

Web-Based Flood Hazard Monitoring

Anna Nur Nazilah Chamim¹, Dwi Cahyo Hardyanto², Karisma Trinanda Putra³

^{1,2,3} Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Yogyakarta, Indonesia

³ Asia University, Taiwan, Province of China

Email: anna_nnc@umy.ac.id¹, dwi.cahyo.hardyanto@umy.ac.id², 107221038@live.asia.edu.tw³

Abstract— Flood is a natural disaster. It occurs in several cities in Indonesia. Floods caused by rivers that overflowed and then flooded residential areas. It comes mostly unexpectedly without early warning. It causes many losses, especially the loss of material, and health threats to surrounding communities. The advance of network technology can reduce the adverse effects of flooding by providing warning alarms and water level monitoring system in real-time that can be accessed via the web. Based on the problem, a monitoring system was designed to monitor water levels via a web that work in real time for 24 hours, and store water level data into the database. The use of this website requires an internet connection, so that internet services must be available.

Keywords— Flood, alarm warning, Realtime.

I. INTRODUCTION

Flood is a natural phenomenon, and a real threat to several cities in Indonesia. It includes cold lava floods that threat the regions along the river through which the volcanic eruption of Merapi erupted. The floods can be evaded in several ways, including providing a system of forest parenting and maintenance, and providing a water level monitor device placed upstream of the river, to give a warning to the surrounding residents near the river. The river water level monitoring system has been carried out manually, but it was not efficient because there must be people who come directly to the river to check it.

Flood detection has been investigated by several previous researchers. Detection of flood disasters was examined by Liu using satellite imagery. The homogeneous pixel transformation is used to change the heterogeneous remote sensing image. The K-nearest neighbor's algorithm is used to detect flooding [1]. Flood detection by using a synthetic aperture radar image (SAR) was examined by Monti-Guarnieri. Detection of coherent changes for multipurpose SAR can be used to detect flooding [2]. The synthetic radar aperture (SAR) image examined by Sumaiya is used to detect flooding. Detection of unattended changes to flood-affected areas in SAR images using the Rayleigh Bayesian-based threshold [3]. The synthetic aperture radar (SAR) image examined by Amitrano was used to map the flood area. Sentinel-1 GRD SAR Images are used for rapid flood mapping research [4].

Flood detection research using PolSAR images was examined by Sui. Detection of flooding in the PolSAR image based on the level set method considers prior geoinformation

[5]. Flood detection research with satellite imagery was carried out by Pradhan using RADARSAT-2 imagery. Optimized rule-based flood mapping techniques using multitemporal RADARSAT-2 images in the tropics [6]. The SAR image is examined by Chini using a split-based hierarchical approach to the parametric threshold of the image [7]. Satellite images such as Landsat-8, ALOS-2, and RADARSAT-2 are used by Pierdicca to map stagnant vegetation using COSMO-SkyMed with polarimetric and optical data on comparable fields [8]. Detection of flooding caused by lake ice in the Tibetan plateau area was investigated by Chen using Landsat 8 satellite images and google earth. 8 Landsat Imagery and the Google Earth Engine are used for extracting Ice lake lines in the Tibetan plateau [9].

The initial flood detection and flood control system based on real-time WSN was investigated by Thekkil. The system consists of many sensors that are connected using the ZigBee type WSN. The system is connected with a GSM module to send data using GPRS [10]. Rain detection system to estimate weather levels using the Mamdani fuzzy inference system. This system consists of rainfall detection sensors and microcontrollers. Fuzzy Mamdani algorithm is used to predict rain and flood [11]. Design of a flood detection system based on speed and water level of DAM with ESP8266 [12].

This paper presents the detection of flood hazards that are different from previous studies that detected flood hazards using SAR images, Landsat-8, ALOS-2, and RADARSAT-2 satellite images. This paper aims to present flood detection by using a high-level water detection sensor in the river and displaying the results on the Web. Information on flooding of an area displayed on the web aims to prevent vehicles outside the area from going into flooded areas.

