

PAPER • OPEN ACCESS

Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications

To cite this article: Dwi Ely Kurniawan *et al* 2019 *J. Phys.: Conf. Ser.* **1351** 012006

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications

Dwi Ely Kurniawan^{1*}, Mohd Iqbal², John Friadi³, Rohmat Indra Borman⁴, Rio Rinaldi¹

¹Department of Informatics Engineering, Politeknik Negeri Batam, Indonesia

²Department of Informatics Engineering, STIKOM Muhammadiyah Batam, Indonesia

³Department of Information System, Batam University, Indonesia

⁴Faculty of Engineering & Computer Science, Universitas Teknokrat Indonesia, Indonesia

*dwialikhs@polibatam.ac.id

Abstract. The server room is a room that stores data, there is company information (Data Center). Hot server room conditions can cause performance on the device and network to decrease. Therefore an admin must keep the server space stable so that the server and network performance is maintained. The temperature monitoring system can be used to monitor temperature and humidity in the server room. In addition, it provides a response to reduce the temperature if there is a temperature increase in the server room. This study uses the Internet of Thing (IoT), which is a device used by the Raspberry Pi and Wemos DHT Shield wireless sensor as a device that is able to read the temperature and humidity conditions of the room. Temperature and humidity log data is stored in the MySQL database, then displayed in a chart diagram in real time. IoT response based on predetermined temperature standards by giving notifications to users via Whatsapp Application on mobile devices.

1. Introduction

The server room is a very sensitive space because there is a server computer and data center infrastructure in a company [1]. Therefore the condition of the server room must always be protected from temperature and humidity. The server room certainly has an Air Conditioner (AC) which has an important role to maintain room temperature. If there is damage to the air conditioner, it can increase in room temperature. AC damage can also lead to increased humidity in the server room, for example, the Freon gas leak. This AC damage cannot be detected because the AC does not have a system to warn of damage resulting in a decrease in room temperature. Therefore there is a need for additional tools to provide warning information regarding server room temperature.

Another case are when the main power source goes out in the server room, the main power source should be diverted to the UPS [2]. When the main electricity source of the *Perusahaan Listrik Negara* (PLN) goes out, the AC is turned off so that the power consumption of the Uninterruptible Power Supply (UPS) battery is enough for the server computer to survive until the main electricity source of PLN is restarted. In connection with this, a server room monitoring system is needed in real time [3]. By knowing the temperature information through the raspberry pi microcomputer device in the server room, it can maintain the standard temperature and humidity of a server room. When the server room admin is not in the vicinity of the server room, a system is needed that can provide notifications to the



admin about the real-time conditions of server room temperature. Admin can control such as shutdown the server computer [4] when the room temperature is hot and turn on the server again when the room temperature has returned to normal.

Some related studies such as designing a room temperature controller device based on the Atmega8535 microcontroller [5], monitoring temperature and humidity in a web-based pressure and volume calibration laboratory in real time [6], implementing a server room temperature monitoring system based on Internet of Things (IoT) [7] using the Message Queue Telemetry Transport (MQTT) communication protocol [8] [9] [10] [11], the design of temperature devices measuring temperature devices using LM35 sensors based on SMS gateways [12], focus on giving notifications via mobile device [13] [14] [15], design of monitor systems and control of lab space on Arduino [16] [17]. Some of these studies do not have system integration in one problem, for example, focus on giving notifications via SMS but there is no control, there is control but there is no real-time monitoring. Therefore this study tries to integrate from the system with web-based monitoring, notifications on Whatsapp, controlling of on / off server computers and viewing log history the temperature of the server room.

2. Literature Review

2.1. Raspberry pi

Raspberry Pi is a small computer device the size of a credit card. Like ordinary computers on this device, we can install the Linux operating system. Raspy can be used as a web server, router, media center, and others. The input-output of the raspberry is pretty much like a USB port, LAN, HDMI, and 40 Pin GPIO headers that can be optimized as needed. Raspberry runs using the Apache, MySQL, web server, and python library Yowsup services. The raspberry specifications used are equipped with an internal wifi device that functions to connect and send data between sensors with raspberry. Raspberry saves enough electricity, only requires 2.5A and 5V to turn it on. Data storage on raspberry does not use disks like hard drives on computers. This device uses Micro SD card which is also a data storage area on a smartphone.



