International Journal of Emerging Research in Engineering and Technology
Pearl Blue Research Group | Volume 7 Issue 2 PP 165-175, 2026
ISSN: 3050-922X | https://doi.org/10.63282/3050-922X.IJERET-V7I2P121

Original Article

Environmental Transparency and Air Pollution Control using Continuous Emissions Monitoring Systems (CEMS)

Pankaj Raj[1], Prof.Harsh Vikram Singh[2], Dr.Snehi Srivastava[3], Praveen Kumar[4] , Yuvraj Yadav[5], Utkarsh Tiwari[6]
[1,4,5,6] Research Scholar, Kamla Nehru Institute of technology Sultanpur (228118),Uttar Pradesh.
[2]Head of Department, Kamla Nehru Institute of technology Sultanpur (228118),Uttar Pradesh .
[3]Guide, Kamla Nehru Institute of technology Sultanpur (228118),Uttar Pradesh.

Received On: 24/03/2026 ***Revised On: 23/04/2026*** ***Accepted On: 01/05/2026*** ***Published On: 07/05/2026***

Abstract - *Due to the high levels of pollutants like SO_2, NO_2, CO and PM2.5/PM10, air pollution has become a large problem for the environment and human health, caused by high rate of industrialization and urbanization. Real-time assessment of air quality is challenging in absence of low-cost and continuous monitoring systems, particularly in developing regions. The objective of this study is to create an affordable IoT-based Continuous Emission Monitoring System (CEMS) for real-time monitoring and analysis of air pollutants. The methodology comprises of integrating MQ135, MQ7, PMS7003 and BMP280 sensors with Arduino Uno and ESP32 microcontroller and making this device compatible with cloud connectivity by using IoT based cloud service called ThingSpeak. Multiple Linear Regression is used for machine learning-based calibration to improve the accuracy by correlating the sensor outputs with CPCB reference data and minimizing the environmental effects. It can be seen that the system is able to monitor the pollutant change in real time with a calibrated performance of R^2 = 0.87 and RMSE = 6.5 µg/m³. The hardware testing proved the sensors to be stable in operation: the MQ135 sensor was able to detect air quality changes in the range of 50–180 µg/m³, the MQ7 sensor measured the concentration of CO up to 25,000 µg/m³, and the PMS7003 sensor consistently measured the concentration of particles in the range of 60–200 µg/m³. The cloud transmission was reliable with an average delay of 10-15 seconds, achieved by ESP32, and the buzzer alert system responded quickly to hazardous conditions. The suggested CEMS provides an economical, scalable, and effective way to monitor air pollution continuously, improving environmental transparency and facilitating improved air quality management.*

Keywords - *Internet of Things (IoT), Hardware, Air Pollution, CEMS, Environmental Transparency, Machine Learning.*

1. Introduction

In the meanwhile, the concept of the IoT system is growing quickly in this age of industry. The health and safety and other consequences of employees have become an important concern for many industrial organizations and other sectors [1]. The Internet of Things can monitor and control physical devices connected to the Internet (wired or wireless networks), from anywhere in the world. These days, environmental problems may lead to major catastrophes. Air pollution and sound pollution are two major problems [2]. With increasing industrial production, the amount of cars on the road and the number of people using air conditioning, air pollution has become a serious environmental issue [3]. The monitoring of industrial emissions including sulphur dioxide (SO_2), nitrogen oxides (NO_x), carbon dioxide (CO_2) and particulate matter (PM2.5/PM10) is important to ensure sustainability and transparency in the environment [4]. The work of CEMS is to monitor pollutants in real time and to assist industries in enhancing the efficiency of pollution control. CEMS are continuous emissions monitoring systems that measure and record the emissions from industrial processes in real-time [5]. These systems help industries and government authorities track pollution levels accurately and maintain environmental transparency and air pollution control [6]. CEMS improves environmental transparency, enhances data accuracy, and supports pollution control strategies for both government and industrial authorities[7]. It also provides data to create public awareness and help people take necessary precautions. Furthermore, it helps reduce emissions and supports pollution-free environmental monitoring for the public [8].

Due to the rapid growth of industries and transportation, emissions of SO_2, NO_x, CO_2, and PM2.5/PM10 in the atmosphere are continuously increasing. These emissions negatively affect human health, ecosystems, and environmental conditions, which lead to detrimental effects on the economy and social life of India [9]. Therefore, it has become necessary for industries and manufacturing units to install Continuous Emissions Monitoring Systems (CEMS) for proper monitoring and control of industrial emissions. However, several challenges are associated with the implementation and operation of CEMS [10]. High maintenance requirements, extreme temperatures, heavy dust, and corrosive chemicals often cause sensor failures. In addition, calibration challenges require frequent manual recalibration to maintain accuracy, which becomes difficult in remote industrial areas [11]. These systems also have high setup, installation and continuous maintenance expenses making them cost prohibitive for many industries. These issues cause the monitoring to become less efficient and indirectly lead to higher environmental pollution due to continuous emission of pollutants like SO_2, NO_x, CO_2, PM2.5/PM10.

This research paper seeks to assess the environmental transparency and data quality analysis of CEMS in the industrial sector, analyze the performance efficiency of CEMS and control the air pollution emitted in real-time in the industrial sectors, based on the primary and secondary data resources used in the real-time monitoring of emissions of SO_2, NO_x, CO_2 and PM2.5/PM10.

1.1. Objectives and Hypothesis of the Study

The following research Objectives and hypotheses are:

- The main purpose of this project is to develop a feasible and low-cost CEMS for real-time monitoring and control of air pollution. The goal of the system is to use inexpensive sensors and an ESP32 microcontroller to measure significant air pollutants like SO_2, NOx, CO_2, CO, VOCs and PM2.5/PM10.
- This study is focused on developing an IoT based monitoring system to continuously monitor the level of pollutants and to compare the results with the standard values of air quality limits. The project also aims to create an affordable system with a budget of less than ₹10,000, which will allow small industries and local units to monitor environmental pollution without relying on costly monitoring equipment.
- Another objective of this research is to send real-time sensor data to cloud platforms such as Thing Speak for live monitoring, storage, and environmental transparency. Basic calibration methods and AQI calculation techniques are also included to improve sensor performance and provide early pollution alerts.
- The study further aims to encourage industries to regularly monitor emissions and take steps to reduce air pollution. It also focuses on protecting human health, nature, ecosystems, and environmental conditions through continuous monitoring and awareness. Through this work, the researcher hopes to make a small contribution toward a cleaner environment, sustainable development, and a healthier India.

