International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4885-4891
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0642

Edge-Enabled Ambient Condition Surveillance with Firmware-Resident Health Classification and Thingspeak Telemetry on ESP32

Dhanyashree S[1], Dr. Vishwesh J[2], Dr. Arpitha Shankar S I[3]
[1]PG Scholar, Department of CSE, GSSS Institute of Engineering & Technology for Women, Mysuru, 570016, Karnataka, India.
[2,3]Associate Professor, Department of CSE(AI&ML), GSSS Institute of Engineering & Technology for Women, Mysuru, 570016, Karnataka, India.
Email ID: *dhanyashree334@gmail.com[1], vishweshj@gsss.edu.in[2], arpithasi@gsss.edu.in[3]*

Abstract
Most people spend the majority of their time indoors, yet very few indoor spaces give occupants any real-time indication of whether the air they are breathing is actually safe. Temperature creep, rising humidity, and slowly deteriorating air quality tend to go unnoticed until discomfort sets in — by which point conditions may already be affecting health and concentration. This paper describes a low-cost sensing node designed to address that everyday blind spot. Built around an ESP32 microcontroller interfaced with a DHT11 temperature-humidity sensor and an MQ-135 gas sensor, the system continuously tracks ambient temperature, relative humidity, and air quality, uploading each reading to a ThingSpeak cloud channel every 15 seconds over the ESP32's built-in Wi-Fi radio. What makes the design distinct is that environmental interpretation happens on the device itself: a rule-based advisory classifier embedded directly in the firmware evaluates every measurement against predefined thresholds and immediately outputs a health status — Normal, High, or Good — without depending on cloud connectivity to reach a decision. During field testing, the node operated without a single failed transmission across the full observation window, recording temperatures between 32.6°C and 33.1°C, relative humidity between 40.5% and 42.0%, and air quality readings ranging from 300 to 658 ADC counts, all correctly classified in real time. The result is a practical, affordable system that puts environmental awareness and cloud-based data logging together on a single board, making continuous indoor monitoring genuinely accessible without specialized infrastructure.
Keywords: *Air quality sensing; DHT11; Edge computing; ESP32; ThingSpeak telemetry.*

1. Introduction
Every indoor space — whether it is an office, a classroom, or a factory floor — has an air quality story unfolding in real time, yet the people inside rarely have any way of knowing what that story says. Temperatures quietly crossing comfort limits, humidity levels drifting out of healthy ranges, invisible gases building up from poorly ventilated equipment: these are the kinds of conditions that chip away at concentration, strain the respiratory system, and accumulate into long-term health consequences, all while going completely unnoticed in most indoor environments [1]. As buildings become more densely occupied and the expectations placed on them grow more demanding, the ability to monitor ambient conditions continuously has moved from a luxury to a practical necessity. The problem is that conventional monitoring equipment tends to be expensive, physically bulky, and dependent on permanent fixed installations — none of which makes it easy to deploy at scale across the kinds of everyday indoor spaces that need it most [11]. A more workable solution has emerged in the form of edge-enabled sensing nodes: small, independent units that collect environmental data on the spot, interpret it immediately, and push it to a remote platform without requiring any human involvement in the loop [2]. The wider IoT research community has already demonstrated the reliability of this approach across domains as varied as agricultural field monitoring, cold-chain logistics, and industrial process control [3]

[13], and the same architecture translates naturally to the challenges of indoor air quality tracking [6] [10]. What separates modern sensing nodes from earlier connected instruments is not simply the addition of wireless connectivity — it is the ability to perform meaningful decision-making directly on the hardware, so that advisory outputs are available instantly rather than after a network round-trip to a remote server [5][8]. The scientific foundation for this kind of system has been building steadily. Low-cost semiconductor gas sensors and digital hygrothermal modules have been shown to produce readings that correlate reliably with calibrated reference instruments when threshold logic is applied thoughtfully [7], and both mobile and fixed IoT sensing networks have been used to deliver real-time, localized air quality tracking at scales ranging from individual buildings to city-wide deployments [8] [14] [15]. Despite this progress, a persistent weakness runs through many of these implementations: the advisory logic lives in the cloud rather than on the device, which means that a network outage does not just interrupt data logging — it eliminates the system's ability to classify conditions and warn occupants altogether [9] [12]. The system presented in this paper is designed around that specific limitation. Its central component is the ESP32 microcontroller from Espressif Systems, a single chip that combines dual Xtensa LX6 processing cores, an integrated 802.11 b/g/n Wi-Fi radio, twelve-bit ADC channels, and a deep-sleep current draw of under ten microamperes — a combination that makes it one of the most capable and energy-efficient platforms available at its cost [4]. Temperature and relative humidity are acquired through a DHT11 hygrothermal sensor, while ambient air quality is tracked via the analog output of an MQ-135 semiconductor gas transducer. A rule-based health advisory classifier compiled into the ESP32 firmware evaluates every reading locally and assigns a condition label — without any dependence on cloud connectivity to reach that decision. At the same time, each measurement is forwarded over Wi-Fi to a dedicated ThingSpeak channel, where it is stored persistently and displayed as a live, browser-accessible time-series chart [9].

