**PROGRESS IN ENGINEERING APPLICATION AND TECHNOLOGY**
e-ISSN: 2773-5303

**PEAT**

Vol. 7 No. 1 (2026) 298-306
https://publisher.uthm.edu.my/periodicals/index.php/peat

# Development of Flood Risk Prediction Model Using Water Level and Rainfall Volume Using Multiple Regression Analysis

**Brian Law Jia Ching[1], Azlina Bahari[1*], Aimi Syamimi Ab Ghafar[1], Muhammad Faizal Ismail[1]**

[1] *Department of Electrical Engineering Technology, Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, Pagoh, 84600, Muar, Johor, MALAYSIA*

*Corresponding Author: lina@uthm.edu.my
DOI: https://doi.org/10.30880/peat.2026.07.01.034

**Article Info**

Received: 12 January 2026
Accepted: 20 February 2026
Available online: 25 May 2026

**Keywords**

Flash flood, Multiple Linear Regression, IoT, MySQL

**Abstract**

Flash floods are a very serious natural risks that usually happened in the tropical areas like Malaysia whereby heavy rainfall causes a quick rise in the surface water levels. Traditional predictive models often rely on single variable indicators, which easily leads to slow or inaccurate hazard assessment. The current research developed a flood risk model, the implementation of involving Multiple Linear Regression (MLR) models along with the use of real-time Internet of Things (IoT) data collection. Tipping bucket rainfall sensor and ultrasonic sensor were used to collect sensor data thus send them using LoRaWAN, and Wi-Fi to a central cloud-based MySQL database. The data streams were treated by a Python based analytical pipeline to make predictions of the flood risk index and classification. The results were sent to the final users over Telegram notification bot and a real-time web dashboard named "FloodWatchAI". The model was validated on twenty consecutive days empirical observations in Panchor, Johor in Malaysia, collaborating with SENA Traffic Sdn. Bhd and obtained an $R^2$ coefficient of 0.67 on the test set. The system proved to be associated not only with the practically immediate processing of data and spreading of alerts but also with significantly increased chances of the timely evacuation and proper management of the disaster.

## 1. Introduction

Flash flood is defined as the accumulation of water within a water body and the overflow of excess water onto adjacent floodplain land, known as low-lying areas. It is classified as one of the geo-hazards, a natural phenomenal that typically triggering the natural disasters that cause fatalities, damages, and severe impacts to social and economic [1]. Flash flood often happen in tropical country such as Malaysia. The impact causing the residents in Malaysia forced to evacuate from their residential to evacuation centres.

As flash flood continues to rise at an alarming rate, flash flood prediction is needed urgently to take precautions of the excessive rise water level and rainfall volume monitoring to forecast the incoming weather. Malaysia, one of the tropical region countries on the equator line, has yearly precipitations between 2000 mm and 4000 mm that fall all over the country, with temperatures ranging between 26 °C and 32 °C. Generally, Malaysia has a uniform temperature, high humidity, and abundant rainfall all year round [2]. Therefore, water level and rainfall volume monitoring is crucial as to trigger the alerts based on the rainfall thresholds in mm per hour.

The flash flood prediction model consists of hardware and software combined to forecast the upcoming flash flood risks. This allows the user to acquire real-time data of water level and rainfall volume via hardware thus required machine learning such as Python to insert calculation into formula to calculate the risk predictions. However, only user can acknowledge the prediction risk ahead of schedule than other authorities due to the lack of technology to provide public accurate information about the flood occurrence. To overcome this challenge, alert systems via cloud-based alert notification, Telegram is designed to notify authorities that might be affected.

### 1.1 Problem Statement

In recent years, flash floods have become the most destructive natural disaster that happened often due to global climate change especially in Southeast Asia such as Malaysia. Malaysia is prone to floods, with its low-lying coastal areas, concentrated development areas, and mountainous regions. The economic activities of plantations make them vulnerable to flash floods. The country experiences monsoon seasons, with heavy rainfall during the North-East monsoon from November to March and the South-West monsoon from May to September [3]. Flash floods pose significant risks to communities, infrastructure, and ecosystems, requiring accurate early warning systems for effective disaster response. Traditional flood prediction models primarily rely on single-variable indicators such as water level or rainfall volume in isolation but not as both together, which may lead to inaccuracy or delayed predictions, which could cause less preparation for affected authorities or calamities to evacuate. The time gap to provide communication to affected authorities is vital regardless of the presence of real-time flood monitoring if there is absence of early warning systems capable of providing alert promptly. In order to overcome these challenges, this project will develop an early warning systems of flash flood prediction using multiple regression analysis to predict flood through both predicted water level, actual water level and rain volume as variables to alert affected area and authorities. These initiatives are mended to reduce the damage that will cause by the flash flood such as economic loss, property damages, loss of life etc. A study showed that machine learning (ML) such as multiple regression analysis is widely used in flood prediction that could hold significance in the uncertainty assessment of flood risk [4].

