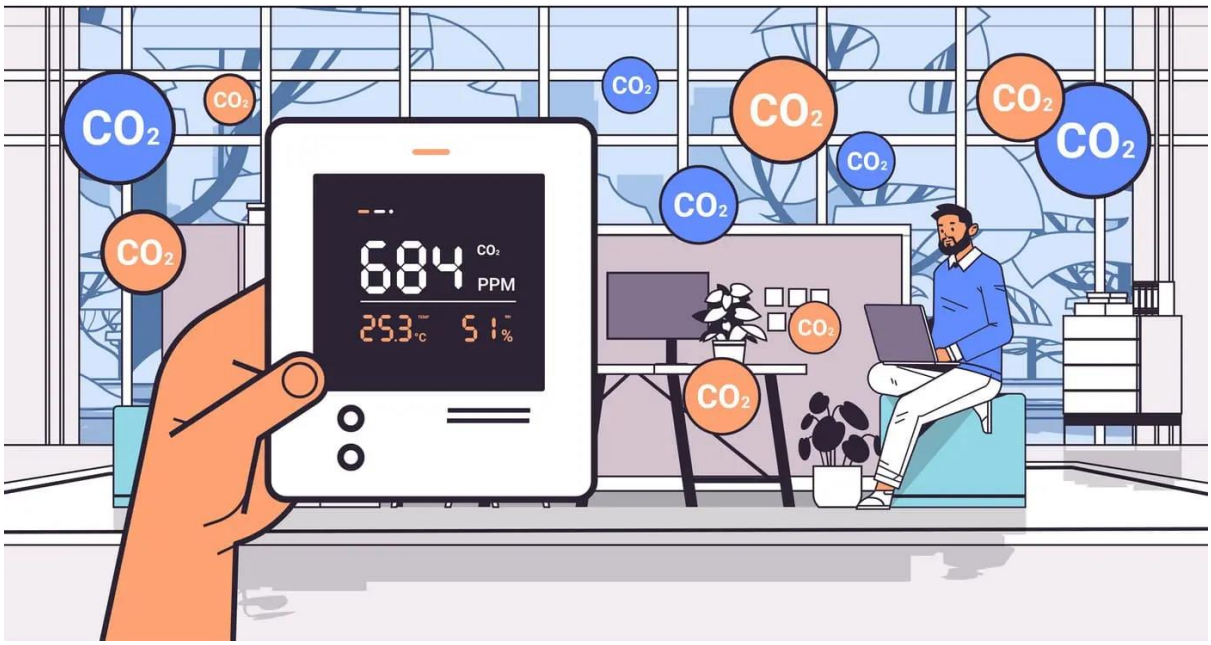
# IOT PROJECT TITLE:

## Advancing Air Quality Monitoring using NodeMCU and ThingSpeak




**Prepared by:**

★ **S.SAMEERA TASNEEM : BU21EECE0100100**

★ **R.INDUMATHI : BU21EECE0100360**

★ **S.SAI LOHITHA : BU21EECE0100103**

# AGENDA

- **OVERVIEW**
- **BILL OF MATERIALS**
- **INTRODUCTION TO AIR QUALITY MONITORING**
- **COMPONENTS USED**
- **SOURCE CODE**
- **TESTING AND RESULTS**
- **CONCLUSION**

## Overview

The AQI is an index that tells you how clean or polluted your air is, and what associated health effects might be a concern for you.

The level of pollution has increased with times by a lot of factors like the increase in population, increased vehicle use, industrialization and urbanization which results in harmful effects on humans by directly affecting the health of the population exposed to it. So, we need to monitor the Air Quality Index. In this project, we are going to make an IoT Based Air Quality Index Monitoring System in which we will monitor the Air Quality Index over a Thingspeak server using the internet. We will use MQ135 Air Quality Sensor that can detect the level of various air pollutant.

---

## Bill of Materials

Following are the components required for making this project.

| S.N. | Components Name | Quantity |
|---|---|---|
| 1 | NodeMCU ESP8266 | 1 |
| 2 | 1602 LCD Display | 1 |
| 3 | Connecting Wires | 7 |
| 4 | MQ-135 Air Quality Sensor | 1 |

---

## Introduction To Air Quality Monitoring

The AQI is an index for reporting daily air quality. It tells you how clean or polluted your air is, and what associated health effects might be a concern for you. The AQI focuses on health affects you may experience within a few hours or days after breathing polluted air.

