International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

FLOOD MITIGATION IN RURAL AREAS BY AUTOMATIC RAIN GAUGE SYSTEM

Ebin Peter

Department of Computer Science & Engineering Mangalam College of Engineering Kottayam ebinpeter95@gmail.com

ABSTRACT —

The 2018 Kerala floods, particularly in regions such as Ernakulam Panchayat, were a massive wake-up call. They revealed to us how unprepared we were when it came to receiving timely and accurate rainfall data. To prevent such catastrophes in the future, an automatic rain gauge system has been created. It is a low-cost, real-time, IoT-based system that records rainfall and issues warnings in advance before things get out of hand. At the core of the system is an ultrasonic sensor combined with an ESP32 microcontroller. This combination continuously reads the water level and computes the rainfall that has fallen. To prevent false warnings and provide accurate values, a simple average is used. The ESP32 also checks the rainfall that has fallen and decides to provide a Yellow, Orange, or Red warning, based on standard values. What's more, the data doesn't get stuck in the device. It's streamed in real-time to a Firebase Real-time Database over Wi-Fi. A Flutter-based mobile app retrieves this data and displays users the current rainfall values and warning status right on their phone. There's also a small OLED display taped on for local readings, and the entire setup runs either on USB power or a rechargeable battery—so it can run in remote areas too. The village authorities are at time alerted via the app In the future, it can be expanded with other sensors and features such as weather forecasting and instant push notifications. This automatic rain gauge isn't a tech project—it's a people-first solution. It's designed to help local citizens stay informed and safe, particularly in flood-prone regions. If it is used extensively, it can actually make a significant difference in how we deal with heavy rainfall and prepare for natural disasters in the future.

Keywords— Flood; landslides; Kerala Floods; alert system; Flutter app;

Introduction

In August 2018, Kerala witnessed a tragedy that shook the state to its core — the worst flood in nearly a century. It was not heavy rain; it was days of continuous rain that swelled rivers, swept away roads, and left thousands without food, shelter, or help. Ernakulam Panchayat was one of the most affected, where the unexpected rise in water levels caught locals entirely off guard. Kerala is used to monsoons, but this flood taught us that we were lacking something very critical — real-time information on how much it's raining and how fast the water is rising. Back then, most areas didn't have any system to track rainfall as it happened. There were weather forecasts on TV, sure, but for someone in a small panchayat or farmland, there was no way to know how serious things had gotten — until it was too late. That's where the idea for this project was born. The goal was simple: build an automatic rain gauge system that can monitor rainfall in real time and alert people the moment it reaches dangerous levels. The working is simple but intelligent. The ultrasonic sensor is this system is constructed with low-cost components such as the ESP32 microcontroller, an ultrasonic sensor, and an OLED display, all connected with cloud technology and a mobile application. It's made in such a manner that even a small village can install it and begin using it — no complex setup, no experts required. The working is simple but intelligent. The ultrasonic sensor is mounted over a container or rain gauge. When the container fills with rainwater, the sensor continuously checks how far the water level is from it. It measures this against the initial dry reading (baseline), computes the increase in the water level, and translates that into rainfall in millimeters. To ensure the values aren't erratic or wrong, the system employs a rolling average of the previous 10 readings. This makes things smooth and reliable.

The system brain is the ESP32 chip, which not only receives the sensor data but also monitors how much rain has accumulated in total. If it crosses specific thresholds, it sends alerts. The alerts are according to the standard system applied in India:

Yellow Alert if rainfall between 64.5 mm and 124.4 mm

Orange Alert if between 124.5 mm and 244.4 mm

Red Alert if it exceeds 244.4 mm

After the system determines the alert level, it sends the information to a Firebase Realtime Database via Wi-Fi. That's where the magic occurs — a Flutter mobile application (which we also created) receives updates from Firebase and displays the real-time rainfall and alert level to the user. The design is clean, clear, and easy to comprehend even for a person who isn't tech-literate.

