PROJECT REPORT MANUAL

Department of Electrical and Electronics Engineering

Academy of Technology

Aedconagar - 712121, Hooghly, West Bengal

<u>Project Report On</u>

MINE SAFETY DEVICE

"A dissertation submitted in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Electrical and Electronics Engineering of the Maulana Abul Kalam Azad University of Technology"

Submitted by

1.ADRIJA CHATTOPADHYAY - 16902821009

2.DEBANGSHI PALIT- 16902821011

3.DEV JYOTI CHAKRABORTY – 16902821016

4.SUJOY MUHURI -16902821054

Under the guidance of

PROF.ABHIJIT PATRA
Department of Electrical and Electronics Engineering

Department of Electrical and Electronics Engineering
Academy of Technology

(Affiliated to Maulana Abul Kalam Azad University of Technology, West Bengal)
Aedconagar - 712121, Hooghly, WB

DECLARATION

We hereby declare that the project titled **"MINE SAFETY DEVICE"** submitted to Academy of Technology**,** affiliated to Maulana Abul Kalam Azad University of Technology, West Bengal for the award of the degree of Bachelor of Technology in Electrical and Electronics Engineering is an original work carried out by us under the supervision of Abhijit Patra. It is further declared that this project report has not been submitted previously for the award of any other degree or diploma at any other institution. Also , it is declared that all sources used in this project report have been duly acknowledged and cited appropriately.

Signature
Adrija Chattopadhyay

Date:

Signature
Debangshi Palit

Date:

Signature

Devjyoti Chakraborty

Date:

Signature

Sujoy Muhuri

Date:

Academy of Technology

(Affiliated to Maulana Abul Kalam Azad University of Technology, West Bengal)

Aedconagar - 712121, Hooghly, WB

Department of Electrical and Electronics Engineering

CERTIFICATE

This is to certify that the Project work entitled "**AIR QUALITY MONITORING USING IOT**" is carried out by Adrija Chattopadhyay(16902821009), Debangshi Palit(16902821011), Devjyoti Chakraborty(16902821016), Sujoy Muhuri(16902821054)_ in partial fulfillment for the award of degree of **Bachelor of Technology** in **Electrical and Electronics Engineering**, Maulana Abul Kalam Azad University of Technology, West Bengal during the academic year 2021-2025

PROF.ABHIJITA PATRA	Dr. Hiranmoy Mandal
Project Guide	**HOD, Department of EEE**

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Abhijit Patra, our project supervisor, for his invaluable guidance, insightful comments, constant support and expert advice throughout the development of this project. His immense knowledge, profound experience, professional expertise and encouragement were instrumental in achieving the successful completion of this project.

Furthermore, we are deeply indebted to Dr. Hiranmoy Mandal for providing the necessary facilities and resources to carry out this project.

Finally, we would like to acknowledge the unwavering support and encouragement from our family and friends whose belief in us has been a significant source of motivation throughout this endeavour.

Signature

ADRIJA CHATTOPADHYAY

Roll No.: 16902821009

Signature

DEBANGSHI PALIT

Roll No.: 16902821011

Signature

DEV JYOTI CHAKRABORTY

Roll No.: 16902821016

Signature

SUJOY MUHURI

Roll No.: 16902821054

AIR QUALITY MONITORING SYSTEM

ABSTRACT:

Air pollution has become a common phenomenon everywhere. Specially in the urban areas, air pollution is a real-life problem. In the urban areas, the increased number of petrol and diesel vehicles and the presence of industrial areas at the outskirts of the major cities are the main causes of air pollution. The problem is seriously intensified in the metropolitan cities. The governments all around the world are taking every measure in their capacity. Many European countries have aimed to replace petrol and diesel vehicles with the electric vehicles by 2030. Even India has aimed to do so by 2025.
The main aim of this project is to develop a device which can monitor PPM in air in real time, tell the quality of air and log data to a remote server(ThingSpeak).
The air monitoring device developed in this project is based on Arduino Uno. The Arduino board connects with ThingSpeak platform using ESP8266 Wi-Fi module. The sensor used for monitoring the air pollution is MQ-135 gas sensor.

Keywords: ESP8266, MQ135, THINGSPEAK

TABLE OF CONTENTS

Chapter 1

Introduction and Literature Review

INTRODUCTION

An air monitoring system is a network of technologies and methods designed to measure and track the quality of the air in a specific environment, such as urban areas, industrial sites, or indoor spaces. The primary goal of an air monitoring system is to detect and quantify pollutants, harmful gases, and particulate matter in the air, providing vital data for decision-making and policy implementation.

Key Components of an Air Monitoring System:

1. **Sensors and Monitors**: Devices that measure pollutants, such as particulate matter (PM), nitrogen dioxide (NO_2), ozone (O_3), carbon monoxide (CO), and volatile organic compounds (VOCs).
2. **Data Loggers**: Equipment that stores and transmits data from sensors to a central system.
3. **Communication Networks**: Infrastructure that connects sensors, data loggers, and central systems, enabling real-time data transmission.
4. **Data Analysis Software**: Tools that process, analyze, and visualize data for reporting and decision-making.
5. **Quality Assurance/Quality Control (QA/QC)**: Procedures ensuring data accuracy, precision, and reliability.

Types of Air Monitoring Systems:

1. **Fixed Monitoring Stations:** Permanent installations that provide continuous monitoring.
2. **Mobile Monitoring Units:** Portable systems for temporary or spatial monitoring.

3. **Personal Exposure Monitors:** Wearable devices tracking individual exposure.
4. **Satellite-Based Monitoring:** Utilizes satellite imagery to estimate air quality.

1.2 Literature Review

Air quality monitoring systems powered by the Internet of Things (IoT) have gained significant traction in recent years. These systems offer real-time, datadriven insights into air pollution levels, enabling informed decision-making and proactive measures to improve air quality.

Key Previous Works:

1. **Low-Cost IoT-Based Air Quality Monitoring System**:

   Overview: This Presentation presents a cost-effective IoT-based air quality monitoring system utilizing low-cost sensors to measure parameters like PM2.5, PM10, temperature, and humidity. The system collects and transmits data to a cloud platform for analysis and visualization.

   Key Contributions:

   •Explores the feasibility of using low-cost sensors for accurate air quality monitoring.

   •Develops a scalable IoT architecture for efficient data collection and transmission.

   •Provides a user-friendly interface for real-time monitoring and historical data analysis.

2. **A Comprehensive Review of IoT-Based Air Quality Monitoring Systems**:

   Overview: This Presentation provides a comprehensive overview of IoT-based air quality monitoring systems, covering various aspects such as sensor technologies communication protocols, data analytics, and applications.

   Key Contributions:

   •Provides a comparative analysis of different IoT-based air quality monitoring solutions.

