iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: V Month of publication: May 2025

DOI: https://doi.org/10.22214/ijraset.2025.70007

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Air Quality Monitoring System Using FOG Computing

Dr. A. Senthilselvi[1], Rajdeep Saha[2], Anushka Rai[3], Animesh Kumar Choudhary[4], Manya Gupta[5]
Department of Computer Science and Engineering SRM Institute of Science and Technology Chennai, India

Abstract: The focus of this project is real time Air Quality Monitoring (AQI) within indoor environment using a hybrid Edge–Fog–Cloud architecture. The edge node is an ESP32 microcontroller and PMS5003 particulate matter sensor pair that collects, and preprocesses data for air quality, including PM1.0, PM2.5 and PM10 concentration. Data collected from the edge node is sent to a local Fog node that consists of an InfluxDB instance running in Docker container which gives the advantage of low latency data storage, processing and real-time alerts generation. In order to scale and ensure long term analytics, the fog node writes the data to an InfluxDB Cloud instance to store the data long time and make it remote available. Dashboard with InluxDB's native visualization tools helps to get current and historical air quality trends. The Fog layer monitors continuously and remains resilient in the case of network outages while the Cloud layer is for large scale data analysis and report generation. The system responsiveness is optimized with this hybrid architecture, so as to reduce the reliance on external networks to carry out critical operations and to achieve a scalable solution for indoor air quality management. In future, multi room expansion, more advanced alerting mechanisms and predictions of air quality deterioration are also expected.
Keywords: Air Quality Monitoring, IoT-based Environmental Sensing, Edge–Fog–Cloud Architecture, Real-Time Data Processing

I. INTRODUCTION

The need to maintain healthy indoor air quality has become essential as the environment is more urbanized and more time is spent indoors. PM1.0, PM2.5 and PM10 can have great influence on respiratory health, particularly for vulnerable populations. The traditional air quality monitors are based on a cloud-based architecture that is powered by air quality information, which incurs latency and the reliance on constant internet connectivity. In order to address these, the system proposed in this project utilizes Edge, Fog and Cloud computing to deliver real time, reliable air quality monitoring and alert.

An ESP32 microcontroller with PMS5003 sensor is used as a direct environmental sensor at the edge layer. Particulate data is collected by the ESP32, some processing occurs (e.g. averaging, smoothing), and the results are transmitted wirelessly to the Fog layer. The data is captured and acted upon promptly from this design, which sets a foundation for the ability to make immediate local decisions, with alarms rang when air quality worse than allowable figures.

First of all, the Fog layer is local system running InfluxDB instance inside a Docker container. This layer acts as a temporary storage of incoming data and allows those data to be queried or displayed on a dashboard using native InfluxDB tools. Local processing and storage of data by the Fog layer reduces reliance on network and enhances resilience in case of internet failures as well as yielding low latency responses for deteriorating air quality in the environment monitored.

The system has a Cloud layer based on InfluxDB Cloud for long-term storage, analytics, and remote monitoring. While periodical replication of Data from the Fog node to the Cloud facilitates for historical analysis and visualization of trends in air quality, the model also considers capability of Fog nodes to both compute and aggregate air quality observations (like those generated by IoT devices) along with GPS position values. The Edge–Fog–Cloud architecture is an intelligent indoor air quality management architecture that provides a combined real time responsiveness, reliability, and scalability for the hybrid architecture.

II. LITERATURE SURVEY

A study presented in [1] introduces a heterogeneous Edge–Fog–Cloud architecture designed to optimize task distribution and lower response time in IoT applications. While effective for general-purpose edge computing, the system does not specifically address real-time environmental sensing or air quality data handling.

An extended work in [2] adds to this by implementing an Edge–Fog–Cloud architecture specifically for aircraft component monitoring, demonstrating improved performance and fault detection. However, the work focuses on industrial assets and lacks emphasis on small-scale, indoor environmental sensing relevant to residential or healthcare environments.

In [3], a real-time air quality monitoring system is proposed for smart cities using IoT-enabled optical sensors. Though promising in scalability and integration, the study is largely oriented toward outdoor urban environments, offering limited insights into the challenges of indoor air monitoring in closed, controlled settings.

A broader review presented in [4] surveys recent innovations in air quality monitoring, emphasizing new sensor types, IoT protocols, and future research directions. Despite valuable technological mapping, it lacks a practical deployment framework combining edge analytics and cloud storage, which is crucial for real-time decision-making in indoor scenarios.