EPA calculates the AQI for five major air pollutants regulated by the Clean Air Act: ground-level ozone, particle pollution (also known as particulate matter), carbon monoxide, sulfur dioxide, and nitrogen dioxide. For each of these pollutants, EPA has established national air quality standards to protect public health. Ground-level ozone and airborne particles are the two pollutants that pose the greatest threat to human health in this country.

| Air Quality Index Levels of Health Concern | Numerical Value | Meaning |
|---|---|---|
| Good | 0-50 | Air quality is considered satisfactory, and air pollution poses little or no risk. |
| Moderate | 51-100 | Air quality is acceptable; however, for some pollutants there may be a moderate health concern for a very small number of people who are unusually sensitive to air pollution. |
| Unhealthy for Sensitive Groups | 101-150 | Members of sensitive groups may experience health effects. The general public is not likely to be affected. |
| Unhealthy | 151-200 | Everyone may begin to experience health effects; members of sensitive groups may experience more serious health effects. |
| Very Unhealthy | 201-300 | Health alert: everyone may experience more serious health effects. |
| Hazardous | > 300 | Health warnings of emergency conditions. The entire population is more likely to be affected. |

Think of the AQI as a yardstick that runs from 0 to 500. The higher the AQI value, the greater the level of air pollution and the greater the health concern. For example, an AQI value of 50 represents good air quality with little potential to affect public health, while an AQI value over 300 represents hazardous air quality.

---

## MQ135 Air Quality Sensor

The **MQ-135** gas sensor senses the gases like **ammonia nitrogen, oxygen, alcohols, aromatic compounds, sulphid and smoke**. The MQ-3 gas sensor has a lower conductivity to clean the air as a gas sensing material. In the atmosphere, we can find polluting gases, but the conductivity of the gas sensor increases as the concentration of polluting gas increases. MQ-135 gas sensor can be implemented to detect the **smoke, benzene, steam and other harmful gases**. It has the potential to detect different harmful gases. It is with low cost and particularly suitable for Air quality monitoring application.

The **MQ135 sensor** is a signal output indicator instruction. It has two outputs: analog output and TTL output. The **TTL output** is low signal light which can be accessed through the IO ports on the Microcontroller. The **analog output** is an concentration, i.e. increasing voltage is directly proportional to increasing concentration. This sensor has a long life and reliable stability as well.

---

# 1602 LCD

### 16×2 Character LCD Pinout

Before we get into, let's check out the pinout. A standard character LCD has 16 pins (except for an RGB LCD, which has 18 pins).


GND is the ground pin.

VCC is the LCD's power supply and is typically connected to 5 volts.

Vo (LCD Contrast) pin controls the contrast of the LCD. Using a simple voltage divider network and a potentiometer, we can make precise contrast adjustments.

RS (Register Select) pin is used to separate the commands (such as setting the cursor to a specific location, clearing the screen, etc.) from the data. The RS pin is set to LOW when sending commands to the LCD and HIGH when sending data.

R/W (Read/Write) pin allows you to read data from or write data to the LCD. Since the LCD is only used as an output device, this pin is typically held low. This forces the LCD into WRITE mode.

E (Enable) pin is used to enable the display. When this pin is set to LOW, the LCD ignores activity on the R/W, RS, and data bus lines; when it is set to HIGH, the LCD processes the incoming data.

