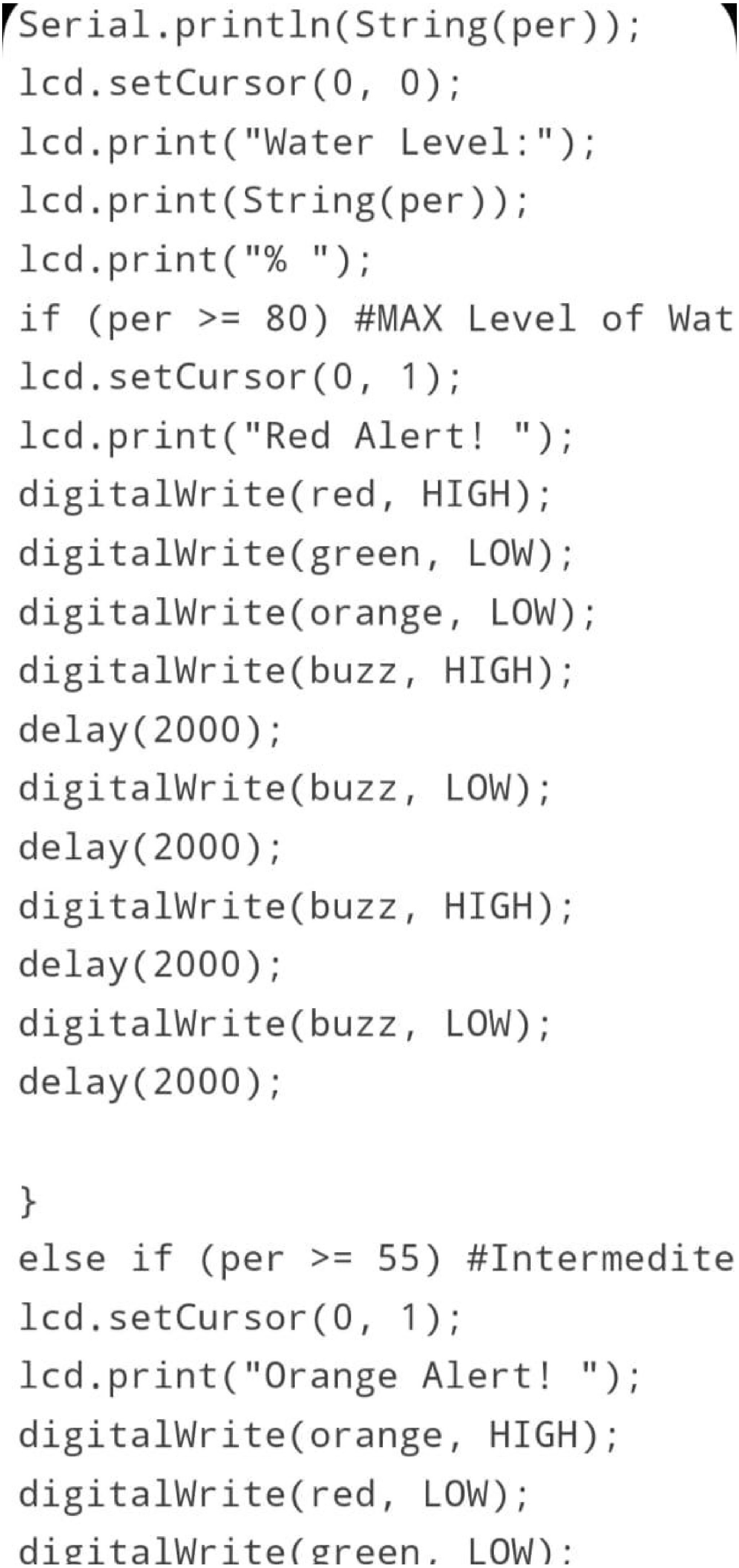
- Open the Arduino IDE(Downloaded from the above section).
- Click on new file. Choose the correct file path to save the file. Give appropirate name to the file and add .ino extention to the file and save the file.