An integrated solution is suggested in [5], where a low-cost IoT architecture is developed for indoor air quality monitoring. Using lightweight devices and local data storage, the system shows promise for home deployment. However, it primarily emphasizes low-cost implementation without addressing fault tolerance, data synchronization across layers, or resilience to network failures.

In [6], an optimized Fog-enabled air quality monitoring and prediction system is proposed using lightweight deep learning models. This study effectively demonstrates edge-based prediction capabilities but remains largely simulation-based, offering limited real-world validation under varying environmental conditions.

Reference [7] presents a practical implementation of an IoT-based air quality monitoring system during sleep periods. The study highlights human exposure risks and captures long-duration indoor data; however, it does not adopt a multi-layer architecture and remains dependent on single-point data capture without fog replication or backup.

Study [8] focuses on developing a unified IoT platform to assess meteorological and air quality data in tropical environments. While highly relevant for environmental sensing, the system primarily addresses outdoor monitoring and lacks mechanisms for real-time edge alerting in enclosed indoor spaces.

In [9], the design and implementation of ESP32-based IoT devices are discussed. This work demonstrates the feasibility of building compact, wireless sensing devices, yet focuses more on device-level engineering rather than integration with fog and cloud systems for hierarchical data management and real-time processing.

Reference [10] discusses how InfluxDB can be used to handle IoT data streams, offering insights into time-series storage, visualization, and alerting capabilities. Although foundational, the work primarily addresses cloud-centric data storage and does not explore hybrid architectures that combine local (fog) and cloud layers for optimized reliability and latency control.

Collectively, these studies showcase the growing trend toward distributed, intelligent environmental sensing systems. While significant strides have been made in sensor integration, cloud storage, and IoT frameworks, few existing solutions simultaneously balance real-time responsiveness, fog resilience, and scalable cloud analytics. This reinforces the need for a hybrid Edge–Fog–Cloud system, like the one proposed, which ensures robust real-time monitoring, local processing, and historical data storage to create an adaptive, efficient air quality management solution.

III. PROPOSED METHODOLOGY

In order to monitor indoor air quality in real time AND also provide scalable historical data storage, the proposed system utilizes a Hybrid Edge - Fog - Cloud architecture. An ESP32 microcontroller at the Edge layer is used to interface a PMS5003 particulate matter sensor to record PM1.0, PM2.5 and PM10. Preliminary data processing, such as sampling, averaging, and AQI calculation by standard breakpoints is performed on the ESP32. Lightweight reliable protocols are used for transmitting processed data over Wi-Fi to the Fog node with low latencies and minimal packet loss.

The local server running an InfluxDB instance in a Docker container is part of the Fog layer. This layer is a fast pass filter and repository working as a temporary ‘pipeline’ for incoming data for visualization, real-time alert generation and rapid access. AQI values monitor the dynamics and if they reach the defined range, it sends notifications. Uninterrupted operation during short term cloud disconnections is ensured by the Fog node for alert triggering capability, even with lack of internet connection, keeping system up and operational.

Data sampled at the Cloud layer is replicated on periodic basis from the Fog node to InfluxDB Cloud to participate in storing long term, performing historical analysis. By enabling broader trend assessment, multi-site comparisons and remote accessibility to stakeholders, this promotes broader market assessment and comparison of the sites. The system achieves both real time decision making and big data analytics using combined localized processing with cloud storage. Through this methodology, an efficient air quality monitoring framework is developed with a resilient and low-latency performance for performing various kinds of monitoring while providing both immediate and historical insights on indoor environmental health.

