InComTech: Jurnal Telekomunikasi dan Komputer
vol.13, no.2, Agustus 2023, 96-112
http://publikasi.mercubuana.ac.id/index.php/Incomtech
P-ISSN: 2085-4811 E-ISSN: 2579-6089

Monitoring and Control System of Server Devices Based on IoT Using Sugeno Fuzzy

Achmad Syafriyal[1], Ilham Firman Ashari[1*], Eko Dwi Nugroho[1]

[1]*Teknik Informatika, Institut Teknologi Sumatera*
Jl. Terusan Ryacudu. Lampung Selatan
*irman.ashari@if.itera.ac.id

Abstract **:**
Fire is a disaster or disaster caused by a fire, it can happen anywhere and anytime. Server is a very important asset for owners of companies or institutions that implement information technology. The importance of a server in an agency is because the server has applications and databases that store all important and valuable information for the company. A server must have security standards that protect the work of the devices in it and one of them is the temperature on the server device must always be in good condition. For this reason, in this study, a system that can perform supervision and control related to temperature, smoke and fire levels and can send notifications of the state of the server device is safe or not using the fuzzy logic of the Sugeno method. From the test results, the system can determine various server device conditions with accuracy reaching 100%. In fuzzy testing, the percentage of success is 100%. This shows that the results of testing the fuzzy method on the system are in accordance with the design that has been made. The system can determine the condition of the server device (Safe, Standby, Danger).

Keywords:
Fire;
Server;
IoT;
Fuzzy;

Riwayat Artikel:
Received September 15, 2022
Revised July 04, 2023
Accepted July 11, 2023
Published August 15, 2023

DOI:
10.22441/incomtech.v13i3.17132

1. INTRODUCTION

Fire is a disaster or disaster caused by a fire that can occur anywhere and anytime. Fires can cause property loss, injury and even death [1]. According to the Big Indonesian Dictionary, fire can be defined as “the incident of burning something (a dwelling, forest, building, etc.). Fires occur can be caused by three factors that are elements of fire, namely oxygen, heat, and fuel (solid, liquid, and gas). The type and amount of combustible compounds can cause a severe explosion [1].

Internet of Things (IoT) is a concept that aims to expand the benefits of continuously connected internet connectivity that allows us to connect machines, tools, and other physical objects with a network of sensors and actuators to acquire data and manage their own performance. , thus enabling machines to collaborate and even act on the newly acquired information [2][3][4].

Server devices are an important asset for owners of companies or institutions that apply information technology as a support in their daily activities [5][6]. The importance of a server in an agency is because on the server there are applications and databases that store all important and valuable information for the company or institution concerned, therefore the server device must always be in good condition [7][8][9]. A server device must have security standards that protect the work of the devices in it and one of them is the temperature on the server device must always be in good condition [10].

The lack of temperature monitoring from the server admin is one of the factors causing damage, fires on existing devices. It would be very impossible if this monitoring and temperature monitoring only relied on human capabilities, moreover the admin on duty would not be able to stand-by or stay in the server room for 24 hours just to monitor the temperature of the server device [11].

Based on these problems, the author will create a tool that can monitor and control temperature and fire on Android-based server devices that are integrated with Android applications. The android application is used to supervise and notify the server admin when certain conditions occur such as temperature increases, smoke levels and detecting a fire on the server device. This is because Android is an operating system that is widely used and practical in its use. This monitoring and control system for temperature and fire server devices based on Android can make it easier for admins who only have very efficient time by seeing the results of the temperature display, the smoke level of the server device from the Android application, so admins can monitor from anywhere.

Fuzzy logic is one form of soft computing. The fuzzy system is a structured and dynamic numerical estimator. This system can develop intelligence systems in an uncertain environment. There are several methods to represent fuzzy logic results, namely the Tsukamoto, Sugeno, and Mamdani [12][13]. In the Tsukamoto method, each result of the rules that have been made, must be represented by a fuzzy set with a monotonic membership function. As a result, the output of the inference results from each rule is given in a crisp, but not as clear as with the other two methods. This method does not follow a strict composition rule where the output is always crisp even though the input is fuzzy.