II. RESEARCH METHOD

The functional requirements of the system are the presentation of the features that will be included in the application to be built. These features must meet several criteria: able to display information on water level by indicator lights and provide a warning, able to display information in the graphs and sensor data in realtime, able to display stored data based on a certain range and displayed in graphics. In general, this system is composed of several parts as shown in figure 1. It displays the overall system diagram block. The functions of each parts of the diagram block are



as follows: the sensor informs the water level, the Arduino serves [13]–[18] as an indicator light controller and alarm, and processes data commands to be sent via HTTP client to the web server, the database serves as data storage from sensor readings. Web Server serves as the recipient of the data to be saved to the database and sends the data in the form of JSON (JavaScript Object Notation) to be processed. Finally, the data are displayed in graphical form completed with sensor reading data to be displayed on the web client.

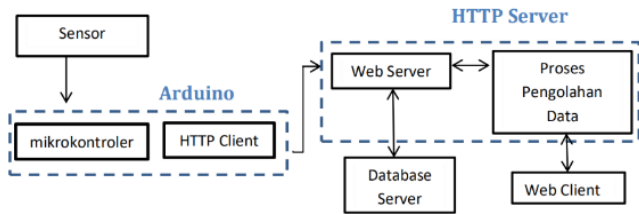


Fig. 1. Block diagram of web-based flood hazard monitoring

A. Arduino Design as HTTP Client

The Arduino Ethernet Library provides various functions/methods and several protocols (HTTP, TCP, UDP) which are likely to make Arduino a client or server [19]–[22]. Client or server is an application/software, while the physical form can vary such as PCs, mainframes, laptops, microcontrollers, and others. The server is passive/service waiting for a request from the client. The microcontroller type Arduino design used an HTTP client to process a sensor reading data and sent to a web server via LAN [23], [24]. Subsequently, the web server receives the sensor readings and saves it to the database.

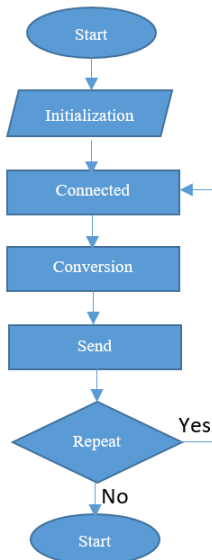


Fig. 2. Flowchart of sensor reading and data sending to a web server.

The process of reading sensors and sending data to the web server illustrated in figure 2 started from the initialization process and some configurations located in the setup function. The setup function is a function that starts when the program is run at the first time. In other words, the setup function runs when the circuit is connected to a voltage source. Connection is an attempt to connect Arduino to a web

server, while Conversion is the process of changing signals from sensors converted into units of centimeters (cm). The sensor produces ultrasonic waves and determines the water level by counting the number of high pulses received by the sensor from the wave reflection. Send is the process of sending information of water level to a web server using the http protocol. The information sent is water level in centimeters (cm). This information will be stored in the database as a recording. Finish is checking the condition of whether the program has finished or not. When the program is completed or stopped, all program is completed. Conversely, when the program is still running, it returns to the connection to repeat the sending of information.

B. Designing a Web Server

Web server is a computer device to access all types of files contained on web pages via HTTP/HTTPS by using a particular application or program known as a web browser. When the request reaches the correct web server (hardware), the HTTP server accepts request, finds the requested file and sends it back to the browser, also through HTTP. The main function of a web server is to transfer files or user requests through the specified communication protocol. To run a web server using software called XAMPP as shown in figure 3 [25]–[29].



Fig. 3. XAMPP Control Panel.

Figure 4 illustrates the process of receiving sensor data to the database. Connection to database is an attempt to connect Arduino to the database via a web server with the GET method. If the connection is successful, the web server will receive the data from Arduino to be stored in the sensor table.

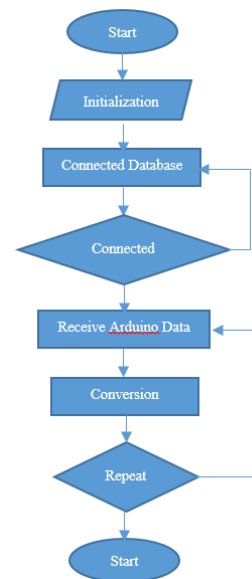


Fig. 4. Flowchart of sensor data reception.