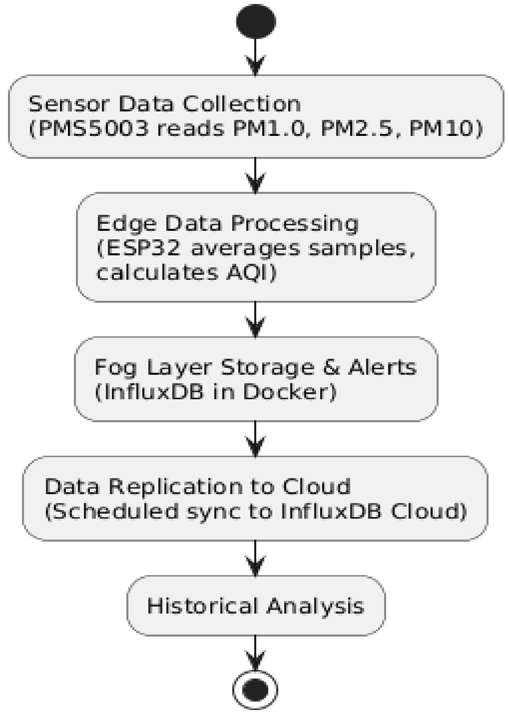


Fig 3.1 Methodology Flow Diagram

A. Sensor Data Collection

The primary data acquisition device for capturing particulate matter concentrations in indoor environment is a PMS5003 sensor. It uses a laser scattering method for reading PM1.0, PM2.5 and PM10 levels, but is high sensitive and it can provide full time reading. The ESP 32 microcontroller takes the data from the sensor outputs via a UART serial interface. By sampling continuously, air quality is continuously smoothed out and thus represents it in a stable and accurate way. The raw sensor data from this raw data is behind subsequent processing and calculation of the AQI in order to detect air pollution quickly and in real-time in the environment.

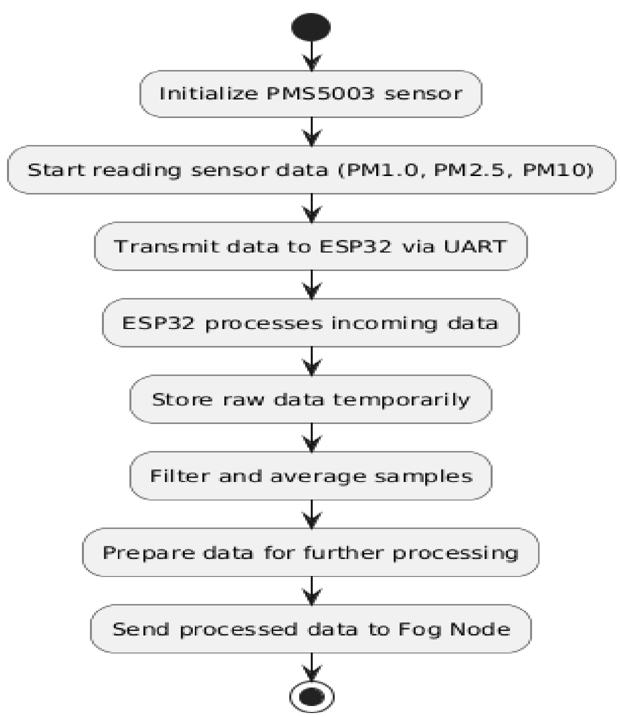


Fig 3.1.1 Sensor Data Collection Flow Chart

The ESP32 is interfaced electrically to each sensor and data is read at regular time intervals to make sure the data is being acquired in a consistent manner. UART, analogue input protocols are used to interface the microcontroller with the sensors. Timestamped, the raw data is put in memory until processed. The aim of this module is to be efficient, so as to have minimal power consumption and its latency to be very low, in order to keep the real-time tracking of health uninterrupted.

B. Edge Processing and Filtering

The ESP32 microcontroller is highly important as it is the one that processes the raw data that it receives from the PMS5003 sensor at the edge layer. Thus, the ESP32 takes particulate concentration values (PM1.0, PM2.5, and PM10), and filters it based on transient anomalies or noise. The readings are collected many times over very short intervals to allow statistical averaging in order to get more stable and accurate values. It avoids false alarms and guarantees the monitoring of only meaningful environmental variations that contributes to the reliability of real-time monitoring.

The ESP32 goes beyond filtering, the air quality index (AQI) is calculated using standard breakpoint tables that are based on PM2.5 and PM10 levels. On device computation of the AQI ensures that the data being sent to the fog layer do not comprise just raw measurement values, but the meaningful insight for the purpose of decision making. Through edge processing and filtering directly on the edge device, the bandwidth of the network is saved, the latency is reduced, and real time alerting is possible without the continuous cloud communication. The localized intelligence at the edge improves resilience and responsiveness for system control in indoor air quality management.

C. Edge Node and Fog Node

The PMS5003 sensor is affixed to an ESP32 microcontroller, the first point of environmental data collection and first point of cheque of these data — acting as an Edge Node within the system. Apart from reading PM1.0, PM2.5, PM10 concentrations, the ESP32 processes the data by filtering and performing AQI calculation algorithms locally. This makes sure that the transmitted data is small, understandable, and prepared to be acted upon if needed. The processed AQI and particulate readings are transmitted via Wi-Fi by the Edge Node to the next layer, reducing the raw data overhead by transmission and increasing the response time.