In the Sugeno method, the output (consequently) is not in the form of a fuzzy set but in the form of a constant or a linear equation. The Sugeno method requires estimation of the parameters contained in the data. The advantage of this method is that it is very easy to use for various stability analysis techniques. The mamdani method is also known as the MAX-MIN method. The resulting output is in the form of fuzzy numbers, so a certain crisp value must be determined as output. The weakness of the mamdani method is that it cannot distinguish specific information from the input space and requires reference rules that cover all input spaces [14].

In this study, the author uses the Sugeno fuzzy method to produce the output of a server room security condition status. The use of the Sugeno fuzzy method is because this method is very flexible in making decisions and has tolerance for existing data [15]. The Sugeno fuzzy method in this study is used to process the measurement data from the DHT11 sensor in the form of temperature, the MQ-2 sensor in the form of smoke, and the Flame sensor in the form of fire. The results of defuzzification on temperature, smoke, and fire are parameters used for security conditions in the server room.

In 2020, Pingki, Huda Ubaya, Kemahyanto Exhaudi conducted research entitled "Telegram Web Based Server Room Temperature and Humidity Monitoring". This research produces a system that monitors the temperature and humidity of the server room [16]. In 2020, Budi Indra Gunawan, Unan Yusmaniar Oktiawati doing research, their research produces a prototype of a server room temperature monitoring and controlling system consisting of two nodes, namely the rack node and the controller node. The rack node is used to read the temperature and humidity on the server rack [12].

In this study, a system will be built that can help the server admin at the Engineering Laboratory 1 at the Sumatra Institute of Technology in monitoring and controlling the temperature and fire of server devices. The system can provide temperature information, smoke levels, detect when a fire occurs, the system can control the temperature using a fan to cool server devices so that no damage occurs when the room temperature is abnormal, the system can turn off the current on server devices when the system detects a fire , sends notifications when the Mq-2 sensor detects smoke, the DS18B20 sensor detects when there is an increase in the temperature of the server device, the flame sensor detects a fire, the values of each sensor can be seen on the android admin server that has installed the application.

2. METHOD

2.1. Research Flow

This research design section will provide an overview of the stages carried out in the development of this system. Figure 3.1 is a flow chart in this study.

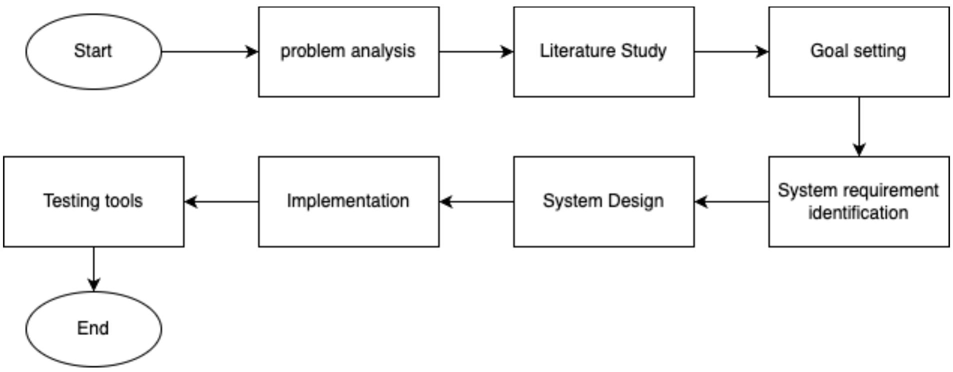

Figure 1. Research Flow

In Figure 1 is a research flow starting from problem analysis, literature study, goal setting, system requirement identification, system design, implementation, tool testing. The following is an explanation of the stages of the research:

1. Problem Analysis

The problem analysis stage is the stage to find and identify problems that occur, especially related to damage, server room fires. At this stage the authors also look for supporting data to deepen the information and strengthen the research objectives.

