

Fews: IoT-Based Flood Early Warning System for Barangay Doña Imelda, Quezon City

Emmanuel E. Crucillo., Harold R. Lucero., Ma. Abegail N. Gelito., Lyka May C. Medina., Jhon Rey O. Palconit

School of Information Technology Colegio de Sta. Teresa de Avila

DOI: <https://doi.org/10.51584/IJRIAS.2026.110100105>

Received: 31 January 2026; Accepted: 05 February 2026; Published: 15 February 2026

ABSTRACT

Flooding remains one of the most destructive natural hazards affecting urban communities in the Philippines, particularly those located near major river systems. This study presents the design, development, and evaluation of an Internet of Things (IoT)-based Flood Early Warning System (FEWS) for Barangay Doña Imelda, Quezon City, an area highly vulnerable to recurring floods. The system utilizes an ESP32 microcontroller integrated with an HC-SR04 ultrasonic sensor for water-level measurement, a Neo-6M GPS module for geolocation, and a Ra-01 LoRa module for long-range data transmission. Real-time water-level data are transmitted to a cloud-based database and visualized through a web dashboard and an Android mobile application, enabling timely dissemination of flood alerts to residents and local authorities. A simple linear regression model is incorporated to forecast short-term water-level trends, enhancing preparedness and response capability. The system is powered by solar energy to ensure continuous operation during power interruptions. Development followed the Rapid Application Development (RAD) methodology to support iterative prototyping and user-centered design. System evaluation using the Technology Acceptance Model (TAM) yielded an overall weighted mean of 4.52, indicating high user acceptance in terms of perceived usefulness, ease of use, and behavioral intention to use. Compliance assessment based on ISO 22328-1 resulted in an overall weighted mean of 4.45, demonstrating strong conformity with international standards for community-based early warning systems. The results confirm that the proposed FEWS effectively enhances flood preparedness, supports informed decision-making, and contributes to community resilience.

Keywords: Internet of Things, Flood Early Warning System, Linear Regression, Rapid Application Development, ESP32, LoRa

INTRODUCTION

Early warning systems (EWS) are an important mechanism to enable disaster risk reduction and improve disaster preparedness in communities at both the local and national level by providing warning information and give time to take early action, to avoid unnecessary consequences, helping reduce fatalities as well as the loss of critical assets and livelihoods. [1][2] In the past decades, the interest in implementing Early Warning Systems (EWS) to mitigate flood risks has grown as floods pose significant challenges as one of nature's most devastating disasters, making the development of accurate forecasting models complex. This issue has led to severe consequences such as crop loss, population displacement, damage to infrastructure, and disruption of essential services. [3][4] Given the severe implications of floods, it is important for governments and communities to implement highly effective early warning systems and hazard assessment strategies to mitigate the damage. [5] For flood prone areas or countries, an essential part of their governance is flood management. The necessity to continuously review and analyze the adverse or ambient environmental conditions in real-time demands developing a monitoring system so that floods could be detected beforehand. [6] Bridging this concern to the community level, Barangay Doña Imelda illustrates the persistent challenges posed by recurring floods. Its geographic characteristics and exposure to surrounding waterways highlight the need to examine localized flood risks and their implications for disaster preparedness and management.

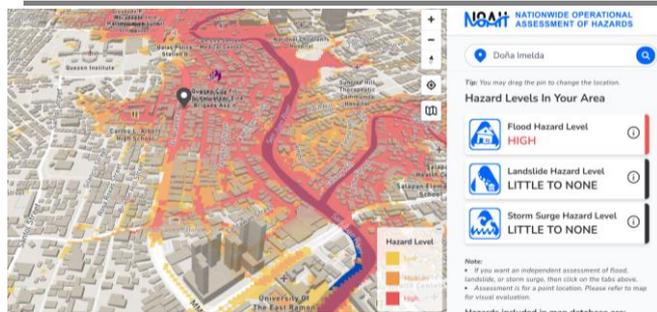


Figure 1. Barangay Doña Imelda Hazard Map [7]

