2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT), October 23, 2017, Jakarta, Indonesia

Design of Real-time Weather Monitoring System Based on Mobile Application using Automatic Weather Station

Aris Munandar*, Hanif Fakhrurroja, Muhammad Ilham Rizqyawan, Rian Putra Pratama, Jony Winaryo Wibowo, Irfan Asfy Fakhry Anto
Technical Implementation Unit for Instrumentation Development (UPT BPI)
Indonesian Institute of Sciences (LIPI)
Bandung, Indonesia
*aris001@lipi.go.id

***Abstract*—This paper proposes the design of real-time weather monitoring system based on a mobile application using Automatic Weather Station (AWS). The system connects to the AWS equipped with several sensors for collecting data and storing the data to the web server. Data from weather sensor is taken from the AWS-Davis Instrument using the WeatherLink software. The data is transmitted through the data logger using serial communication, uploaded via FTP and stored on a web-server. The Android application reads the files and displays the information provided by the web server in real-time. The system has successfully show real-time monitoring of weather through the mobile application with a flexibility in the parameters and the need of user interface (UI) design compared to the other solution.**

Keywords—real-time monitoring; automatic weather station; mobile application

I. Introduction

Weather is an air condition in a certain place and in the relatively short time that includes conditions of temperature, humidity, and barometric pressure as its main component. Weather changes can be observed by using a device called Automatic Weather Station (AWS). AWS has been widely applied in various fields such as environmental research for geo-statistical[1], analysis of temperature measurement [2], prediction of wind energy potential location [3], measurement of the movement of the mass balance [4], and estimation of crop water needs [5].

There are many kinds of AWS used for a monitoring system. One of them is the Wireless Vantage Pro 2 weather station by Davis Instrument. To display the data on a website or mobile devices, an additional module called WeatherLinkIP is needed. This module serves as a data logger to upload data from the console Vantage Pro directly to the website Weather link. The weaknesses of this system are the limited user interface and the additional fee. The user must pay the modules purchased separately. Furthermore, users cannot change the default user interface and manually select the required data to be displayed on the website or mobile devices.

This research aims to develop a custom information system and UI design for the weather monitoring system. Therefore, it can be managed on a standalone web server and can be developed on a mobile application such as Android.

II. Automatic Weather Station

AWS is a meteorological station to observe the weather and to send the results automatically [6][7]. In AWS, measuring tools reads or receives a data using the data-acquisition device unit. The data from measurement devices can be processed locally at the AWS itself or processed in other places such as the central data processing [7]. AWS can be designed in an integrated manner using a variety of measurement devices such as integration of instrument systems, interfaces and processing, and transmission unit called Automatic Weather Observing System (AWOS) or Automatic Surface Observing System (ASOS).

Based on presentation of the data, AWS can be grouped into real-time AWS and off-line AWS. Real-time AWS is a weather system that presents data in real-time. In general, this AWS is equipped with communication and alarm system to alert the user in case of extreme weather conditions. A real-time AWS features has the collection of data units, the data storage and the wireless data communications using GSM/GPRS module that has capability for providing data communications in a wide range [8]. Off-line AWS is weather stations that only record data and store data on storage media. Stored data can be retrieved at any time as necessary [9]. AWS has features with several sensors, including a thermometer for measuring the temperature, an anemometer for measuring wind speed and direction, a hygrometer for measuring humidity, a barometer for measuring air pressure, a rain gauge for measuring rainfall and pyrometer for measuring solar radiation.

III. Methods

A. System Design

Wireless weather-link Vantage Pro 2 by Davis is used as the AWS tool for the system development. Fig. 1 shows the

978-1-5386-0510-3/17/$31.00 ©2017 IEEE

design of the weather monitoring system. Weather sensor consists of a rain sensor, temperature sensor, humidity sensor, wind speed sensor and a solar radiation sensor. It is placed on the roof of UPT BPI LIPI Building with the height of approximately 17 meters from ground level. The PC Server and the data logger is placed on the second floor. The web server can be accessed at http://meas.bpi.lipi.go.id with IP 192.168.236.99.

Wireless transmission with the range of 300 meters is used for the outdoor data transmission from the weather sensors to the data logger. Weather data is processed on the web server and can be accessed via the internet using a PC, tablet, or mobile device.

