Information system for extreme rain index prediction and potential flood warning based on rain radar in Jakarta area

Muhammad Faisal Hidayatullah[1], Rama Pratama[1], Taufiq Salman Sya'bani[1], Syukri Darmawan[2], Rudy Komarudin[3], Umar Ali Ahmad[1], Asif Awaludin[1,3]

[1] Department of Computer Engineering, School of Electrical Engineering, Telkom University, Indonesia
[2]Research Center for Smart Mechatronics, National Research and Innovation Agency
[3]Research Center for Climate and Atmosphere, National Research and Innovation Agency

Article Info

Article history:

Received August 20, 2023
Revised September 8, 2023
Accepted September 8, 2023

Keywords:

Information System
Floods Potential Prediction
Machine Learning
Random Forest
PyStep

ABSTRACT

In Indonesia, significant variations in rainfall are a severe challenge, especially in the Special Capital Region (DKI) Jakarta, which is very vulnerable to flooding and the effects of climate change compared to other coastal areas in Southeast Asia. This flood caused considerable losses to society, industry, and government. Inaccurate predictions of rainfall lead to difficulties in anticipating floods and landslides, causing an expansion of the impact of these disasters. Therefore, a solution such as "Extreme Rain Index Prediction Information System and Rain Radar-Based Flood Potential Warning System for Jakarta" is needed. This system uses SANTANU rain radar and GSMaP satellite data to visualize real-time rainfall with Server-Sent Event (SSE) techniques. It also employs a Random Forest model to predict extreme rainfall and potential flooding by combining the output from PySteps-based radar nowcasting and Numerical Weather Prediction (NWP). Even though the predicted flood potential is accurate (93.75% accuracy, AUC 0.93), the prediction of flood movement using PyStep is not precise (RMSE 2.8, IoA 0.57), affecting the accuracy of the flood potential prediction. This system visualizes rain from SANTANU and GSMaP radars in real time and estimates the potential for flooding. Further efforts are needed to improve predictions of flood movements for better accuracy of flood potential information.

Corresponding Author:

Muhammad Faisal Hidayatullah
Department of Computer Engineering
School of Electrical Engineering, Telkom University
Bandung, Indonesia
Email: mfaisalhidayatullah@student.telkomuniversity.ac.id

1. INTRODUCTION

The territory of Indonesia is one of the regions with a tropical climate, so Indonesia has high rainfall variability. Several weather/climatic factors affect rainfall in Indonesia, both globally and locally, so that it can be daily, seasonal, or annual [1]. Several areas in Indonesia have very diverse rainfall, which makes it difficult to predict the extreme rainfall associated with floods, resulting in disasters such as floods and landslides in some areas. Therefore, the incident caused losses to society, government, industry, agriculture, and other sectors.

The Special Capital Region (DKI) Jakarta is the most flood-prone area and the most vulnerable to climate change impacts compared to other coastal regions in Southeast Asia [2]. The Meteorology Climatology and Geophysics Agency (BMKG) and the National Disaster Management Agency (BNPB) stated that one of the causes of flooding in Jakarta was rainfall [3]. Based on data reported by the BMKG, extreme rain in Jakarta and its surroundings occurred on December 31, 2019, and January 1, 2020. The highest rainfall reaches up to 377 millimeters (mm) per day. BNPB mentioned that as of January 6, 2022, as many as 67 people died, and 173,040 had to be relocated due to floods and landslides, which were almost evenly distributed in Jabodetabek [4].

Information on extreme rainfall associated with floods in Jakarta and its surroundings is needed to reduce the impact of flooding in the Jakarta area. In this case, accurate observation and forecast data with relatively fast access are key to an early warning system for potential flood disasters. Utilizing technological advances is critical in supporting the accuracy of the information received. The development of this component is essential and will be very beneficial for stakeholders and communities affected by floods in Jakarta and its surroundings.

The proposed solution to this problem is creating an information system for extreme rainfall associated with flood prediction for Greater Jakarta. The information system displays visualized rain radar and GSMaP satellite-based rain observation information. It also employs a Random Forest model to predict extreme rainfall and potential flooding by combining the output from PyStep-based radar nowcasting and Numerical Weather Prediction (NWP). This information system develops predictions of rain movements using an algorithm from PyStep so that they can provide results of rain movements in the next 1 hour. Determining whether the rain has met the extreme rain index and predicting the motion of rain is done by taking the maximum rain value from a given location point. Then the results from Pystep are used as input data in the Random Forest algorithm to predict flood potential. The hope is that the rain index prediction solution for Jakarta and its surrounding areas can be delivered in real-time and provide notifications to users if there is a potential for flooding in Jakarta and surrounding areas. In addition, this system has an interface easily understood by the public.

