SMART AIR QUALITY MONITORING SYSTEM USING IOT

A PROJECT REPORT

submitted by

DHARANIKUMAR S (230701073)
VENKATESH HEMNATH (230701115)
GOPIKRISHNAN L (230701096)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE,

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2024

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this project report titled "**SMART AIR QUALITY MONITORING SYSTEM"** is the bonafide work of "**DHARANIKUMAR S (230701073), VENKATESH HEMNATH (230701115), GOPIKRISHNAN L (230701096)"** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Ms. S. Ponmani M.E.,MBA,

SUPERVISOR

Assistant Professor

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on ____________________

Internal Examiner **External Examiner**

ABSTRACT

Air pollution has emerged as a critical environmental and public health challenge, contributing to numerous respiratory and cardiovascular diseases, while also adversely impacting ecosystems and climate. To address this growing concern, this project introduces the design and development of a Smart Air Quality Monitoring System capable of continuously tracking real-time environmental parameters, including the Air Quality Index (AQI), temperature, and humidity. The system leverages a combination of cost-effective sensors and microcontrollers, specifically the ESP8266 Wi-Fi module, the MQ135 gas sensor for air quality detection, and the DHT11 sensor for temperature and humidity monitoring. Data collected by the sensors is transmitted wirelessly and stored in Firebase, a cloud-based real-time database, enabling seamless remote access and historical data analysis.

A user-friendly interface is developed using Blynk and web-based dashboards, providing users with intuitive visualizations of environmental trends over time. Additionally, the system enhances user engagement by offering context-aware recommendations, such as outdoor activity advisories and personalized health tips based on current air quality conditions. This integrated approach not only ensures real-time monitoring but also empowers users to make informed lifestyle decisions that prioritize health and environmental consciousness.

The proposed solution highlights the potential of Internet of Things (IoT) technologies in promoting environmental awareness, public safety, and data-driven decision-making. By bridging sensor technology, cloud computing, and user-centric design, the Smart Air Quality Monitoring System serves as a scalable and accessible tool for individuals, communities, and local authorities in the collective effort to combat air pollution.

ACKNOWLEDGEMENT

First, we thank the almighty God for the successful completion of the project. Our sincere thanks to our chairman **Mr. S. Meganathan, B.E., F.I.E.,** for his sincere endeavor in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan, Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project, and Vice-Chairman **Mr. Abhay Shankar Meganthan**, **B.E., M.S.,** for providing us with the requisite infrastructure. We also express our sincere gratitude to our college principal, **Dr.S.N.Murugesan M.E., PhD.,** and **Dr. P. Kumar M.E., Ph.D., Head of the Department of Computer Science and Engineering,** and our project guide **Ms. S. Ponmani M.E.,MBA,** for her encouragement and guiding us throughout the project. We would like to thank our parents, friends, all faculty members, and supporting staff for their direct and indirect involvement in the successful completion of the project for their encouragement and support.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Air pollution has emerged as a critical global challenge, especially in densely populated urban regions. Prolonged exposure to pollutants such as nitrogen dioxide (NO_2), carbon monoxide (CO), ammonia (NH_3), and particulate matter (PM) has been linked to respiratory diseases, cardiovascular conditions, and other serious health issues. As awareness about environmental health grows, so does the demand for accurate, real-time air quality monitoring solutions.Traditional air quality monitoring systems are expensive, stationary, and often limited in number, making it difficult for individuals to access localized environmental data. In response to this gap, this project proposes a Smart Air Quality Monitoring System that leverages the Internet of Things (IoT) to provide continuous, personalized, and cloud-connected air quality insights.This system uses the ESP8266 NodeMCU microcontroller integrated with an MQ135 gas sensor for air quality detection and a DHT11 sensor for temperature and humidity readings. The collected data is sent in real time to a Firebase Realtime Database, where it is stored, analyzed, and visualized through both a mobile application (via Blynk) and a web-based dashboard. The platform not only displays Air Quality Index (AQI) levels but also offers dynamic health tips and activity recommendations based on environmental conditions.

By combining hardware sensing with cloud computing and user-friendly interfaces, this project aims to make environmental awareness more accessible.

1.1 Motivation

-Poor Air Quality and Public Health Impact:

Air pollution has become a major global issue, adversely affecting the health of millions. Prolonged exposure to polluted air is linked to respiratory infections, asthma, heart disease, and other serious conditions. The World Health Organization (WHO) considers air pollution one of the top environmental risks to health.

