Design and Implementation of Real-Time Flood Early Warning System (FEWS) Based on IoT Blynk Application

Hasbi Nur Prasetyo Wisudawan

Department of Electrical Engineering, Universitas Islam Indonesia, Yogyakarta, Indonesia
Corresponding Email: hasbi.wisudawan@uii.ac.id

*Abstract –***Disaster occurrence in Indonesia needs attention and role from all parties including the community to reduce the risks. Disaster mitigation is one of the ways to reduce the disaster risk through awareness, capacity building, and the development of physical facilities, for example by applying disaster mitigation technology (early warning system, EWS). EWS is one of the effective methods to minimize losses due to disasters by providing warning based on certain parameters for disasters which usually occur such as floods. This research promotes a real-time IoT-based EWS flood warning system (Flood Early Warning System, FEWS) using Arduino and Blynk as well as Global System for Mobile Communication network (GSM) as the communication medium. The steps for implementing FEWS system in real locations are also discussed in this paper. Parameters such as water level, temperature, and humidity as well as rain conditions that are read by the EWS sensor can be accessed in real-time by using android based Blynk application that has been created. The result of the measurement of average temperature, humidity, and water level were 28.6 ºC, 63.7 %, and 54.5 cm. Based on this analysis, the parameters indicated that the water level is in normal condition and there are no signs indicating that there will be flooding in the 30 days observation. Based on the data collected by the sensor, FEWS can report four conditions, namely Normal,** *Waspada Banjir* **(Advisory),** *Siaga Banjir* **(Watch), and** *Awas Banjir* **(Warning) that will be sent immediately to the Blynk FEWS application user that has been created.**

Keywords: Flood Disaster, Early Warning System, IoT, Blnyk Application, Arduino

I. Introduction

Geographical and astronomical location of Indonesia have influenced several aspects in this country, especially the economic sector [1]. Besides, the local climate is also affected causing Indonesia to have two seasons, i.e., rainy and dry seasons. Extreme weather as a result of tropical disturbance can trigger the occurrence of long drought and high rainfall in certain seasons [2]. Disaster caused by climate change and weather is usually called as hydro-meteorology disaster [3]–[5]. Apart from natural factors such as geological and hydro-meteorological aspects [6], low environmental resistance is also the cause of several disasters, such as fires and floods, in Indonesia.

According to Cambridge Dictionary, disaster is (an event that results in) great harm, damage, or death, or serious difficulty[7]. According to Law Number 24 of 2007 the explanation of disaster can be divided into three, namely natural, non-natural, and social disaster. Natural disasters are disasters caused by event or series of events caused by nature such as earthquake, tsunami, volcanic eruption, flood, drought, hurricane, and landslide. Disasters caused by non-natural event or series of events such as technology failure, modernization failure, epidemic, and disease outbreak are referred to non-natural disaster.

Besides the current Covid-19 pandemic, hydro-meteorological disasters, caused by extreme parameters for example rainfall, humidity, temperature, wind and other factors, occupy the highest disasters ranking in Indonesia. In July 2021, the National Disaster Management Agency (BNPB) recorded a total of 53, 42, and 22 occurrences of floods, forest and land fires, and hurricanes. Additionally, there were 11 landslides and one drought cases. [8]. Besides high rainfall, flood is also caused by massive environmental damage which leads to a decrease in environmental carrying capacity. The use of land without paying attention to the carrying capacity as well as the capacity and function of the watershed area (DAS) have also increased the amount of critical land in Indonesia so the disaster intensity has an increasing trend recently.

Systematic efforts to reduce the disaster risk, through physical development and capacity building, are required. It is referred to disaster mitigation [9]. In the Law (UU) Number 24 of 2020 Article 44 letter c, it is also stated that mitigation is the effort to reduce the disaster risk for the society living in the disaster-prone areas. These mitigation activities include the implementation of spatial planning, development and infrastructure arrangements as well as counselling and training activities. An effort that is continually developed in the disaster mitigation is the use and development of technology before (pre-disaster), during, and after a disaster (post-disaster).

An early warning system or EWS is the implementation of disaster mitigation technology [10], [11]. An example of EWS currently being used by Indonesian government is inaRISK which is a disaster risk assessment portal that provides the information of disaster threats, vulnerabilities, capacities, and risks. The inaRISK can also show the monitoring of disaster risk index [12]. More

specific disaster risk such as earthquake and tsunami can be monitored using Indonesia Tsunami Early Warning System (InaTEWS) developed by the Meteorological, Climatological, and Geophysical Agency in collaboration with National Disaster Management Agency (BNPB) and Agency for the Assessment and Application of Technology (BPPT) of the Republic of Indonesia [13]. Besides InaTEWS, another early warning system for other disasters such as landslide (Landslide Early Warning System, LEWS) has also been developed by BPPT [14]. Specifically, information related to flood can be monitored through Jakarta Flood Early Warning System (J-FEWS) [15] [16].