Hypothesis: The Null Hypothesis (H_0) states that the implementation of a low-cost CEMS does not create a significant improvement in real-time air pollution monitoring or pollution-control efficiency. The Alternative Hypothesis (H_1) states that the implementation of a low-cost CEMS improves real-time monitoring and analysis of pollutants such as SO_2, NOx, CO_2, CO, VOCs, PM2.5, and PM10, while also supporting environmental transparency and better pollution-control practices.

1.2. Structure of Paper

The following paper are organized as: Section II provide the literature review, Section III give the methodology of this system with block diagrams, then Section IV evaluate the results of design system with hardware specification and last Section v provide the conclusion and future work.

2. Literature Review

Air pollution is an important environmental problem especially in cities that can cause a huge impact on the health of their inhabitants. Air quality monitoring and control are critical in the context of increased industrialization and vehicle emissions that adversely affect air quality. The various existing researcher analysis the control of air pollution using various tools and techniques some are discussed below:

For example, R. Barrientos-Mauricio (2026) study outlines an air pollution monitoring system based on the IoT system for an artisanal brick kiln, which is a source of air pollution because of the use of fossil fuels and manual methods that can affect surrounding areas, particularly due to the high fine particulate matter (PM2.5) concentration that is known to have adverse effects on human health, such as respiratory and cardiovascular diseases. The system takes readings from an SPS30 sensor, which measures PM2.5 and PM10, and sends them over the MQTT protocol to the AWS IoT Core service. The information is then analyzed and shown via a web interface in real-time. During validation, the system showed great accuracy, outperforming an official monitoring station with a correlation of r=0.95 for PM2.5 and r=0.99 for PM10. The experimental results from the artisanal brick kiln showed that PM2.5 concentrations were higher than the 75 $\mu g/m^3$ limit on all days that were observed, and that PM10 concentrations were higher than 150 $\mu g/m^3$ on 30% of the days that were analyzed. The proposal suggests a low cost, scalable solution to improving environmental management in a variety of industries [12].

In this paper, G. Aprilia et al. (2025) present a study of an air quality monitoring system based on IoT, designed for efficient real-time monitoring. Several sensors are used, and the proposed system utilizes these to 83.3% levels. The 48 hour test was performed and the system was able to maintain stable and accurate readings, indicating its potential for long-term environmental monitoring. The system developed for this research is designed as a basic system to acquire real-time data and further research will involve incorporating machine learning models that can be used for predictive analytics [13].

In this paper, M. Ramadan et al. (2024) unveils a system created especially for the chrome plating industry to monitor and forecast air pollution in real time. The system monitors air pollution levels in real-time and can identify a variety of pollutants, including NH3, CO, NO2, CH4, CO2, SO2, O3, PM2.5, and PM10, with the use of IoT sensors and AI techniques. The sensors' data is analyzed using models such as LSTM, RF, and Linear Regression in order to forecast the levels of pollution. For humidity and temperature forecasting, the LSTM model attained a R^2 of 99% and an MAE of 0.33. The RF model achieved the best results for PM2.5, with a R^2 of 84% and an MAE of 10.11. The next step for the gadget is to proactively enhance air quality before problems even start by activating exhaust fans at the plant to circulate air before the next several hours of heavy pollution are predicted [14].

L. Tang et al. (2023) showcase the CIED, a countrywide database of industrial emissions that improves the estimate accuracy by utilizing actual smokestack concentrations from China's CEMS network from 2015 to 2018. This hourly data from the CEMS system allows us to directly estimate the industrial emission factors and absolute emissions, doing away with the many assumptions and indirect parameters often utilized in earlier research. The uncertainty analysis of the CIED database proves that their calculations are reliable, showing that the uncertainty ranges are quite narrow, falling within ±7.2% for emission factors and ±4.0% for emissions. The usage of this dataset by scientists and policymakers in China can improve their understanding of the sources of industrial emissions, such as the concentrations in smokestack, the emissions variables, activity data, and the absolute emissions [15].

S. Listyarini et al. (2023) investigation into the possibility of creating CEMS that are IoT-based was undertaken. On the Arduino Uno microcontroller are mounted pollutant sensors that detect methane (CH4), butane (C4H10), CO, PM2.5, and ammonia (NH3). If they want to know how much methane (CH4), butane (C4H10), carbon monoxide (CO), PM2.5, or ammonia (NH3) present in the air, they may use a MQ-7 or MQ-2 sensor. A Liquid Crystal Display (LCD) receives data from the sensor and displays it numerically on smartphones using the Arduino Uno microcontroller. Continuous tracking of air pollution emissions should be achieved via this web-based air quality monitoring platform. The most important discovery is that on June 15, 2022, in South Tangerang, Indonesia, the concentration of Particulate Matter was 52.29 µg/m3, which is higher than the emission quality requirement compared to other sensors. In light of these findings, it is recommended to regularly assess air quality for PM2.5 [16].

Thus S. Malleswari et al. (2022), it is possible to monitor the local air quality with the help of sensors and IoT devices connected to an Arduino or Raspberry Pi. The study's overarching goal is to shed light on environmental pollution-related data by gaining a better grasp of environmental variable information and making it easy to integrate with other forms of IoT. This will enable the use of sensors that can gather data on smart city environment measurements [17].

Harish G N et al. (2021) study introduces a method for monitoring air pollution using the IoT and sensor networks to gather data on air pollutants such PM2.5, CO, and NO_2 in real-time. For further processing, the data is sent wirelessly to a cloud service. The goal of using machine learning methods such as SVM and DT is to forecast patterns in pollution and identify outliers. The design of the system is described, showcasing its flexibility and scalability, beginning with the deployment of sensors and ending with data analytics. Experiments performed over a 30-day period in an urban environment have shown the system to be reliable in measuring pollution and predicting it. The incorporation of IoT and ML into the system offers a timely, economical way to monitor and predict air pollution, benefiting public health and urban development [18].

The above literature highlights the successful implementation of low-cost sensors, cloud platforms and machine learning methods for the real-time monitoring of pollutants like PM2.5, NO_2, CO and other gases in CEMS applications. In addition, some systems can also integrate advanced AI models like LSTM and RF for enhanced predictions and smart environmental monitoring. Most of the existing studies are limited to a few pollutants or are not calibrated with the reference data provided by official CPCB or rely on high-cost infrastructure for high accuracy. There is thus a research gap in the development of a cost-effective multi-pollutant IoT-based CEMS that can be reliably calibrated using machine learning and has real-time cloud monitoring for practical deployment in developing regions.