2. Literature Study
   In this literature study stage, the writer collects and studies every reference related to this research. Literature study was conducted to collect theories to be poured which were obtained from research that had been done previously.
3. Goal Setting
   At this goal setting stage, the author formulates a solution to the problem obtained to help solve the problem that occurs. The purpose of this research will be the target to be achieved by the author during the research.
4. Identify System Requirement
   At the stage of identifying the needs of this system, the author determines the tools and applications that will be used in making the system. Identification of system requirements is determined based on needs that are in accordance with the problem solutions that have been prepared in the previous stage.
5. System Design
   At this design stage, the design of the tools and applications used is carried out and at this stage the workflow of the entire system is arranged.
6. Implementation
   At this implementation stage, the author begins making the system that has been designed until the system is successfully completed and can run according to the required functionality.
7. Testing tool
   At this testing stage, the author tests the system that has been made at the implementation stage, this stage aims to see the performance of the system as a whole and repairs will be made if there are errors or tools that do not work properly.

2.2.Hardware Design

2.2.1. Hardware Design Inside View

The internal monitoring hardware design that will be built can be seen in Figure 2.

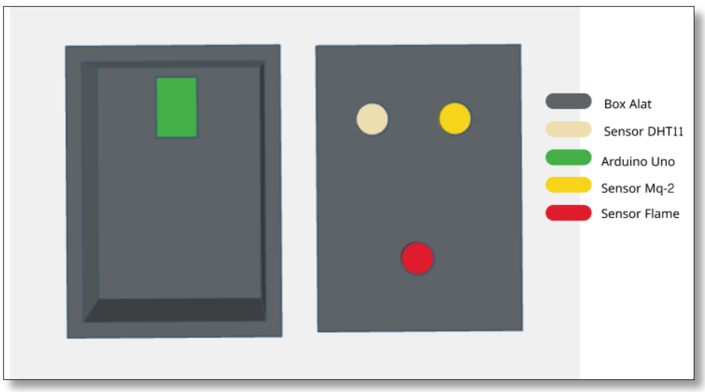

Figure 2. Hardware Design Inside View

In Figure 2 is the design of this temperature and fire monitoring and control tool using a box installed with Arduino Uno, DHT11 sensor, flame sensor, MQ-2 sensor, wifi module.

2.2.2. Hardware Design Side View

The side view monitoring hardware design to be built can be seen in Figure 3.

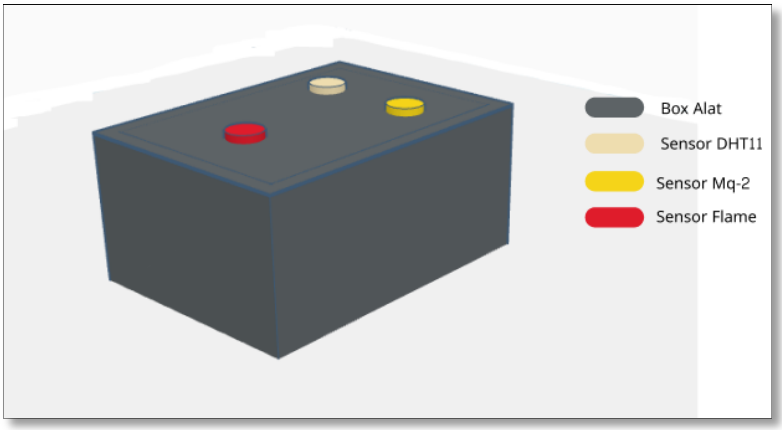

Figure 3. Hardware Design Side View

In Figure 3 is the design of a top view monitoring and early detection tool for fire, the design of this tool uses a box attached to a DHT11, Mq-2, Flame sensor.

2.2.3. Tool Installation Implementation Design Side View

The design implementation of the side view of the tool installation can be seen in Figure 4.

