iJRASET
International Journal For Research in Applied Science and Engineering Technology



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 14 **Issue:** IV **Month of publication:** April 2026

**DOI:** https://doi.org/10.22214/ijraset.2026.79549

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue IV Apr 2026- Available at www.ijraset.com*

# IoT-Based Smart Shelter for Flood and Earthquake

Mr. Raj Dere[1], Mr. Atharva Sakunde[2], Prof. Aasifa Bagwan[3]
[1, 2, 3]*Department of Computer Engineering, KBP Polytechnic, Satara, India*

***Abstract:*** ***Natural disasters such as floods and earthquakes pose a severe threat to human life and infrastructure, often leaving little time for people to react. Most existing disaster management systems are limited to monitoring and alerting functions, without providing any immediate physical protection. This paper presents an IoT-Based Smart Shelter System integrated with the GeoSafe mobile application, designed to detect floods and earthquakes in real time and respond automatically. The system employs an ESP32 microcontroller connected to a raindrop sensor and a vibration sensor (MPU6050) for continuous environmental monitoring. Upon disaster detection, the system activates a buzzer alert, triggers a relay module to deploy an inflatable shelter, and simultaneously sends real-time notifications through the GeoSafe Android application. The application provides disaster alerts, weather information, global disaster maps, GPS-based shelter location tracking, emergency helplines, and safety guidelines. Firebase is used as the cloud database for real-time data synchronization. The proposed system bridges the critical gap between disaster detection and immediate physical protection, making it a comprehensive, automated, and low-cost disaster management solution suitable for both urban and rural environments.***

***Keywords: IoT, ESP32, Smart Shelter, GeoSafe App, Flood Detection, Earthquake Detection, Firebase, GPS, Disaster Management, Inflatable Shelter, Real-Time Alert.***

## I. INTRODUCTION

Natural disasters such as floods and earthquakes are among the most devastating events that affect human life and infrastructure worldwide. These disasters occur suddenly and often leave very little time for people to take protective action. In many disaster-prone regions, especially in rural and remote areas, access to proper warning systems and emergency infrastructure is extremely limited. Existing disaster management systems primarily focus on detection and alerting — they notify users about an impending or occurring disaster, but do not provide any immediate physical protection. The gap between receiving an alert and finding actual shelter can be life-threatening. Furthermore, many systems operate in isolation, lacking integration between hardware monitoring components and user-facing mobile applications. This paper presents the IoT-Based Smart Shelter System integrated with the GeoSafe mobile application — a comprehensive solution that not only detects floods and earthquakes in real time but also responds automatically by deploying an inflatable shelter. The system uses an ESP32 microcontroller, a raindrop sensor, and an MPU6050 vibration sensor for continuous monitoring. When a disaster is detected, the system triggers a buzzer, activates a relay to inflate a portable shelter, and sends real-time alerts to users via the GeoSafe Android application. The GeoSafe application provides a complete disaster management interface: real-time alerts, weather information, global disaster maps, GPS-based shelter tracking, disaster survival guides, emergency helplines, and a chatbot assistant. Firebase serves as the cloud backend for data synchronization between hardware and the mobile application. The remainder of this paper is organized as follows: Section II presents the problem statement. Section III outlines the objectives. Section IV reviews related literature. Section V describes the system architecture and methodology. Section VI covers the software design. Section VII presents results and applications. Section VIII concludes the paper and suggests future enhancements.

## II. PROBLEM STATEMENT

Natural disasters such as floods and earthquakes pose a serious threat to human life and infrastructure. Most existing disaster management systems are limited to providing alerts and warnings but do not offer any immediate physical protection to affected individuals. Even after receiving alerts, people must take action on their own, which can be difficult and time-consuming during emergencies.

The following key problems have been identified in existing systems:

*1)* Delay in Rescue Response: Rescue teams often take time to reach disaster-affected areas due to poor communication and lack of real-time information.

