International Journal of Computing Sciences and Technology Studies, 1(1), 20–29, 2026

International Journal of Computing Sciences and Technology Studies

https://journals.hu.edu.so/index.php/ijcsts

IoT-Enabled Smart Fire Detection and Automatic Water Suppression Using ESP32 and Multi-Sensor Sensing

Muhsin Hussein Einte, Luqman Mohamed Mohamud, Aisha Abdulkadir Omar, Fatima Muse Addow, Abdirahman Ali Haji*

Department of Computer Science, Faculty of Computer Science and Information Technology, Hormuud University, Mogadishu, Somalia

*Corresponding author's email: abdulrahman@hu.edu.so

Abstract

Fire safety systems in many residential, educational, and small commercial environments rely on local alarms and manual intervention, which can delay notification and suppression. This work presents the design and implementation of a low-cost Internet of Things (IoT)-enabled fire detection prototype that integrates multi-sensor monitoring, cloud-based notification, and automatic water suppression. An ESP32 microcontroller acquires readings from a KY-026 flame sensor, an MQ-series gas/smoke sensor, and a DHT11 temperature/humidity sensor. Threshold-based logic classifies a fire condition when the smoke value exceeds 1500, the temperature exceeds 45.0 °C, or the flame sensor indicates flame presence. When a fire condition is detected, the controller activates a relay-driven water pump, buzzer, and red status indicator, while the Blynk IoT platform provides remote monitoring and event notifications. A local SH1106 OLED display provides status information independently of the cloud interface, and reconnection routines are implemented for temporary Wi-Fi/Blynk interruptions. The completed prototype was assembled and deployed in a representative indoor environment at Hormuud University. Functional tests confirmed successful operation of multi-sensor detection, automatic pump activation, mobile monitoring, notification, and connectivity recovery. Quantitative detection-accuracy, false-alarm, and response-time measurements were not conducted in this study and are identified as future work. The resulting prototype demonstrates the feasibility of an affordable detection-notification-suppression pipeline for the Somali context and provides a basis for future multi-zone, adaptive, and standards-compliant development.

Keywords: Internet of Things, Fire detection, ESP32, Multi-sensor sensing, Blynk, Embedded systems.

1. Introduction

Fire incidents can cause substantial loss of life and property, while the effectiveness of a firesafety system depends strongly on how quickly an abnormal condition is detected and acted upon. Conventional systems commonly provide local alarm functions, but they may depend on occupants or other personnel to recognize the alarm and initiate further action. Delayed

response and limited automation are important practical concerns in the Somali context, particularly where advanced commercial systems may be less accessible.

The Internet of Things (IoT) provides a mechanism for connecting physical sensors to computational and communication services. IoT systems can collect environmental measurements, process them locally, transmit selected information to cloud services, and expose the resulting state through user interfaces. Prior IoT literature has established the relevance of connected sensing and remote monitoring to smart environments [1], [2]. For firesafety applications, integrating multiple sensing modalities can also provide complementary evidence about environmental conditions, while automatic actuation can reduce dependence on human response.

The project therefore focuses on a practical embedded architecture that combines flame, smoke/gas, and temperature sensing with an ESP32 controller, Wi-Fi connectivity, Blynkbased remote monitoring, local status display, audible/visual alarms, and a relay-controlled water pump. The work is intended primarily as a prototype for residential and small commercial settings and as a locally relevant demonstration for environments such as offices, classrooms, and laboratories at Hormuud University.

There is a local gap: limited published work specifically addresses IoT-enabled fire safety in Somalia, and little locally reported performance data exist for comparable systems. Conventional fire detection in Mogadishu is also heavily dependent on standalone alarms, highlighting the need for affordable, scalable, and user-centered solutions. These claims motivate the engineering contribution of this prototype, while the absence of a large quantitative field dataset limits the strength of any performance generalization.

