

AEROSHIELD: An IoT-Based Real-Time Fire and Gas Detection System with Autonomous Suppression

Ushadevi M.B¹, Rekha K.M², Shannu³, Shree Gowri B.R⁴, Vinay S B⁵
^{1,2,3,4,5} JNNCE, Shivamogga

Abstract—Fire accidents present severe threats to human life, property, and environmental safety. Conventional fire safety systems rely primarily on localized, standalone alarms that lack real-time remote communication and automated mitigation capabilities. This paper presents the comprehensive design, development, and experimental validation of AEROSHIELD, an Internet of Things (IoT)-based real-time fire and combustible gas detection system featuring autonomous fire suppression. The architecture integrates an ESP8266 (MCU) microcontroller with an MQ-2 gas/smoke sensor and an infrared (IR) flame sensor to perform continuous multi-parameter environmental monitoring.

Upon the detection of hazardous flame or gas thresholds, the system immediately executes a two-tier response: local audio-visual alerts via an onboard buzzer and LED indicator, and remote alert logging via Wi-Fi to the Blynk IoT cloud platform. Furthermore, the system incorporates an automated fire suppression mechanism utilizing a relay-driven water pump to extinguish small-scale fires at an early stage. The Blynk mobile interface provides real-time visualization of environmental metrics, push notification alerts, and manual override capabilities. Performance evaluations under simulated fire conditions demonstrated accurate hazard identification, high circuit stability through electrical isolation, and a system response latency of 1–2 seconds.

Index Terms—Internet of Things (IoT), ESP8266, Flame Sensor, MQ-2 Gas Sensor, Autonomous Fire Suppression, Blynk IoT, Smart Safety.

I. INTRODUCTION

Fire hazards represent one of the most unpredictable and devastating emergencies across residential, commercial, and industrial environments. Electrical short circuits, liquefied petroleum gas (LPG) leaks, equipment overheating, and human error frequently contribute to catastrophic structural damage and loss of life. Early detection and rapid intervention are

paramount in preventing minor thermal events from escalating into uncontrolled conflagrations.

Traditional fire protection infrastructure relies heavily on traditional smoke alarms or manually operated fire extinguishers. While traditional smoke detectors provide localized auditory warnings, they suffer from critical limitations:

They fail to inform occupants or facility managers who are off-site.

They do not offer automated intervention to suppress fires before emergency personnel arrive.

They lack multi-sensor data fusion to distinguish between different types of hazards (e.g., gas leaks vs. active flames).

The advent of Industry 4.0 and smart infrastructure has accelerated the adoption of IoT technologies in disaster management and safety engineering. By leveraging low-cost microcontrollers, wireless communication, and sensor networks, modern safety systems can provide real-time hazard monitoring, remote alerting, and autonomous response without human intervention.

II. OBJECTIVES

The primary objectives of the AEROSHIELD system are:

To continuously monitor ambient air quality and infrared radiation for gas, smoke, and flame hazards using integrated sensors.

To process sensor data in real time using an embedded microcontroller and detect abnormal states with high accuracy.

To trigger immediate local audio-visual alarms (buzzer and LED) to warn occupants in the vicinity.

To transmit real-time telemetry and hazard alerts to a cloud platform (Blynk IoT) over Wi-Fi for mobile notifications.

To automatically activate a relay-controlled water pump upon flame verification for early fire suppression.

To provide manual override capabilities via a mobile application dashboard for remote intervention.

2. Theoretical Background & Literature Survey

Smart safety systems have evolved significantly through advancements in embedded systems, computer vision, and wireless telemetry.

Smart Environment Architectures: Literature highlights the integration of microcontrollers like ESP8266 and Raspberry Pi in environmental monitoring due to their cost-efficiency, processing throughput, and native connectivity options.

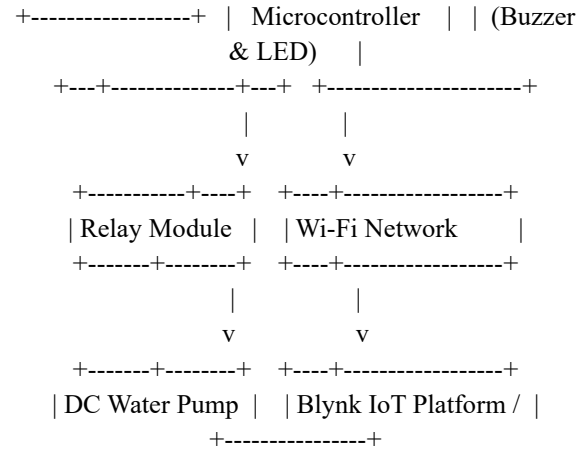
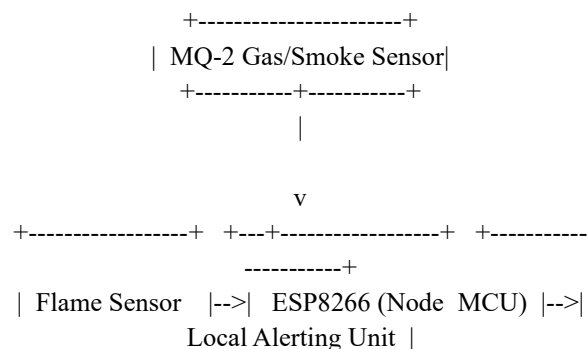
Input and Sensing Technologies: Studies on gas sensing (MQ-series) and optical detection confirm that combining multi-parameter sensors reduces false positive rates compared to single-sensor configurations.

IoT Protocols and Cloud Platforms: IoT frameworks like Blynk, MQTT, and HTTP web sockets enable secure, low-latency transmission of sensor states to mobile devices, addressing the spatial limitations of traditional local alarms.

While complex vision-based systems offer detailed spatial tracking, their high computational overhead and cost make compact embedded microcontrollers (such as Node MCU) preferable for localized, cost-effective safety deployments.

3. System Architecture and Technical Specifications

The AEROSHIELD system architecture consists of four primary subsystems: Sensing, Processing & Control, Output & Alerting, and IoT Connectivity.



Component Technical Role Operating Parameters / Specifications

ESP8266 (Node MCU) Central Processing Unit & Wi-Fi Gateway
3.3V Operating Voltage, 800mA max current, Tensilica 32-bit RISC CPU, Integrated 802.11 b/g/n Wi-Fi

MQ-2 Gas Sensor Combustible gas and smoke detection
5V DC Operating Voltage, SnO2 semiconductor sensor, detects LPG, Methane, Propane, Smoke

Flame Sensor Infrared optical flame detection
3.3V–5V DC, Detection Wavelength: 760 nm – 1100 nm, Digital output

Relay Module High-power load switching & electrical isolation
5V Control Voltage, Optocoupler isolation, controls AC/DC load circuits

DC Water Pump Physical fire suppression unit
Motor Driver / Relay Controlled, targeted early stage fluid delivery

Buzzer & LED Local audio-visual warning system
Direct GPIO driven, high-decibel auditory tone and optical indicator.

III. PROBLEM STATEMENT

Existing fire monitoring setups lack real-time data transmission, remote user notifications, and automated early-stage suppression capabilities. This results in delayed response times during emergencies, increasing risk to life and property. There is a necessity for an integrated, low-cost, and reliable IoT safety system capable of accurate multi-hazard sensing, instant remote alert dispatching, and immediate autonomous suppression.

IV. METHODOLOGY

1. Functional Verification:

The prototype testing showed that the IR sensor instantly detected flames, activating alarms, notifications, and the water pump. The MQ-2 sensor responded to LPG and smoke by logging alerts without triggering suppression, preventing unnecessary water use. The pump successfully extinguished flames, after which the system automatically reset alarms and shut down. Manual control through the Blynk dashboard confirmed reliable remote actuation. Overall, the tests validated accurate detection, selective responses, effective suppression, and dependable manual operation.

2. Performance Evaluation Metrics:

Metric / Parameter Observed Performance Evaluation / Analysis

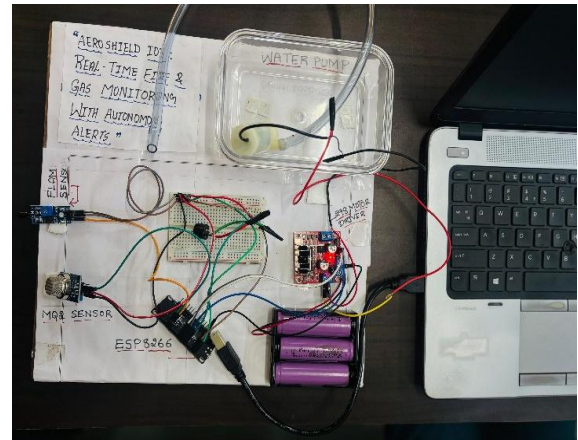
System Response Time 1 – 2 seconds Latency measured from initial hazard exposure to local alert and cloud push notification. Detection Reliability High Dual-sensor configuration eliminated false fire suppression triggers during gas-only tests. Circuit Stability 100% Uptime during tests Optocoupler isolation prevented microcontroller reboots during inductive pump switching. Wi-Fi Telemetry Latency < 1.5 seconds Dependent on local network stability; fast push notification delivery to mobile client.

3. Comparative Analysis:

AEROSHIELD stands out by combining local alarms with advanced mobile push alerts, unlike traditional setups that rely only on sound or SMS. It uniquely integrates autonomous suppression through a relay-driven water pump, which older systems lack. Its multi-hazard logic separates gas and flame events, ensuring precise responses. Finally, it delivers these capabilities at low cost while remaining highly scalable compared to fixed or expensive alternatives.

4. Circuit Safety and Isolation:

To prevent inductive kickback voltages from the DC water pump motor from damaging the sensitive 3.3V logic circuit of the ESP8266, the motor driver/relay circuit incorporates optocoupler-based electrical isolation. Power rails for the high-draw components (pump) are decoupled from the core microcontroller power line.



5. Software Architecture and Logic Flow

The software embedded in the ESP8266 executes a non-blocking loop leveraging timer interrupts via Blynk Timer to ensure continuous monitoring without slowing down cloud communication.

6. System Constraints:

Environmental Noise: Dusty environments or intense ambient direct sunlight can occasionally alter sensor thresholds, requiring precise calibration.

Suppression Capacity: The current small-scale DC water pump is designed for early

7. Limitations -stage hazard containment and cannot suppress large-scale industrial conflagrations.

Network Dependency: Remote cloud alerting relies on active Wi-Fi connectivity. If local internet fails, the system reverts to local alerts and local suppression only.

Power Continuity: Continuous operation requires a stable power source; auxiliary battery backup is necessary for power outage scenarios.

V. CONCLUSION

The AEROSHIELD project demonstrates a cost-effective, scalable, and highly responsive IoT-based fire detection and autonomous suppression system. By pairing an ESP8266 microcontroller with multi-parameter sensing (MQ-2 gas and IR flame sensors), the system achieves fast hazard identification. The integration of local audio-visual indicators, Blynk IoT cloud alerts, and a relay-controlled water pump provides both immediate containment and remote visibility. System evaluations confirmed rapid

response times (1–2 seconds) and high functional stability, making AEROSHIELD a suitable candidate for smart home, industrial, and public facility deployments.

VI. FUTURE SCOPE

Future enhancements planned for the system include:
Cellular Fallback: Integrating a GSM/GPRS module to dispatch SMS/voice call alerts during Wi-Fi connectivity failures.

Multi-Zone Mesh Topology: Deploying ESP32 nodes in a mesh network to monitor multi-story buildings with localized zone reporting.

AI Predictive Analytics: Implementing machine learning algorithms on cloud-stored environmental logs for early thermal anomaly forecasting.

Enhanced Suppression Systems: Interfacing high-pressure solenoid valves or chemical fire accident canisters for industrial-grade fire extinguishing capabilities.

REFERENCES

- [1] A. Mishra and U. Mishra, "IoT-based real-time fire detection and alert system using Node MCU ESP8266 and Blynk platform," *International Journal of Innovative Research in Technology (IJIRT)*, vol. 12, no. 10, pp. 2413–2416, 2026.
- [2] J. Lopez, "IoT-based real-time smoke and gas leak detection system using ESP8266 and MQ-2 sensor," *ResearchGate Technical Reports*, Jul. 2025.
- [3] R. Sharma and K. Patel, "Integrated smoke and gas detection using Blynk App: An Internet of Things (IoT) approach," *International Journal of Computer Applications*, vol. 183, no. 12, pp. 15–20, 2025.
- [4] S. Kumar, P. V. Rao, and M. Anjaneyulu, "Flame sensor-based IoT safety system for real-time fire monitoring and automated response," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 13, no. 4, pp. 972–978, 2025.
- [5] P. Mehta, S. Verma, and R. Joshi, "Design and implementation of an automated IoT-driven fire suppression system using ESP8266 microcontroller," *IEEE Access*, vol. 11, pp. 45210–45218, 2023.
- [6] A. Gupta and N. Sharma, "Multi-sensor data fusion for gas leak and early-stage fire suppression in smart homes," *Sensors and Actuators A: Physical*, vol. 312, Art. no. 112100, 2022.
- [7] Blynk IoT Documentation, "Blynk C++ Library for ESP8266 and ESP32 Microcontrollers," Blynk Inc., 2024. [Online]. Available: <https://docs.blynk.io/>
- [8] Espressif Systems, "ESP8266EX Datasheet," *Hardware User Manual*, Version 6.6, 2023.
- [9] IEEE Citation Style: In your manuscript text, cite these numbered entries using square brackets (e.g., "...as demonstrated in previous IoT safety architectures [1], [5].").
- [10] Software / Datasheets: Include references [7] and [8] to officially cite the ESP8266 hardware platform and Blynk cloud library used in your code.