To ensure the system is accessible even when there's no internet or mobile app, we included a small OLED display that displays the rainfall and alert on the device itself. This is useful for farmers, panchayat offices, or disaster response teams who wish to look at the readings without reaching for their phones. We also considered portability as a feature. The entire system can be installed almost anywhere, including close to fields, riverbanks, rooftops, and low-lying areas that are vulnerable to flooding, because it can be powered by a standard USB power bank or battery. Additionally, because the components are readily interchangeable and reasonably priced, the entire system requires little upkeep and is reasonably priced. Kerala's realities—heavy rains, power outages during the monsoon season, and spotty internet in some rural areas—were taken into consideration when designing this. Giving people advance notice will help them avoid waiting for the news or relying on speculation. If every panchayat had this during the 2018 floods, people could have received alerts more quickly, authorities could have been better prepared, and lives could have been saved.

And this is not just a tool for flood alerts. It can be used in farming, weather research, school science labs, and even by local disaster management cells. It can easily be extended by adding more sensors like temperature, humidity, or wind speed. We're also looking at ways to integrate push notifications, so people get alerts directly on their phones without even opening the app. In short, this automated rain gauge is not just a tech project — it's a reaction to a real, personal experience that impacted thousands of individuals across Kerala. It demonstrates how we can utilize low-cost, simple technology to make our communities smarter, safer, and better equipped to deal with what nature delivers. And all it takes is a bit of assistance from mobile apps and IoT. We can take weather forecasting into our hands — literally.

LITERATURE REVIEW

[1] This paper an Internet of Things (IoT)-enabled flood monitoring system aims at capturing actual environmental data in real time, computing possible flooding hazards, and sending timely warnings to reduce disaster effects. The system structure goes around around a sensor mainly ultrasonic water level sensors and rain sensors—which are connected to microcontrollers like Arduino or NodeMCU. These microcontrollers send data through wireless networks (Wi-Fi or GSM) to a central cloud-based monitoring system, where it is constantly assessed against pre-programmed flood risk thresholds. When critical water levels are identified, the system initiates automated alert mechanisms, including SMS messages or push notifications through mobile applications, to alert emergency services and vulnerable communities. The approach focuses on real-time monitoring, low-cost deployment, and scalability potential. Nonetheless, the research identifies some gaps and limitations. One of the limitations is the absence of contextual intelligence—the system mainly responds to water level thresholds without forecasting ability, like predicting flood risk based on historical and meteorological patterns. Another limitation is sensor accuracy and robustness in extreme weather conditions, particularly in rural or remote areas where maintenance is challenging. Data privacy and security are not rigorously covered, with risks introduced when sensitive infrastructure or geolocation data are passed. Additionally, scalability of the system to the large river catchments or the urban flood regions is constrained due to network connections and the required extensive infrastructure. Integration with the machine learning algorithms, GIS maps, and environmental factors such as soil moisture will greatly enhance system robustness and forecasting accuracy. In summary, although the paper suggests a low-cost, practical IoT flood detection solution, it sets the stage for far more sophisticated, integrated disaster resilience systems yet to be fully realized.

[2] This paper says about mobile-based early warning system that aims to track and predict rainfall-induced landslides in Malaysia, especially in Ulu Kelang. The system take on satellite-derived rainfall data from the Tropical Rainfall Measuring Mission (TRMM) and Global Precipitation Measurement (GPM) missions to identify hazardous levels of precipitation. The approach is to generate landslide rainfall thresholds from the Empirical–Empirical (E-E) diagram, which plots accumulated rainfall for two periods (e.g., 3-day and 15-day) against past landslide events. From these thresholds, the system assigns risk to high, moderate, and low alert levels. Backend process is done on a day and real-time notification and situational awareness are delivered by the mobile app. A web-based platform is also available for wider spread . But the system's dependency on satellite rainfall data brings with it some limitations, including inability to utilize localized ground trothing, time lag in available data, and lack of real-time analysis of terrain movement, which might impact predictive accuracy. Future developments might involve inclusion with ground sensors (e.g., IoT rain gauges, soil moisture sensors), machine learning-driven threshold optimization, and user interface experience improvements in the mobile app. Conceptually, the working model encompasses elements such as satellite data acquisition, data processing engine, threshold risk classification module, and a mobile/web notification layer, all with the goal of enhancing community resilience against landslide risk. The study shows an early warning system in operation, there are certain areas which can be improved further: Integration with Local Ground Data: Integration of local meteorological and geological data could improve the accuracy of prediction. Real-Time Data Processing: Real-time data processing implementation could give more timely alerts. User Interface Rise: Improvement of the user interface of the mobile app for easier accessibility and user experience.

