**AIR QUALITY MANAGEMENT**

PHASE 3 : **DEVELOPMENT PART 1**

**ABSTRACT:**

* **Internet of Things (IoT) may be a worldwide system of**

**“smart devices” which will sense and connect with their surroundings   
 and interact with users and other systems.**

* **Global air pollution is one of the major concerns of our era. The level of pollution has increased with times by lot of things like the increase in population, increased vehicle use, industrialization and urbanization which ends up in harmful effects on human wellbeing by directly affecting health of population exposed to it.**
* **Air quality goes down when enough amount of   
   harmful gases present in the air like carbon dioxide,   
   smoke, alcohol, benzene, NH3,and NO2.**
* **In order to analyses we are developing an IOT Based   
  pollution Monitoring System which we'll monitorthe   
  Air Quality over an internet server.**
* **The system uses MQ2 and MQ7 sensor for monitoring Air Quality. It measures their amount exactly and res = int(res\_data)**

**PYTHON SCRIPT :**

1. **if res <= 50:**
2. **remark = "Good"**
3. **impact = "Minimal impact”**
4. **elif res <= 100 and res > 51:**
5. **remark = "Satisfactory"**
6. **impact = "Minor breathing discomfort to sensitive   
    people"**
7. **elif res <= 200 and res >= 101:**
8. **remark = "Moderate"**
9. **impact = "Breathing discomfort to the people with   
    lungs, asthma and heart diseases"**
10. **elif res <= 400 and res >= 201:**
11. **remark = "Very Poor"**
12. **impact = "Breathing discomfort to most people on   
     prolonged exposure"**
13. **elif res <= 500 and res >= 401:**
14. **remark = "Severe"**
15. **impact = "Affects healthy people and   
     seriously impacts those with existing   
     diseases”**
16. **print(remark)**
17. **print(impact)finds out harmful gases.**