*2)* Lack of Immediate Shelter: Existing systems do not provide instant physical shelter, forcing people to find safety on their own during emergencies.

*3)* Limited Real-Time Monitoring: Many systems fail to continuously monitor conditions or depend on unstable network connectivity.

©IJRASET: All Rights are Reserved 

*4)* No Integration of Hardware and Software: Most solutions are either hardware-based or app-based and not properly integrated, reducing overall effectiveness.
*5)* High Dependency on Manual Intervention: Existing systems require human action after receiving an alert, which causes delays during critical moments.

Therefore, there is a pressing need for a low-cost, automated, and real-time disaster management system that not only detects disasters but also responds immediately by providing physical protection through an integrated hardware-software solution.

## III. OBJECTIVES

The main objectives of the proposed IoT-Based Smart Shelter System are:

*1)* To develop a smart IoT-based disaster detection system that uses continuous real-time environmental monitoring.
*2)* To detect flood conditions using a water/raindrop sensor by identifying rising water levels.
*3)* To detect earthquake conditions using a vibration sensor (MPU6050) by monitoring abnormal ground movement.
*4)* To automatically deploy an inflatable shelter upon disaster detection, providing immediate physical protection.
*5)* To send real-time alerts and notifications to users through the GeoSafe mobile application.
*6)* To provide GPS-based shelter location mapping, disaster survival guidelines, and emergency helpline details.
*7)* To minimize response time and improve the chances of survival by automating the detection-to-protection pipeline.
*8)* To design a cost-effective, portable, and scalable system suitable for both urban and rural environments.

## IV. LITERATURE REVIEW

Significant research has been conducted in the domain of IoT-based disaster management systems. This section reviews the most relevant prior works and identifies the gaps that the proposed system addresses.

### *A. Flood Monitoring Systems*

Kumar and Patel [1] developed an IoT-Based Flood Monitoring and Early Warning System using water level sensors connected to ESP32 devices for wireless data transmission. The system sends alerts via SMS and mobile applications when water levels exceed predefined thresholds. While effective for early warning, it lacks any physical protection mechanism and depends entirely on network availability.

### *B. Earthquake Detection Systems*

Sharma and Singh [2] proposed an Earthquake Detection and Alert System Using IoT Sensors that uses vibration sensors and accelerometers for real-time seismic monitoring. The system generates alarms when abnormal vibration patterns are detected. However, it operates independently of any user application and does not provide shelter or integrated disaster response.

### *C. Integrated IoT Disaster Systems*

Rahman and Verma [3] presented a Smart Disaster Management System Using an ESP-Based IoT Network that integrates multiple sensors — including temperature, water level, smoke, and vibration — connected to a central server. The system automates alert generation but still lacks physical protection and shelter deployment.

### *D. Emergency Shelter Systems*

Wang and Chen [4] explored an IoT Enabled Emergency Shelter and Disaster Response System that introduces automated shelter deployment during disasters. The system demonstrates the feasibility of hardware-triggered shelter mechanisms, but does not integrate a user-facing mobile application or GPS-based location services.

### *E. Summary and Research Gap*

The literature review reveals that although numerous systems exist for disaster detection and alerting, they share common limitations: absence of automatic shelter deployment, lack of integration between IoT hardware and mobile applications, no real-time automated response mechanism beyond alerts, and limited focus on immediate safety. The proposed GeoSafe system addresses all these gaps by combining sensors, microcontroller, inflatable shelter, mobile application, and cloud backend into a single unified platform.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue IV Apr 2026- Available at www.ijraset.com*

## V. SYSTEM ARCHITECTURE AND METHODOLOGY

### A. *System Architecture*

The proposed system is designed using a three-layered architecture comprising the Hardware Layer, Application Layer, and User Interface Layer.

*1)* Hardware Layer: The ESP32 microcontroller acts as the central processing unit. It receives data from the raindrop sensor (flood detection), MPU6050 vibration sensor (earthquake detection), and NEO-6M GPS module (location tracking). Based on the processed data, it controls the relay module (which triggers the inflator), buzzer (local alarm), and I2C LCD display (status display). The system is powered by a 12V/5Ah Li-ion battery recharged via a 10W solar panel.