Apart from its applications, the development of mitigation technology, especially flood, is also continuously carried out by university researchers. Flood disaster prediction is conducted based on the continuous monitoring of weather and climate conditions [17]. The utilization of *Internet-of-Things* (IoT) technology using Raspberry Pi [18], Arduino [19], [20], and ESP8266 [21] can be used as disaster mitigation technology. However, the created tools are only limited to prototype and have not been implemented in the field or actual location.

This research proposed the design of a real-time flood early warning system (FEWS) and its application in the field. The location chosen to implement this early warning system was the Indonesian Islamic University (UII) reservoir, Integrated Campus, Jl. Kaliurang KM 14.5, Sleman, Special Region of Yogyakarta. The developed system is equipped with several sensors, i.e., water level float switch sensor, temperature and humidity sensor, and rain sensor. The sensor reading data was subsequently processed by Arduino and sent to the server by using GSM network. On the user side, Blynk application was used to monitor the result of sensor measurement in the reservoir in real-time. Parameters, such as temperature, humidity, weather condition (rain), and water level, monitored by Blynk IoT application are used as prediction materials. Other users can use this application by using sharing feature which will be explained in the Section III, Sub section C.

This study is similar to the flood EWS research in Citarum river which used Blynk application [22]. The application monitoring result is integrated to social media such as Twitter, but it did not discuss the detailed application of these tools in the field. Another research with application in the field was proposed by using social media Telegram [23]. However, the use of social media is not appropriate because the hazard information must be informed to the public widely and immediately. The use of applications such as Virtuino was also used to report various parameters such as water discharge, water level, and rainfall in real-time [21]. However, the tools developed were still prototypes and had not been implemented in the real location.

The EWS system can effectively be used if it is accompanied by the support of applicable policies [11]. This proposed EWS system is expected to be widely used by the community and able to gain attention and support from the stakeholders both regionally and nationally.

II. METHODOLOGY

The FEWS system design is required to make the system to be able to send, process, and receive information related to early warning properly as well as to minimize the occurrence of technical disturbances in the field. The system design was started by conducting survey to the location of FEWS installation. After doing some studies, the decided location was the Indonesian Islamic University (UII) reservoir as shown in Figure 1. It is approximately 8 m depth and 63.4 m width. During rainy seasons, water discharge is quite high and the water level can reach approximately 1 m from the surface as shown in Figure 2. The change in water condition was used as an input for water level sensor in addition to other parameters such as temperature, humidity, and rainfall.

Figure 1. UII Reservoir as the location of FEWS installation
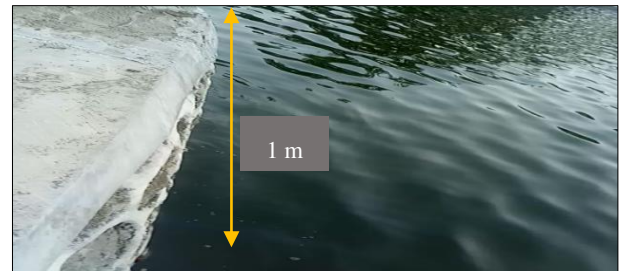

Figure 2. The distance between mean water level and the edge of the reservoir

This selection is not only intended for actual testing in the field, but to be used by the students as a learning medium at the same time. The system design consists of the designs of hardware, software, and communication system as well as *Blynk* application which will be explained in the following sub section..

A. Hardware Design

The flood EWS (FEWS) hardware consists of a main holder and an electronic circuit in form of micro-controller and sensors for water level, temperature, humidity, and rain. The designs of the main holder and sensor along with the other important elements are shown in Figure 3. The main holder and box panel have a very vital role as the protector of the electronic component inside so that the EWS tool can last for a long time in various weather conditions. The hardware required for this EWS were main holder, component house, and sensor holder. The height of the sensor holder was adapted to the condition of the river, especially the normal or average water level measured from the edge of the reservoir as shown in Figure 2. The holder is made from stainless steel