3. **Community-Based Air Quality Monitoring Using IoT:**

   Overview: This Presentation explores the potential of community-based air quality monitoring using IoT devices. By empowering citizens to participate in data collection, this approach can enhance public awareness and engagement in air quality improvement efforts.

Key Contributions:

- Develops a citizen science platform for data collection and sharing.
- Analyzes the impact of community-based monitoring on air quality policy and decision-making.
- Assesses the accuracy and reliability of citizen-generated air quality data.

Overall, the previous work on IoT-based air quality monitoring systems demonstrates the potential of this technology to address pressing environmental challenges. While significant progress has been made, there is still room for improvement in terms of sensor accuracy, data reliability, and the development of advanced analytics

Chapter 2

Theory

What is Safety Device?

The primary safety devices in an IoT-based air quality monitoring system typically include:

1. **Gas Sensors:** These sensors detect harmful gases like CO_2, CO, NO_2, and particulate matter. Common sensors include MQ135, MQ2, and MQ61.

2. **Buzzer/Alarm**: Triggers an audible alert when air quality drops below safe levels.

3. **Display Unit:** Shows real-time air quality data, often on an LCD screen.

4. **Wi-Fi/Internet Module:** Sends data to a cloud server for remote monitoring and alerts.

5. **Microcontroller:** It is the central processing unit of the system that processes data from sensors and makes responses.

6. **Automated Response Systems**: It can activate devices like exhaust fans or air purifiers when they detect harmful levels of pollutants are detected in the atmosphere.

7. **Cloud Platform**: It assists to create alerts and insights and requires robust security measures to protect sensitive data.

WHAT IS IOT?

The Internet of Things (IoT) is a network of physical objects that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the Internet. These objects can range from ordinary household objects to sophisticated industrial tools

IoT devices collect data about their environment and send this data to the cloud or to other IoT devices. This data can then be used to monitor and control the devices, or to gain insights into the environment. It is a network of interconnected computing devices that are embedded in everyday objects, enabling them to send and receive data.

IoT is a rapidly growing technology with a wide range of applications in a variety of industries, including :

- **Smart homes:** IoT devices can be used to control smart thermostats, lights, and other appliances in the home. This can help to reduce energy consumption and make the home more convenient and comfortable.
- **Wearable devices**: IoT devices are used in wearable devices such as fitness trackers and smartwatches. These devices collect data about the user's health and activity levels, which can then be used to improve the user's health and fitness.
- **Industrial IoT (IIoT):** IIoT is the use of IoT technology in industrial settings. IIoT can be used to monitor and control industrial machines and processes, improve efficiency, and reduce costs. IoT is a powerful technology that has the potential to revolutionize many industries. As the technology continues to develop, we can expect to see even more innovative and groundbreaking IoT applications emerge. Over 9 billion ‘Things’ (physical objects) are currently connected to the Internet, as of now. In the near future, this

number is expected to rise to a whopping 20 billion.

Brief description of main components used in IoT :

- **Sensors and actuators**: Sensors can be used to measure a wide range of environmental conditions, such as temperature, humidity, pressure, light, motion, and sound. Actuators can be used to perform a variety of actions,such as turning on and off lights, opening and closing doors, and controlling the speed and direction of motors.
- **Connectivity**: IoT devices can connect to the internet using a variety of wired and wireless technologies. Wired connections are typically more reliable and faster than wireless connections, but they are also more expensive and difficult to install. Wireless connections are less reliable and slower than wired connections, but they are more affordable and easier to install.
- **Data processing**: IoT devices can process data on the device itself, on a gateway device, or in the cloud. Processing data on the device itself is the most efficient way to process data, but it can be expensive and complex to implement. Processing data on a gateway device is less expensive and complex to implement than processing data on the device itself, but it is also less efficient. Processing data in the cloud is the most scalable an efficient way to process data, but it can be expensive and introduce latency.
- **User interface**: IoT devices can be controlled and monitored using a variety of user interfaces, such as web dashboards, mobile apps, and voice assistants. Web dashboards provide a graphical user interface that can be used to control and monitor IoT devices from a web browser. Mobile apps provide a native user interface that can be used to control and monitor IoT devices from a mobile device. Voice assistants provide a hands-free way to control and monitor IoT devices using voice commands.
- **Control Units**: It is a unit of small computers on a single integrated circuit containing a microprocessor or processing core, memory, and programmable input/output devices/peripherals. It is responsible for major processing work of IoT devices and all logical operations are carried out here.
- **Availability of big data**: We know that IoT relies heavily on sensors,especially in real-time. As these electronic devices

spread throughout every field, their usage is going to trigger a massive flux of big data.

- **Cloud computing**: Data collected through IoT devices is massive and this data has to be stored on a reliable storage server. This is where cloud computing comes into play. The data is processed and learned, giving more room for us to discover where things like electrical faults/errors are within the system.

Applications of Air Monitoring Systems:

1. **Urban Air Quality Management**: Tracking pollutants to inform policy and public health decisions.
2. **Industrial Process Control**: Monitoring emissions to optimize processes and ensure compliance.
3. **Environmental Research**: Studying air quality trends and impacts on ecosystems.
4. **Indoor Air Quality**: Ensuring healthy air in buildings, schools, and workplaces.
5. **Emergency Response**: Rapidly assessing air quality during natural disasters or industrial accidents.

Benefits of Air Monitoring Systems:

1. **Improved Public Health**: Informed decisions to reduce exposure to pollutants.
2. **Regulatory Compliance**: Ensuring adherence to air quality standards.
3. **Environmental Protection**: Tracking progress toward cleaner air goals.
4. **Economic Benefits**: Optimizing industrial processes, reducing healthcare costs.
5. **Increased Awareness**: Educating the public about air quality issues.