Table I, available below, summarizes various works of research on IoT-based air pollution monitoring and CEMS, along with the technologies utilized, important contributions and the weaknesses of each of the works. It is evident from its evolution that it started from sensor-based monitoring and has advanced to AI and machine learning-based pollution prediction and analysis systems.

Table 1: Summary of Related Work on IoT-Based Air Pollution Monitoring and CEMS

Ref.	Author & Year	System / Focus	Technology Used	Key Contribution	Limitation
[12]	R. Barrientos-Mauricio (2026)	Brick kiln air monitoring	SPS30 sensor, ESP32, MQTT, AWS IoT	High accuracy PM monitoring with strong correlation (r = 0.95–0.99) and real-time visualization	Limited to industrial brick kiln environment only
[13]	G. Aprilia et al. (2025)	IoT air quality monitoring system	Multiple sensors, IoT platform	Real-time monitoring with stable 48-hour performance	No advanced ML-based prediction or calibration
[14]	M. Ramadan et al. (2024)	Industrial air pollution forecasting	IoT sensors, LSTM, Random Forest	High prediction accuracy (R^2 up to 99% for forecasting) and smart exhaust control	Complex system and high computational requirements
[15]	L. Tang et al. (2023)	Industrial emissions	CEMS network data	Large-scale accurate emission dataset with low	Focus on data analysis, not real-time

		database (CIED)		uncertainty (±4%–7.2%)	low-cost systems
[16]	S. Listyarini et al. (2023)	IoT-based CEMS prototype	Arduino, MQ sensors, GP2Y1010AU0F	Multi-gas monitoring with real-time display and PM detection	Limited calibration and accuracy issues
[17]	S. Malleswari et al. (2022)	IoT environmental monitoring system	Arduino/Raspberry Pi, sensors	Easy integration into smart city IoT systems	Basic system without advanced analytics
[18]	Harish G N et al. (2021)	Air pollution prediction system	IoT sensors, ML (SVM, Decision Tree)	Real-time monitoring with predictive analytics and anomaly detection	Requires large datasets and higher computational resources

3. Methodology

This research has been undertaken in a structured methodology approach of literature analysis, hardware implementation, sensor calibration, cloud-based monitoring and performance evaluation to develop low-cost CEMS in India. Both primary and secondary data sources are used in the study to investigate the performance of the system, the transparency of the environment, and the control of air pollution through Continuous Emissions Monitoring Systems (CEMS). The primary research for the study concentrated on the problems faced in monitoring, sensors calibration techniques, financial considerations, and environmental monitoring practices while implementing CEMS in India. To gather reliable information on air quality monitoring, environmental transparency and CEMS technologies, secondary data were gathered from research papers, government reports, surveys, and digital academic platforms (Google Scholar, Research Gate and other scientific databases). The Practical implementation of the proposed CEMS model for Environmental transparency and air pollution control, was implemented in the following four stages:

3.1. Phase 1 — Hardware Integration

All sensing modules such as MQ-136 sensor (SO_2), MiCS-2714 sensor (NO_2), MH-Z19B sensor (CO_2), MQ-7 sensor, MQ-135 sensor, PMS7003 sensor, BME280 sensor were integrated with ESP32 microcontroller on a solderless breadboard. Particulate matter (PM2.5/PM10) was monitored using the PMS7003 sensor, and the BME280 sensor was used to measure the temperature, humidity and Atmospheric Pressure. Prior to full system integration, each individual sensor was tested to confirm proper functionality and stable data acquisition for the real-time emission monitoring system (CEMS) to ensure proper operation of each sensor.

3.2. Phase 2 — Firmware Development

The firmware is written in C++ with the Arduino IDE. Wireless communication and MQTT data transmission were implemented with the use of libraries like WiFi.h and PubSubClient.h. The firmware was continuously responsible for the acquisition, processing, calibration, and communication with the clouds, to enable the efficient monitoring and control of air pollution in the environment, with the possibility of the proposed CEMS model.

3.3. Phase 3 — Sensor Calibration and Data Validation

The raw sensor data collected were calibrated using Multiple Linear Regression (MLR) method, with reference datasets from CPCB monitoring stations and standard reference instruments. The calibration coefficients were used to reduce SensorDrift and enhance the accuracy of the measurement. The reliability and consistency of the monitored data for the effective environmental transparency and emission analysis by CEMS, were validated with statistical methods.

3.4. Phase 4 — Cloud Deployment and Data Visualization

Real-time environmental monitoring and data quality analysis were achieved by implementing a cloud-based monitoring and visualization platform with ThingSpeak. Cloud fields were setup for the variables SO_2, NO_x, CO_2, PM2.5/PM10 concentration, temperature, humidity and Atmospheric Pressure. The ESP32 securely communicated sensor data in a processed way with MQTT protocol using API keys. Real-time graphs and dashboards were used for continuous environmental data visualization, environmental transparency and air pollution monitoring using Continuous Emissions Monitoring Systems (CEMS).