and all the electronic components, other than sensors, were placed inside the box panel.
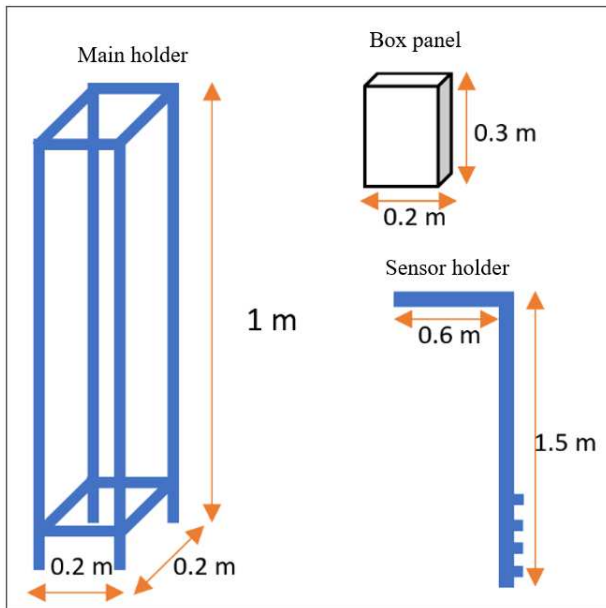

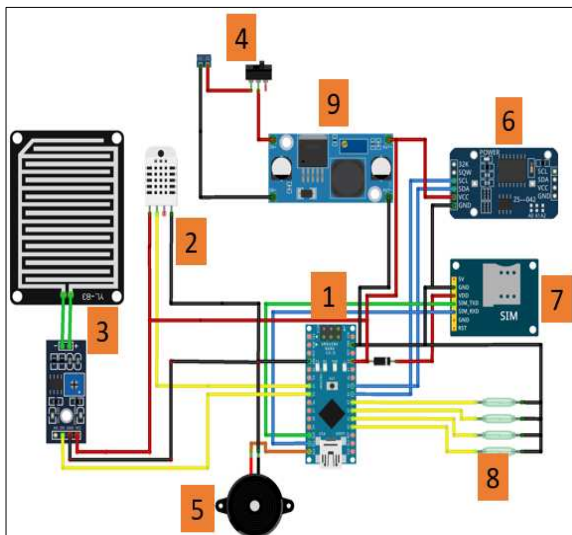**Figure 3**. The design of main holder and sensor holder

Figure 4. FEWS Electronic Circuit

As a micro-controller, Arduino Nano receives the input from all sensors and send it to Internet-of-Thing (IoT) server by using 800L module. The main electronic circuit is shown in Figure 4 with the connection inside the flood EWS circuit which consists of the following:

1. Arduino nano,
2. Temperature and humidity sensor (DHT 21),
3. Rain sensor and module (YL-83),
4. Switch,
5. Buzzer,
6. RTC DS3231,
7. SIM 800L,
8. Magnetic height sensor,
9. Power module.

(a)
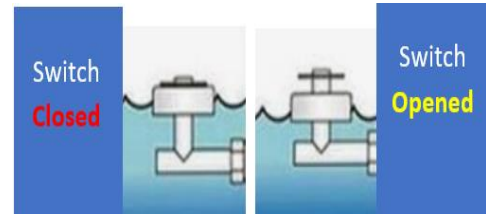
(b)

Figure 5. (a) Water level float (WLF) switch sensor [24], (b) Operation mode [25]

Water level sensor (*float switch sensor of water level*) informs the water condition in a certain height. The appearance of this sensor and how it works are shown in Figure 5(a) and Figure 5(b), respectively. When the water reaches certain heights, the buoy is pulled up so that the switch is closed.

This RTS module acts as a digital timer integrated with the temperature measuring feature. The RTC version used in this work is DS3231. This type of RTC has several advantages compared to the previous version (DS1302) especially the VCC which ranges from 2.3 to 5.5 Volt. Moreover, there are SDA and SCL pins that function as interfaces with micro-controller such as Arduino UNO.

Buzzer acts as a component that converts electrical energy into vibration or sound. It is very important as an indicator for certain processes which require vibration or sound. In the early warning system, sound notification is usually used as the hazard indicator. This buzzer can be replaced with alarm which has wider range of sound and can be modified based on the needs.

The correlation between pins on Arduino Nano with the three sensors, shown by number 2, 3, and 8 in Figure 4, is shown in Table 1. Other components, i.e., real time clock (RTC) module, Buzzer, and SIM 800L are shown in Table 2. All of the electronic components in Figure 4 were placed in the box panel as in Figure 3. DHT21 component in Table 1 reads the temperature between -40 to 80 °C and humidity in the range between 0 to 99.9% with the accuracy of ± 3% RH and ± 0.5 °C, respectively. There are 3 pins used on DHT21, namely VCC, GND, and Data. Data Pin is directly connected to digital D4 Pin in Arduino. YL-83 is a sensor to find out the condition of the surrounding whether it is raining or not. This sensor has two outputs: digital (0 or 1) and analog (in strains), represented by D0 Pin which is connected to D5 Pin in Arduino. The height indicator is monitored by water level float switch sensor based on Pin 1, 2, 3, and 4 where each of them is connected to Pin A0, A1, A2, and A3 in Arduino.