Figure 1. Raspberry Pi II

2.2. Sensor Wemos D1 Mini ESP8266 with DHT Shield

This research uses a mini Wemos D1 sensor. Wemos D1 mini is a low-cost microcontroller device. The mini D1 modem is equipped with a Wifi module so that it can interact with the TCP / IP protocol. Wemos D1 is equipped with a DHT sensor which is a sensor to read the temperature and humidity of the room. The DHT sensor is analog, requiring raspberry to process the data. The DHT sensor has a temperature data accuracy of around 5% from a temperature of 0°-50°, while for measuring humidity, the sensor has an accuracy of about ± 2 °C from humidity between 20% - 80% relative humidity. The reason for choosing this Wemos sensor is because it is cost-effective.

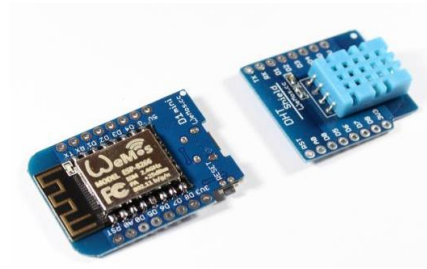


Figure 2. Sensor Wemos D1 Mini ESP8266 with DHT Shield

2.3. Temperature and Humidity Standards of the Server Room

The server room must have sufficient temperature control to maintain the specified operational limits for the hardware devices in the server room. The server room should have an optimal air conditioning system that is sufficient to maintain the temperature and humidity of the server room. Room temperature standards are also reviewed based on standards from ANSI / TIA-942-A (Telecommunications Infrastructure Standard for Data Centers) [18] [19] with the following conditions:

- Temperature: 18-27 ° C (64-81 ° F) dry bulb temperature.
- Maximum relative humidity: 60%
- Maximum dew point: 15 ° C (59 ° F)
- The maximum temperature change rate: 5 ° C (9 ° F) per hour

Apart from the ANSI / TIA-942-A standard [19], the temperature and humidity limit values are also taken based on the Standard Operating Procedure (SOP) to maintain server security and resilience. The temperature in the server room must also be maintained in accordance with the health standards of the server room. The following formula for obtaining temperature and humidity data so that it can provide a warning automatically:

- if: temp \Rightarrow 30 ° The system will provide notifications (1)
- if: humid \Rightarrow 60 ° The system will provide notifications (2)

Where temp is the value of temperature and humid is the value of humidity. The value of temp and humid is obtained in real time through a DHT sensor to be sent and processed by raspberry py. The notification system will use the Yowsup library. Library Yowsup is a python library on a raspberry that works so that the system can send notifications to smartphones via the Whatsapp application.

3. Methods

Overall system design uses wireless technology to avoid the wiring in server space. The raspberry device uses a DHT sensor that can read the temperature and humidity of the room. After reading the raspberry and processing the temperature and humidity, raspberry pi will activate the web server service, so that it can display the website pages so that the website can be accessed via a local wireless network (a private network) or via the internet (a public network).

