Open Access and Peer Reviewed Journal ISSN 2394-2231 https://www.ijctjournal.org/

RESEARCH ARTICLE OPEN ACCESS

AI-Enabled IoT Smart Air Safety System for Kitchens and Elderly People with Anosmia

Girdhar Maheshwari
Student, Department of Computer Science
Institute of Information Technology & Management New Delhi, India
Girdharmaheshwari2005@gmail.com

Rohit Kumar
Assistant Professor, Department of Computer Science
Institute of Information Technology & Management New Delhi, India
Rjangra696@gmail.com

Abstract — Gas leakages in household kitchens pose a consistent threat, especially among elderly people and people with anosmia, a disease that adversely affects their sense of smell. Traditional monitoring mechanisms are limited to basic trigger-based actions and local alerts using buzzers. They lack in intelligent capabilities such as connectivity that would enable them timely and reliable responses. This study focuses on artificial intelligence-enhanced IoT-enabled Smart Air Safety System that continually detects the concentration of gas , temperature and humidity levels using a MQ135 gas sensor and DHT11 sensor is connected to NodeMCU (ESP8266). The detection-based filter algorithm differentiates the status of the environment into three categories: safe, moderate, and danger while it minimizes false alarms that result in sudden bursts of the sensors' outputs. During emergencies, the smart system automatically controls the activation of the exhaust fan using a servo motor, activates buzzer and LED alarms, and pushes notifications through the Blynk IoT platform. A web dashboard integrated within the ESP8266 web server shows live data from the sensors and generates analytics. Results show correct categorization of multiple environmental states, successful false alarm prevention, and less than one-second delay in response time.

Keywords *— IoT, gas detection, anosmia, NodeMCU, ESP8266, smart home safety, false alarm filtering, Blynk.*

I. Introduction

Kitchen gas leakages are one of the most common and potentially most avoidable household accidents. The most popular kitchen fuels, liquefied petroleum gas (LPG) and methane, are colourless and odourless in their pure state; the distinctive smell of domestic gas supply is deliberately introduced as a safety precaution. However, many members of the community, including the elderly and those with anosmia, are unable to consistently smell gas leaks [1]. Congenitally acquired anosmia, disease, head injury and even old age all remove the most common sensory system that the majority of people use in a gas leak situation. Existing gas sensors provide local audible warnings without remote monitoring and physical actuation capability, leaving a sleeping, or hearing-impaired person, or a person not present in the kitchen in a dangerous situation. The recent explosion in the availability of low-cost Wi-Fi-capable microcontrollers - in particular the ESP8266 series - has allowed us to develop embedded safety monitoring systems that integrate sensor fusion, minimal intelligence, cloud communication and electromechanical actuation at a cost that is affordable to the typical household. Our main contributions include: (i) a consecutive detection-based filtering algorithm that discerns hazards from sensor noise; (ii) a close integration with the Blynk IoT platform for mobile alerting; (iii) an on-board web dashboard to display the trend analysis through a graph; and (iv) a closed-loop automated actuation (exhaust fan, buzzer, LED indicators) that occurs within one second following a confirmed danger.

II. Literature Review

Significant progress has been made in IoT-based gas leakage detection and smart kitchen safety in recent years, but there remain some key areas of improvement. Sharma et al. [2] used an MQ-series sensor and an ESP8266 for remote monitoring but used only a constant threshold, leading to a high false alarm rate. Patel and Mehta [3] produced acceptable alert latency with Wi-Fi messages, but there was no automated physical response, leaving the user to take action. Khan et al. [4] used deep-neural networks to process sensor data, successfully lowering false alarms; however, neural inference on an energy-efficient microcontroller was not feasible and required upload to the cloud, with a consequent latency dependence. Yadav et al. [5] used a combination of gas sensors, GSM alerts and cloud logging but did not include any filtering; therefore, every spike in the sensors would trigger an SMS message. Verma et al. [6] extended gas sensing with smoke and temperature sensors but did not include any data visualisation and tested the system in a stationary lab environment. Similarly, Gupta and Singh [8] included auto exhaust control but no data dashboard. More recent research by Singh et al. [9] harnessed AI-based anomaly-detection models, and improved sensitivity, at the expense of complexity and cost of deployment. Overall, the literature shows three major deficiencies: on-device false-alarm filtering; joint provision of cloud logging, on-device analytics, and physical actuation; and specific attention to anosmic and age-impaired users.