One area that experiences recurrent flooding is Quezon City, where five river systems and 44 tributaries contribute significantly to flood risk. When these rivers overflow, approximately 78 areas across 35 communities become vulnerable to flooding. [8] Despite the relatively high elevation of Quezon city, it still experiences flooding due to the undulating shape of the city and being the catchment area of five river systems (San Juan, Tullahan, Marikina, Pasig and Meycauyan river), makes the risk of flooding greater for the low-lying areas. One of the affected barangays is Doña Imelda, which is a community located in the fourth district of Quezon City, Metro Manila, Philippines. It is situated near the junction of Quezon and G. Araneta Avenues, close to the renowned Sto. Domingo Church. The barangay shares its borders with the City of Manila, particularly the Santa Mesa area. [9][10][11] Established during the expansion of Quezon City, Barangay Doña Imelda was named in honor of Imelda Marcos, the wife of former President Ferdinand Marcos, who served from 1965 to 1986. The naming reflects the common practice of dedicating places to prominent figures in Philippine history. [12] Over the years, the population of Doña Imelda has experienced fluctuations. In 1990, the population was recorded at 11,543. This number saw a gradual increase, reaching 19,773 by 2010. However, subsequent years witnessed a decline, with the population decreasing to 16,915 in 2015 and further to 11,724 in 2020. [13] The barangay is located along the San Juan River, making it susceptible to overflow flooding, especially during heavy rainfall when the river backflows from its junction with the Pasig River. This geographical characteristic poses challenges to the community, particularly in terms of flood management and disaster preparedness. [14]

Figure 1 shows the hazard map of Barangay Doña Imelda, Quezon City, from the Nationwide Operational Assessment of Hazards (NOAH). It indicated that the area has a high flood hazard level, particularly in locations near the river highlighted in red, which are highly prone to flooding during heavy rainfall or river overflow. Surrounding areas in orange and yellow represent moderate to low flood risks. This information supports the observed conditions in the community and emphasizes the importance of flood preparedness and mitigation strategies. According to Mr. Joener Letada, Chairman of Disaster Response Team in Barangay Doña Imelda, the barangay is highly vulnerable to flooding, with a 95% susceptibility rate according to assessments. Flood monitoring in the area relies on traditional techniques, with water levels being measured at various street corners, as flooding intensity varies from street to street. Residents near the river are not fully aware of the river's water level due to the lack of flood markings on the river, which are only found on every street corner. To enhance preparedness, residents receive comprehensive training on what actions to take before, during, and after a flood. During floods, first responders in the barangay face significant challenges, especially when floods reach dangerous heights, as these conditions often lead to life-threatening situations, including risks of electrocution from exposed power lines. The most critical aspect of their work involves rescuing individuals trapped in their homes, often in perilous conditions. During typhoons and flooding events, hundreds of residents are rescued, highlighting the scale and severity of the issue. The devastating impact of Typhoon Carina is a stark reminder of the risks, as it brought a record-breaking flood level of 15 feet, underscoring the need for enhanced disaster response strategies and community resilience. Notably, in 2011, a flash flood event, described as "abnormal" by weather experts, hit various parts of Metro Manila, including Doña Imelda, Quezon City. The floodwaters reached waist to chest-high levels in some areas, and rescue teams used rubber boats to help residents. [15] On September 19, 2014, Heavy rains brought by Tropical Storm "Mario" (local name) caused flooding in Doña Imelda, trapping residents in their homes. [16] In July 2015, tropical Depression "Maring" brought heavy rains that caused flooding in several areas in Metro Manila, including Doña Imelda, Quezon City. Floodwaters were reported to be one foot deep in Barangay Doña Imelda. [17] On September 26, 2023: Heavy rainfall caused by a low-pressure area (LPA)

and the southwest monsoon resulted in flooding in four barangays in Quezon City, including Doña Imelda. A total of 500 families or 2,500 people were affected. [18]

The researchers aimed to offer an innovative and technology-driven approach to flood risk management to address the issues faced by Barangay Doña Imelda by developing an IoT-based flood early warning system that efficiently measures and transmits real-time water level data. It seeks to enhance flood awareness and early warning mechanisms by providing timely alerts to residents, first responders, and authorities, improving disaster preparedness and minimizing risks. The study also aimed to replace traditional monitoring methods with automated, real-time data collection to ensure accurate and up-to-date flood information across Barangay Doña Imelda. Additionally, it optimized emergency response efforts by offering real-time insights on flood levels and risk zones, aiding the barangay's Disaster Risk Reduction and Management Team. By equipping residents and first responders with precise flood information, the system helped reduce hazards like electrocution and delayed rescues, ultimately increasing community resilience and safety. Lastly, the study evaluated the system's reliability, efficiency, and overall impact on flood preparedness and response.