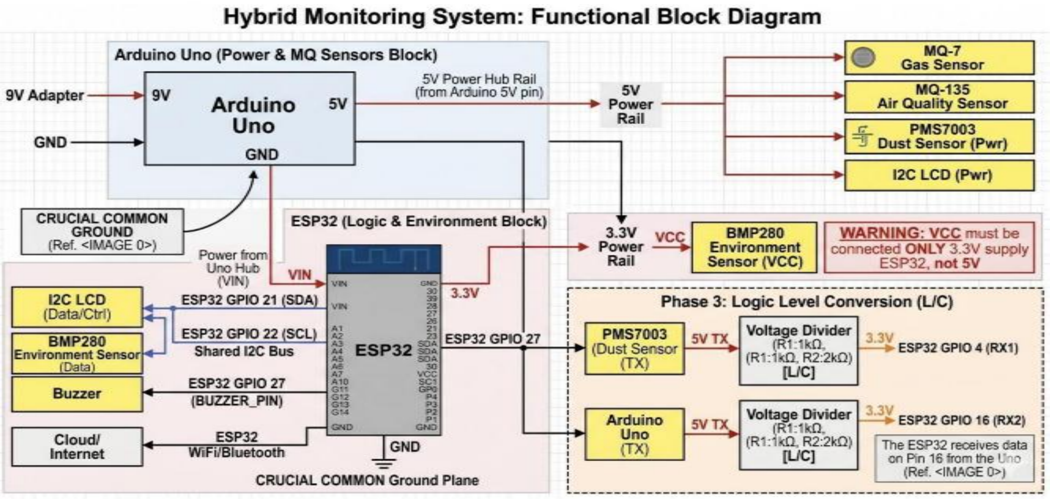

Fig 1: Functional Block Diagram

The functional Block Diagram (Fig. 1) shows a Hybrid Monitoring System where the power distribution and MQ sensor operations are done at 5V using an Arduino Uno and the logic, environment sensing and cloud/internet communication are handled at 3.3V with an ESP32. The Arduino is powered by a 5V power rail, which is used to power the peripherals such as the MQ-7, MQ-135, PMS7003, and I2C LCD. The ESP32 interfaces with the BMP280 environment sensor, buzzer, and cloud connectivity through WiFi/Bluetooth. One of the important design considerations, as shown in the diagram is the protection of ESP32 from overvoltage by using voltage dividers which steps down both the 5V TX signals from the PMS7003 dust sensor and the Arduino UNO to 3.3V for feeding into the ESP32 GPIO pins 4 and 16 during phase 3 Logic Level Conversion. There is an important common ground plane to ensure stable and accurate readings from the sensors throughout the system.

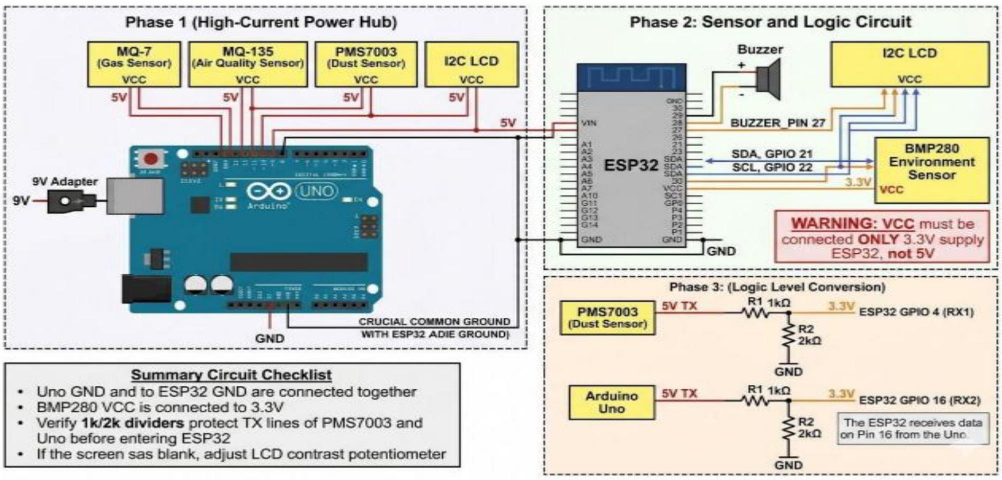

Fig 2: Circuit Diagram

The three phases of the circuit diagram (Fig. 2) are: Phase 1 - High Current Power Hub: The ESP32 is connected to the Arduino Uno using a common ground, and the Arduino Uno is powered by a 9V adapter, providing 5V to the Gas Sensor, Air Quality Sensor, dust sensor, and I2C LCD. Phase 2 deals with Sensor & Logic Circuit where the ESP32 will control the BMP280 environment sensor at 3.3V using the I2C (GPIO 21/22) bus, a buzzer (GPIO 27) and the I2C LCD, VCC can NOT exceed 3.3V or it will damage the ESP32. In Phase 3 Logic Level Conversion, 1kΩ/2kΩ voltage dividers are used to safely reduce the 5V TX outputs from both the PMS7003 and the Arduino Uno to 3.3V, and the summary checklist highlights shared ground connections, correct 5V output voltage from the PMS7003, voltage divider verification, and LCD contrast settings for blank screens.

3.5. System Implementation

Implementation of the Affordable IoT-Based Continuous Emission Monitoring System (CEMS) requires low-cost sensors, low-cost microcontrollers, cloud communication and Machine Learning calibration techniques to provide continuous and reliable environmental monitoring. The system must be able to track air pollutants in real-time, and must be simultaneously affordable, accurate and accessible via the cloud. The integration process involves connecting the hardware, calibrating the sensors, linking with the Internet of

Things, storing the data in the cloud, and finally, monitoring the environment in real-time.

3.5.1. Design and Planning

The first implementation phase involves creating the overall architecture of the proposed CEMS and choosing appropriate hardware components for effective environmental monitoring. Low-cost sensors that work reliably and are compatible and easy to integrate are chosen for gas sensors, particulate matter sensors, microcontrollers and power management units. A detailed circuit schematic is designed to inform the proper connections between the Arduino Uno, the ESP32, the sensors, the LCD display, the buzzer alarm and the cloud communication modules. The actual assembly is done on a solderless breadboard to ease testing and changes during development.

3.5.2. Sensor Calibration

Sensor calibration is performed to improve the accuracy and reliability of low-cost gas sensors. The sensors are first tested in clean air to acquire reference values. The sensors are MQ-series which are sensitive to the variation in temperature and humidity in the surrounding environment, so the technique of environmental compensation is applied by using the sensor data of the BMP280 sensor. Multiple testing cycles are carried out by comparing sensor outputs with standard reference values from CPCB monitoring stations. The next step is to reduce the sensor drift and increase the accuracy of the monitoring using Multiple Linear Regression (MLR) and ML techniques.

3.5.3. IoT Integration and Cloud Setup

The ESP32 microcontroller is programmed to create a Wi-Fi network to monitor the environment in the cloud. The system transmits data from the sensors to the ThingSpeak IoT platform via MQTT or HTTP protocol in real time and after calibration. Cloud fields are set to parameters like PM2.5, CO, AQI, temperature, humidity and atmospheric pressure. A cloud dashboard is built to present the environment conditions in graphical, gauge and historical data record format, for remote monitoring and analysis.

3.6. Machine Learning Calibration

Data from the CPCB (CAAQMS) is used to obtain accurate concentrations of pollutants which are then trained by a supervised regression approach in the proposed CEMS using the raw readings obtained from the low-cost sensors. The process is broken down into distinct steps, each step can be shown to enhance the accuracy and eliminate the sensor error due to temperature, humidity, and drift.