B. *AWS Data management*

AWS data management required data of weather parameter is taken from the Vantage Pro data logger with data acquisition process using serial communication. The process of data acquisition uses WeatherLink software for Windows version 5.7.1. The software is used to configure the station, set the unit of measure components of the weather, set the configuration of the Internet, and retrieve weather data from the Vantage Pro data logger console.

There are two types of data taken from the Vantage Pro data logger console. The first type is the image data that consists of current image data indicating the state of the current weather and historical data image. It present the changes of the weather in 24 hours. The second type is data in the form of HTX file. The HTX file is used to design the UI display and to store data temporarily. This file will be converted to a PHP file. The conversion result of HTX file into PHP file is shown in Fig. 2.

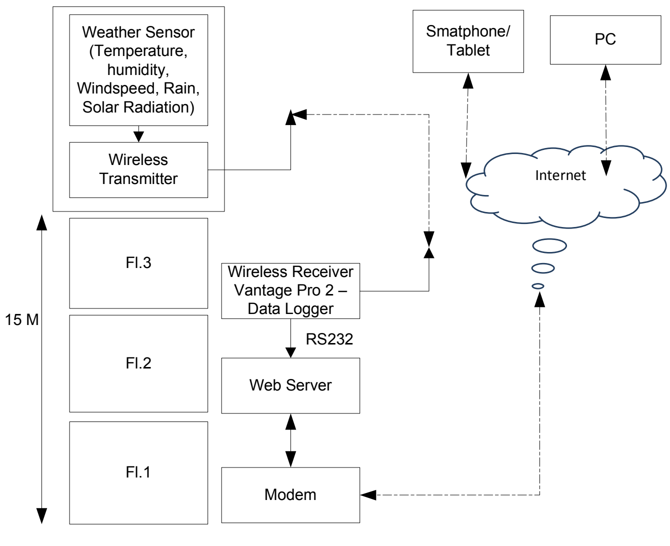


Fig. 1. Design of weather monitoring system.

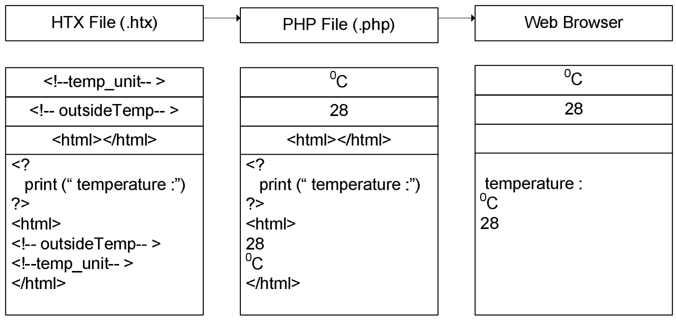


Fig. 2. HTXfiles conversion into PHP files.

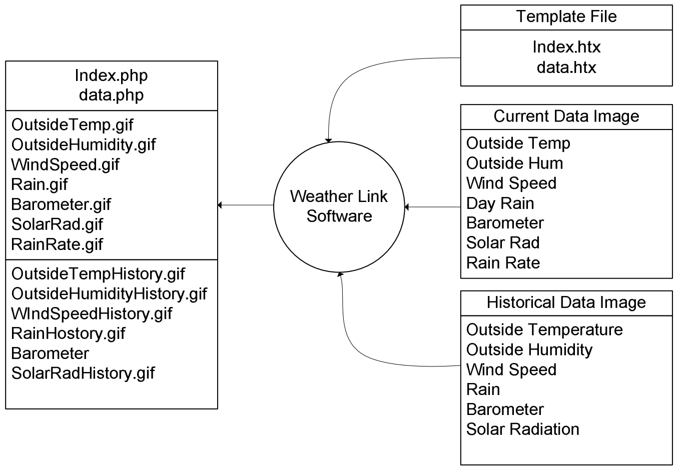


Fig. 3. Web design files scheme of weather monitoring system.

The HTX file consists of HTML, PHP, and weather tags with <! - -> format. Whereas, the PHP file consists of the HTML and PHP tag. The weather tag is translated to the real value of the tag itself. Web design scheme files is used to design a monitoring system as shown in Fig. 3.