Related Studies

The importance of a flood monitoring system cannot be overstated. These systems offer real-time data and insights into factors like water levels, velocity, and rainfall, enabling timely warnings and immediate action. This allows authorities, communities, and organizations to make informed decisions, significantly reducing the potential loss of life and property. Flood monitoring data is also crucial for modeling flood zones to help prevent or mitigate major events and is used by insurance companies to determine flood insurance premiums. Ultimately, flood monitoring is essential for public safety, infrastructure protection, environmental preservation, and effective urban planning. [19]

The study "Real Time Flood Detection, Alarm and Monitoring System Using Image Processing and Multiple Linear Regression" discusses the design and implementation of a smart flood monitoring system aimed at improving flood detection and response. The study underscores the need for automated, real-time flood detection systems to mitigate disaster risks in flood-prone areas. By leveraging image processing and sensor-based technologies, the proposed system enhances community preparedness and resilience. [20]

There is a crucial need for flood monitoring alert systems in Malaysia, especially in areas prone to flooding according to M.I. Zakaria, et al. in their study, titled "Development of a smart sensing unit for LoRaWAN-based IoT flood monitoring and warning system in catchment areas". According to them, floods can cause severe damage to lives, infrastructure, and essential services, and traditional warning methods are often slow and ineffective. This research introduces an innovative IoT-based system that leverages LoRaWAN technology to provide continuous monitoring, early warnings, and predictive modeling. The system sends real-time alerts to residents and authorities, helping communities prepare for and mitigate flood risks. By integrating smart sensors, cloud-based data analysis, and automated notifications, this system enhances disaster preparedness, making flood response faster and more effective. [21]

Indonesia led the development of ISO 22328-1:2020 as a global framework for multi-hazard early warning systems, building on its experience with ISO 22327:2018 for landslides and expanding its scope to include hazards like floods, tsunamis, and volcanic eruptions. The article, titled "Strengthening the Resilience by Implementing a Standard for Landslide Early Warning System" affirms this standard's applicability to flood monitoring systems, emphasizing its integration of technical specifications, community-based approaches, and risk communication protocols. By adopting ISO 22328-1, flood monitoring initiatives can align with international best practices, enhance operational consistency, and ensure system scalability and effectiveness in diverse hazard contexts. [22]

The study, titled "Surigao River Flood Watch and Warning App System" study highlights the Rapid Application Development (RAD) methodology as essential in developing the flood monitoring system due to its iterative, flexible, and user-centered approach. RAD enabled the researchers to rapidly prototype and test the system components—including ultrasonic sensors, GSM modules, and the mobile app—without restarting the entire process for each change. This flexibility enabled rapid adaptation based on field test results and user feedback

from field testing and users in flood-prone communities, ensuring the system met local needs efficiently. By allowing quick updates and improvements, RAD supported the creation of a low-cost, highly functional, and responsive flood monitoring and early warning system that delivers real-time alerts and app-based monitoring—crucial for timely disaster response and community safety. [23]

DESIGN AND METHODOLOGY

Methodology

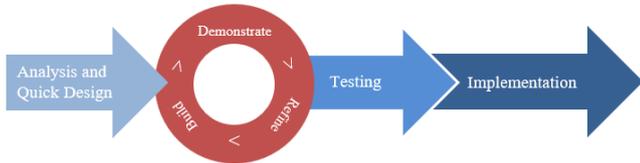


Figure 2. Rapid Application Development Methodology[24]

The study adopted the Rapid Application Development (RAD) methodology to guide the structured and iterative development of the IoT-Based Flood Early Warning System (FEWS). The methodology was organized into five major phases: Requirements Planning, User Design, Construction, Implementation, and Evaluation. Each phase produced specific deliverables that informed the succeeding stage and ensured traceability from requirements to system validation.

The Requirements Planning

This phase established the operational context, user needs, and technical scope of the proposed system. Inputs to this phase included the hazard profile of Barangay Doña Imelda, documented flood incidents, and current flood monitoring practices within the community. Consultations and interviews were conducted with barangay officials and disaster response personnel to gather firsthand information on monitoring gaps, response delays, and safety risks encountered during flood events.

Based on these inputs, the researchers analyzed the limitations of manual flood monitoring and identified the need for an automated, real-time, and remotely accessible warning system. Functional requirements such as continuous water-level sensing, automated alert generation, long-range data transmission, and multi-platform access were defined. Non-functional requirements, including reliability, low power consumption, and weather-resistant deployment, were also specified. As an output of this phase, a formal requirements specification and a defined set of hardware and software components were produced to guide the system design.

User Design Phase

During the User Design phase, the approved requirements were translated into a detailed system architecture and interface design. Using the identified components and user roles, the researchers developed the logical and physical structure of the FEWS. The architecture defined the interaction among sensing devices, the microcontroller, communication modules, cloud database, and user-facing applications.

