IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, June 2025
Impact Factor: 7.67

IoT Based Disaster Alert System

Abhishek Mane[1], Tatyaso Dhanawade[2], Prathamesh Suryawanshi[3] ,Rushikesh Nikam[4]
Students, Electronics & Telecommunication
Assistant Professor, Electronics & Telecommunication
Padmabhooshan Vasantraodada Patil Institute of Technology, Sangli, India

Abstract: *In today's world, ensuring personal safety has become a growing concern due to increasing threats from accidents, crimes, and environmental hazards. This project proposes the development of a Human Safety Device designed to provide real- time monitoring, emergency alerting, and location tracking capabilities. The device integrates various sensors such as GPS, accelerometers, temperature sensors, and panic buttons to detect abnormal conditions like sudden falls, high body temperature, or distress situations. When triggered, it sends instant alerts via SMS or mobile app notifications to pre-defined contacts and emergency services, along with the user's real-time location. Compact, wearable, and energy-efficient, this device aims to offer a reliable safety solution for children, women, elderly individuals, and workers in hazardous environments. The ultimate goal is to enhance personal security and enable quicker emergency response, potentially saving lives.*

Keywords: IoT, Disaster Alert System, Smart Disaster Management, Real-time Alert System

I. INTRODUCTION

Natural disasters—such as floods, cyclones, heatwaves, and other extreme weather phenomena—have grown more frequent and destructive in recent decades. This escalation is largely attributed to rapid urbanization, unchecked environmental degradation, and the global impact of climate change. These disasters not only endanger human life and destroy infrastructure. Traditionally, disaster management systems have relied on manual observation and reactive strategies. To address these challenges, there's an urgent need for smart, real-time, and automated systems that can provide early warnings and support rapid decision-making. An ideal disaster alert system should not only detect potential hazards but also disseminate this information effectively to responsible agencies and the general public. Recent advances in the Internet of Things (IoT) provide a powerful framework for achieving this.

The IoT paradigm connects everyday objects to the internet, enabling them to collect and share data. In the context of disaster management, IoT allows a network of smart sensors to monitor environmental parameters such as temperature, humidity. In addition to real-time data collection, alert mechanisms such as SMS notifications using Twilio and visual cues via a web-based dashboard are employed to warn citizens and emergency personnel about imminent threats. This dual approach— automated data collection and instant communication—bridges the critical gap between hazard detection and response. This project, titled "IoT Based Disaster Alert System", proposes the development of a robust and modular solution for natural disaster detection and early warning.

II. LITERATURE REVIEW

The application of Internet of Things (IoT) in disaster management has gained significant attention in recent years due to its ability to collect, transmit, and process real-time environmental data. This section reviews existing research works and systems that have inspired or contributed to the development of the IoT-Based Disaster Alert System in this project.

2.1 Disaster Monitoring Using IoT

A paper published in the International Journal of Scientific Research in Computer Science emphasizes the use of environmental sensors like temperature, humidity, wind speed, and rainfall sensors for real-time disaster detection. The collected data is transmitted to a cloud platform for monitoring and alert generation. This approach supports the use of microcontrollers like ESP32 or Arduino along with cloud databases such as Firebase. The methodology aligns closely

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, June 2025
Impact Factor: 7.67

with our project, which uses DHT11, anemometers, and rain sensors connected to an ATmega328P microcontroller to collect real-time data and process it using a Python backend.