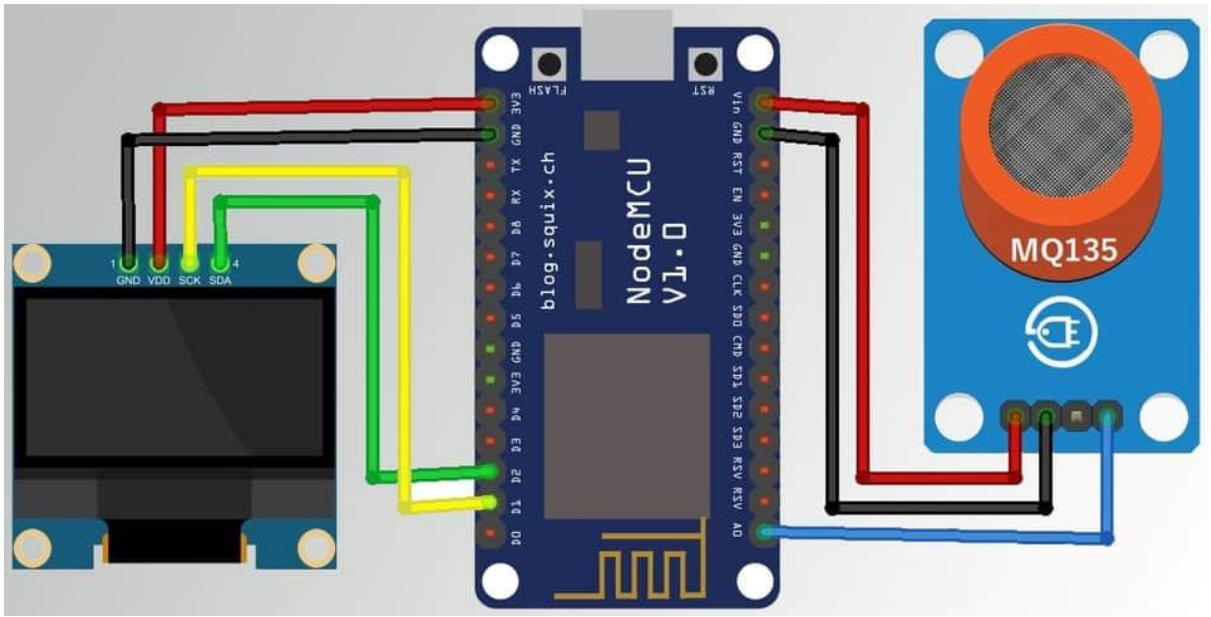
D0-D7 (Data Bus) pins carry the 8 bit data we send to the display. To see an uppercase 'A' character on the display, for example, we set these pins to 0100 0001 (as per the ASCII table).

A-K (Anode & Cathode) pins are used to control the backlight of the LCD.

---

## Air Quality Index Monitoring with ESP8266 & MQ135

Now let us assemble the hardware and do the coding for the great IoT project. We will interface MQ135 Air Quality Sensor with NodeMCU ESP8266 Board and 0.96" I2C OLED Display. The circuit diagram is given below.



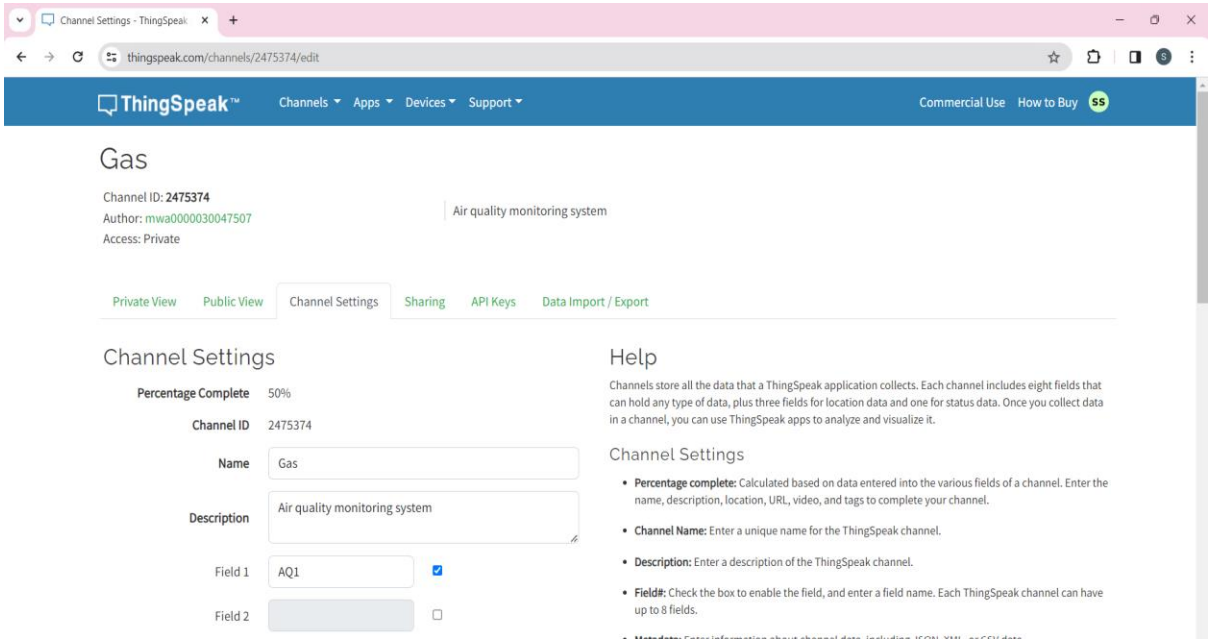
You can assemble the circuit in a breadboard. First, connect the MQ135 Analog input pin to A0 of NodeMCU. Then connect its VCC and GND to NodeMCU Vin & GND respectively. Similarly, 0.96" OLED Display is an I2C Module. So, connect its SDA & SCL Pin to Nodemcu D2 & D1 Pins. Connect its VCC to 3.3V GND to GND.