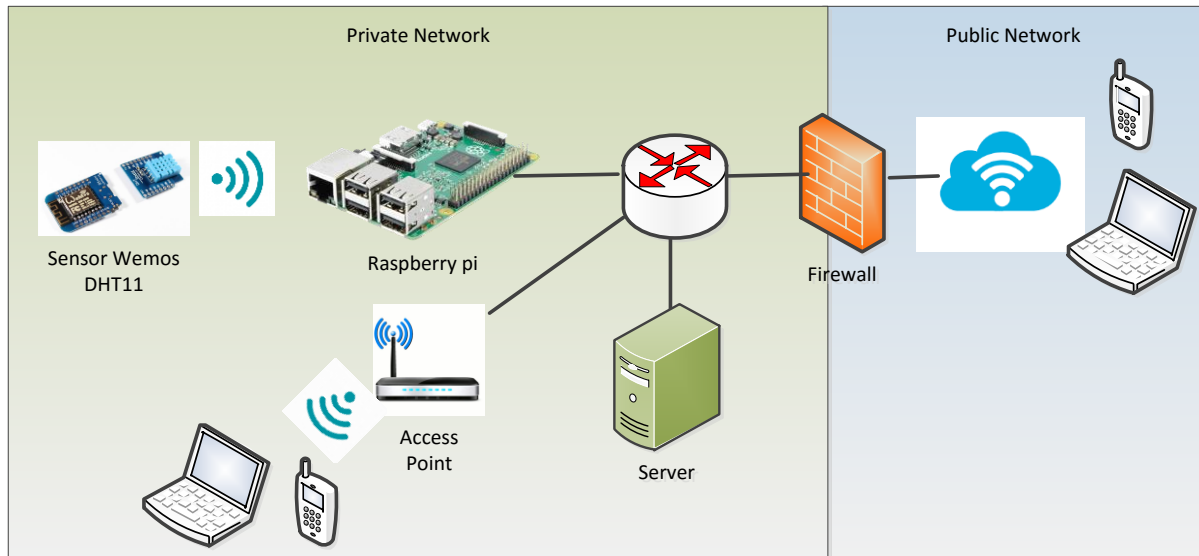


Figure 3. Design of the smart monitoring temperature and humidity on room server

The first step is to configure the wireless network so that ESP8266 is connected to raspberry. Configuration can be done on the local network first on raspberry by doing command `ifconfig eth0`. Next, configure the route `-n gateway` on raspberry so that it can be accessed via the internet (public network). Next, give the SSID name on the WiFi Raspberry according to the SSID that was previously set on the ESP8266 Wemos uploaded script.

The next step to install the program on Raspberry is recommended to first update the system on Raspberry. Configuration can be done with the command `sudo apt-get update`. Then continue installing the Apache web server by prompting on raspberry `apt-get install apache2 -y`. Next, configure the web server so that raspberry hosts can be accessed through the local network. To ensure the web server is properly configured, the host address on the raspberry can be accessed through a web browser. After the web server is running, then install the MySQL-server service by doing the command `apt-get install mysql-server`. To facilitate the management of data in MySQL, it requires phpmyadmin. Perform the command `apt-get install phpmyadmin` so that the data on MySQL can be accessed via the PHPMyAdmin web GUI (Graphical User Interface).

After ESP8266 is connected with raspberry, it continues to PHP configuration stage so that the data uploaded from ESP8266 can be processed on raspberry and the data will be entered into MySQL database. By using the PHP post ESP8266 script can send temperature and humidity data directly into MySQL. MySQL can directly record the time when ESP8266 sends data using the current timestamp system. After the data from ESP8266 was successfully entered into the database then created the main website page to visualize the data into graphs and numbers. Graph visualization will use the Highcharts library so first preparing a data server rather Highcharts can display it. The website layout also uses a bootstrap library so that layouts stay organized when accessed via a mobile device or smartphone. To display temperature data records stored in MySQL database will use the Datatables script library. Datatables make it easy to find data and make it more dynamic [20].

The next step is to install the Yowsup library on raspberry. Yowsup uses a base python script.

1) Install the package needed by Yowsup by typing this in the terminal.

```
sudo apt-get install python-dateutil
sudo apt-get install python-setuptools
sudo apt-get install python-dev
sudo apt-get install libevent-dev
sudo apt-get install ncurses-dev
```

- 2) Then go to the directory where Yowsup will be cloned.
`cd / home / pi /`
`git clone https://github.com/tgalal/yowsup.git`
- 3) Then go to the directory where Yowsup is installed and install.
`cd yowsup`
`sudo python setup.py install`
- 4) After installation is complete, you can proceed with registering the cellular number that will be used by Yowsup.
`yowsup-cli registration -questcode sms -phone 628xxxxxxxxx -cc 62 -mcc 510 -mnc`
- 5) After receiving the registration code message from whatsapp then continue registration to the verification stage.
`yowsup-cli registration -register xxx-xxx -phone 628xxxxxxxxx -cc 62`
- 6) After entering the registration code correctly, the data will appear from Whatsapp account created, such as cellular numbers and passwords, the data into a text file to be used as a login from Yowsup, with the following format:
`phone = 628XXXXXXXXXX`
`password = AzQ1XXXX`
- 7) After saving the credentials, then we test using the sample from Yowsup.
`yowsup-cli demos -config /home/pi/yowsup/whatsapp_config.txt -end 628XXXXXX`
`"TEST123"`
- 8) If the message is successfully received, it means a successful installation. The first Yowsup configuration is to register a telephone number that will be used as a number that will send data from raspberry. Then configure the command that will be added to the Yowsup. Make sure the script that is added to Yowsup runs by running it on the Whatsapp application on the smartphone.