- Inadequacy of Existing Monitoring Systems:

Government-operated air quality monitoring stations are often expensive to install and maintain, and they are typically limited to major urban centers. As a result, many residential and suburban areas lack access to real-time, localized air quality data. This makes it difficult for individuals to assess the safety of their immediate environment.

- Need for Affordable, Real-Time Monitoring:

There is a growing demand for low-cost, scalable, and user-friendly air quality monitoring systems that can provide real-time data to everyday users. Such systems can enable people to take timely precautions, especially in high-risk groups such as children, the elderly, and those with pre-existing medical conditions.

Role of IoT in Environmental Monitoring:

The Internet of Things (IoT) offers a promising solution by enabling the integration of sensors, cloud platforms, and mobile applications. IoT allows for the continuous monitoring of environmental parameters like air quality, temperature, and humidity, with instant access to data from anywhere.

1.2 Objectives

Monitor air quality using MQ135 sensor:

The MQ135 gas sensor is used to detect harmful gases such as ammonia (NH_3), nitrogen oxides (NOx), benzene, carbon dioxide (CO_2), and smoke. This sensor provides an analog signal representing the concentration of pollutants in the air. By using this sensor, the system can calculate an approximate Air Quality Index (AQI), which reflects the cleanliness or pollution level of the surrounding environment.

- Measure temperature and humidity using DHT11:

The DHT11 sensor is a widely used digital sensor that measures both temperature and relative humidity. Monitoring these environmental parameters helps provide a more comprehensive view of air conditions, as temperature and humidity can influence the concentration and impact of air pollutants.

- Connect to Firebase for real-time data logging:

Firebase is a cloud-based real-time database platform provided by Google. By integrating Firebase with the ESP8266 microcontroller, the system can continuously upload sensor data to the cloud. This allows users to access historical and real-time air quality data from any location through connected applications**.**

- Provide alerts and suggestions via Blynk app and web dashboard:

The Blynk IoT platform enables remote monitoring and control of IoT devices via smartphone applications. In this system, Blynk is used to display real-time readings, send notifications, and offer personalized suggestions such as avoiding outdoor activities when AQI levels are high. A web dashboard complements this by providing broader data visualization and recommendations.

CHAPTER 2

LITERATURE REVIEW

[1] Design and Implementation of Real-Time Air Pollution Monitoring System Using IoT
This paper presents a low-cost IoT-based air quality monitoring system using ESP8266 and gas sensors like MQ135. It describes how real-time data acquisition is achieved using cloud platforms such as ThingSpeak. The system helps in determining pollutant concentration in urban areas and alerts users through mobile notifications, demonstrating the feasibility of portable air quality monitoring systems.

[2] Cloud-based Air Quality Monitoring and Notification System Using Firebase The authors propose an air quality monitoring system that integrates Firebase as the primary backend for storing and retrieving environmental data. Using sensors like MQ135 and DHT11, data is transmitted via Wi-Fi and analyzed in real time. The system also includes a mobile application that alerts users when pollutant levels exceed safe thresholds.

[3] IoT-Based Health Monitoring System for Air Pollution
This paper focuses on health-related impacts of air pollution and proposes a wearable and home-based IoT system for monitoring AQI. It includes personalized health recommendations based on current and historical exposure to pollutants. Machine learning is used to assess risk levels and suggest preventive measures to users, especially those with respiratory issues.

[4] Development of a Smart Environment Monitoring System Using Blynk and NodeMCU
This research outlines a smart environmental monitoring prototype that uses NodeMCU, MQ135, and DHT11 sensors to capture data on air quality, temperature, and humidity. The Blynk IoT platform is used for real-time visualization on mobile devices. The study demonstrates the importance of using cloud and mobile interfaces for continuous and accessible environmental monitoring.

[5] A Survey on Air Quality Index and Real-Time Monitoring Using IoT This review paper analyzes various methodologies for calculating and displaying AQI using IoT-based devices. It discusses sensor accuracy, data normalization techniques, and AQI standards used across countries. The importance of integrating user-friendly dashboards and mobile apps for increasing public awareness and participation in pollution control is emphasized.

2.1 Existing System

Government Monitoring Stations (e.g., CPCB, EPA):

Highly accurate but expensive, fixed in location, and limited in coverage. They do not provide real-time localized data or personalized feedback.

- Commercial Devices (e.g., Atmotube, AirVisual Pro):

Portable and user-friendly but often costly. Some lack cloud integration or health-based alerts.