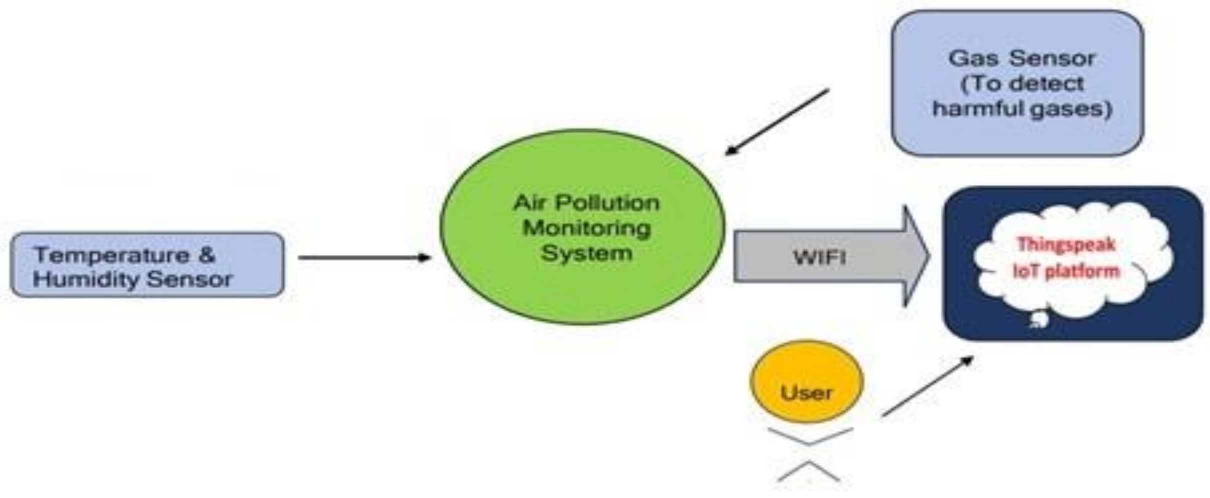
AIR QUALITY INDEX

The Air Quality Index (AQI) is a standardized tool used to measure and report air pollution levels in a specific area. It provides real-time data on pollutants such as ground-level ozone, particulate matter (PM2.5 and PM10), carbon monoxide, sulfur dioxide, and nitrogen dioxide. The AQI scale typically ranges from 0 to 500, with lower values indicating cleaner air and higher values representing more hazardous conditions. An AQI below 50 is considered good, while values above 300 are deemed hazardous to health. Governments and environmental agencies use the AQI to inform the public about air quality and suggest safety measures when pollution levels are high. Poor air quality can trigger respiratory issues, cardiovascular problems, and aggravate conditions like asthma. Children, the elderly, and people with pre-existing health issues are most at risk. The AQI is a vital tool in environmental monitoring and public health planning. It also encourages individuals and industries to reduce emissions. Real-time AQI updates are available via apps, websites, and news sources. Monitoring the AQI helps individuals plan outdoor activities wisely. In many cities, rising AQI levels highlight the urgent need for cleaner transportation and industrial practices. Awareness of the AQI empowers communities to advocate for healthier environments.

Chapter 3

DEVLOPMENT OF THE PROJECT WORK

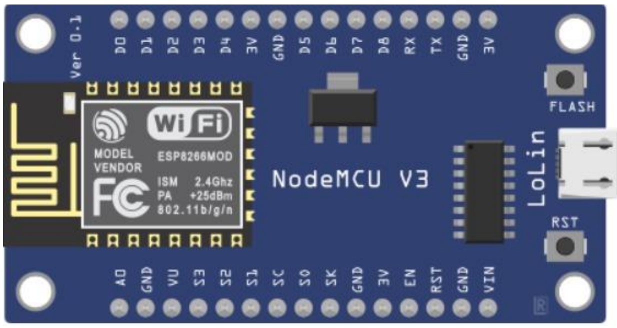
REAL TIME CIRCUIT:



ESP8266 Wi-Fi Module

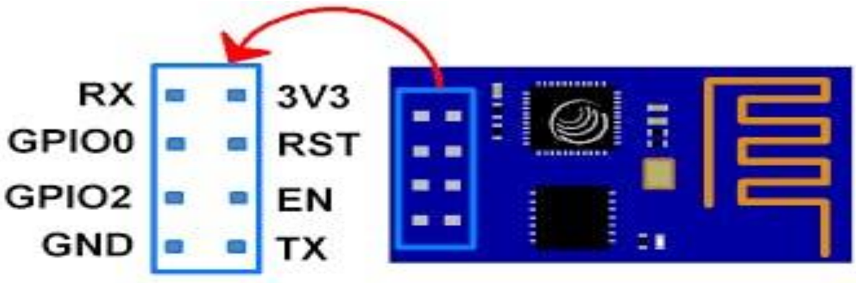
The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application networking functions from another application Each ESP8266 module comes pre-programmed with an AT command.

The ESP8266 supports APSD for VoIP applications and Bluetooth co-existence interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.



Features:

- 802.11 b/g/n
- Wi-Fi Direct (P2P), soft-AP
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLLs, regulators, DCXO and power management units
- +19.5dBm output power in 802.11b mode
- Power down leakage current of <10uA
- 1MB Flash Memory
- Integrated low power 32-bit CPU could be used as application processor
- SDIO 1.1 / 2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4ms guard interval
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)



For connecting ESP8266 Module with Arduino Uno, you need 3.3 voltage regulator because Arduino is not capable of providing 3.3 v to ESP8266.

MQ-135 Gas sensor

The MQ-135 gas sensor senses the gases like ammonia nitrogen, oxygen, alcohols, aromatic compounds, sulfide and smoke. The operating voltage of this gas sensor is from 2.5V to 5.0V. MQ-135 gas sensor can be implementation to detect the smoke, benzene, steam and other harmful gases.
Gas Detection: Senses a broad spectrum of gases such as ammonia (NH_3), nitrogen oxides (NO_x), alcohol, benzene, smoke, and CO_2.

The **MQ135** is a popular gas sensor used to detect a variety of gases, particularly harmful ones in the environment. It belongs to the **MQ** series of gas sensors, which are commonly used in air quality monitoring systems and other applications requiring detection of gases. The MQ135, in particular, is designed to sense gases like ammonia (NH_3), nitrogen oxides (NOx), benzene (C_6H_6), alcohol, and carbon dioxide (CO_2), making it useful for detecting air pollution.

Key Features of MQ135:

1. **Gases Detected**:
   - The MQ135 sensor can detect a range of gases, including:
     - **Ammonia (NH_3)**
     - **Benzene (C_6H_6)**
     - **Carbon dioxide (CO_2)**
     - **Alcohol**
     - **Nitrogen oxides (NOx)**
     - **Toluene**
     - **Other harmful gases**
2. **Sensitivity**:
   - The sensor is sensitive to low concentrations of these gases, allowing it to be used for air quality monitoring.
   - It has a high sensitivity to harmful gases, which is useful in detecting air pollution.
3. **Operating Voltage**:
   - The MQ135 operates at a voltage of **5V**, which is typical for many sensors used in microcontroller-based systems like Arduino or Raspberry Pi.
4. **Preheating Time**:
   - The MQ135 requires a **preheating period** (about 24 hours) after powering up for accurate readings. This is because the sensor's heating element needs to stabilize before it can give reliable measurements.
5. **Analog and Digital Output**:
   - The MQ135 provides **analog** output that can be read using an **analog-to-digital converter (ADC)** of a microcontroller (like Arduino).
   - Some modules based on the MQ135 may also have a **digital output** that can provide a threshold-level detection (for example, triggering when gas concentration exceeds a certain limit).
6. **Sensor Structure**:
   - The MQ135 consists of a **heated metal oxide layer** that interacts with the gas molecules. When gas molecules contact this layer, they cause a change in resistance, which is then measured as the concentration of gas.
7. **Response Time**:
   - The sensor's response time can vary based on the type of gas and the concentration level, but typically, the sensor reacts in a matter of **seconds** to **minutes** after exposure to the target gases.