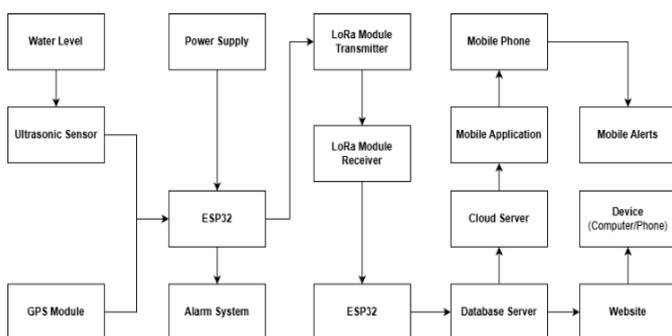


Figure 3. Block Diagram of the System

Hardware layout and wiring configurations were specified for the ultrasonic sensor, ESP32 controller, LoRa communication module, GPS module, and solar power subsystem. Data flow from water-level acquisition to cloud storage and user notification was modeled through block and architecture diagrams. In parallel, the structure of the database and server endpoints was designed to support real-time data ingestion and historical storage.

User interface designs for both the web dashboard and mobile application were drafted with emphasis on clarity, minimal navigation steps, and immediate visibility of alert status. Administrative modules for monitoring, reporting, and configuration were defined. The primary outputs of this phase included the system architecture diagrams, hardware configuration plan, database design, and interface mockups that served as blueprints for prototype development.

Construction Phase

The Construction phase focused on building, integrating, and refining the hardware and software components of the system through iterative prototyping. Guided by the design specifications, the researchers assembled the hardware prototype and conducted controlled sensor testing to verify measurement accuracy and stability. Initial tests revealed inconsistent readings from certain sensor types under varying environmental conditions, which led to component substitution and recalibration before final selection.

Firmware for the ESP32 microcontroller was developed to perform sensor reading, data preprocessing, threshold checking, and LoRa-based transmission. Concurrently, the server-side backend and cloud database were implemented to receive, validate, and store incoming measurements. The web-based dashboard was developed in modular form, including monitoring, alerts, history, records, activity logs, sensor management, and maintenance configuration features.

An Android-based mobile application was also constructed to provide real-time alerts and status updates to residents. In addition, a simple linear regression module was integrated into the server logic to compute short-term forecasts of water levels based on recent readings. Iterative testing and refinement were performed across hardware and software components. The outputs of this phase included a fully functional prototype, calibrated sensors, an operational dashboard and mobile app, and an integrated prediction feature.

Implementation Phase

The Implementation phase involved operational demonstration and controlled deployment of the completed prototype to intended stakeholders. The integrated FEWS unit, together with its dashboard and mobile application, was presented in a simulated setup representative of field conditions. Demonstration inputs included configured alert thresholds, connected sensors, and active server and application services.

To simulate flood behavior, a movable floater target was positioned within the sensing range and adjusted vertically to represent rising and falling water levels. The ultrasonic sensor measurements, LoRa transmission, database updates, dashboard visualization, alert triggering, audible alarms, and mobile notifications were observed in real time. Stakeholders were guided through the system workflow, including device operation, dashboard interpretation, and alert response features.

This phase produced verified end-to-end operational results, confirmed alert workflows, and documented observations regarding system responsiveness and usability. The demonstration outputs provided the basis for structured user evaluation.

Evaluation Phase

The Evaluation phase assessed the acceptability, usability, and standards alignment of the developed system using formal evaluation instruments. Inputs to this phase included the operational prototype, the completed demonstration session, and respondent interaction with the dashboard and mobile application.

User acceptance was measured using a survey instrument based on the Technology Acceptance Model (TAM), covering perceived usefulness, perceived ease of use, and behavioral intention to use. Respondents rated system functions, interface clarity, and perceived value in flood preparedness. A second evaluation instrument based on ISO 22328-1 criteria was administered to assess alignment with recognized early warning system standards, including risk knowledge support, monitoring and warning capability, dissemination and communication effectiveness, and response support. Collected responses were tabulated and analyzed using weighted mean scoring to derive quantitative acceptance and compliance indicators. The outputs of this phase included TAM acceptance scores, ISO-based compliance ratings, and consolidated validation metrics, which were used to support the overall effectiveness and reliability claims of the proposed FEWS.