[3] This paper describes design and development of an IoT-based flood warning system with the purpose of offering real-time monitoring and early warning in flood-affected zones. The approach revolves around a NodeMCU ESP8266 microcontroller with a potentiometer sensor, which physically monitors water level variations. The sensor unit is installed on a straigh mounted shaft rising with the water, and the displacement is converted into electrical signals. The system is designed to transmit data over Wi-Fi to the cloud platform named ThingSpeak, to support real-time data visualization and alert generation. The alarm is triggered when water levels reach 85% of the configured danger level with less than 10 seconds reported data transmission and response time. The four core elements of the operating framework are a sensor module (potentiometer and float), a communication and processing module (NodeMCU), an analytics and cloud storage platform like ThingSpeak, and a user interface (web-based dashboard for alarms and real-time monitoring). Even though it is success, the research identifies some problems that needed to be address, including not being able to include other environmental sensors (like soil moisture or rainfall), restrict sensor calibration to different field conditions, suspected to data leakage, and uncertainty in terms of scalability of the system to larger geography areas or more complicated hydrological track. These limitations suggest that while the system is operational and cost-effective, subsequent versions would significantly improve its performance, reliability, and versatility.

[4] This paper says on new flood monitoring system with the Internet of Things (IoT) driven Automatic Water Level Recorder (AWLR-IoT). The system integrate sensors, including rain gauges, water level sensors, and flow rate sensors, to monitor environmental conditions such as humidity, temperature, water level, water rise rate, and rain. Data sensed by the sensors are processed and transferred to a cloud-based data store for real-time analysis. The system hold a Wireless Sensor Network (WSN) to enable sensors to communicate with the cloud platform. If the system detects conditions that result in

an impending flood, it notifies through different channels like mobile apps, emails, or SMSes, giving early warnings to vulnerable communities. The approaches aims at enhancing the power and optimization of flood detection and alert system. While the system designed demonstrate significant enhancement in flood detection, some areas are still to be researched: Sensor Calibration: Ensuring accuracy and reliability of sensors across varied environmental conditions.Data Security, Having strong security practices in place to safeguard the transmitted data from unauthorized use. Scalability, Measuring system performance and scalability for larger, more sophisticated flood-prone areas. Integration, Integrating with other environmental sensors, like soil moisture sensors, to give an advanced flood monitoring system. The system has several key components like Sensor Module, which is Equipped with rain gauges, water level sensors, and flow rate sensors to detect environmental parameters. Microcontroller (e.g., NodeMCU ESP8266), which translates data from the sensors and enables communication between the sensors and the cloud platform. Cloud Platform (e.g., ThingSpeak which Collects, stores, and processes data for real-time monitoring. User Interface, which Presents the alerts and real-time monitoring to users through web portal or mobile app. This structure allows for ongoing monitoring of environmental conditions, early warning of possible flood events, and rapid dissemination of the warning to the public, thus improving disaster response and readiness.