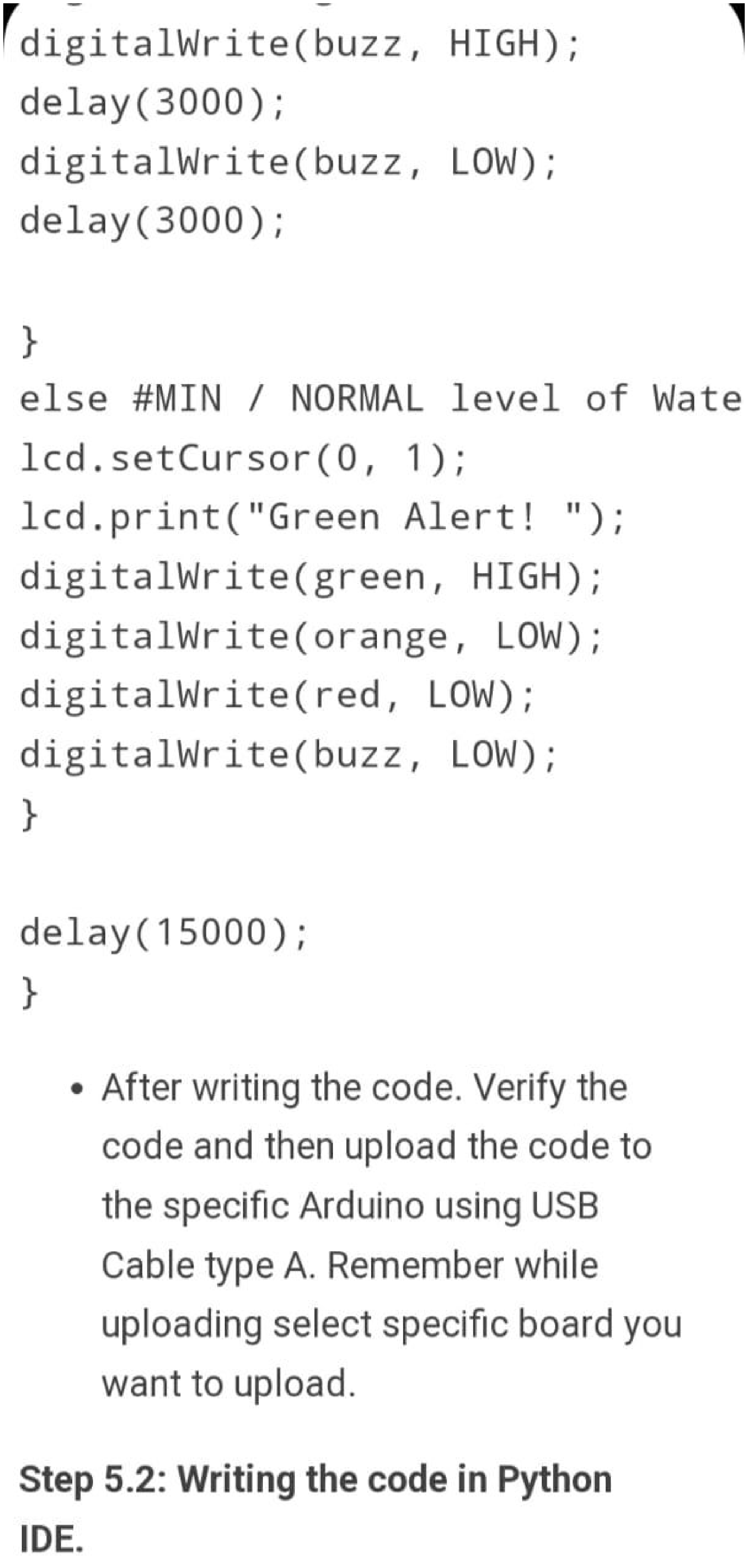
```
//IOT Based Flood Monitoring And
#include<LiquidCrystal.h>

LiquidCrystal lcd(2, 3, 4, 5, 6,

const int in = 8;
const int out = 9;
const int green = 10;
const int orange = 11;
const int red = 12;
const int buzz = 13;

void setup() {
Serial.begin(9600);
lcd.begin(16, 2);
pinMode( in , INPUT);
pinMode(out, OUTPUT);
pinMode(green, OUTPUT);
pinMode(orange, OUTPUT);
pinMode(red, OUTPUT);
pinMode(buzz, OUTPUT);
digitalWrite(green, LOW);
digitalWrite(orange, LOW);
digitalWrite(red, LOW);
digitalWrite(buzz, LOW);
lcd.setCursor(0, 0);
```



```
lcd.print("Flood Monitoring");
lcd.setCursor(0, 1);
lcd.print("Alerting System");
delay(5000);
lcd.clear();
}

void loop() {
long dur;
long dist;
long per;
digitalWrite(out, LOW);
delayMicroseconds(2);
digitalWrite(out, HIGH);
delayMicroseconds(10);
digitalWrite(out, LOW);
dur = pulseIn( in , HIGH);
dist = (dur * 0.034) / 2;
per = map(dist, 10.5, 2, 0, 100)
#map
function is used to convert the
if(per < 0) {
per = 0;
}
if (per > 100) {
per = 100;
}
```



```
Serial.println(String(per));
lcd.setCursor(0, 0);
lcd.print("Water Level:");
lcd.print(String(per));
lcd.print("% ");
if (per >= 80) #MAX Level of Wat
lcd.setCursor(0, 1);
lcd.print("Red Alert! ");
digitalWrite(red, HIGH);
digitalWrite(green, LOW);
digitalWrite(orange, LOW);
digitalWrite(buzz, HIGH);
delay(2000);
digitalWrite(buzz, LOW);
delay(2000);
digitalWrite(buzz, HIGH);
delay(2000);
digitalWrite(buzz, LOW);
delay(2000);

}
else if (per >= 55) #Intermedite
lcd.setCursor(0, 1);
lcd.print("Orange Alert! ");
digitalWrite(orange, HIGH);
digitalWrite(red, LOW);
digitalWrite(green, LOW);
```