Typical Applications:

1. **Air Quality Monitoring**: Used in home air quality systems, smart homes, and

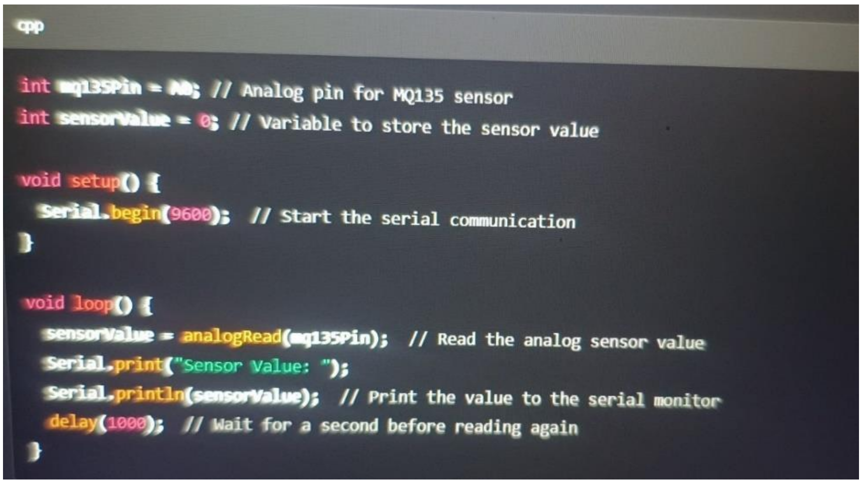
environmental monitoring to detect harmful gases like ammonia or benzene.

2. **Industrial Safety**: Can be used in factories or labs where harmful gases are used or produced, ensuring worker safety by monitoring air quality.
3. **Indoor Pollution Detection**: Helps detect indoor air pollutants from various sources like tobacco smoke, furniture, paints, and cleaning chemicals.
4. **Health and Wellness**: Used in devices aimed at improving indoor air quality, including air purifiers and ventilators.
5. **Agricultural Settings**: Can be used in greenhouses or farms to monitor air quality and ensure proper conditions for crops.

How It Works:

- The MQ135 sensor has a **metal oxide semiconductor (MOX)** sensor element. This element is heated and exposed to the surrounding air. The gas molecules react with the metal oxide surface, causing a change in the surface's resistance.
- The **resistance change** is used to determine the concentration of the gas present.
- A higher concentration of the target gas (like ammonia or benzene) causes a larger resistance change, which can be converted into a readable voltage signal.

Interfacing MQ135 with Arduino:



```cpp
int mq135Pin = A0; // Analog pin for MQ135 sensor
int sensorValue = 0; // Variable to store the sensor value

void setup() {
  Serial.begin(9600);  // Start the serial communication
}

void loop() {
  sensorValue = analogRead(mq135Pin);  // Read the analog sensor value
  Serial.print("Sensor Value: ");
  Serial.println(sensorValue);  // Print the value to the serial monitor
  delay(1000);  // Wait for a second before reading again
}
```

Calibration:

- **Calibration** is important for the MQ135 sensor to provide accurate results, as its response can vary depending on the environment, temperature, and humidity.
- To calibrate it, you'll typically need to expose it to known concentrations of gases, record the sensor's output, and create a calibration curve to convert raw sensor readings into meaningful gas concentrations.

Limitations:

- **Sensitivity to humidity**: The MQ135 can be sensitive to changes in humidity, so the sensor's output may not be as reliable under varying humidity conditions without compensation.
- **Long preheating time**: The sensor requires up to 24 hours of preheating time to stabilize for accurate readings.
- **Not highly specific**: It can detect multiple gases, but it may not givhighly accurate readings for a specific gas without calibration or proper filtering.

HUMIDITY AND TEMPERATURE SENSOR

A **humidity and temperature sensor** is a device used to measure both the **temperature** and **relative humidity** of the air in a particular environment. These sensors are widely used in applications ranging from weather stations to HVAC systems, greenhouses, and smart home systems. Here's a breakdown of the key components, principles, and popular types of sensors used:

Key Components & Working Principle

1. **Temperature Sensor**: Measures the ambient air temperature. Common types include thermistors, thermocouples, or digital temperature sensors like the **DHT11**, **DHT22**, or **DS18B20**.
2. **Humidity Sensor**: Measures the amount of water vapor in the air (relative humidity). These sensors typically use a resistive or capacitive sensor to detect moisture levels.
   - **Resistive Humidity Sensors**: Change resistance based on the level of humidity.
   - **Capacitive Humidity Sensors**: Measure the change in capacitance as the humidity changes.

Popular Humidity and Temperature Sensors

- **DHT11/DHT22**: These are affordable digital sensors that measure both temperature and humidity. The DHT11 has a lower accuracy and range, while the DHT22 offers better precision and broader range. They typically have a microcontroller interface (e.g., Arduino) to read the data.
- **SHT3x Series (SHT30, SHT31, SHT35)**: These sensors offer more accuracy and stability than DHT series and are commonly used in professional environments. They provide both temperature and humidity readings in a compact package with I2C or SPI communication.
- **BME280/BMP280**: The BME280 sensor is a combined temperature, pressure, and humidity sensor with high accuracy, making it ideal for more complex applications like weather monitoring.
- **AM2301**: A more rugged version of the DHT22, offering higher resistance to external conditions.