[5] The study proposes an IoT system that is designed to track and monitor floods and landslides in real time, generating early warning alerts to minimize dangers of disasters. The process involves installing various detectors, including vibration detectors, rain detectors, and rain gauges, in positions of vulnerability at strategic locations. The detectors record environmental information, which is then transmitted wirelessly through Wi-Fi or cellular connections to the cloud platform. The cloud platform performs real-time analysis to identify potential threats so that early warnings can be sent to authorities and citizens in a timely manner. The strategy is driven towards improving disaster preparedness and response by continuous monitoring and early detection. Although the system outlined provides tremendous improvement in the early detection of floods and landslides, there are numerous issues that need to be researched Sensor Calibration, which Ensuring the accuracy and reliability of sensors under diverse environmental conditions. Data Security which is to Enforce stringent security measures to secure the data in transit against any unsuitable access. Scalability for Testing the system's performance and flexibility in larger and more complex flood-risk areas. Integration to Incorporating other environmental sensors, e.g., soil moisture sensors, to make it a broader flood monitoring solution. Landside and flood tide monitoring and discovery can be made using this prototype. It's an efficient and cost-effective pre-warning system of the landslide as well as real-time monitoring of the data. For assigning the parameters for landslide threshold values are assigned for vibration, rain drop detection as well as humidity detector. It is designed such that threshold values crossing can make the system capable of sending the alert to the end stoner. The prototype also has the possibility of use in large area by scaling up the number of detectors to the NodeMCU ESP8266. Warning advising system for landslides can save mortal lives and loss of property.

SYSTEM DESIGN

Hardware architecture of the real-time rain monitoring system includes various integrated components that are used for detecting environmental changes, processing data, and communication with cloud services. The arrangement provides accurate measurement and smooth data transmission with low power requirements for remote deployment.

The ultrasonic sensor is used to measure the water level, and it also has a straight link with the volume of rainfall. It works based on emitting ultrasonic pulses and measuring the time it takes for the echo to travel back after reflecting off the surface of the water. The time delay is then used to calculate the distance using the formula:

$$Distance = \frac{Time * Speed\ of\ Sound}{2}$$

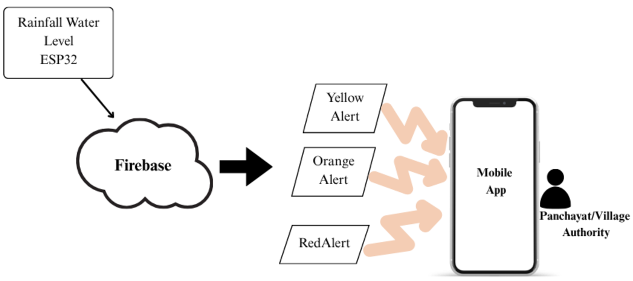

Figure 1: Rainfall alert system for different categories of warning according to Kerala conditions

The sensor is mounted vertically over a rain collection container. When rain falls, the water rises, reducing the measured distance. The ESP32 microcontroller receives this data through a UART interface (TX/RX), making it monitor changes in water level. Such changes are used to estimate rainfall amount.

The ESP32 is the system's central processing unit. It takes in the data from the ultrasonic sensor, performs the calculation of the water level, and translates the measurement into rainfall values. The equation for the calculation of the water level is:

$$Water\ Level = Baseline\ Distance - Current\ Distance$$

This water level is then converted into rainfall in millimeters using the equation:

$$Rainfall(mm) = Waterlevel(cm) * 10$$

The ESP32 and the app are the main parts of the whole project. ESP32, it actually collects data from all the sensors—like rain, water level, and flow rate—and then clears it up a bit by simply applying an averaging method to remove any noisy or random readings. After data cleansing, it checks to determine whether the rain has crossed a certain threshold value that would confirm problems. If it has, the system triggers an alert level. So yeah, it's doing all the backend work to make sure you're not getting false alarms or missed warnings.

After that, the ESP32 handles communication too. It uploads all the processed info—like total rainfall and alert levels—directly to Firebase. This happens in present time so the data gets updated instantly. And also along with the data time is also recored. This directly is measures the intensity of the rainfall Anyone tracking the system through the cloud can see what's going on without any delays. It's super useful for real-time monitoring, especially if you're not near the device but still want to keep tabs on the situation.

And for local information updates, there's an OLED screen directly on the device itself. It displays to you the present rainfall, water level, and alert level without opening an app or anything. It updates every minute because the ESP32 keeps reading fresh values and updating Firebase. Power-wise, you have USB when you're developing or testing and rechargeable battery when you are out in the field. The phone even goes into low-power mode (like deep sleep) when idle, so the battery lasts longer without being constantly plugged back in. Total convenience, especially for off-grid sites.