### 1.2 Objectives

The following objectives will be obtained within the context of this study. Firstly, to develop a cloud-based database that can gather water level and rainfall volume data from existing sensor data. Therefore, to develop a flash flood prediction model using multiple regression analysis, combining historical and real-time water level and rainfall volume data as predictive variables. Lastly, to evaluate the accuracy and efficiency of the prediction model by comparing real-time forecasts with actual flood events and accessing system responses times.

## 2. Related Works

This section offered an organized framework for this project, covering the challenges and scopes in creating a functional IoT-based machine learning model. This project focuses on a flash flood warning system in the Philippines that utilizes ultrasonic sensors to monitor water level and speed. It provides early warnings to registered users via automated SMS alerts when calculated risk thresholds are exceeded [5]. This study enhances flood forecasting by integrating real-time sensor data, such as humidity and temperature, with an Artificial Neural Network (ANN) model. A Raspberry Pi 3 is used as the central hub to control the sensors and upload monitoring data to the ThingSpeak Cloud platform for analysis [6]. This research proposes an IoT-FSP model that employs tree-based algorithms like Decision Tree and Random Forest to predict river flood status. The system is developed using MATLAB and Simulink to simulate online data transmission and classify flood risk based on multivariate environmental data [7]. This system utilizes a multi-layer ANN to process data from tipping bucket rain gauges, water level sensors, and soil moisture sensors. It monitors rainfall intensity and soil saturation levels in real-time to provide robust predictions via a cloud server [8]. The FLOODWALL system uses Long Short-Term Memory (LSTM) networks and IoT technology to forecast flash floods based on parameters like wind speed and humidity. It delivers real-time monitoring and automated alerts to users through a dedicated mobile application and SMS messages [9].

## 3. Methodology

The methodology further explaining the development of flood risk prediction using multiple regression analysis, collaboration with SENA Traffic Sdn. Bhd, including the software system flowchart and coding flowchart. It provides a comprehensive breakdown of all the tools, resources, methods, and techniques used on how the sensor data sent through end-to-end database connection, fetched for machine learning model testing and validation.

Figure 1 shows the block diagram of overall software setup of end-to-end database connection where the sensor data from ultrasonic and tipping bucket sensor are fetched to The Things Network gateway and sent to Node-RED for database integration to MySQL database for storing data. The stored data are subsequently retrieved by python for machine learning purposes. Streamlit and Telegram are used for dashboard viewing and alert notification regarding the situation and classification of flood risk in Panchor, Johor, Malaysia.

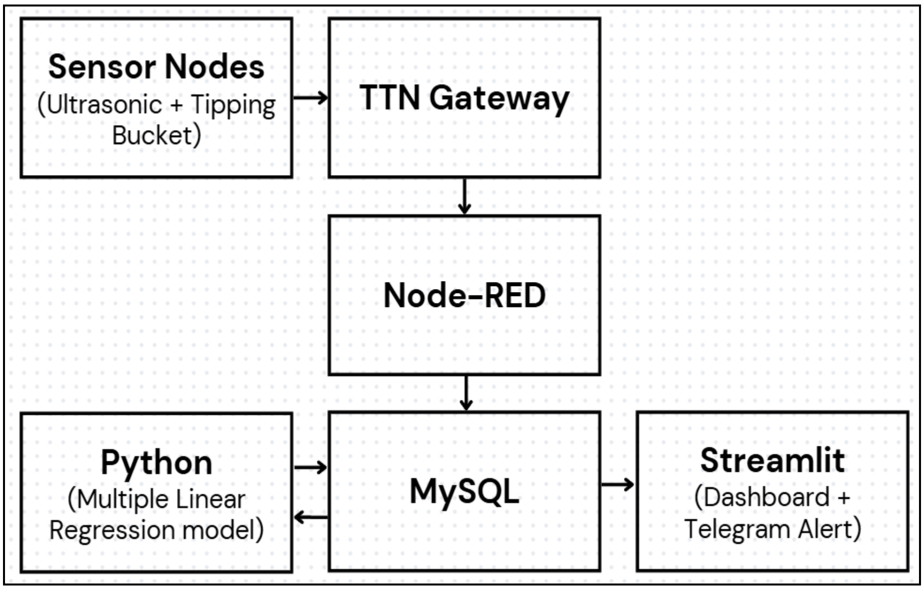


Figure 1: Block Diagram of software development

The flowchart shows the process of building a machine learning model by fetching sensor data from MySQL database. The process begins by attempting to establish a connection to a MySQL database. The system logs the error and ends the process if the connection fails. The system checks the query configuration to retrieve previous training data after a successful connection. The three main variables in this data are the timestamp, rain volume, and distance. For simpler manipulation, this raw data is subsequently organized into a DataFrame. Once the data is structured, it undergoes a reprocessing stage where the "Distance" measurement from the ultrasonic sensor (in cm) is converted into an actual water level using the formula:

