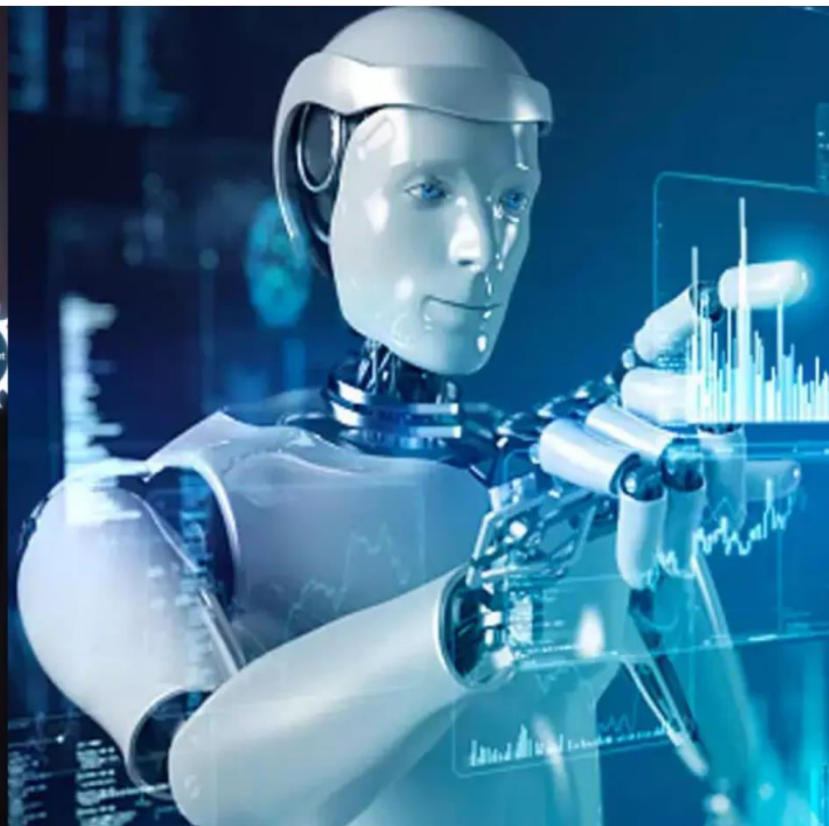




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 5, May 2026

Proactive Industrial Hazard Detection Using AI-Augmented IoT Sensor Networks and Large Language Model Reasoning

Noor Ayesha, Ganesh Kumar, Darshan B S, Gireesh A, Ajith H R

Assistant Professor, Department of Electronics and Communication Engineering, P.E.S. College of Engineering,
Mandya, Karnataka, India

U.G. Student, Department of Electronics and Communication Engineering, P.E.S. College of Engineering, Mandya,
Karnataka, India

U.G. Student, Department of Electronics and Communication Engineering, P.E.S. College of Engineering, Mandya,
Karnataka, India

U.G. Student, Department of Electronics and Communication Engineering, P.E.S. College of Engineering, Mandya,
Karnataka, India

U.G. Student, Department of Electronics and Communication Engineering, P.E.S. College of Engineering, Mandya,
Karnataka, India

ABSTRACT: Conventional industrial safety systems are fundamentally reactive, triggering alerts only after hazardous thresholds have already been breached. This paper proposes an AI-augmented IoT framework that shifts the safety paradigm from detection to anticipation. The system deploys NodeMCU V2 microcontroller nodes arranged in a self-healing mesh topology, continuously measuring CO, LPG, smoke, flame, temperature, and humidity across industrial zones. Sensor telemetry is securely transmitted to a HiveMQ MQTT broker over TLS and persisted in a TimescaleDB time-series database via a Go-based backend, accumulating over 10,000 records during evaluation. A Model Context Protocol (MCP) server bridges this stored telemetry to a large language model (LLM), enabling spatially-aware, contextual hazard reasoning beyond the capability of static rule engines. Dynamic edge-level thresholds provide near-zero-latency autonomous actuation independent of cloud connectivity, while the LLM layer performs root-cause analysis and issues structured safety assessments at the node level. Experimental validation across two monitored zones demonstrates reliable pipeline continuity, accurate hazard localization, and prompt actuator response, establishing a deployable and scalable architecture for proactive industrial safety management.