- Mobile AQI Apps (e.g., AQICN, Air Report):

Rely on third-party or satellite data rather than local sensors. They lack accuracy in specific locations and offer limited personalization.

- IoT Research Prototypes:

Built using platforms like Arduino or ESP8266 but typically remain in development stages. They are not widely available for public use.

2.1.1Advantages of the existing system

Real-time monitoring and alerts

The system continuously tracks vital signs (such as heart rate, temperature, etc.) and immediately notifies users or healthcare providers if any abnormal readings are detected. This rapid response helps in preventing health emergencies and ensures timely medical intervention.

• Remote data access via app and web

Users and doctors can view health data anytime, anywhere, using a mobile app or a web portal. This flexibility is particularly beneficial for remote patients, elderly individuals, or those who need constant health supervision without frequent hospital visits.

• Personalized health tips

Based on the user's health data trends, the system can generate customized suggestions for diet, exercise, or lifestyle changes. These tips aim to improve the user's overall health and prevent potential issues before they arise.

• Easy to deploy and scale

The system can be quickly installed and configured for use in homes, clinics, or hospitals. Additionally, it supports scalability—more devices or users can be added easily without major changes to the existing infrastructure.

2.1.2 Drawbacks of the existing system

High Cost:

Government and commercial monitoring systems are expensive to install and maintain, making them inaccessible for personal or community-level use.

- Lack of Portability:

Most systems are fixed in one location and cannot be easily moved to monitor air quality in different areas.

- Limited Coverage:
Sparse deployment leads to gaps in monitoring, especially in suburban and rural regions.

- No Real-Time Personalized Feedback:
Existing systems typically do not offer real-time alerts or customized health recommendations for individuals

- Dependence on External Data:
Mobile AQI apps rely on data from third-party sources, which may not reflect actual conditions in the user's immediate surroundings.

- Technical Complexity in DIY Systems:
Research or prototype systems often require technical expertise, making them unsuitable for non-technical users.

2.2 Proposed System

To address the limitations of existing air quality monitoring solutions, we propose a smart, portable, and cost-effective system built using IoT technologies. The core of the system is based on the ESP8266 NodeMCU microcontroller, which provides built-in Wi-Fi capability for real-time data transmission. It is integrated with the MQ135 gas sensor for detecting harmful gases such as ammonia, carbon monoxide, and nitrogen dioxide, and the DHT11 sensor for measuring temperature and humidity.

Sensor readings are continuously collected and transmitted to the Firebase Realtime Database, enabling secure cloud storage and remote data access. Users can monitor air quality data in real time via the Blynk mobile application, which displays key parameters like AQI, temperature, and humidity in a user-friendly interface. In addition to live readings, the app and web dashboard provide health-based alerts, activity suggestions (e.g., "Avoid outdoor exercise"), and tips tailored to the current air quality level.

The system is lightweight, low-cost, and easy to deploy in homes, schools, offices, or public places. Unlike government or commercial solutions, it offers real-time, location-specific, and personalized feedback. Moreover, the use of cloud storage and mobile/web interfaces ensures that users can access data and insights anytime, from anywhere.

This proposed system not only empowers individuals with environmental awareness but also promotes proactive health decisions through intelligent, data-driven recommendations.

2.2.1 Advantages of the proposed system

- Low Cost and High Accessibility:

The use of affordable components like ESP8266, MQ135, and DHT11 makes the system cost-effective and accessible to a wider population compared to commercial and government-grade equipment.

- Real-Time Monitoring:

The system provides continuous, real-time data on air quality, temperature, and humidity, allowing users to make timely and informed decisions.

- Portability:

The compact size and wireless connectivity enable the device to be used in various locations such as homes, schools, offices, or while traveling.

- Cloud Integration via Firebase:

Sensor data is automatically stored in the cloud, enabling remote access, long-term storage, and historical trend analysis.

- User-Friendly Interface:

The Blynk mobile app and a custom web dashboard present sensor data in a clear and intuitive manner with graphical charts and color-coded AQI indicators.

- Personalized Alerts and Health Tips:

The system offers dynamic suggestions and safety recommendations based on current air quality, improving public awareness and health outcomes.

- Scalability and Flexibility:

Designed using modular components and open platforms, the system can be easily expanded or customized for different use cases or environments.