C. Database Design

A database is a collection of data or information stored systematically on a computer. It can be processed or manipulated using software (application programs) to produce information. The definition of the database includes specifications of data types, structures, and also the limitations of the data to be stored. A database is an important aspect in an information system since it is a storage container of the data that will be processed further.

The design used MySQL, a software DBSM (database management system) that allows users to maintain, control and access data in a practical and efficient manner. DBMS handles all access to the database. Several functions that must be handled by the DBMS are handling the definition process by the DBA (Database Administrator), handling the failures in accessing data caused by system or disk damage and handling the performance of all functions efficiently.

TABLE I. DATABASE STRUCTURE

Name	Type	Null	Key	Default	Extra
Id	Int(5)	No	PRI	None	Auto_increment
Waktu	timestamp	No		CURRENT_TIMESTAMP	
Sensor	Int(5)	No		None	

The Id attribute used type Int (5) as a unique marker that will be added to the MySQL table up to a maximum of 5 digits (99999) automatically (Auto_Increment). The Time Attribute used Timestamp to give the date and time with Default Current_Timestamp so that if there are data entered, the default time will be immediately added. The sensor attribute used type Int (5) as the recipient of data sent by Arduino via a web server.

III. RESULTS AND ANALYSIS

The tests performed in the research described the process to connect Arduino to a web server and store the data of the sensor readings that have been processed by the Arduino and saved to the database server.

A. Main Controller Series Test

This test was performed to find out whether the main controller circuit in this case Arduino Uno with Ethernet Shield could function properly or not. The test included an Arduino connection test with a computer serial port and with the added Ethernet Shield on the Arduino board. The goal is to add Arduino capabilities to connect to computer networks.

To test the Arduino connection, the USB cable is connected from the computer to the Arduino series, then the Arduino IDE software is run. If the Arduino has been connected to a computer, the Arduino IDE give a serial port connection on the computer that Arduino uses. For the test with the Ethernet Shield, the module is connected by placing the ethernet shield circuit board on the Arduino board. UTP cable with an RJ-45 connector crossed by a crossover

connected localhost computers to the Ethernet Shield. Program examples from the Ethernet library, the IP address of the computer with Ethernet Shield adjusted to one network were uploaded on the Arduino IDE. If the Arduino works properly, the monitor on the Arduino IDE will display connected, and at the serial width of the monitor shows the port used by Arduino as shown in Figure 5.

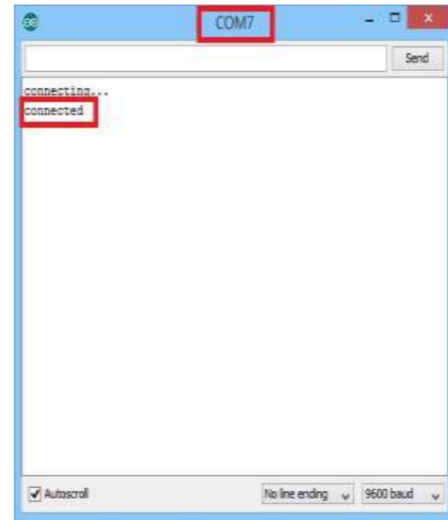


Fig. 5. Arduino monitor series

B. Sensor Performance Test

1) Monitor Series

In the monitor series test, the USB cable connected the computer to the Arduino series. When the Arduino was connected, the program to be executed by the Arduino should have been prepared. In the test on the appearance of sensor, the library contained in the IDE Arduino software was used. The sensor readings used Example Library New Ping program.

The program was carried out to find out the sensor readings displayed on the monitor series. If what is displayed is in accordance with what is read (in this case the sensor reads the water level) then the sensor works properly.