Table 1. The Correlation between Sensor Pins

Component	Sensor Data Pin	Arduino UNO Pin
DHT 21	Pin Data	Pin D4
Sensor YL-83	Pin D0	Pin D5
Water level float switch	Pin 1	Pin A0
	Pin 2	Pin A1
	Pin 3	Pin A2
	Pin 4	Pin A3

Table 2 shows the correlation between communication component and sound notification such as RTC, Buzzer, and SIM 800L with Arduino. RTC component has SCL and SDA Pins which are connected to Pin A4 and A5 to communicate. Besides RTC, there is also Buzzer which gives sound notification that is connected to D12 Pin. The sending of the sensor reading result was carried out by using Pin TX_D and Pin RX_D in the GSM SIM 800L Module where each of them is connected to Pin D10 and D11, respectively.

Table 2. The Correlation between Communication Pin and Buzzer

Component	Sensor Data Pin	Arduino UNO Pin
RTC	Pin SCL	Pin A4
	Pin SDA	Pin A5
Buzzer	Pin 2	Pin D12
SIM 800L	Pin TX_D	Pin D10
	Pin RX_D	Pin D11

B. Software Design

Software design includes program design by using Arduino IDE. There are three main parts, i.e., sensor connection and notification program of water level float switch, DHT21 connection program, and rain sensor connection program (YL-83) which are explained as follows.

1. DHT21 Sensor Connection

DHT21 sensor is used to measure temperature and humidity. Parameters such as temperature and humidity are factors that can be considered in predicting the flood. In some places, continuous rain and high humidity are the triggers of flooding. Data reading is applied by using the following codes

float h = dht.readHumidity() and

float t = dht.readTemperature();

The pseudocode of this sensor reading program is shown in Figure 6.

```
Define DHTPIN 4    //Set Pin D4
Define DHTTYPE DHT21   // DHT 21 (AM2301)
Read humidity input h = dht.readHumidity();
Read temperature input t = dht.readTemperature();
if (isnan(h) || isnan(t)) {
Set "SENSOR FAIL!"
end
```

Figure 6. The pseudocode DHT21reading sensor program

Temperature is measured in degrees Celsius (°C) while humidity is measured in percentage (%). Based on the Regulation of the Minister of Health of the Republic of Indonesia No. 1077/Menkes/PER/V/2011 on Guidelines for Indoor Air Sanitation, the ideal humidity is expressed in relative humidity (RH) ranges from 40 to 60 % [26]. The air is stated as very humid if the RH is above 60%. If it is less than 40%, the humidity is concluded as too dry.

2. Water level float (WLF) Sensor Connection

There are four connections which will be reported to the users as notifications: Normal, *Waspada Banjir* (Advisory), *Siaga Banjir* (Watch), and *Awas Banjir* (Warning). This notification will be sent directly based on the reading of the four sensors in Figure 5 to the Blynk application users. The pseudocode of this water level sensor program is shown in Figure 7.

```
Define Pin  A0  for sensor WLF 1 //sensor WLF 1
Define Pin  A1  for sensor WLF 2  // sensor WLF 2
Define Pin  A2  for sensor WLF 3 // sensor WLF 3
Define Pin  A3  for sensor WLF 4 // sensor WLF 4
// Define water level notification
if WLF1, WLF2, WLF3 and WLF4 are high // Default
      Set "NORMAL" // Display Normal Level
elseif WLF4 is low
      Set "AWAS BANJIR"
elseif WLF3 is low
      Set "SIAGA BANJIR"
elseif WLF2 is low
      Set "WASPADA BANJIR"
else WLF1 is low
      Set "NORMAL"
endif
```

Figure 7. Pseudocode reading program and notification of the water level float switch sensor

3. YL-83 Sensor Connection

The rain sensor used in this EWS trial is the YL-83 sensor. It has a dimension of 5.5 cm × 4 cm and uses a voltage in the range of 3.3 to 5 volt with a controller separated from the main sensor. The result of sensor reading will be displayed in two conditions, namely raining and non-raining condition as shown in Figure 8.

```
Define Pin Data 5   // pin D5

digitalRead(hujan) // Read data sensor

    if digitalRead(hujan) is low

      set  "HUJAN"

    else

      set "TIDAK HUJAN"

    endif
```