$$\text{Water level (cm)} = 162\text{cm} - \text{distance}$$

The timestamps are also transformed into a common datetime format. The system's core computes a Flood Risk Index (Y) using the Ordinary Least Squares (OLS) regression method. Based on the calculated flood risk index, the system is classified into 4 risk levels where Low: <1.49, Average: 1.5 to 2.49, High: 2.5 to 3.49, Critical: >3.5. This system implements a time-split test using 20 days of obtained data. The datasets are divided into training, validation and testing sets. For these datasets, it then determines the actual and predicted risk levels. Lastly, the system assesses its own performance by producing statistical metrics (MSE, RMSE, MAE, and $R^2$) and visualizing the outcomes with analysis graphs.

The system responds to external notifications through the first processing path (Path A). Alert notifications are sent via a Telegram bot configured with a specific bot token and group chat ID. These alerts include the latest sensor readings and the model-determined Risk Level classification. The system is configured to dispatch real-time alerts at five-minute intervals to ensure timely updates. For data persistence and visualization, the system stores the processed datasets and their corresponding classifications in a MySQL database, which is updated every five minutes. The stored data are subsequently uploaded to a Streamlit-based web dashboard, providing a centralized platform for real-time monitoring and visualization. In Path C, the Python execution is terminated either when the user manually stops the process or when execution is redirected due to an initial connection failure.

