IoT based air pollution monitoring system

Name: Srujan Pailwar
Contact: 9875705275
Email Id: 2018.srujan.pailwar@ves.ac.in
Github Link: https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12
Github ID: **@srujan2309**

1. Index

2. Introduction

a. Existing problem

Air pollution kills an estimated seven million people worldwide every year. WHO data shows that 9 out of 10 people breathe air that exceeds WHO guideline limits containing high levels of pollutants, with low- and middle-income countries suffering from the highest exposures. Ambient air pollution accounts for an estimated 4.2 million deaths per year due to stroke, heart disease, lung cancer, acute and chronic respiratory diseases. Household air pollution is one of the leading causes of disease and premature death in the developing world. Exposure to smoke from cooking fires causes 3.8 million premature deaths each year, mostly in low- and middle-income countries.

Effects:

Air pollution can also cause long-term damage to people's nerves, brain, kidneys, liver, and other organs. Serious effects of air pollution also include

- Global Warming.
- Climate Change.
- Acid Rain.
- Smog effect.
- Deterioration of fields.
- Extinction of animal species.

b. Overview

In this project we made an IoT Based Air Pollution Monitoring System which monitors the Air Quality over a webserver using internet and triggers an alarm when the air quality goes down beyond a certain level, means when there are sufficient amounts of harmful gases are present in the air like CO2, smoke, alcohol, benzene and NH3. It shows the air quality in PPM on the LCD and as well as on webpage so that it can be monitored very easily.

c. Purpose

To find out air quality index in ppm and representing it for ease of monitoring while it is updated real time. Also uploading data onto firebase so as to get real time updates on web app and use the same for prediction of further AQI using machine learning model.

3. Literature review

Ambient air pollution is the leading environmental risk factor for disease globally. Air pollutants can increase the risk of some respiratory infections. The studies were assessed for overall quality and risk of bias using standard criteria. The pollutant most frequently associated with statistically significant outcomes was fine particulate matter (< 2.5 µm; $PM_{2.5}$); 6/11 studies assessed $PM_{2.5}$, of which 4/6 demonstrated a significant association). There was some evidence of significant associations between PM_{10} (< 10 µm), nitrogen dioxide (NO_2) and Sulphur dioxide (SO_2), but these associations were inconsistent. The existing epidemiological evidence is limited and shows mixed results. However, it is plausible that exposure to air pollutants, particularly $PM_{2.5}$, may suppress important immune defense mechanisms, increasing an individual's susceptibility, thus reducing mortality.

Types of Air pollution:

- **Ambient air pollution:**
  Ambient air pollution is a broader term used to describe air pollution in outdoor environments. Poor ambient air quality occurs when pollutants reach high enough concentrations to affect human health and/or the environment.
  From smog hanging over cities to smoke inside the home, air pollution poses a major threat to health and climate. Sources of air pollution are multiple and context-specific. The major outdoor pollution sources include residential energy for cooking and heating, vehicles, power generation, agriculture/waste incineration, and industry. Policies and investments supporting integrated policies that support sustainable land use, cleaner household energy and transport, energy-efficient housing, power generation, industry, and better municipal waste management can effectively reduce key sources of ambient air pollution.

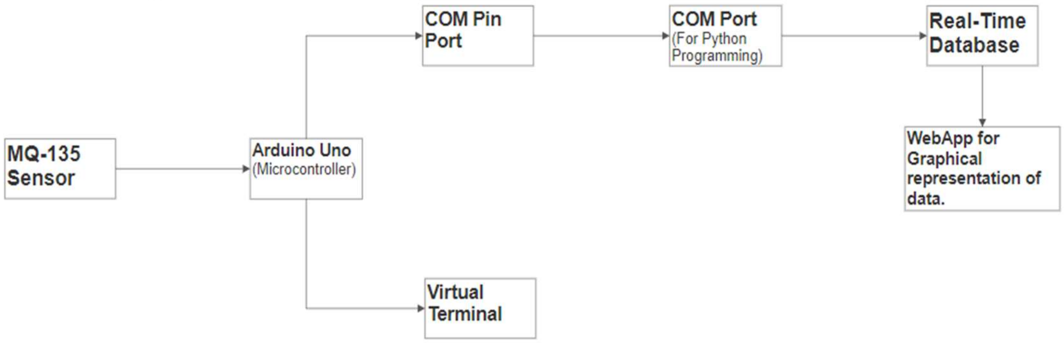
- **Outdoor Air pollution:**
  The common sources of outdoor air pollution are emissions caused by combustion processes from motor vehicles, solid fuel burning and industry. Other pollution sources include smoke from bushfires, windblown dust, and biogenic emissions from vegetation (pollen and mold spores).

- **Household Air pollution:**

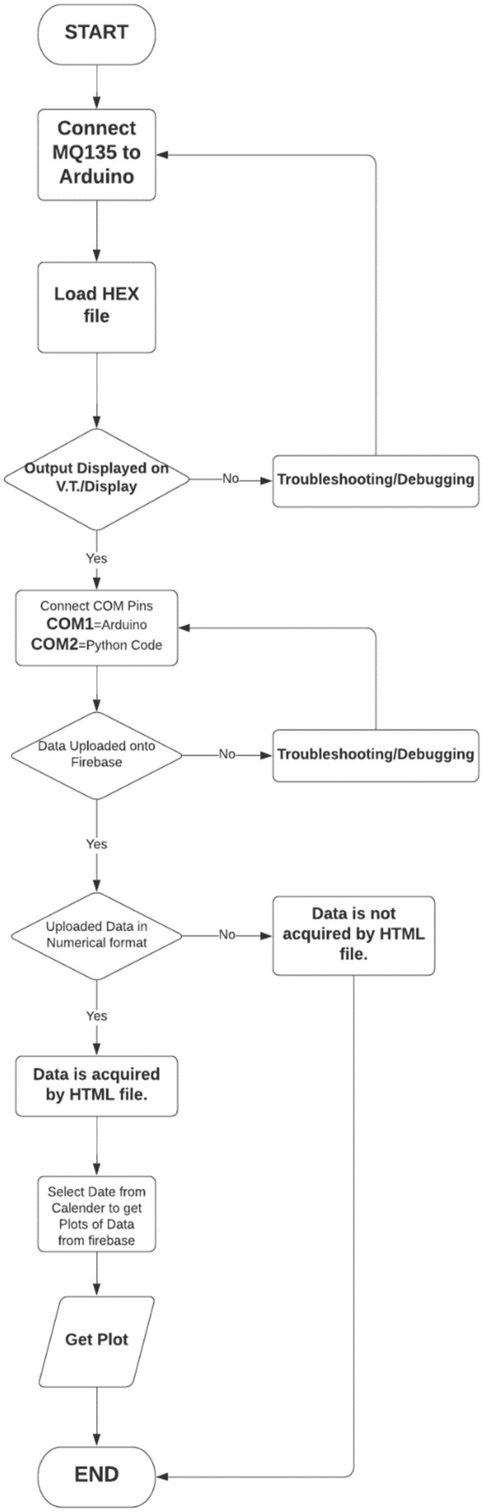
  Indoor air pollution is the degradation of indoor air quality by harmful chemicals and other materials; it can be up to 10 times worse than outdoor air pollution Exposure to smoke from cooking fires Burning fuels such as dung, wood and coal in inefficient stoves or open hearths produces a variety of health-damaging pollutants, including particulate matter (PM), methane, carbon monoxide, polyaromatic hydrocarbons (PAH) and volatile organic compounds (VOC). Burning kerosene in simple wick lamps also produces significant emissions of fine particles and other pollutants. Particulate matter is a pollutant of special concern. Many studies have demonstrated a direct relationship between exposure to PM and negative health impacts. Smaller-diameter particles (PM2.5 or smaller) are generally more dangerous and ultrafine particles (one micron in diameter or less) can penetrate tissues and organs, posing an even greater risk of systemic health impacts.