TABLE I

Feature Comparison With Related Work

System	Filter	Auto Act.	Dashboard	Mobile Alert	Vuln. User
Sharma et al. [2]	No	No	No	Yes	No
Patel & Mehta [3]	No	No	No	Yes	No
Khan et al. [4]	Partial	No	No	No	No
Yadav et al. [5]	No	No	No	Yes	No
Gupta & Singh [8]	No	Yes	No	Yes	No
Verma et al. [6]	No	No	No	No	No
Proposed System	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

III. Proposed System Design

A. System Architecture

The proposed system follows a four-layer architecture comprising: (1) a Sensing Layer that acquires environmental data; (2) a Processing and Filtering Layer that classifies hazard states and suppresses false alarms; (3) a Communication Layer that disseminates alerts and sensor data; and (4) an Actuation Layer that executes autonomous physical responses. Fig. 1 illustrates the data flow between these layers.

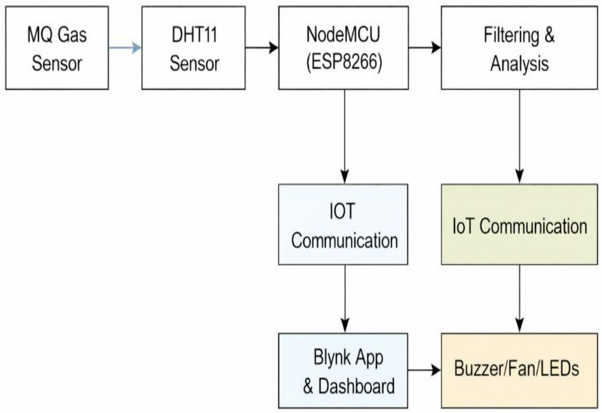

Fig. 1. System block diagram showing the four-layer architecture.

B. Hardware Components

The NodeMCU ESP8266 microcontroller serves as the central processing unit. Operating at 80 MHz with 80 KB of RAM and built-in 802.11 b/g/n Wi-Fi, it provides sufficient computational headroom for sensor reading, classification logic, web serving, and Blynk communication running concurrently. The MQ-2 gas sensor detects LPG, methane, smoke, and hydrogen through a chemi-resistive mechanism, with its analog output sampled on pin A0 yielding a 10-bit ADC reading in the range 0–1023. The DHT11 sensor, connected to digital pin D4, measures ambient temperature (0–50 °C, ±2 °C) and relative humidity (20–90% RH, ±5%), providing supplemental environmental context.

Output components include three LEDs (green D5 — Safe, yellow D6 — Moderate, red D7 — Danger), an active buzzer on D0 for audible alerts, and a servo motor on D3 that actuates a miniature exhaust fan. An OLED display (128 × 64 px, SSD1306, I2C on D1/D2) presents local feedback. The complete pin-to-component mapping is shown in Table I, and Fig. 2 shows the assembled circuit.

TABLE II

Hardware Pin Connections

Component	Pin	Function
MQ Gas Sensor	A0	Analog gas level input
DHT11 Sensor	D4	Temperature & humidity
Green LED	D5	Safe condition indicator
Yellow LED	D6	Moderate condition indicator
Red LED	D7	Danger condition indicator
Buzzer	D0	Audible alert output
Servo Motor	D3	Exhaust fan control
OLED SDA	D2	I2C data line
OLED SCL	D1	I2C clock line

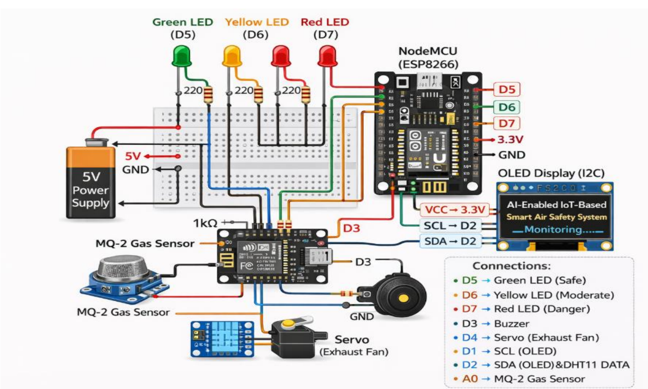

Fig. 2. Circuit diagram showing hardware pin connections and wiring.