KEYWORDS: Industrial IoT, Hazard Monitoring, MQTT, ESP-Mesh Networking, TimescaleDB, Model Context Protocol, Large Language Model, Gas Detection, Real-Time Telemetry, Proactive Safety.

I. INTRODUCTION

The human and economic costs of industrial safety failures have been well documented across decades of catastrophic incidents. The 1984 Bhopal methyl isocyanate leak, which caused mass casualties and long-term environmental damage, exposed the deadly consequences of neglected monitoring infrastructure [1]. Similar risks remain present in cold-chain and refrigeration facilities, where undetected anhydrous ammonia leaks have repeatedly resulted in fatalities [2]. A common thread running through these disasters is the fundamental limitation of conventional safety systems: they respond only after dangerous conditions have already manifested, leaving no margin for prevention. Static threshold-based alarms, though widely deployed, cannot reason about the trajectory of environmental change or identify which zone or process is the source of a developing hazard. Their inability to contextualize sensor data means that by the time an alert fires, conditions may already be beyond safe intervention.

The Industrial Internet of Things (IIoT) has created the technical foundation for a more capable approach. Networked sensor arrays enable continuous, simultaneous monitoring of multiple environmental parameters across spatially distributed zones, supporting the kind of data-rich smart factory architectures described in recent literature [3]. Communication infrastructure has matured in parallel: MQTT-based brokers such as HiveMQ and EMQX now deliver reliable, low-latency telemetry across large multi-node deployments [4]. At the hardware layer, ESP8266 and ESP32 microcontrollers support self-organizing mesh topologies that maintain coverage without relying on centralized wireless access points [8, 9, 11].

Telemetry collection alone, however, does not constitute intelligence. This work addresses that gap by integrating Anthropic's Model Context Protocol (MCP) [5, 6, 7] as a structured interface between a time-series database and a large language model. Rather than evaluating each reading against a fixed rule, the LLM can reason across live and historical data simultaneously, identify spatial patterns of hazard escalation, and autonomously initiate actuator responses before conditions become critical.

The resulting system represents a substantive architectural advance over reactive monitoring: a proactive, AI-driven safety layer capable of anticipating and mitigating industrial hazards at the speed and granularity that conventional systems cannot achieve.

II. LITERATURE SURVEY

[1] Industrial Safety — Bhopal Gas Tragedy (Amina Sharif — August 2020) .The catastrophic 1984 leak of methyl isocyanate (MIC) gas from a Union Carbide pesticide plant in Bhopal, India is examined as one of the worst industrial disasters in history. Approximately 45 tons of toxic gas escaped, causing thousands of immediate deaths and severe long-term health consequences — including respiratory disease, vision loss, and birth defects — for hundreds of thousands of survivors. The work analyses how inadequate safety procedures, poorly maintained infrastructure, understaffed operations, and a complete lack of emergency preparedness collectively led to the catastrophe. It also discusses the lasting human, environmental, and legal consequences, including groundwater contamination and the inadequacy of corporate compensation. This case study is essential for contextualising the role of industrial safety failures and corporate negligence in large-scale chemical accidents and for motivating the need for real-time gas monitoring systems[2] Industrial Safety of Ammonia Accidents in Cold-Storage Facilities (Kocsis, Dobor, Kórodi, Kuk — 2015) .This paper addresses the industrial safety risks associated with anhydrous ammonia, one of the most widely produced and transported hazardous substances, with particular emphasis on cold-storage facilities where ammonia is commonly used as a refrigerant. The authors present a comprehensive safety analysis of documented ammonia-related industrial accidents, examining root causes such as mechanical failures, safety valve malfunctions, and circuit integrity losses. Consequence analysis is conducted to map hazardous danger zones arising from hypothetical ammonia release scenarios. The paper also provides guidance for first responders, industrial safety authorities, and higher education students on emergency preparedness and hazard management. This work is directly relevant to the proposed system as it establishes the critical need for continuous real-time ammonia monitoring in industrial environments. [3] Smart Factory Based on IoT Platform (Prihatno, Nurcahyanto, Jang — September 2020) .This conference paper investigates the application of an Internet of Things (IoT) platform built on the oneM2M standard for enhancing smart factory operations, including production, maintenance, and environmental monitoring. The authors demonstrate how physical and virtual devices can be interconnected through the oneM2M technology to achieve reliable data interoperability across factory systems. The implementation focuses on sensing environmental parameters within a factory setting, validating the platform's ability to support low-cost, efficient, and scalable manufacturing workflows. The paper highlights how IoT-based sensing infrastructure can contribute to Industry 4.0 goals by enabling real-time data collection and system integration. This work provides a foundational reference for IoT-driven environmental monitoring architectures applicable to industrial gas detection systems. [4] EMQX Broker for MQTT Protocol in IoT (Kashyap, Dev, Sharma — 2024) .This paper presents a detailed implementation and performance analysis of the EMQX broker operating the MQTT (Message Queuing Telemetry Transport) protocol in an IoT data transfer environment. The study evaluates broker performance across all three Quality of Service (QoS) levels — QoS 0, 1, and 2 — assessing published and subscribed message rates, average throughput, and actual QoS delivery. The authors introduce new metric sets for measuring message delivery performance, providing a rigorous quantitative framework for broker selection in IoT deployments. The findings demonstrate that EMQX offers reliable, scalable, and low-latency message brokering suitable for real-time sensor data applications. This work directly informs the communication architecture of the proposed gas monitoring system, where MQTT over EMQX is used for efficient,