4. Proposed solution

a. Block diagram

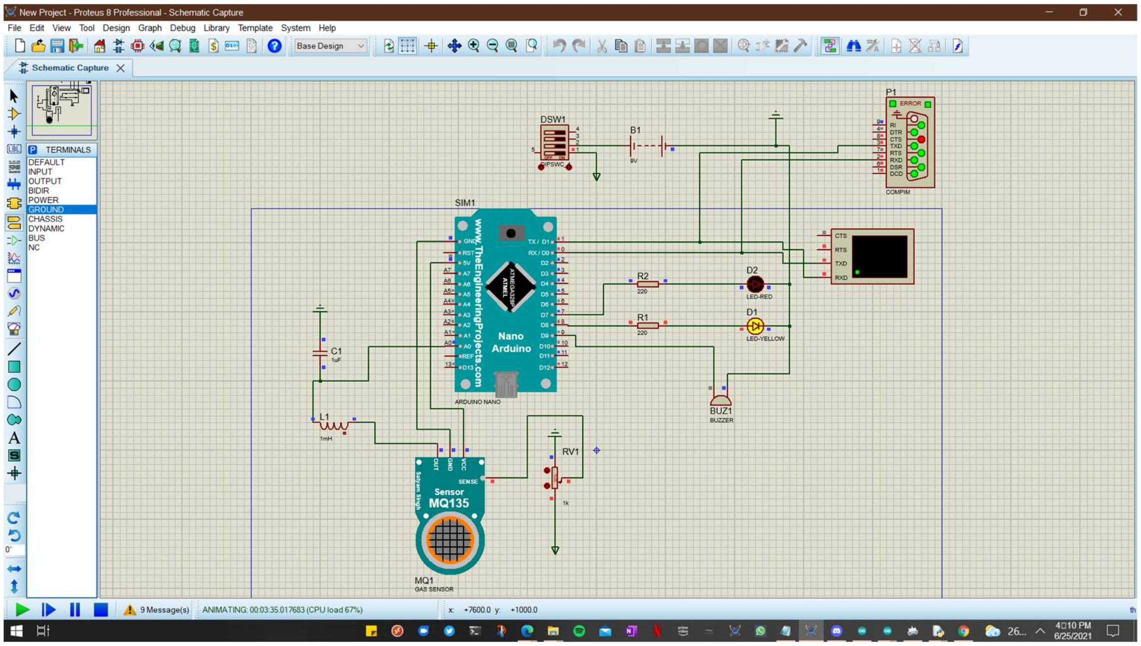

MQ-135 Sensor
Arduino Uno (Microcontroller)
COM Pin Port
COM Port (For Python Programming)
Real-Time Database
WebApp for Graphical representation of data.
Virtual Terminal

b. Flowchart

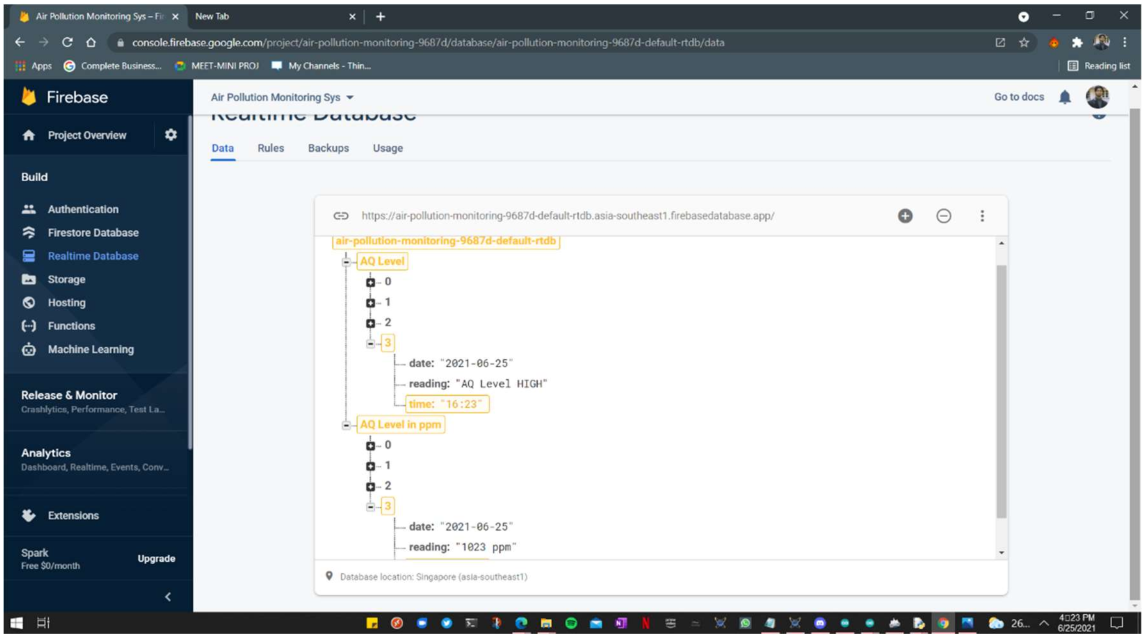

START
Connect MQ135 to Arduino
Load HEX file
Output Displayed on V.T./Display
No
Troubleshooting/Debugging
Yes
Connect COM Pins
COM1=Arduino
COM2=Python Code
Data Uploaded onto Firebase
No
Troubleshooting/Debugging
Yes
Uploaded Data in Numerical format
No
Data is not acquired by HTML file.
Yes
Data is acquired by HTML file.
Select Date from Calender to get Plots of Data from firebase
Get Plot
END

5. Result

a. Fullscreen Screenshot of proteus.



b. Submit the Screenshot of the Firebase real-time database (data can be of your choice).