C. Software and Firmware Design

The firmware is written in Embedded C/C++ and compiled via the Arduino IDE with the ESP8266 board package. Key libraries include ESP8266WiFi for network connectivity, BlynkSimpleEsp8266 for cloud communication, Adafruit_SSD1306 and Adafruit_GFX for OLED rendering, the DHT sensor library for environmental readings, and the built-in Servo library for exhaust fan control. The ESP8266WebServer library instantiates an HTTP server on port 80. The root route serves a single-page dashboard built from PROGMEM-stored HTML/CSS/JavaScript chunks to avoid RAM exhaustion, while the "/data" route returns a JSON object with real-time sensor values polled by the dashboard every 2 seconds using XMLHttpRequest. Blynk communication uses a non-blocking 3-second timeout so service disruptions do not halt local sensing; a BlynkTimer pushes gas, temperature, humidity, and status to seven virtual pins (V0–V7) every 2 seconds.

D. Consecutive Detection-Based Filtering Algorithm

Raw ADC readings from the MQ sensor are susceptible to short-duration spikes caused by temperature changes, supply voltage noise, or cross-sensitivities to non-hazardous compounds. The proposed filtering algorithm requires a hazardous condition to be observed consecutively for a configurable number of cycles before the system confirms and acts upon it. Each reading is classified using two thresholds: GAS_SAFE_MAX = 170 ADC units and GAS_MODERATE_MAX = 250 ADC units. A confirmed status is promoted from Safe only after CONSECUTIVE_THRESHOLD = 5 consecutive readings exceed the relevant boundary — a 5-second persistence window that responds promptly to genuine leaks while filtering sub-5-second transients. The classification rule is expressed as:

$$Status = f\,(gas,\ N\_consecutive) \in \{Safe,\ Moderate,\ Danger\} \qquad (1)$$

where N_consecutive is the count of successive readings in the same hazard category, and promotion to a non-Safe confirmed state requires N_consecutive ≥ 5. Fig. 3 shows the complete decision flowchart.

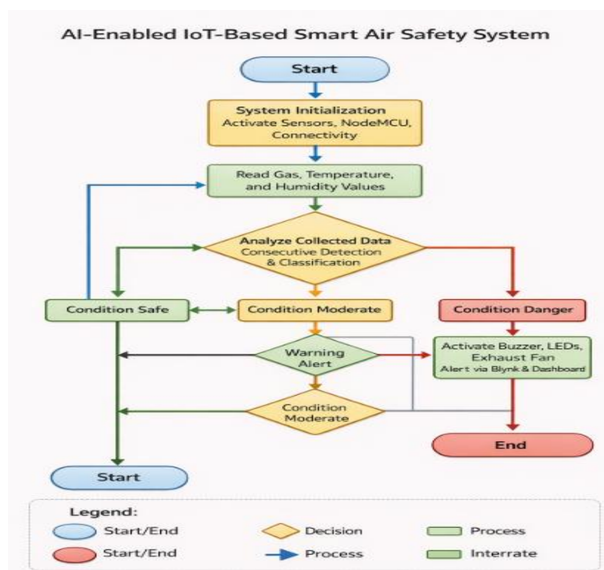

Fig. 3. System flowchart illustrating the consecutive detection-based filtering logic.

E. Actuation Logic

Once a confirmed status is established, the actuation layer responds deterministically: Safe — green LED lit, servo at 90° (fan stopped); Moderate — yellow LED active, servo at 135° (fan at medium speed to begin diluting elevated concentrations); Danger — red LED and buzzer activated simultaneously, servo at 180° (maximum fan speed for strongest ventilation). LED and buzzer updates complete within a single loop iteration, giving an actuation latency of approximately 1 second after confirmation.