The main contributions of the work are as follows: 1) a low-cost embedded fire-safety architecture based on ESP32 and heterogeneous sensors; 2) an integrated detectionnotification-suppression pipeline combining local actuation with Blynk cloud monitoring; 3) edge-triggered alerting, local OLED status, and reconnection logic intended to improve operational continuity; and 4) a functional prototype deployed in a representative indoor environment, providing a basis for subsequent multi-zone and long-term validation.

2. Related Work

IoT architectures typically combine a physical sensing layer, a communication layer, and an application or service layer. The literature emphasizes the use of networked sensors, cloud services, and communication protocols to support remote monitoring and data management [3]. In smart-building contexts, connected safety systems can also interact with other building services, enabling coordinated responses such as alerting occupants and controlling building infrastructure [4].

Fire detection technologies include smoke, heat, and flame detection mechanisms. Fireprotection literature distinguishes sensor behavior according to the physical characteristics of fires and emphasizes appropriate sensor placement, maintenance, and suppression design [5]. This work applies this principle by combining three complementary indicators rather than relying on a single measurement. The resulting decision rule is intentionally simple: a fire state is asserted when any configured fire indicator crosses its threshold. This OR-based strategy favors sensitivity and reduces the risk that a single failed or insensitive sensing modality will prevent actuation, although it also creates a need for careful threshold calibration to control false alarms.

Wireless sensor networks and IoT safety systems also require attention to reliability, authentication, communication continuity, and energy management [6]. The proposed prototype addresses only part of this broader problem. It provides automatic Wi-Fi/Blynk reconnection and retains local sensing, display, and actuation functions on the ESP32, but it does not implement a redundant communications network or a dedicated backup communication channel.

Water-based suppression is another critical component. The reviewed fire-suppression literature discusses sprinkler selection, water supply, electronic valves, maintenance, and backup activation [7]. The prototype uses a relay-controlled pump rather than a certified building sprinkler installation. Consequently, the system should be interpreted as an academic proof of concept rather than a replacement for code-compliant commercial fire-protection equipment.

This work also applies the Technology Acceptance Model conceptually, emphasizing perceived usefulness and ease of use as factors that can influence adoption [8]. In the implemented system, these considerations are reflected in automatic operation, simple mobile monitoring, clear status indicators, and manual pump control. However, although the methodology proposes questionnaires and user feedback, no quantitative acceptance analysis was conducted; therefore, user-acceptance claims are treated as design considerations rather than experimentally established outcomes.

3. System Design and Methodology

3.1. Research and Development Approach

The project follows an applied design-and-development approach with iterative development and experimental validation. The development process is organized around requirement identification, architecture design, prototype construction, testing and validation, and deployment. An iterative and incremental model was followed, in which basic detection is implemented first and additional capabilities—automatic suppression, cloud monitoring, notifications, and logging—are integrated progressively.

3.2. IoT Architecture

The system follows a three-layer IoT architecture. The perception layer contains the flame, smoke/gas, and temperature/humidity sensors and the water-pump actuator. The network layer uses Wi-Fi to connect the ESP32 to the Blynk cloud service. The application layer provides threshold evaluation, alarm and pump control, local OLED status, mobile monitoring, and cloud event logging. Fig. 1 summarizes the implemented architecture.