How Yowsup works are like making a virtual mobile device on raspberry by using a python script. So this Yowsup program processes data from chat sent from Whatsapp and triggers the command that has been given.

4. Testing and Result

After raspberry boots up, it ensures that the network configuration on the raspberry doesn't change and the wireless host service also runs. Connect ESP8266 from Micro USB to USB power with 5-volt power. The device will be connected automatically with a previously configured Wifi network. The LED light on ESP8266 will turn on, blink blue when sending data to a raspberry pi. To find out the results of sensor readings can be done by accessing the MySQL database on raspberry. Because the ESP8266 module immediately posts data to MySQL via the add.php script.

```
mysql> use SMRT;
Database changed
mysql> show tables;
+-----+
| Tables_in_SMRT |
+-----+
| cmd             |
| tempLog         |
+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM `tempLog` ORDER BY `timeStamp` DESC limit 1;
+-----+-----+-----+-----+-----+
| unixtime | timeStamp | temperature | humidity | sensor |
+-----+-----+-----+-----+-----+
| 1487896821 | 2017-02-24 07:40:21 | 18 | 44 | 2 |
+-----+-----+-----+-----+-----+
1 row in set (8.53 sec)
```

Figure 4. TempLog table on the MySQL Database

ESP8266 as a module that transmits analog data from the sensor to raspberry. Data sent by ESP8266 is received through the get.php file. The data is recorded into MySQL with the database name, SMRT, and table, namely tempLog.

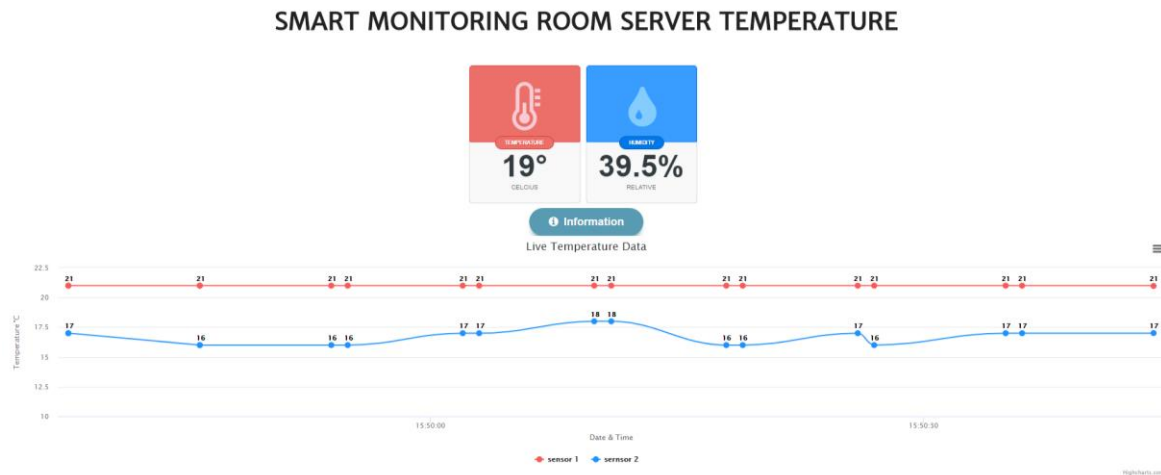


Figure 5. Website page from monitoring the temperature and humidity of the server room