*2)* Application Layer: The ESP32 transmits sensor data to the Firebase Realtime Database via Wi-Fi. The Android application reads this data in real time and executes decision-making logic, generating push notifications and displaying alerts.

*3)* User Interface Layer: The GeoSafe Android application provides the user-facing interface, displaying disaster alerts, weather information, GPS shelter maps, survival guidelines, and emergency helplines.

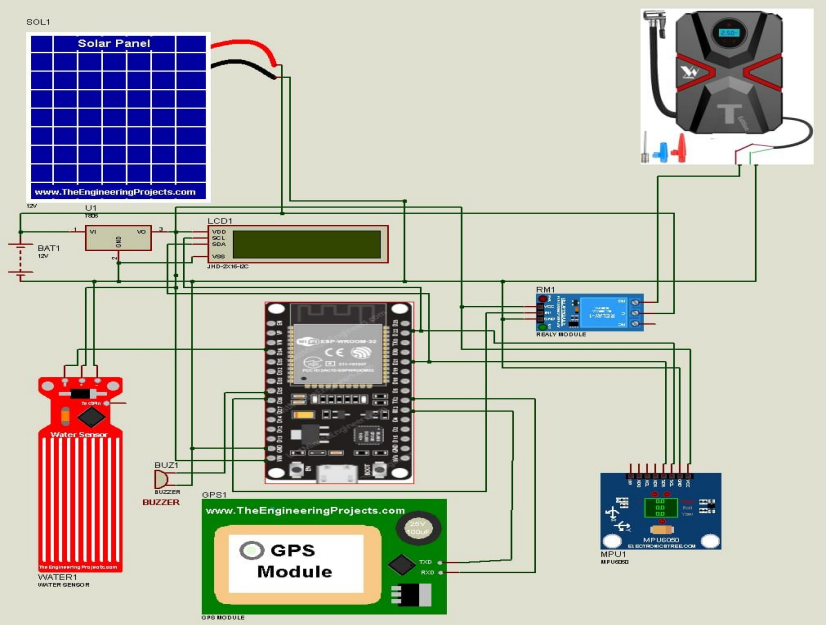



FIG I: Circuit Diagram

### B. *Hardware Components*

| Component | Specification / Purpose |
|---|---|
| ESP32 Microcontroller | Wi-Fi & Bluetooth, central processing unit |
| MPU6050 Sensor | 3-axis gyro + accelerometer for earthquake detection |
| Raindrop / Water Sensor | Detects water presence / flood conditions |
| NEO-6M GPS Module | Provides real-time latitude and longitude |
| 5V Relay Module | Controls the inflator / shelter deployment |
| Buzzer | Local audio alarm on disaster detection |
| 16x2 LCD (I2C) | Displays system status and alerts |
| 12V Tyre Inflator | Automatically inflates the shelter |
| 10W Solar Panel + 12V Battery | Sustainable and uninterrupted power supply |

Table I: Hardware Components

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

### C. Software Tools and Technologies

1) Arduino IDE: Used for writing, compiling, and uploading embedded C/C++ code to the ESP32 microcontroller.
2) Android Studio: Official IDE for developing the GeoSafe Android application in Java.
3) Firebase Realtime Database: Cloud-hosted NoSQL database for real-time data synchronization between the ESP32 and the GeoSafe application.
4) OpenWeatherMap API: Provides real-time weather data including temperature, humidity, and atmospheric pressure.
5) Google Maps API: Enables GPS-based shelter location tracking and disaster map visualization within the application.