3.6.1. Data Acquisition (Virtual Collocation)

In this step, the CEMS prototype is run continuously for 12–24 hours to collect real-time environmental data. For example, suppose the MQ135 sensor gives raw values like X1 = 420, 380, 450, while temperature and pressure from BMP280 are X2 = 32°C, 31°C, 30°C and X3 = 1012 hPa, 1010 hPa, 1008 hPa. At the same time, corresponding CPCB reference values for PM or gas concentration are collected, such as y = 55 µg/m³, 60 µg/m³, 65 µg/m³. This synchronized dataset forms the base for calibration.

3.6.2. Data Pre-processing and Feature Engineering

The collected datasets are synchronized and standardized by converting local sensor readings into hourly averages. In this step, the raw data is cleaned and converted into hourly averages. For example, 15-minute readings like 420, 430, 410, 400 are averaged to form a single hourly value like X1 = 415. Similarly, temperature may be averaged to 31.5°C and pressure to 1011 hPa. The final dataset is structured as input features: X1 (sensor output), X2 (temperature), X3 (pressure) with target output y (CPCB value) such as y = 60 µg/m³.

3.6.3. Multiple Linear Regression (MLR) Training

The MLR algorithm is used to estimate the calibration coefficients and intercept values that minimize the deviations between prototype readings and CPCB readings. The environmental cross-sensitivity and thermal drift effects are minimized by incorporating temperature and pressure variables. In this stage the relationship between the inputs and output is modeled by regression. The equation used is:

$$\mathbf{y = \beta 0 + \beta 1X1 + \beta 2X2 + \beta 3X3}$$

For example, after training, assume the model produces coefficients like:

$$\mathbf{\beta 0 = 5.2,\ \beta 1 = 0.08,\ \beta 2 = -0.3,\ \beta 3 = 0.01}$$

If inputs are X1 = 415, X2 = 31.5, X3 = 1011, then predicted value becomes:

$$\mathbf{y = 5.2 + (0.08\times 415) + (-0.3\times 31.5) + (0.01\times 1011)}$$
$$\mathbf{y \approx 5.2 + 33.2 - 9.45 + 10.11 = 39.06\ \mu g/m^3}$$

This predicted value is closer to CPCB reference after repeated training iterations.

3.6.4. Statistical Validation

Various statistical parameters like RMSE and Coefficient of Determination (R^2) are used to assess the performance of the calibration model. High R^2 values mean that there is good agreement between the values from the prototype and the reference monitoring station.

3.6.5. Firmware Integration and Deployment

Lastly, the trained coefficients (β0, β1, β2, β3) are stored in the ESP32 code. The microcontroller calculates the equation in real-time. In this case, the ESP32 takes the live sensor inputs in the form of X1 = 500, X2 = 33°C, X3 = 1009 hPa and immediately computes the calibrated output before pushing it to LCD and ThingSpeak. Also applied is a safety rule which prevents negative values (minimum = 0 µg/m³). Quasi-periodic retraining (for example, every 3 months) is employed to keep the accuracy and minimize long-term sensor drift.

774	2026-05-0	773	31.84		42	175	105	396	993.7065
775	2026-05-0	774	31.95		43	179	67	267	993.7346
776	2026-05-0	775	31.99		42	175	66	250	993.7221
777	2026-05-0	776	30.72		43	179	81	348	994.1461
778	2026-05-0	777	31.07		47	195	79	332	994.1619
779	2026-05-0	778	31.2		47	195	73	269	994.173
780	2026-05-0	779	31.46		48	200	72	238	994.1652
781	2026-05-0	780	31.04		51	212	76	318	994.3315
782	2026-05-0	781	31.16		51	212	65	248	994.3348
783	2026-05-0	782	31.26		51	212	62	228	994.3297
784	2026-05-0	783	31.33		51	212	63	220	994.3746
785	2026-05-0	784	31.39		52	216	71	219	994.351
786	2026-05-0	785	31.45		51	212	62	208	994.3698
787	2026-05-0	786	31.51		50	208	62	204	994.3848
788	2026-05-0	787	31.57		566	2358	69	246	994.3525
789	2026-05-0	788	31.65		153	637	65	235	994.3265
790	2026-05-0	789	31.69		226	941	62	224	994.3436
791	2026-05-0	790	31.73		108	450	63	218	994.367
792	2026-05-0	791	31.83		97	404	73	208	995.8003
793	2026-05-0	792	31.84		95	395	61	189	995.82
794	2026-05-0	793	31.91		96	400	61	186	995.6705
795	2026-05-0	794	31.93		94	391	60	184	995.6253
796	2026-05-0	795	32		94	391	59	185	995.6914
797	2026-05-0	796	32		94	391	62	185	995.6962
798	2026-05-0	797	31.98		96	400	62	184	995.7241
799	2026-05-0	798	31.97		96	400	62	183	995.6699
800	2026-05-0	799	31.97		94	391	62	183	995.5744
801	2026-05-0	800	31.98		103	429	58	177	995.6422
802	2026-05-0	801	31.96		106	441	57	175	995.6214

cems data 1

Fig 3: Sensors Extracted CEMS data

The CEMS data includes continuous real-time sensor data that is fairly uniform with temperature ranging from 31-32°C and pressure from 993-995 hPa for the majority of the entries, as seen in Fig. 3. A large increase in pollution is seen at entries 788-790, which increases from the normal value (~50) to 566, and then back down to 2358 and then back down close to the normal value, suggesting a short duration, but high emission event. This shows that the system can capture both normal conditions and sudden pollution incidents, effectively performing real-time environmental monitoring and responding to pollution incidents in time.

4. Results and Discussion

The hardware requirements, the obtained results of system implementation, and discussion of monitoring performance obtained by using the proposed model of CEMS are shown in the following section.

4.1. System Requirements Specification

The SRS for the proposed IoT-Based CEMS specifies the equipment and software needed to ensure real-time air pollution monitoring and establishing environmental transparency. The system integrates gas sensors, environmental sensors, IoT microcontrollers, and cloud-based platforms to provide reliable, accurate, and cost-effective monitoring of air pollutants for research and industrial applications.

4.1.1. Hardware Requirements

The proposed IoT-based CEMS consists of several hardware components. The hardware requirements of the proposed system are discussed below.

4.1.2. Arduino Uno

The ATmega328P is the basis of the open-source microcontroller board known as Arduino Uno. The prototype industry relies on it because of its power jack, USB connection, digital and analog connectors. In this project, the Uno acts as a power and data hub. It provides stable 5V power to the MQ sensors and transmits raw analog gas data to the ESP32. Fig. 4 shows the Arduino Uno hardware.