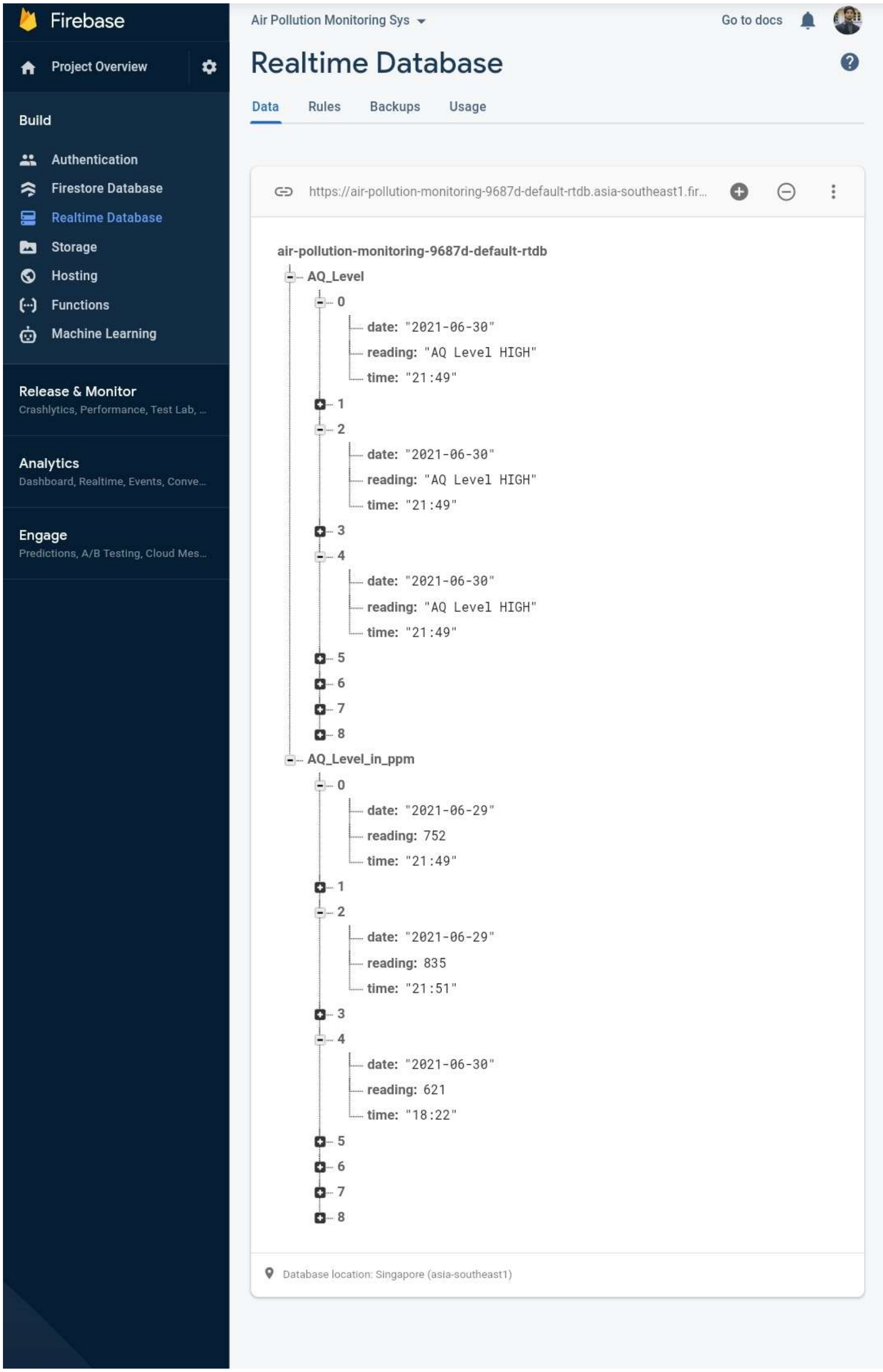


https://air-pollution-monitoring-9687d-default-rtdb.asia-southeast1.firebasedatabase.app/

```
air-pollution-monitoring-9687d-default-rtdb
  AQ Level
    0
    1
    2
    3
      date: "2021-06-25"
      reading: "AQ Level HIGH"
      time: "16:23"
  AQ Level in ppm
    0
    1
    2
    3
      date: "2021-06-25"
      reading: "1023 ppm"
```

Database location: Singapore (asia-southeast1)

c. Data uploaded on firebase. (Firebase Window)



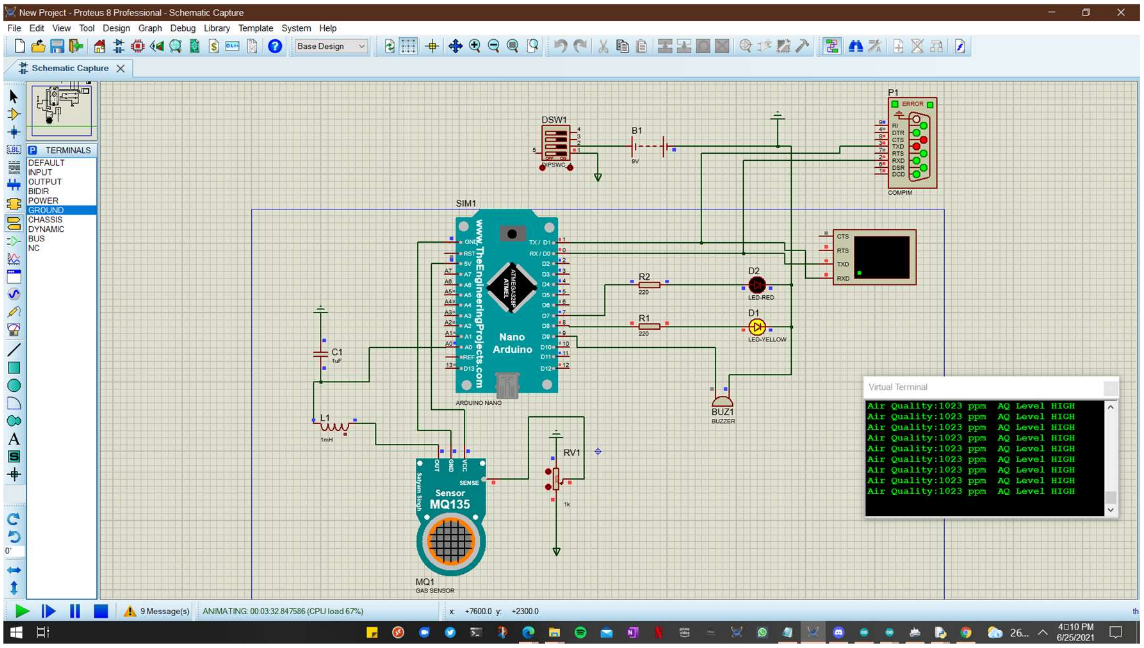
Realtime Database

https://air-pollution-monitoring-9687d-default-rtdb.asia-southeast1.fir...

```
air-pollution-monitoring-9687d-default-rtdb
 - AQ_Level
    - 0
        date: "2021-06-30"
        reading: "AQ Level HIGH"
        time: "21:49"
    + 1
    - 2
        date: "2021-06-30"
        reading: "AQ Level HIGH"
        time: "21:49"
    + 3
    - 4
        date: "2021-06-30"
        reading: "AQ Level HIGH"
        time: "21:49"
    + 5
    + 6
    + 7
    + 8
 - AQ_Level_in_ppm
    - 0
        date: "2021-06-29"
        reading: 752
        time: "21:49"
    + 1
    - 2
        date: "2021-06-29"
        reading: 835
        time: "21:51"
    + 3
    - 4
        date: "2021-06-30"
        reading: 621
        time: "18:22"
    + 5
    + 6
    + 7
    + 8
```

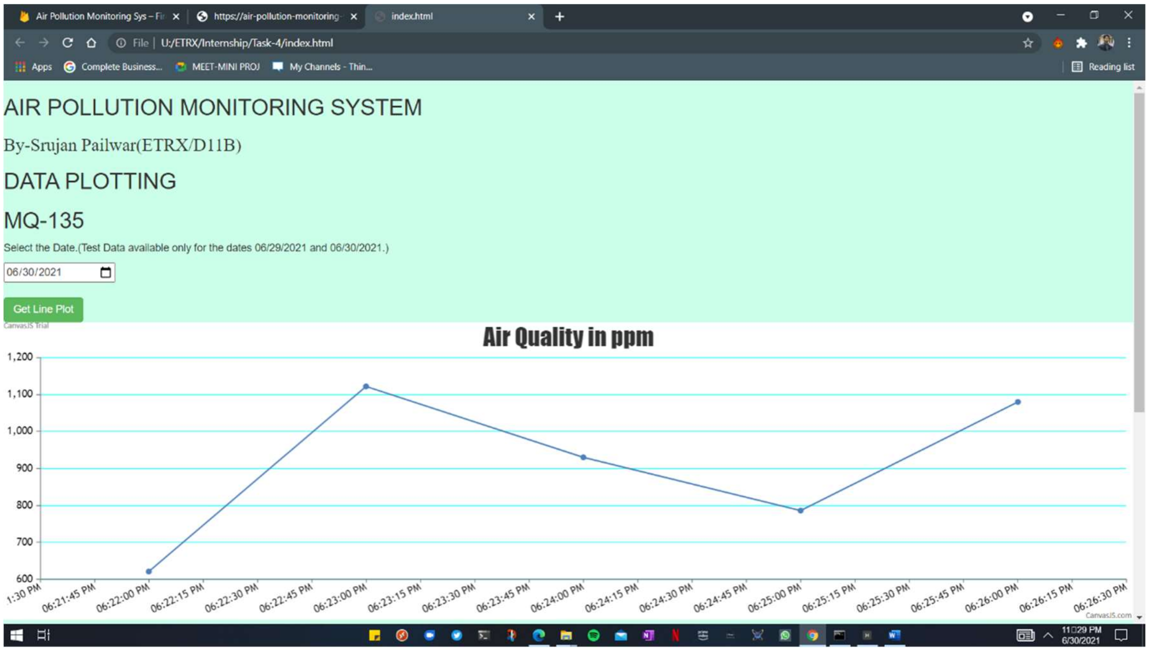
Database location: Singapore (asia-southeast1)