Template files of the data that will be uploaded to the web server are shown in TABLE I. The template of image data on current weather and the file template image data of 24 hours historical weather are shown in 0TABLE III.

TABLE I. TEMPLATE FILES UPLOADED TO THE WEB SERVER

Input File .htx	Output File (.php)	Description
index.htx	index.php	Files that will be accessed when you first opened web page provides weather tag <! - ->
data.htx	data.php	Row files that stores data used to determine the weather parameter at sunrise, sunset, rain, and no rain in the displayed User Interface

TABLE II. IMAGE DATA CURRENT WEATHER PARAMETER

Parameter Input	File Output (.gif)	Description
Outside Temp	OutsideTemp.gif	Current state of the temperature (ºC)
Outside Hum	OutsideHumidity.gif	Current state of humidity (%)
Wind Speed	WindSpeed.gif	Current state of wind speed (Km/h)
Day Rain	Rain.gif	Current state of rainfall situation at this time (mm)
Barometer	Barometer.gif	Current state air of barometer (mb)
Solar Rad	SolarRad.gif	Current state of solar radiation (W/m^2)

TABLE III. IMAGE DATA 24 HOURS HISTORICAL WEATHER PARAMETER

Parameter Input	File Output (.gif)	Description
Outside Temperature	OutsideTempHistory.gif	Graph of outside temperature changes during 24 hours
Outside Humidity	OutsideHumidityHistory.gif	Graph of outside humidity changes during 24 hours
Rain	RainHistory.gif	Graph of rain changes during 24 hours
Barometer	BarometerHistory.gif	Graph of barometer changes during 24 hours
Solar Radiation	SolarRadHistory.gif	Graph of solar radiation changes during 24 hours

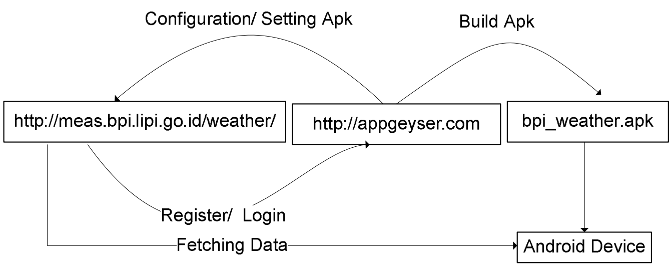


Fig. 4. Android application making use AppGeyser.

PHP files and image files are uploaded to the web server using FTP with the data transmission time interval for 5 minutes. Then, the data is read by the mobile application that installed on mobile device. The mobile application is created using open source web service named AppGeyser (www.appsgeyser.com) which serves to make the web interface in android application packet format (APK). The processing and the reading of weather data in the application are shown in Fig. 4.

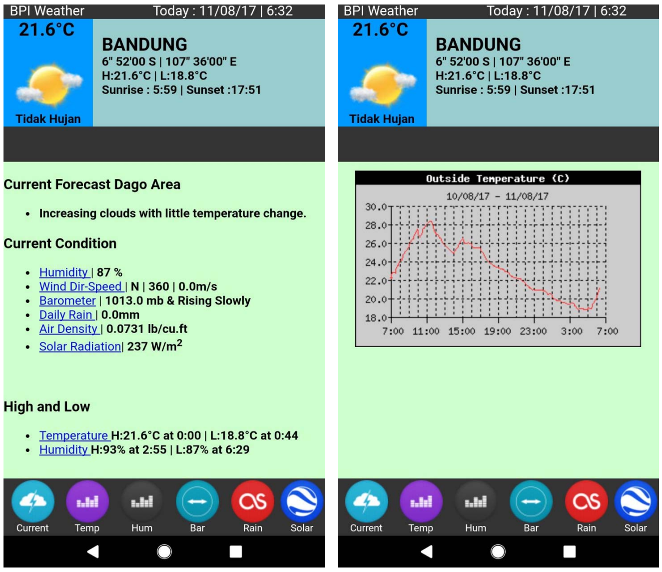
IV. EXPERIMENTAL RESULTS

The system has been tested using a mobile device with the 5.5-inch screen and a resolution of 720 x 1280 pixels (~ 267 ppi pixel density) using Android operating system MIUI v.6 Kitkat 4.4.2. The main display consists of six menus, namely:

- current menu: display all current measurement parameters of weather conditions;
- menu-temp: display the graphs of temperature changes;
- menu-hum: display the graphs of humidity changes;
- menu-bar: display the graph of air pressure changes;
- menu-rain: display the graph of precipitation changes;
- menu-solar: display the graph of solar radiation changes. This menu can be seen in Fig. 5.