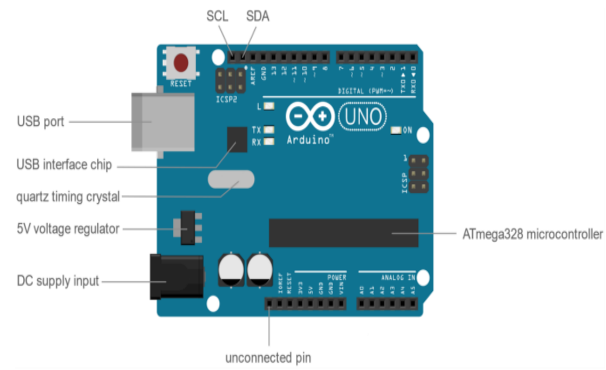

Fig 4: Arduino Uno

4.1.3. ESP32 Microcontroller

The ESP32 is a powerful, low-cost SoC featuring integrated Wi-Fi and dual mode Bluetooth. Designed for IoT applications, it boasts a dual-core processor, multiple GPIOs, and low Power Consumption. n this project, the ESP32 acts as the CPU and gateway. It receives raw data from the Arduino and PMS7003, applies Machine Learning calibration formulas, and manages the I2C LCD. Notably, it can be remotely monitored in real-time because to its Wi-Fi connectivity, which allows data transfer to the ThingSpeak cloud platform. Fig. 5 shows the ESP32 hardware.

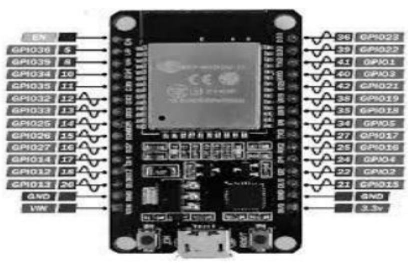

Fig 5: ESP32 Pin Configuration

4.1.4. PMS7003 Sensor

The PMS7003 is a high-precision digital laser dust sensor, which is equipped with light scattering sensor technology. It is designed to measure the concentration of

PM1.0, PM2.5 and PM10. With its compact design, it has an internal fan for uniform air flow, rapid, stable, accurate readings through a serial UART interface for environmental monitoring. The PMS7003 is the main detector of PM2.5 and AQI level in this CEMS project. It connects a voltage-protected serial link directly to ESP32. This high-resolution data is very important for the Machine Learning model to correlate the particulate trends with the gaseous pollutants on ThingSpeak. Fig. 6 shows the PMS7003 hardware.

Fig 6: PMS7003

4.1.5. MQ Sensors (MQ135 and MQ7)

MQ sensors are metal-oxide semiconductor (MOS) gas sensors used for detecting different atmospheric gases through changes in electrical resistance. In the proposed CEMS, the MQ-135 sensor is utilized for monitoring harmful gases such as NH_3, NO_x, and benzene to assess overall air quality, while the MQ-7 sensor is used for detecting CO emissions from combustion sources. Both sensors provide analog output signals to the Arduino Uno, and the collected data are further calibrated using Machine Learning techniques to improve monitoring accuracy according to CPCB standards. Fig. 7 shows the MQ hardware.

Fig 7: MQ Sensors

4.1.6. BMP280 Sensor

The BMP280 is a high-accuracy digital sensor used for measuring ambient temperature and atmospheric pressure with low power consumption and support for I2C/SPI communication. In the proposed CEMS, it provides real-time environmental data that help compensate thermal drift in MQ-series gas sensors and improve calibration accuracy using Multiple Linear Regression (MLR) techniques. The LCD panel shows the calibrated temperature and pressure measurements, and they are uploaded to the ThingSpeak cloud platform so that the environment can be monitored and analyzed in real-time. Fig. 8 shows the BMP280 hardware.

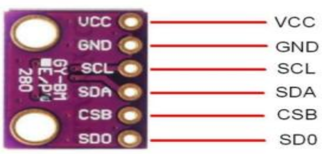

Fig 8: BMP280

4.1.7. Buzzer Alarm

A buzzer is a type of audio signaling equipment designed to produce an alarm sound in an electronic circuit. The buzzer in the proposed CEMS is used as a real-time safety alarm which is automatically triggered whenever the PM2.5 or AQI level is higher than the safety threshold, thus giving timely alert for the poor AQI condition, and also working in a controlled ON/OFF cycle which helps to minimize the continuous noise. Fig. 9 shows the buzzer hardware.

Fig 9: Buzzer Alarm

4.1.8. LCD Display

The I2C LCD display is a visual interface used for showing real-time environmental data with minimal wiring through I2C communication. In the proposed CEMS, it displays live PM2.5, CO, AQI, temperature, and system status information while dynamically cycling through different monitoring parameters and presenting calibrated sensor values for accurate environmental monitoring. Fig. 10 shows the I2C LCD display hardware.

Fig 10: I2C LCD Display

4.1.9. Resistor (1kΩ)

A resistor is a passive electronic component used to control electrical current and protect sensitive devices in a circuit. In the proposed CEMS, 1kΩ and 2kΩ resistors are used for voltage division, logic protection, and maintaining stable serial communication between the Arduino Uno, PMS7003 sensor, and ESP32 microcontroller. Fig. 11 shows the Resistor hardware.

Fig 11: Resistor(1k ohm0

4.1.10. Exhaust Fan

Exhaust fan is used to provide a continuous flow of the required air to be measured, and this exhaust fan operates on 12 v and 0.2 amp. Fig. 12 shows the Exhaust fan hardware.

Fig 12: Exhuast Fan

4.1.11. Power Adapter

Power adapter (or AC-DC converter) is an external power supply that steps down high-voltage household electricity (230V AC) to a safe, low-voltage DC output (9V or 12V). It converts alternating current into a steady direct current required by microelectronic components. It acts as a safety barrier, protecting the sensitive project from power surges or fluctuations in the main grid. Fig. 13 shows the Power adapter hardware.

Fig 13: Power Adapter

4.1.12. ThingSpeak Platform

Users are able to collect, view, and analyze real-time data streams in the cloud using ThingSpeak, an open-source IoT analytics platform. For complex data processing and real-time visualizations, it has built-in MATLAB support. The ESP32 acts as a gateway, transmitting calibrated PM2.5, CO, and AQI data to ThingSpeak via Wi-Fi. It serves as the project's central database, enabling remote monitoring through realtime graphs and providing the historical CSV logs required for Machine Learning re-calibration. Fig. 14 shows the ThingSpeak CEMS data.