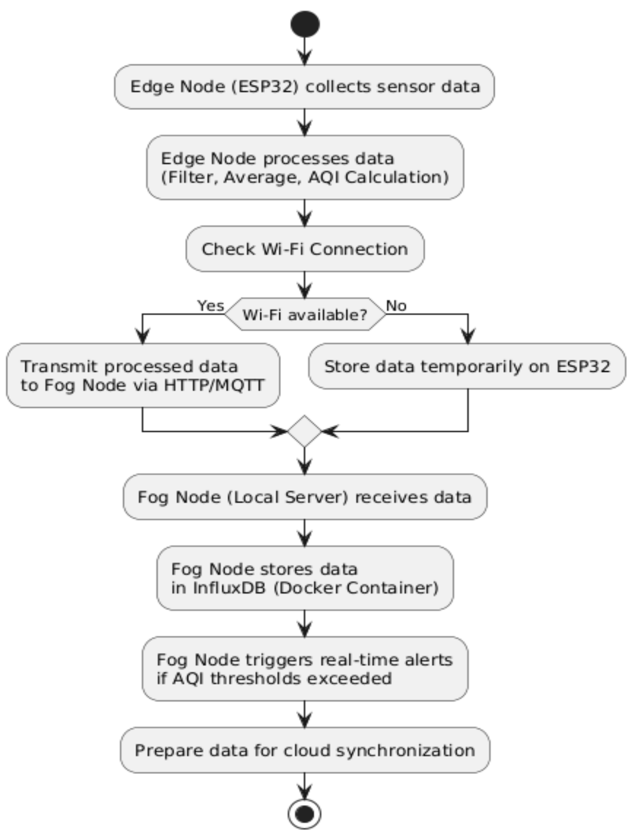


Fig 3.2.1 Local and Cloud Synchronization

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

Fog Node is implemented inside Docker container, where it runs within the local server which has an InfluxDB instance there. It acts like an edged server, a mid-layer between edge and cloud, stores, provides real time monitoring, and alerts generation. Once data from an edge device is received, it stores the data in a time series database to enable visualize it immediately and send an alert if the data violates a threshold. Even when the internet connection is not available, real time decision making is possible locally at the Fog Node. This allows us to assure the system's resilience and meanwhile keep monitoring the indoor air quality continuously, also data preparation for a furtive scheduling with cloud for long term data storage and deeper trend analysis.

D. Latency Analysis

A hybrid Edge–Fog–Cloud architecture is proposed to minimize data delivery time, and yet provide real time responsiveness as well as the long-term data reliability. This system has latency that can be decomposed into three key layers: Edge Processing Latency, Fog Communication and Storage Latency, and Cloud Synchronization Latency.

Typically, the Edge Processing Latency is below 50 ms. These tasks of filtering, averaging, and AQI calculation are light enough that ESP32 can do it locally. Since the sampled data packet size is small and no complex computation or encryption is needed, the micro controller can easily process and queue the data for transmission almost immediately after sampling.

Fog Node Latency is introduced as the time of Wi-Fi transmission and the time taken for local storage to the InfluxDB database. In an ideal local network environment, latency is less than 50ms, which makes it possible to visualize and alert without waiting for the video delivery. Absolutely critical for triggering timely alerts as AQI values cross thresholds, particularly in indoor environments where air quality can degrade quickly, the low delay aspect of the PIC agent is very attractive.

Cloud Synchronisation Latency is high as it is subject to the variability of internet and periodic batching. Scheduling the data replication to InfluxDB Cloud is usually carried out in intervals (roughly every 5–10 minutes) which results in the reported latency in the range of few seconds to minutes. But this delay is acceptable since the cloud layer is meant for historical analytics rather than real time deciding.

E. Architecture Overview

A three tier hybrid architecture i.e. Edge, Fog and Cloud is proposed for building the proposed system. The whole, therefore, operates as a layered approach to optimally serve real time air quality monitoring while preserving scalability to long term data storage and analysis. Entering data, processing data, alerting data and visualising data are also done at each layer.

The ESP32 microcontroller is interfaced with the sensor PMS5003 at the Edge layer to acquire air quality data including concentration of PM1.0, PM2.5 and PM10. In a local processing of this data, the microcontroller takes samples and averages them before computing the Air Quality Index based on EPA standards. It reduces transmission load and pass only interesting, actionable data to layers on top.