Data changes will occur during the menu transition (page refresh) or when the 5-minute time interval has run-out. For example, to determine the status of rain, the data is taken by reading the data in the dat.php file. The status will display "rain" when the value of the <! - Rain -> is greater than 0.0 mm. The changes of weather parameters are displayed in 24-hour time scale, so the characteristic of weather parameter changes can be observed by hourly-basis.

To optimize the usability of the user interface, the color on the graph changes based on the weather parameter. This way, it is easier to distinguish each parameter. The temperature changes, humidity changes, air pressure changes, precipitation changes, and solar radiation changes are indicated by red, cyan, black, blue, and yellow, respectively. Fig. 6 to Fig. 10 show the graphics display of weather parameter changes during 24-hour on the mobile device.



(a) (b)

Fig. 5. Weather monitoring system in the Android application a) The current weather condition, b) Graph changes the parameters for 24 hours.

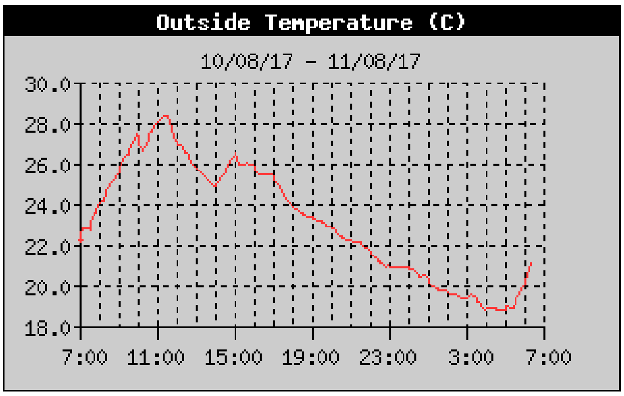


Fig. 6. Graphic display of temperature change for 24 hours.

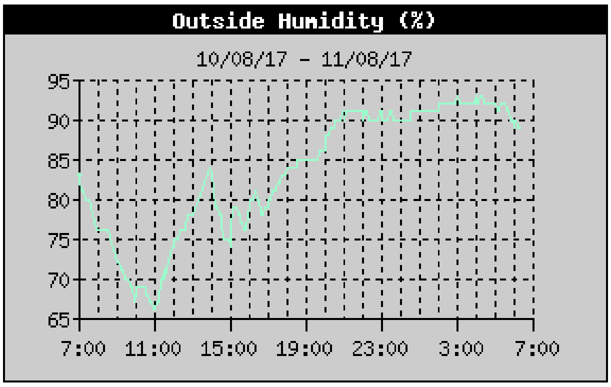


Fig. 7. Graphic display of humidity change for 24 hours.

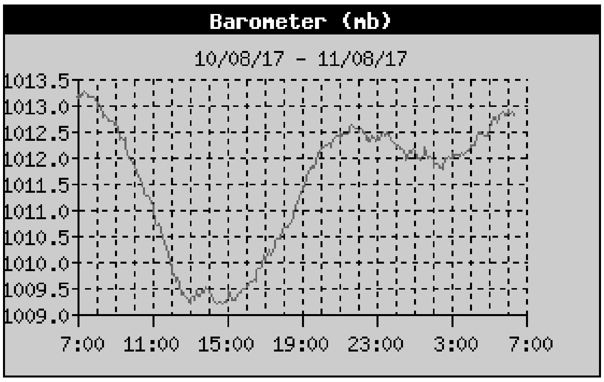


Fig. 8. Graphic display of barometer change for 24 hours.