---

## Setting up Thingspeak

ThingSpeak is an open-source Internet of Things application and API to store and retrieve data from things using the HTTP and MQTT protocol over the Internet or via a Local Area Network. Thingspeak enables you to collect, store, analyse, visualize, and act on data from sensors.

To setup the Thingspeak Server, visit https://thingspeak.com/. Create an account or simply sign in if you created the account earlier. Then create a new channel.




Copy the API key and use it in the code below.

---

## Source Code/Program

The source code/program for making IoT Based Air Quality Index Monitoring with ESP8266 & MQ135 is given below. In the below code you need to make changes to Wifi SSID & Password. Also make sure to make changes to API Key from Thingspeak.

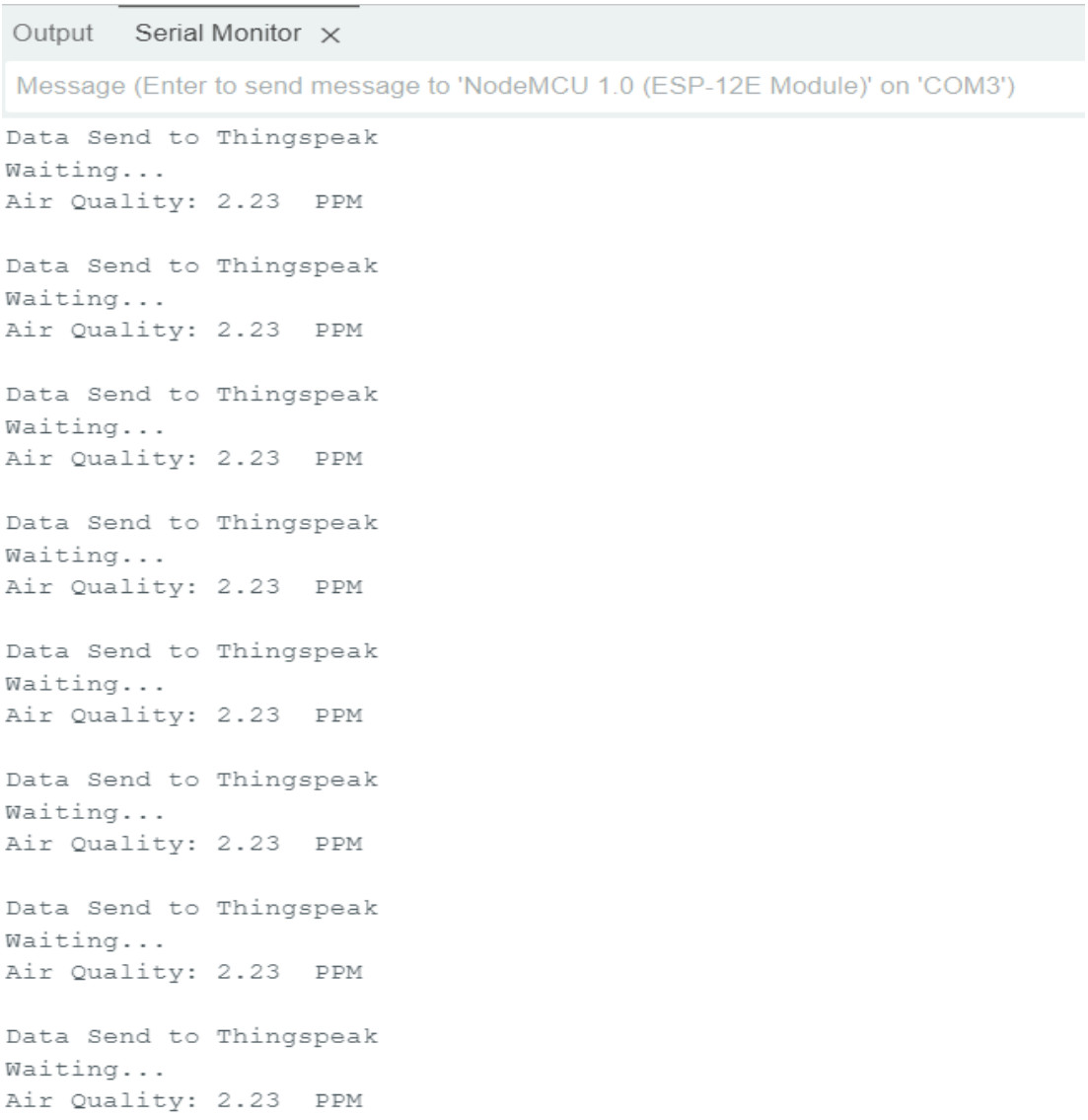
```
#include <ESP8266WiFi.h>
#include <SPI.h>
#include <Wire.h>
#include "MQ135.h"
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128    // OLED display width, in pixels
#define SCREEN_HEIGHT 64    // OLED display height, in pixels
#define OLED_RESET -1      // Reset pin # (or -1 if sharing Arduino reset pin)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

String apiKey = "K0D9UCBRGKSE4MYH"; // Enter your Write API key from ThingSpeak