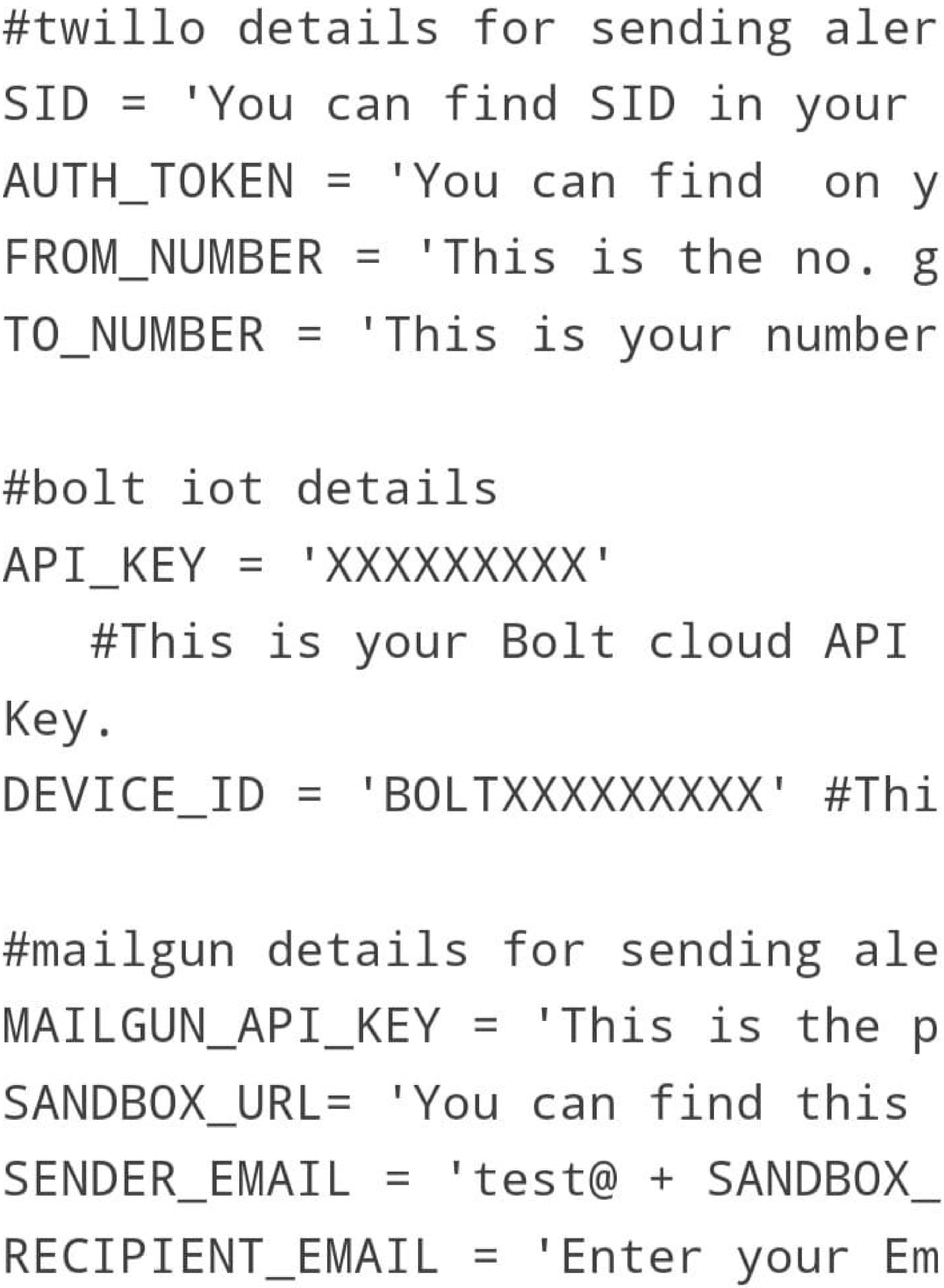


```
digitalWrite(buzz, HIGH);
delay(3000);
digitalWrite(buzz, LOW);
delay(3000);

}
else #MIN / NORMAL level of Wate
lcd.setCursor(0, 1);
lcd.print("Green Alert! ");
digitalWrite(green, HIGH);
digitalWrite(orange, LOW);
digitalWrite(red, LOW);
digitalWrite(buzz, LOW);
}

delay(15000);
}
```

- After writing the code. Verify the code and then upload the code to the specific Arduino using USB Cable type A. Remember while uploading select specific board you want to upload.