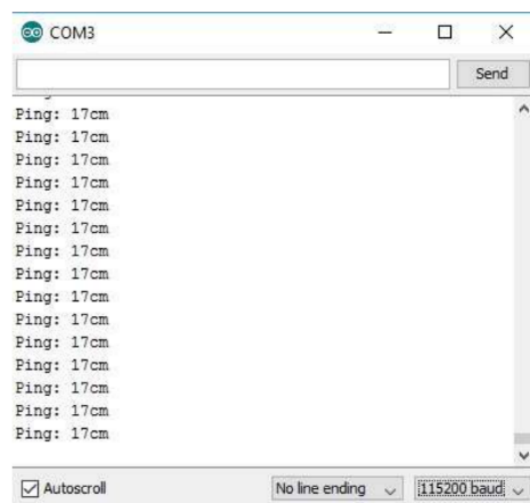


Fig. 6. Results of sensor readings

Figure 6 displays the result of sensor readings. The sensor reading is said satisfactory if the data read by the sensor are the same as displayed on the monitor series.

2) Process of Changing Indicator Lights and Alarm Alerts

In the process of changing indicator lights and alarm alerts, first, a USB connects the computer to the Arduino series and Ethernet shield. When the Arduino has been connected, the indicator light will automatically turn on according to the command on the program and the sensor data will automatically enter the database.

3) Test on Blue Indicator Light

The blue indicator light is on when the water level is less than 5 cm. The blue indicator light is to inform that the water level is within the safe limit of less than 5cm.

4) Test on Green Indicator Light

The green indicator light is on when the water level is more than 5 cm and less than 10 cm. The green indicator light is to inform that the high level of water in the Alert limit 1.

5) Test on Yellow Indicator Light

The yellow indicator light is on when the water level is more than 10 cm and less than 17 cm. The yellow indicator light is to inform that the high level of water in the Alert limit 2.

6) Test on Red Indicator Light

The red indicator light is on when the water level is more than 17 cm. The red indicator light is to inform that the high level of water in the Alert limit 3.

7) Process for Displaying Graph Data and Water Level Warning

In testing this time connect Arduino, the USB cable is connected from the computer to the Arduino serial and Ethernet shield if Arduino is connected then the indicator light will automatically turn on according to the command on the program and the sensor data will automatically enter the database.

C. Testing of Realtime Data

In testing this realtime data is to display high level water data and graphics in realtime, where the data is taken from the database. To test realtime data requires a program to retrieve data from the following databases are programs for retrieving data from the database. The program appears. Js to retrieve data converted to JSON data in the file (appear.php) and each function from the data results is added and sorted through the id, time and sensor as in Figure 7.



Fig. 7. Javascript Dsisplay

D. Test on Graph

The graph test used a file (datas.php) to retrieve data from the database and called by highchart located in the file (datasensor.php) to be made into graphical form as shown in figure 8.

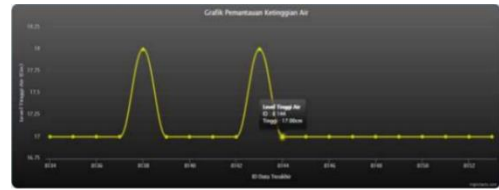


Fig. 8. Realtime Graph

Figure 8 displays a realtime graph. The graph finds out the water level read by the sensor. In the reading process, that is by taking the latest data recorded by the database, the Y-axis is the high value of water in units (cm) and the x-axis is the sampling data based on the id in the database.

E. Test on Alert Programs

Alerts gives messages or feedback to users. In the test on alert warning program, test the Arduino connection and the USB cable connected to the Arduino port then connect the Ethernet cable from the computer to the Ethernet shield that has been connected to Arduino and the microcontroller.

F. Test on Alert Information Program

The test on alert information program was performed to find out the water level in a graph and give a warning, and when the high level of water is less than 5 cm, a notification will appear in the form of a yellow alert on the web.

G. Testing the Alert Success Program

The alert success program test was performed to find out the water level in a graph and give a warning, and when the high level of water is above 5 cm and less than 10, a notification will appear in the form of a green alert on the web. Figure 4.16 shows when the program works well.