In this layer, there is a local server with InfluxDB running inside a Docker container (the Fog layer). It serves as a layer that sits between the Edge and the Cloud, thus serving as a buffer and data ingestion of low latency results, short lived storage and real time alert generation upon AQI threshold breach. Users can use the InfluxDB dashboard to see and react quickly to hazardous indoor air levels without the reliance of internet connectivity.

The historical air quality data is stored in the last Cloud layer using the InfluxDB Cloud which is powered by the Fog node to be replicated periodically. Such is the power of statistics that reports one can-do long-term reporting and detect trends. Robust performance is ensured on the architecture under various network conditions at hand, with immediate local insight, paired with scalable cloud-based analysis.

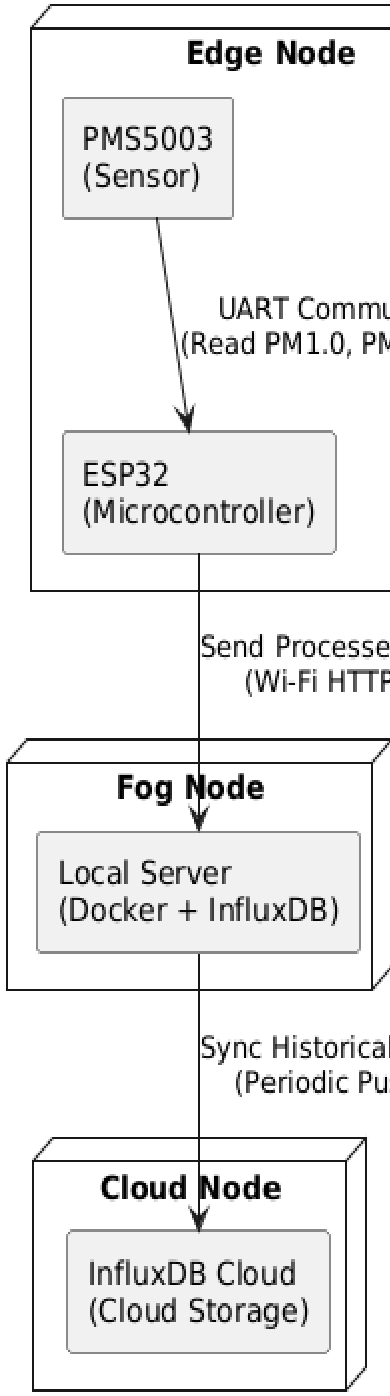


Fig 3.5.1 Architecture Diagram

F. Hardware Overview and Data flow

The hardware architecture comprises of several interrelated devices for the purpose of collecting, processing, and transmitting air quality data. The PMS5003 sensor is connected to the ESP32 microcontroller at the core of the edge layer and is used to sample particulate matter concentrations. The PMS5003 sensor is a PM1.0, PM2.5 and PM10 sensor with laser scattering technology. It communicates with the PMS5003 as UART and this data is filtered, averaged and AQI calculated in that ESP32.

The hybrid approach ensures data durability, therefore in case of a failure, the replication process just halts and it continues after recovery.