Figure 8. Pseudocode YL-83 sensor reading program

C. Setting of Communication System and IoT Application

1. Setting of SIM800L

Communication system used in this research is GSM because there is no internet network or WiFi in the EWS installation location. The setting of this communication system is shown in Figure 9.

```
Define GSM Module (TINY_GSM_MODEM_SIM800)

Input Library TinyGsmClient.h

Input BlynkSimpleTinyGSM.h

// Setting GPRS

Set apn, user, and pass to "3gprs"

Leave empty, if missing user or pass
```

Figure 9. Pseudocode SIM800L setting program

2. Setting of Application and Connection to Blnyk

Blnyk is an IoT platform which can be used to control hardware through mobile network or internet [27]. This platform consists of three main parts, i.e., Blynk App, Blynk Server, and Blynk Library. The main part of the communication to Blynk is the token generated during registration in the application.

The steps to create an application using Blynk in the trial case of the EWS tool can be explained as follows.

1. Download the Blynk official application (legacy) on Playstore.
2. After downloading ends, next step is *Create New Account* by entering an active email and password.
3. In My Devices, click New Device then enter:
   - Device name: for example, project1
   - Hardware Model: Arduino Nano
   - Connection Type: GSM

4. Authentication Code or Token Delivery.

Athentication code or token will be sent to user email authomatically This Token will be used as a key to connect Blynk application with the FEWS device. In this case, the token used in this FEWS is

char auth[] = "DB3SfD9Mvy2dchOByCoMT5WhBQJ9XXXX";

The token is entered to Arduino main program.

III. FINDING AND DISCUSSION

A. Result of Hardware Design of FEWS Holder

This holder, consisting of main holder and sensor, has a very vital role in the developed FEWS. The result of the holder designs is shown in Figure 10 according to the size defined in Figure 3. Apart from protecting the electronic components, the developed system has a power supply coming from solar panels placed on top of the holder. The battery in the panel box acts as an energy store to supply electrical energy to all of the electronic components. The electronic circuit in Figure 11 is implemented according to the design in Figure 4.
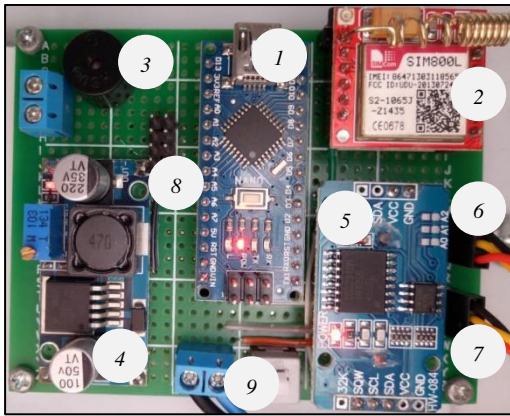

Figure 10. FEWS holder

Figure 11. Major operator electronic network

There are several components on the network, i.e., 1. Arduino Uno, 2. SIM 800L, 3. Buzzer, 4. Power Module, 5. RTC, 6. DHT 21 Pin Sensor, 7. YL-83 Pin Sensor, 8. Water Level Pin Sensor, 9. Power Switch.

B.   Result of Software Design as an Android Application

The software used in this study is Blynk application which is available for free on Playstore. Users can design easily with standard components provided by Blynk as long as the quota is available. The addition of certain components can be done by making payments in the application. There are four conditions reported by a sensor, processed by a microprocessor, and sent to Blynk application on android. They are the temperature, humidity, rainy or light condition, and information of the water condition: Normal, *Waspada Banjir* (Advisory), *Siaga Banjir* (Watch), and *Awas Banjir* (Warning). The result of the application design is shown in Figure 12.

(c)

Figure 12. Blynk EWS interface view for flood disaster

C.   Application Sharing to Users

The application shown in Figure 12 can be used by other users who have downloaded and installed the application. Generally, the purpose of this EWS is to provide the widest possible information or notification to users. The steps for sharing the application are as follows.

1. Download the application via these links (only for those who do not have Blynk application)

Download Blynk App: http://j.mp/blynk_Android or http://j.mp/blynk_iOS

2.   Scan the QR-Code in Figure 13.

Figure 13. QR-Code to get the designed FEWS application

3.   The application has been installed on the user's device and is ready to use.

D.   Implementation of FEWS in UII Reservoir

The FEWS installation steps began with site preparation including licensing, installation of the main holder and sensor holder, installation of the main board, installation of YL-83, DHT21, and water level sensor as shown in Figures 14, 15, and 16. The next step was installation and FEWS electronic circuit testing in Figure 17.