const char *ssid = "GITAM-5GHz";    // replace with your wifi ssid and wpa2 key
const char *pass = "Gitam$$123";
const char* server = "api.thingspeak.com";

WiFiClient client;


void setup()
{
 Serial.begin(115200);
 display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //initialize with the I2C addr 0x3C (128x64)
 display.clearDisplay();
 delay(10);

 Serial.println("Connecting to ");
 Serial.println(ssid);

 display.clearDisplay();
 display.setCursor(0,0);
 display.setTextSize(1);
```

```
display.setTextColor(WHITE);
display.println("Connecting to ");
display.setTextSize(2);
display.print(ssid);
display.display();

WiFi.begin(ssid, pass);

while (WiFi.status() != WL_CONNECTED)
{
 delay(500);
 Serial.print(".");
}
 Serial.println("");
 Serial.println("WiFi connected");

 display.clearDisplay();
 display.setCursor(0,0);
 display.setTextSize(1);
 display.setTextColor(WHITE);
 display.print("WiFi connected");
 display.display();
 delay(4000);
}


void loop()
{
 MQ135 gasSensor = MQ135(A0);
 float air_quality = gasSensor.getPPM();
 Serial.print("Air Quality: ");
 Serial.print(air_quality);
 Serial.println("  PPM");
 Serial.println();

 display.clearDisplay();
 display.setCursor(0,0);  //oled display
 display.setTextSize(1);
 display.setTextColor(WHITE);
 display.println("Air Quality Index");