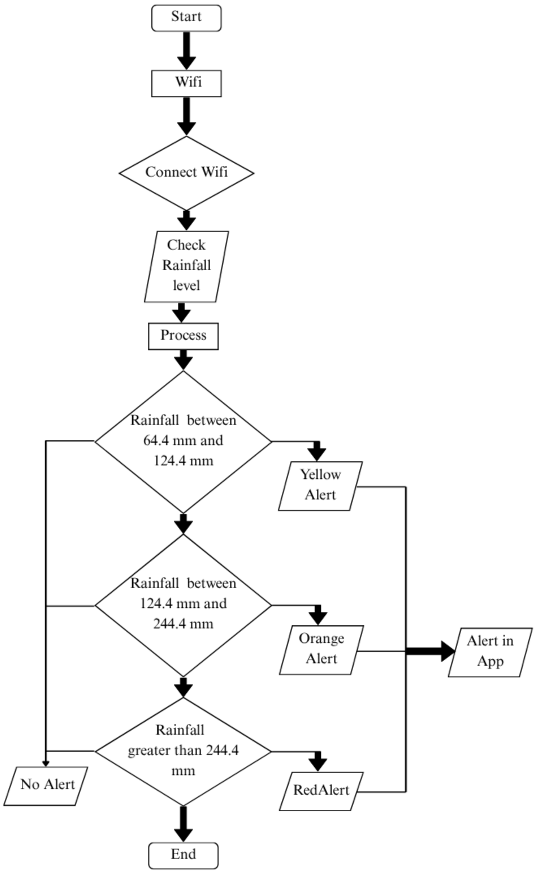

Figure 2: Flow chart of the Rainfall alert system

The Arduino program targets to monitor and record rainfall through an ESP32 microcontroller and an ultrasonic sensor and an OLED display. It started by including the necessary libraries for I2C communication and graphics, then initialize the OLED display with a given width, height, and I2C address. The ultrasonic sensor communicates with the ESP32 through UART at specific RX and TX pins. Another hardware serial port is also initialized for it.

To measure rainfall, the system calculates the distance from the sensor to the water level. A baseline dry distance is established (30 cm), and the difference between this and the present sensor reading gives the water level. This water level is then converted to rainfall in millimeters. The code applies a moving average algorithm to filter the readings with a buffer of the last ten values. This serves to alleviate oscillations due to noise or imperceptible disturbances on the surface of the water.

Rainfall amounts are updated each minute. The ESP32 checks if the water level has increased and subsequently calculates the quantity of rainfall received in that interval and adds the sum. Based on this running total, the system determines an alert level: Yellow, Orange, or Red, based on standard rainfall levels. These values are shown on the OLED screen for real-time local observation.

Apart from local display, the system prints debug information to the serial monitor. Though the current version of code does not upload data to the cloud, it is designed to be extended with Firebase or other cloud database integration easily, allowing remote monitoring through a mobile app. Its modular, low-cost design makes it perfect for community-level deployment in flood-affected areas

RESULTS

The following table summarizes the data collected during the testing process. The table includes the time, water level measurements, calculated rainfall for each interval, total rainfall, and the corresponding alert level based on the system's predefined thresholds.

Table 1

Time	*Water Level (cm)*	*Rainfall This Interval (mm)*	*Total Rainfall (mm)*	*Alert Level*
08:00 AM	**0.5**	**5.0**	**5.0**	**No Alert**
08:10 AM	**1.2**	**12.0**	**17.0**	**No Alert**
08:30 AM	**6.0**	**60.0**	**77.0**	**Yellow Alert**
09:00 AM	**8.0**	**80.0**	**157.0**	**Orange Alert**