Fig. 9. Warning Alert 1

Figure 9 shows the water level is between 5 cm and 10. The notification displayed is "Attention Water Level Alert 1!" and turn on the green indicator light alert.

H. Test on Alert Warning Program

The alert warning program test was performed to find out the water level in a graph and give a warning, and when the high level of water is above 10 cm and less than 17, a

notification will appear in the form of a yellow alert on the web. Figure 10 shows when the program works well.

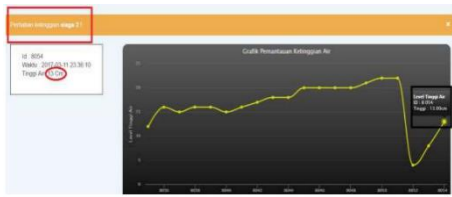


Fig. 10. Warning if high level is idle water 2

Figure 10 shows the water level is above 10 cm and below 17 cm. The notification displayed is “Attention Water Level Alert 2!” and turn yellow indicator light alert on.

I. Testing the Danger Alert Program

The danger alert warning program test was performed to find out the water level in a graph and give a warning, and when the high level of water is above 17 cm, a notification will appear in the form of a red alert on the web. Figure 4.20 shows when the program works well.

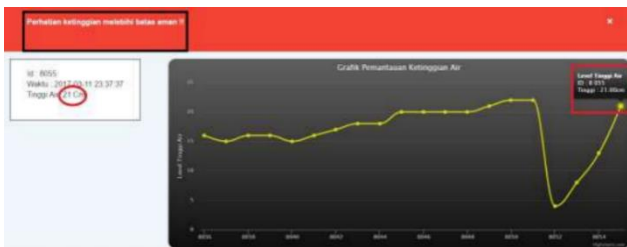


Fig. 11. Warning if the high level exceeds the safe limit

Figure 11 shows the water level is between 5 cm and 10. The notification displayed is “Attention to Water Height exceeds the safe limit!” and turn on the red indicator light alert.

J. The process of retrieving data per unit time in realtime

The test on the data based on the range aimed to find information on characteristics of water level based on hours, days, months, or years. The history menu provides a reference to see the characteristics of water that is read by the sensor day by day. The graph displays the data entered to the form by date.

K. Test on form

The form used is date to search a certain date to call the data of the date. The data are displayed in the form of a graph with a description of time and water level.

Fig. 12. Date Based Search Form

Figure 12 shows the form to retrieve the data recorded in the database. The data retrieval process starts when the form is filled with the date entered by the client.

L. Test on Time

The test this time is based on time that will display a water level for 24 hours. The graph in figure 13 shows the data stored in the database.

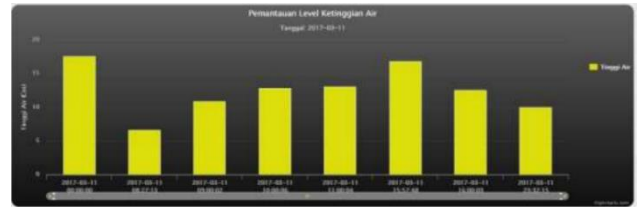


Fig. 13. Water level test per hour

The time (hour) taken from the database based on sensor data are on x-axis, while the water levels in centimeters (cm) are on y-axis. The yellow bars are a graph of the water level data based on the time entered in the yellow form. The yellow square contains information about the data of the time (hour) input in the form.

M. Test on Day

The test was based on the date that will display the water level from and to the desired date. The graph in Figure 14 shows the data stored in the database.

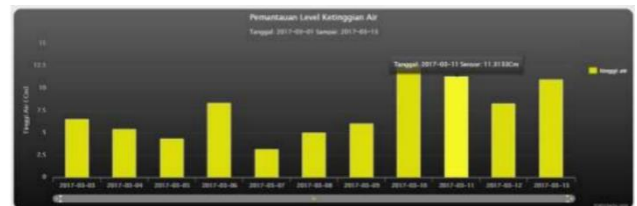


Fig. 14. Water levels test per day

The date taken from the database based on sensor data are on x-axis, while the water levels in centimeters (cm) are on y-axis. The yellow bars are a graph of the water level data based on the date entered in the yellow form. The yellow square contains information about the data of the day input in the form.