low-power sensor-to-server data transmission. [5] Model Context Protocol — Anthropic Official Announcement (Anthropic — November 2024). This is the official release announcement by Anthropic introducing the Model Context Protocol (MCP), an open standard designed to enable AI systems to securely and consistently connect with external data sources, tools, and services. MCP establishes a universal interface — analogous to a USB-C standard for AI — that allows large language models (LLMs) to communicate with file systems, databases, APIs, and development environments through a standardised client-server architecture. The announcement outlines the motivation for MCP, its design principles, and its intended role in reducing the fragmented, custom integration overhead that currently limits scalable AI deployment. This primary source is referenced to establish the origin, purpose, and design intent of the MCP framework used in the proposed system. [6] Model Context Protocol — Official Technical Documentation (Model Context Protocol — 2024). This is the official introductory technical documentation for the Model Context Protocol (MCP), providing a structured overview of its architecture, core concepts, and implementation guidelines for developers. The documentation describes MCP's client-server model, the standardized JSON-RPC messaging layer, the four core primitives (Resources, Tools, Prompts, and Sampling), and the dynamic tool discovery mechanism that allows AI hosts to connect with multiple MCP servers simultaneously. It also outlines security considerations and the protocol's transport layer options. This reference serves as the primary technical specification consulted during the implementation of MCP-based AI integrations within the proposed system architecture. [7] Survey of the Model Context Protocol (Singh, Ehtesham, Kumar, Khoei — April 2025). This preprint provides the first comprehensive academic survey of the Model Context Protocol (MCP), critically reviewing its foundational architecture, standardised messaging protocols, dynamic tool discovery, and built-in security mechanisms. The authors position MCP against the backdrop of fragmented, custom API-based integrations that currently hinder scalable Agentic AI deployment, arguing that MCP offers a structured and interoperable alternative. The survey evaluates MCP's potential applications across multiple domains including finance, healthcare, and customer service, while also identifying current limitations in empirical performance data. It serves as the primary academic reference for understanding MCP's design rationale, architectural principles, and place within the broader landscape of LLM integration technologies relevant to this work. [8] ESP-MESH Protocol for Indoor and Outdoor Air Quality Monitoring (Khan et al. — December 2022). This paper presents the design and deployment of a wireless sensor network (WSN) based on the ESP-MESH networking protocol using ESP32 microcontrollers for real-time indoor and outdoor air quality monitoring across a college campus. Multiple sensor nodes each equipped with arrays measuring CO₂, CO, air quality, temperature, humidity, and PM_{2.5} are interconnected through the ESP-MESH network, which eliminates the need for a central access point and supports self-healing topology. Data is collected from each node every two minutes over a 72-minute monitoring window, demonstrating the system's ability to maintain continuous multi-parameter surveillance over a wide physical area at low cost. The study validates ESP-MESH as a robust, scalable, and energy-efficient alternative to conventional Wi-Fi star networks for distributed environmental monitoring, directly informing the mesh-based communication strategy of the proposed gas detection system. [9] Performance Evaluation of ESP8266 Mesh Networks (Yoppy et al. — 2019). This paper evaluates the communication performance of a mesh network formed using ESP8266 modules running the open-source `painlessMesh` library, which enables low-cost microcontroller nodes to self-configure and form an ad-hoc Wi-Fi mesh without a central access point. The study measures one-way network delay and data rate across network configurations of 2, 3, 10, and 16 nodes, finding that a 2-node network achieves a delay of approximately 2.49 ms, while larger networks exhibit progressively increased latency. It also identifies a critical payload threshold — messages exceeding 4,400 bytes begin to produce incomplete or erroneous transmissions. These quantitative performance benchmarks are directly applicable to the proposed system's design, providing a basis for selecting appropriate message sizes and anticipating latency behaviour in multi-node ESP8266 or ESP32 mesh deployments. [10] Remote Monitoring System Using ESP32 and ThingSpeak (Nurmiainen — 2023). This master's thesis documents the end-to-end development of a remote IoT monitoring system for a smart home environment, built around the ESP32 microcontroller and the ThingSpeak cloud platform. The work covers hardware selection and sensor integration, firmware development, Wi-Fi-based data transmission, and cloud-side data visualisation and analysis via ThingSpeak's MATLAB-compatible analytics engine. The thesis evaluates system reliability, data transmission stability, and the suitability of ThingSpeak as a low-cost cloud backend for continuous IoT monitoring. It demonstrates how a cost-effective ESP32-based architecture can provide persistent remote visibility into environmental conditions — a design pattern directly relevant to the cloud connectivity and dashboard components of the proposed gas monitoring system. [11] Performance Assessment of ESP8266 Wireless Mesh Networks (Santos et al. — May 2022). This paper presents a systematic experimental evaluation of a wireless mesh network testbed constructed from ESP8266 nodes using the `painlessMesh` library, assessed for its feasibility as a monitoring solution for perishable goods in refrigerated storage and transportation. The study conducts performance testing under unicast transmission across varying node counts, traffic loads, and message payload sizes, measuring delivery ratio and delivery delay as primary metrics. Results reveal