2. METHOD

2.1. System Architecture

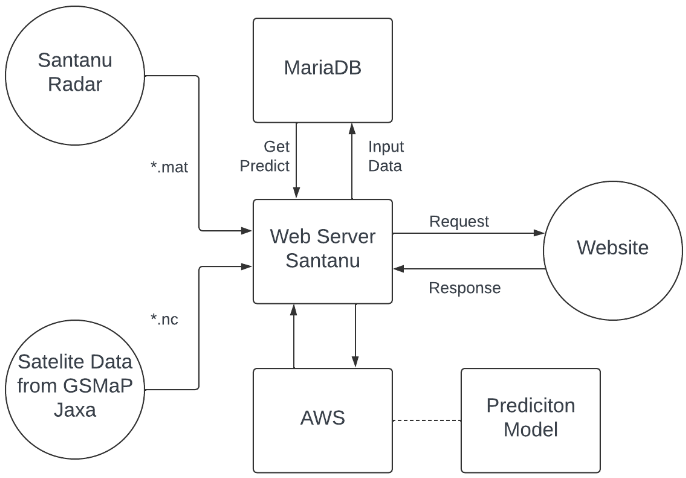

Figure 1. System Architecture Design

Figure 1 represents the central component in the system architecture that focuses on the web server, which plays a significant role in facilitating data communication. The web server receives rainfall data from radar data and GSMaP data. GSMaP (Global Satellite Measurement of Precipitation) is a project initiated by JST (Japan Science and Technology Agency), then the Japan Aerospace Exploration Agency (JAXA) promotes it to produce accumulated rainfall globally [5]. The GSMaP products have been provided to distribute hourly global precipitation map with 0.1 x 0.1 lat/lon grid [6]. Both data use different file formats, *.mat for radar data and *.nc for GSMaP data. Both types of data are stored within the storage component of the web server. However, the role of the web server is not limited to storage alone; the web server is also responsible for processing radar data, including important information such as rain locations, average rainfall, time signatures, and radar locations. All of these are then stored in a database for other purposes. Important record. After that, the stored data undergoes further processing before being sent to the client for visualization, enabling the representation of rainfall distribution based on radar and GSMaP data. In essence, the web server is the main foundation of this architecture, including data reception, storage, processing, database integration, and dissemination of processed data for client-driven rainfall visualization.

In addition, the web server also establishes communication with the Amazon Web Services (AWS) server that represents the server from the machine learning model. The AWS server is used to predict potential flooding based on available radar data. Data from the radar acts as input for machine learning models to predict the possibility of flooding in Jakarta and its surroundings. The prediction results are then forwarded to the web server, which sends it to the client via response data from the web server. Each element in this system is ultimately interconnected and interacts, forming an integrated unit to process and present important information related to rainfall and potential flooding.

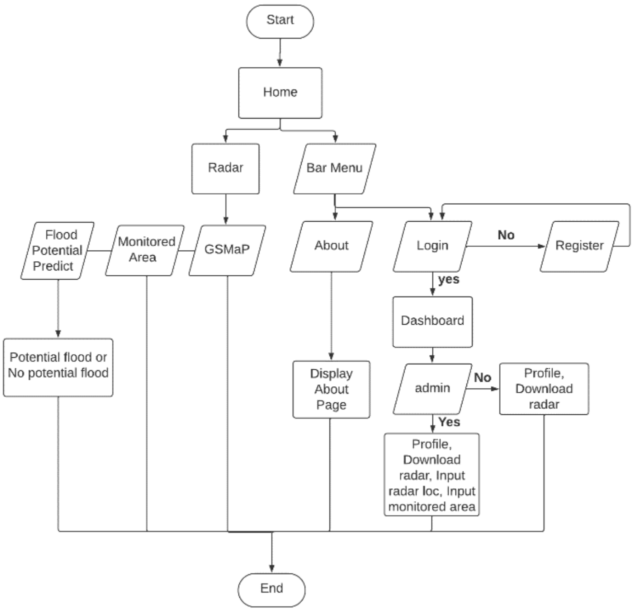

Figure 2. Flowchart of general website flow

Figure 2 represents the website's main page, and there is a map that visualizes the rain index for the Jakarta area based on radar data from SANTANU [7]. On the right side of the page, there are four buttons with different functions. The "Monitoring Area" button is used to display the area being monitored, while the "GSMaP" button directs the user to the GSMaP page, which shows a visualization of the rain index based on JAXA GSMaP satellite data, but covers areas all over the world, not just Jakarta. The "QPE" button is used to change the unit data display from DBZ to QPE. The last button is "Notifications", which functions to check for potential flooding in that area.

At the top left of the page, there is a menu bar that allows navigation to dashboard features. This feature can only be accessed after going through the authentication system. Users are required to perform the login process first before being able to access the dashboard. If you don't have an account, the user can register to create a new one. With this view, users can access various features provided by the website, from visualization of the rain index to the potential flood notification feature.