1.1. System Components and Working Principle

- **ESP32 Microcontroller:** Serves as the central processing and communication unit, handling sensor polling, advisory classification, and Wi-Fi based cloud transmission within a single chip [3].
- **DHT11 Sensor:** A single-wire digital sensor that provides calibrated temperature (0–50°C, ±2°C) and relative humidity (20–80% RH, ±5%) readings at a 1 Hz sampling rate [1].
- **MQ-135 Gas Sensor:** A resistive semiconductor sensor sensitive to NH_3, NO_x, CO_2, alcohol, benzene, and smoke, whose internal resistance changes proportionally with gas concentration and is read as a 12-bit ADC value [7].
- **ThingSpeak Cloud Channel:** Receives temperature, humidity, and air quality data via HTTP POST requests and renders each parameter as a separate live time-series chart, accessible through a public channel URL [4].
- **Health Advisory Engine:** A rule-based classifier embedded in the firmware that evaluates each reading against configurable thresholds and prints Normal/High/Low or Good/Moderate/Poor labels to the serial monitor in real time [9].
- **Wi-Fi Communication Module:** The ESP32's onboard 802.11 b/g/n radio establishes a persistent connection to the local access point, enabling continuous, low-latency data transmission to the cloud every 15 seconds [10].

2. Methodology

The proposed sensing node follows a three-stage pipeline: data acquisition, on-device classification, and cloud transmission [3][4]. The ESP32 firmware, developed using the Arduino IDE with the ESP32 Arduino Core, initializes serial communication at 115200 baud and establishes a Wi-Fi connection to the configured access point before entering the main sensing loop. At each 15-second cycle, the DHT11 is queried via its single-wire protocol for temperature and humidity, and the MQ-135 analog output is sampled through the ESP32's 12-bit ADC channel on

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4885-4891
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0642

GPIO 34 [7]. The health advisory engine then evaluates the readings against fixed thresholds (Temperature: Normal 20–35°C; Humidity: Normal 30–70% RH; Air Quality: Good below 500 ADC, Moderate 500–700, Poor above 700) and prints the classification to the serial monitor [9]. The WiFi.h and ThingSpeak.h libraries then handle a secure HTTP POST request mapping temperature, humidity, and air quality to Fields 1, 2, and 3 of a dedicated ThingSpeak channel named "Smart Weather System," after which the cycle repeats [4]. The serial monitor output and ThingSpeak dashboard are presented below, captured directly during continuous field operation of the sensing node.