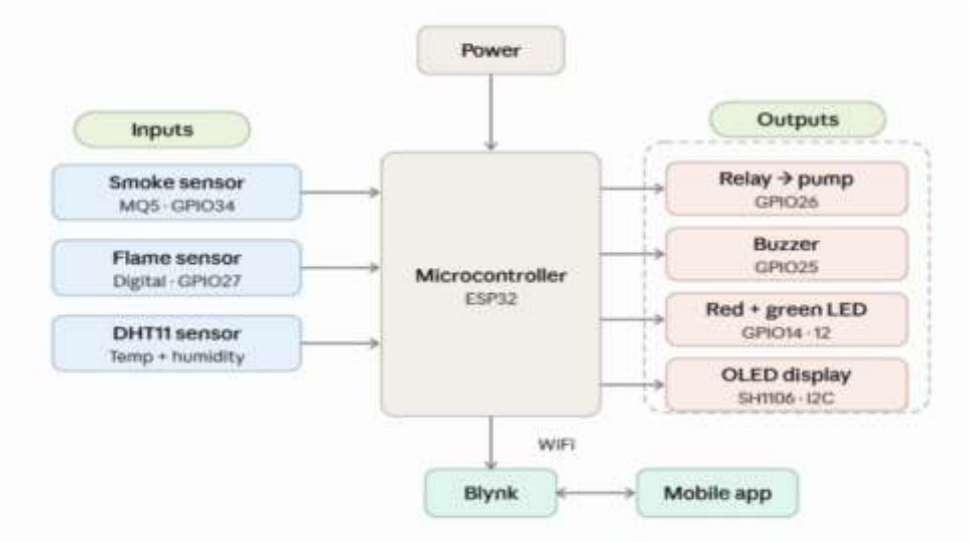


Fig. 1. Overall architecture of the implemented smart fire detection and suppression prototype.

3.3. Hardware Components

Table 1. Hardware Components and Functions

Component	Role	Function in prototype

ESP32	Main controller with WiFi/Bluetooth	Sensor processing, decision logic, actuation, cloud communication
KY-026 flame sensor	Infrared flame detector	Digital/analog flame indication
MQ-series gas/smoke sensor	Combustible gas and smoke sensing	Analog smoke/gas measurement
DHT11	Temperature and humidity sensor	Environmental monitoring
Relay module	Electrical switch	Interfaces ESP32 control with pump
Water pump	Actuator	Water delivery for suppression prototype
Buzzer + LEDs	Local indicators	Audible and visual alarm/status
SH1106 OLED	128×64 I2C display	Local system status
Power, wiring, breadboard/PCB	Supporting hardware	Prototype assembly and power

As listed in Table 1, an ESP32 is used as the central controller because of its integrated Wi-Fi capability, GPIO availability, processing capability, and low cost. The flame sensor provides flame presence information, the MQ-series sensor provides an analog smoke/gas signal, and the DHT11 provides temperature and humidity measurements. A relay isolates the controllerside logic from the pump-driving circuit. The physical assembly connects these components through designated ESP32 GPIO pins, with the OLED connected through the I2C bus.

3.4. Data Acquisition and Fire Decision Logic

Sensor data are acquired continuously by the firmware. The implementation uses a two-second timer for sensor acquisition and cloud synchronization. The smoke/gas value is compared with a threshold of 1500, the temperature is compared with 45.0 °C, and the flame sensor is interpreted as indicating flame when its digital state is LOW. The fire state is therefore defined as:

Fire = (Smoke > 1500) OR (Temperature > 45.0 °C) OR (Flame = present).

When the fire state is true, the firmware activates the relay-controlled pump, buzzer, and red LED, turns off the green LED, updates the OLED, and synchronizes the state with Blynk. When no fire is present and no manual override is active, the pump is turned off and the green LED indicates the normal state. The edge-triggered alert mechanism compares the current state with the previous state and sends a cloud event only when the system transitions into a fire condition, reducing repeated notifications.

3.5. Cloud Data Structure and User Interface

Table 2. Blynk Virtual-Pin Data Structure

Virtual Pin	**Type**	**Meaning**
V0	Float	Temperature (°C)
V1	Float	Relative humidity (%)
V2	Integer	Smoke/gas analog value
V4	String	Flame status
V5	String	Overall system status

V6	Integer	Manual pump control (0/1)
V7	Integer	Pump-status LED value (0–255)

Table 2 lists the Blynk virtual pins. Blynk functions as the cloud data and user-interface layer rather than a conventional relational database. Virtual pins carry sensor and status values between the ESP32 firmware and the mobile dashboard. Fire events are additionally recorded using Blynk event logging. The mobile interface (Fig. 2) provides live temperature and humidity values, smoke trends, flame and overall status labels, a manual pump switch, and a pump-status indicator.