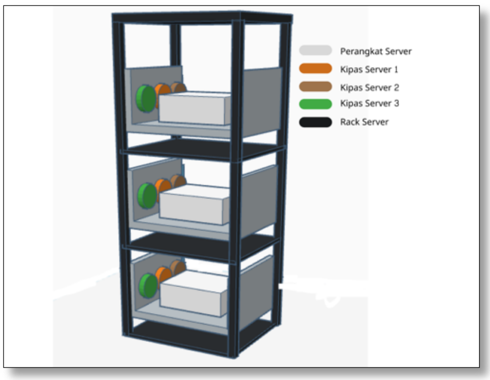

Figure 4. Implementation Design of Tool Installation Side View

In Figure 4 is a side view of the implementation design of the tool installation, this implementation design uses a rack server, server equipment, server fan 1, server fan 2, server fan 3.

2.3.Software Design

Server 1 page display design can be seen in Figure 5.

Figure 5. Server Page Display Design 1

2.4.Fuzzy Logic Design

There are 4 stages that will be used in fuzzy logic to determine whether the server device is safe or not, including:

2.4.1. *Fuzzification Step*

At the fuzzification stage, it functions to translate input data from a sensor into a fuzzy set by determining the value of the degree of membership. There are temperature data, smoke level [16], fire [17]. The stages of these changes can be seen in Table 1.

Table 1. Fuzzification Step

No.	Input	
	Parameter Definition and Value	**Fuzzy Variabel**
1.	Room temperature	Temperature (S)
	0 – 10° C	Cool (D)
	10 - 25° C	Normal (N)
	25 - 40° C	Hot (P)
2.	Room smoke level	Smoke level (KA)
	0 – 25 ppm	Light (TP)
	25 – 50 ppm	Medium (S)
	50 – 100 ppm	Thick (T)
3.	Fire in the room	Fire (AP)
	0	No (Ta)
	1	Yes (A)

2.4.2. Stages of Formation of Fuzzy Basis Rule

At the stage of forming the fuzzy base rule, namely for the formation of the rules used as fuzzy data management, there are input data, namely temperature, smoke, and fire levels in the room. There are 18 rules in the fuzzy logic basis, including the following:

1. The temperature is cold, the smoke level is thin and there is no fire.
2. The temperature is cold, the smoke level is thin and there is a fire.
3. Cold temperatures, moderate smoke levels and no fire.
4. The temperature is cold, the smoke level is moderate and there is a fire.
5. Cold temperatures, thick smoke levels and no fire.
6. The temperature is cold, the smoke level is thick and there is a fire.
7. Normal temperature, thin smoke content and no fire.
8. The temperature is normal, the smoke level is thin and there is a fire.
9. Normal temperature, moderate smoke level and no fire.
10. Normal temperature, moderate smoke level and no fire.
11. Normal temperature, thick smoke and no fire.
12. Normal temperature, thick smoke and fire.
13. The temperature is hot, the smoke level is thin and there is no fire.
14. The temperature is hot, the smoke level is thin and there is a fire.
15. Hot temperature, moderate smoke level and no fire.
16. The temperature is hot, the smoke level is moderate and there is a fire.
17. Hot temperature, thick smoke and no fire.
18. Hot temperature, thick smoke and fire.

2.4.3. Inference Step

At the inference stage, there is a fuzzy rule/rule to produce the output of each rule. At this time, there are 18 fuzzy logic outputs with a value range of 0 to 1. The output on this system is the secure status of the server room, as follows:

1. The temperature is cold, the smoke level is thin and there is no fire, so it's safe
2. The temperature is cold, the smoke level is thin and there is a fire, so it's dangerous
3. The temperature is cold, the smoke level is moderate and there is no fire, so be on standby
4. The temperature is cold, the smoke level is moderate and there is a fire, so it's dangerous
5. Cold temperatures, thick smoke levels and no fire are dangerous
6. The temperature is cold, the smoke level is thick and there is a fire, so it's dangerous
7. Normal temperature, thin smoke level and no fire, so be on standby
8. Normal temperature, thin smoke level and there is a fire then Danger
9. Normal temperature, moderate smoke level and no fire then Standby