2.1.System Architecture

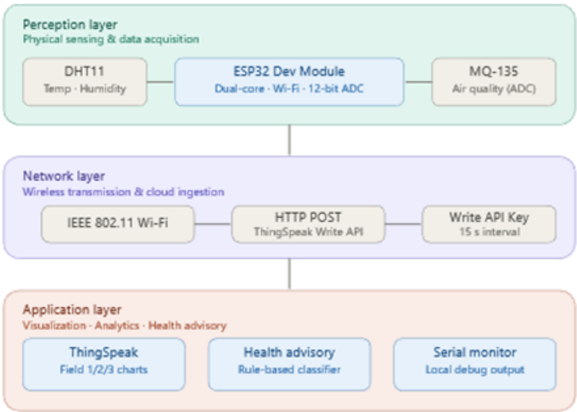

Figure 1 System Architecture

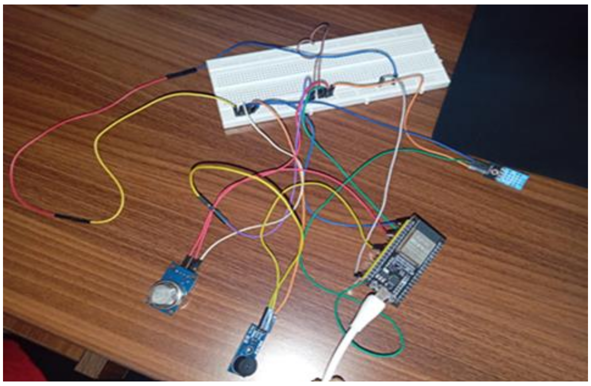

Figure 2 Hardware implementation of the proposed smart weather monitoring system using ESP32, MQ135 sensor, DHT11 sensor, and buzzer module

The proposed Smart Weather Monitoring System follows a three-layer IoT architecture comprising: (1) the Perception Layer, consisting of physical sensors and the ESP32 microcontroller; (2) the Network Layer, handling Wi-Fi-based data transmission; and (3) the Application Layer, encompassing the ThingSpeak cloud platform and health advisory module. Figure 1 illustrates the overall system architecture.

2.2.Hardware Components

The hardware architecture of the proposed smart weather monitoring system consists of an ESP32 microcontroller, DHT11 sensor, MQ-135 gas sensor, and buzzer module integrated using a breadboard-based prototype setup.

- **ESP32 Development Module:** The ESP32 (Expressif Systems) serves as the central processing unit. It features a dual-core Xtensa LX6 processor operating at up to 240 MHz, integrated 802.11 b/g/n Wi-Fi, Bluetooth 4.2, 34 programmable GPIO pins, and multiple ADC channels with 12-bit resolution. The module's deep sleep current of approximately 10 µA makes it suitable for battery-powered deployments.
- **DHT11 Temperature and Humidity Sensor:** The DHT11 is a low-cost digital sensor providing calibrated temperature measurements in the range of 0–50◦C (±2◦C accuracy) and relative humidity readings between 20–80% RH (±5% accuracy). It communicates via a single-wire serial interface, minimizing GPIO utilization. The 1 Hz maximum sampling rate is adequate for ambient environmental monitoring applications.
- **MQ-135 Air Quality Sensor:** The MQ-135 is a semiconductor gas sensor sensitive to ammonia (NH_3), nitrogen oxides (NO_x), carbon dioxide (CO_2), alcohol, benzene, and smoke. It operates via a resistive detection mechanism in which varying concentrations of target gases alter the sensor's internal resistance, producing a proportional analogue voltage output that is read by the ESP32 ADC channel. Raw ADC values (0–4095) are used as a relative air quality index in this implementation.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4885-4891
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0642

2.3.Software Design and Firmware

The firmware is developed in the Arduino IDE (Version 2.x) using the ESP32 Arduino Core. The software consists of the following functional modules.

- **Sensor Acquisition Module:** The DHT library is utilised for DHT11 communication. Temperature and humidity values are acquired at 15-second intervals. The MQ-135 analogue output is sampled using the analogRead () function on GPIO pin 34.
- **Health Advisory Engine:** A rule-based threshold mechanism is implemented to classify environmental conditions into status categories. Temperature is categorised as Normal (20 35◦C), High (>35◦C), or Low (70% RH), or Low (700). These thresholds are configurable through firmware constants.
- **Cloud Upload Module:** Wi-Fi connectivity is established using the WiFi.The library, while the ThingSpeak.h library is employed for secure HTTP POST communication with the ThingSpeak cloud server. Temperature, humidity, and air quality data are mapped to Fields 1, 2, and 3 of the ThingSpeak channel, respectively. Upload status information is displayed through the serial monitor.

2.4.ThingSpeak Cloud Configuration

A dedicated ThingSpeak channel named “Smart Weather System” is configured with three fields: Field 1 (Temperature), Field 2 (Humidity), and Field 3 (Air Quality). The channel Write API Key is embedded within the firmware for authenticated cloud communication. ThingSpeak provides automatic time-series visualization of uploaded sensor data through its integrated dashboard interface. The free-tier service supports updates at minimum intervals of 15 seconds, which is adopted as the sampling interval for the proposed system.