Fig. 2. Blynk mobile dashboard used for real-time monitoring and manual pump control.

3.6. Connectivity Resilience and Local Operation

The firmware contains a connection-check routine that periodically monitors Wi-Fi and Blynk states and attempts reconnection when a connection is lost. The BLYNK_CONNECTED callback also re-synchronizes the manual pump switch state. Importantly, fire sensing and local actuation remain on the ESP32, while the OLED provides local visibility when cloud connectivity is unavailable. This architecture reduces the dependence of immediate local response on the availability of the remote dashboard, although remote notification necessarily remains connectivity-dependent.

4. Implementation and Validation

4.1. Firmware and Hardware Integration

The firmware was developed in C/C++ using the Arduino IDE and relevant libraries for Blynk, DHT sensing, and the SH1106 OLED. The control program initializes the display and sensors, connects the ESP32 to Wi-Fi and Blynk, periodically reads sensor values, evaluates the fire rule, controls the actuators, updates the cloud dashboard, and handles connectivity recovery. The implementation is organized as a non-blocking control loop using timer-driven tasks. The prototype was physically assembled by connecting the flame sensor, MQ-series gas sensor, and DHT11 to the ESP32, while the relay, buzzer, LEDs, and OLED were connected to the designated output and I2C interfaces. The assembled hardware is shown in Fig. 3, and the circuit diagram in Fig. 4.