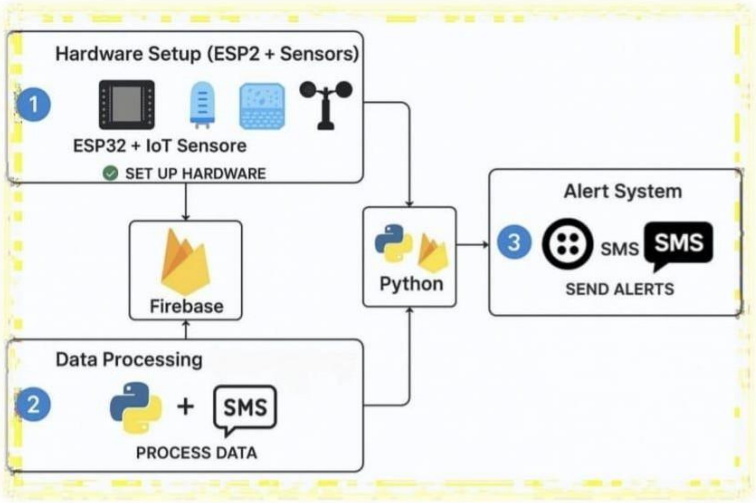
2.2 IoT-Based Flood Monitoring and Alerting System Published in IEEE Xplore, this research work introduces a flood alert system using water level and rain sensors integrated with ESP8266 or ESP32 modules. The system transmits data to cloud platforms like ThingSpeak or Firebase and triggers alerts using Twilio or mobile notifications. The use of Twilio API for sending SMS in this study closely resembles our project's communication strategy. Furthermore, it highlights the importance of early warning systems in reducing flood damage, which supports our system's inclusion of soil moisture sensors for detecting floodprone conditions.

2.3 Real-Time Cyclone Alert System Using IoT A study in the International Journal of Advanced Research in Computer and Communication Engineering presents a cyclone alert system using anemometers and pressure sensors. Firebase is used for data storage, and the system includes a web-based dashboard with location mapping through the Google Maps API. This closely relates to our project's architecture, where sensor data is pushed to Firebase and visualized on a web dashboard, making the data accessible to authorities and users in real-time.

2.4 IoT Framework for Smart Disaster Management in Cities Published in Springer's Smart Cities Journal, this paper proposes a complete IoT framework for urban disaster management, combining real-time dashboards, cloud data storage, and automated SMS alerts. It emphasizes the need for a modular and scalable system architecture, which directly supports our system design that allows easy sensor integration, real-time data processing, and cloud-based alerting.

III. WORKING AND BLOCK DIAGRAM

Working Explanation of IoT-Based Disaster Alert System



The working of the IoT-based disaster alert system is based on the real-time sensing , transmission, analysis, and alerting of environmental parameters such as temperature, humidity, wind speed, rainfall, and ground vibration. The system is designed to function in multiple stages to ensure timely and accurate disaster detection.

Step-by-Step Working Process

1. Data Collection (Sensing Layer)

Multiple sensors are used to collect environmental data:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 8, June 2025 Impact Factor: 7.67

DHT11: Measures temperature and humidity. BMP280: Measures atmospheric pressure.
Rain Sensor: Detects rainfall intensity. Anemometer: Measures wind speed.
Piezoelectric Sensor: Detects ground vibration or shock.
These sensors are physically connected to a microcontroller (ATMega), which reads their data at regular intervals.

2. Data Processing (Edge Device Logic)

The ESP32 microcontroller processes raw sensor data. Predefined threshold values are used to identify hazardous conditions. For example:

Temperature > 45° C → possible heatwave. Wind speed > 80 km/h → cyclone alert.

Sudden vibration → earthquake warning.

If sensor readings exceed these thresholds, the ESP32 flags a disaster-prone event.

3. Threshold Evaluation & Alert Trigger (Backend Processing)

- A Python-based backend (e.g., FastAPI or Flask) constantly monitors the Firebase database.
- The script evaluates the sensor readings against safe threshold levels.
- If a threshold is breached, the backend automatically triggers alerts.