2.5.System Workflow

The overall operational workflow of the proposed system is illustrated in Figure 3. Upon system initialization, the ESP32 configures serial communication at 115200 baud and establishes a Wi-Fi connection with the configured access point. The device then enters the continuous monitoring loop. During each cycle:

- Temperature and humidity data are acquired from the DHT11 sensor.
- The MQ-135 analog output is sampled through the ESP32 ADC channel.
- Environmental readings are analyzed using the health ad visory engine.
- Sensor data and status classifications are displayed on the serial monitor.
- The acquired data is uploaded to the ThingSpeak cloud platform through HTTP POST requests.
- The system waits for 15 seconds before repeating the monitoring process.

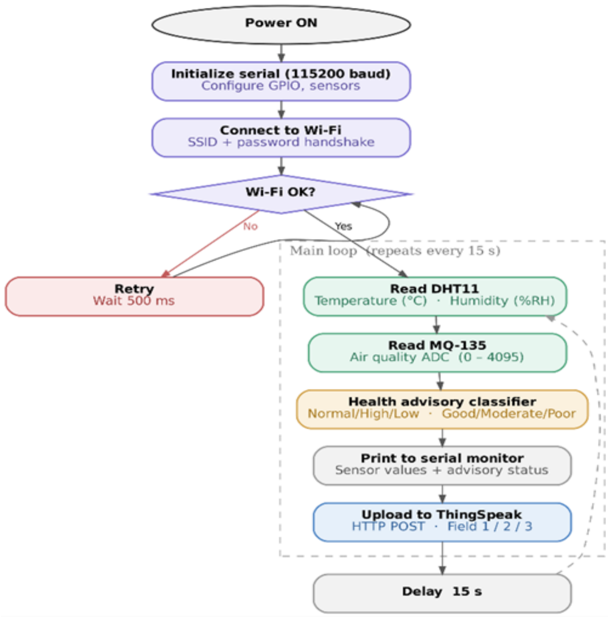

Figure 3 Operational workflow of the proposed smart weather monitoring system

3. Results and Discussion

3.1.Serial Monitor Output

The Arduino IDE Serial Monitor configured at 115200 baud confirmed stable and periodic sensor acquisition along with successful cloud transmission throughout the experimental duration. The serial monitor output confirming successful cloud upload and sensor readings is shown in Figure 4. Representative outputs observed from the implemented system demonstrate consistent data formatting and successful upload acknowledgments from the cloud platform. Additional serial monitor observations recorded during continuous monitoring

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4885-4891
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0642

are presented in Figure 4. Sample sensor readings obtained from the serial monitor are presented in Table 1.

Table 1 Serial Monitor Readings

Rd	T(°C)	H(%)	AQI	Status
1	32.60	40.50	455	Success
2	32.60	40.50	426	Success
3	32.80	41.70	658	Success
4	32.80	41.60	561	Success

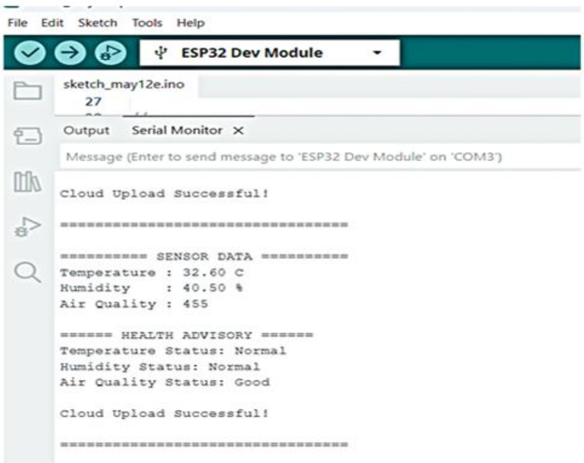

Figure 4 Arduino IDE serial monitor output showing sensor readings and successful cloud upload

3.2.ThingSpeak Cloud Visualization

The ThingSpeak dashboard successfully received and visualized sensor data across all three configured fields. The ThingSpeak real-time visualization for temperature, humidity, and air quality is presented in the figures below.