d. **Screenshot of proteus while simulation.**



```
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
Air Quality:1023 ppm  AQ Level HIGH
```

e. **Submit the screenshot of the plotted graph in the HTML page. (The website URL bar and chrome should be visible in the screenshot, to prove that you plotted it in HTML page)**



Air Pollution Monitoring Sys – Fi
https://air-pollution-monitoring-
index.html
File | U:/ETRX/Internship/Task-4/index.html
Apps
Complete Business...
MEET-MINI PROJ
My Channels - Thin...
Reading list

AIR POLLUTION MONITORING SYSTEM

By-Srujan Pailwar(ETRX/D11B)

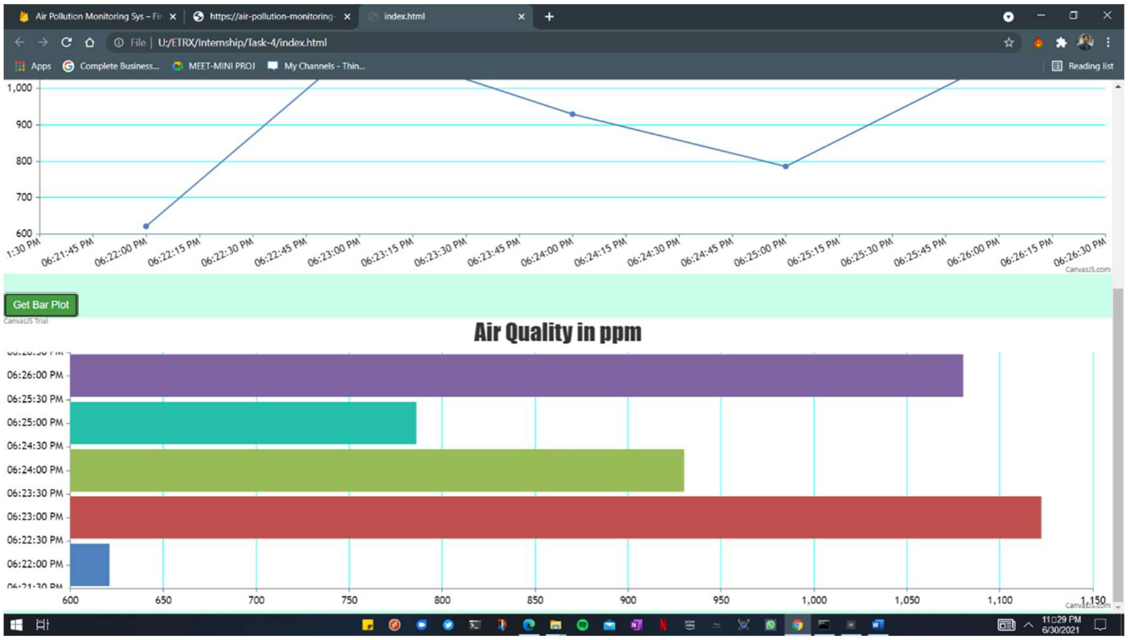
DATA PLOTTING

MQ-135

Select the Date.(Test Data available only for the dates 06/29/2021 and 06/30/2021.)

06/30/2021

Get Line Plot



f. Link of deployed Webapp

- https://air-pollution-monitoring-9687d.web.app

6. Advantages and disadvantages

- **Advantages:**
  1. Easy installation
  2. Easy maintenance and so on
  3. Worldwide monitoring
  4. Real-time alerts
  5. Reduced wiring
  6. Reduce cost
  7. Ease of availability of sensors.
  8. Detecting a wide range of gases like CO2, CO etc.
  9. Simple, compact and easily handle.
  10. Continuous update of change in ppm of quality.
- **Challenges:**
  1. Requires PC/laptop to view the pollution level as no direct LCD display is connected.

7. Applications

a. Assessing the level of pollution in relation to the ambient air quality standards.
b. Providing the real-time contamination in air in ppm.
c. Used in hospitals to maintain quality of air.
d. Used in air purifiers.
e. Used in Industries to measure air contamination.
f. Twitter access to share the details on social media.

8. Learning Outcomes

a. Better Knowledge of proteus software, Arduino IDE and creation of Hex File.
b. Basic Understanding of Firebase – How data gets uploaded in real-time on it.
c. Practical experience with C programming for Arduino and Python programming for data upload.
d. Designing of Web App using HTML, CSS, JAVASCRIPT.
e. Linking HTML code with firebase to retrieve data from it.
f. Hosting, creation and deployment of WebApp.

9. Conclusion

The air pollution monitoring system is successfully implemented as a prototype. In this project A few sensors are deployed in the atmosphere to constantly monitor the Temperature, Humidity, Carbon Monoxide, Smoke, LPG, PM2.5 and PM10 levels in the atmosphere. In this work, a one-way communication between ThingSpeak, an open-source cloud platform. Arduino has been used as a gateway to interface the hardware system. The Graphs are plotted in ThingSpeak according to the sensors data received. The Air Pollution Monitoring System will be implemented and tested in real-life working condition after simulation is done. A large-scale node placement and data gathering can be planned for the purpose of forecasting of air pollution. All the sensors and other stuff works as per the expectations. The sensors effectively sense the surrounding conditions and give the readings. The AQI readings are successfully acquired using MQ135 sensor. The system senses data and transmits it continuously. The data is successfully uploaded over firebase from where it is pushed to a machine learning code which then successfully displays the graph.

10. References

a. Frances Moore, "Climate Change and Air Pollution: Exploring the Synergies and Potential for Mitigation in Industrializing Countries", Sustainability, 2009. Vol. 1(1), pp. 43-54.

b. Brook RD, F.B., Cascio W, Hong Y, Howard G, Lipsett M, Luepker R, Mittleman M, Samet J, Smith SC Jr, Tager I; "Air pollution and cardiovascular disease: a statement for healthcare professionals from the Expert Panel on Population and Prevention Science of the American Heart Association", Circulation, Vol. 109 (21) June 1, 2004.