System Architecture of the System

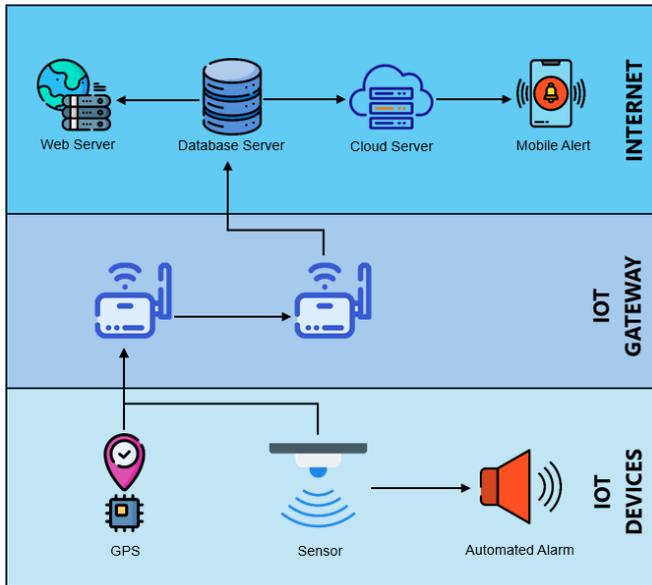


Figure 4. System Architecture of the System

The proposed FEWS consists of three primary layers: IoT Devices, IoT Gateway, and Cloud Services. The IoT device layer includes the ultrasonic sensor for water-level detection and the GPS module for location tracking. Data collected by the ESP32 microcontroller are transmitted via LoRa modules to the gateway, which forwards the data to a cloud-based database server. The cloud layer processes and stores the data, enabling access through the web dashboard and mobile application. Automated alerts and local audible alarms are triggered when predefined threshold levels are reached.

Linear Regression

To forecast short-term water levels, the system implements simple linear regression, which models the linear relationship between a predictor variable (time) and a response variable (water level). The model supports trend estimation and near-term prediction based on recent sensor readings.

For system development and algorithmic implementation, the model is formally expressed through the following sequence. While basic algebra uses $y = mx + b$, academic and computational literature adopts the Beta (β) notation, which is scalable to multiple regression models.

The core equation used in application development to compute predictions is:

$$\hat{y} = \beta_0 + \beta_1x \quad \text{Equation 1}$$

Where \hat{y} (y-hat) represents the predicted value or the output of the system for a given input x ; β_1 is the slope coefficient that quantifies the expected change in the dependent variable for every one-unit increase in the

independent variable; and β_0 is the y-intercept, representing the constant or baseline value of the system when the input x is zero.

To implement this in a system, the coefficients β_1 and β_0 are calculated from a training dataset using the following estimators:

$$\beta_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad \text{Equation 2}$$

Slope(β_1), where the numerator represents the covariance of the sample observations x_i and y_i relative to their respective means \bar{x} and \bar{y} , and the denominator represents the sum of squares for the independent variable, effectively normalizing the relationship by the variance of x .

$$\beta_0 = \bar{y} - \beta_1 \bar{x} \quad \text{Equation 3}$$

Intercept(β_0), where \bar{y} and \bar{x} are the calculated arithmetic means of the dependent and independent variables, respectively, ensuring that the resulting regression line passes through the centroid of the data distribution for mathematical balance. [25]

RESULT AND DISCUSSION

The Developed Prototype



Figure 5. Prototype of the Developed System

The prototype of the developed Flood Early Warning System (FEWS), illustrated in Figure 5, represents the hardware implementation designed to monitor, process, and transmit real-time flood data. The unit integrates an ESP32 microcontroller with an HC-SR04 ultrasonic sensor for accurate water level detection. Housed within a sturdy metal frame and PVC enclosure, the structure features a vertically mounted water-level scale marked with distinct level indicators to provide a clear visual reference. To ensure continuous operation even during adverse weather or power interruptions, the prototype is equipped with a solar panel and a rechargeable battery, offering a sustainable and energy-efficient power supply. Data collected from the sensor are processed by the microcontroller and transmitted via the Ra-01 LoRa module for long-range, low-power communication to the central server. Supporting modules such as the PAM8403 audio amplifier and speaker were integrated to produce audible warnings when the detected water level exceeds predefined thresholds, while the NEO-6M GPS module provides precise location coordinates for real-time tracking and data logging. The overall design emphasizes durability, accuracy, and reliability, making it suitable for deployment in flood-prone areas. This integrated prototype demonstrates the system's capability to deliver continuous environmental monitoring, automated alert generation, and sustainable operation, forming the foundation of the IoT-based FEWS architecture.