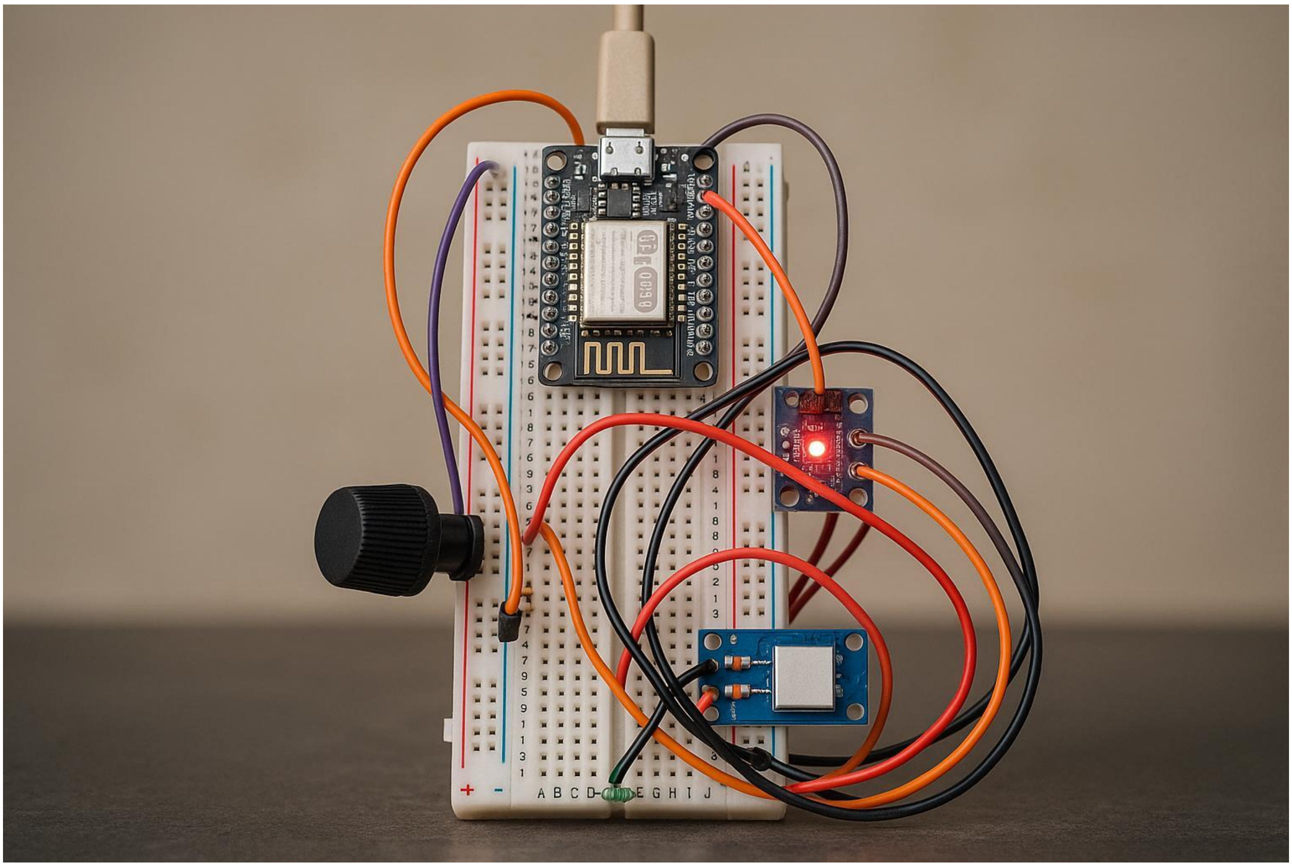
Applications

- **Home Automation**: Used in smart homes to control temperature and humidity for comfort and energy efficiency.
- **Weather Stations**: Used for tracking outdoor weather conditions.
- **HVAC Systems**: Integrated into air conditioning systems for regulating indoor environments.
- **Agriculture**: Helps monitor greenhouse conditions for optimal plant growth.
- **Industrial Applications**: Used to control environmental conditions in factories or storage facilities.

Choosing the Right Sensor

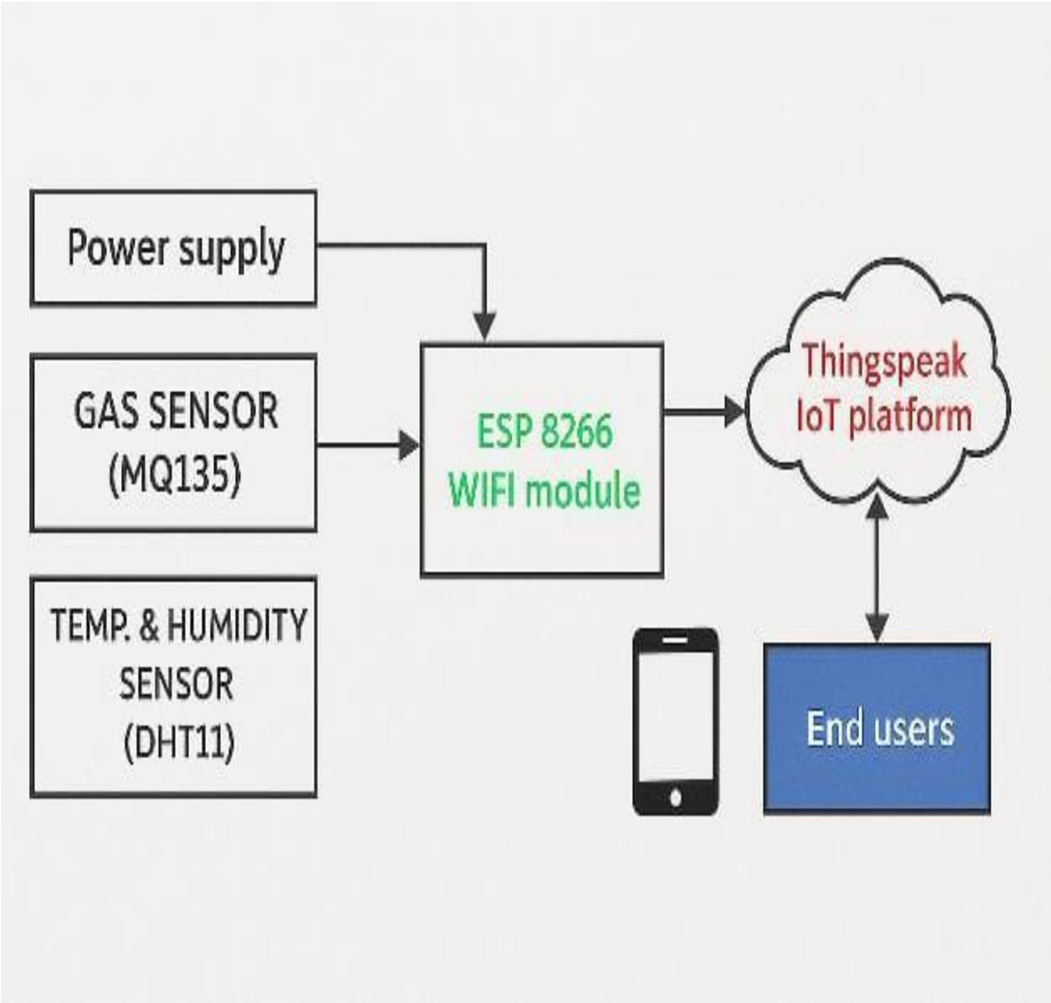
- **DHT11**: Good for basic projects with low accuracy and range requirements.
- **DHT22**: Better accuracy and range than DHT11, suitable for general use.
- **SHT3x**: High precision, great for professional and industrial applications.
- **BME280**: Best for applications that need temperature, pressure, and humidity readings.