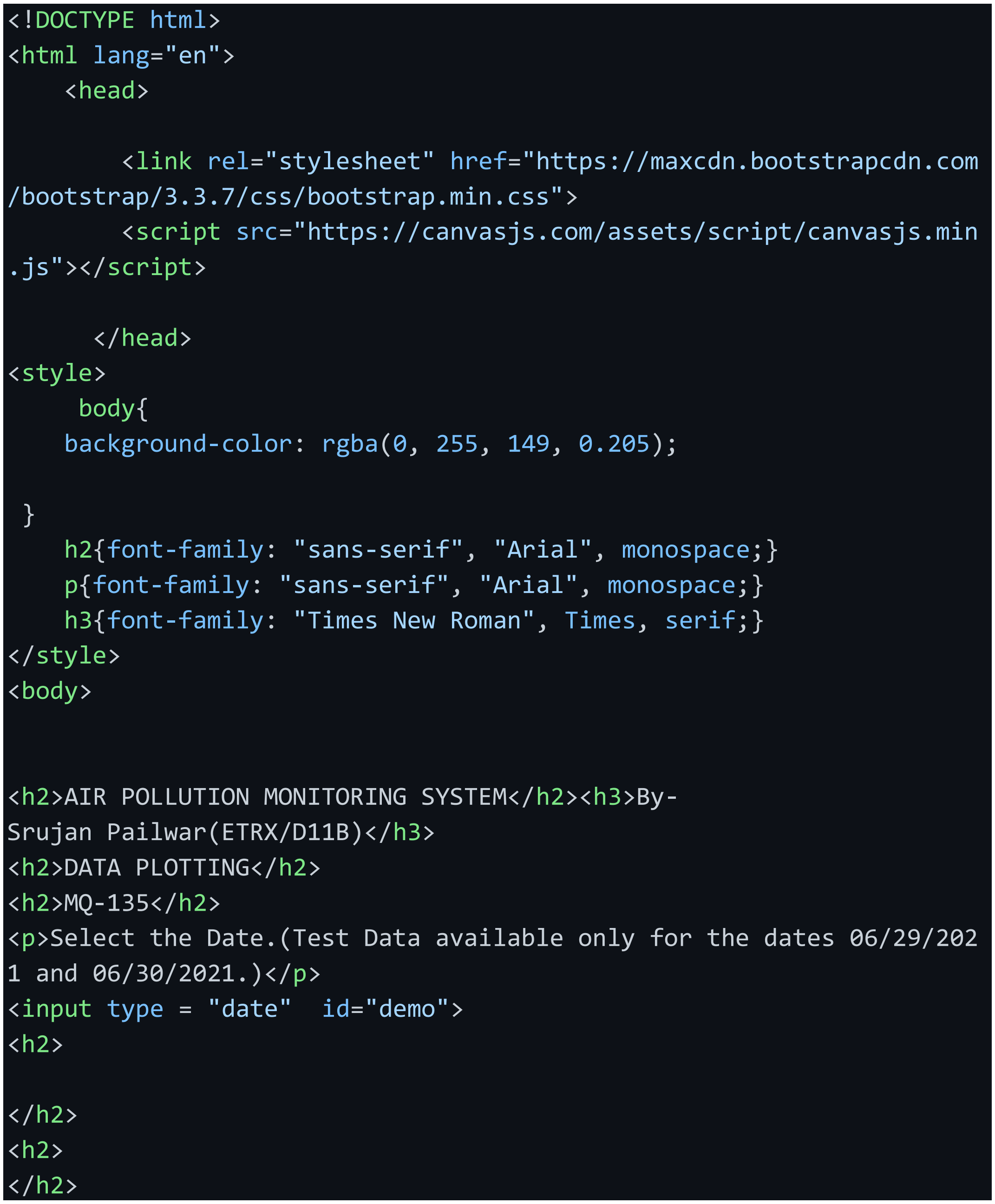
c. IoT Based Air Pollution Monitoring System – Arduino – DIY Electronics Projects (electronics- project-hub.com)

d. Priyanka, V., “Review: Air Quality Monitoring System," International Journal of Advanced Research in Computer and Communication Engineering, 5(6), 2016.

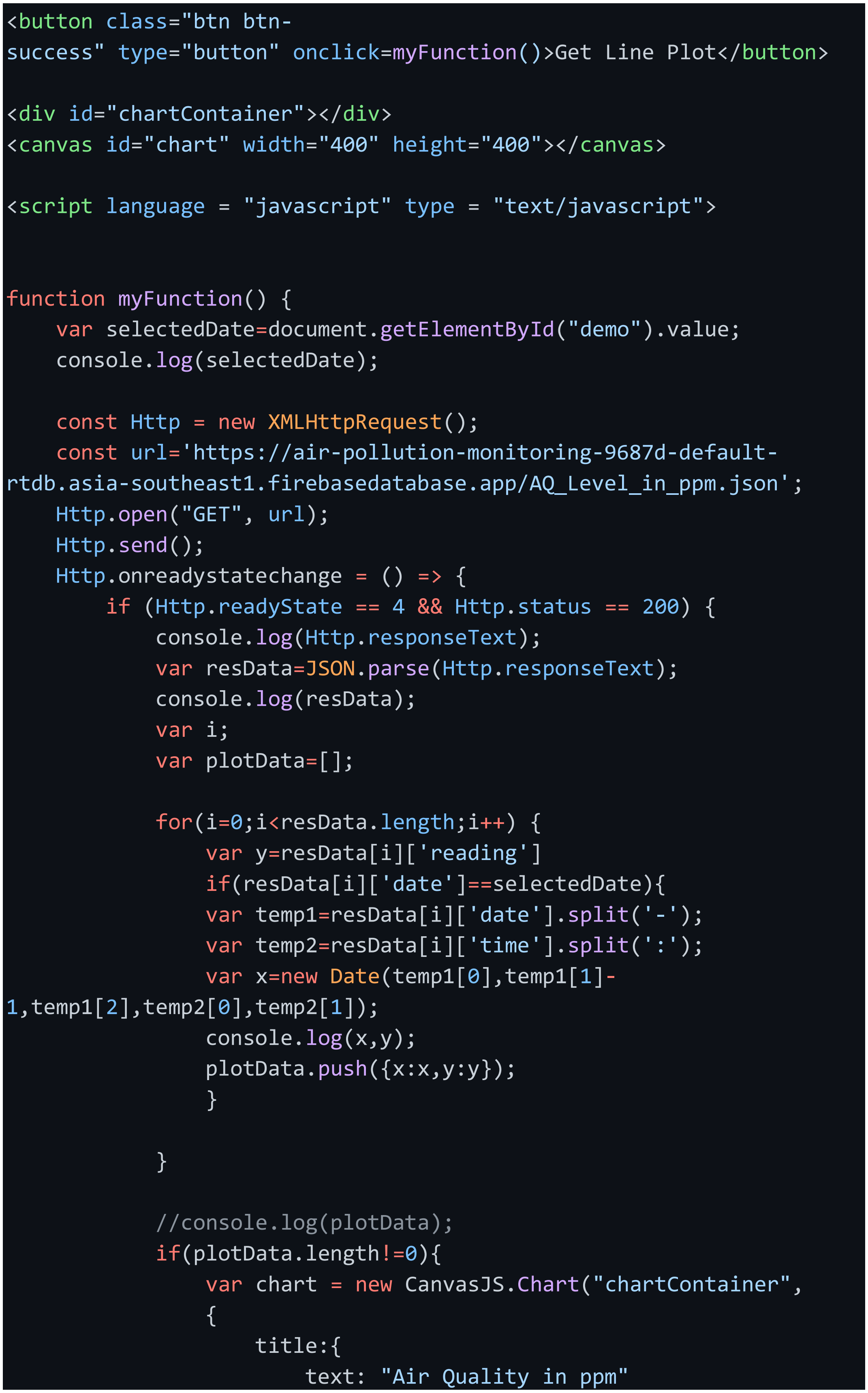
11. Appendix

a. Source code (web app)