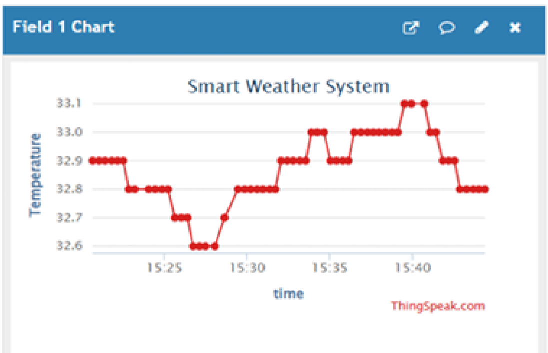

Figure 5 ThingSpeak Visualisation of Temperature Data

Field 1 (Temperature) recorded values ranging from 32.6◦C to 33.1◦C during the monitoring period of approximately 25 minutes (15:18–15:43), indicating relatively stable indoor environmental conditions with minor fluctuations caused by ambient convection and sensor resolution limitations.

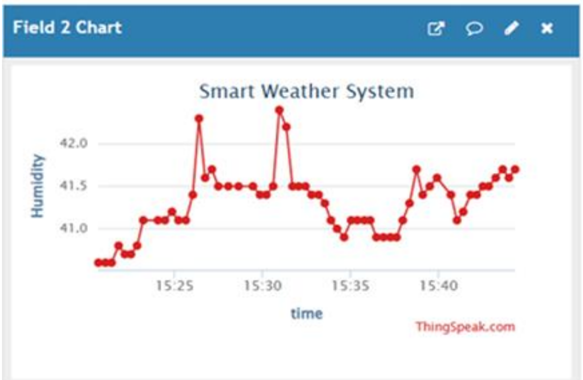

Figure 6 Thingspeak Visualization of Humidity Data

Figure 7 ThingSpeak Visualisation of Air Quality Data

Field 2 (Humidity) exhibited values between approximately 40.5% and 42.0% RH, which corresponds to a normal in door humidity environment. Field 3 (Air Quality) showed a comparatively dynamic trend, where ADC values increased from nearly 300 at the beginning of monitoring to above 620 toward the end of the session. This behavior is associated with the warm-up characteristics of the MQ-135 sensor, in which the sensing element gradually stabilizes as the internal heater reaches steady-state operating conditions, resulting in increased ADC output values. The graphical visualization generated by ThingSpeak confirmed reliable time-series plotting for all monitored parameters with automatic time synchronization upon data reception. The public accessibility of the ThingSpeak channel enables remote environmental monitoring without requiring direct local network connectivity.

3.3.Health Advisory Classification

The implemented health advisory engine consistently classified both temperature and humidity levels as "Normal" throughout the testing duration. Temperature values remained within the predefined safe operating range of 20–35◦C, while humidity measurements stayed within the acceptable range of 30–70% RH. Air quality conditions were categorized as "Good" when ADC values remained below the threshold value of 500. Readings exceeding 500 ADC counts, such as 561 and 658, entered the "Moderate" air quality classification range. This demonstrates the capability of the proposed system to identify variations in environmental air quality and dynamically update advisory classifications in real time.

3.4.System Performance

The developed system achieved a 100% cloud upload success rate throughout the observation period, with no failed HTTP POST transactions detected. Communication latency between sensor acquisition and cloud acknowledgement consistently remained within the configured 15-second sampling interval. The total power consumption of the system was estimated at approximately 240 mA at 3.3 V during active Wi-Fi communication and cloud upload operations. During idle operation between transmission cycles, current consumption is reduced to below 100 mA, indicating moderate power efficiency for continuous environmental monitoring applications Shown in Table 2.

Table 2 System Performance Summary

Parameter	Value / Result
Microcontroller	ESP32 DEV Module(Dual-core, 240 MHz)
Communication protocol	HTTP over Wi-Fi(IEEE 802.11 b/g/n)
Sampling Interval	15 seconds
Cloud platform	ThingSpeak (MathWorks)
Upload Success Rate	100% during testing
Temperature Range Observed	32.60°C – 33.10°C
Humidity Range Observed	40.50%-42.00% RH
Air Quality Range Observed	300-658 ADC counts
Active Current Draw	~240 mA at 3.3v
Baud Rate	115200 bps