IV. Experimental Results and Analysis

A. Test Methodology

The system was tested in three different scenarios using a NodeMCU 1.0 (ESP-12E) board powered by a 5 V regulated power supply. Scenario 1 (Safe): the MQ sensor exposed to normal ambient kitchen air. Scenario 2 (Moderate): a small leak of LPG gas around 50 cm from the sensor for 10 seconds. Scenario 3 (Danger): a continuous LPG leak until the system confirmed and remained in a Danger state for at least 30 seconds. The Arduino IDE Serial Monitor recorded the gas level, temperature, humidity, non-classified and confirmed classification, and a count for each cycle.

B. Observed Results and Device Behaviour

In the Safe scenario, gas ADC values remained consistently in the 73–85 range, temperature at approximately 27.4 °C, and humidity at 44% RH. In the Moderate scenario, readings rose to the 110–130 range; after five consecutive readings in this band, the system promoted the confirmed status to MODERATE, activating the yellow LED and setting the fan to 135°. In the Danger scenario, gas values ranged from 283 to 526; after five consecutive readings above 250, the system confirmed DANGER, triggered the red LED and buzzer, advanced the servo to 180°, and dispatched a Blynk push notification. Fig. 4 shows the physical device during operation, with LED indicators reflecting the active safety state.

Fig. 4. Physical device during operation — green LED indicating Safe condition.

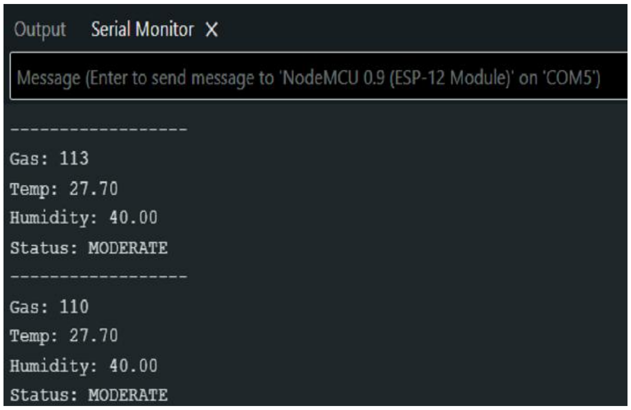

Fig. 5. Output of Serial Monitor from Arduino

This Fig 5 shows the output of serial monitor from arduino, it shows the readings from the sensors. The status includes gas, temp, humidity and system status.

C. Web Dashboard

The web dashboard updated automatically every 2 seconds to reflect the current system status, displaying live gas, temperature, and humidity values along with Chart.js trend graphs. Fig.6 shows the dashboard under a Moderate/Danger condition with elevated gas readings clearly visible in the analytics panel.

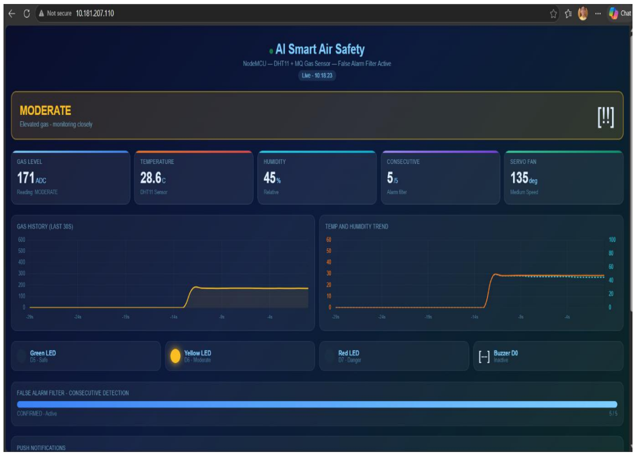

Fig. 6. Web dashboard displaying elevated gas readings under Moderate/Danger condition.