significant volatility in delivery delay with high confidence error margins, and confirm that additional nodes increase network congestion. The study provides both a performance baseline and practical design constraints for ESP8266-based mesh deployments, offering valuable empirical guidance for configuring the multi-node mesh architecture used in the proposed IoT gas monitoring system. [12] Multimodal Gas Detection and Classification Dataset (Shah — 2021). This Kaggle-hosted dataset, based on the MultimodalGasData publication, provides a novel multimodal collection of simultaneous gas sensor readings and thermal camera images intended for training and benchmarking gas detection and classification models. The dataset comprises 6,400 samples across four classes smoke, perfume, a mixture of smoke and perfume, and a neutral (no-gas) environment collected using an array of seven metal-oxide gas sensors (MQ2, MQ3, MQ5, MQ6, MQ7, MQ8, MQ135) combined with a thermal imaging camera. The dual-modality approach addresses the limitation of sensor-only detection, which is constrained by low sensitivity and short range, by incorporating thermal spatial information. This dataset is used in the proposed work to develop and validate the machine learning model responsible for gas type classification from multi-sensor array readings.

III. METHODOLOGY

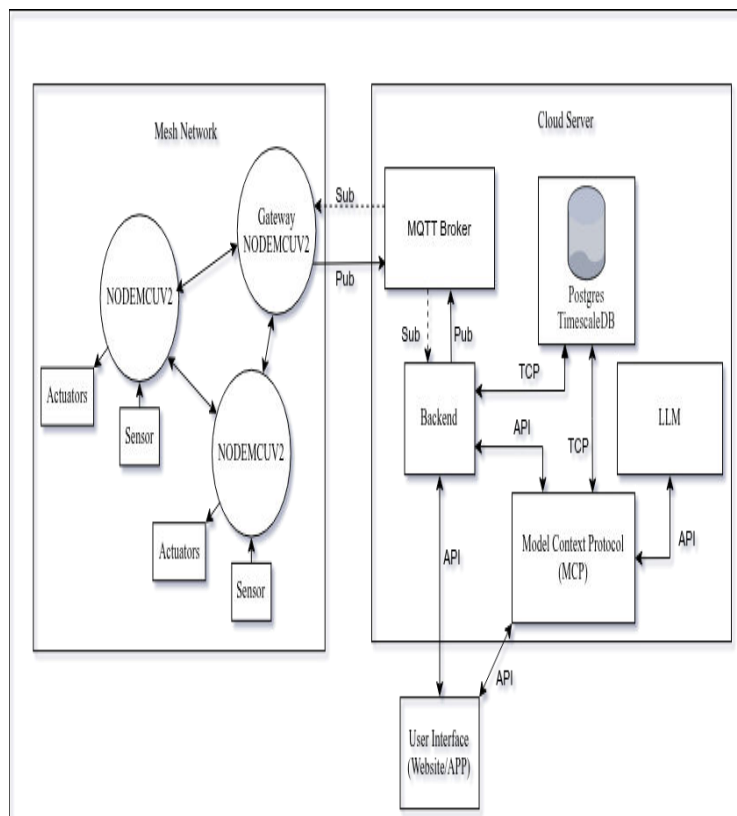


Figure 1 End-to-End IoT system architecture

The system is organized across two layers: an edge mesh network and a cloud backend. At the edge, NodeMCU V2 microcontrollers operate in a mesh topology, each interfacing with environmental sensors and local actuators. Telemetry is relayed node-to-node until it reaches a designated gateway, which serves as the sole bridge to the cloud, publishing all consolidated sensor data to an MQTT broker and subscribing to a command topic through which actuation instructions are distributed back into the mesh, forming a closed-loop control cycle. On the cloud side, a Go-based backend subscribes to the MQTT broker, ingests incoming payloads, and persists them into a TimescaleDB-extended PostgreSQL database optimized for high-frequency time-series writes. This telemetry is exposed through a Model Context Protocol (MCP) server, which provides a standardized interface between the stored data and the LLM over TCP, enabling contextual reasoning over both live and historical readings rather than simple threshold evaluation. Hazard assessments and actuation commands generated by the LLM are routed back through the

backend and broker to the edge actuators. A React-based dashboard connects to the backend via REST API, giving operators real-time visibility and manual control across all monitored nodes.

IV. EXPERIMENTAL RESULTS

A. System Testing Overview

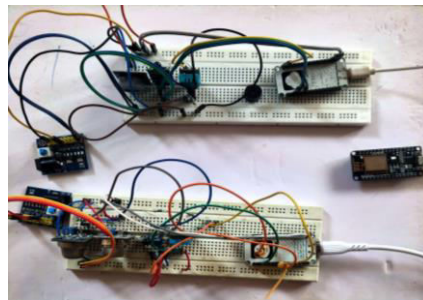


Figure 2: Physical Hardware Setup Two Sensor Nodes and Gateway

The system was evaluated using two sensor nodes and one gateway. Node 1 interfaced four sensors and Node 2 three sensors, providing simultaneous environmental coverage across two independent zones. Fig. 2 shows the physical hardware setup comprising two breadboard-mounted sensor nodes and the gateway used throughout testing. Over the testing period, approximately 10,000 time-series records were accumulated and partitioned within the TimescaleDB backend, confirming the storage pipeline's capacity for sustained multi-node ingestion.

B. Data Acquisition and Latency

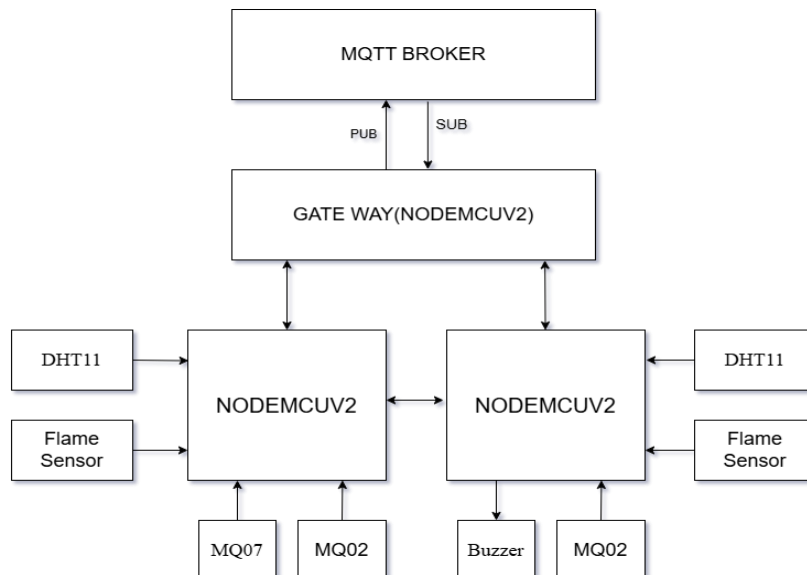
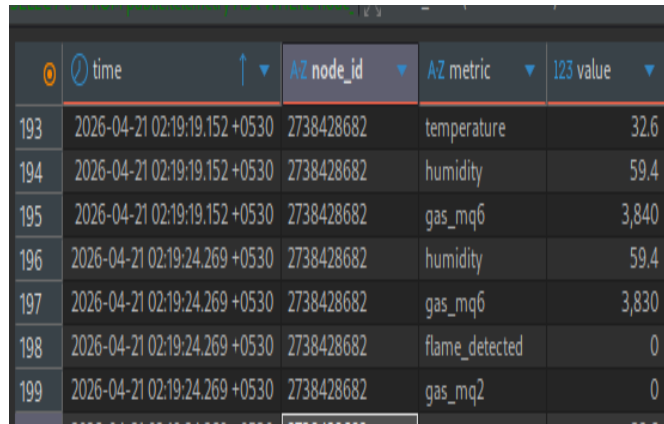


Figure 3: Sensor Block Diagram Node Configuration and Monitored Parameters

Both nodes successfully relayed environmental readings through the mesh to the gateway, which forwarded telemetry to the HiveMQ broker over MQTT with TLS. Fig. 3 illustrates the sensor configuration for each node, detailing the parameters monitored per zone. No significant transmission delays were observed during continuous dual-node operation, consistent with prior benchmarks reporting sub-3 ms one-way mesh delay for two-node ESP8266

deployments [9] and low-latency MQTT delivery under EMQX and HiveMQ brokers [4]. The backend ingested and persisted all payloads without data loss across all sensor channels.



	time	AZ node_id	AZ metric	123 value
193	2026-04-21 02:19:19.152 +0530	2738428682	temperature	32.6
194	2026-04-21 02:19:19.152 +0530	2738428682	humidity	59.4
195	2026-04-21 02:19:19.152 +0530	2738428682	gas_mq6	3,840
196	2026-04-21 02:19:24.269 +0530	2738428682	humidity	59.4
197	2026-04-21 02:19:24.269 +0530	2738428682	gas_mq6	3,830
198	2026-04-21 02:19:24.269 +0530	2738428682	flame_detected	0
199	2026-04-21 02:19:24.269 +0530	2738428682	gas_mq2	0

Figure 4: TimescaleDB Accumulated Sensor Telemetry Records

Fig. 4 shows the accumulated telemetry records within TimescaleDB, confirming reliable pipeline continuity throughout testing.

C. Hazard Detection and Actuator Response

Edge-level dynamic thresholds enabled each node to evaluate sensor readings locally, independent of cloud connectivity. Threshold breaches triggered the buzzer actuator with near-zero latency while simultaneously publishing alerts to the MQTT broker, demonstrating resilient local detection. A real-world MQ sensor dataset was loaded into the MCP context to simulate representative industrial gas exposure scenarios, allowing the LLM to reason over realistic hazard patterns rather than synthetic values.

D. LLM and MCP Performance

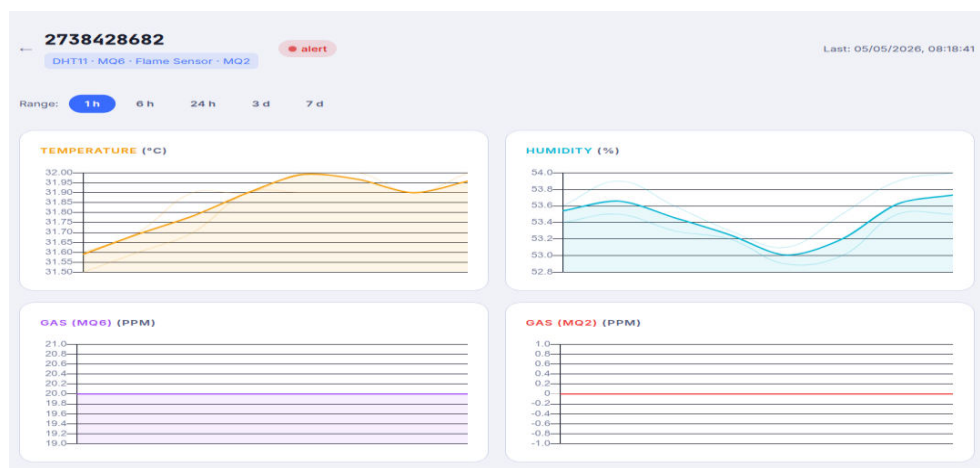


Figure 5: React Dashboard Live Sensor Monitoring Across Two Nodes

The LLM integration performed accurately under a query-response interaction model. Fig. 4 shows the React Dashboard displaying live sensor readings and node-level status across both monitored zones during active testing.

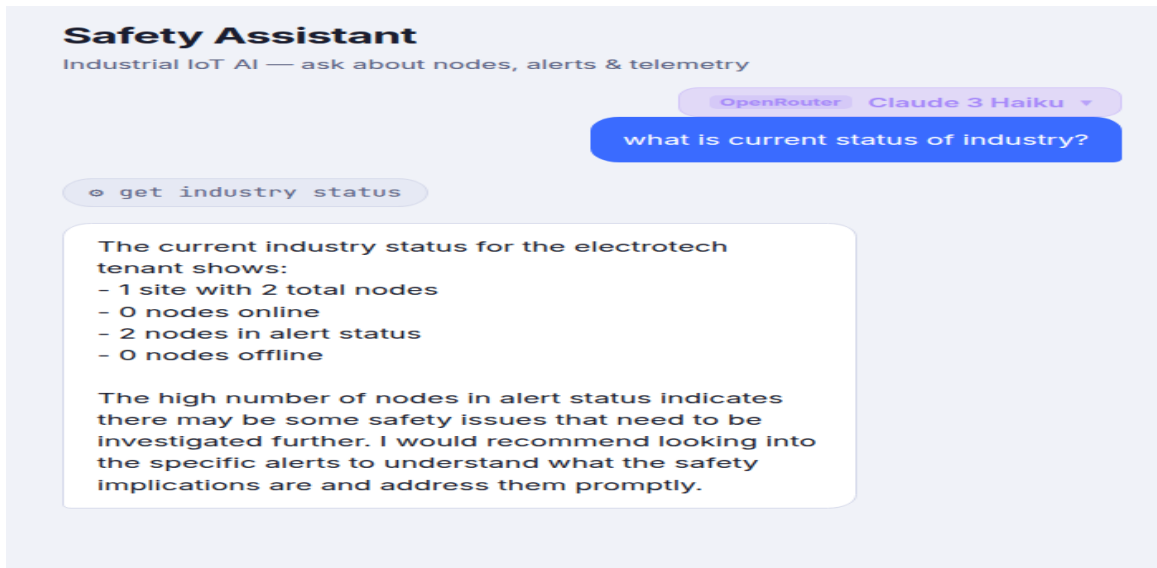


Figure 6: LLM Safety Assessment Response via Model Context Protocol

The system returned contextually appropriate safety assessments derived from live and historical telemetry. Fig. 6 shows the LLM response captured through the interface, where the system correctly identified the originating node and confirmed the issued alert, demonstrating that MCP-provided context supplied sufficient spatial and environmental information for meaningful AI reasoning. Edge-level thresholds handled immediate detection, with the LLM serving as the contextual analysis and structured reporting layer.

E. Objectives Achieved

All four project objectives were met. Distributed multi-node hardware was deployed and validated. Over 10,000 sensor readings were captured with high integrity via the TimescaleDB pipeline. MCP integration successfully bridged stored telemetry to the LLM, enabling node-level hazard identification. Simulated industrial testing confirmed low-latency actuator response and accurate AI-generated safety assessments across both monitored zones.

V. CONCLUSION

This paper presented an AI-augmented IoT safety system designed to overcome the fundamental shortcomings of conventional industrial hazard monitoring. By deploying a painlessMesh network of NodeMCU V2 sensor nodes to continuously track CO, LPG, smoke, flame, temperature, and humidity, and routing telemetry through a secured MQTT gateway into a Go-based backend with TimescaleDB storage, the system established a robust and scalable data pipeline capable of accumulating over 10,000 time-series records across two monitored zones without data loss. Dynamic edge-level thresholds enabled near-zero-latency local actuation independent of cloud availability, ensuring that immediate hazard responses were never constrained by network conditions. The integration of a large language model via the Model Context Protocol elevated the system beyond simple alarm generation: the LLM performed node-level hazard localization, interpreted environmental trends in context, and issued structured safety assessments that static threshold engines are incapable of producing. All four stated objectives were fulfilled. The architecture was validated as reliable, responsive, and extensible, confirming its suitability for deployment in real-world high-risk industrial environments where proactive safety infrastructure is essential.

REFERENCES

- [1] A. Sharif, "Case Study for Bhopal Gas Tragedy," *ResearchGate*, Aug. 2020, doi: 10.13140/RG.2.2.16473.75364. https://www.researchgate.net/publication/343601060_Case_study_for_Bhopal_Gas_Tragedy
- [2] Z. Kocsis, J. Dobor, G. Kóródi, and E. Kuk, "Industrial Safety Analysis of Accidents Involving Ammonia, with Special Regard to Cold-Storage Facilities," *Academic and Applied Research in Military and Public Management Science (AARMS)*, vol. 14, no. 3, pp. 273–284, 2015, doi: 10.32565/aarms.2015.3.2. https://www.researchgate.net/publication/358292833_Industrial_Safety_Analysis_of_Accidents_Involving_Ammonia_with_Special_Regard_to_Cold-Storage_Facilities
- [3] A. T. Prihatno, H. Nurcahyanto, and Y. M. Jang, "Smart Factory Based on IoT Platform," in *Proc. Korea IoT Conf. (KIC) Summer Conf.*, Sep. 2020, pp. 2–4, doi: 10.3390/MACHINES6020023 https://www.researchgate.net/publication/344199006_Smart_Factory_Based_on_IoT_Platform
- [4] M. Kashyap, A. K. Dev, and V. Sharma, "Implementation and Analysis of EMQX Broker for MQTT Protocol in the Internet of Things," *e-Prime: Advances in Electrical Engineering, Electronics and Energy*, vol. 10, Art. no. 100846, 2024, doi: 10.1016/j.prime.2024.100846. <https://www.sciencedirect.com/science/article/pii/S277267112400425X>
- [5] Anthropic, "Model Context Protocol," Nov. 2024. <https://www.anthropic.com/news/model-context-protocol>
- [6] Model Context Protocol, "Introduction," 2024. <https://modelcontextprotocol.io/docs/getting-started/intro>
- [7] A. Singh, A. Ehtesham, S. Kumar, and T. T. Khoei, "A Survey of the Model Context Protocol (MCP): Standardizing Context to Enhance Large Language Models (LLMs)," *Preprints.org*, Apr. 2025, doi: 10.20944/preprints202504.0245.v1. <https://www.preprints.org/manuscript/202504.0245>
- [8] A. U. Khan *et al.*, "An Efficient Wireless Sensor Network Based on the ESP-MESH Protocol for Indoor and Outdoor Air Quality Monitoring," *Sustainability*, vol. 14, no. 24, p. 16630, Dec. 2022, doi: 10.3390/su142416630. <https://www.mdpi.com/2071-1050/14/24/16630>
- [9] Yoppy *et al.*, "Performance Evaluation of ESP8266 Mesh Networks," *Journal of Physics: Conference Series*, vol. 1230, no. 1, p. 012023, 2019, doi: 10.1088/1742-6596/1230/1/012023. <https://iopscience.iop.org/article/10.1088/1742-6596/1230/1/012023/pdf>
- [10] O. Nurmiainen, "Development of a Remote Monitoring System for a Smart Home Using an ESP32 Microcontroller and ThingSpeak," Master's Thesis, Turku University of Applied Sciences, 2023 https://www.theseus.fi/bitstream/handle/10024/871661/Nurmiainen_Olli.pdf
- [11] L. Santos *et al.*, "Performance Assessment of ESP8266 Wireless Mesh Networks," *Information*, vol. 13, no. 5, p. 210, May 2022, doi: 10.3390/info13050210. <https://www.mdpi.com/2078-2489/13/5/210>
- [12] A. Shah, "Multimodal Gas Detection and Classification Dataset," *Kaggle*, 2021. <https://www.kaggle.com/datasets/aryashah2k/multimodal-gas-detection-and-classification/data>



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details