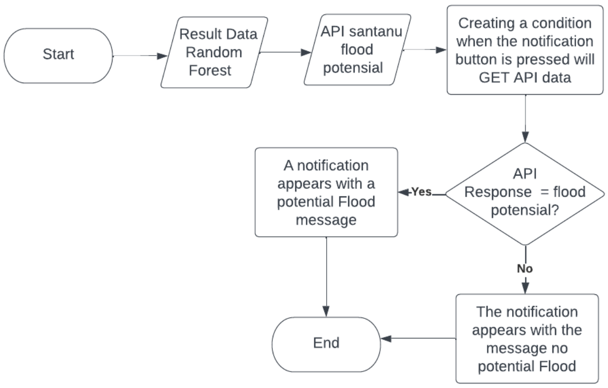

Figure 3. Flowchart Notification Flood Prediction

Figure 3 represents the output of the Random Forest algorithm will be implemented as an API which will be displayed on the front-end. On the page, there is a notification button that sends GET requests to the API to get information about potential flooding. When the notification button is pressed, the system will perform an inspection based on the results obtained from the API. If the API results indicate a potential flood, a pop-up containing a message informing you of a potential flood will appear. However, if the results from the API show no potential for flooding, another pop-up will appear, giving a message that there is no potential for flooding to worry about. With this mechanism, users will quickly get relevant information about potential flooding based on the results of the Random Forest algorithm through easy interaction with the notification button on the front-end.

2.2. System Server Architecture

This research develops two servers that have particular roles and are connected. First, a web server functions as a user request manager for presenting web data via the internet. This web server is tasked with displaying a visualization of rain distribution based on information from radar and GSMaP. On the other hand, a machine learning model server is used to calculate predictions of potential flooding in Jakarta and its surroundings. The two servers communicate with each other to support an information system for predicting potential floods in Jakarta and its surroundings.

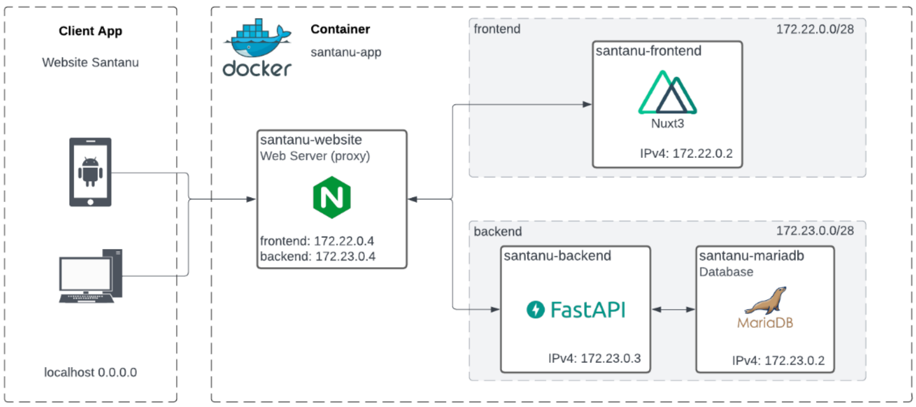

Figure 4. Block diagram of the web server

Figure 4 is a system schematic that is operated on a web server. Each system component will be isolated in its container, aiming to facilitate development and provide a stable environment. A container is a running instance of an image with multiple containers running from the same image [8]. Four containers will run on different IP addresses and interact with each other. The back of the system will have two containers, namely santanu-backend as the system core and santanu-mariadb as the database. Then, in the front-end system, there is one container, namely the santanu-frontend. The backend system runs on the IP address 172.23.0.0/28, while the front-end operates on the IP address 172.22.0.0/28. The connection between the two systems is via the santanu-website container. The Santanu-website container acts as a web management system that runs both the front-end and back-end. Provides a layer of security by using a proxy; when the user (client) sends a request via a predefined URL, then the proxy will redirect to the appropriate system IP address (frontend or backend).

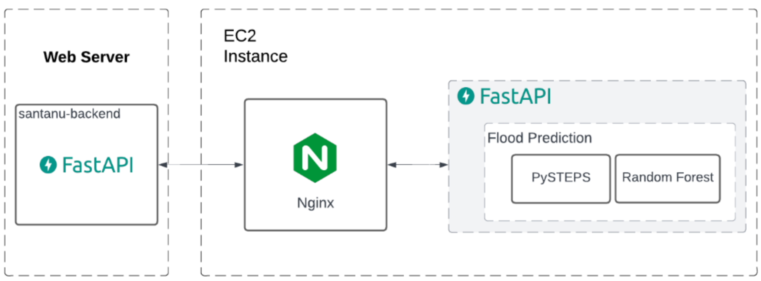

Figure 5. Block diagram of the machine learning server

Figure 5 depicts a schematic system implemented in a machine-learning server. The machine learning server interacts with the backend server on the web server, aiming to support predictions of potential flooding in the system. On the machine learning server, EC2 instances are used. EC2 is a service from AWS that allows setting up and configuring a computing environment according to system requirements. The advantages of EC2 include ease of management and optimal configuration according to specifications. In addition, using EC2 instances also provides cost benefits that are more economical compared to similar service alternatives. Therefore, EC2 was chosen to improve performance and optimize performance in machine learning tasks at an affordable cost.