4. Alert Mechanism (Communication Layer)

- When a disaster condition is detected:
- The system sends SMS alerts via Twilio.
- Optionally, it can send emails or mobile push notifications.
- Alerts include the type of disaster, current sensor reading, and location details

IV. SOFTWARE DESIGN AND IMPLEMENTATION

System Architecture Overview

The software system of the IoT-based Disaster Alert System is designed to ensure realtime environmental data acquisition, cloud-based storage, automated threshold checking, and instant alert delivery. The software stack involves Arduino IDE (for ESP32 programming), Firebase (cloud database), Python (for backend logic and alert handling), Twilio API (for SMS alerts) 4.2 System Block Diagram [Environmental Sensors] -> [ESP32 Microcontroller with Arduino Code] -> Wi-Fi -> [Firebase Real-Time Database] -> [Python Backend Script (FastAPI/Flask)] -> [Twilio API (SMS)]

Software Modules

A. Sensor Data Acquisition (Arduino IDE) Platform: Arduino IDE Language: C/C++ Functionality:- Reads data from DHT22, BMP280, Rain sensor, and Anemometer.- Formats the sensor data into a JSON object.- Pushes data to Firebase using HTTP or WebSocket libraries. Sample Code Snippet: #include #include #include "DHT.h" #define DHTPIN 4 #define DHTTYPE DHT22 DHT dht(DHTPIN, DHTTYPE); Software Design and Implementation void loop() { float t = dht.readTemperature(); float h = dht.readHumidity(); Firebase.setFloat("/sensor/temperature", t); Firebase.setFloat("/sensor/humidity", h); delay(2000); }

B. Cloud Integration (Firebase) Tool: Google Firebase Real-Time Database Usage:- Stores sensor readings in real time.- Allows read/write access via REST API or Firebase SDKs. Database Structure: { } "sensor": { "temperature": 34.2, "humidity": 65.5, "wind_speed": 12.4, "pressure": 1011, "rain": 0 }, "alerts": { "status": "normal" }

C. Backend Logic (Python + Flask/FastAPI) Platform: Python 3.x Framework: FastAPI or Flask Functionality:- Continuously fetches data from Firebase.- Compares current readings to predefined thresholds. Software Design and Implementation- Sends alerts via Twilio and logs to Firebase. Sample Logic: if data["temperature"] > 45 or data["humidity"] < 15: send_sms("Heatwave Alert! Take precautions.") firebase.put("/alerts", "status", "Heatwave")

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, June 2025
Impact Factor: 7.67

D. Alert System (Twilio API) Service: Twilio Messaging API Purpose: Sends real- time SMS alerts to registered numbers. Example SMS: ALERT: Wind Speed exceeds safe limit! Possible storm condition.

Threshold Management and Flexibility

All thresholds used to trigger alerts (e.g., temperature > 45°C, humidity < 20%, wind speed > 60 km/h) are stored in Firebase for real-time modification. Benefits:-

Region-specific customization- Threshold tuning during testing- Easy modification from dashboard

Testing and Validation

Simulated weather data tested system responses.- Response time verified from sensor trigger to SMS alert.- Firebase log used to validate data integrity and frequency. Software Design and Implementation- System reliability ensured through multiple test cases.

Summary

The software part integrates sensor acquisition, cloud data handling, automated alerting, and live monitoring. It ensures:- Fast, reliable disaster detection.- Scalable and remotely manageable setup.- Compatibility with additional sensors or future ML enhancements. It serves as a smart, real-time disaster warning platform for both rural and urban environments.

V. ADVANTAGES

The integration of the Internet of Things (IoT) into disaster alert and monitoring systems introduces a revolutionary approach to managing natural and man-made disasters. The IoT-based disaster alert system leverages real-time sensor data, cloud connectivity, and instant communication to enable proactive disaster prevention, detection, and response.

Real-Time Monitoring

• Continuously collects data from multiple sensors (e.g., temperature, humidity, wind, rain, vibration). • Ensures up- to-date status of environmental conditions. • Facilitates timely decision-making for authorities and responders.

Example: Rising temperature and low humidity data from the DHT22 can be used to predict forest fires in real-time.