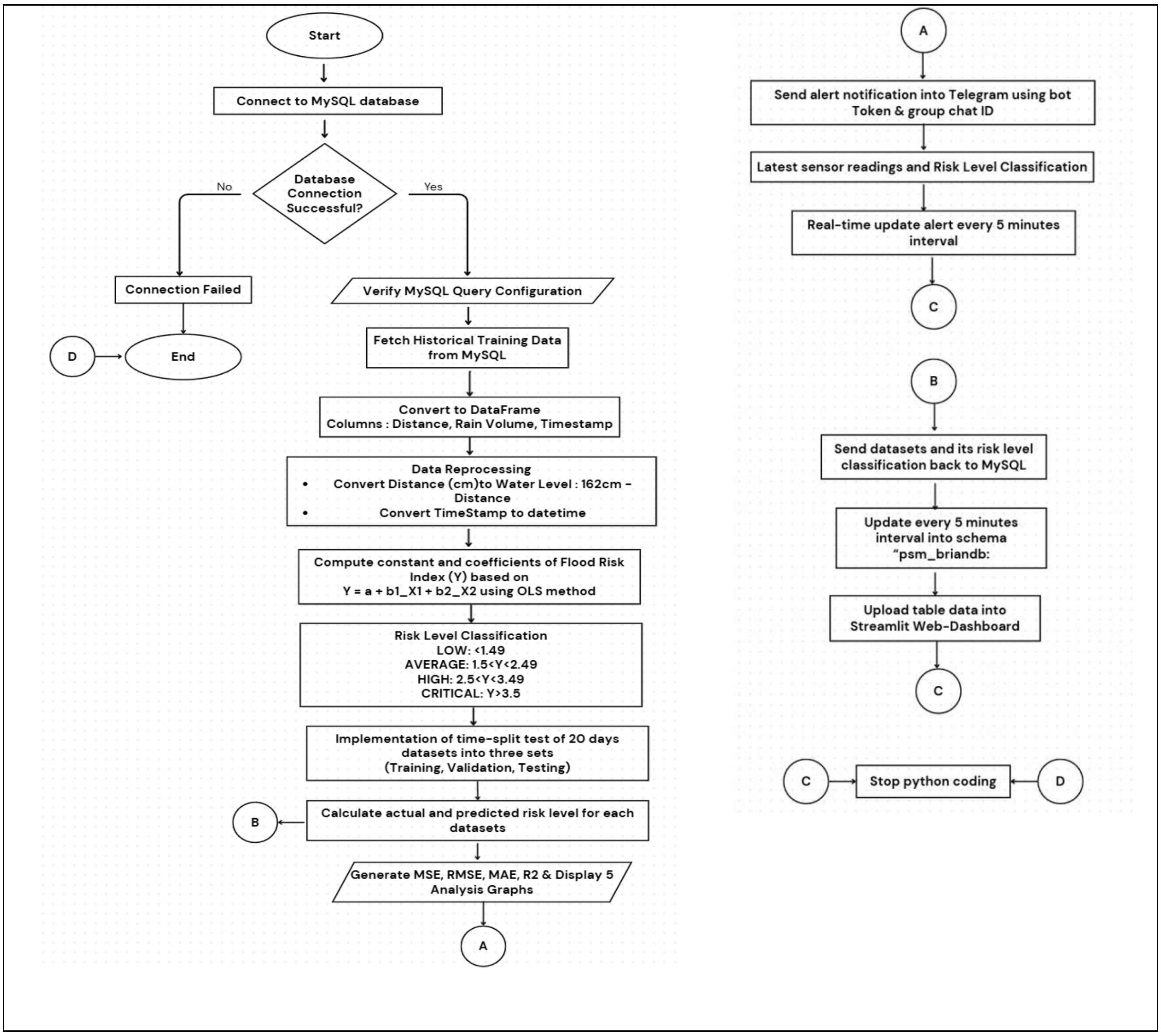


Figure 2: Coding flowchart of flood risk prediction using multiple regression

## 4. Results and Discussion

The sensor data were successfully transmitted into MySQL database and retrieved using python coding to compute Risk Index (Y) using the OLS method shown in Figure 3. The coefficients for water level and rainfall volume are 0.0174 and 0.0124 respectively. This shows that for every 1 cm increase in water level and 1 mm increase in rainfall volume, the Risk Index (Y) will increase by 0.0174 and 0.0124 respectively. Mathematically, this explains that water level has a slightly greater impact on the risk index than the rainfall volume for every increment. The rainfall coefficient, on the other hand, ensures that the model remain proactive because rising rainfall will increase the Risk Index and may cause early warnings before the river reaches a critical overflow state. The value of 0.3895 for the constant is the baseline risk of the Risk Index (Y), means that this constant is the starting value of Risk Index (Y) if both water level and rainfall volume are zero.

```
Regression Equation:
Y = 0.3895 + (0.0174 * X1) + (0.0124 * X2)
---------------------------------------------
Where:
X1 = Water Level (cm)
X2 = Rainfall Volume (mm)
```

Figure 3: Equation of Risk Index (Y) computed using OLS method

The 20 days of datasets were separated into 60%, 20%, and 20% for training, validation and testing sets respectively using time-split series as shown in Figure 4.