CHAPTER 3

SYSTEM DESIGN

3.1Development Environment

3.1.1 Hardware Requirements

- ESP8266 NodeMCU
- MQ135 Gas Sensor
- DHT11 Sensor
- Jumper Wires
- Breadboard
- USB Cable
- Power Bank (optional for portability)

ESP8266 NodeMCU:

A low-cost microcontroller board with built-in Wi-Fi capabilities. It serves as the brain of the system, collecting sensor data and transmitting it to cloud platforms like Firebase. It supports programming via the Arduino IDE and is ideal for IoT applications.

- MQ135 Gas Sensor:

An air quality sensor used to detect a range of harmful gases including ammonia (NH_3), nitrogen oxides (NOx), benzene, carbon dioxide (CO_2), and smoke. It provides analog output corresponding to gas concentration, which is used to estimate the Air Quality Index (AQI).

- DHT11 Sensor:

A digital sensor used to measure temperature and relative humidity. It is simple, low-cost, and provides reasonably accurate readings for environmental monitoring applications.

- Jumper Wires:

Flexible wires with pins used to make electrical connections between components on a breadboard and the microcontroller without soldering. They simplify the prototyping process.

- Breadboard:

A reusable platform for building and testing electronic circuits without soldering. It allows quick and easy insertion and removal of components and wires for prototyping.

- USB Cable:

Used to connect the ESP8266 NodeMCU to a computer for programming and also to provide power during operation or testing.

- Power Bank (optional for portability):

A portable power source that can be used to run the system when a direct power supply is not available. This makes the system mobile and suitable for outdoor or remote deployment.

3.1.1 Software Requirements

Arduino IDE

- Firebase Realtime Database
- Blynk IoT Platform
- Web Development Tools (HTML, CSS, JavaScript)
- MQTT or HTTP Protocol Libraries (optional)

CHAPTER 4

PROJECT DESCRIPTION

4.1 SYSTEM ARCHITECTURE

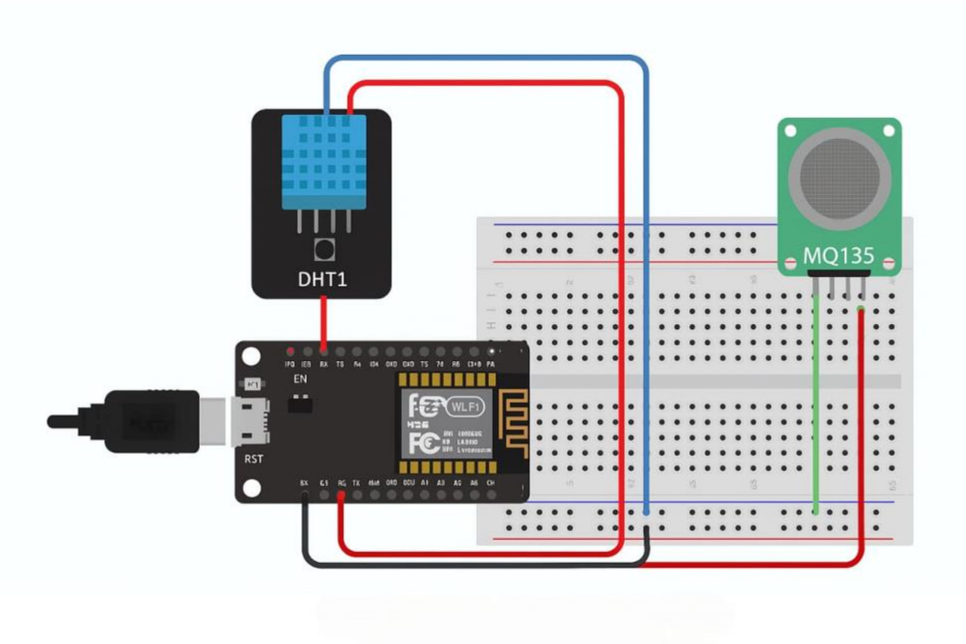

Fig 4.1 System Architecture

4.2 METHODOLOGY

1. Problem Definition

The rising levels of air pollution in urban and semi-urban areas pose significant health risks to the population. Traditional air monitoring stations are costly, fixed in location, and provide generalized data that may not reflect the real-time air quality experienced by individuals in specific locations. Additionally, people lack access to tools that provide real-time, location-specific air quality data along with actionable insights.This project aims to develop a Smart Air Quality Monitoring System that offers real-time monitoring of air quality, temperature, and humidity, stores the data to the cloud, and provides dynamic user guidance through a mobile app and web dashboard. The system will utilize affordable components such as ESP8266, MQ135, and DHT11 sensors to make the solution accessible and scalable.