Web Interface

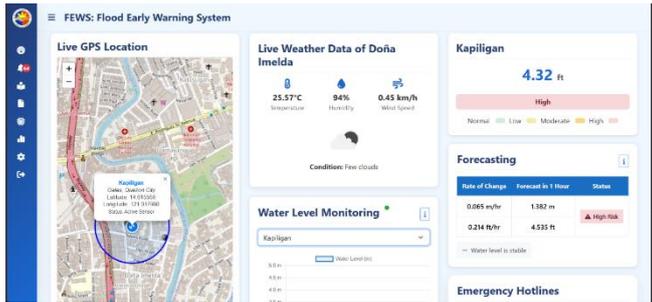


Figure 6. Dashboard of the Developed System

The web-based system, illustrated in Figure 6, serves as the administrative platform for managing, monitoring, and analyzing flood-related data. Developed using Visual Studio Code and SQLyog Ultimate, the web application is directly linked to the database that receives real-time data from the hardware subsystem. The system provides a user-friendly interface composed of several key modules, such as Dashboard, Alerts, History, Records, Activity Logs, Sensors, and Maintenance, which collectively support system oversight and data-driven decision-making. The Dashboard Module displays essential information such as the current water level, weather condition, predicted possible water level, and emergency hotlines. The Alerts Module provides warnings when thresholds are exceeded, ensuring timely dissemination of information to residents. The History Module presents graphical and tabular views of past flood data, allowing comparative analysis and trend evaluation. The Records Module enables users to generate downloadable PDF reports of sensor readings and alerts, useful for documentation and reporting. The Activity Logs Module records all user and administrative actions within the platform, promoting transparency and accountability. The Sensors Module displays a comprehensive list of all connected sensors or devices, allowing the administrator to update sensor information, manage configurations, or remove inactive devices as necessary. Lastly, the Maintenance Module allows the administrator to modify system parameters such as prediction intervals, data collection frequency, and interface customization settings, including background color and logo adjustments. The web system's modular structure ensures that all administrative tasks, from monitoring and evaluation to reporting, can be performed efficiently through an integrated interface.

Mobile Application

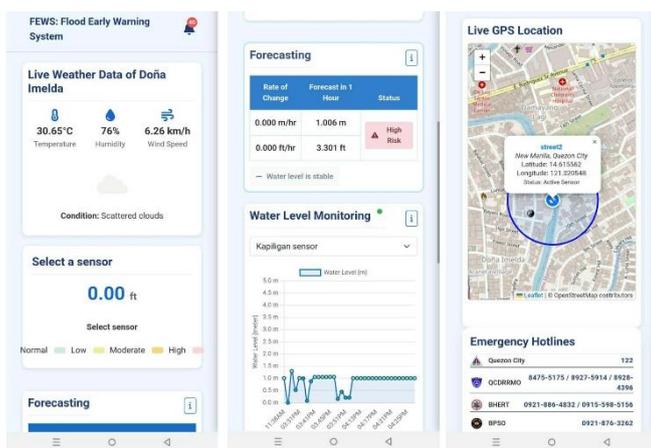


Figure 7. Mobile Application of the Developed System

As shown in Fig. 7, the mobile application functions as the primary user-facing platform for delivering flood alerts and system updates to the community. The application was developed using Android Studio to ensure compatibility across a wide range of Android devices commonly used by residents. It is integrated with the cloud database, allowing users to view real-time water-level readings, receive automated alert notifications, and monitor current river conditions.

The interface was designed with emphasis on accessibility, clarity, and operational reliability, enabling users to quickly interpret warning information with minimal interaction steps. When a threshold level is exceeded, the system automatically sends push notifications that include the alert category, timestamp, and recommended safety actions. The application is optimized to operate efficiently under low-bandwidth conditions, ensuring that critical warnings are delivered without significant delay. Through timely and accurate mobile notifications, the application strengthens community preparedness and directly supports the FEWS objective of improving early warning dissemination and life safety.

Linear Regression Implementation

The proponents used a PHP function to implement simple linear regression in the system using the figure below. By multiplying the slope by 60 (minutes for the 1-hour prediction), the system can predict what the water level will be in the next hour.

Building on this, the Forecasting section of the FEWS interface integrates the results of the linear regression model by displaying the computed rate of change and the corresponding one-hour water level prediction. The calculated slope, derived from recent water level readings, represents the rate at which the water level is increasing or decreasing over time. This slope is then multiplied by 60 to project the water level one hour into the future. The system automatically converts the predicted value into both meters and feet for clarity, and classifies the risk level (e.g., High Risk) based on predefined threshold values. This dynamic presentation ensures that users can easily interpret both the predicted water behaviour and the associated risk level in real time.

```

40 function regressionForecast(array $times, array $levels, int $window = 6, int $ahead = 60): array {
41     if (count($times) < $window || count($levels) < $window) {
42         return ['slope_hr' => 0, 'forecast' => 0, 'intercept' => 0];
43     }
44
45     $t_slice = array_slice($times, -$window);
46     $h_slice = array_slice($levels, -$window);
47
48     $m = $window;
49     $mean_t = array_sum($t_slice) / $m;
50     $mean_h = array_sum($h_slice) / $m;
51
52     $num = 0.0;
53     $den = 0.0;
54     for ($i = 0; $i < $m; $i++) {
55         $dt = $t_slice[$i] - $mean_t;
56         $dh = $h_slice[$i] - $mean_h;
57         $num += $dt * $dh;
58         $den += $dt * $dt;
59     }
60
61     if ($den == 0) {
62         return ['slope_hr' => 0, 'forecast' => end($h_slice), 'intercept' => 0];
63     }
64
65     $slope_per_min = $num / $den;
66     $intercept = $mean_h - $slope_per_min * $mean_t;
67
68     $t_future = end($t_slice) + $ahead;
69     $forecast = $intercept + $slope_per_min * $t_future;
70
71     return [
72         'slope_hr' => $slope_per_min * 60,
73         'forecast' => $forecast,
74         'intercept' => $intercept
75     ];
76 }