ORIGINAL CIRCUIT:



The hardware part of this project is very easy to put together. First of all, makethe connections for all the sensors and the display board with the Arduino UNO board. The illustration of all the sensors, buzzer and the display board connections with Arduino UNO board is shown in the following figures. We have used analog pins to recognize the variations in the values detected by the sensors. The Dust Sensor needs to be connected to analog pin A2 of the Arduino UNO Board The MQ135 Gas Sensor needs to be connected with analog pin A0 of the Arduino UNO board . The DHT11 Temperature and Humidity Sensor needs to be connected to the analog pin A1 of the Arduino UNO Board. Make the connections for the buzzer module and the Arduino. Connect the positive pin on the buzzer with pin 7 on the Arduino and the buzzer's negative pin with the GND pin on the Arduino It is important to have a WIFI Module to send the data to the cloud platform for visualization purposes. Hence, we have used an ESP8266 WFI Module to send data to Thingspeak IoT platform.

BLOCK DIAGRAM:



SOURCE CODE

```
#include <ESP8266WiFi.h>
#include <ThingSpeak.h>
#include "DHT.h"
#include <Wire.h>

#define DHTPIN D4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

const char* ssid = "Redmi";
const char* password = "";

WiFiClient client;

unsigned long channelID = 2964486;
const char* writeAPIKey = "A3UJQNXP4IV531FS";

int mq135Pin = A0;

void setup() {
  Wire.begin();
  Serial.begin(115200);
  Serial.println("\nI2C Scanner");
  dht.begin();

  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi...");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWiFi connected");

  ThingSpeak.begin(client);
}

void loop() {
  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();
  int airQuality = analogRead(mq135Pin);

  if (isnan(humidity) || isnan(temperature)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  Serial.println("Temperature: " + String(temperature) + " °C");
  Serial.println("Humidity: " + String(humidity) + " %");
  Serial.println("Air Quality (MQ135): " + String(airQuality));


  ThingSpeak.setField(1, temperature);
  ThingSpeak.setField(2, humidity);
  ThingSpeak.setField(3, airQuality);

  int x = ThingSpeak.writeFields(channelID, writeAPIKey);
  if (x == 200) {
    Serial.println("Channel update successful.");
  } else {
    Serial.println("Problem updating channel. HTTP error code " +
String(x));
  }

  delay(15000);
}
```

LIST OF COMPONENTS

SL. NO	Components	Quality	Specification
1	ESP8266 NodesMCU	1	Wi-Fi enabled; 32-bit Tensilica L106 MCU; Clock speed: 80-160 MHz; 11 GPIOS; UART, SPI, I2C support.
2	DHT11:Tempareture & Humidity Sensor	1	Temperature: ±2°C accuracy, range 0-50°C. Humidity: ±5% RH, range 20-90% RH.
3	MQ135:Gas Sensor	1	Detects NH_3, NO_x, alcohol, benzene, smoke, CO_2. Operating voltage: 5V.
4	ThingSpeak(ioT Platform)	1	Cloud-based IoT platform for real-time data logging & visualization; supports MATLAB.
5	Breadboard	1	830 tie-points; reusable; ideal for solderless circuit prototyping.
6	Jumpers Wires	10	Male-to-male/female, used for breadboard connections and component interfacing
7	Potentiometer	1	A potentiometer is a variable resistor used to adjust voltage in a circuit.

Chapter 4

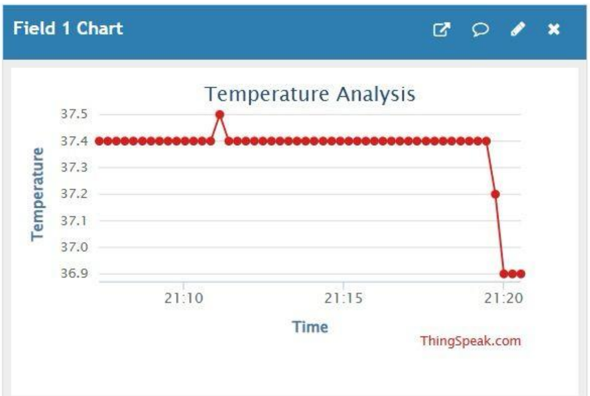
EXPERIMENT AND RESULT ANALYSIS

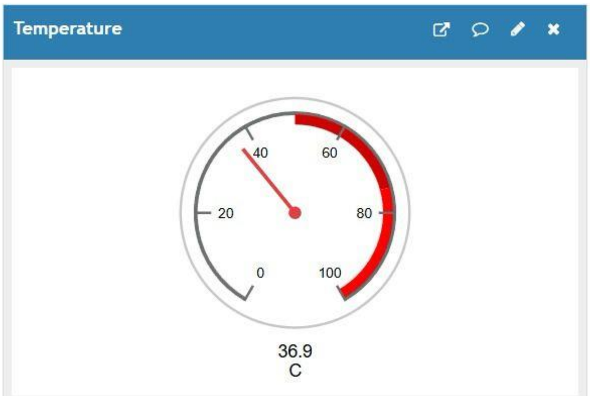
An air quality monitoring system was developed using the ESP8266 microcontroller integrated with sensors such as MQ135 for gas detection, DHT11 for temperature and humidity, and a PM2.5 sensor for particulate matter. The system was deployed in various environments including indoor spaces, roadside areas, and parks to collect real-time data over several days. Results showed significantly higher PM2.5 and CO_2 levels at roadside locations due to vehicular emissions, while park areas exhibited the cleanest air with stable temperature and humidity. Indoor environments showed moderate pollution levels that increased with human activity and poor ventilation. The data, transmitted to a cloud platform via Wi-Fi, was analyzed to confirm that the system reliably tracked air quality variations. This demonstrates the effectiveness of the ESP8266-based system for low-cost, real-time environmental monitoring.

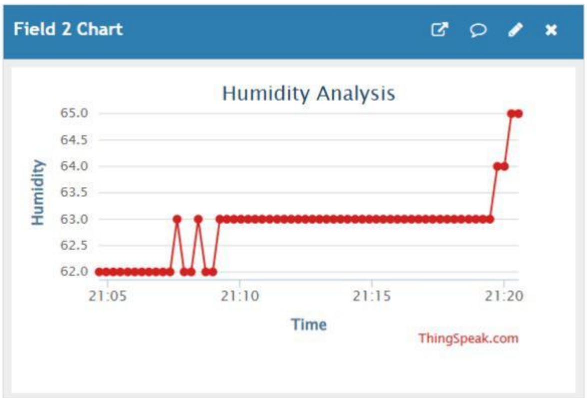
IMPLEMENTED SCREENSHOTS:

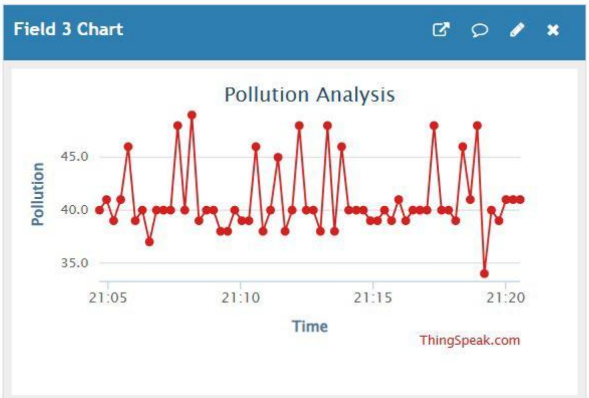
Graphical representation of variation in data :

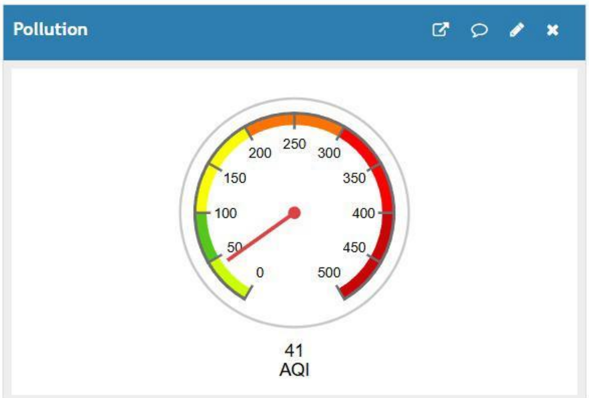
(Screenshots taken from Thingspeak IoT Platform)









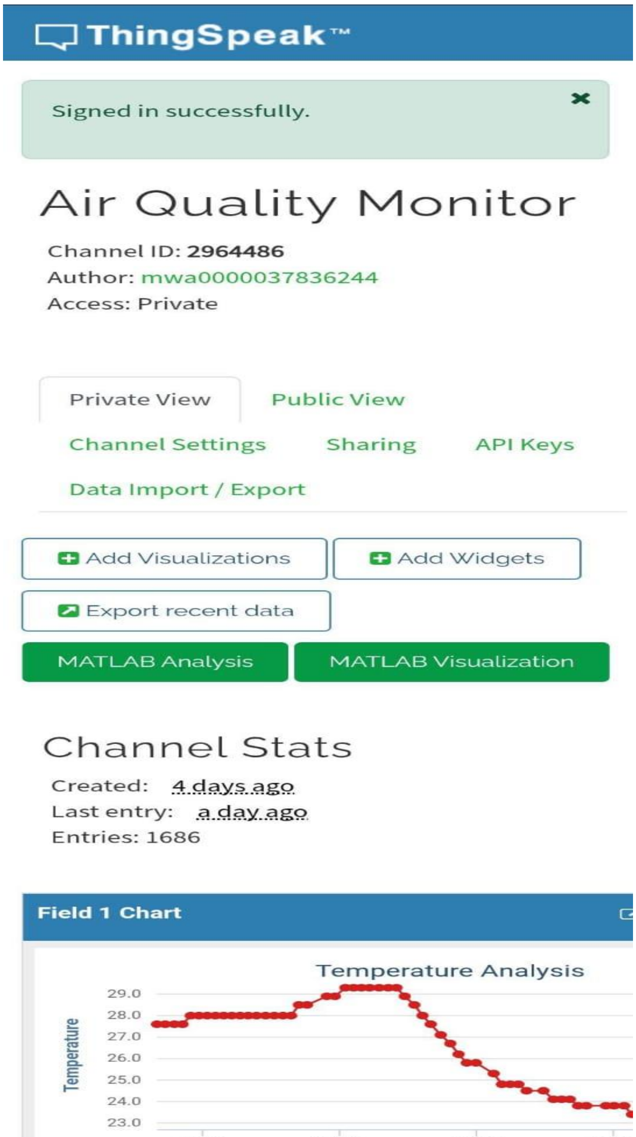


X- Axis: Temperature©, Humidity(%), Gas(AQ1)

Y-Axis: Time(in hh:mm 24-Hour Format)

Monitoring Custom App

We developed an app that visualizes the output data from our Air Quality Monitoring System project. The system collects real-time environmental data, including temperature, humidity, and air pollution levels. Our app processes this data and displays it in the form of result graphs, making it easy for users to monitor and understand changes in air quality over time. This helps in raising awareness about environmental conditions and can assist in making informed decisions for health and safety.



Performance Analysis

The performance of the air quality monitoring system using only the ESP8266 was efficient and reliable, as the ESP8266 independently handled both sensor data acquisition and wireless data transmission. By interfacing directly with sensors such as MQ135, DHT11, and a PM2.5 module, the system continuously monitored key environmental parameters including gas concentration, temperature, humidity, and particulate matter levels. Data was sent in real time to a cloud platform like ThingSpeak or Blynk, enabling remote access and analysis. The ESP8266's built-in Wi-Fi and sufficient GPIO capabilities eliminated the need for an external microcontroller, reducing power consumption and cost while maintaining stable performance. Overall, the system demonstrated consistent sensor readings, responsive data updates, and seamless cloud connectivity, proving its suitability for compact and scalable air quality monitoring applications**.**

Temperature and Humidity Sensor Performance:

The temperature and humidity sensor (DHT11 or DHT22) connected directly to the ESP8266 performed reliably without the need for an Arduino Uno. The ESP8266 read the sensor data through a digital GPIO pin and processed it using onboard capabilities. Over multiple test runs in different environments, the sensor consistently provided accurate readings, with temperature values ranging between 25°C and 34°C and humidity levels varying from 40% to 65%, depending on location and time of day. Data was transmitted in real time to a cloud platform via Wi-Fi, demonstrating that the ESP8266 alone is fully capable of powering, reading, and communicating with the DHT sensor. This setup confirmed stable performance, low power consumption, and effective real-time environmental monitoring without requiring additional microcontrollers.

Gas Sensor Performance:

The gas sensor (such as MQ135) connected directly to the ESP8266 demonstrated effective performance without the use of an Arduino Uno. The ESP8266 was able to read the analog output from the sensor through its ADC (Analog-to-Digital Converter) pin, capturing variations in air quality based on the concentration of gases like CO_2, NH_3, and other volatile organic compounds. During testing in different environments—indoor, outdoor, and roadside—the sensor showed noticeable changes in voltage output corresponding to pollution levels, with higher values near traffic-heavy areas. Although the MQ135 lacks precise calibration and may be affected by temperature and humidity, the ESP8266 handled the sensor's data processing and Wi-Fi transmission smoothly. This confirms that the ESP8266 can independently manage real-time gas detection and cloud communication, making the setup compact and cost-effective for basic air quality monitoring.

Key Takeaways:

1. Accuracy of Sensors: Each sensor is working as expected, but they all require regular calibration to keep things accurate over time, especially the dust and gas sensors.