2.3. Communication System

This website system uses two communication protocol methods: Polling and SSE. The Polling method, also known as client-pull, requires the client to periodically send requests to the server to get data at specific intervals. Meanwhile, the SSE method, or server push, allows the server to proactively send client information or data updates without requiring special requests [9]. In implementing this website system, the polling method is used for components that do not require real-time data exchange and use the JSON format via the API system. Meanwhile, the server-sent event method is applied to situations that require real-time data communication through an event source system.

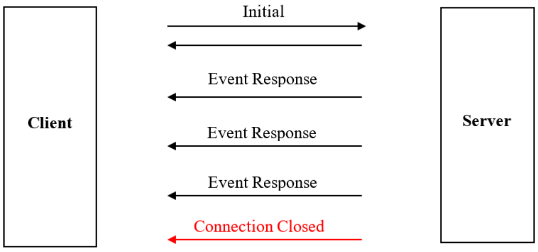

Figure 6. Server-Sent Event communication flow

SSE is a technology that provides a server to update a website page (frontend) automatically if there is a change in data on the server side. So that the client does not need to request data updates from the server repeatedly. SSE will be applied to visualize the movement of rain clouds so that they can display real-time visualizations. Thus, users can get up-to-date information following existing data.

2.4. Blok Diagram Prediction

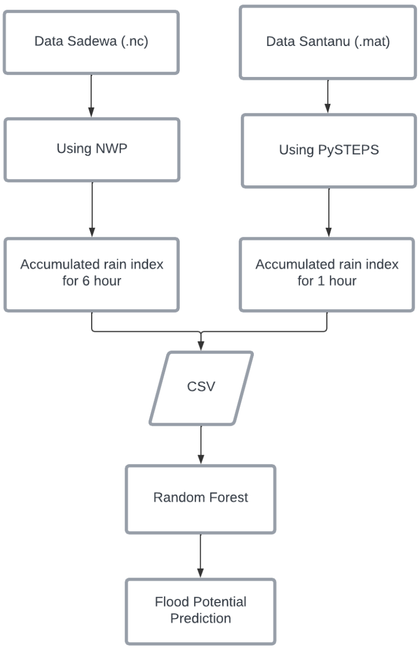

Figure 7. Machine learning workflow block diagram

The diagram illustrates the workflow of the machine learning method applied in this study. The process begins by collecting SANTANU data in (.mat) and SADEWA data in (.nc) format. The SANTANU data is then processed using pySTEPS to produce an output of an accumulated rain index within 1 hour. On the other hand, SADEWA data is processed through Numerical Weather Prediction (NWP) to have a rain index with an accumulation of 6 hours [10]. After that, both rain indexes are combined and saved in CSV format. The next step is utilizing the Random Forest algorithm to classify data and predict the potential for flooding. Thus, this flow describes how data is processed and used effectively to apply machine learning to classify and predict flood potential.

2.5. PySTEPS

The dataset used in this study was obtained from the SANTANU rain radar [11, 12]. which is managed by the National Research and Innovation Agency (BRIN). This dataset is presented in data format (*.mat) and consists of radar data spanning 2020 from June to December and 2022 from June to August. In total, there were 756 data analyzed. This radar dataset contains information about rain intensity. By leveraging pySTEPS, this dataset is used to make predictions, allowing researchers to understand patterns of rain movement over that period.

PySTEPS is a framework created in Python specifically for processing weather radar data and generating forecasts of rain movements using quantitative forecasting methods. This platform consists of modules that can be arranged modularly, provides ease of use, is open source, and is overseen by the collaboration of the community of contributors [13]. Its function is for nowcasting, which refers to predicting the weather in very short time intervals. Within its framework, pySTEPS provides a series of efficient algorithms and techniques for nowcasting analysis, including the application of ensemble weather models and error correction techniques [14] .

In making predictions on pySTEPS, there is a method for estimating rain movement, namely by using Lucas Kanade. The Lucas-Kanade method is used to estimate and track the activity of clouds and rain using radar imagery. In the Lucas-Kanade method, the rain intensity patterns on the two radar images are compared. Then the shift vector is calculated to represent changes in the position of clouds and rain from one image to another. This method makes it possible to understand and predict the movement of rainfall more accurately and efficiently based on the captured radar data.