N. Test on Month

The test this time is based on monthly data collection which will show the water level from and to the desired month. The graph in figure 15 shows the data stored in the database



Fig. 15. Testing High Levels of Water per month

The month taken from the database based on sensor data are on x-axis, while the water levels in centimeters (cm) are on y-axis. The yellow bars are a graph of the water level data based on the month entered in the yellow form. The yellow square contains information about the data of the month input in the form.

O. Test on Year

The test was based on data collection per year which will display the water level from and to the specified year. The graph in figure 16 shows the data stored in the database.



Fig. 16. Water level test per year

The years taken from the database based on sensor data are on x-axis, while the water levels in centimeters (cm) are on y-axis. The yellow bars are a graph of the water level data based on the year entered in the form. The yellow square contains information about the data of the year input in the form.

IV. CONCLUSION

After designing, manufacturing, and testing the final project, the test results showed that a WEB that can display Water Level Height in Realtime that can work satisfactorily. The water level can be successfully identified by using indicator lights can work satisfactorily. An alarm alerting the water level exceeding the safe limit can work satisfactorily. A Web to find out the characteristics of water level unity of time can work satisfactorily.

REFERENCES

- [1] Z. Liu, G. Li, G. Mercier, Y. He, and Q. Pan, "Change Detection in Heterogenous Remote Sensing Images via Homogeneous Pixel Transformation," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1822–1834, Apr. 2018.
- [2] A. V. Monti-Guarnieri, M. A. Brovelli, M. Manzoni, M. Mariotti d'Alessandro, M. E. Molinari, and D. Oxoli, "Coherent Change Detection for Multipass SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6811–6822, Nov. 2018.
- [3] M. N. Sumaiya and R. Shanthal Selva Kumari, "Unsupervised change detection of flood affected areas in SAR images using Rayleigh-based Bayesian thresholding," *IET Radar, Sonar Navig.*, vol. 12, no. 5, pp. 515–522, May 2018.
- [4] D. Amitrano, G. Di Martino, A. Iodice, D. Riccio, and G. Ruello, "Unsupervised Rapid Flood Mapping Using Sentinel-1 GRD SAR Images," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3290–3299, Jun. 2018.
- [5] H. Sui, K. An, C. Xu, J. Liu, and W. Feng, "Flood Detection in PolSAR Images Based on Level Set Method Considering Prior Geoinformation," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 699–703, May 2018.
- [6] B. Pradhan, M. I. Sameen, and B. Kalantar, "Optimized Rule-Based Flood Mapping Technique Using Multitemporal RADARSAT-2 Images in the Tropical Region," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 10, no. 7, pp. 3190–3199, Jul. 2017.
- [7] M. Chini, R. Hostache, L. Giustarini, and P. Matgen, "A Hierarchical Split-Based Approach for Parametric Thresholding of SAR Images: Flood Inundation as a Test Case," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 6975–6988, Dec. 2017.
- [8] N. Pierdicca, L. Pulvirenti, G. Boni, G. Squicciarino, and M. Chini, "Mapping Flooded Vegetation Using COSMO-SkyMed: Comparison With Polarimetric and Optical Data Over Rice Fields," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 10, no. 6, pp. 2650–2662, Jun. 2017.
- [9] F. Chen, M. Zhang, B. Tian, and Z. Li, "Extraction of Glacial Lake Outlines in Tibet Plateau Using Landsat 8 Imagery and Google Earth Engine," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 10, no. 9, pp. 4002–4009, Sep. 2017.
- [10] T. M. Thekkil and N. Prabakaran, "Real-time WSN based early flood detection and control monitoring system," in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT)*, 2017, vol. 2018-Janua, pp. 1709–1713.