Data on tempLog is in the form of time & date, sensor nodes of temperature and humidity. Time & date on timeStamp is converted first to unix-time to make it easier to display time series data used in Javascript. Temperature and humidity data will be displayed in graphical form in real time on web pages using the Highcharts library. Access the web host with the local or public URL address to display unix-time data, temperature sensor 1, temperature sensor 2. The data is taken through the PHP page which is server-data.php which contains data (unix-time, temperature, humidity, sensor) with the variable using the standard JSON format. Figure 4 shows the website page from monitoring the temperature and humidity of the server room. The red box displays a temperature of 22 °C and a blue box displays humidity of 32.5%. The temperature and humidity are obtained from the average value between sensor 1 and sensor 2. Under the box there is a graph of two sensors displayed in realtime, the red graph is sensor 1 and the blue graph is sensor 2.

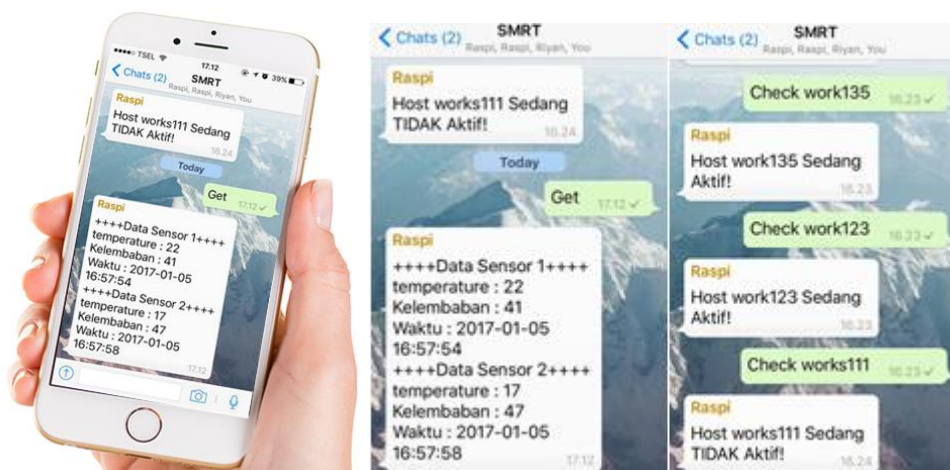


Figure 6. Whatsapp notification of smart monitoring temperature and humidity on smartphone

Monitoring the system uses notifications to users via Whatsapp by creating a trigger or counter on MySQL based on a predetermined value. Notifications use the yowsup library from Github. This

yowsup uses a python script with a programming layer. To activate it requires a telephone number to confirm a new account from Whatsapp. After configuring yowsup it can then define the word that will be used to request data from the sensor sent to the web server on raspberry. For example, by sending the Get command, yowsup will request data on the web server and process it and then send information to the Whatsapp group. Get command to get temperature and humidity information. Check (Host) command to check. The shutdown (Host) command to turn off. The wake (Host) command to turn it on.

5. Conclusion

The results of system design have been able to send temperature and relative humidity information in real time to a database that is on raspberry. Data is sent via a wireless network between ESP8266 and raspberry. The results will be displayed on the PHP website page which hosts raspberry pi. The data of temperature and humidity is relatively stored in the MySQL database and can be displayed as a report. Data consists of time, unix-time, temperature, humidity, and sensor type. The monitoring system in the form of notification to the user when there is a temperature rise above the specified value. Notifications are sent via the Whatsapp application on the Smartphone. The system can perform a command to check the host server and the command to turn off the server directly through the Whatsapp application with the number that has been registered on the yowsup.

Acknowledgments

Authors thanks to Politeknik Negeri Batam P3M support to the research.