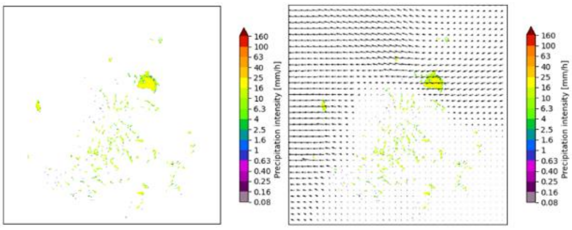

Figure 8. Visualization of rainfall data and estimation of its movement

Furthermore, an Extrapolation process is carried out using rainfall data that has been observed at the previous time, which aims to project the direction and speed of movement of clouds and rain. It is used to provide short-term weather predictions or nowcasting. Through this method, estimates of cloud and rain movements are obtained, which allow for more accurate predictions in a concise period. Extrapolation has a crucial role in the PySTEPS framework as a component that optimizes nowcasting analysis to improve the quality and accuracy of weather predictions on a very short time scale.

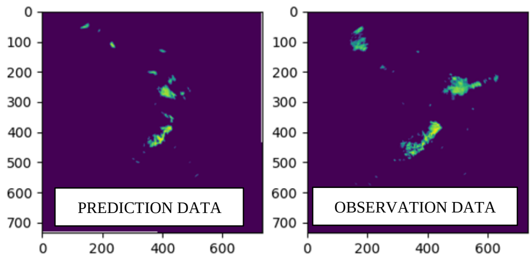

Figure 9. Visualization of future precipitation forecasts

At the evaluation stage, an analysis of the predicted results of the data and rainfall observation data is carried out. This evaluation utilizes two metrics: the Root Mean Square Error (RMSE) and the Index of Agreement (IoA).

The output of pySTEPS will take several variable values, which will be used as input data in a random forest. The data includes heavy rain variables, duration, and maximum rain index values. Rain is very heavy if the rain intensity value is greater than 20 mm/hour or more than 100 mm/day [15]. Then this data will be combined with flood data from BPNPB (National Disaster Management Agency) data to be used as a target parameter in the random forest algorithm to predict flood potential.

2.6. Random Forest

This research uses two rainfall prediction datasets. The first is pySTEPS-based nowcasting data derived from SANTANU radar, and the second is SADEWA Numerical Weather Prediction data. Both data are important predictors of the precipitation predictions by the radar nowcasts followed by the precipitation predictions by the numerical weather predictions [16]. The datasets include information on the accumulated extreme rainfall of 756 data. This data is then processed through the Random Forest algorithm to produce a combined prediction output that indicates the potential of flooding.

Random Forest is a machine learning algorithm that builds multiple decision trees and merges them together to get a more accurate and stable prediction [17]. Each tree contributes to the prediction, and the final prediction result is determined based on a majority vote from all trees (majority voting) [18]. This algorithm represents a form of ensemble learning combining several simple models to produce a final prediction.

The process begins by pre-processing the data by reading the dataset from the CSV file. This dataset contains information related to the potential for flooding in an area. Then, an oversampling technique is applied to create synthetic samples for the minority class to balance the majority and minority classes. The next step is to divide the data into two parts: training data and testing data. The training data is used to train the model, while the test data is used to test the model's performance that has undergone the training process. The next step involves building a random forest model by applying the following parameters.

Table 1. Default parameter Random Forest

Default Parameter Random Forest				
Max_depth	Min_leaf	Min_Split	N_estimators	Random State
None	1	2	100	None

The following process involves model evaluation using metrics such as classification reports, AUC ROC (Area Under the ROC Curve), and Confusion Matrix. Using these metrics is beneficial in evaluating the performance of potential flood prediction models based on extreme rainfall data.

3. RESULTS AND DISCUSSION

3.1 Realtime Visualization

On the web page which displays visualization in real-time, the website displays the results of data processing which are converted into images so that they can be visualized. The data used is from the SANTANU radar and the GSMaP satellite.

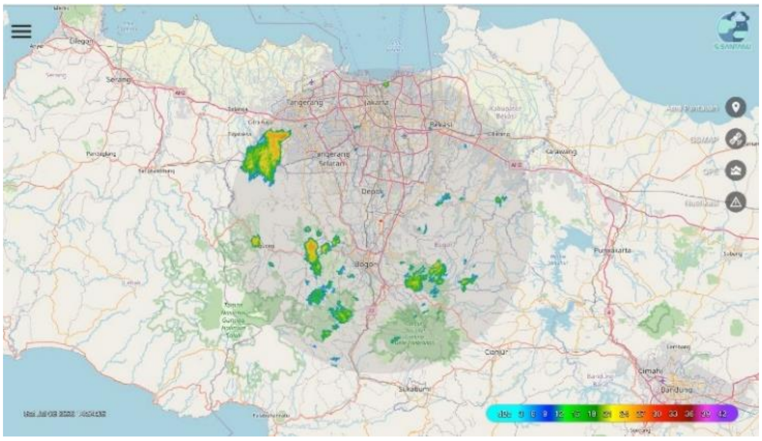

Figure 10. Visualization Data Radar

Figure 10 represents the website can visualize as a rain index based on data from the SANTANU radar. The data is processed into images and then stored on the server SANTANU, then the front-end communicates using SSE so that the visualization is carried out in real-time.