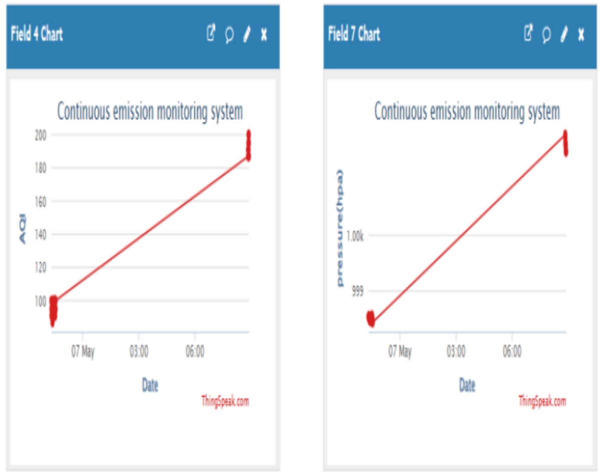

Fig 14: Thingspeak CEMS Data

4.2. Analysis and Interpretation

The analysis of pollutant concentrations monitored through the proposed CEMS shows that increasing levels of SO_2, NO_2, CO, and PM2.5/PM10 negatively impact human health and environmental quality. Table II presents the standard limits, air quality categories, and health impacts of major pollutants (SO_2, NO_2, CO, and PM2.5/PM10) monitored through the proposed CEMS. Classifies pollutants by concentrations as Good, Moderate, Poor, Unhealthy, Severe and Hazardous to facilitate real-time air quality assessment. SO_2 and NO_2 have most harmful effects on the respiratory system and are responsible for the formation of acid rain and smog at high concentrations. CO interferes with the transport of oxygen in the body and has significant health effects at high exposures. Particulate matter (PM2.5/PM10) has an important role in relation to visibility and lung health. In conclusion, the overall classification allows for efficient monitoring of pollution, clarity of the environment, and prompt action to control emissions by monitoring them in real time. Real-time continuous monitoring with a CEMS system enhances pollution monitoring, environmental transparency and pollution control.

Table 2: Pollution Classification and Health Impact Analysis of Major Air Pollutants Using CEMS

Pollutant	Standard Limit	Good	Moderate	Poor	Unhealthy	Severe	Hazardous	Major Environmental and Health Effects
Sulphur Dioxide (SO_2)	50 µg/m³ (24-hour)	0–40 µg/m³	40–80 µg/m³	80–380 µg/m³	380–800 µg/m³	800–1600 µg/m³	1600–2600 µg/m³	Causes respiratory irritation, coughing, breathing discomfort, mucus secretion, and contributes to acid rain formation affecting soil, crops, water bodies, and buildings.
Nitrogen Dioxide (NO_2)	40 µg/m³	0–40 µg/m³	40–80 µg/m³	80–180 µg/m³	180–190 µg/m³	190–400 µg/m³	400–500 µg/m³	Leads to lung irritation, respiratory infections, photochemical smog formation, reduced air quality, and decreased visibility.

Carbon Monoxide (CO)	2 mg/m³	0–8330 µg/m³	8330–16670 µg/m³	16670–25000 µg/m³	25000–33330 µg/m³	33330–41670 µg/m³	41670–50000+ µg/m³	Reduces oxygen transport in blood, causing headaches, dizziness, nausea, weakness, and breathing difficulties.
Particulate Matter (PM2.5 / PM10)	—	0–50 µg/m³	50–100 µg/m³	100–250 µg/m³	250–350 µg/m³	350–430 µg/m³	430–510+ µg/m³	Causes asthma, lung damage, respiratory diseases, breathing problems, and reduced visibility in industrial and urban areas.

Table 3: Statistical Validation Results of ML-Based Sensor Calibration Model

Performance Metric	Obtained Value	Interpretation
Coefficient of Determination (R^2)	0.87	Indicates strong correlation between predicted CEMS values and CPCB reference data, showing high model accuracy and reliability.
Root Mean Squared Error (RMSE)	6.5 µg/m³	Shows low prediction error, which is acceptable for low-cost IoT-based air quality monitoring systems.
Residual Error Analysis	No significant bias observed	Confirms that predictions are not consistently overestimating or underestimating pollutant levels, ensuring model stability.

The performance of the proposed CEMS calibration model based on ML is evaluated in Table III. The results prove the calibrated model's accuracy and effectiveness in real-time environmental monitoring and pollution control applications.

4.3. Discussion

This overall assessment of SO_2, NO_2, CO, PM2.5, and PM10 emissions shows the reliability and transparency of Continuous Emissions Monitoring Systems (CEMS) for monitoring industrial air pollutants. CEMS' real-time monitoring functionality enables industries to detect pollution sources, ensure compliance with regulations, and enhance environmental sustainability. The interpretation of pollutant levels also reveals that when the level of pollutants is higher than the acceptable level, they have a direct impact on the health of humans and the quality of the environment. The role of the implementation of CEMS is therefore an important one in matters of transparency in the environment, pollution prevention, industrial responsibility, and control of air pollution. The continuous monitoring and appropriate interpretation of the emission data can contribute to the sustainable development of industry and assist the regulatory authorities to timely take the necessary corrective measures to protect the well-being of people and the environment.

4.4. Limitations and Future Work

The proposed air pollutant monitoring system based on IoT technology is a CEMS that is based on low-cost gas and particulate sensors for measuring air pollutants like SO_2, NO_2, CO and PM2.5/PM10, which has some limitations. The sensors can exhibit lower long-term stability and changes in measurement precision when the environment changes. The MQ-series gas sensors are especially sensitive to temperature, humidity, and aging, and can cause drift in pollutant measurements over time, even after Machine Learning calibration. Moreover, if the internet connection is interrupted, the real-time cloud monitoring through ThingSpeak might cause data loss for a period of time. The system is also reliant on periodic re-training based on the CPCB reference data, which may not be synchronized with local variations in air pollutants.

Future enhancement of the CEMS could include the use of more accurate sensors for air pollutants, such as high precision industrial grade gas sensors (SO_2, NO_2, CO and fine particulate matter, PM2.5 and PM10). The possibility exists of implementing advanced ML techniques (RF, SVR or NN) to enhance the calibration of air pollutant data beyond MLR. Edge AI can introduce to make it possible to predict the quality of the air right on the device without always depending on the cloud. Moreover, the integration of mobile apps, GPS-based air pollution mapping, and deployment at a large scale of air pollution monitoring in industrial and urban areas can further enhance environmental transparency and facilitate smart city air pollution management systems.