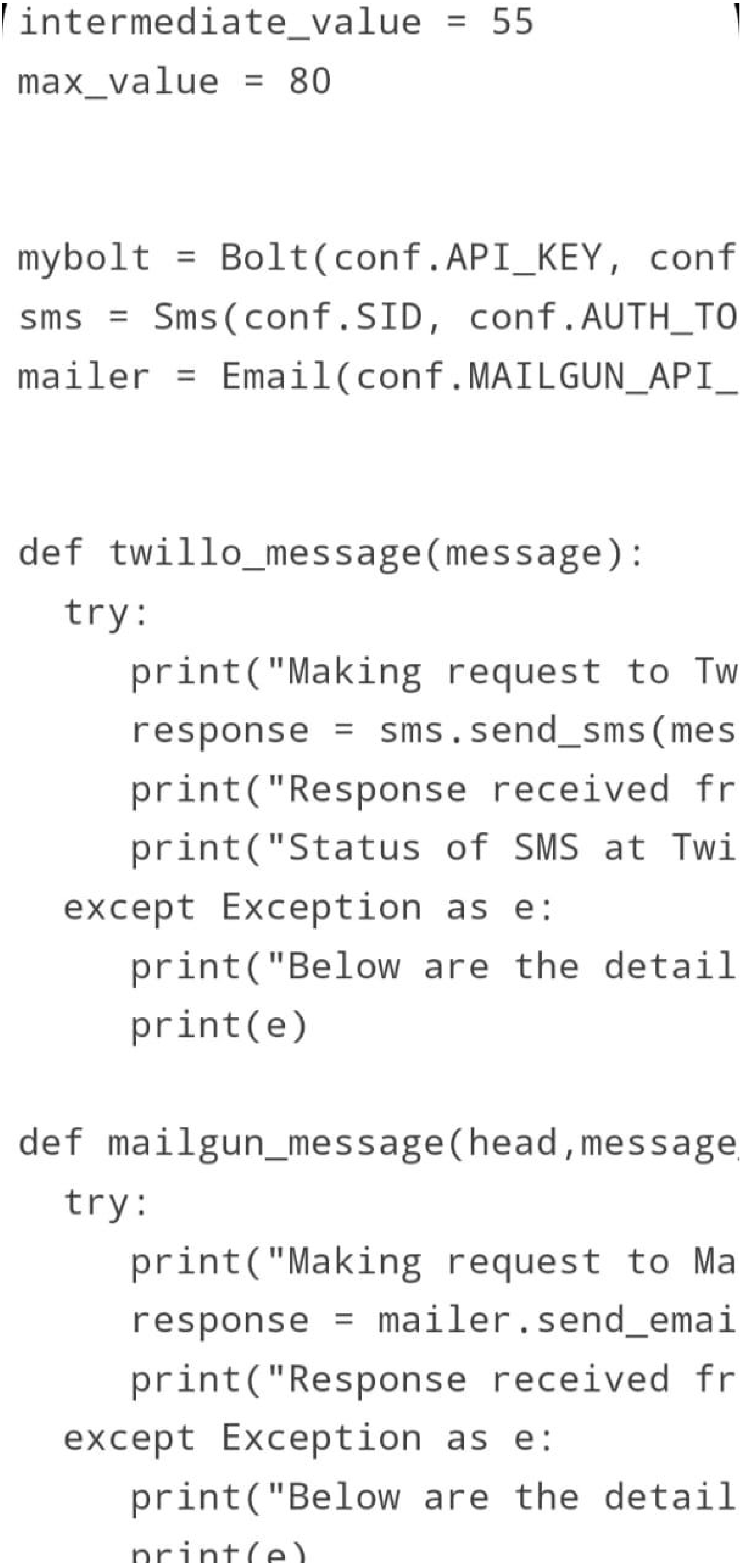
Step 5.2: Writing the code in Python IDE.

- For writing python code we will be using python IDE.
- In this project we will be making two python files. One will be saved in the name of conf.py and other will be main.py.
- The purpose of making two files is to make the code understandable. Also this both python files will be usefull in sending sms and emails alerts to users.
- Now the most important part is arrived writing code in Python IDE. The full code is divided into two parts. The detailed code is given below.
- Open Python 3.7 IDE(Downloaded from the above section).
- Click on new file. Save the file in the name conf.py