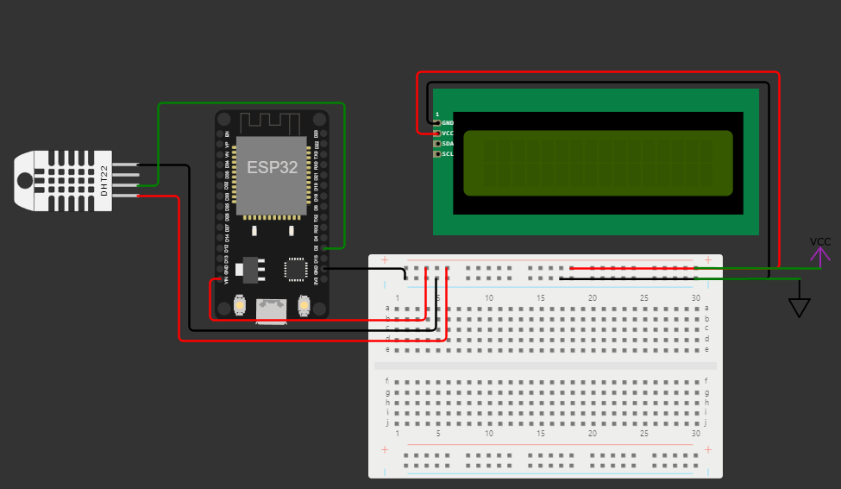
**ARDUINO UNO R3 :**

1. **#include <DHT.h>**
2. **#include <WiFi.h>**
3. **#include <ThingSpeak.h>**
4. **#include "DHTesp.h"**
5. **#define DHT\_PIN 2  // Replace with the GPIO pin connected to the DHT22 sensor**
6. **#define LED\_GREEN\_PIN 21  // Replace with the GPIO pin connected to the green LED bulb**
7. **#define LED\_RED\_PIN 22  // Replace with the GPIO pin connected to the red LED bulb**
8. **char ssid[] = "Wokwi-GUEST";**
9. **char pass[] = "";**
10. **WiFiClient client;**
11. **unsigned long myChannelNumber = 2308799;**
12. **const char \*myWriteAPIKey = "Y5D386LU3W5X66Y2";**
13. **int statusCode;**
14. **DHTesp dhtSensor;**
15. **int ledGreen = LED\_GREEN\_PIN;**
16. **int ledRed = LED\_RED\_PIN;**
17. **struct Data {**
18. **float temperature;**
19. **float humidity;**
20. **};**
21. **Data data;  // Declare a variable to store the data**
22. **void setup() {**
23. **Serial.begin(115200);**
24. **WiFi.mode(WIFI\_STA);**
25. **ThingSpeak.begin(client);**
26. **dhtSensor.setup(DHT\_PIN, DHTesp::DHT22);**
27. **pinMode(ledGreen, OUTPUT);**
28. **pinMode(ledRed, OUTPUT);**
29. **}**
30. **void connectToCloud() {**
31. **if (WiFi.status() != WL\_CONNECTED) {**
32. **Serial.print("Attempting to connect");**
33. **while (WiFi.status() != WL\_CONNECTED) {**
34. **WiFi.begin(ssid, pass);**
35. **for (int i = 0; i < 5; i++) {**
36. **Serial.print(".");**
37. **delay(1000);**
38. **}**
39. **}**
40. **Serial.println("\nConnected.");**
41. **}**
42. **}**
43. **void computeData() {**
44. **TempAndHumidity sensorData = dhtSensor.getTempAndHumidity();**
45. **data.temperature = sensorData.temperature;**
46. **data.humidity = sensorData.humidity;**
47. **Serial.println("-----------");**
48. **Serial.println("Humi: " + String(data.humidity));**
49. **Serial.println("Temp: " + String(data.temperature));**
50. **Serial.println("-----------");**
51. **}**
52. **void writeData() {**
53. **ThingSpeak.setField(1, data.humidity);**
54. **ThingSpeak.setField(2, data.temperature);**
55. **statusCode = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);**
56. **if (statusCode == 200)**
57. **Serial.println("Channel update successful.");**
58. **else**
59. **Serial.println("Problem Writing data. HTTP error code: " + String(statusCode));**
60. **delay(15000);  // Data to be uploaded every 15 seconds**
61. **}**
62. **void loop() {**
63. **connectToCloud();**
64. **computeData();**
65. **writeData();**
66. **// Read temperature and humidity**
67. **float temperature = dhtSensor.getTemperature();**
68. **float humidity = dhtSensor.getHumidity();**
69. **// Print the results**
70. **Serial.print("Temperature: ");**
71. **Serial.print(temperature);**
72. **Serial.println("°C");**
73. **Serial.print("Humidity: ");**
74. **Serial.print(humidity);**
75. **Serial.println("%");**
76. **// Add a condition for air quality**
77. **if (temperature > 25.0 && humidity > 70.0) {**
78. **Serial.println("Air quality might be poor (high temperature and humidity).");**
79. **digitalWrite(ledRed, HIGH);**
80. **digitalWrite(ledGreen, LOW);**
81. **} else if (temperature < 20.0 && humidity < 30.0) {**
82. **Serial.println("Air quality might be poor (low temperature and humidity).");**
83. **digitalWrite(ledRed, HIGH);**
84. **digitalWrite(ledGreen, LOW);**
85. **} else {**
86. **Serial.println("Air quality appears to be normal.");**
87. **digitalWrite(ledRed, LOW);**
88. **digitalWrite(ledGreen, HIGH);**
89. **}**
90. **delay(10000);  // Wait for some time before the next measurement (10 seconds)**
91. **}**

**IOT DEVICE CODE :**

1. **{**
2. **"version": 1,**
3. **"author": "B. Mohan",**
4. **"editor": "wokwi",**
5. **"parts": [**
6. **{ "type": "wokwi-breadboard-half", "id": "bb1", "top": 93, "left": 185.2, "attrs": {} },**
7. **{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -43.3, "left": 43, "attrs": {} },**
8. **{**
9. **"type": "wokwi-dht22",**
10. **"id": "dht1",**
11. **"top": -33.6,**
12. **"left": -113.7,**
13. **"rotate": 270,**
14. **"attrs": {}**
15. **},**
16. **{**
17. **"type": "wokwi-lcd1602",**
18. **"id": "lcd1",**
19. **"top": -60.8,**
20. **"left": 245.6,**
21. **"attrs": { "pins": "i2c" }**
22. **},**
23. **{ "type": "wokwi-vcc", "id": "vcc1", "top": 77.56, "left": 595.2, "attrs": {} },**
24. **{ "type": "wokwi-gnd", "id": "gnd1", "top": 115.2, "left": 575.4, "attrs": {} }**
25. **],**
26. **"connections": [**
27. **[ "esp:TX0", "$serialMonitor:RX", "", [] ],**
28. **[ "esp:RX0", "$serialMonitor:TX", "", [] ],**
29. **[ "dht1:VCC", "bb1:tp.5", "red", [ "h38.4", "v134.4", "h259.2" ] ],**
30. **[ "dht1:GND", "bb1:tn.4", "black", [ "h48", "v153.6", "h230.4" ] ],**
31. **[ "dht1:SDA", "esp:D2", "green", [ "h19.2", "v-76.9", "h172.8", "v134.4" ] ],**
32. **[ "esp:VIN", "bb1:tp.3", "red", [ "h-9.6", "v38.4", "h153.6" ] ],**
33. **[ "esp:GND.1", "bb1:tn.1", "black", [ "h0" ] ],**
34. **[ "lcd1:GND", "bb1:tn.14", "black", [ "h-9.6", "v-38.4", "h316.8", "v182.4" ] ],**
35. **[ "lcd1:VCC", "bb1:tp.15", "red", [ "h-19.2", "v-57.5", "h336", "v182.4", "h-192" ] ],**
36. **[ "bb1:tp.25", "vcc1:VCC", "green", [ "v0" ] ],**
37. **[ "bb1:tn.25", "gnd1:GND", "green", [ "v0" ] ]**
38. **],**
39. **"dependencies": {}**
40. **}**

**IOT DEVICE :**

****

**THANK YOU**

**THESE CODE AND THEORY ARE INCLUDED IN PHASE 3: AIR QUALITY MONITORING**

**BY:**

**MADHANAKUMAR .T**

**422621104024**

**UNIVERSITY COLLEGE OF   
 ENGINEERING,PANRUTI.**