- [11] A. Y. Ardiansyah, R. Sarno, and O. Giandi, "Rain detection system for estimate weather level using Mamdani fuzzy inference system," in *2018 International Conference on Information and Communications Technology (ICOACT)*, 2018, pp. 848–854.
- [12] H. Yulindoko, Subono, V. A. Wardhany, S. H. Pramono, and P. Siwindarto, "Design of flooding detection system based on velocity and water level DAM with ESP8266," in *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISSEE)*, 2017, vol. 2018-Janua, pp. 396–401.
- [13] J. Priyadharshini, "Development of embedded web server for health care system using E-card," in *2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)*, 2017, vol. 2017-Janua, pp. 1–5.
- [14] M. Sahani, A. Nayak, R. Agrawal, and D. Sahu, "A GSM, WSN and embedded web server architecture for Internet based kitchen monitoring system," in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, 2015, pp. 1–6.
- [15] P. J. Alexander and N. Radhakrishnan, "Remote lab implementation on an embedded web server," in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, 2015, pp. 1–5.
- [16] I. Bosznai, F. Ender, and H. Santha, "Web server based remote health monitoring system," in *2009 32nd International Spring Seminar on Electronics Technology*, 2009, pp. 1–6.
- [17] M. Shahabuddin, M. R. T. Hossain, and M. R. Siyam, "Design and implementation of smart taxi fare meter and server based electronic payment system," in *2017 3rd International Conference on Electrical Information and Communication Technology (EICT)*, 2017, vol. 2018-Janua, no. December, pp. 1–5.
- [18] A. Sariga, S. Jaiganesh, T. Vengattaraman, and P. Sujatha, "An implementation of embedded web server in farming sector," in *2016 International Conference on Communication and Electronics Systems (ICES)*, 2016, pp. 1–6.
- [19] E. Raduca, L. Nistor, C. Hatiegan, M. Raduca, I. Padureanu, and S. Draghici, "Web server for command, control and monitoring of industrial equipment," in *2015 9th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*, 2015, pp. 61–66.
- [20] P. G. Salunke and A. M. Sayyed, "Design of embedded web server based on NIOS-II soft core processor," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, pp. 488–492.
- [21] F. H. Purwanto, E. Utami, and E. Pramono, "Design of server room temperature and humidity control system using fuzzy logic based on microcontroller," in *2018 International Conference on Information and Communications Technology (ICOACT)*, 2018, vol. 2018-Janua, pp. 390–395.
- [22] M. H. Siddiqui, V. Purohit, and S. Mane, "Embedded web server based NetLab for remote access," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, 2016, vol. 2016, pp. 1–5.
- [23] W. T. Sung, J. H. Chen, C. L. Hsiao, and J. S. Lin, "Multi-sensors Data Fusion Based on Arduino Board and XBee Module Technology," in *2014 International Symposium on Computer, Consumer and Control*, 2014, pp. 422–425.
- [24] A. F. Rachmani and F. Y. Zulkifli, "Trial and Evaluation of LoRa Performance for Smart System Multi-Client Model," *Proceeding - 2018 Int. Semin. Intell. Technol. Its Appl. ISITIA 2018*, pp. 39–44, 2018.

- [25] H. Singh, V. Pallagani, V. Khandelwal, and U. Venkanna, "IoT based smart home automation system using sensor node," in 2018 4th International Conference on Recent Advances in Information Technology (RAIT), 2018, pp. 1–5.
- [26] K. Kaushal et al., "The E-Restaurant," in 2016 Ninth International Conference on Contemporary Computing (IC3), 2016, pp. 1–5.
- [27] M. S. Nair, R. Jishnu, K. M. Rakesh, and A. Ramachandran, "Implementation of a web-based programming tool for distributed, connected Arduino systems," in 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, pp. 1595–1600.
- [28] J. P. Ventura, N. A. Cruz, and F. P. Lima, "A remote monitoring and control system for ecosystem replication experiments," in OCEANS 2016 MTS/IEEE Monterey, 2016, pp. 1–6.
- [29] Z. Davitadze, G. Partenadze, and E. Djincharadze, "Graphical visualization of data measurement of programmable microcontroller according to ARDUINO-project example," in 2016 IEEE East-West Design & Test Symposium (EWDTS), 2016, pp. 1–5.