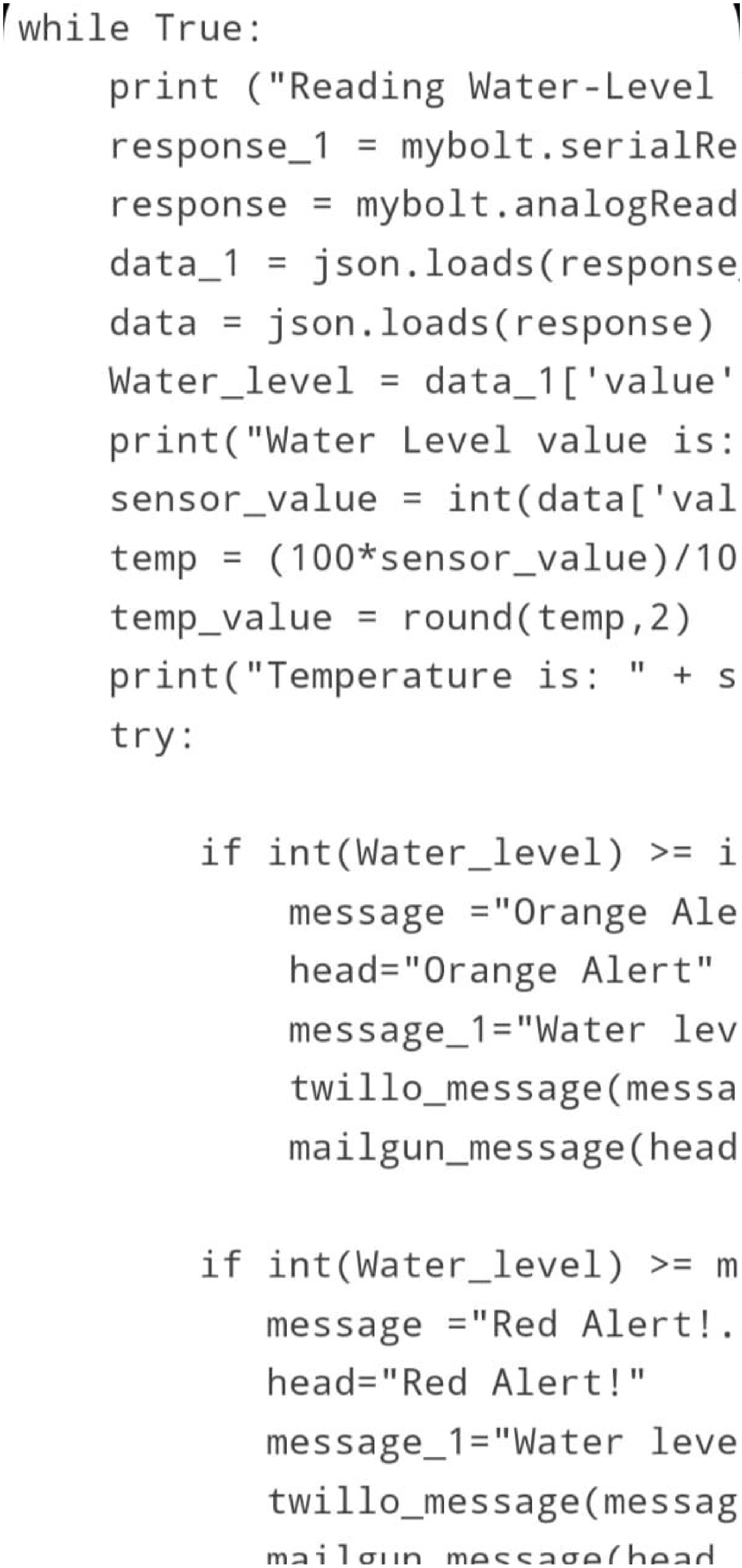


```
#twillo details for sending aler
SID = 'You can find SID in your 
AUTH_TOKEN = 'You can find  on y
FROM_NUMBER = 'This is the no. g
TO_NUMBER = 'This is your number

#bolt iot details
API_KEY = 'XXXXXXXXX'
   #This is your Bolt cloud API
Key.
DEVICE_ID = 'BOLTXXXXXXXXX' #Thi

#mailgun details for sending ale
MAILGUN_API_KEY = 'This is the p
SANDBOX_URL= 'You can find this 
SENDER_EMAIL = 'test@ + SANDBOX_
RECIPIENT_EMAIL = 'Enter your Em
```