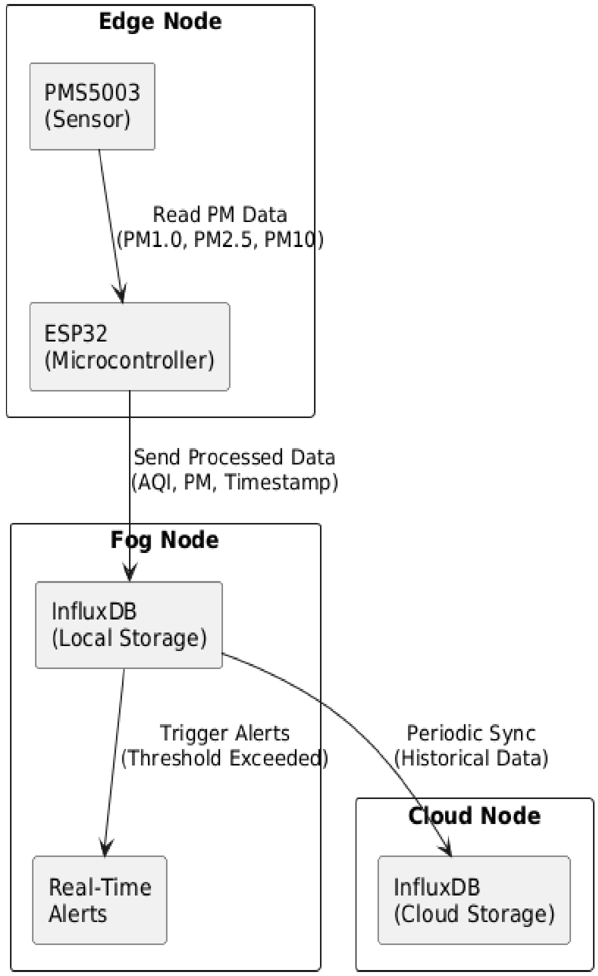


Fig 3.6.1 Hardware Overview and Data flow

As soon as the data has been processed on the locally placed ESP32, it will be sent to the Fog Node, which is a local server containing an InfluxDB inside a Docker container, and it will deliver the Air Quality Index (AQI) and the particulate data. This fog node serves as a buffer which allows for real time data storage and generation of alerts when AQI value crosses defined thresholds. Secondly, the fog layer offers fast local insights, for example, showing real time AQI value, or even trigger instantaneous warning for air quality that is harmful. In addition, the Fog Node also keeps periodically syncing the processed data to the Cloud Node, which also stores the historical data for more advanced analysis and trend detection into the InfluxDB Cloud.
At the sensor level, the data flow starts with the air quality data coming from PMS5003, which is passed to the ESP32 for local processing. Process the data and sends it to Fog Node to hold in real time for alerts, long term storage and periodic synchronization with the cloud. The hybrid architecture is able to provide real time response to environmental changes with scalability, security and privacy for storage of historical data.

IV. RESULTS AND DISCUSSION

The proposed hybrid architecture can be used to realize real time monitoring of AQI with low latency and high data reliability through combining edge, fog, and cloud layers. Air quality datais successfully collected and processed through ESP32 with PMS5003 sensor and then transmit them to Fog Node. AQI thresholds are set to trigger real time alerts which will prompt for immediate response when air quality is regarded as hazardous. Periodic synchronization is done between the Fog Node and the Cloud Node in order to store and analyze the data in the Cloud Node for a longer period of time. Such a setup, with the scalability, efficiency of real time as well as historical data handling and hence becoming an appealing option for smart environments.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

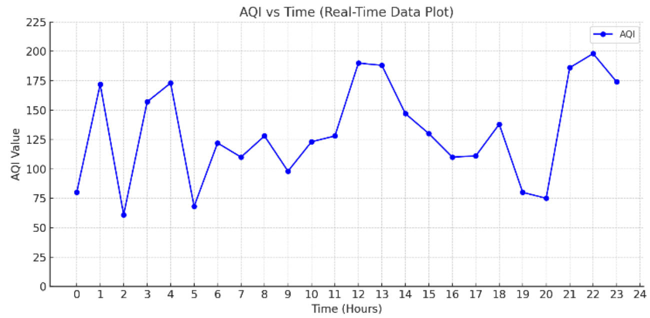


Fig 4.1 Trends in AQI vs Time

This is a graph showing variation of the Air Quality Index (AQI) during 24 hours.Between hours 1, 12–13, and 21–22 there are significant peaks indicating poorer air quality during those periods. Yet one can infer AQI values around hours 2, 5, and 20 to be comparatively cleaner compared to hours 1, 6, 14, and 21.

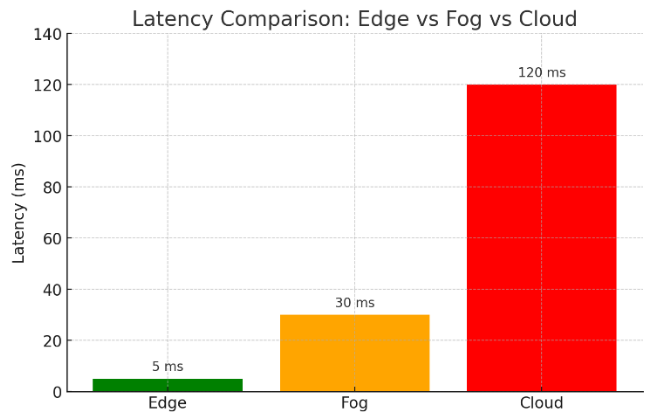


Fig 4.2 Latency Comparison Edge vs Fog vs Cloud

Latency difference figure given between Edge, Fog and Cloud computing models is shown in the graph. Low latency applications will be served up to 5 milliseconds, offering the lowest latency is edge computing which processes the data closest to source data. In comparison, Fog computing, which falls in between Edge and Cloud, has moderate latency, which is around 30 milliseconds, that offers the level of speed and processing power. The speed of cloud computing is 120 milliseconds due to the fact that the data needs to take a route to go to the data centers, therefore its latency is the highest.