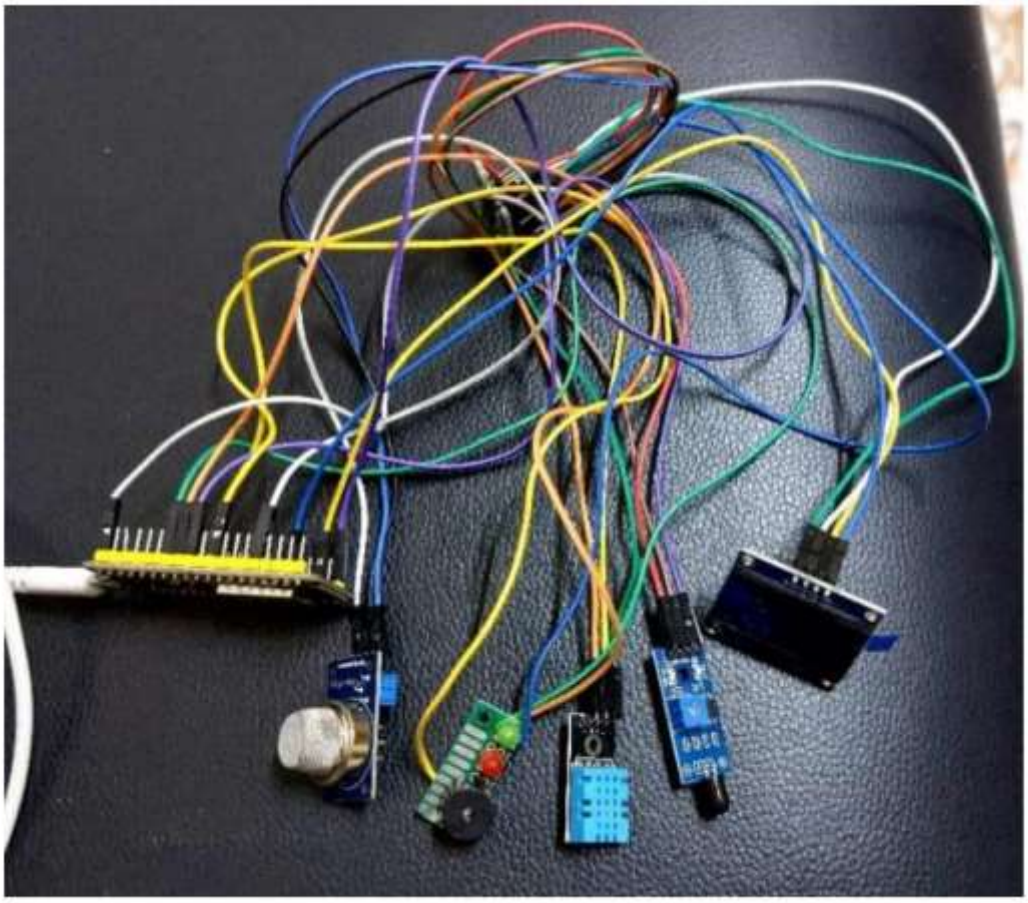


Fig. 3. Assembled hardware prototype.

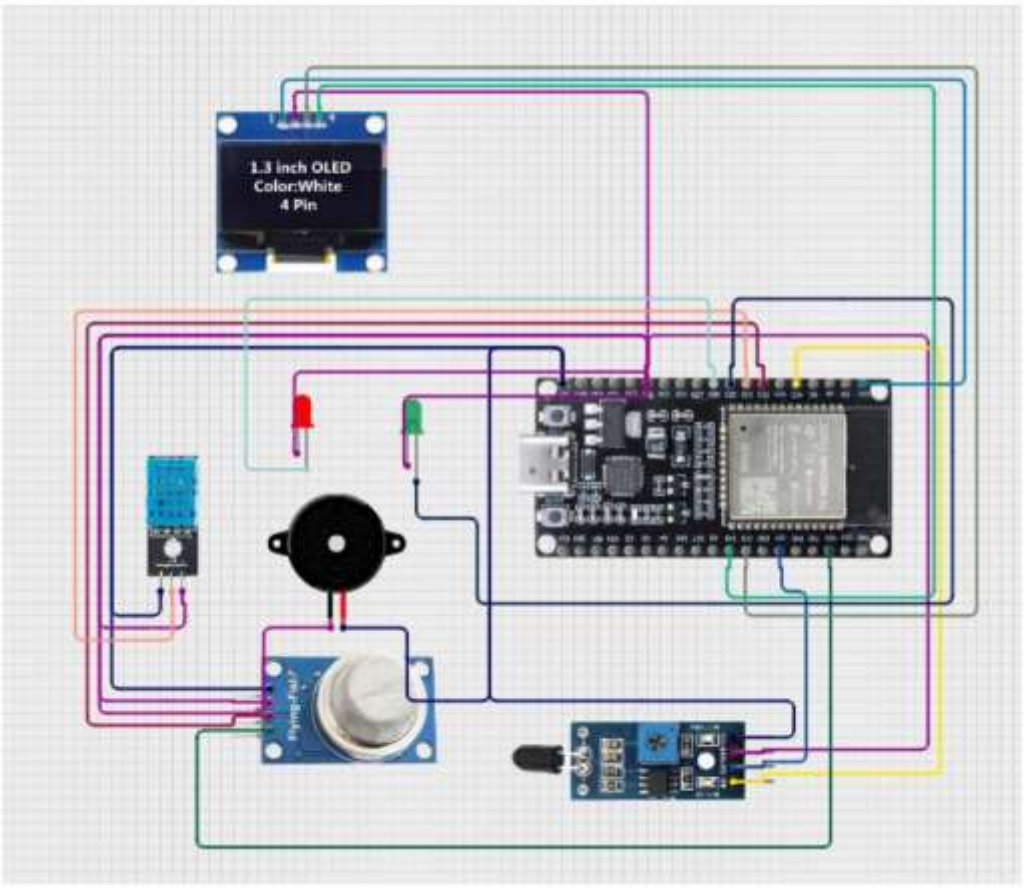


Fig. 4. Circuit/wiring diagram of the smart fire detection prototype.