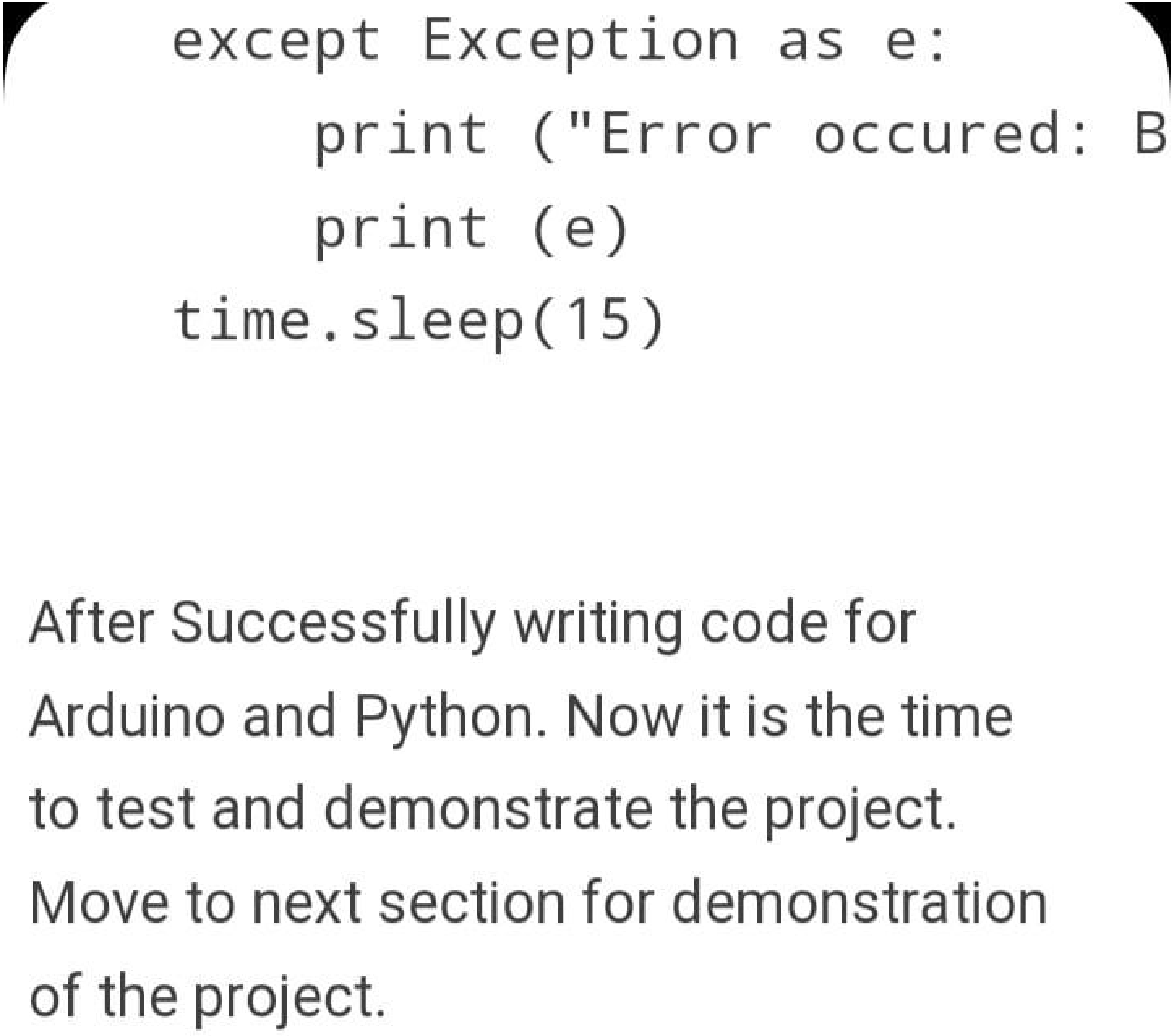
```
import conf
from boltiot import Sms, Email,
import json, time
```



```
intermediate_value = 55
max_value = 80


mybolt = Bolt(conf.API_KEY, conf
sms = Sms(conf.SID, conf.AUTH_TO
mailer = Email(conf.MAILGUN_API_


def twillo_message(message):
  try:
     print("Making request to Tw
     response = sms.send_sms(mes
     print("Response received fr
     print("Status of SMS at Twi
  except Exception as e:
     print("Below are the detail
     print(e)

def mailgun_message(head,message
  try:
     print("Making request to Ma
     response = mailer.send_emai
     print("Response received fr
  except Exception as e:
     print("Below are the detail
     print(e)
```



```
while True:
    print ("Reading Water-Level
    response_1 = mybolt.serialRe
    response = mybolt.analogRead
    data_1 = json.loads(response
    data = json.loads(response)
    Water_level = data_1['value'
    print("Water Level value is:
    sensor_value = int(data['val
    temp = (100*sensor_value)/10
    temp_value = round(temp,2)
    print("Temperature is: " + s
    try:

        if int(Water_level) >= i
            message ="Orange Ale
            head="Orange Alert"
            message_1="Water lev
            twillo_message(messa
            mailgun_message(head

        if int(Water_level) >= m
           message ="Red Alert!.
           head="Red Alert!"
           message_1="Water leve
           twillo_message(messag
           mailgun_message(head
```