(a)   (b)

Figure 14. Site Preparation

Figure 15. Installation of Main Holder
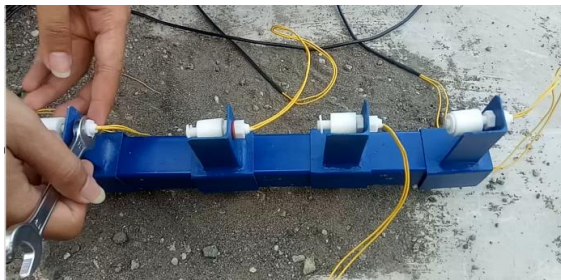
(a)

(b)   (c)

Figure 16. Installation of (a) Altitude Sensor, (b) YL-83 Sensor, and (c) DHT21 Sensor

Water level sensor will measure the water level in the reservoir. WLF1, WLF2, WLF3, and WLF4 sensors in Figure 16a will show some changes of the water level in the reservoir. Each level will trigger a notification to the user in the form of Normal, *Waspada Banjir* (Advisory), *Siaga Banjir* (Watch), and *Awas Banjir* (Warning). Before implementing in the field, the sensors in Figure 16 were tested in Electronics Laboratory to ensure that they function properly and provide minimum error. An example of temperature measurement by DHT 21 sensor was compared to the result of room temperature

measurement. The altitude sensor was also tested by applying it in the water along with the expected notifications. YL-83 sensor testing is simple, i.e., by dripping water on the sensor surface as long as the notifications appear on the Blynk application screen.

Figure 17. EWS Electronic Component Installation and Testing

The normal voltage of the solar charge controller is 14.4 volt which is sufficient to supply the power needs for all EWS electronic components. The process of charging from solar panel was operating properly indicated by an arrow appearing under the voltage (Figure 17).

(a)

(b)

Figure 18. FEWS installation. (a) Main holder structure with solar panel, (b) sensor holder structure
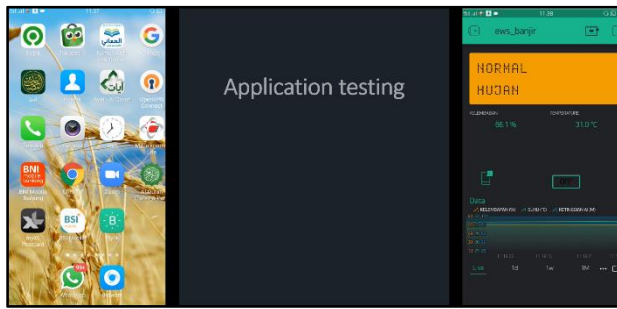
Figure 19. Application Testing in the Field

The overall appearance of the tools is shown in Figure 18a, while water level sensor is shown in Figure 18b. There are four levels of sensor from level one to level four, each of which indicates Normal, *Waspada Banjir* (Advisory), *Siaga Banjir* (Watch), and *Awas Banjir* (Warning). The installed Blynk application on a Smartphone as depicted in Figure 19, shows that the designed application has successfully shown the result of monitoring parameters in the field (UII reservoir).

E. Measurement Results

The result of observation conducted for approximately one month or thirty days is shown in Figure 20. The graph shows three parameters, i.e., temperature, humidity, and water level. The details of each magnitude are shown in Table 3. This data is directly generated from one of the Chart components in Blynk. The average temperature, humidity, and water level are 28.6 °C, 63.7 %, and 54.5 cm, respectively. Although the humidity was quite high but the water condition was still within the normal limit. The selection of this UII reservoir was not only based on the geographical aspect but also based on the strategic consideration. First, besides observing temperature, humidity, and raining/non raining condition, there are changes in water level that can be observed using sensor. Sensor reading data must also be guaranteed to be sent to application users with minimum errors. The second strategic consideration is that the FEWS created can be used as a live learning medium for students especially UII Electrical Engineering students and students outside of UII. The steps of creating this FEWS can be downloaded for free in http://tiny.cc/FEWS01.
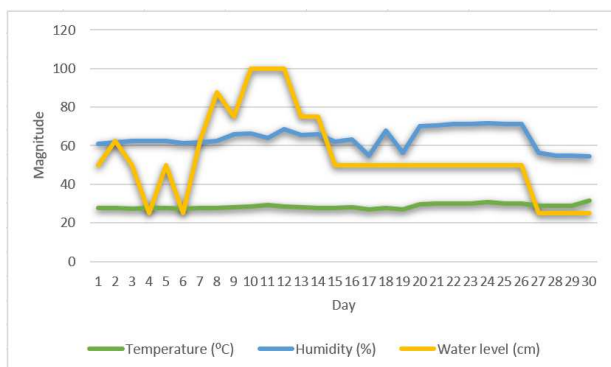