D. Blynk IoT Notifications

Email and mobile notifications from Blynk arrived within 2–3 seconds of the DANGER state transition. Fig. 7 shows the Blynk mobile app interface displaying real-time sensor data and alert status. State-change detection in the firmware ensured notifications were dispatched only on status transitions, preventing notification fatigue. Fig 8 shows blynk console that reflects the reading directly from the sensors and Fig 9 shows Mobile notification received on user phone through blynk app which gives alert to the user when system changes its state.

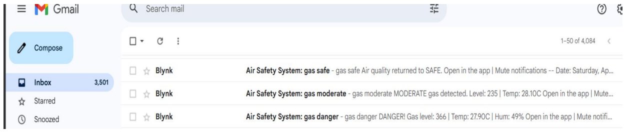

Fig. 7. Blynk IoT mobile app interface showing live sensor readings and alert status.

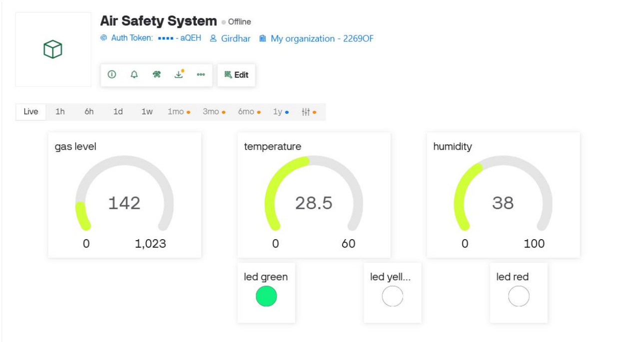

Fig. 8. Blynk Console

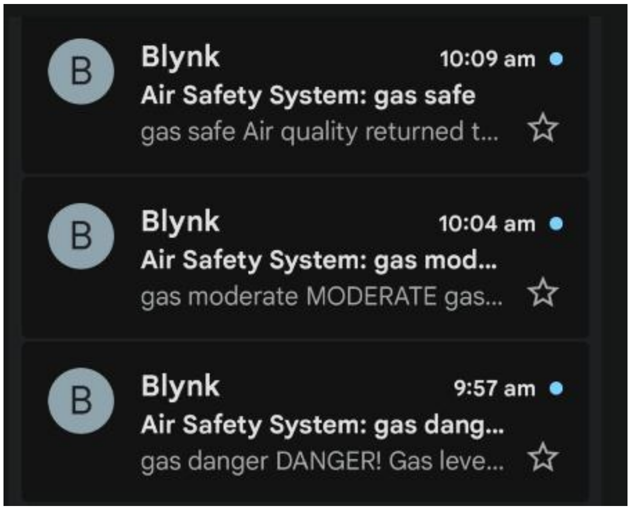

Fig. 9. Mobile notifications from blynk app

E. False Alarm Suppression

Brief sensor spikes of 1–2 readings above GAS_MODERATE_MAX were deliberately induced by momentarily heating the sensor. In every case, the consecutive counter incremented once or twice then reset without triggering a status change, confirming that the 5-reading window effectively suppresses transient false alarms — a meaningful practical improvement over single-threshold approaches.

F. Comparison With Related Work

Table II positions the proposed system against representative prior work across five key dimensions: false-alarm filtering, automated actuation, real-time dashboard, mobile notification, and design consideration for vulnerable users.

V. Challenges and Limitations

We faced several implementation challenges. The gas detection feature is highly sensitive to temperature and humidity and also to the environment changes due to the dependence of the MQ sensor sensitivity on these factors, as well as change of the output resistance. Therefore, temperature correction procedures can be used to get accurate measurements in the future. A stable Wi-Fi signal was important to transmit information via Blynk; sometimes the "no signal" situation led to a delay in notifications, although it did not prevent local control of actuators as it is not dependent on the network. As ESP8266 is a single-threaded system, multiple tasks (reading sensors, web requests, Blynk polling, OLED update) need to be scheduled using yield() and non-blocking libraries. The MQ-2 sensor is a gas detector and cannot select a particular gas. So, it's not selective to LPG and empirical thresholds must be set (170 and 250 ADC). The consecutive filtering method is rule-based and, therefore, cannot adapt to the changing baseline.