2. Literature Review

Several air quality monitoring solutions have been developed by government and research institutions. However, most of them are:

- Expensive to deploy and maintain.
- Limited to a few locations.
- Do not provide real-time alerts or health suggestions to individuals.

Some earlier works include:

- Government AQI portals that display city-wide air quality data but lack personalization.
- Commercial air quality devices (e.g., AirVisual, Atmotube) which are accurate but costly.
- Academic projects using Raspberry Pi or Arduino for monitoring pollutants but lacking cloud integration or mobile interfaces.

Our proposed system enhances previous efforts by providing a portable, low-cost, IoT-based device integrated with cloud services (Firebase), mobile access (Blynk), and web dashboard capabilities.

3. Requirements Analysis

3.1 Functional Requirements

- Collect air quality data using MQ135 sensor.
- Measure temperature and humidity using DHT11 sensor.
- Transmit data to Firebase in real-time using ESP8266.
- Display live readings and historical data on a web dashboard and mobile app.
- Provide dynamic health tips and outdoor activity suggestions based on AQI.
- Send alerts when AQI exceeds safe thresholds.

3.2 Non-Functional Requirements

- The system must be low cost and energy efficient.
- It should support real-time data upload and access.

- The interface should be user-friendly and responsive.
- The system should be scalable to support multiple sensor nodes.

4. System Design

4.1 Hardware Architecture

- MQ135 → measures gas concentration (CO2, NH3, etc.)
- DHT11 → measures temperature and humidity
- ESP8266 → microcontroller for data processing and Wi-Fi connectivity
- Power Source → USB or battery backup

4.2 Software Architecture

- ESP8266 code written in Arduino IDE
- Firebase Realtime Database for storing sensor data
- Blynk App for mobile interaction
- HTML/CSS/JS + Firebase APIs for the web dashboard
- AQI classification and activity/health tip logic in backend script

(Diagram can be inserted here showing Sensors → ESP8266 → Firebase → Blynk & Web)

5. Prototype Development

- Built and tested the circuit on a breadboard.
- Calibrated MQ135 sensor to map analog readings to approximate AQI.
- Wrote Arduino code to read sensor data and push to Firebase.
- Developed a Blynk dashboard showing temperature, humidity, and AQI.
- Created a simple web interface using Firebase SDK and Chart.js to show AQI trends.
- Implemented logic to display outdoor activity recommendations and health advice based on AQI ranges.

6. Evaluation and Testing

6.1 Functional Testing

- Verified sensor readings using controlled environments (e.g., incense stick for poor AQI).

- Tested Firebase integration by observing real-time updates.
- Checked UI display consistency across devices.

6.2 Performance Testing

- Ensured data updates every 10 seconds without lag.
- Verified system behavior under poor Wi-Fi conditions (data buffering or delay).

6.3 Usability Testing

- Collected user feedback on app interface and web dashboard.
- Improved button layout, font sizes, and real-time responsiveness based on input.

6.4 Limitations

- MQ135 provides approximate AQI; not as accurate as industrial sensors.
- DHT11 has limited temperature range and precision.
- No GPS-based location awareness (can be a future enhancement).

CHAPTER 5
RESULTS AND DISCUSSION

RESULT:

The Smart Air Quality Monitoring System was successfully implemented and tested in a real-world environment. The following outcomes were observed:

- The MQ135 sensor was able to detect air quality variations effectively. During controlled testing (e.g., exposure to smoke), the AQI values increased, reflecting poor air quality conditions.
- The DHT11 sensor accurately measured temperature and humidity within the expected range (±2°C for temperature and ±5% for humidity).
- The ESP8266 microcontroller successfully connected to Wi-Fi and sent real-time data to the Firebase Realtime Database without noticeable delays.
- The Blynk mobile app displayed live AQI, temperature, and humidity values, and users received real-time alerts when AQI values crossed safe thresholds.
- The web dashboard effectively visualized AQI trends using interactive charts, allowed user login/profile access, and dynamically updated health tips and activity suggestions based on AQI levels.

Dynamic suggestions during testing:

- When AQI < 50: "It's safe to go jogging or cycling outdoors."
- When AQI > 100: "Avoid strenuous outdoor activity. Consider wearing a mask."

Discussion:

The results confirmed that the system meets its core objectives:

- It provides real-time, continuous environmental monitoring.
- The data is effectively stored and retrieved from Firebase.
- The Blynk app and web dashboard offer user-friendly, informative interfaces.
- Health tips and outdoor suggestions change dynamically with changing air quality.