Figure 20 Graph of Temperature, Humidity, and Rainfall Measurement Result in 30 days

Table 3. Temperature, Humidity, and Water Level Measurement Result in 30 days

Day	Temp. (°C)	Humidity (%)	Water Level (cm)
1	27.7	60.9	50
2	27.7	61.7	62.5
3	27.5	62.4	50
4	27.7	62.65	25
5	27.8	62.6	50
6	27.5	61.3	25
7	27.6	61.55	62.5
8	27.6	62.6	87.5
9	28.1	65.9	75
10	28.3	66.2	100
11	29.2	64	100
12	28.5	68.7	100
13	28	65.6	75
14	27.8	65.9	75
15	27.9	62.1	50
16	28.2	63.4	50
17	27.1	54.8	50
18	27.85	67.65	50
19	27.1	56.3	50
20	29.75	70.1	50
21	29.9	70.5	50
22	30.15	71.2	50
23	30.15	71.2	50
24	30.65	71.5	50
25	30.2	71.3	50
26	30.2	71.3	50
27	28.8	56.2	25
28	28.7	54.7	25
29	28.7	54.8	25
30	31.7	54.5	25

The data in Table 3 is summarized based on the measurement results of the FEWS sensors in reservoir. These data were extracted from **SuperChart** module in the Blynk application via **Export to CSV** menu. Based on these data, the highest water level occurred on the 10th, 11th and 12th days because on those days it rained quite heavily and the humidity was relatively high (68 %).

IV. **CONCLUSION**

The research on a real-time flood early warning system by using Arduino and Blynk application and its implementation in the field (UII reservoir) has successfully been carried out. Three sensor data, i.e., water level, rain, temperature, and humidity can be accessed by using the created Blynk application. The statistical data for a certain duration can be directly

generated as an analysis material. Based on the statistical results, the average temperature, humidity, and water level were 28.6 °C, 63.7%, and 54.5 cm, respectively. These data showed that the measurement results are still within the normal limits. The Blynk application can easily be installed on the user's phone. Notifications on temperature, humidity, water level, and raining/non raining conditions, as well as Normal, *Waspada Banjir* (Advisory), *Siaga Banjir* (Watch), and *Awas Banjir* (Warning) conditions can be sent properly to the user of FEWS application implemented in UII reservoir.

REFERENCES

[1] D. Sungkawa, *Buku Ajar: Geografi Regional Indonesia*, 1st ed. Jurusan Pendidikan Geografi, Fakultas Pendidikan Ilmu Pengetahuan Sosial, Universitas Pendidikan Indonesia, 2010.

[2] D. Hermon and E. Barlian, *Mitigasi bencana hidrometeorolgi: banjir, longsor, ekologi, degradasi lahan, puting beliung, kekeringan*. Padang: UNP Press, 2012.

[3] R. Z. Singal and N. Jumario, "Pemetaan Curah Hujan Dalam Upaya Mengurangi Resiko Bencana Hidrometeorologi Dengan Sistem Informasi Geografis (SIG) Untuk Wilayah Kalimantan Utara," *J. Tek. Sipil*, vol. 3, no. 2, p. 14, 2019.

[4] S. H. S. Hengkelare and O. H. A. Rogi, "Mitigasi Resiko Bencana Banjir di Manado," *J. Perenc. Wil. Dan Kota*, vol. 8, no. 2, p. 8, 2021.

[5] "Bencana Hidrometeorologi, Apa itu? – Konservasi DAS." https://konservasidas.fkt.ugm.ac.id/2017/03/23/bencana-hidrometeorologi-apa-itu/ (accessed Sep. 01, 2021).

[6] T. Yanuarto, P. Sridewanto, A. Cipto Utomo, and I. Toto Satrio, *Buku Saku Tanggap Tangkas Tangguh Menghadapi Bencana*, 4th ed. Pusat Data Informasi dan Humas BNPB, 2019.

[7] https://dictionary.cambridge.org/dictionary/english/disaster (accessed Sep. , 2021).

[8] "Review Bencana Juli 2021: Bertemunya Bencana Hidrometeorologi Basah dan Kering - BNPB." https://www.bnpb.go.id/berita/review-bencana-juli-2021-bertemunya-bencana-hidrometeorologi-basah-dan-kering (accessed Sep. 20, 2021).

[9] C. E. Haque, Ed., *Mitigation of natural hazards and disasters: international perspectives*. Dordrecht: Springer, 2005.