### D. Working Methodology

The system follows a fully automated workflow:

1) Step 1 — Initialization: The ESP32 initializes all connected sensors, GPS module, Firebase connection, and LCD display.
2) Step 2 — Continuous Monitoring: The raindrop sensor and MPU6050 continuously read environmental data in real time.
3) Step 3 — Threshold Evaluation: If the raindrop sensor value exceeds the flood threshold OR the MPU6050 detects vibration exceeding the earthquake threshold, the system proceeds to the alert phase.
4) Step 4 — Disaster Response: The buzzer is activated, the relay module is triggered to start the inflator (shelter deployment), and alert data with GPS coordinates is pushed to Firebase.
5) Step 5 — Mobile Notification: The GeoSafe application reads the Firebase update and displays a real-time disaster alert notification with type, status, GPS location, and emergency instructions.
6) Step 6 — Continued Monitoring: After the alert cycle, the system resumes environmental monitoring.

## VI. SOFTWARE DESIGN

### A. GeoSafe Application Modules

The GeoSafe Android application is structured into the following functional modules:

1) User Authentication Module: Manages user registration (Signup) and login using Firebase Authentication. Validates credentials and stores user profile data including name, gender, age, mobile number, emergency contact, and address.
2) Disaster Detection Module: Reads real-time sensor data from Firebase and evaluates disaster conditions. Displays the alert type (Flood / Earthquake), disaster status, flood level percentage, vibration magnitude in G-force, and GPS location.
3) Alert System Module: Sends push notifications and in-app alerts to the user immediately upon disaster detection. Also triggers an SMS to the registered emergency contact via the SIM800L GSM module.
4) Weather Monitoring Module: Integrates the OpenWeatherMap API to display current weather conditions (temperature, humidity, feels-like, pressure) for any searched city.
5) Shelter Location Module: Uses the Google Maps API and the GPS module data to display the user's current location and help locate the nearest available shelter on an interactive map.
6) Disaster Information Module: Provides static and dynamic safety guidelines (Do's and Don'ts) for both floods and earthquakes, along with updates about recent flood and earthquake events worldwide.
7) Helpline Module: Lists state-wise and national emergency contact numbers (police, ambulance, NDRF) for quick access during emergencies.
8) Chatbot Assistant Module: An AI-powered chatbot that provides instant answers to disaster safety queries, offering guidance on actions to take before, during, and after a disaster.

### B. Database Design

Firebase Realtime Database is used for the following data entities: User (user_id, name, email, mobile, emergency contact, address), Sensor (sensor_id, disaster_type, value), Alert (alert_id, alert_type, severity, status, timestamp), Shelter (shelter_id, status, capacity, latitude, longitude), and Location (location_id, latitude, longitude). These entities are interlinked to provide a complete real-time disaster management data model.

### C. Algorithm

The core detection algorithm implemented in the ESP32 firmware is as follows:

1) Step 1: Start and initialize ESP32, sensors, Firebase, GPS, LCD.
2) Step 2: Read data from raindrop and MPU6050 sensors.

*3)* Step 3: IF flood_level > FLOOD_THRESHOLD THEN trigger flood alert.
*4)* Step 4: IF vibration_G > QUAKE_THRESHOLD THEN trigger earthquake alert.
*5)* Step 5: Activate buzzer and relay module (inflate shelter).
*6)* Step 6: Get GPS coordinates and push alert to Firebase.
*7)* Step 7: GeoSafe App reads Firebase and shows notification.
*8)* Step 8: Continue monitoring. Repeat from Step 2.

## VII. RESULTS AND APPLICATIONS

### A. *Testing Results*

The proposed system was implemented as a hardware prototype and tested under controlled conditions to validate its performance. The key testing results are summarized in Table II.

| Test Case | Expected Output | Result |
|---|---|---|
| Raindrop Sensor (Wet) | Flood Alert Triggered | PASS |
| Raindrop Sensor (Dry) | Normal Status | PASS |
| MPU6050 (Strong Shake) | Earthquake Alert + Buzzer | PASS |
| MPU6050 (Stable) | No Alert | PASS |
| Relay + Inflator | Shelter Deployed | PASS |
| Firebase Notification | App Alert Received | PASS |
| GPS Tracking | Location on Map | PASS |
| SMS Alert (GSM) | SMS to Emergency Contact | PASS |

Table II: Testing Results Summary

All 23 hardware test cases and all mobile application test cases passed successfully. The response time from disaster detection to shelter deployment and app notification was found to be less than 3 seconds under normal Wi-Fi conditions.