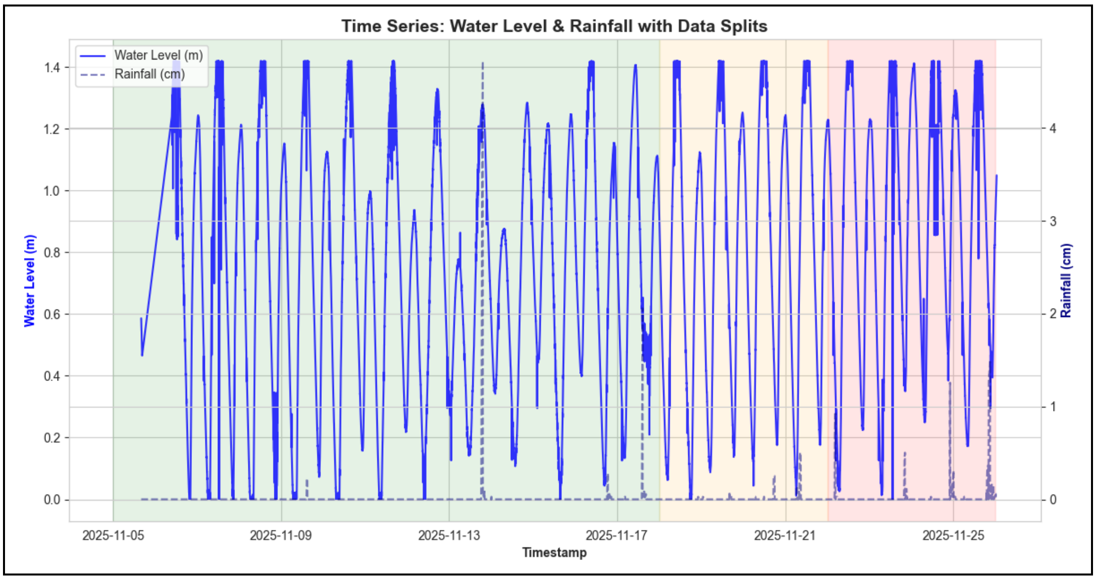

Figure 4: Graph plot of water level and rainfall volume in time series

Figure 4 presents the datasets of water level and rainfall volume in time series separated into three different sets which are Training (green), Validation (yellow), and Testing (red) from timestamp of 5 November 2025 to 25 November 2025. Measurements of rainfall and water level that were synchronized and converted into minute-level intervals made up the integrated dataset. The data was divided chronologically into training (Nov. 5–17), validation (Nov. 18–21), and testing (Nov. 22–25) phases to guarantee the predictive model's robustness. The Time Series Analysis shows that while rainfall events were marked by distinct, high-intensity spikes, the water level showed cyclic fluctuations. This temporal split guarantees that future data points that were not included in the initial training set are used to assess the model's capacity to predict risk.

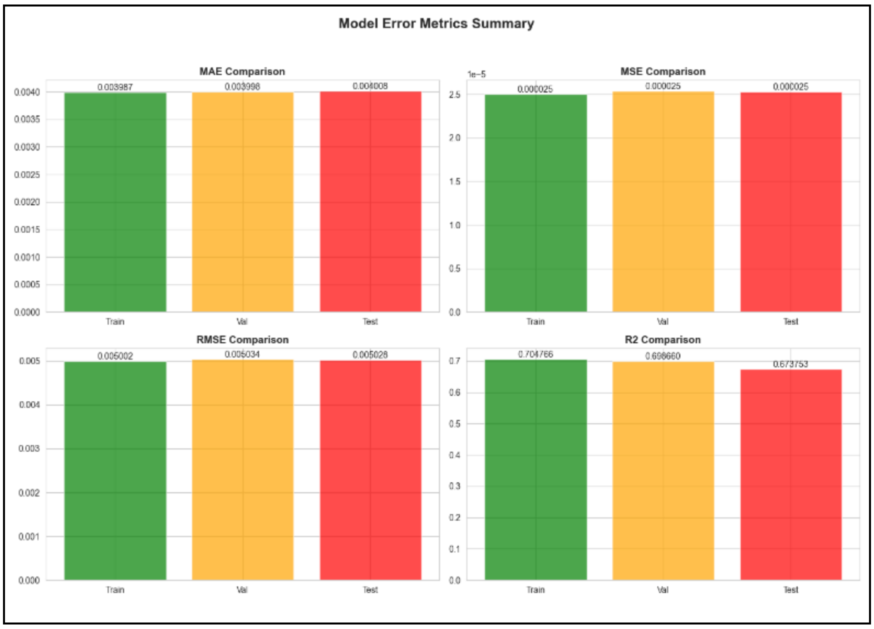

Figure 5: Comparison of values of metrics between Train, Validation, Test sets

Figure 5 shows the values of metrics such as Mean Absolute Error (MAE), Mean Square Error (MSE), RMSE (Root Mean Square Error), and R-Squared (R^2) for the Training set, Validation set, and Testing set. By obtaining a $R^2$ value of 0.67 for testing set, this demonstrates that the Test set is quite strong for the environmental modelling. In other hand, this model can explain up to 67% of the variance in flood risk by using only hourly rainfall volume and water level. The difference of $R^2$ between the test set and train set is approximately 0.03 only, indicating the model is not overfitted. The higher the $R^2$ value closer to 1, the more the model gets to perfect.