4.2. Functional Validation

The methodology specifies controlled tests involving small flames, smoke, temperature changes, combined fire indicators, and false-alarm conditions. It also defines detection accuracy, response time, false-alarm rate, notification delivery time, and operational reliability as evaluation measures. However, numerical results for these metrics were not recorded in this study; the evidence reported here is therefore primarily functional and deployment-oriented (Table 3).

Table 3. Functional Validation Outcomes

Validation item	Evidence	Reported outcome
Multi-sensor detection	Flame, smoke/gas, and temperature sensors integrated with ESP32	Functionally implemented; operated successfully
Automatic suppression	Relay-controlled pump activated by fire state	Functionally implemented; operated successfully
Local alarm/status	Buzzer, red/green LEDs, and OLED display	Functionally implemented; operated successfully
Remote monitoring	Blynk dashboard with live sensor/status data	Functionally implemented; operated successfully
Fire notification	Blynk fire-alert event and push notification	Functionally implemented; operated successfully

Manual override	Mobile pump control through V6	Implemented; operated as a reliable fallback
Connectivity recovery	Automatic Wi-Fi/Blynk reconnection	Implemented; functional

The prototype was subsequently deployed in a representative indoor environment at Hormuud University. The OLED display, mobile dashboard, alert mechanisms, and manual override operated consistently during deployment. The deployment setup is illustrated in Fig. 5. These observations support the feasibility of the integrated architecture, but they should not be interpreted as statistically validated field performance.



Fig. 5. Representative deployment setup of the prototype at Hormuud University.

5. Results

The principal result of the project is a functioning end-to-end prototype that integrates sensing, local decision-making, automatic suppression, cloud monitoring, and user notification. The implementation demonstrates that a single ESP32 can coordinate heterogeneous sensors, local indicators, a pump actuator, an OLED display, and a cloud dashboard within the proposed architecture.

The system successfully implements a detection-to-response sequence: environmental sensing is performed locally; a threshold violation produces a fire state; the local pump, buzzer, and visual indicators are activated; the OLED reports the state; and Blynk receives the updated measurements and fire event. The edge-triggered notification mechanism avoids repeated alerts while the fire state remains continuously active.

The deployment evidence also demonstrates that the system can remain locally informative during temporary network interruptions because the core sensing, decision, and actuation functions are executed on the ESP32. Cloud synchronization is restored through the reconnection routine. The manual pump control provides an additional user-initiated actuation path through the mobile application.

An important limitation is that the proposed accuracy, false-alarm, response-time, notification-latency, and reliability measurements were not quantified. Consequently, this paper reports only functional outcomes and does not claim an experimentally measured detection accuracy or response time. The expectation that the system can reduce response time to under one minute is a design target rather than a measured result.

6. Discussion

The implemented architecture addresses the central limitation identified by the project: a fire-safety system should not depend entirely on a person being present, recognizing a local alarm, and manually initiating suppression. By placing the decision logic and pump actuation at the edge, the prototype can react locally to sensor conditions while using IoT connectivity for remote visibility and notification.

The use of heterogeneous sensors is a practical design choice. Flame detection can provide direct evidence of combustion, smoke/gas sensing can indicate combustion products or combustible gases, and temperature sensing can capture abnormal thermal conditions. The ORbased rule increases the opportunity for any one indicator to initiate an alert, but it also means that environmental phenomena such as smoke, steam, or elevated temperature can trigger the system if thresholds are not appropriately calibrated. This threshold sensitivity is acknowledged as a limitation.

The cloud interface improves observability by exposing sensor values and system status through a mobile dashboard. The local OLED complements this interface by maintaining visibility without relying on the internet. This combination is particularly appropriate for an IoT prototype in which connectivity can be intermittent. Nevertheless, a production safety system would require stronger fault tolerance, backup communication, power continuity, pump-failure detection, water-reserve monitoring, and compliance with applicable fire-safety standards.