Early Warning Capability

• Sends automatic alerts (SMS, email, app notification) before the disaster fully strikes. • Prevents loss of life by giving people time to evacuate or take precautions. • Helps in avoiding infrastructure damage by enabling pre-disaster safety measures.

Example: If wind speed from the anemometer crosses a safe threshold, the system can trigger a cyclone warning.

24/7 Remote Accessibility

• Cloud-based platforms (like Firebase) enable access to data from anywhere in the world. • Beneficial for monitoring remote and rural areas where on-site observation is difficult. • Web dashboards provide real-time visualizations for administrators and users.

Example: A city authority can monitor rainfall data from a different location using just a smartphone or PC.

Automated Alert System

• Eliminates the need for manual monitoring and human error. • Uses services like Twilio or Firebase Cloud Messaging for instant SMS and push notifications. • Alerts are generated based on pre-programmed threshold conditions.

Example: In case of sudden vibration spikes (earthquake), the system sends automatic alerts to registered users.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 8, June 2025
Impact Factor: 7.67

VI. CONCLUSION

The IoT-Based Disaster Alert System developed in this project provides a practical, scalable, and cost-effective solution for real-time environmental monitoring and early disaster warning. By integrating multiple sensors—including DHT11 for temperature and humidity, BMP280 for atmospheric pressure, rain sensor, anemometer for wind speed, soil moisture sensor, and piezoelectric sensor for vibration—this system continuously collects and transmits vital environmental data to a central microcontroller (ATmega328P). The microcontroller processes the sensor data and communicates with a Pythonbased backend, which evaluates the data against predefined disaster thresholds. The system uses Firebase for real-time data storage and visualization, and Twilio API for sending automated SMS alerts when critical values are detected. This ensures timely warnings are delivered to users, helping them take preventive action and minimize damage to life and property.

In conclusion, the IoT-Based Disaster Alert System is a step toward smart, technology-driven disaster management. It empowers individuals, communities, and authorities with accurate, real-time information, paving the way for safer and more resilient environments

VII. ACKNOWLEDGMENT

The authors would like to express their heartfelt gratitude to the faculty members of the Electronics & Telecommunication Engineering, PVPIT College Budhgaon, for their invaluable guidance and academic support throughout the course of this research. We are especially thankful to our project supervisor for their expert advice, timely feedback, and encouragement, which played a crucial role in shaping the direction and outcome of this work. We also acknowledge the contributions of the technical staff and laboratory team for providing the resources and environment necessary for successful experimentation and implementation. This research would not have been possible without the support and infrastructure offered by our institution. .

REFERENCES

[1] P. Rajalakshmi, G. R. Karpagam, and S. Vignesh, ―IoT based early flood detection and avoidance system, ‖ in Proc. IEEE Int. Conf. on Intelligent Computing and Control Systems (ICICCS), Madurai, India, Jun. 2018, pp. 1125 – 1129. [Online]. Available: https://ieeexplore.ieee.org/document/8529251

[2] IoT based real-time flood monitoring and alert management system, ‖ International Journal of Engineering Research & Technology (IJERT), vol. 9, no. 5, pp. 765 – 768, May 2020. [Online]. Available: https://www.ijert.org/iot-based-real- time-flood-monitoring- and-alert-management-system

[3] K. T. Patil, R. R. Deore, and M. S. Jadhav, ―Design and implementation of IoT based weather monitoring and alert system, ‖ International Research Journal of Engineering and Technology (IRJET), vol. 8, no. 6, pp. 1404 – 1408, Jun. 2021. [Online]. Available: https://www.irjet.net/archives/V8/i6/IRJET-V8I6900.pdf

[4] ―Smart disaster management system using IoT and machine learning, ‖ in Proc. IEEE Int. Conf. on Artificial Intelligence in Engineering and Technology (IICAIET), Kota Kinabalu, Malaysia, Sep. 2022, pp. 1 – 6. [Online]. Available: https://ieeexplore.ieee.org/document/9800369