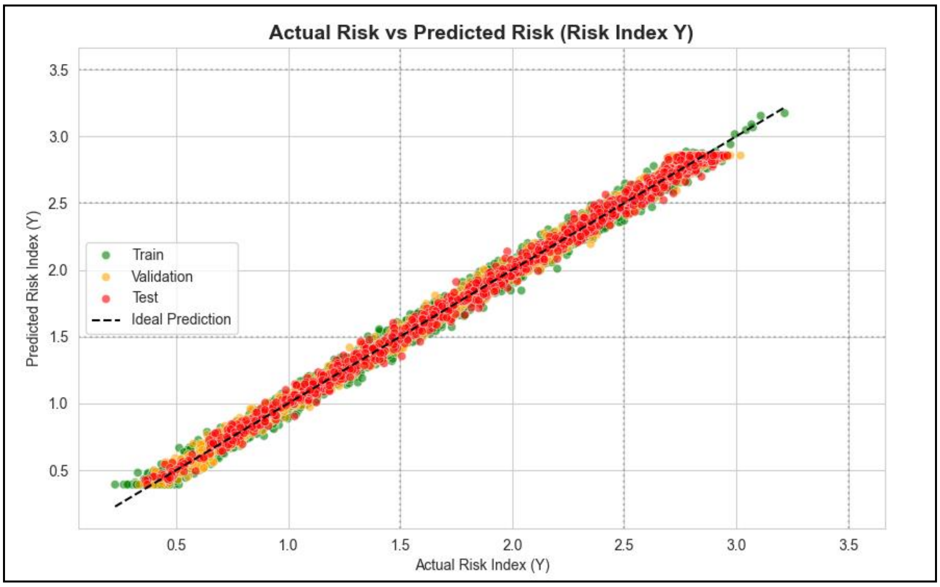


Figure 6: Graph of Actual Flood Risk vs Predicted Flood Risk

Figure 6 illustrates the graph of relation between actual flood risk values and predicted flood risk values generated by the baseline Multiple Linear Regression (MLR) model. The following green, yellow and red colour data points are Train, Validation, and Test set respectively. The x-axis represents the Predicted Flood Risk and y-axis represents the Actual Flood Risk. The black dashed line ($y=0.3895+0.0174x_1+0.0124x_2$) represents the ideal prediction line where perfect predictions occur, the prediction flood risk matches perfectly with the actual flood risk. Most of the data points are clustered near to the ideal prediction line. This indicates the model successfully captures the general trend between dependent variables (water level and rainfall volume), and independent variable (flood risk).

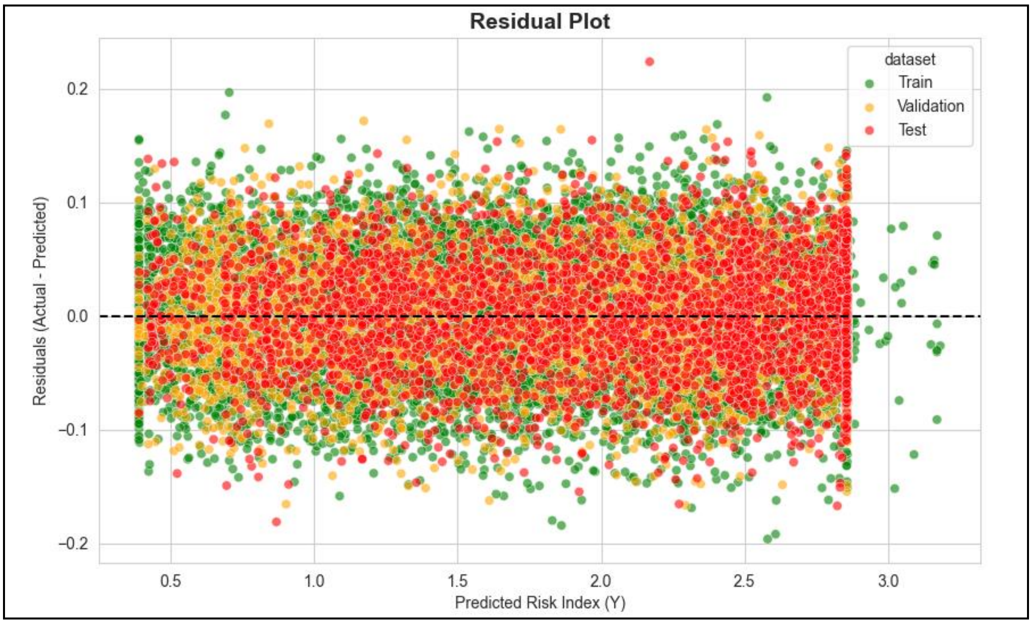


Figure 7: Graph of Residual Plot (Actual – Prediction)

Figure 7 is the graph of residual plot that shows the difference between the actual risk values and the predicted risk values, where mistakes were made by the model. The horizontal dashed line at 0.00 represents the perfect prediction. Points that lying above 0.00 usually means that the actual flood risk was higher than the predicted flood risk, shows that the model has under-predicted the values. The points that lying below 0.00 means that the actual flood risk values were lower than that of the predicted, indicating the model has over-predicted. Since most of the data points are close to the zero line within ±0.1 scattered randomly without forming any specified shape such as U-shaped, the model can be considered accurate due to able to capture all the main information in the data from the datasets provided computing in python.