However, some observations and insights emerged during testing:

- The MQ135 sensor, while affordable, is sensitive to multiple gases and not specific to any single pollutant. This affects AQI accuracy and calibration.
- Temperature and humidity readings were sometimes delayed during high sensor polling frequency.
- Firebase's free-tier usage limits may affect data throughput and logging for larger deployments.
- The system's usefulness increases significantly when installed in multiple locations (scalability potential).

Despite limitations, the prototype proved reliable and responsive in normal environmental conditions and showed the potential for wide adoption in schools, homes, and public places.

CHAPTER 6

CONCLUSION AND FUTURE WORK

Conclusion

The Smart Air Quality Monitoring System developed in this project effectively fulfills its goal of providing real-time monitoring of environmental conditions, including air quality (AQI), temperature, and humidity. Using low-cost components such as the ESP8266 microcontroller, MQ135 gas sensor, and DHT11 sensor, the system successfully gathers and transmits environmental data to Firebase and displays it on both a mobile application (Blynk) and a custom web dashboard.

The system not only visualizes live sensor readings but also offers personalized suggestions regarding outdoor activity suitability and health precautions based on AQI levels. This makes it a valuable tool for individuals to make informed decisions about their daily routines and well-being in response to environmental conditions.

The integration of IoT with cloud computing in this system demonstrates the potential for scalable, user-friendly, and low-cost air monitoring solutions, especially for urban and semi-urban areas where air pollution is a growing concern.

Future Work

While the current prototype is functional and efficient, several enhancements can be made to increase its accuracy, scalability, and usability:

- Upgrade to more precise sensors for measuring specific pollutants such as PM2.5 and PM10 for improved AQI accuracy.
- Implement GPS functionality to provide location-specific air quality monitoring and mapping.
- Introduce SMS or email alerts for critical AQI levels to reach users even without internet access.
- Add machine learning to predict AQI trends and offer personalized health

recommendations.

- Enable historical data export and advanced analytics on the web dashboard.
- Integrate the system with smart home devices (e.g., air purifiers) for automatic response to poor air quality.
- Develop a native Android/iOS application with an improved UI/UX beyond the Blynk interface.

Sample code

```
#include <DHT.h>

// Pin Definitions

#define DHTPIN D4        // DHT11 connected to digital pin D4

#define DHTTYPE DHT11    // Define the sensor type

#define MQ135PIN A0      // MQ135 connected to analog pin A0

#define ALERT_PIN D2     // LED/Alert pin for high AQI

DHT dht(DHTPIN, DHTTYPE);

void setup() {

  Serial.begin(9600);

  dht.begin();


  pinMode(ALERT_PIN, OUTPUT);

  digitalWrite(ALERT_PIN, LOW);


  Serial.println("Smart Air Quality Monitor Initialized");

}
```

```
void loop() {
  // Read DHT11 values
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  // Read MQ135 value (raw)
  int aqi = analogRead(MQ135PIN);

  // Handle DHT failure
  if (isnan(temperature) || isnan(humidity)) {
    Serial.println("Failed to read from DHT sensor");
    delay(2000);
    return;
  }

  // AQI Status Category
  String status;
  if (aqi <= 50) status = "Good";
  else if (aqi <= 100) status = "Satisfactory";
  else if (aqi <= 200) status = "Moderate";
  else if (aqi <= 300) status = "Poor";
  else if (aqi <= 400) status = "Very Poor";
  else status = "Severe";
```

```
  // Alert if AQI is too high
  if (aqi > 300) {
    digitalWrite(ALERT_PIN, HIGH);
  } else {
    digitalWrite(ALERT_PIN, LOW);
  }

  // Print to Serial Monitor
  Serial.println("----------------------------");
  Serial.print("Temperature: "); Serial.print(temperature); Serial.println(" °C");
  Serial.print("Humidity: "); Serial.print(humidity); Serial.println(" %");
  Serial.print("AQI (raw): "); Serial.print(aqi); Serial.print(" => ");
  Serial.println(status);
  Serial.println("----------------------------");

  delay(5000); // Wait 5 seconds
}
```

REFERENCES

[1] R. Kumar et al. Air quality monitoring system based on IoT using ESP8266 and MQ135.

[2] G. Anjana et al. IoT based air pollution monitoring and prediction system using ML techniques.

[3] R. Singh et al. Real-time air quality monitoring system using IoT.

[4] T. Malche et al. Internet of Things (IoT) for building smart home system.

[5] S.P. Mohanty et al. Everything you wanted to know about smart cities: The IoT is the backbone.