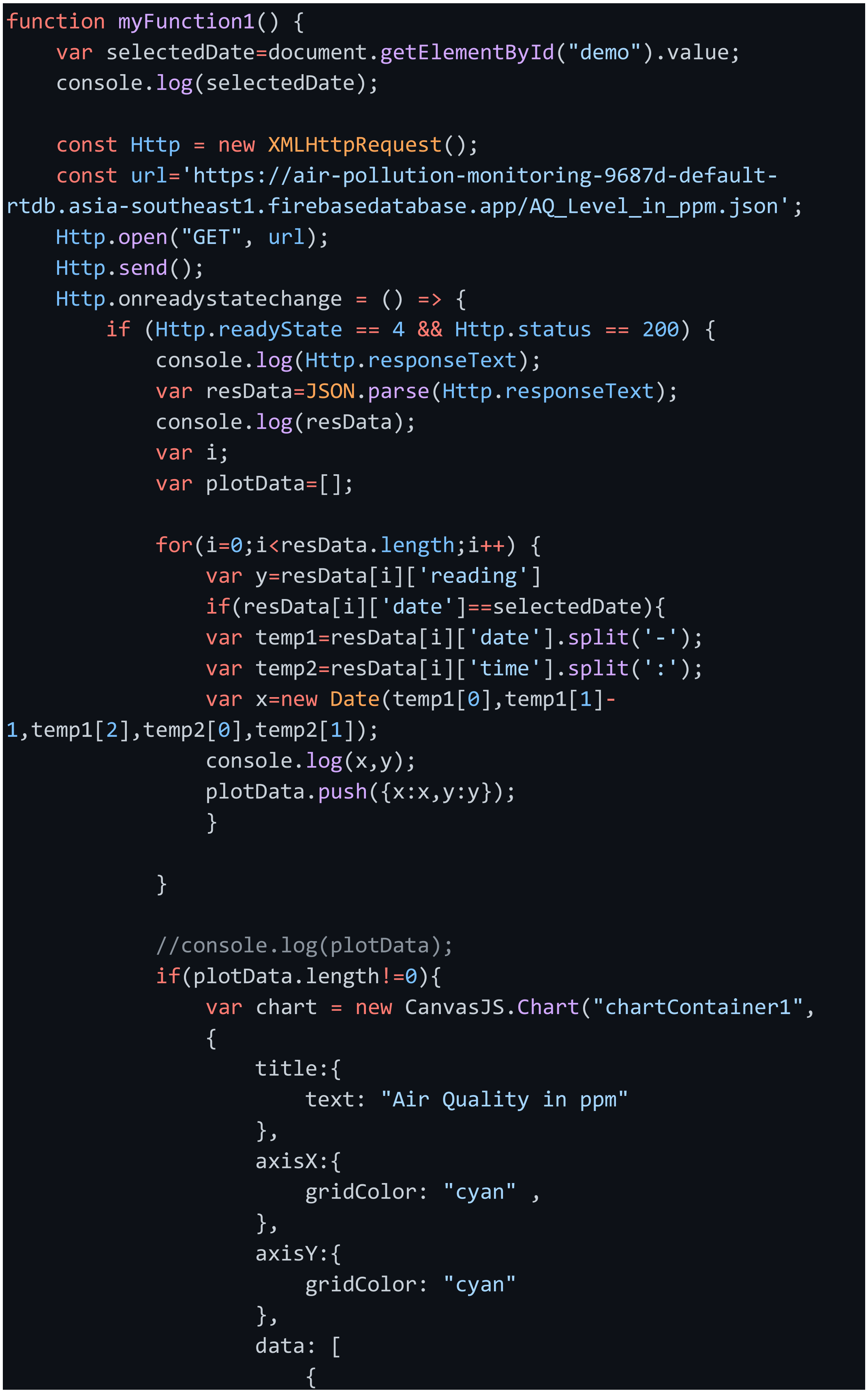
- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/main/index.html
- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/a63b54b4d0e4ae1cb680424f11339fef68a14185/index.html



```
<!DOCTYPE html>
<html lang="en">
    <head>

        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com
/bootstrap/3.3.7/css/bootstrap.min.css">
        <script src="https://canvasjs.com/assets/script/canvasjs.min
.js"></script>

      </head>
<style>
     body{
    background-color: rgba(0, 255, 149, 0.205);

 }
    h2{font-family: "sans-serif", "Arial", monospace;}
    p{font-family: "sans-serif", "Arial", monospace;}
    h3{font-family: "Times New Roman", Times, serif;}
</style>
<body>


<h2>AIR POLLUTION MONITORING SYSTEM</h2><h3>By-
Srujan Pailwar(ETRX/D11B)</h3>
<h2>DATA PLOTTING</h2>
<h2>MQ-135</h2>
<p>Select the Date.(Test Data available only for the dates 06/29/202
1 and 06/30/2021.)</p>
<input type = "date"  id="demo">
<h2>

</h2>
<h2>
</h2>
```