Results indicate that a hybrid approach of edge–fog–cloud architecture for indoor AQI monitoring strikes a suitable trade off among ultra-low latency local responses and secure historical storage in the cloud at an acceptable expense. Real time sensing at the edge with the ESP32 and PMS5003 allows for generation of critical air quality alerts immediately to improve safety and response time. Local processing at the fog level with a Dockerized InfluxDB system allows for quick processing and several seconds of analytics, while exchanging storage at longer time scales with InfluxDB Cloud for perspective on longer term environmental trends. The setup is found reliable for real time air quality monitoring in smart homes and resource limited indoor space via comparative latency tests and system validations. In conclusion, this architecture presents a practical, scalable, and responsive approach to enhance indoor environment health management.

V. CONCLUSION AND FUTURE SCOPE

Indoor AQI monitoring with the proposed hybrid edge–fog–cloud architecture is successfully combined by combining the strengths of low latency local processing and scalable cloud storage. PMS5003, the ESP32 are used at the edge to capture the critical air quality parameter and process in real time to generate alerts and quick response for deteriorating air conditions. A fog layer provided by a Dockerized InfluxDB system optimizes local data handling and analytics; with periodic synchronization to the InfluxDB Cloud, securely stored old data is processed. The experimental results validate the reliability, responsiveness and the suitability to be deployed in smart homes, offices and resource constrained environment.

As part of future enhancements, machine learning models could actually be incorporated in the fog or cloud layers to leverage the capability of the models to perform predictive analytics, like predicting air quality trends and predicting ventilation actions. The system can be further expanded to support more environmental parameters such as temperature, humidity and CO_2 levels. Also, building of a user facing mobile or web dashboard with live visualization and alert features can change the usability as well as the system impact. Additionally, the architecture could be scaled into a multi room, building wide deployment with limited modifications.

Moreover, the system supports easy upgrade and customization for various domains of applications like industrial safety, school air monitor, public health surveillance. With the collaboration of IoT security frameworks further data privacy and protection can be achieved and it will also ensure compliance with emerging regulations. All in all, this work provides a solid base for developing intelligent, reliable and scalable indoor environmental monitoring which would sound and create healthier and working space.

REFERENCES

[1] Fernando, N., Shrestha, S., Loke, S. W., & Lee, K. (2025). On Edge-Fog-Cloud Collaboration and Reaping Its Benefits: A Heterogeneous Multi-Tier Edge Computing Architecture.

[2] Dogea, R., Yan, X.-T., & Millar, R. (2023). Implementation of an edge-fog-cloud computing IoT architecture in aircraft components. MRS Communications,

[3] Aserkar, A. A., Godla, S. R., El-Ebiary, Y. A. B., Krishnamoorthy, M., & Ramesh, J. V. N. (2024). Real-time Air Quality Monitoring in Smart Cities using IoT-enabled Advanced Optical Sensors. International Journal of Advanced Computer Science and Applications, 15(4)

[4] Shahid, S., Brown, D. J., Wright, P., Khasawneh, A. M., Taylor, B., & Kaiwartya, O. (2025). Innovations in Air Quality Monitoring: Sensors, IoT and Future Research. Sensors, 25(7), 2070.

[5] Othman, H., Azari, R., & Guimarães, T. (2024). Low-Cost IoT-based Indoor Air Quality Monitoring. Technology Architecture and Design, 8(2),