Figure 11. Visualization Data GSMaP

Figure 11 represents the visualization system based on GSMaP satellite data is the same as visualizing based on radar data, but the difference is that the GSMaP visualization is communicated to the JAXA server first to obtain the data. then the next process will be the same as radar data visualization

Table 2. Visualization Test

No	Scenario	Output	Description
1	Open the website on the main page	Displays a map with visualizations such as a rain index based on real-time SANTANU radar data	Success
2	Press the QPE button on the main page	Change DBZ units to APE units and change the legend based on those units	Success
3	Press the GSMaP button on the main page	Moves to the gsmap page and displays a map with visualizations such as a rain index based on gsmap satellite data in real-time and adjusts the legend to the index color	Success

3.2 Prediction System Test

The developed information system can provide predictions of potential flooding based on rain data obtained from radar. The flood potential prediction feature can be accessed via a website's main page button. When the button clicks, the website server communicates with the machine learning server to run a flood prediction process in the next 1 hour. When the results from the machine learning server provide a prediction that an area has the potential for flooding, the website will display a flood warning notification popup to the user.

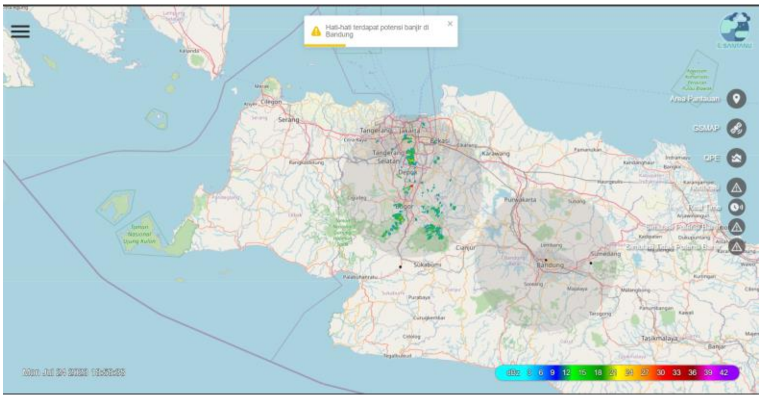

Figure 12. Potential flood warning feature

3.3 Score Prediction of Flood Potential

After the dataset is read, the data is divided into training and observation data, each with 30 data. The extreme rainfall associated with floods prediction model was built using the Lucas Kanade method to estimate rain movements and followed by an extrapolation process to predict future rainfall using rainfall observation data in the previous period. These results are evaluated by visualizing them in the form of plots to find out how far the predicted data results approach the observation data, which can be seen in Figure 13.

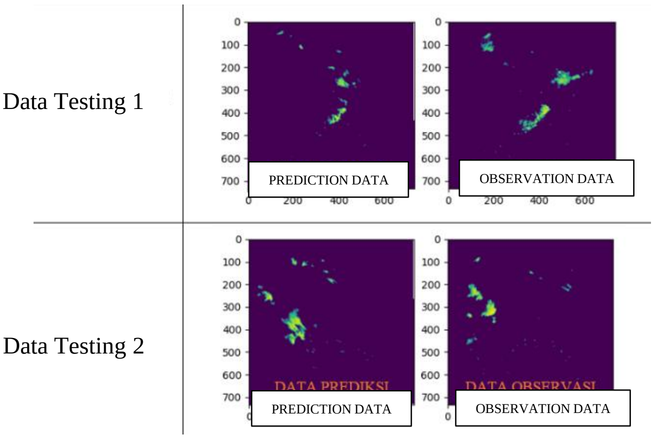

Figure 13. Results of data processing observations & predictions Lucas-Kanade method

In this test, the results of the predictive data have little difference from the observation data. Therefore, the prediction and observation data processing is not good enough. Next, an evaluation is carried out using two performance metrics: the Root Mean Squared Error (RMSE) and the Index of Agreement (IoA). The results of the two metrics are presented in Table 3.

Table 3 The results of the evaluation value index RMSE and IoA Lucas-Kanade method

Data Testing 1				Data Testing 2			
No	Time	Nilai RMSE	Nilai IoA	No	Time	RMSE Value	IoA Value
1	2	0.9472	0.9195	1	2	1.0578	0.8913
2	4	1.3443	0.8741	2	4	1.5217	0.8206
3	6	1.6165	0.8401	3	6	1.7777	0.7776
…	...	…	…	…	…	…	…
29	58	2.6235	0.6757	29	58	2.8677	0.5491
30	60	2.6130	0.6796	30	60	2.8225	0.5546
Average		2.461	0.705	Average		2.657	0.598

This evaluation uses the RMSE value to measure the average difference between the observed data and the predicted rainfall. The lower the RMSE value, the more accurate the prediction model. On the other hand, IoA is used to measure the degree of agreement between model predictions and observational data. The range of IoA values is 0 to 1, with 1 indicating a perfect fit between forecast and observation. The evaluation results illustrate that using the Lucas-Kanade method in predictions does not reach an optimal level of accuracy.