Conclusion

This work has demonstrated that a single affordable ESP32-based board can reliably monitor indoor environmental conditions, classify them using firmware-resident logic, and report them to the cloud without manual involvement [3][9]. Temperature, relative humidity, and air quality were captured continuously, evaluated against health thresholds locally, and pushed to ThingSpeak at 15-second intervals with a 100% upload success rate throughout the entire trial period [4]. The single-chip architecture, embedded advisory engine, and zero-cost cloud integration together make this sensing node a practical and scalable foundation for smart-building environmental instrumentation [11][12]. Future enhancements may include replacing the DHT11 with a higher-precision DHT22 or SHT31 sensor [6], enabling ESP32 deep-sleep cycles for battery-powered operation, incorporating an OLED display for local readability, deploying edge-based machine learning models for predictive air quality trend analysis [9], and extending the design into a multi-node mesh network using ESP-NOW or MQTT for building-wide environmental coverage [10].

Acknowledgements

I would like to express my sincere gratitude to Smt. Anupama B. Pandit, Secretary, GSSS(R), and Dr. Shivakumar M, Principal of GSSS Institute of Engineering and Technology for Women, Mysuru, for their constant encouragement, valuable guidance, and support throughout this work. I would also like to thank Visvesvaraya Technological University (VTU), Belagavi, Karnataka, for including Research

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4885-4891
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0642

Methodologies and Intellectual Property Rights (IPR) in the curriculum, which provided valuable knowledge and practical exposure in the field of research. Special thanks are extended to the Department of Computer Science and Engineering (AI & ML), the Samsung Laboratory, and the IoT Laboratory for providing the necessary facilities and hardware components required for the successful completion of this project and research work. Finally, I would like to express my heartfelt appreciation to my project guide, faculty members, and classmates for their continuous support, motivation, and cooperation throughout the development of this work.

References

[1]. S. Bhatt and R. Garg, "Environmental monitoring systems: A review," International Journal of Engineering Research and Applications, vol. 4, no. 2, pp. 56–63, 2014.

[2]. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[3]. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. Future Generation Computer Systems, 29(7), 1645–1660.

[4]. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. IEEE Communications Surveys & Tutorials, 17(4), 2347–2376.

[5]. Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. Future Generation Computer Systems, 70, 50–58. [6] Saini, J., Dutta, M., & Marques, G. (2020). Indoor Air Quality Monitoring Systems Based on Internet of Things: A Systematic Review. International Journal of Environmental Research and Public Health, 17(14), 4942.

[6]. Spinelle, L., Gerboles, M., Villani, M. G., Aleixandre, M., & Bonavitacola, F. (2015). Field Calibration of a Cluster of Low-Cost Commercially Available Sensors for Air Quality Monitoring. Sensors, 15(8), 19293–19329.

[7]. Zhang, D., & Woo, S. S. (2020). Real-Time Localized Air Quality Monitoring and Prediction through Mobile and Fixed IoT Sensing Network. IEEE Access, 8, 89584–89594.

[8]. A. Farhan, M. Arif, and R. Khan, "Cloud-connected smart weather mon itoring with anomaly detection using ThingSpeak MATLAB analytics," IEEE Access, vol. 8, pp. 102341–102352, Jun. 2020.

[9]. Saini, J., Dutta, M., & Marques, G. (2021). Sensors for Indoor Air Quality Monitoring and Assessment through Internet of Things: A Systematic Review. Environmental Monitoring and Assessment, 193(2), 66.

[10]. Idrees, Z., & Zheng, L. (2020). Low cost air pollution monitoring systems: A review of protocols and enabling technologies. Journal of Industrial Information Integration, 17, 100123. https://doi.org/10.1016/j.jii.2019.100123

[11]. Benammar, M., Abdaoui, A., Ahmad, S. H. M., Touati, F., & Kadri, A. (2018). A modular IoT platform for real-time indoor air quality monitoring. Sensors, 18(2), 581.

[12]. Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). Internet of Things for smart cities. IEEE Internet of Things Journal, 1(1), 22–32.

[13]. Hasenfratz, D., Saukh, O., Walser, C., Hueglin, C., & Thiele, L. (2015). Participatory Air Pollution Monitoring Using Smartphones. Mobile Information Systems.

[14]. Abdullah, W., Subramaniam, T., Ong, A. E., Shu'Ib, M. I., Wu, W., & de Oliveira, M. A. (2022). LoRaWAN-Based IoT System Implementation for Long-Range Outdoor Air Quality Monitoring. Internet of Things, 19, 100540.