VI. Future Scope

There are several improvements that can be implemented in future versions. The implementation of an LPG detector along with the MQ-2 would ensure more accurate detection of gases. Including sensors that detect smoke and flames would increase the scope of detecting fires other than those caused by gas leaks. Logging data in the cloud would allow for analysis of trends over time using statistical techniques. A mobile app that allows users to set custom thresholds for different rooms and control the sensors from their smartphones would enhance the user experience. A mesh network comprising multiple nodes could be considered for larger homes or offices.

VII. Conclusion

The paper presents an AI-driven smart air safety system with the Internet of Things (IoT) technology, which is targeted at the elderly and people with anosmia to keep them safe from household gas leaks. This system is novel as it integrates constant monitoring of the environment with multiple sensors, implementation of a consecutive detection algorithm for false alarm filtering, automated actuation via mechanical means, mobile notifications (powered by Blynk app), and a web dashboard interface on the device. The testing revealed that the system described in this paper does a good three-state classification, is able to filter transient peaks (false alarms) from the sensors and is able to do real-time actuation after status confirmation. By comparing its features with other systems, it was evident that, unlike others, the system proposed in the paper offers all five key features: false alarm filtering, automatic actuation, an analytics web dashboard, mobile notifications, and vulnerable user protection - all for a low-cost microcontroller (only $3).

Acknowledgment

The author thanks Ms. Preeti (Assistant Professor), Dr. Ganesh Kumar Wadhwani (Head of Department), Dr. Manzoor Ansari (Assistant Professor) and Mr. Ashish kumar Nayyar (Training Coordinator) at the Institute of Information Technology & Management, New Delhi, for their guidance and institutional support. The author also acknowledges the open-source contributions of the ESP8266 Arduino community, the Blynk team, and the Adafruit sensor library maintainers.

References

[1] World Health Organization, "Deafblindness and Anosmia: Sensory Impairments in the Elderly," WHO Technical Report, Geneva, Switzerland, 2019.

[2] A. Sharma, R. Gupta, and P. Singh, "IoT-Based Smart Gas Leakage Detection System for Domestic Safety," Int. J. Eng. Research & Technology (IJERT), vol. 12, no. 4, pp. 210–218, 2023.

[3] S. Patel and M. Mehta, "Smart Gas Monitoring System Using IoT and Mobile Applications," Procedia Comput. Sci., vol. 220, pp. 350–357, 2023.

[4] Z. Khan, A. Rauf, and S. Hassan, "Deep Learning-Based Gas Leakage Detection Using IoT Sensor Networks," IEEE Access, vol. 11, pp. 44210–44221, 2023.

[5] R. Yadav, S. Mishra, and P. Tiwari, "IoT-Based Gas Leakage Detection with SMS and Cloud Integration," Int. J. Innovative Research in Sci., Eng. and Technology, vol. 12, no. 6, pp. 7815–7823, 2023.

[6] N. Verma, K. Sharma, and A. Jain, "IoT-Based Smart Home Safety System with Gas and Fire Detection," J. Network and Comput. Appl., vol. 225, pp. 102–115, 2024.

[7] R. Reddy and P. Kumar, "Intelligent Gas Leakage Detection and Alert System Using NodeMCU," in Proc. Int. Conf. on Smart Systems and Inventive Technology (ICSSIT), pp. 45–52, 2024.

[8] M. Gupta and S. Singh, "Smart Kitchen Safety System Using IoT and Automation," Int. J. Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 13, no. 2, pp. 890–897, 2024.

[9] P. Singh, A. Yadav, and R. Mishra, "AI-Based Environmental Monitoring System Using IoT Sensors," Engineering, Technology & Applied Science Research (ETASR), vol. 15, no. 1, pp. 5500–5508, 2025.

[10] M. Gupta, S. Arora, and D. Malhotra, "Real-Time Gas Monitoring System with Web-Based Data Analytics," Expert Systems with Applications, vol. 245, pp. 119–130, 2025.

[11] K. Shah and M. Patel, "Smart Air Quality Monitoring and Alert System Using IoT," Int. J. Advanced Comput. Sci. and Applications, vol. 16, no. 2, pp. 120–128, 2025.