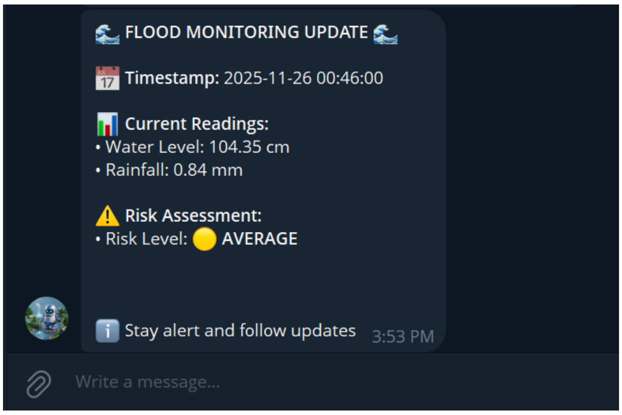


Figure 8: Notification received in Telegram

Figure 8 shows the messages received in the Telegram group displaying the latest sensor readings and the corresponding risk level, where the status "AVERAGE" is determined based on the predicted risk value calculated by the model. The system response time, including data transmission, risk prediction processing, and message delivery to Telegram, is effectively instantaneous, indicating no observable alert delay.

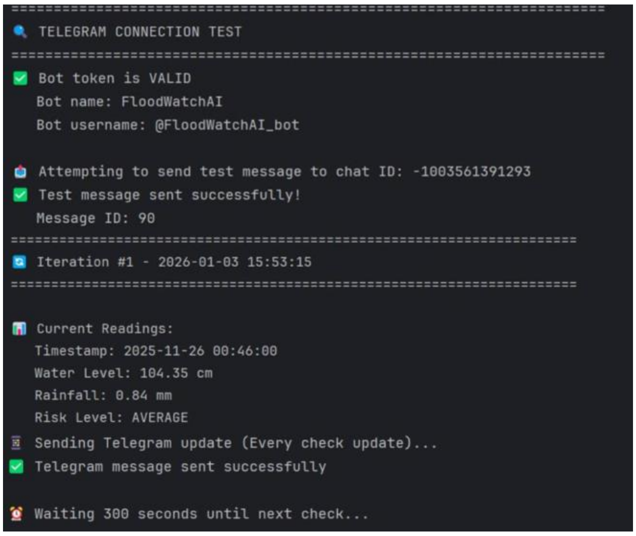


Figure 9: Output console of transmitted data to Telegram Bot

Based on Figure 9, the data transmission timestamp occurs at 3:53 p.m., and the Telegram notification is received within the same minute, as indicated at the bottom right of the message in Figure 8. This demonstrates the robustness and reliability of the system, ensuring that alerts are delivered in real time, allowing users to take timely action and plan for evacuation when the risk level reaches a critical threshold.

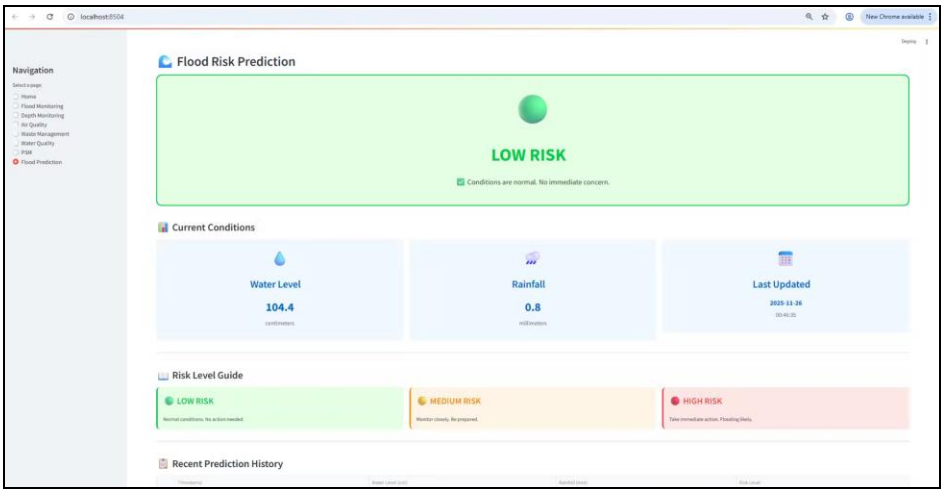


Figure 10: Dashboard for Flood Risk Prediction

Figure 10 shows the dashboard for Flood Risk Prediction under Environmental Monitoring Dashboard. The dashboard consists of current conditions for latest risk level, water level and rainfall volume at timestamp 26 November 2025. The risk level was calculated and classified into “LOW RISK” level (green) by the MLR model that are trained and tested in 20 days. In this dashboard, for better viewing, the LOW and AVERAGE risk level is combined into LOW RISK as there should be no flood concern. Therefore, there are only three risk level shown, which are “LOW” (green), “MEDIUM” (yellow), “HIGH” (red). Residents or communities that live in Panchor, Johor, Malaysia can not only receive messages from telegram but also able to watch the flood indicators through this domain “FloodWatchAI” to follow up update on their river status.