```

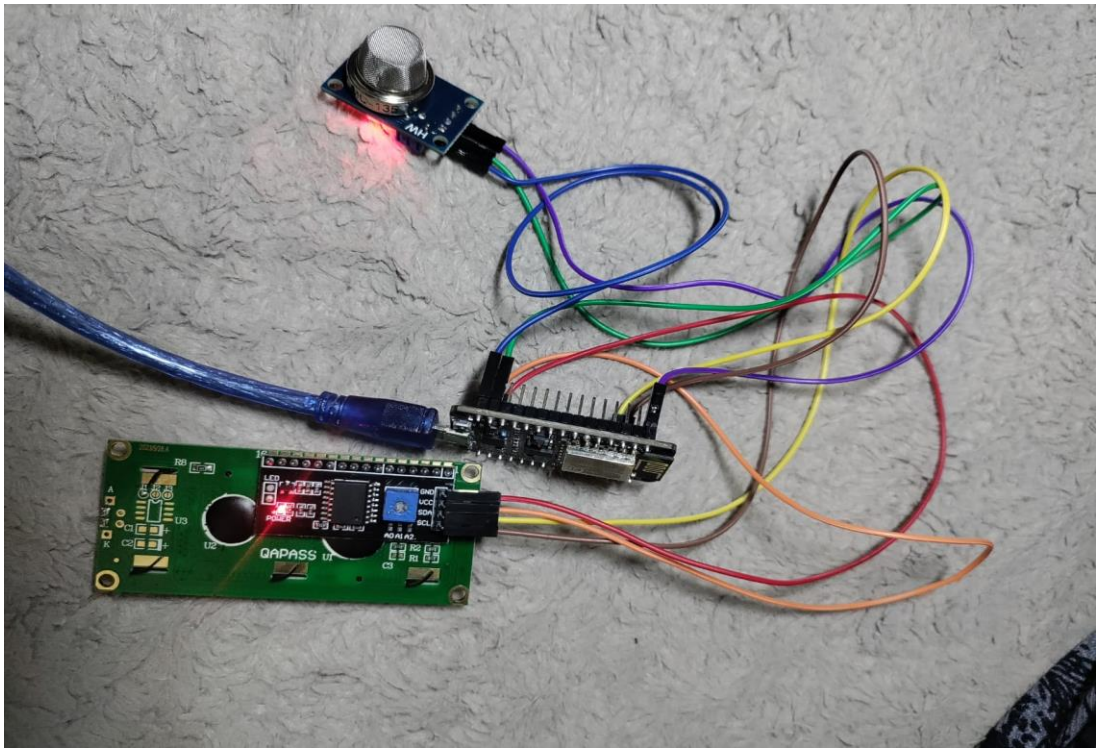
```
  display.setCursor(0,20); //oled display
  display.setTextSize(2);
  display.setTextColor(WHITE);
  display.print(air_quality);
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.println(" PPM");
  display.display();


  if (client.connect(server, 80)) // "184.106.153.149" or api.thingspeak.com
 {
  String postStr = apiKey;
  postStr += "&field1=";
  postStr += String(air_quality);
  postStr += "r\n";

  client.print("POST /update HTTP/1.1\n");
  client.print("Host: api.thingspeak.com\n");
  client.print("Connection: close\n");
  client.print("X-THINGSPEAKAPIKEY: " + apiKey + "\n");
  client.print("Content-Type: application/x-www-form-urlencoded\n");
  client.print("Content-Length: ");
  client.print(postStr.length());
  client.print("\n\n");
  client.print(postStr);

  Serial.println("Data Send to Thingspeak");
 }
  client.stop();
  Serial.println("Waiting...");

  delay(2000);    // thingspeak needs minimum 15 sec delay between updates.
}
```

## Testings & Results

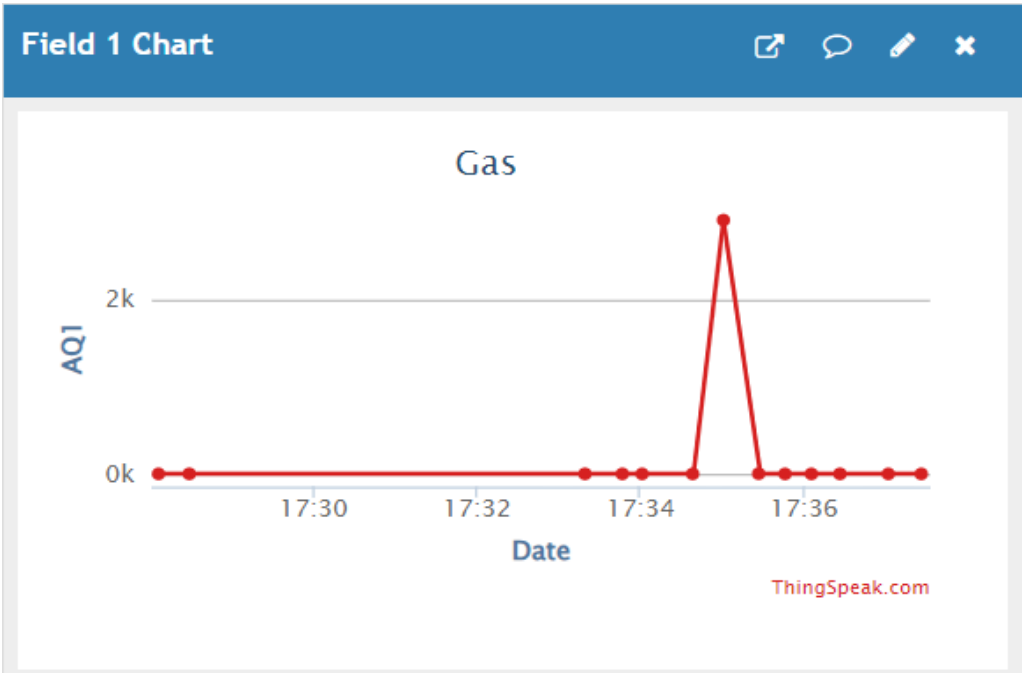
Once the code is uploaded you can open serial monitor. The Nodemcu will first start connecting to wifi network. All the happening can se observed on Serial Monitor.