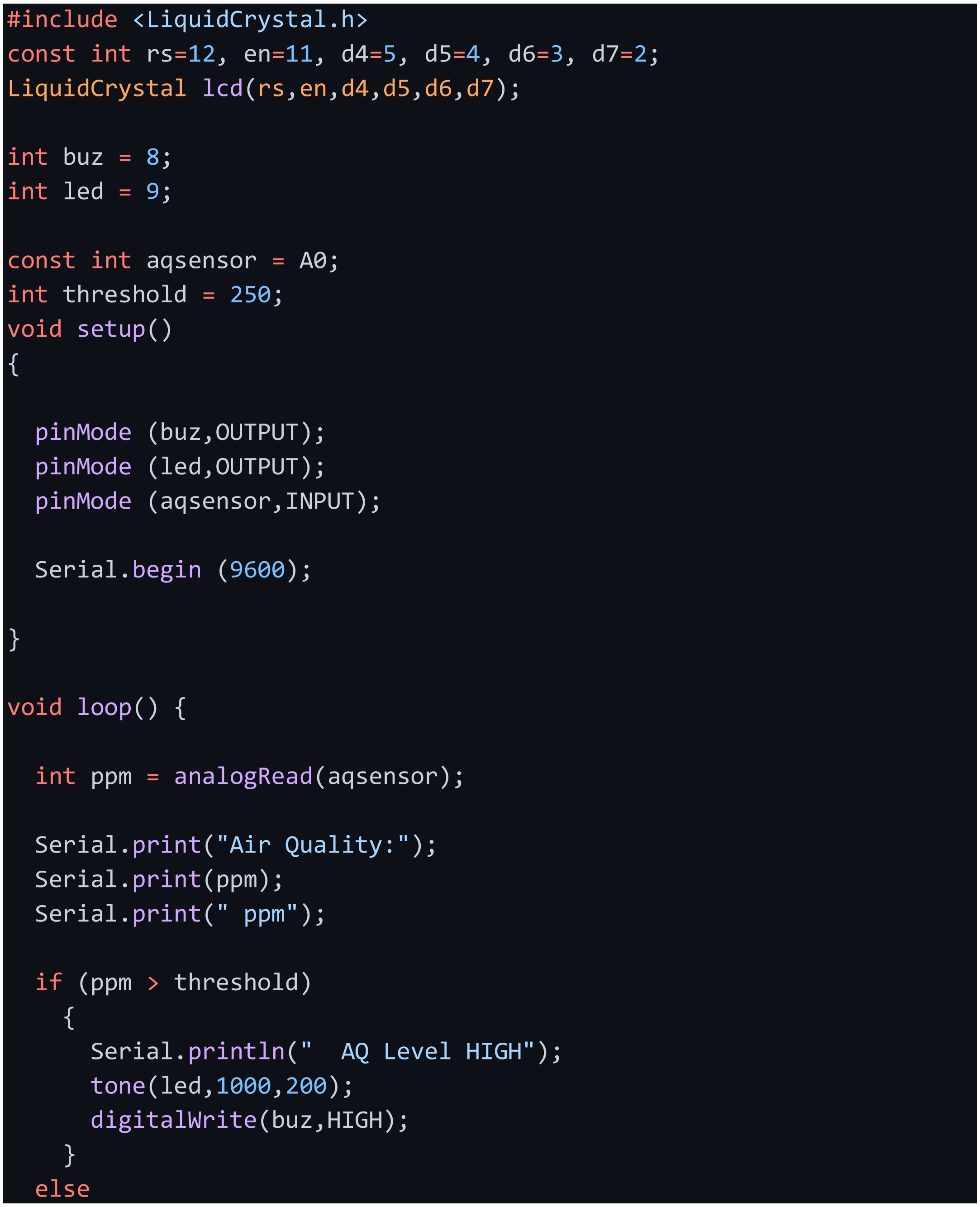
```
<button class="btn btn-
success" type="button" onclick=myFunction()>Get Line Plot</button>

<div id="chartContainer"></div>
<canvas id="chart" width="400" height="400"></canvas>

<script language = "javascript" type = "text/javascript">


function myFunction() {
    var selectedDate=document.getElementById("demo").value;
    console.log(selectedDate);

    const Http = new XMLHttpRequest();
    const url='https://air-pollution-monitoring-9687d-default-
rtdb.asia-southeast1.firebasedatabase.app/AQ_Level_in_ppm.json';
    Http.open("GET", url);
    Http.send();
    Http.onreadystatechange = () => {
        if (Http.readyState == 4 && Http.status == 200) {
            console.log(Http.responseText);
            var resData=JSON.parse(Http.responseText);
            console.log(resData);
            var i;
            var plotData=[];

            for(i=0;i<resData.length;i++) {
                var y=resData[i]['reading']
                if(resData[i]['date']==selectedDate){
                var temp1=resData[i]['date'].split('-');
                var temp2=resData[i]['time'].split(':');
                var x=new Date(temp1[0],temp1[1]-
1,temp1[2],temp2[0],temp2[1]);
                console.log(x,y);
                plotData.push({x:x,y:y});
                }

            }

            //console.log(plotData);
            if(plotData.length!=0){
                var chart = new CanvasJS.Chart("chartContainer",
                {
                    title:{
                        text: "Air Quality in ppm"
                    },
                    axisX:{
                        gridColor: "cyan" ,
                    },
                    axisY:{
                        gridColor: "cyan"
                    },
                    data: [
                        {
                            type: "line",

                            dataPoints: plotData
                        }

                    ]
                });

                chart.render();
            }
            else{
                document.getElementById("chartContainer").innerHTML=
"No data available for this date."
            }
        }
    }
    Http.open("GET", url);

    Http.send();


}


</script>
<h2>

</h2>
<button class="btn btn-
success" type="button" onclick=myFunction1()>Get Bar Plot</button>
 <div id="chartContainer1"></div>
    <canvas id="chart" width="400" height="400"></canvas>

<script language = "javascript" type = "text/javascript">
```



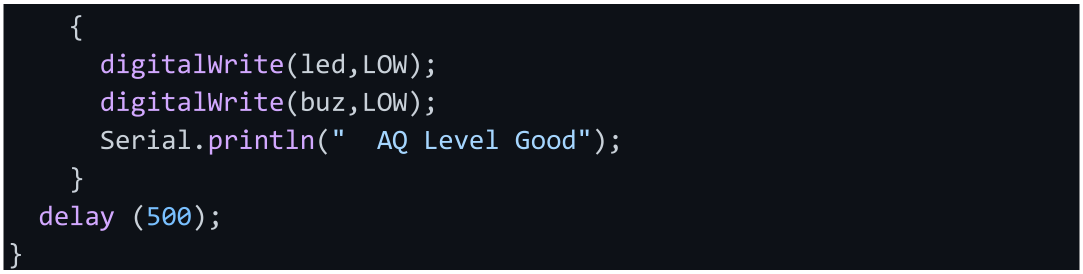
```
function myFunction1() {
    var selectedDate=document.getElementById("demo").value;
    console.log(selectedDate);

    const Http = new XMLHttpRequest();
    const url='https://air-pollution-monitoring-9687d-default-
rtdb.asia-southeast1.firebasedatabase.app/AQ_Level_in_ppm.json';
    Http.open("GET", url);
    Http.send();
    Http.onreadystatechange = () => {
        if (Http.readyState == 4 && Http.status == 200) {
            console.log(Http.responseText);
            var resData=JSON.parse(Http.responseText);
            console.log(resData);
            var i;
            var plotData=[];

            for(i=0;i<resData.length;i++) {
                var y=resData[i]['reading']
                if(resData[i]['date']==selectedDate){
                var temp1=resData[i]['date'].split('-');
                var temp2=resData[i]['time'].split(':');
                var x=new Date(temp1[0],temp1[1]-
1,temp1[2],temp2[0],temp2[1]);
                console.log(x,y);
                plotData.push({x:x,y:y});
                }

            }

            //console.log(plotData);
            if(plotData.length!=0){
                var chart = new CanvasJS.Chart("chartContainer1",
                {
                    title:{
                        text: "Air Quality in ppm"
                    },
                    axisX:{
                        gridColor: "cyan" ,
                    },
                    axisY:{
                        gridColor: "cyan"
                    },
                    data: [
                        {
                            type: "bar",

                            dataPoints: plotData
                        }

                    ]
                });

                chart.render();
            }
            else{
                document.getElementById("chartContainer1").innerHTML
="No data available for this date."
            }
        }
    }
    Http.open("GET", url);

    Http.send();
}

</script>


</body>
</html>
```

b. Source code (Arduino code)

- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/main/Air%20Pollution%20Monitoring%20System.ino
- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/a63b54b4d0e4ae1cb680424f11339fef68a14185/Air%20Pollution%20Monitoring%20System.ino