The economic argument is also relevant to the intended context. The estimated hardware cost is approximately USD 130–184, compared with commercial fire-alarm installations estimated at USD 500–3,000, excluding additional sprinkler costs. These figures support the feasibility of a low-cost prototype, although they should be treated as project estimates rather than a comprehensive lifecycle cost analysis.

7. Conclusion

This paper presented an IoT-enabled smart fire detection and automatic water suppression prototype based on an ESP32 and multi-sensor environmental monitoring. The implemented system integrates flame, smoke/gas, and temperature sensing with local threshold-based decision-making, automatic pump actuation, audible and visual alarms, OLED status display, Blynk cloud monitoring, mobile notification, and connectivity-recovery routines.

The prototype was successfully integrated and deployed in a representative indoor environment at Hormuud University. Its main contribution is an affordable end-to-end detection-notification-suppression architecture tailored to the practical constraints described for the Somali context. The prototype is therefore a useful proof of concept and educational reference design. At the same time, the absence of reported quantitative performance measurements, single-zone coverage, fixed thresholds, water-supply dependence, limited field duration, and lack of formal certification prevent the system from being characterized as a validated commercial fire-safety solution. Addressing these limitations through multi-node deployment, adaptive calibration, redundant communication, backup power and water, longerterm trials, and standards-based certification represents the most important path toward future development.

7.1. Limitations

The prototype monitors a single location using one sensor set and therefore does not provide native multi-zone building coverage. Its fixed thresholds—particularly smoke above 1500 and temperature above 45.0 °C—require site-specific calibration. Further limitations are the limited duration of field testing, the absence of formal regulatory certification, and dependence on an adequate water supply. These factors constrain generalization from the prototype to real-world commercial deployment.

A further limitation for scientific evaluation is the lack of quantitative experimental results. Although the methodology defines performance metrics and test scenarios, the corresponding numerical measurements were not collected. Future evaluation should therefore collect repeated measurements of true-positive detection, false positives, response latency, notification latency, pump activation, and connectivity recovery under controlled and real-world conditions.

7.2. Future Work

Future work should extend the architecture to multiple sensor nodes connected to a central hub so that several rooms or an entire building can be monitored. Adaptive threshold calibration using statistical or machine-learning methods could also reduce false alarms by accounting for environmental variation. Backup power, a dedicated water reservoir, and pump-failure detection would improve resilience during power or water interruptions. Alternative communication mechanisms could improve notification availability when internet connectivity is lost. Finally, extended field trials and formal testing against recognized fire-safety requirements are necessary before any transition from academic prototype to deployable safety product.

Conflict of Interest

The authors declare no conflict of interest.

References

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

[2] S. D. T. Kelly, N. K. Suryadevara, and S. C. Mukhopadhyay, “Towards the implementation of IoT for environmental condition monitoring in homes,” IEEE Sensors Journal, vol. 13, no. 10, pp. 3846–3853, Oct. 2013.

[3] R. Buyya and A. V. Dastjerdi, Internet of Things: Principles and Paradigms. Cambridge, MA, USA: Morgan Kaufmann, 2016.

[4] A. H. Buckman, M. Mayfield, and S. B. Beck, “What is a smart building?” Smart and Sustainable Built Environment, vol. 3, no. 2, pp. 92–109, 2014.

[5] A. E. Cote, Ed., Fire Protection Handbook, 20th ed. Quincy, MA, USA: National Fire Protection Association, 2008.

[6] I. F. Akyildiz and M. C. Vuran, Wireless Sensor Networks. Hoboken, NJ, USA: Wiley, 2010.

[7] C. Grant and D. Lowe, Fire Suppression Systems: Design and Installation. London, U.K.: Routledge, 2015.

[8] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology,” MIS Quarterly, vol. 13, no. 3, pp. 319–340, 1989.