After the pre-processing process, the data is considered clean and ready to be used. The dataset related to flood potential shows an imbalance in the target class. The solution to this problem is through the resampling method by applying the oversampling technique. Furthermore, the data is divided by the proportion of 30% for the test data and followed by model building using Random Forest with default parameters. The results of the performance evaluation of the performance of this model can be seen in Table 4.

Table 4. Performance evaluation using default parameters.

Accuracy	AUC
0.93014	0.92975

With an Area Under the ROC Curve (ROC AUC) value of 0.9297, the model has good classification quality in separating positive and negative classes. The graph can be seen in Figure 14.

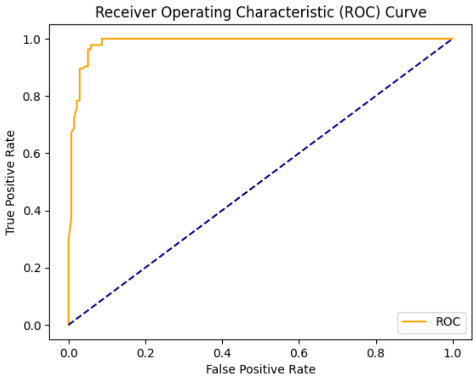

Figure 14. ROC results use the default parameters

The evaluation was also carried out using the Confusion Matrix, as shown in Figure 15. The results of the Confusion Matrix show that 132 data are correctly classified as True Positive (TP), 6 data are incorrectly classified as False Positive (FP), 13 data are incorrectly classified as False Negative (FN), and 121 data are correct. -correctly classified as True Negative (TN). This indicates that the Random Forest model has few errors in its predictions but still shows an excellent ability to classify data with high accuracy and minimal errors in positive and negative predictions.

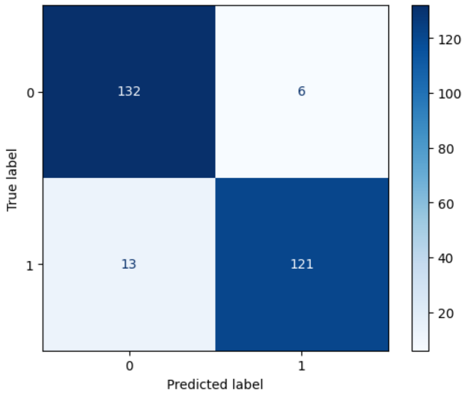

Figure 15. Confusion Matrix results using default parameters

3.4 Web Application Testing

Alpha testing is the first step in software testing, where developers test the software that has been built before releasing it to end users. Within the alpha testing period, testing is conducted in a controlled environment to check crucial features and ensure the system complies with predetermined requirements. In this research, testing was carried out by testing the performance of the website system. Verification is carried out through at least two tools to validate the test results in testing website performance. The primary tool used is JMeter, which measures system response times. JMeter is a powerful, user-friendly, and open-source performance testing tool [19]. Other tools used are Lighthouse to test system loading time and Postman to test system response time.

Table 5. (a) response time using Postman, (b) response time using JMeter

Label API	Average (a)	Average (b)
Radar location	26 ms	16 ms
Monitored area	29 ms	15 ms
Download path	188 ms	15 ms
DBZ	.96 ms	17 ms
QPE	89 ms	15 ms
GSMaP stream	23 ms	15 ms
Radar stream	20 ms	16 ms

Based on Table 5, this information system has a reasonably fast response time. This can be seen from each average value resulting from the test. Response time is fast if the execution time is less than 1.5 seconds. Based on this, this information system can provide users with real-time visualization of rain distribution.

4. CONCLUSION

Based on the research conducted in developing this information system, the system has been successfully developed according to the requirements. This system can provide data in real-time using SSE (Server-Sent Events) communication and implementing polling communication through the developed API. The system's ability to communicate between web servers and machine learning servers is the main feature that supports the efficiency and accuracy of sending information about potential flooding to users. However, in predicting extreme rainfall with pySTEPS, the results still require improvement for model optimization. This information system also provided predictions of potential flooding based on the extreme rain index using the Lucas Kanade method, which is superior in RMSE and IoA evaluation compared to other methods. Extreme rainfall visualization features and potential flood warnings in Bandung have also been implemented. However, data limitations in Jakarta led to using Bandung data in making prediction systems. The research shows the potential for further development regarding weather monitoring and flood prediction by combining two predictions from radar nowcasting and NWP.

REFERENCES

[1] B.Mufarida, "Begini Penjelasan BMKG Kenapa Curah Hujan Tinggi Sepekan Terakhir," 15 10 2022. [Online]. Available: https://nasional.okezone.com/read/2022/10/15/337/2687539/begini-penjelasan-bmkg-kenapa-curah-hujan-tinggi-sepekan-terakhir. [Accessed 28 10 2022].