References

- [1] Pakbaznia E and Pedram M 2009 Minimizing Data Center Cooling and Server Power Costs *Proceedings of the 2009 ACM/IEEE International Symposium on Low Power Electronics and Design ISLPED '09* (New York, NY, USA: ACM) 145–50
- [2] Nelmiawati N, Kushardianto N, Tohari A, Hasibuan Y and Kurniawan D 2018 Rancang Bangun Lab Komputer Virtual Berbasis Cloud Computing Menggunakan Openstack Pada Jaringan Terpusat *J. Appl. INFORMATICS Comput.* **2**
- [3] Sethi P and Sarangi S R 2017 Internet of things: architectures, protocols, and applications *J. Electr. Comput. Eng.* **2017**
- [4] Kurniawan D E, Nashrullah M, Kurniasih N, Achmad Daengs G S and Kurniawan C 2018 Performance analysis virtual server VMware Vsphere 5.5 with physical enterprise server *IOP Conf. Ser. Mater. Sci. Eng.* **420** 012107
- [5] Triyanto R 2014 Perancangan Alat Pengendali Temperatur Ruangan Server Berbasis Mikrokontroler Atmega8535 *PIKSEL (Penelitian Ilmu Komput. Sist. Embed. dan Logic)* **2** 82–9
- [6] Jumaila S I and Maulida S 2018 Pemantauan Suhu dan Kelembaban di Laboratorium Kalibrasi Tekanan dan Volume Berbasis Web Secara Real Time *J. Otomasi Kontrol dan Instrumentasi* **9** 9
- [7] Shah J and Mishra B 2016 IoT enabled environmental monitoring system for smart cities 2016 *International Conference on Internet of Things and Applications (IOTA)* 383–8
- [8] Kang D H, Park M S, Kim H S, Kim D, Kim S H, Son H J and Lee S G 2017 Room temperature control and fire alarm/suppression IoT service using MQTT on AWS 2017 *International Conference on Platform Technology and Service (PlatCon)* 1–5
- [9] Periyaldi P, Bramanto A and Wajiansyah A 2018 Implementasi Sistem Monitoring Suhu Ruang Server Satnetcom Berbasis Internet Of Things (IOT) Menggunakan Protokol Komunikasi Message Queue Telemetry Transport (MQTT) *JTT (Jurnal Teknol. Terpadu)* **6** 23–9
- [10] Kurniawan D E 2017 Push Notification System Pada Prototype Kendali Listrik Rumah *CESS (Journal Comput. Eng. Sci. Syst.)* **2** 89–92
- [11] Rochman H A, Pramananda R and Nurwasito H 2017 Sistem Kendali Berbasis Mikrokontroler

- Menggunakan Protokol MQTT pada Smarthome *J. Pengemb. Teknol. Inf. dan Ilmu Komput.* **1** 445–55
- [12] Suherman S, Andriyanto I and Dwiyatno S 2015 Rancang Bangun Alat Ukur Temperatur Suhu Perangkat Server Menggunakan Sensor LM35 Bebasis SMS Gateway *PROSISKO J. Pengemb. Ris. dan Obs. Sist. Komput.* **2**
- [13] Najmurrokhman A, Rahim R, Kusnandar K, Wibowo B, Kurniawan D E and Sumaryat Y 2018 Design and Implementation of A Low Cost Heartbeat Meter using Vibration Sensor, ATmega 16 Microcontroller, and Android Based Smartphone *2018 International Conference on Applied Engineering (ICAE)* 1–5
- [14] Kurniawan D E, Fatulloh A and Kushardianto N C 2018 Ship Ticketing System Using Near Field Communication (NFC) and SMS Gateway for Batam Island Transportation (EAI)
- [15] Kurniawan D E, Janah N Z, Wibowo A, Mufida M K and Prasetyawan P 2018 C2C marketplace model in fishery product trading application using SMS gateway *MATEC Web Conf.* **197**
- [16] Nugraha N 2017 Rancang Bangun Sistem Monitor Dan Kendali Ruang Laboratorium Berbasis Arduino Ethernet Shield *Buffer Inform.* **2**
- [17] Jindarat S and Wuttidittachotti P 2015 Smart farm monitoring using Raspberry Pi and Arduino *2015 International Conference on Computer, Communications, and Control Technology (I4CT)* 284–8
- [18] DiMinico C 2010 TIA-942 presentation 36
- [19] ANSI/TIA 2012 *Telecommunications Infrastructure Standard for Data Centers*
- [20] Mufida M K, Santiputri M, Janah N Z, Kurniawan D E and Idris M 2018 Lecturer Workload Optimization Applying Interactive Visualization *IOP Conference Series: Materials Science and Engineering* **407** 12113