```

Figure 8. Linear Regression Implementation Source Code

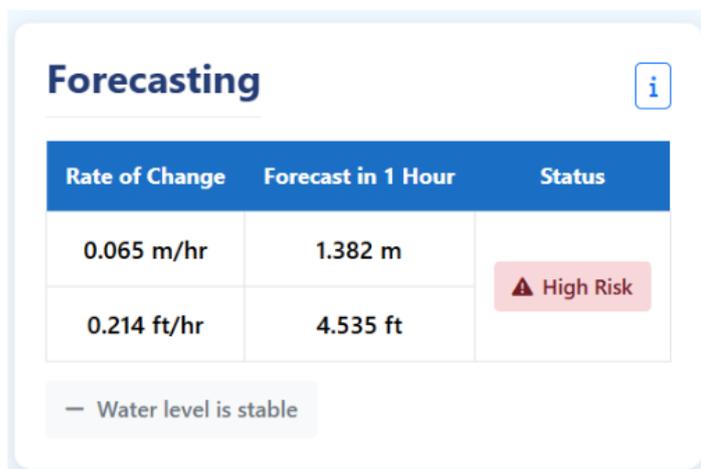


Figure 9. Predicted Water Level Forecasting

User Evaluation

Table I User Acceptance Test Result

Factor	Weighted Mean	Interpretation	Rank
Perceived Usefulness	4.62	Strongly Agree(SA)	1
Perceived Ease of Use	4.42	Agree(A)	3
Behavioral Intention of Use	4.52	Strongly Agree(SA)	2
Over-all Mean	4.52	Strongly Agree(SA)	

Table I presents the user acceptance results using the Technology Acceptance Model (TAM). The results indicated excellent performance, with an average weighted mean of 4.62 for perceived usefulness, 4.42 for perceived ease of use, and 4.52 for behavioral intention to use, resulting in an overall weighted mean of 4.52. These results demonstrate a high level of acceptance and satisfaction among users, confirming that the developed system is effective, easy to use, and valuable in supporting flood monitoring and alert dissemination.

Table II User Evaluation Result

Factor	Weighted Mean	Interpretation	Rank
Risk Knowledge	4.50	Agree(A)	2
Dissemination and Communication	4.46	Agree(A)	3
Monitoring and Warning Service	4.53	Strongly Agree(SA)	1
Response Capability	4.33	Agree (A)	4
Over-all Mean	4.45	Agree(A)	

Table II presents the user evaluation result using ISO 22328-1 Standard. The results showed the following average weighted means for each characteristic: Risk Knowledge, 4.50; Monitoring and Warning Service, 4.53; Dissemination and Communication, 4.46; and Response Capability, 4.33, with an overall weighted mean of 4.45. These results indicate that the developed Flood Early Warning System conforms to the standards of ISO 22328-1, ensuring its reliability, efficiency, and alignment with international guidelines for community-based early warning systems.

CONCLUSION

The IoT-based Flood Early Warning System (FEWS) developed in this study effectively addresses the flood monitoring and alerting needs of Barangay Doña Imelda. By integrating real-time sensing, long-range communication, predictive analytics, and user-centered interfaces, the system enhances disaster preparedness and response capability. Evaluation results confirm high user acceptance and strong alignment with ISO 22328-1 standards. The proposed FEWS provides a scalable and sustainable solution that can be adapted for deployment in other flood-prone communities, contributing to improved resilience and public safety.