## 5. Conclusion

In conclusion, all project goals have been successfully met by the development of the flood risk prediction model using Multiple Linear Regression (MLR) analysis. The study produced a working cloud-based infrastructure that easily combines a MySQL database for real-time data collection with Internet of Things hardware. When tested against actual data from Panchor, Johor, the MLR model showed a strong ability to classify flood risks by using water level and rainfall volume as primary predictive variables. It achieved a testing accuracy of $R^2$ 0.67. Additionally, the instant alert delivery with no delays via the Telegram notification bot and the "FloodWatchAI" web dashboard validated the system's effectiveness. This system, including hardware and software offers a dependable alert notification system that can greatly improve community preparedness and response during flash flood events, making it a scalable and affordable disaster mitigation solution.

## 5.1 Recommendations

While the current system is effective for real-time monitoring, it can be significantly enhanced through several technical and functional upgrades. Future improvements should focus on replacing linear models with advanced architectures like Long Short-Term Memory (LSTM) or Artificial Neural Networks (ANN) to better capture complex hydrological trends. Accuracy can be further refined by integrating additional environmental variables such as soil moisture and water flow velocity. The accuracy and efficiency can be further enhanced by training the model on years of data to prevent false alarms during sudden weather spikes. On the user-facing side, the "FloodWatchAI" dashboard could utilize Geographical Information Systems (GIS) for localized spatial visualization, while a dedicated mobile app with location-based push notifications would improve the speed and relevance of emergency alerts.

## Acknowledgement

The author would like to express his sincere appreciation to three supervisors, Ts. Azlina Binti Bahari, Dr. Aimi Syamimi Binti Ab Ghafar, and Dr. Muhammad Faizal Bin Ismail for their guidance, constructive feedback, and continuous support throughout this research.

Appreciation is also extended to the Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia (UTHM), for providing the facilities and resources required for this study. The author also acknowledge the contributions of researchers and developers in the fields of flood forecasting, cloud-based systems, and Internet of Things (IoT), which served as valuable references for this work.

## References

[1] F. S. Buslima, R. C. Omar, T. A. Jamaluddin and H. Taha, "Flood and flash flood geo-hazards in Malaysia," International Journal of Engineering & Technology, pp. 760-764, 2018.

[2] N. Bahari, S. A. Mohammad, M. R. M. Esa, M. R. Ahmad, N. A. Ahmad and Z. A. Malek, "Analysis of lightning flash rate with the occurrence of flash floods and hailstorms in Peninsular Malaysia," *Journal of Atmospheric and Solar-Terrestrial Physics,* pp. 1-13, 2023.

[3] M. S. H. Saad, M. I. Ali, P. Z. Razi, N. I. Ramli and R. P. Jaya, "Exploring the Factors and Impacts of Flash Floods Vulnerability in Various Areas of Malaysia: A Content Analysis," *Disaster in Civil Engineering and Architecture,* vol. 1, no. 1, pp. 55-82, 2024.

[4] H. Byaruhanga, D. Kibirige, S. Gokool and G. Mkhonta, "Evolution of flood prediction and forecasting models for flood early warning systems: A scoping review," *Water,* vol. 16, no. 13, p. 1763, 2024.

[5] J. T. de Castro, G. M. Salistre Jr, Y. C. Byun and B. D. Gerardo, "Flash flood prediction model based on multiple regression analysis for decision support system," *In Proceedings of the World Congress on Engineering and Computer Science,* pp. 23-25, 2013.

[6] S. Bande and V. V. Shete, "Smart flood disaster prediction system using IoT & neural networks," In 2017 international conference on smart technologies for smart nation (SmartTechCon), pp. 189-194, 2017, August.

[7] F. M. Aswad, A. N. Kareem, A. Khurdur, B. A. Khalaf and S. A. Mostafa, "Tree-based machine learning algorithms in the Internet of Things environment for multivariate flood status prediction," Journal of Intelligent Systems, pp. 1-14, 2021.
[8] F. R. G. Cruz, M. G. Binag, M. R. G. Ga and F. A. A. Uy, "Flood prediction using multi-layer artificial neural network in monitoring system with rain gauge, water level, soil moisture sensors," In TENCON 2018-2018 IEEE Region 10 Conference, pp. 2499-2503, 2018, October.
[9] C. Prakash, A. Barthwal and D. Acharya, "FLOODWALL: a real-time flash flood monitoring and forecasting system using IoT," *IEEE Sensors Journal,* pp. 787-799, 2022.