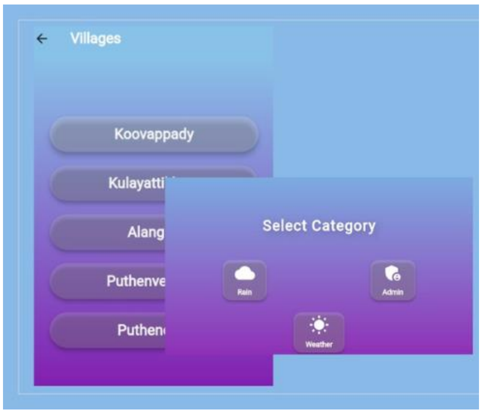

Figure 3: App showing different villages with different categories

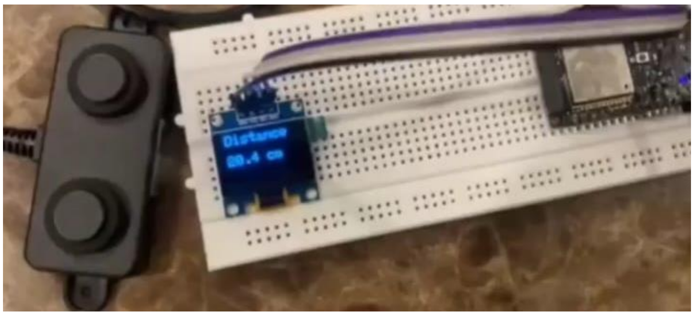

Figure 4: OLED Display showing the water level distance

CONCLUSION

The automatic rain gauge system proved to be an effective and efficient means of real-time rainfall observation and alerting. Its ability to accurately monitor increasing water levels and convert them into meaningful rainfall data shows the potential that using low-cost hardware, one can create a system that will help in local weather awareness and flood preparedness. The alert mechanism responded appropriately at each stage, providing timely warnings as rain levels increased, which is crucial for heavy monsoon-prone populations.

Most importantly, the seamless real-time synchronization with the mobile app kept users updated wherever they were. Whether light rain or the beginnings of something heavier, the system kept users abreast in real-time via Firebase updates. In total, this project appears to offer a practical, scalable solution to local disaster preparedness—especially in places like Kerala, where the memory of the 2018 floods still fuels the demand for increased preparedness.

ACKNOWLEDGMENT

I would like to thank all those who guided and supported me during the course of developing my project. I would like to thank Mangalam College of Engineering for the support and academic space that enabled the completion of this project. I also wish to thank Department of Computer Science & Engineering for their appropriate guidance, valuable assistance and helpful comments during the proofreading process. Heartfelt thanks also to my fellow students and friends for their inspiring words and morale support throughout each stage of completion. Lastly, I would like to express my gratitude to the open-source community and researchers whose work paved the way for the implementation. This project has been a very enriching learning experience, and I am grateful for the opportunity to be a part of the emerging field of assistive technologies.

REFERENCES

[1] Y. Suppakhun, "Flood Surveillance and Alert System: An Advance in IoT," *Proceedings of the 2019 Asia Pacific Conference on Circuits and Systems (APCCAS)*, pp. 325–328, 2019.

[2] N. Ya'acob and W. N. W. Muhamad, "Mobile Application for Rainfall-Landslide Early Warning System (RLEWS) using Global Precipitation Measurement (GPM)," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 9, pp. 4939–4946, 2023

[3] Silaban, F. A., Taufiq, Y., Silalahi, L. M., & Sihombing, G. L. A. (2023). Flood Detection Design based on the Internet of Things. *Buletin Ilmiah Sarjana Teknik Elektro*, 5(4), 427–437.

[4] Joshi, M., & Murali, S. (2025). An Efficient Smart Flood Detection and Alert System based on Automatic Water Level Recorder Approach using IoT. *International Journal of Computational and Experimental Science and Engineering*, 11(1), 449–463.

[5] Badiger, G., Gaikwad, S., Kulkarni, S., & Kodgirwar, V. (2024). IoT Based: Landslide and Flood Detection System. *International Journal of Electronics, Automation and Computer Applications*, 1(1), 1–4.