REFERENCES

1. B. Golding, "Towards the 'Perfect' Weather Warning," 2022.

2. M. Sahana *et al.*, “Assessing the effectiveness of existing early warning systems and emergency preparedness towards reducing cyclone-induced losses in the Sundarban Biosphere Region, India,” *International Journal of Disaster Risk Reduction*, vol. 90, May 2023, doi: 10.1016/j.ijdr.2023.103645.
3. M. J. Henao Salgado, L. Alfonso, and J. J. Vélez Upegui, “Towards integrating community and institutional flood early warning systems: A framework applied to an Andean tropical case,” *International Journal of Disaster Risk Reduction*, vol. 116, Jan. 2025, doi: 10.1016/j.ijdr.2024.105126.
4. M. Wajid, M. K. Abid, A. Asif Raza, M. Haroon, and A. Q. Mudasar, “Flood Prediction System Using IOT & Artificial Neural Network,” *VFAST Transactions on Software Engineering*, vol. 12, no. 1, pp. 210–224, Mar. 2024, doi: 10.21015/vtse.v12i1.1603.
5. P. Mysak and I. Mysak, “A COMPREHENSIVE REVIEW OF EMERGING TECHNOLOGIES FOR FLASH FLOOD PREDICTION INCLUDING: AI, ML, IoT APPROACHES,” *European Science*, no. sge32-03, pp. 73–83, Sep. 2023, doi: 10.30890/2709-2313.2024-32-00-012.
6. M. Siddique, T. Ahmed, and M. S. Husain, “Flood Monitoring and Early Warning Systems – An IoT Based Perspective,” Jul. 31, 2023, *European Alliance for Innovation*. doi: 10.4108/eetiot.v9i2.2968.
7. Nationwide Operational Assessment of Hazards, “Know Your Hazards.”
8. UNDRR, “2010-2011 World Disaster Reduction Campaign.” Accessed: Dec. 16, 2024. [Online]. Available: <https://www.un.org/en/observances/disaster-reduction-day>
9. Wikipedia, “List of barangays in Quezon City.” Accessed: Feb. 07, 2025. [Online]. Available: https://en.wikipedia.org/wiki/List_of_barangays_in_Quezon_City
10. Quezon City Planning and Development Department, “2022 Quezon City Ecological Profile,” 2020.
11. State-of-the-Art Emergency Operations Center, “2018 QUEZON CITY ECOLOGICAL PROFILE CHAPTER 7: RISK PROFILE,” 2018.
12. S. Samonte, “Quezon City: Home to beautifully-named barangays.” Accessed: Feb. 07, 2025. [Online]. Available: <https://www.pna.gov.ph/articles/1046787>
13. Citypopulation.de, “Doña Imelda Barangay in Quezon City.” Accessed: Feb. 07, 2025. [Online]. Available: https://www.citypopulation.de/en/philippines/quezoncity/137404031__do%C3%B1a_imelda/
14. Quezon City Planning and Development Department, “2018 Ecological Profile of Quezon City,” 2018, Accessed: Feb. 07, 2025. [Online]. Available: https://quezoncity.gov.ph/wp-content/uploads/2020/09/2018-QC-Ecological-Profile_11-11-2022.pdf
15. Inquirer.net, “Flash floods swamp Metro.” Accessed: Dec. 16, 2024. [Online]. Available: <https://shorturl.at/sdxZV>
16. Inquirer.net, “Trapped in Flood.” Accessed: Dec. 16, 2024. [Online]. Available: <https://shorturl.at/zW2n4>
17. GMA Network, “Flooded areas in Metro Manila.” Accessed: Dec. 16, 2024. [Online]. Available: <https://shorturl.at/7jh4Q>
18. Reliefweb, “Philippines DSWD DROMIC Report on the Flooding Incident in Quezon City, 26 September 2023, 6PM.” Accessed: Dec. 16, 2024. [Online]. Available: <https://tinyurl.com/ys3w77vz>
19. YSI xylem, “Flood Monitoring Systems: Building Block of Resilience.” Accessed: Oct. 26, 2025. [Online]. Available: https://www.ysi.com/applications/flooding?srsId=AfmBOopVSmCRm2ALd2sHOd4n88BhmS_35JriJxP-Ov7XHbC40yrXPpDR
20. L.S. Karlo Tolentino *et al.*, “Real Time Flood Detection, Alarm and Monitoring System Using Image Processing and Multiple Linear Regression,” 2022.
21. M. I. Zakaria, W. A. Jabbar, and N. Sulaiman, “Development of a smart sensing unit for LoRaWAN-based IoT flood monitoring and warning system in catchment areas,” *Internet of Things and Cyber-Physical Systems*, vol. 3, pp. 249–261, Jan. 2023, doi: 10.1016/j.iotcps.2023.04.005.
22. T. F. Fathani, D. Karnawati, W. Wahyu, and H. Setiawan, “Strengthening the Resilience by Implementing a Standard for Landslide Early Warning System,” 2023.
23. E. Mag-Usara, “Surigao River Flood Watch and Warning App System,” *International Research Journal of Advanced Engineering and Science*, vol. 7, no. 1, pp. 263–269, 2022.
24. maybe.works, “Rapid App Development: Full RAD Methodology Overview and Practical Advice.” Accessed: Oct. 26, 2025. [Online]. Available: <https://maybe.works/blogs/rapid-application-development>
25. J. Waples, “Simple Linear Regression: Everything You Need to Know.” Accessed: Jan. 13, 2026. [Online]. Available: <https://www.datacamp.com/tutorial/simple-linear-regression>