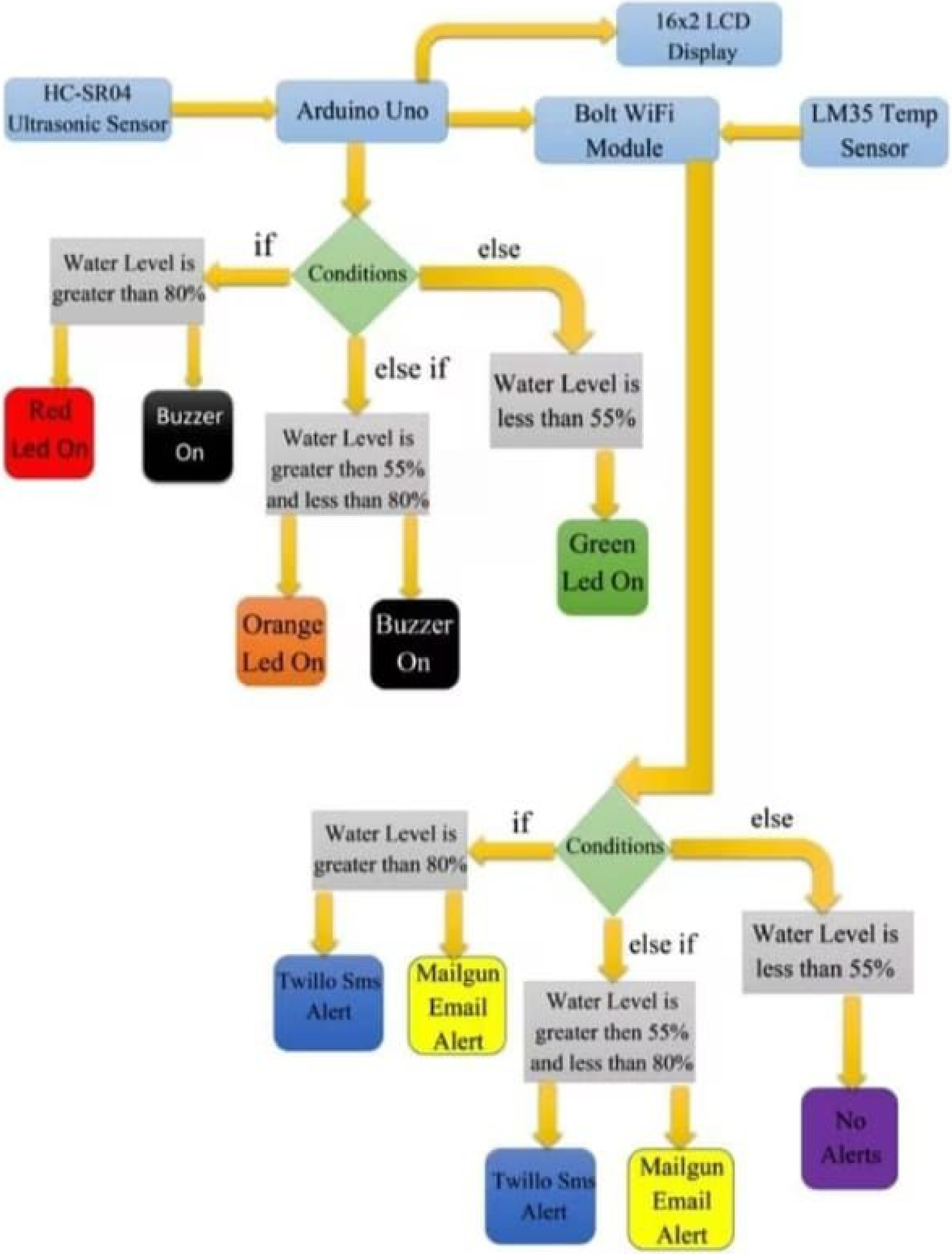


```
    except Exception as e:
        print ("Error occured: B
        print (e)
    time.sleep(15)
```

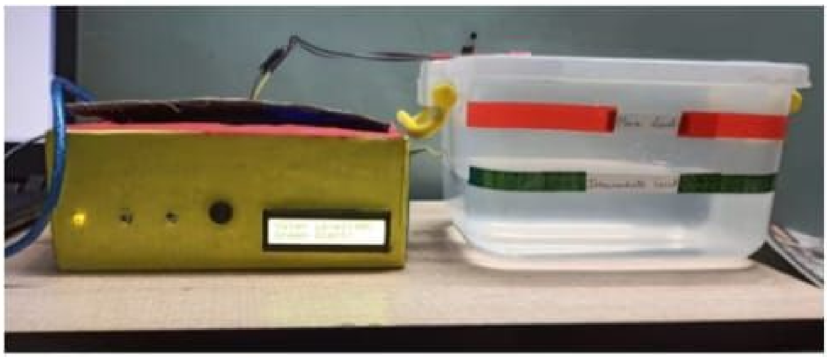
After Successfully writing code for Arduino and Python. Now it is the time to test and demonstrate the project. Move to next section for demonstration of the project.

Demonstration

Lets First have a look at the workflow of this project.

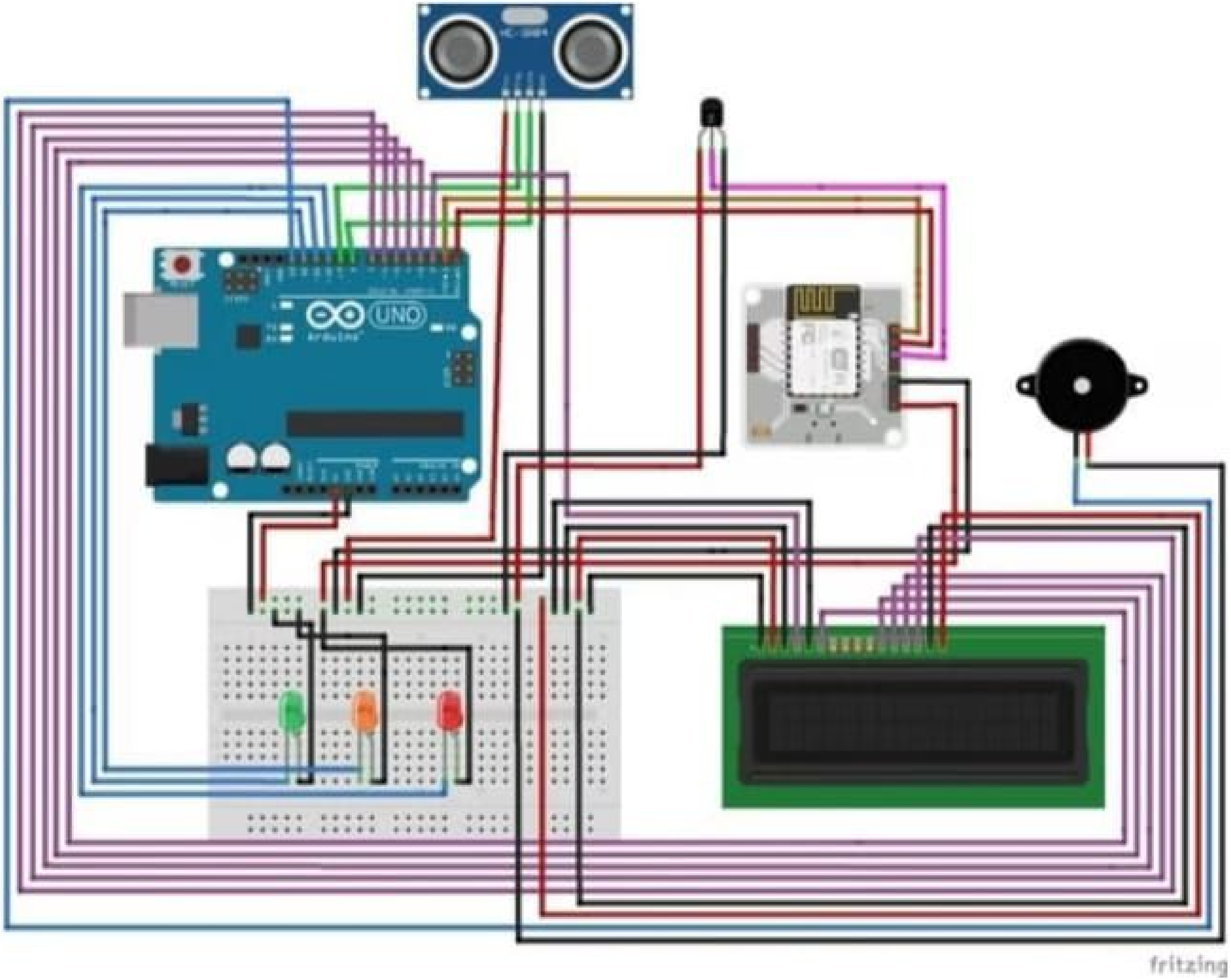

16x2 LCD Display
HC-SR04 Ultrasonic Sensor
Arduino Uno
Bolt WiFi Module
LM35 Temp Sensor
if
Conditions
else
Water Level is greater than 80%
else if
Water Level is less than 55%
Red Led On
Buzzer On
Water Level is greater then 55% and less than 80%
Green Led On
Orange Led On
Buzzer On
if
Conditions
else
Water Level is greater than 80%
else if
Water Level is less than 55%
Twillo Sms Alert
Mailgun Email Alert
Water Level is greater then 55% and less than 80%
No Alerts
Twillo Sms Alert
Mailgun Email Alert

- When water level is at Min/Normal level. That resembles 'Green Alert'. This means that water is at normal position and no sign about flood condition. Also green led will glow and it will also show green alert in Lcd display with water level.



- When water level crosses the Intermediate level. That resembles 'Orange Alert'. This means that water has crossed the 55% mark and there can be chances of flood condition at that place. With increase in water level the system sends Sms and Email alerts to the authority or registered user from Twillo and Mailgun Services respectively. Also orange led will glow and buzzer will buzz. It will also show orange alert in Lcd display.

Schematics Diagram



Conclusion

Nowadays the Internet Of things (IoT) is broadly used in worldwide, this system will display the data of the water level measured on lcd display. This project can be very helpful to the Meteorological Department to continuously monitor the dams and river beds water level. With this project it can save many people lives by giving alerts when the water level crosses beyond the limit. This project is very cost-effective, flexible and productive in areas where flood conditions happens everytime