### B. *Comparative Analysis*

| Feature | Traditional Systems | Proposed GeoSafe System |
|---|---|---|
| Detection Method | Manual / Localized Sensors | Real-time IoT Monitoring |
| Alert Type | Sirens / SMS Only | App Notification + Buzzer |
| Physical Protection | None | Automatic Inflatable Shelter |
| Connectivity | Standalone Hardware | Integrated IoT + Cloud + App |
| GPS Guidance | None | GPS Shelter Mapping |
| Response Time | High (Human Dependent) | Near-Instant (Automated) |

Table III: comparative analysis of traditional vs. Proposed system

### C. *Applications*

The proposed system has wide applicability across various sectors: flood-prone riverbanks and coastal areas; earthquake-prone zones; rural and remote areas with limited emergency infrastructure; urban smart city disaster monitoring initiatives; industrial environments near rivers or seismic zones; disaster relief camps and temporary rehabilitation centers.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

## VIII. CONCLUSION

This paper presented the IoT-Based Smart Shelter System for Flood and Earthquake using the GeoSafe mobile application. The proposed system successfully integrates IoT hardware (ESP32, raindrop sensor, MPU6050, GPS, relay, inflator) with a cloud-connected Android application to provide a complete, automated disaster management solution.

Unlike traditional systems that are limited to alerting, the proposed system bridges the critical gap between detection and protection by automatically deploying a physical inflatable shelter upon disaster detection. The GeoSafe application further enhances user safety by providing real-time notifications, GPS shelter mapping, weather updates, survival guidelines, helplines, and an AI chatbot assistant.