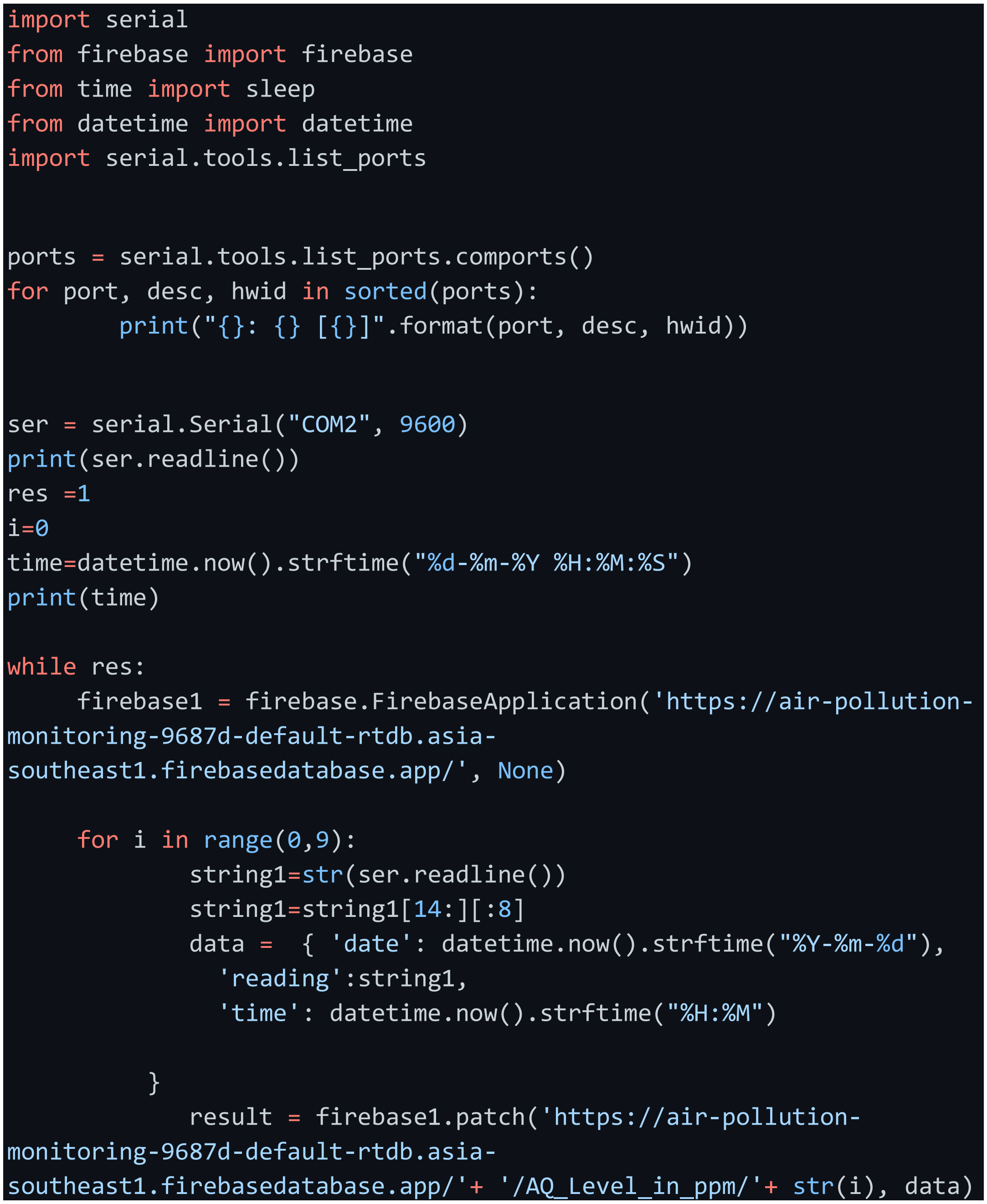
```
#include <LiquidCrystal.h>
const int rs=12, en=11, d4=5, d5=4, d6=3, d7=2;
LiquidCrystal lcd(rs,en,d4,d5,d6,d7);

int buz = 8;
int led = 9;

const int aqsensor = A0;
int threshold = 250;
void setup()
{

  pinMode (buz,OUTPUT);
  pinMode (led,OUTPUT);
  pinMode (aqsensor,INPUT);

  Serial.begin (9600);

}

void loop() {

  int ppm = analogRead(aqsensor);

  Serial.print("Air Quality:");
  Serial.print(ppm);
  Serial.print(" ppm");

  if (ppm > threshold)
    {
      Serial.println("  AQ Level HIGH");
      tone(led,1000,200);
      digitalWrite(buz,HIGH);
    }
  else
```



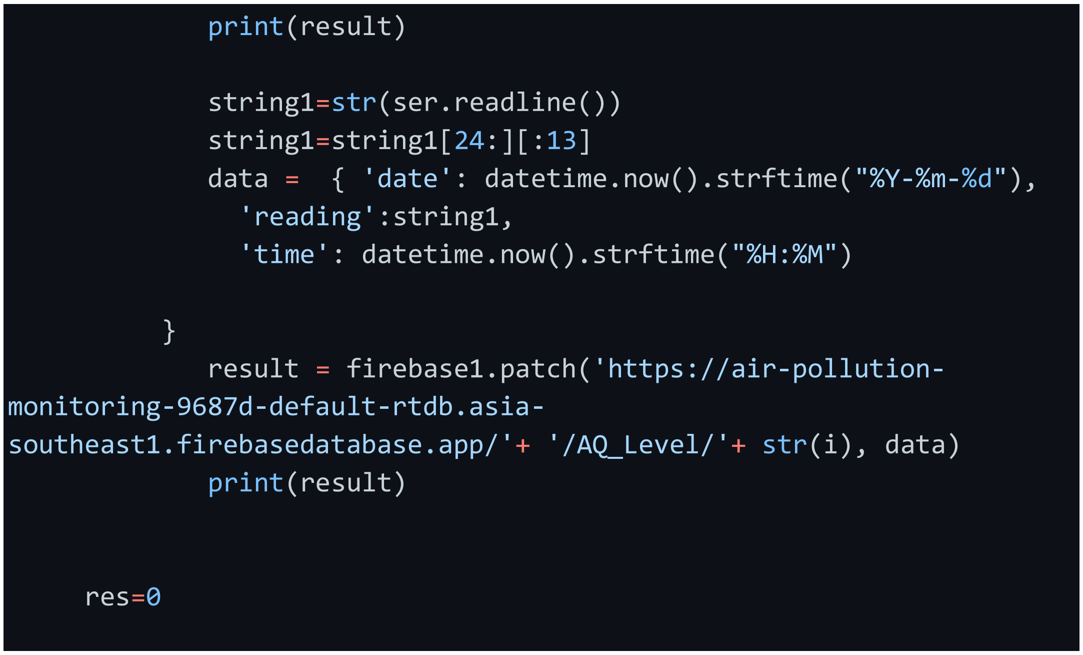
```
    {
      digitalWrite(led,LOW);
      digitalWrite(buz,LOW);
      Serial.println("  AQ Level Good");
    }
  delay (500);
}
```

c. Source code (Python data upload)

- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/main/Air%20Pollution%20Monitoring%20System.py
- https://github.com/Tinkerers-Lab-VESIT-ETRX/IoT-based-air-pollution-monitoring-system-12/blob/a63b54b4d0e4ae1cb680424f11339fef68a14185/Air%20Pollution%20Monitoring%20System.py



```
import serial
from firebase import firebase
from time import sleep
from datetime import datetime
import serial.tools.list_ports


ports = serial.tools.list_ports.comports()
for port, desc, hwid in sorted(ports):
        print("{}: {} [{}]".format(port, desc, hwid))


ser = serial.Serial("COM2", 9600)
print(ser.readline())
res =1
i=0
time=datetime.now().strftime("%d-%m-%Y %H:%M:%S")
print(time)

while res:
     firebase1 = firebase.FirebaseApplication('https://air-pollution-
monitoring-9687d-default-rtdb.asia-
southeast1.firebasedatabase.app/', None)

     for i in range(0,9):
             string1=str(ser.readline())
             string1=string1[14:][:8]
             data =  { 'date': datetime.now().strftime("%Y-%m-%d"),
               'reading':string1,
               'time': datetime.now().strftime("%H:%M")

          }
             result = firebase1.patch('https://air-pollution-
monitoring-9687d-default-rtdb.asia-
southeast1.firebasedatabase.app/'+ '/AQ_Level_in_ppm/'+ str(i), data)
```



```
             print(result)

             string1=str(ser.readline())
             string1=string1[24:][:13]
             data =  { 'date': datetime.now().strftime("%Y-%m-%d"),
               'reading':string1,
               'time': datetime.now().strftime("%H:%M")

          }
             result = firebase1.patch('https://air-pollution-
monitoring-9687d-default-rtdb.asia-
southeast1.firebasedatabase.app/'+ '/AQ_Level/'+ str(i), data)
             print(result)


     res=0
```