[10] Azmeri and E. Fatimah, *SIDIK CEPAT, Ancaman Banjir Bandang*, 1st ed. Yogyakarta: DeePublish, 2017.

[11] "Early Warning Systems A State of the Art Analysis and Future Directions," *Environ. Dev.*, vol. 4, pp. 136–171, Oct. 2012, doi: 10.1016/j.envdev.2012.09.004.

[12] "inaRISK." https://inarisk.bnpb.go.id/ (accessed Sep. 01, 2021).

[13] "Indonesia Tsunami Early Warning System (InaTEWS - BMKG) - Versi Bahasa." https://inatews.bmkg.go.id/ (accessed Sep. 01, 2021).

[14] R. Fitriani *et al.*, "Pembangunan Sistem Peringatan Dini Bencana Longsor di Kampung Jatiradio, Desa Cililin, Kecamatan Cililin, Kabupaten Bandung Barat, Provinsi Jawa Barat," *J. Alami*, vol. 3, no. 2, p. 8, 2019.

[15] "JFEWS Bulletin." https://buletin-jfews.pusair-pu.go.id/ (accessed Sep. 01, 2021).

[16] S. Ginting and W. M. Putuhena, "Sistem Peringatan Dini Banjir Jakarta-Flood Early Warning System (J-FEWS)," *J. Sumber Daya Air*, vol. 10, no. 1, pp. 71–84, May 2014.

[17] M. M. Orozco and J. M. Caballero, "Smart disaster prediction application using flood risk analytics towards sustainable climate action," *MATEC Web Conf.*, vol. 189, p. 10006, 2018, doi: 10.1051/matecconf/201818910006.

[18] J. P. Nainggolan, M. E. I. Najoan, and S. D. S. Karouw, "Pengembangan Sistem Informasi Peringatan Dini Banjir Di Kota Manado Berbasis Internet of Things," *J. Tek. Inform.*, vol. 15, no. 1, Art. no. 1, Mar. 2020, doi: 10.35793/jti.15.1.2020.29064.

[19] M. Z. Arifin, E. Utami, and E. Pramono, "Perancangan Sistem Deteksi Dini Bencana Banjir Menggunakan Teknik Pengiriman DTMF Berbasis Modul RF 433 Mhz Dan Arduino," *J. Teknol. Inf. Dan Komun. TIKomSiN*, vol. 8, no. 2, Oct. 2020, doi: 10.30646/tikomsin.v8i2.465.

[20] D. Danang, S. Suwardi, and I. A. Hidayat, "Mitigasi Bencana Banjir dengan Sistem Informasi Monitoring dan Peringatan Dini Bencana menggunakan Microcontroller Arduino Berbasis IoT," *TEKNIK*, vol. 40, no. 1, p. 55, Jul. 2019, doi: 10.14710/teknik.v40i1.23342.

[21] S. S. Mahardika, W. Kurniawan, and F. A. Bakhtiar, "Implementasi Sistem Real Time untuk Pendeteksi Dini Banjir berbasis ESP8266 dan Weather API, " *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer,* vol. 3, no. 8, p. 8238-8247, Agustus 2019.

[22] A. R. Alfaridzi, E. Kurniawan, and A. Sugiana, "IoT Blynk untuk Sistem Monitoring Pendeteksi Dini Banjir Sungai Citarum Terintegrasi Media Sosial," *e-Proceeding of Engineering,* vol. 7, no. 1, p. 43-52, 2020

[23] Siregar and P. Henrico Samuel, "Desain dan Implementasi Early Warning System Bencana Banjir menggunakan Sensor Ultrasonic dengan Notifikasi Via Telegram." Universitas Sumatera Utara, 2021.

[24] "Float Switch or Float Sensor with Arduino » PIJA Education," *PIJA Education*. https://pijaeducation.com/arduino/sensor/float-switch/ (accessed Sep. 20, 2021).

[25] S. Alimsyah and L. Mawaryuningtyas, "Automasi Sistem Pelumasan Terukur Pada Perakitan Elemen-Elemen Mesin Berbasis Mikrokontroler," *J. Sinusioda*, vol. xx, no. 1, pp. 71–78, 2018.

[26] "Peraturan Menteri Kesehatan Republik Indonesia Nomor 1077/Menkes/per/v/2011 Tentang Pedoman Penyehatan Udara dalam Ruang Rumah." Menteri Kesehatan Republik Indonesia, 2011.

[27] "Blynk IoT platform: for businesses and developers." https://blynk.io/ (accessed Sep. 20, 2021).