5. Conclusion

The proposed Affordable IoT Based CEMS can be very well demonstrated as a low cost and real-time solution for monitoring major air pollutants like SO_2, NO_2, CO, PM2.5/PM10 and other Environmental parameters like temperature and pressure. It is a system that combines several gas and particle sensors, as well as ESP32-based IoT communication and cloud visualization, to provide air quality monitoring at all times. Applying ML-based calibration significantly enhanced pollutant measurement accuracy, making low-cost sensors more dependable for air quality evaluation and helping to mitigate the impacts of ambient interference and sensor drift. The live streaming of data to the ThingSpeak platform gives environment transparency and remote monitoring of air pollution levels. Further, the warning system can issue immediate warnings when concentrations of pollutants exceed safe levels, which can help with timely preventive measures. Overall, the developed CEMS provides a cost-effective, scalable, and convenient solution for monitoring air pollutants continuously, which helps to promote environmental protection and sustainable air quality management.

References

[1] V. K. Sharma, “Cloud Computing & IoT: 5G Focused IoT with Cloud Solutions,” *Int. J. AI, BigData, Comput. Manag. Stud.*, vol. 6, no. 3, 2025, doi: 10.63282/3050-9416.IJAIBDCMS-V6I3P103.

[2] A. H. Vasanth, “Air Pollution Monitoring System Using IOT,” *Int. J. Creat. Res. Thoughts*, vol. 9, no. 7, pp. 1–7, 2021.

[3] L. Bai, J. Wang, X. Ma, and H. Lu, “Air Pollution Forecasts: An Overview,” *Int. J. Environ. Res. Public Health*, vol. 15, no. 4, p. 780, Apr. 2018, doi: 10.3390/ijerph15040780.

[4] S. Sen, “Artificial Intelligence in Mining, Petroleum and Natural Gas Extraction Process Optimization,” *Int. J. Emerg. Res. Eng. Technol.*, vol. 6, no. 1, pp. 121–125, 2025, doi: 10.63282/3050-922X.IJERET-V6I1P115.

[5] A. V. Kychkin, O. V. Gorshkov, and M. A. Kukarkin, “Predictive models integration with an environmental monitoring IoT platform,” *J. Appl. Informatics*, vol. 17, no. 4, pp. 5–16, Aug. 2022, doi: 10.37791/2687-0649-2022-17-4-5-16.

[6] Y. Muni, “Sustainable Data Centers: a Systematic Review of Technologies and Practices for Carbon Emission Reduction,” *Int. J. Adv. Res. Comput. Sci.*, vol. 16, no. 4, pp. 125–131, 2025, doi: 10.26483/ijarcs.v16i4.7309.

[7] U. Korat and A. Alimohammad, “Efficient Hardware Implementation of Eigen Solver,” in *2026 IEEE 16th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, Jan. 2026, pp. 1376–1385. doi: 10.1109/CCWC67433.2026.11393888.

[8] Y.-L. Zhao *et al.*, “Development of IoT Technologies for Air Pollution Prevention and Improvement,” *Aerosol Air Qual. Res.*, vol. 20, no. 12, pp. 2874–2888, 2020, doi: 10.4209/aaqr.2020.05.0255.

[9] N. Ding *et al.*, “State-of-the-art carbon metering: Continuous emission monitoring systems for industrial applications,” *Heliyon*, vol. 11, no. 3, p. e42308, Feb. 2025, doi: 10.1016/j.heliyon.2025.e42308.

[10] S. Chen *et al.*, “Carbon and air pollutant emission dynamics of small-scale industrial sectors based on CEMS: a case of tobacco curing industry,” *Environ. Pollut.*, vol. 376, p. 126427, Jul. 2025, doi: 10.1016/j.envpol.2025.126427.

[11] X. Zhou, J. Aurell, W. Mitchell, D. Tabor, and B. Gullett, “A small, lightweight multipollutant sensor system for ground-mobile and aerial emission sampling from open area sources,” *Atmos. Environ.*, vol. 154, pp. 31–41, Apr. 2017, doi: 10.1016/j.atmosenv.2017.01.029.

[12] R. G. Barrientos-Mauricio, A. Q. Medrano-Jacobo, and E. A. Carrera-Salas, “IoT-based air pollution monitoring system: A case study in Artisanal Brick Kilns,” *Sustain. Environ.*, vol. 12, no. 1, Dec. 2026, doi: 10.1080/27658511.2025.2607841.

[13] G. Aprilia, H. Fakhrurroja, and D. Pramesti, “IoT-Base Air Pollution Monitoring System in Urban Area,” in *2025 International Conference on ICT for Smart Society (ICISS)*, IEEE, Sep. 2025, pp. 1–6. doi: 10.1109/ICISS66954.2025.11389321.

[14] M. N. A. Ramadan, M. A. H. Ali, S. Y. Khoo, M. Alkhedher, and M. Alherbawi, “Real-time IoT-powered AI system for monitoring and forecasting of air pollution in industrial environment,” *Ecotoxicol. Environ. Saf.*, vol. 283, p. 116856, Sep. 2024, doi: 10.1016/j.ecoenv.2024.116856.

[15] L. Tang, M. Jia, J. Yang, L. Li, X. Bo, and Z. Mi, “Chinese industrial air pollution emissions based on the continuous emission monitoring systems network,” *Sci. Data*, vol. 10, no. 1, p. 153, Mar. 2023, doi: 10.1038/s41597-023-02054-w.

[16] S. Listyarini, L. Warlina, and A. Sambas, “The Air Quality Monitoring Tool Based on Internet of Things to Monitor Pollution Emissions Continuously,” *Environ. Ecol. Res.*, vol. 10, no. 6, pp. 824–829, Dec. 2022, doi: 10.13189/eer.2022.100616.

[17] S. M. S. D. Malleswari and T. K. Mohana, “Air pollution monitoring system using IoT devices: Review,” *Mater. Today Proc.*, vol. 51, pp. 1147–1150, 2022, doi: 10.1016/j.matpr.2021.07.114.

[18] Harish G N, Asharani R, and Nayana R, “IoT-based air pollution monitoring and data analytics using machine learning approach,” *World J. Adv. Res. Rev.*, vol. 12, no. 1, pp. 521–528, Oct. 2021, doi: 10.30574/wjarr.2021.12.1.0411.