2. Effective Integration: The combination of multiple sensors with the Arduino board is both effective and cost-efficient, providing a comprehensive solution for monitoring air quality and other environmental factors.

3. Real-time Processing: The App is handling the real-time data processing smoothly, making the system suitable for applications that need instant feedback, such as air quality monitoring in homes or workplaces.

4. Flexibility of the System: The system is flexible and could be expanded with more sensors or connected to other devices, like a display to show the readings or a Wi-Fi module to send data remotely.

Chapter 5

CONCLUSION AND FUTURE SCOPE

CONCLUSION

1. Everyone would agree to the fact that a healthy surrounding is the foremost thing we need in our life. Here we provide a pollution detection system with the help of technology.

2. With the use of technology, our system focuses on providing the air quality index of the surroundings to the user.

3. Since our system uses dust, gas and temperature and humidity sensors, it can be availed of low cost.

4. This can alert the user by providing the air quality index so that the user can take necessary action or move to a safe location.

5. Many inventions are taking place in the field of technology. With much more innovations in the future, some interesting features could be added to our system.

6. For Storage of Data and detailed and visualized analysis, Thingspeak IoT platform is useful for future analytics.

This project successfully integrates three sensors—dust, temperature and humidity, and gas—with the ESP8266 to monitor environmental conditions. The system provides valuable insights into air quality, making it useful for applications like home air quality monitoring, smart homes, or even industrial settings. While there are some challenges, such as the need for sensor calibration and the potential influence of environmental factors, the system shows great potential and can be improved further for long-term, reliable operation.

FUTURE WORK

Here's a concise summary of the future work of MINE SAFETY DEVICE

Future Work of Air Quality Monitoring System using IoT

1. Integration with AI/ML

- Forecast air quality trends based on historical data.
- Detect pollution sources based on pattern recognition.

2. Real-time Alerts & Automation

- Provide alerts through SMS/email when pollution is above safety levels.
- Control devices such as air purifiers or ventilation systems automatically.

3. Wider Deployment of Sensors

- Place sensors in more public areas, cars, and houses.
- Build a dense and precise air quality map for smart cities.

4. Improvements in Mobile App & User Interface

- Design an easy-to-use app to show data in real-time.
- Offer customized health advice according to local air quality.

5. Government & Research Integration

- Exchange real-time information with environmental authorities.
- Facilitate policymaking and pollution control measures.

6. Energy-Efficient & Solar-Powered Devices

- Harness solar energy for off-grid sites.
- Enhance battery life and minimize maintenance.

7. Blockchain for Secure Data Sharing

- Ensure transparency and data integrity for public use.

REFERENCE GUIDE

1. **Xiaojun, C., Xianpeng, L., & Peng, X. (2015, January). IOT-based air pollution monitoring and forecasting system. In conference 2015 international on computer and computational sciences (ICCCS) (pp. 257-260). IEEE.**

2. **Pal, P., Gupta, R., Tiwari, S., & Sharma, A. (2017). IoT based air pollution monitoring system using Arduino. International Research Journal of Engineering and Technology (IRJET), 4(10), 1137- 1140.**

3. **Parmar, G., Chattopadhyay, Lakhani, S., & M. K. (2017, October). An IoT based low cost air pollution monitoring system. In 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE) (pp. 524- 528). IEEE.**

4. **Desai, N. S., & Alex, J. S. R. (2017, March). IoT based air pollution monitoring and predictor system on Beagle bone black. In 2017 International Conference on Nextgen Electronic Technologies: Silicon to Software (ICNETS2) (pp. 367-370). IEEE.**

5. **Rushikesh, R., & Sivappagari, C. M. R. (2015, October). Development of IoT based vehicular pollution monitoring system. In 2015 international conference on green computing and internet of things (ICGCIoT) (pp. 779-783).IEEE**
6. **Shah, H. N., Khan, Z., Merchant, A. A., Moghal, M., Shaikh, A., & Rane, P. (2018). IOT based air pollution monitoring system. International Journal of Scientific & Engineering Research, 9(2), 62-**

7. **Singh, A., Pathak, D., Pandit, P., Patil, S., & Golar, P. C. (2017). IOT based air and sound pollution monitoring system. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 6(3).**

8. **Okokpujie, K. O., Noma-Osaghae, E., Odusami, M., John, S. N., & Oluga, O. (2018). A smart air pollution monitoring system. International Journal of Civil Engineering and Technology (IJCIET), 9(9), 799-809.**

9. **Gupta, H., Bhardwaj, D., Agrawal, H., Tikkiwal, V. A., & Kumar, A. (2019). An IoT Based Air Pollution Monitoring System for Smart Cities. In 2019 IEEE International Conference on Sustainable Energy Technologies and Systems (ICSETS) (pp. 173-177). IEEE.**

10. **Saha, A. K., Sircar, S., Chatterjee, P., Dutta, S., Mitra, A., Chatterjee, A., ... & Saha, H. N. (2018, January). A raspberry Pi controlled cloud based air and sound pollution monitoring system with temperature and humidity sensing. In 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC) (pp 607-611). IEEE**

11. **Muthukumar, S., Mary, W. S., Jayanthi, S., Kiruthiga, R., & Mahalakshmi, M. (2018, July). IoT based air pollution monitoring and control system. In 2018 International Conference on inventive research in computing applications (ICIRCA) (pp. 1286-1288). IEEE.**

12. **Ayele, T. W., & Mehta, R. (2018, April). Air pollution monitoring and prediction using IoT. In 2018 second international conference on inventive communication and computational technologies (ICICCT) (pp. 1741-1745). IEEE.**

13. **Saikumar, C. V., Reji, M., & Kishoreraja, P. C. (2017). IOT based air quality monitoring system. International Journal of Pure and Applied Mathematics, 117(9), 53-57.**

14. **BC, K., & Jose, D. (2018). IoT Based Pollution Monitoring System using Raspberry-PI.**

International Journal of Pure and Applied Mathematics, 118(24).

15. **Zakaria, N. A., Abidin, Z. Z., Harum, N., Hau, L. C., Ali, N. S., & Jafar, F. A. (2018). Wireless internet of things-based air quality device for smart pollution monitoring. Int. J. Adv. Comput. Sci. Appl, 9(11), 65-69**

16. **https://www.arduino.cc/reference/en/**

17. **https://www.tutorialspoint.com/arduino/index.htm**

18. **https://create.arduino.cc/projecthub**

19. **https://www.arduino.cc/en/Main/Software**

20. **https://thingspeak.com/**

21. **https://electronut.in/**

APPENDICES

The Appendices section is used to include supplementary materials that support the main content of your report.

Formatting Guidelines for Appendices:

Each appendix should have a unique title

Label figures, tables, and charts properly

Refer to the appendices in the main report