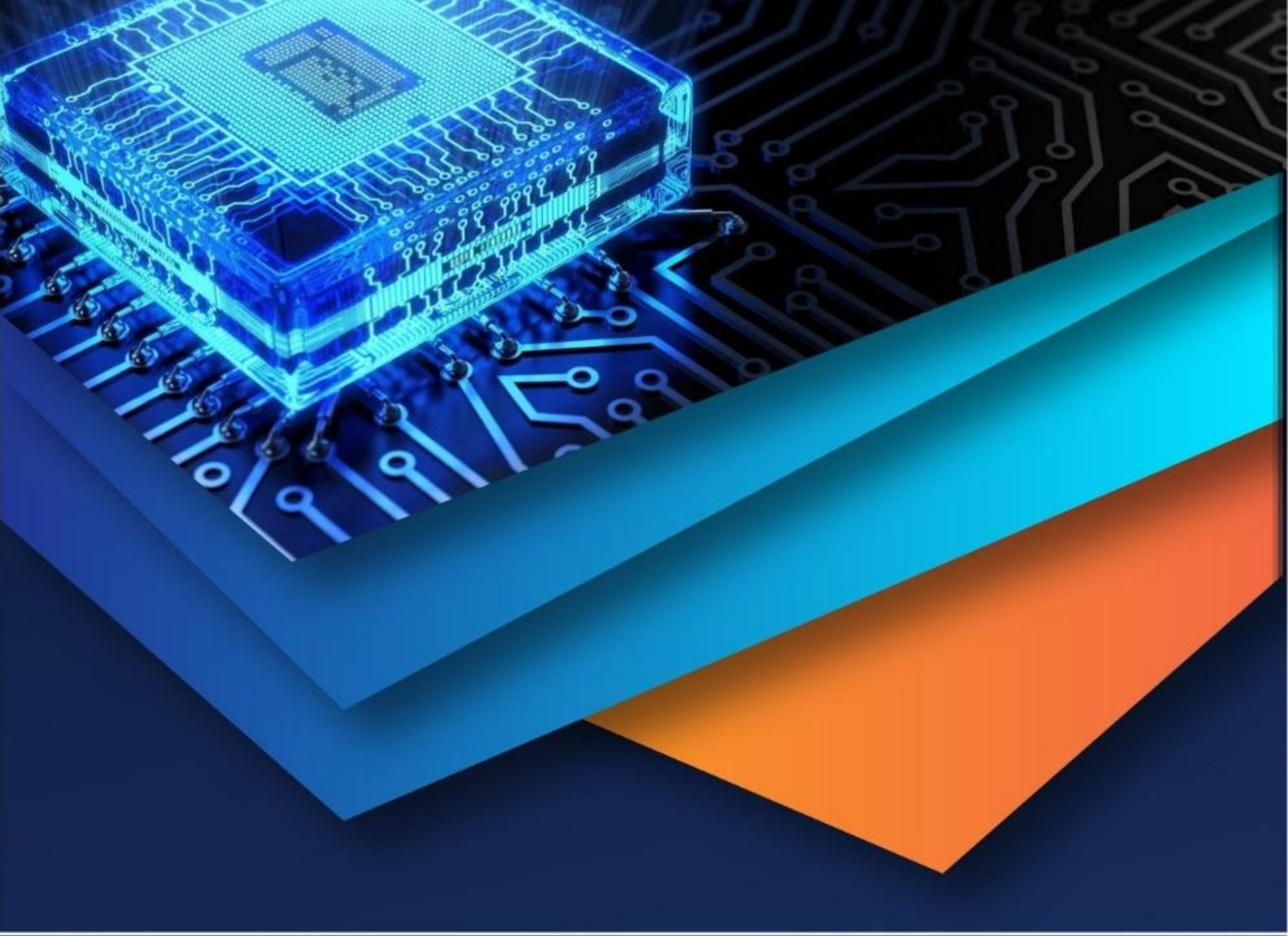
[6] Pazhanivel, D. B., Velu, A. N., & Palaniappan, B. S. (2024). Design and Enhancement of a Fog-Enabled Air Quality Monitoring and Prediction System: An Optimized Lightweight Deep Learning Model for a Smart Fog Environmental Gateway. Sensors, 24(15)

[7] Mota, A., Serôdio, C., Briga-Sá, A., & Valente, A. (2025). Implementation of an Internet of Things Architecture to Monitor Indoor Air Quality: A Case Study During Sleep Periods. Sensors, 25(6),

[8] Kairuz-Cabrera, D., Hernandez-Rodriguez, V., Schalm, O., Martinez, A., Merino Laso, P., & Guardia, A. (2024). Development of a Unified IoT Platform for Assessing Meteorological and Air Quality Data in a Tropical Environment. Sensors, 24(9),

[9] Hercog, D., Lerher, T., Truntič, M., & Težak, O. (2023). Design and Implementation of ESP32-Based IoT Devices. Sensors, 23(15),

[10] Sargent, A. (2021). How InfluxDB Works with IoT Data. InfluxData Blog.



doi crossref
10.22214/IJRASET

INDEX COPERNICUS
45.98

ISRA JIF
IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY