SPRINT 3

INITIALIZING HARDWARE

Team ID	PNT2022TMID52810
Project Name	Project - Signs with smart connectivity for Better road safety

Integrate hardware that can connect to cloud services and modify output based on output processing.

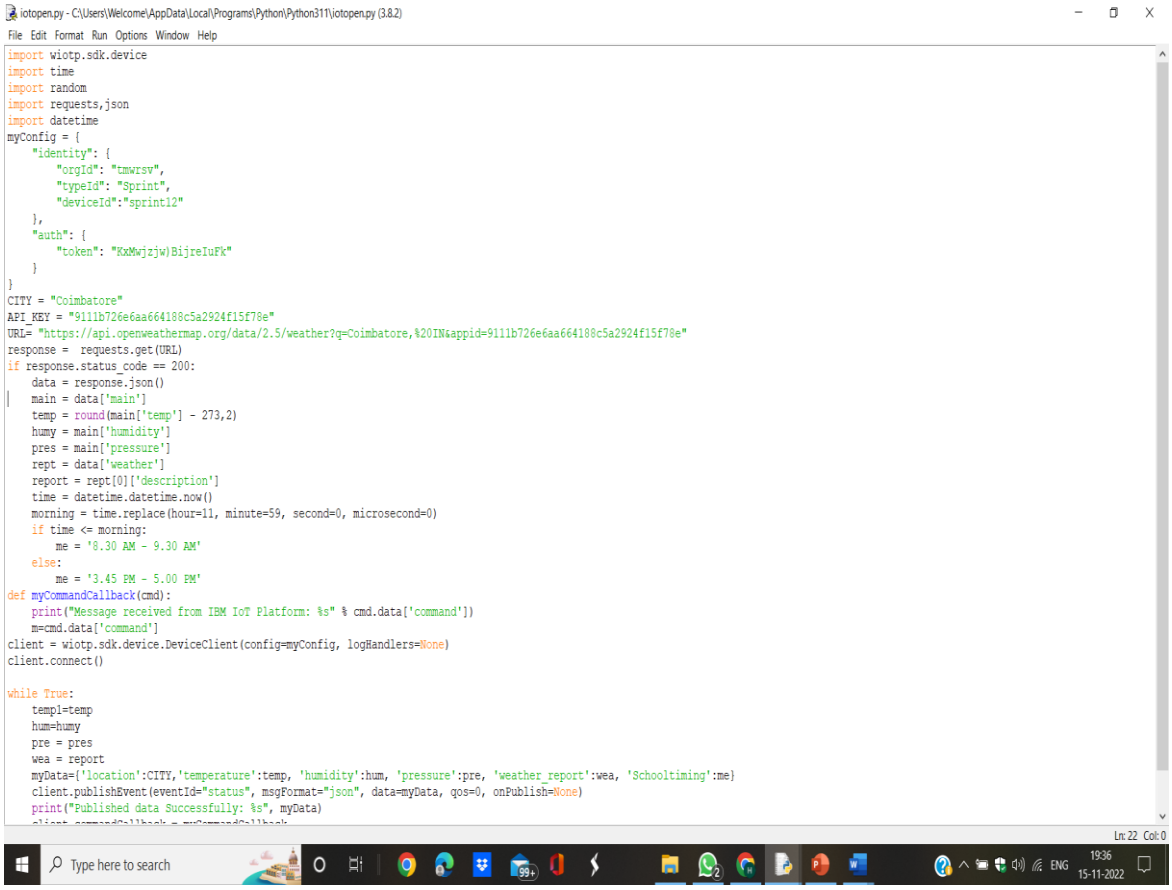
PYTHON SCRIPT:

```
import wiotp.sdk.device
import time
import random
import requests,json
import datetime
myConfig = {
   "identity": {
      "orgId": "tmwrsv",
      "typeId": "Sprint",
      "deviceId":"sprint12"
   },
   "auth": {
      "token": "KxMwjzjw)BijreIuFk"
   }
}
CITY = "Coimbatore"
API_KEY = "9111b726e6aa664188c5a2924f15f78e"
URL=
"https://api.openweathermap.org/data/2.5/weather?q=Coimbatore,%20IN&appid=9111
b726e6aa664188c5a2924f15f78e"
response =  requests.get(URL)
if response.status_code == 200:
   data = response.json()
   main = data['main']
   temp = round(main['temp'] - 273,2)
   humy = main['humidity']
   pres = main['pressure']
   rept = data['weather']
   report = rept[0]['description']
   time = datetime.datetime.now()
   morning = time.replace(hour=11, minute=59, second=0, microsecond=0)
   if time <= morning:
      me = '8.30 AM - 9.30 AM'
   else:
      me = '3.45 PM - 5.00 PM'
```

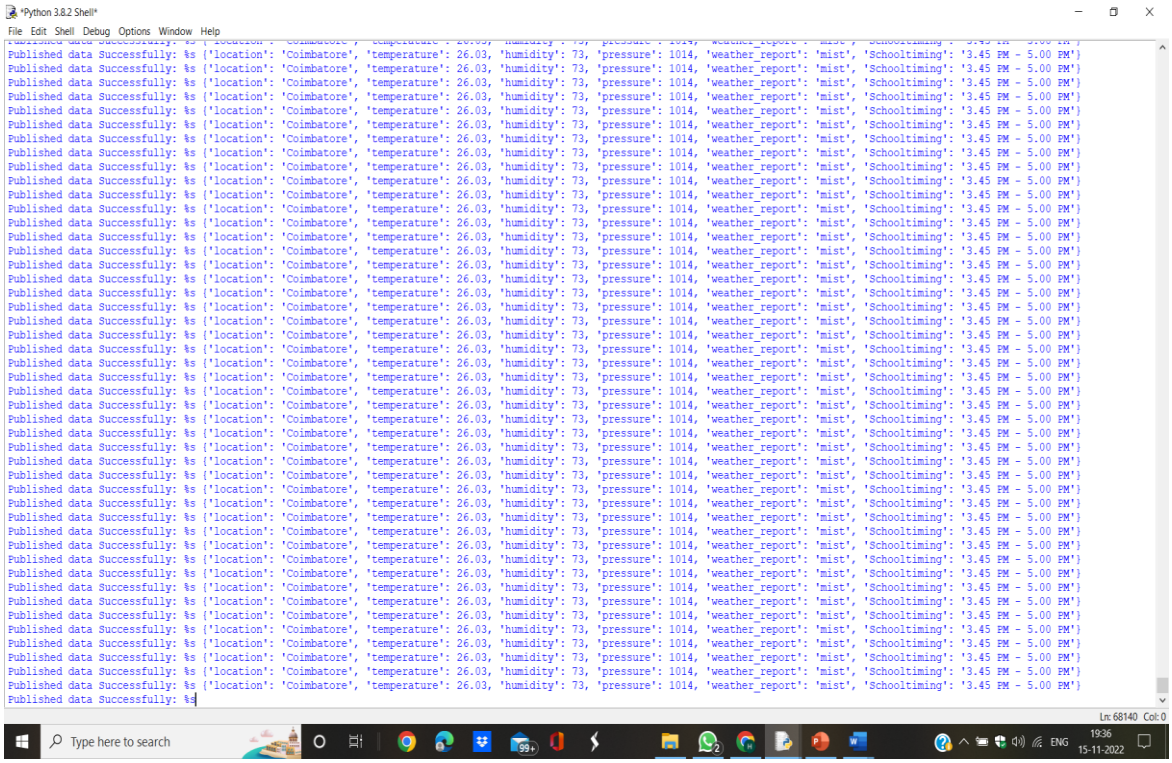
```
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    temp1=temp
    hum=humy
    pre = pres
    wea = report
    myData={'location':CITY,'temperature':temp, 'humidity':hum, 'pressure':pre,
'weather_report':wea, 'Schooltiming':me}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
client.disconnect()
```

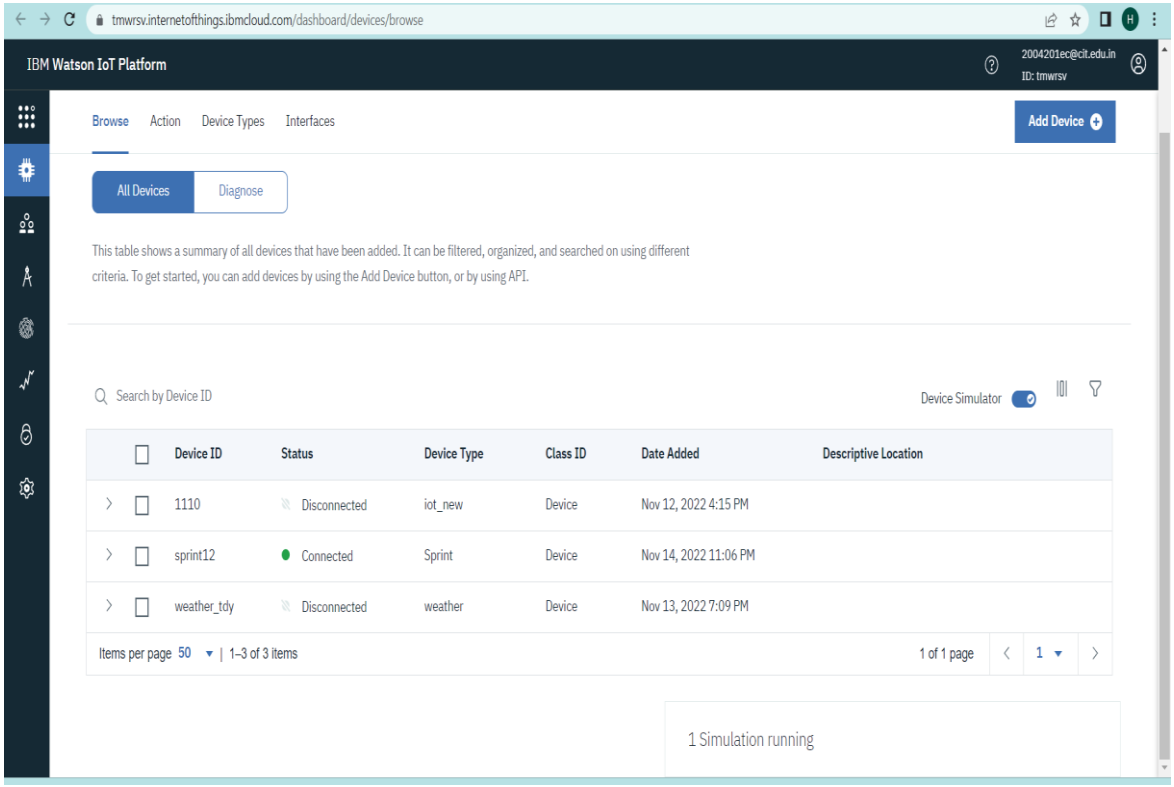
PYTHON IDE:

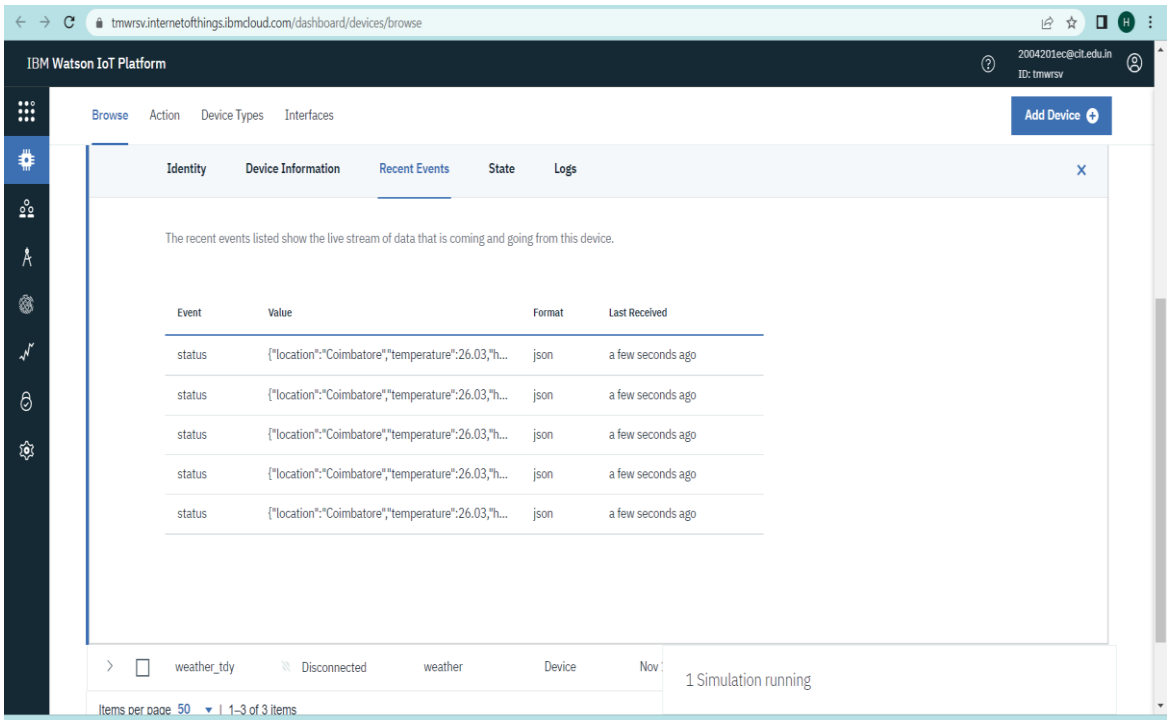


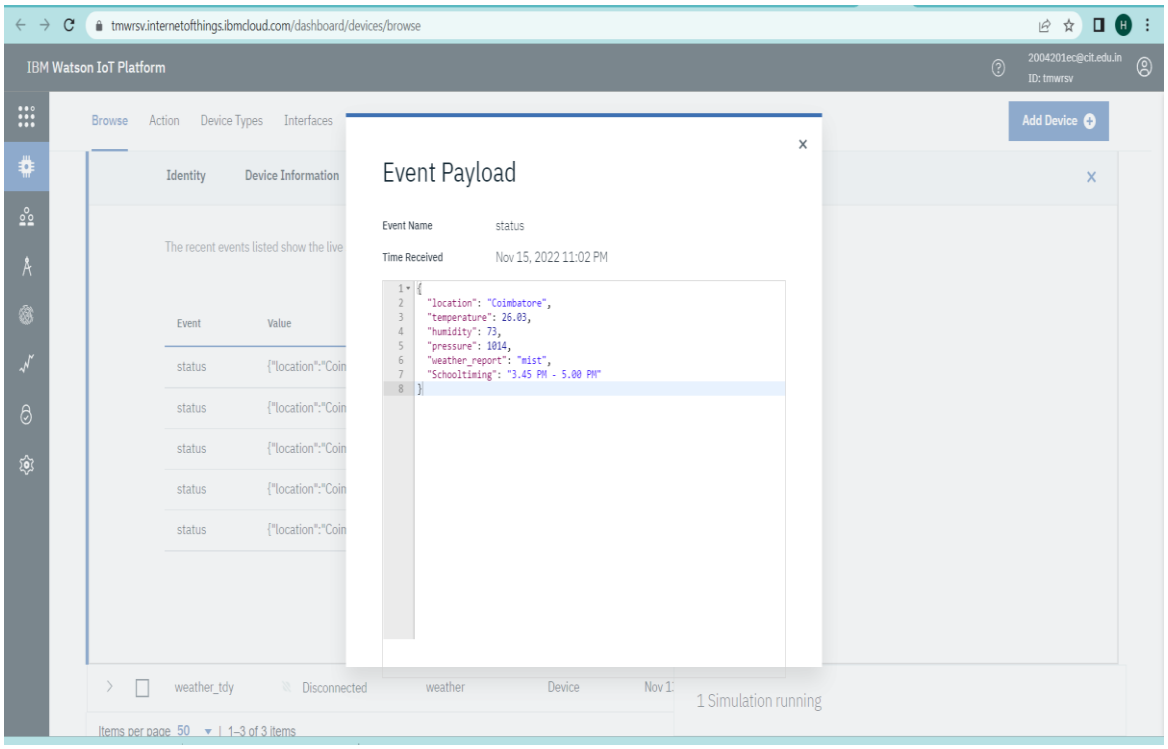
PYTHON IDE OUTPUT:



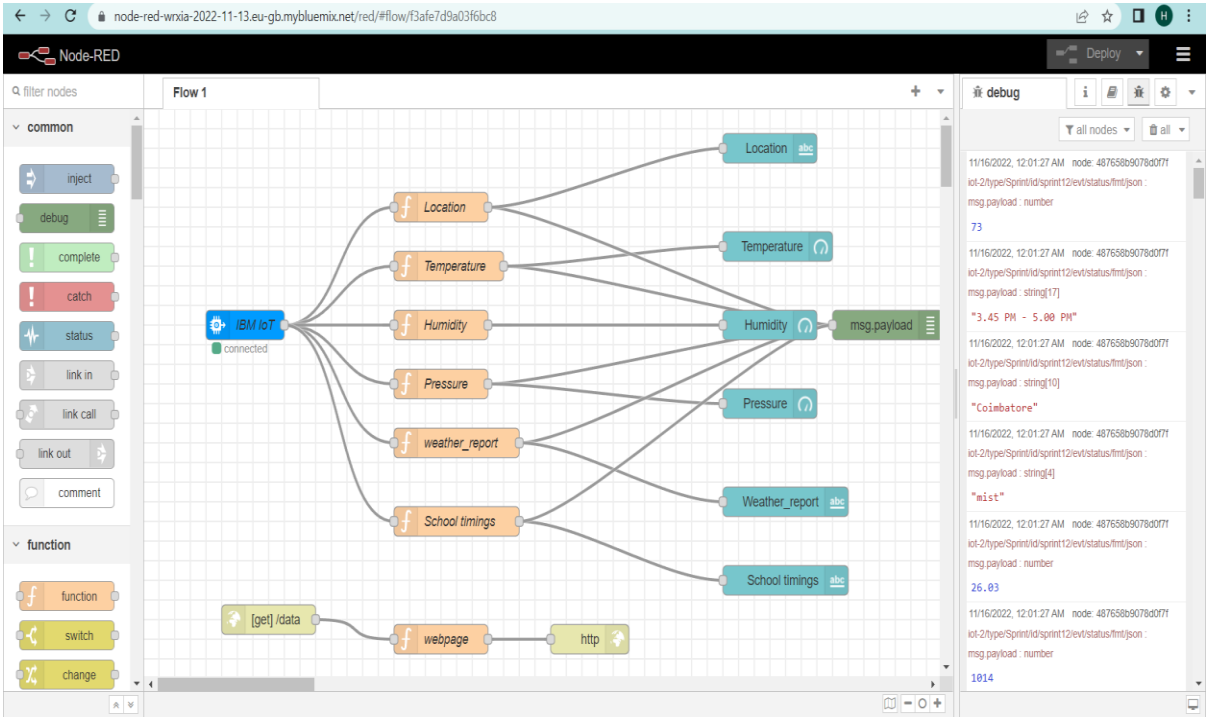
IBM WATSON IOT PLATFORM OUTPUT:

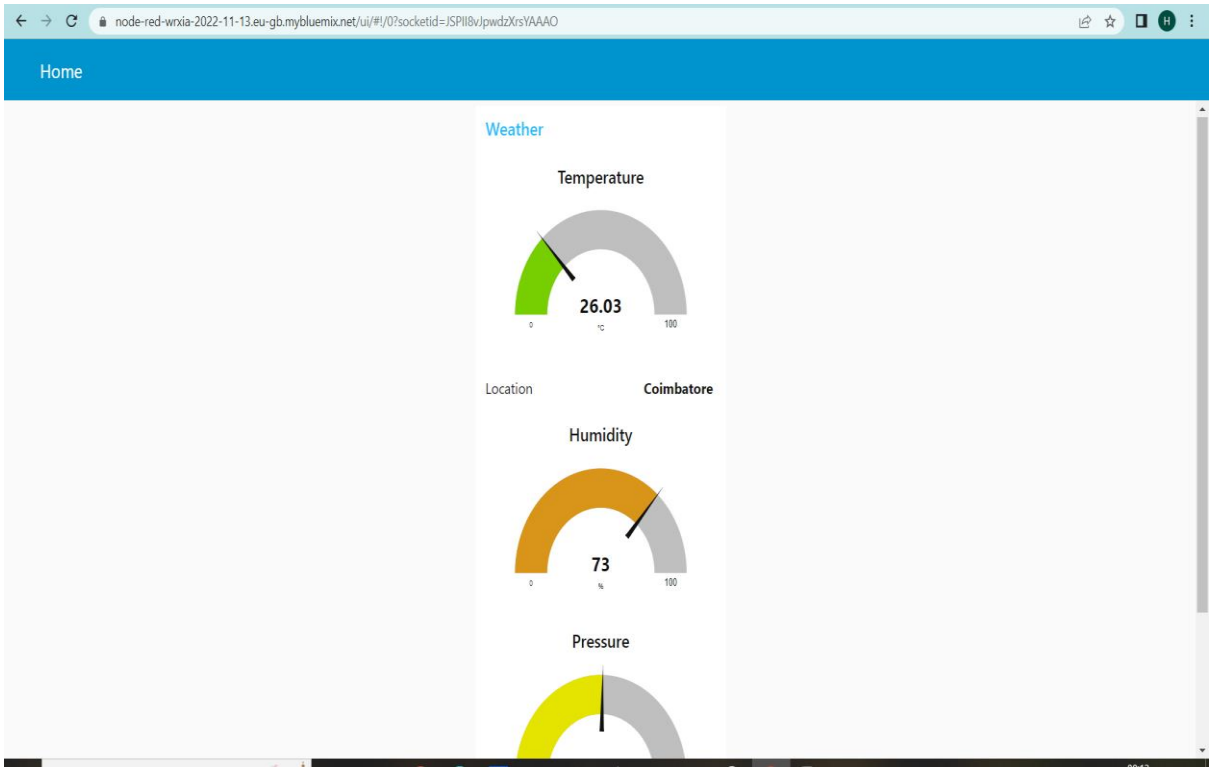


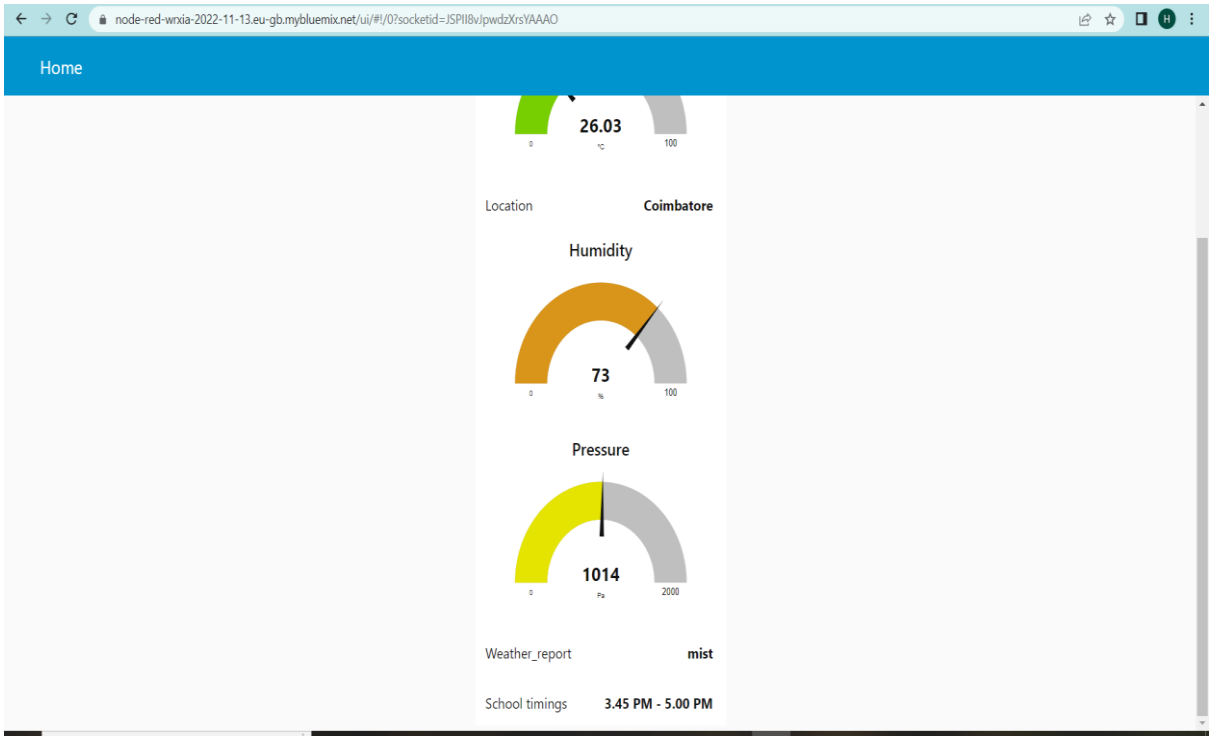

tmwrsv.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
2004201ec@cit.edu.in
ID: tmwrsv
Browse
Action
Device Types
Interfaces
Add Device
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
status
{"location":"Coimbatore","temperature":26.03,"h...
json
a few seconds ago
weather_tdy
Disconnected
weather
Device
1 Simulation running
Items per page 50
1–3 of 3 items


tmwrsv.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
2004201ec@cit.edu.in
ID: tmwrsv
Browse
Action
Device Types
Interfaces
Add Device
Identity
Device Information
Event Payload
Event Name
status
Time Received
Nov 15, 2022 11:02 PM
{
"location": "Coimbatore",
"temperature": 26.03,
"humidity": 73,
"pressure": 1014,
"weather_report": "mist",
"Schooltiming": "3.45 PM - 5.00 PM"
}
weather_tdy
Disconnected
weather
Device
1 Simulation running
Items per page 50
1–3 of 3 items

WEB APPLICATION USING NODE-RED SERVICE:







{"location":"Coimbatore","temperature":26.03,"humidity":73,"pressure":1014,"weather_report":"mist","Schooltiming":"3.45 PM - 5.00 PM"}

HARDWARE USED:

1. **ESP32:**

   ESP32 is a microcontroller with built-in Bluetooth and Wi-Fi. It is used to get data from the sensors and send control signals to the LCD display. Also used to send the data to the cloud.

2. **ULTRASONIC SENSOR:**

   Ultrasonic sensor is used detect the presence of object over a distance. So here it is used to find the traffic density by detecting the presence of number of vehicles.

3. **DHT11:**

   The **DHT11** is a commonly used **Temperature and humidity sensor that** comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data.

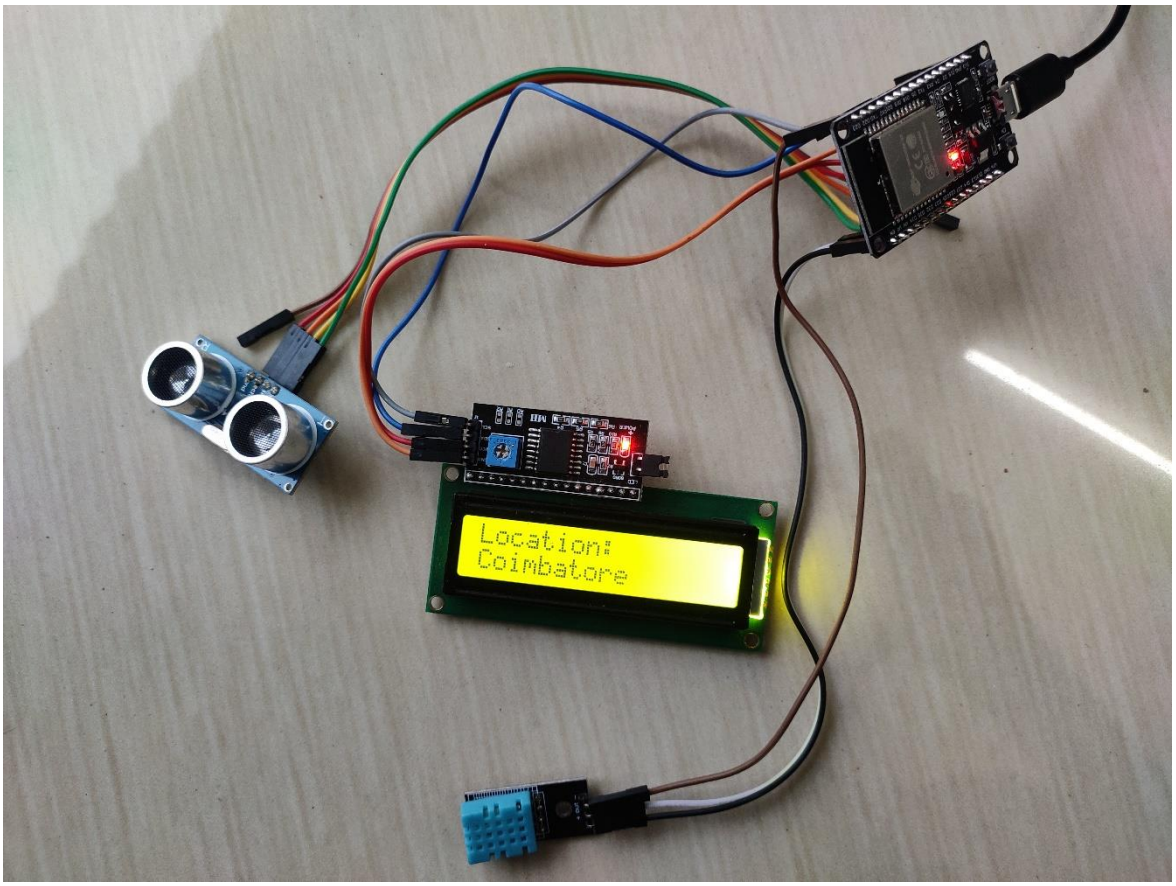
4. **I2C:**

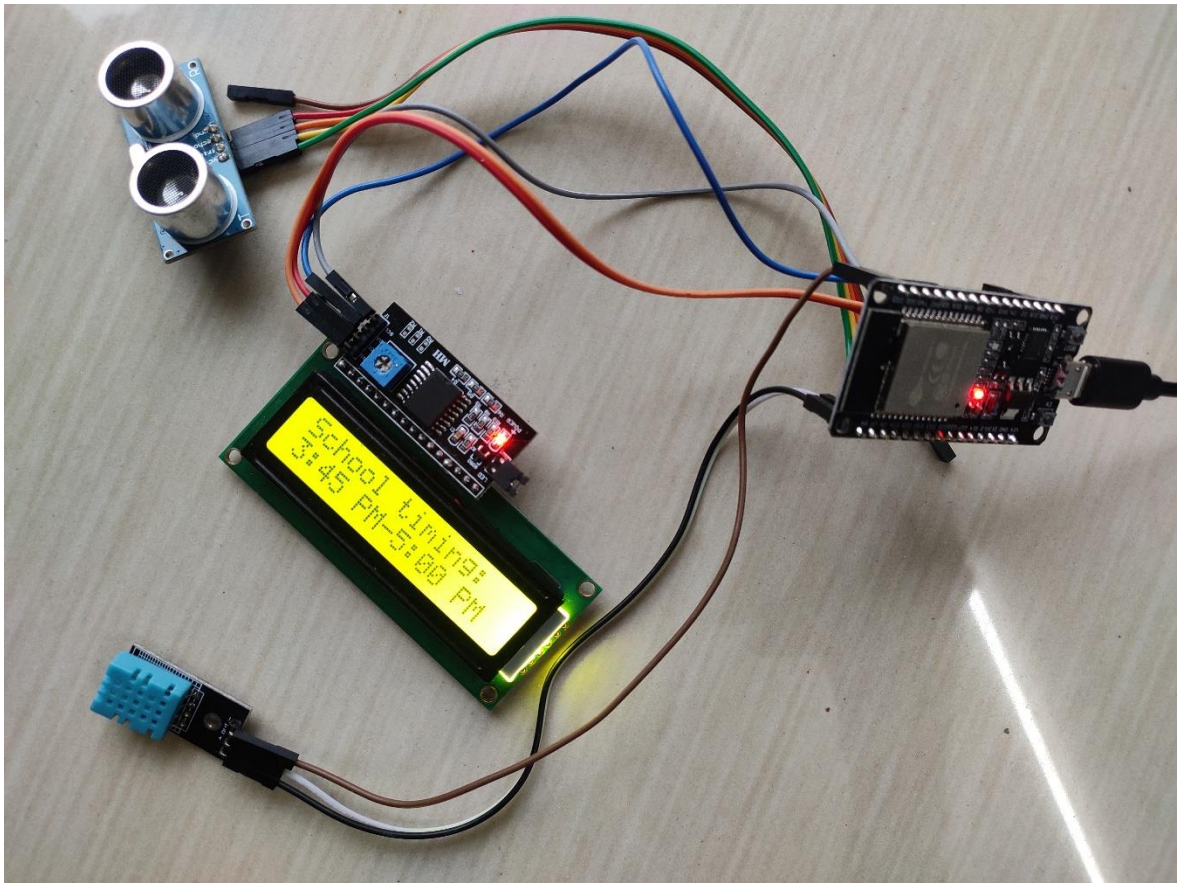
   The I2C bus is a very popular and powerful bus used for communication between a master (or multiple masters) and a single or multiple slave device. Here it is used to simplify 16 pins of LCD display into 4 pins.

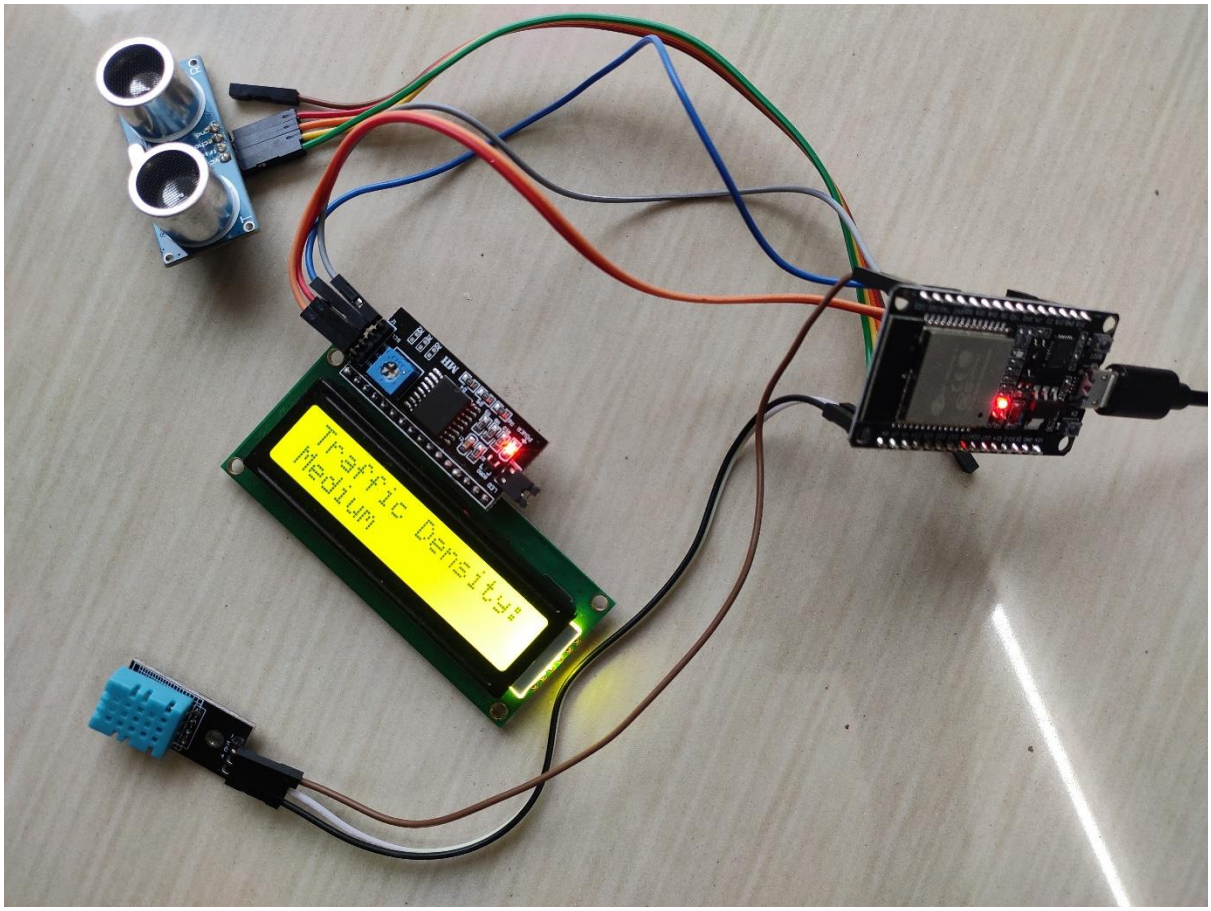
5. **LCD DISPLAY:**

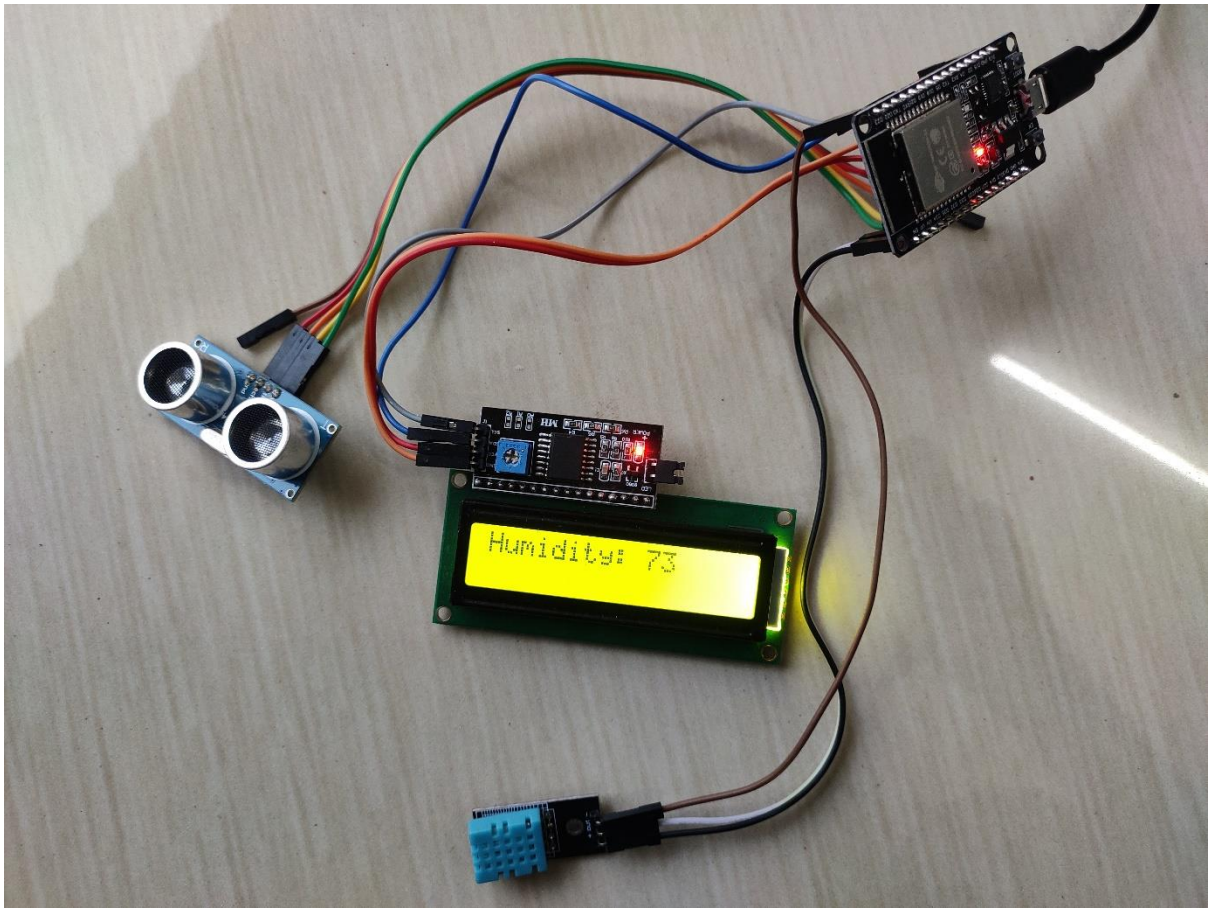
   A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. The 16 x 2 intelligent alphanumeric dot matrix display is capable of displaying 224 different characters and symbols. This LCD has two registers, namely, Command and Data.

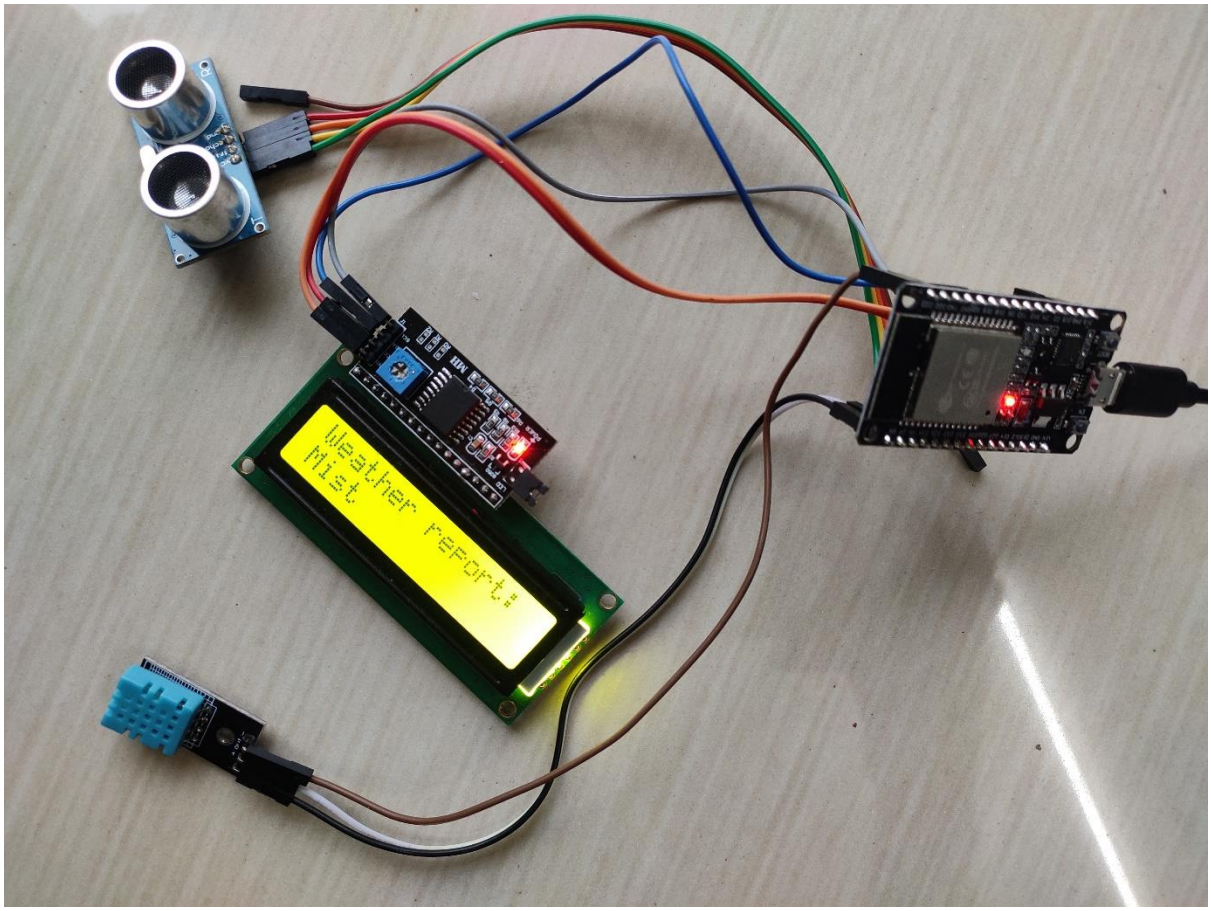
Output:

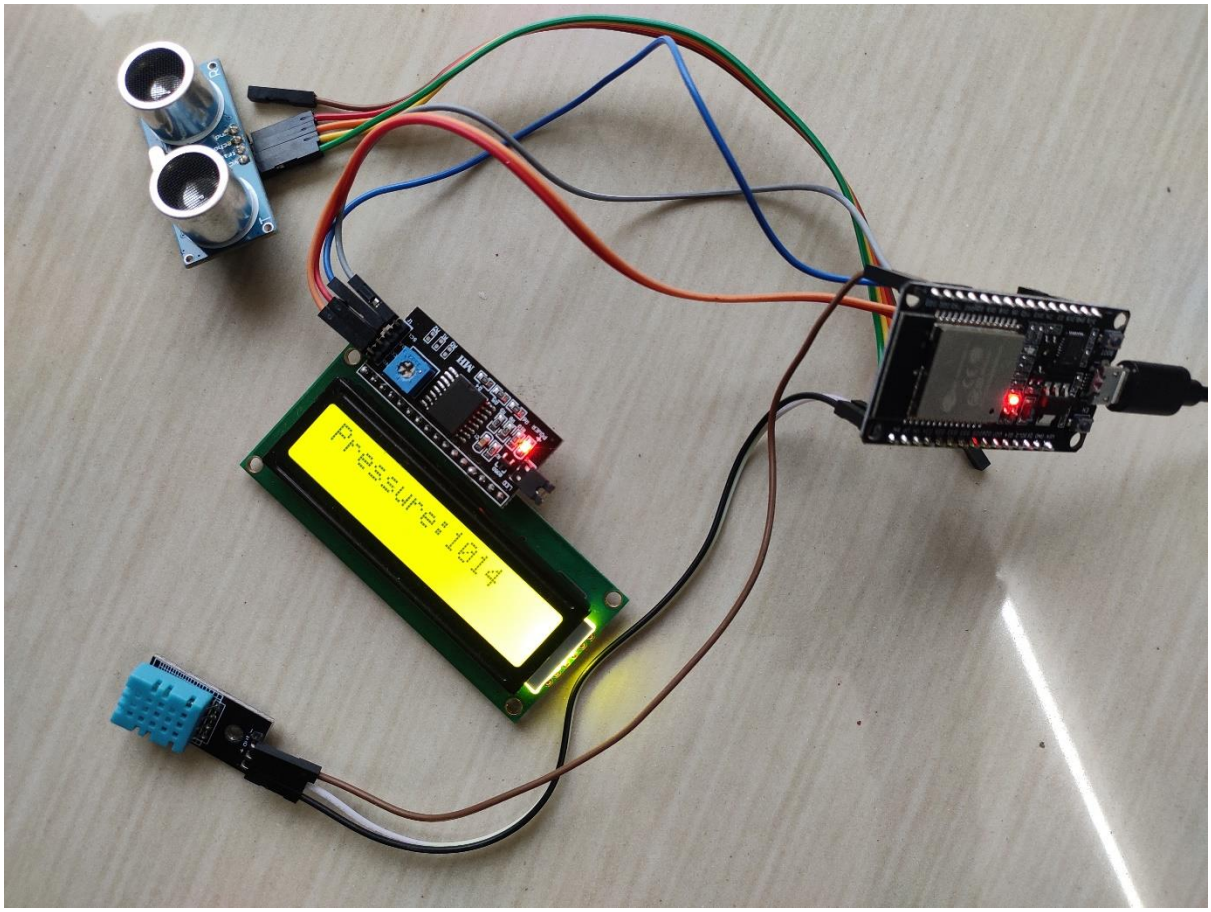


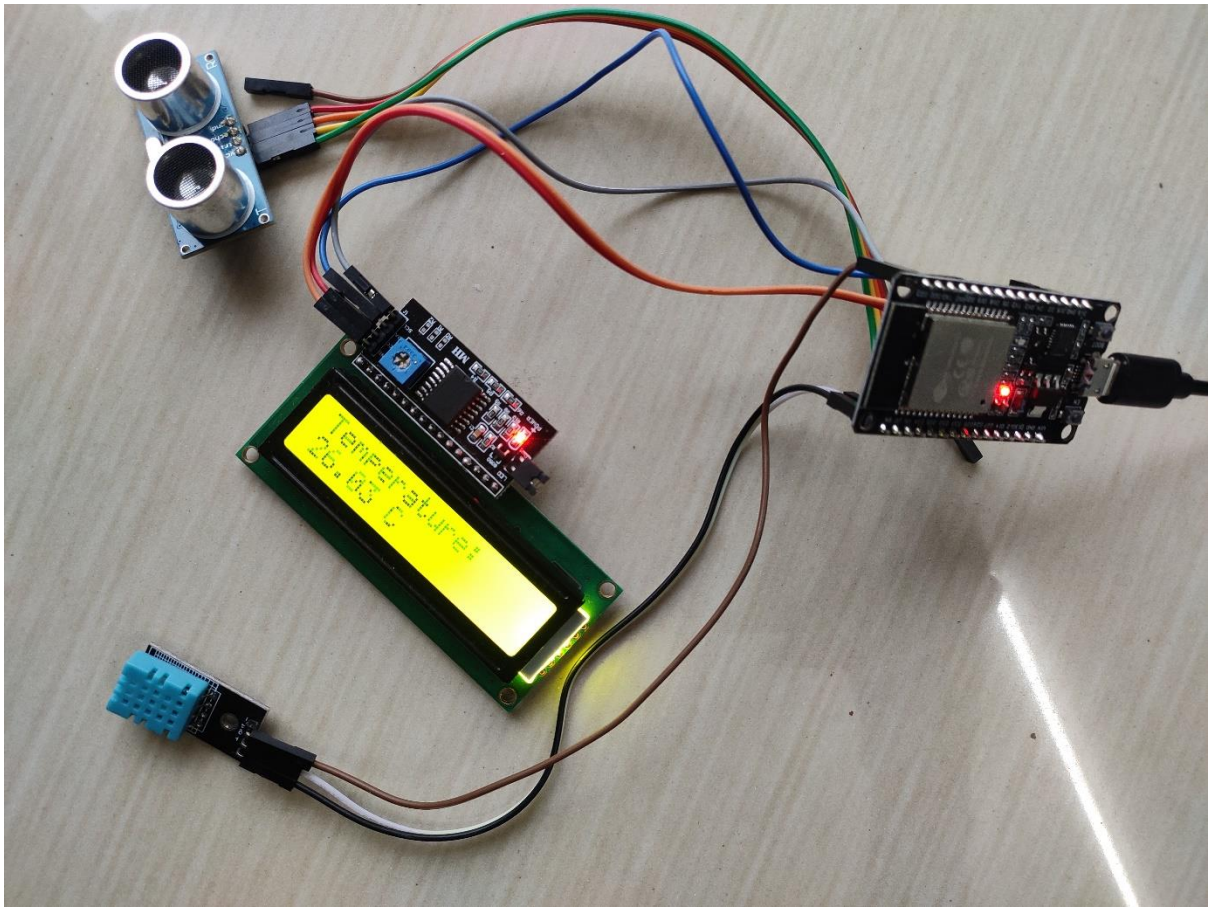



Traffic Density:
Medium


Humidity: 73


Weather report:
Mist


Pressure:1014


Temperature:
26.00 C