Once connected to a Wifi network, the sensor will read the value and the value will be displayed on LCD Screen.



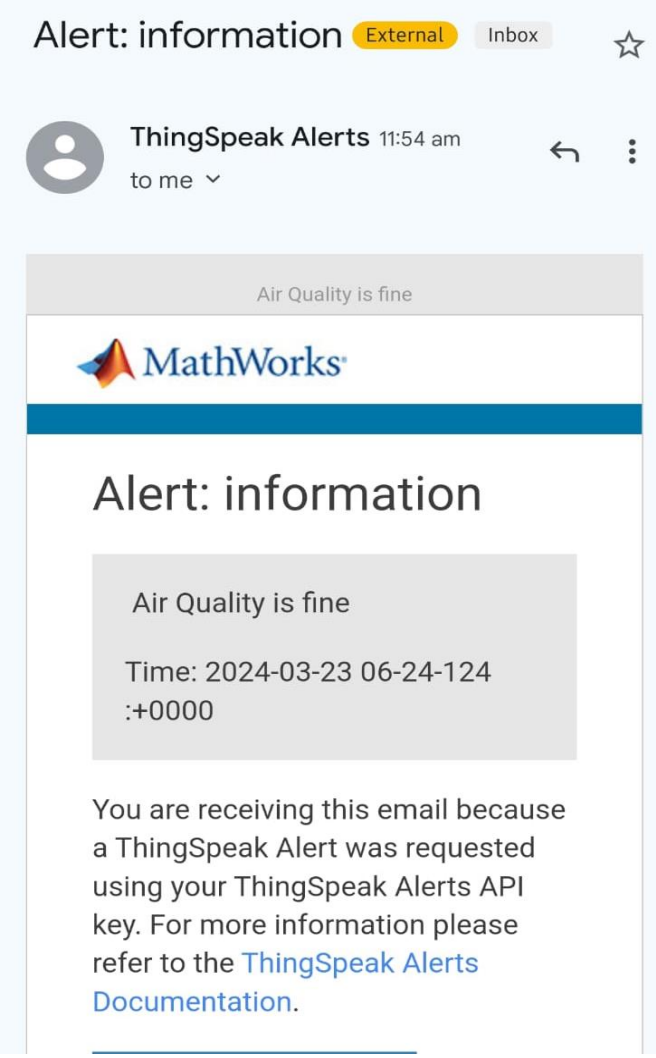
Similarly you can see the online data of Air Quality Index on Thingspeak Server. Just go to Thingspeak Private view and check the data being uploaded after the interval of 15 seconds.



## Alerts through email:

ThingSpeak is an IoT platform that allows users to collect, analyse, and visualize data from sensors or other devices. One of its features is the ability to set up alerts based on predefined conditions. These alerts can be configured to notify users via email when certain thresholds or criteria are met within the data being monitored. This allows users to stay informed about important events or changes in their data remotely, enhancing their ability to monitor and manage IoT systems effectively.

To set up ThingSpeak alerts through email, first, define conditions using MATLAB Analysis to trigger alerts based on data patterns or thresholds. Then, access the React App section on ThingSpeak and create a new React app. Configure the React app to monitor MATLAB Analysis results and trigger alerts accordingly. Specify email notification settings within the React app to receive alerts via email. Save your React app settings and activate it to begin monitoring data and sending alerts. Finally, validate your setup by intentionally triggering conditions to ensure that email alerts are received as expected.



## Conclusion:

Innovative hardware solutions offer immense potential to advance air quality monitoring systems, enhancing data accuracy, real-time capabilities, and scalability. Through effective integration of hardware components, stakeholders can develop resilient monitoring infrastructure capable of addressing the complex challenges of urban air pollution and supporting evidence-based decision-making for public health and environmental protection.

## Reference:

**IoT Based Air Pollution Monitoring System**

A Banerjee, B Sikdar, S Roy - A Project report submitted in partial …, 2022 - rcciit.org

It is our great fortune that we have got an opportunity to carry out this project work under the supervision of Dr. Debasish Mondal in the Department of Electrical Engineering

## VIDEO LINK:

https://drive.google.com/drive/folders/1zx857-I0VIztJ9dN-Aa0tAgb4I6uxIpH?usp=drive_link