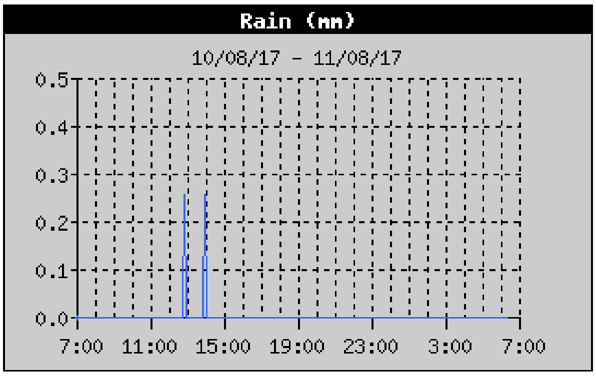


Fig. 9. Graphic display of daily rain for 24 hours.

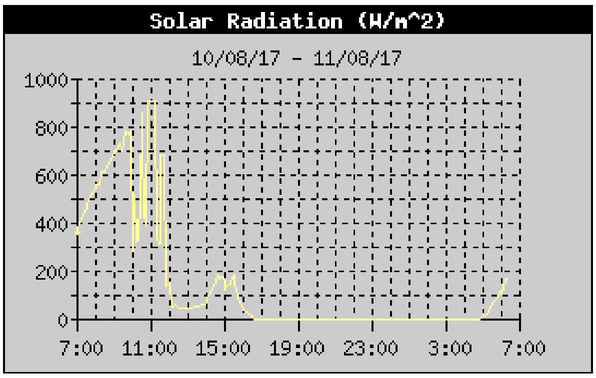


Fig. 10. Graphic display of solar radiation for 24 hours.

Before being installed on the mobile device, the file size of this application is 684 KB. After the application is installed, it takes 1.93 MB of the storage on the Android device. The size increase is mainly caused by the allocation of cache memory. In addition, the application uses an approximately 58.7 Mb of RAM.

V. Conclusions

This paper presents the development of real-time weather monitoring system based on a mobile application using automatic weather station. The system is able to display the information of the weather parameters via a mobile application installed on the Android device. The application shows the real-time weather information through the chart. The measurement parameter includes temperature, humidity, air pressure, rainfall and solar radiation. The user can monitor the weather continuously through this mobile application and can leverage the information to their needs. Through the development of this monitoring system, the cost of the service can be reduced, and the UI including the required weather parameters can be adjusted. The use of 58.7 Mb RAM allows applications to run and installed on Android devices that have minimal RAM of 512 Mb.

Acknowledgment

This research was supported by the thematic program 2017 through Technical Implementation Unit for Instrumentation Development, Deputy for Scientific Services, Indonesian Institute of Sciences Indonesia.

References

[1] M. Ashraf, J. C. Loftis, and K.G. Hubbard, "Application of geostatistics to evaluate partial weather station networks," Agricultural and Forest Meteorology, vol. 84, pp. 255-271, 1997.

[2] G. Urban, "An attemp to assess the results of air temperature measurements from automatic weather stations in comparison to glass thermometer measurements in the context of weather types," Bulletin of Geography Series, No. 9, pp. 67-79, 2015.

[3] S. AL-Yahyai , Y. Charabi , A. Gastli, and S. Al-Alawi, "Assessment of wind energy potential locations in Oman using data from existing weather stations," Renewable and Sustainable Energy Reviews, vol. 14, pp. 1428–1436, 2010.

[4] S. Abbatea, M. Avvenutia, L. Carturan, and D. Cesarinia, "Deploying a communicating Automatic Weather Station on an Alpine glacier," The 3rd international Workshop on Sensor Networks for Intelligence Gathering and Monitoring (SNIGM), Procedia Computer Science, vol. 19, pp. 1190 – 1195, 2013.

[5] R. Kus´mierek-Tomaszewska, J. Zarski, and S. Dudek, "Meteorological automated weather station data application for plant water requirements estimation," Computers and Electronics in Agriculture, vol. 88, pp. 44–51, 2012.

[6] World Meteorological Organization, 1992a: International Meteorological Vocabulary, Second edition., WMO-No. 182, Geneva.

[7] World Meteorological Organization, 1989a: Guide on the Global Observing System, WMO-No. 88, Geneva.

[8] X. Guo and Y. Song, "Design of automatic weather station based on GSM module," International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE), vol. 5, August24-26, pp. 80-82

[9] World Meteorological Organization, Guide to Meteorological Instruments and Methode of Observation,Geneva, 2008.