[2] Y. Kawata, Green Infrastructures in Megacity Jakarta: Current Status and Possibilities of Mangroves for Flood Damage Mitigation, Osaka, Japan: Ecological Research Monographs, 2022.

[3] H. I. N. P. Budi Prasetyo, "KARAKTERISTIK CURAH HUJAN BERDASARKAN RAGAM TOPOGRAFI DI SUMATERA UTARA," *Jurnal Sains & Teknologi Modifikasi Cuaca,* vol. 19, pp. 11-20, 2018.

[4] A. M. Ginting, "DAMPAK EKONOMI DAN KEBIJAKAN MITIGASI RISIKO BANJIR DI DKI JAKARTA DAN SEKITARNYA TAHUN 2020," *Singkat Bidang Ekonomi dan Kebijakan Publik DPR-RI,* vol. XII, no. 1, 2020.

[5] S, Sharma et al;, "Evaluation of GPM-Era Satellite Precipitation Products on the Southern Slopes of the Central Himalayas Against Rain Gauge Data," *Remote Sensing,* vol. 12, no. 11, p. 1836, 2020.

[6] T. Kubota, K. Aonashi, T. Ushio, S. Shige, Y. N. Takayabu and M. , "Global Satellite Mapping of Precipitation (GSMaP) Products in the GPM Era," *Advances in Global Change Research,* vol. 67, 2020.

[7] T. Sinatra, G. A. Nugroho and A. Awaludin, "Optimasi Data Radar Hujan dan Perluasan Jangkauannya Menggunakan Jaringan Radar," *Pusat Sains dan Teknologi Atmosfer - LAPAN,* vol. 18, no. 1, pp. 43 - 54, 2020.

[8] I. Miell and A. H. Sayers, Docker In Practice Second Edition, Shelter Island, New York: Manning Publications Co., 2019.

[9] G. Ayrancioglu, "What is Server-Sent Events (SSE) and how to implement it?," Medium, 7 Februari 2022. [Online]. Available: https://medium.com/yemeksepeti-teknoloji/what-is-server-sent-events-sse-and-how-to-implement-it-904938bffd73. [Accessed 3 Maret 2023].

[10] F. Lasmono, R. and I. Fathrio, "Evaluasi Skema Parameterisasi Model WRF untuk Pengamatan Kemunculan Awan dari Satelit Himawari-8 di atas Indonesia," *Jurnal Sains Dirgantara,* vol. 18, no. 2, pp. 89 - 98, 2021.

[11] G. A. Nugroho, A. Awaludin and S. A. Rahayu, "Pemanfaatan Noise Radar Kapal untuk Pemantauan," *JURNAL ILMIAH ELITE ELEKTRO,* vol. 3, no. 1, pp. 43-48, 2012.

[12] A. Awaludin, G. A. Nugroho, S. Alam, D. F. K and R. Yuwono, "Development of marine radar signal acquisition and processing system," *2013 International Conference on Information Technology and Electrical Engineering (ICITEE),* pp. 97-100, 2013.

[13] PySTEPS, "pysteps – Inisiatif nowcasting," [Online]. Available: https://pysteps.readthedocs.io/en/stable/. [Accessed 6 June 2023].

[14] S. Pulkkinen, D. Nerini, . A. A. Pérez Hortal, C. Velasco-Forero, A. Seed, U. Germann and L. Foresti, "Pysteps: an open-source Python library," *journal Geoscientific Model Development,* 27 October 2019.

[15] D. Yudistira and R. C. Hamonangan, "Peluang Bencana Banjir Pada Saat Hujan Lebat dan Sangat Lebat di Kawasan Pantura Provinsi Jawa Barat," *Buletin GAW Bariri (BGB),* vol. 2, no. 1, pp. 16 - 23, 2021.

[16] Y. MAO and A. SORTEBERG, "Improving Radar-Based Precipitation Nowcasts with Machine Learning Using an Approach Based on Random Forest," *American Meteorological Society,* vol. 35, pp. 2461-2478, 2020.

[17] P. Firoozishahmirzadi, S. Rahimi and Z. Esmaeili Seraji, "Application of machine learning models for flood risk assessment and producing map to identify flood prone areas: literature review," *International Journal of Data Envelopment Analysis,* vol. 9, no. 4, pp. 43-90, 2021.

[18] K. Nam and F. Wang, "An extreme rainfall-induced landslide susceptibility assessment using autoencoder combined with random forest in Shimane Prefecture, Japan," *Geoenviron Disasters,* vol. 7, no. 6, pp. 1-16, 2020.

[19] H. Lumbantobing, "Tutorial Performance Testing Menggunakan JMeter — Beginner," 28 Juli 2021. [Online]. Available: https://hariatylumbantobing.medium.com/tutorial-performance-testing-menggunakan-jmeter-beginner-54d3c7ef4ee.