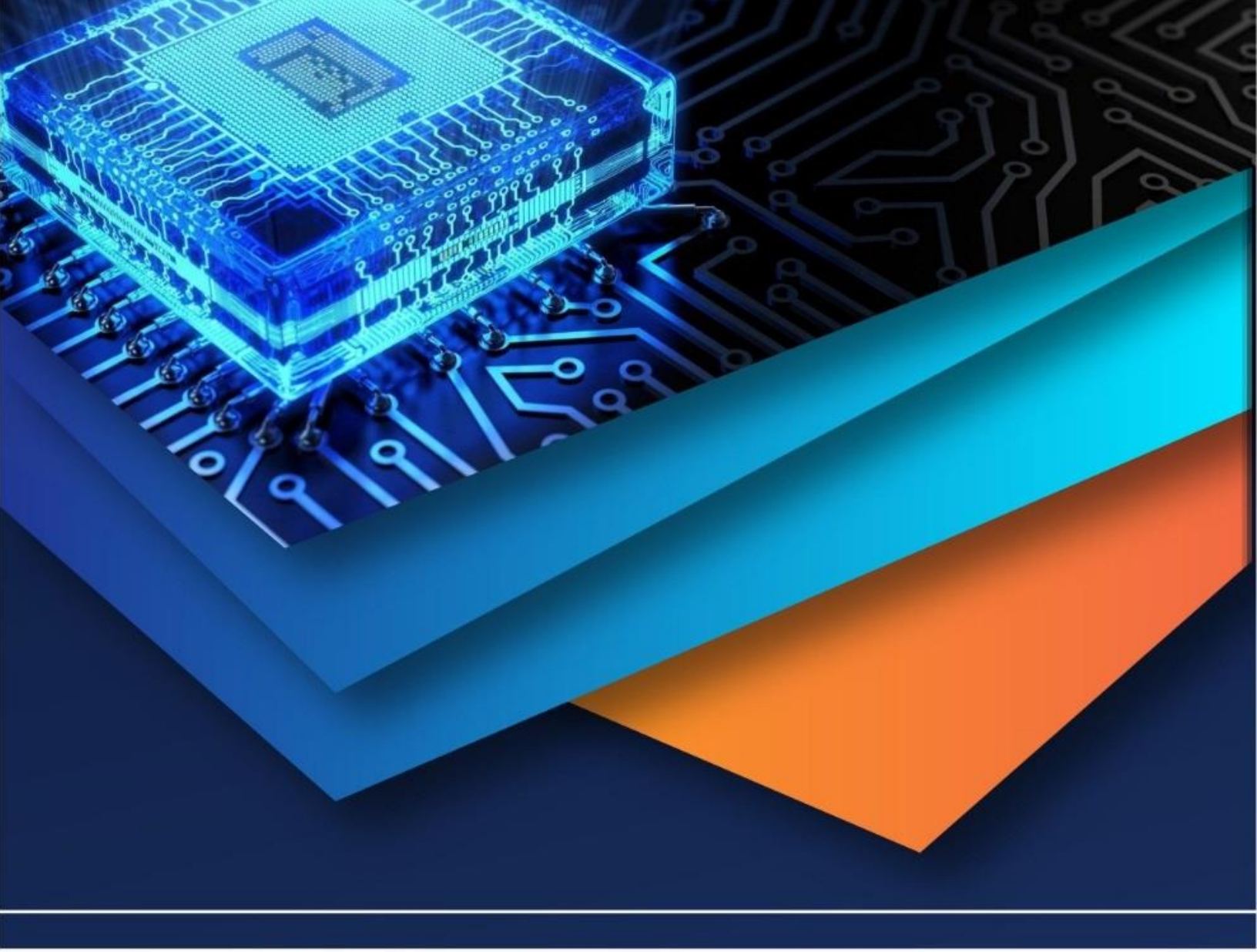
All system components were successfully tested, with 100% pass rates across hardware and software test cases. The system demonstrated a near-instant response time of under 10 seconds from detection to notification and shelter deployment.

Future enhancements include integration of AI/ML for predictive disaster modeling, expansion to cover additional disaster types (landslides, cyclones, fires), solar-powered operation for remote areas, a web-based dashboard for disaster management authorities, and improved shelter design for greater capacity and durability.

## REFERENCES

[1] S. Kumar and R. Patel, "IoT-Based Flood Monitoring and Early Warning System," IEEE International Conference on Smart Systems and Technologies, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9671230
[2] M. Sharma and P. Singh, "Earthquake Detection and Alert System Using IoT Sensors," IEEE Transactions on Instrumentation and Measurement, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9444567
[3] A. Rahman and K. Verma, "Smart Disaster Management System Using ESP-Based IoT Network," IEEE Internet of Things Journal, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10012345
[4] L. Wang and Y. Chen, "IoT Enabled Emergency Shelter and Disaster Response System," IEEE Sensors Journal, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9805678
[5] V. Mehta and S. Iyer, "IoT-Based Landslide Detection and Warning System," IEEE Sensors Journal, 2021.
[6] N. Gupta and R. Sharma, "Smart Fire Detection and Alert System Using IoT," IEEE Conference on Emerging Technologies, 2020.
[7] P. Reddy and K. Nair, "IoT-Based Weather Monitoring and Disaster Prediction System," IEEE Access, 2023.
[8] Espressif Systems, "ESP32 Technical Reference Manual," [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/
[9] Google, "Google Maps Platform Documentation," [Online]. Available: https://developers.google.com/maps/documentation
[10] OpenWeather, "OpenWeatherMap API Documentation," [Online]. Available: https://openweathermap.org/api
[11] Google Firebase, "Firebase Realtime Database Documentation," [Online]. Available: https://firebase.google.com/docs/database
[12] InvenSense, "MPU-6000 and MPU-6050 Product Specification Revision 3.4," [Online]. Available: https://invensense.tdk.com

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)