10. Normal temperature, moderate smoke levels and there is a fire then Danger
11. Normal temperature, thick smoke and no fire, so be on standby
12. Normal temperature, thick smoke levels and there is a fire then Danger
13. Hot temperatures, thin smoke levels and no fire are dangerous
14. The temperature is hot, the smoke level is thin and there is a fire, so it's dangerous
15. Hot temperatures, moderate levels of smoke and no fire are dangerous
16. The temperature is hot, the smoke level is moderate and there is a fire, so it's dangerous
17. Hot temperatures, thick smoke levels and no fire are dangerous
18. Hot temperatures, thick smoke levels and a fire are dangerous

2.4.4. Defuzzification Step

The defuzzification stage is the stage of calculating the crisp output. At this stage the maximum value of the membership degree will be taken and displayed to determine the condition of the server device is safe or not.

3. RESULT AND DISCUSSION (55%)

3.1. Hardware Implementation

The following is a hardware implementation design for the monitoring and control system for server devices, which can be seen in Figure 6.

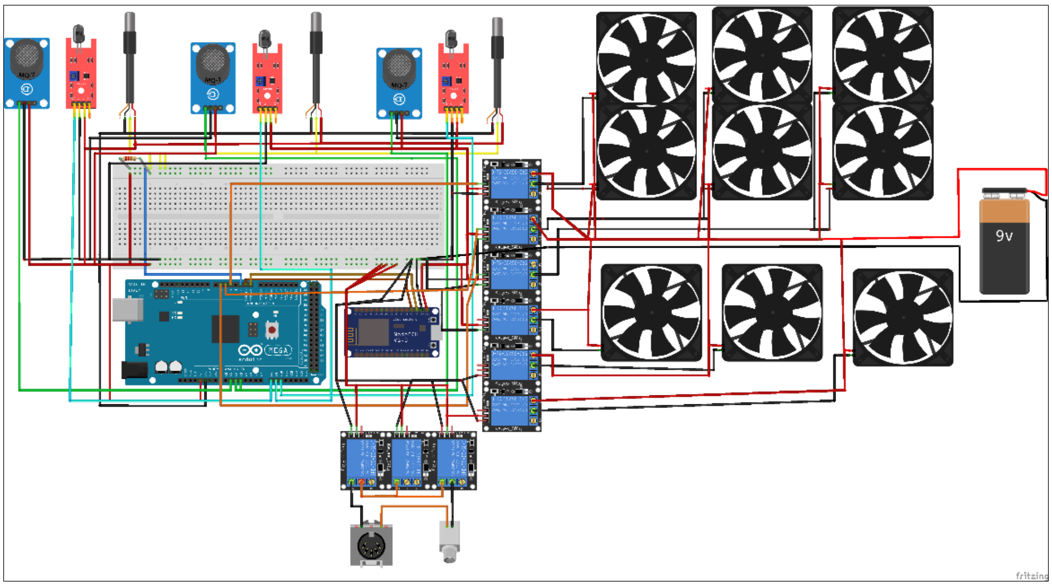

Figure 6. Hardware Design

The monitoring and control system for temperature and fire server devices utilizes the Arduino Mega which has 54 input/output pins connected to the PCB board with jumper cables. There are nine main sensors that will be used to read server room conditions, namely the DS18B20 sensor to detect temperature, the MQ2 sensor to detect smoke, and the flame sensor to detect fire. There are nine fans and nine relays in this study. Each server device will be placed with three sensors, three fans, two of which will be used to cool the server equipment and one to blow smoke. For the temperature sensor when the reading value from the sensor reaches 30°C then the fan to cool the server device will turn on, for the smoke sensor when the reading value from the sensor reaches 300 ppm then the fan for the exhaust smoke will light up, for the fire sensor when the sensor reading value changes to Yes then arduino will instruct the relay to turn off the electric current. The results of the implementation of the entire hardware monitoring and control system for server devices can be seen in Figure 7.

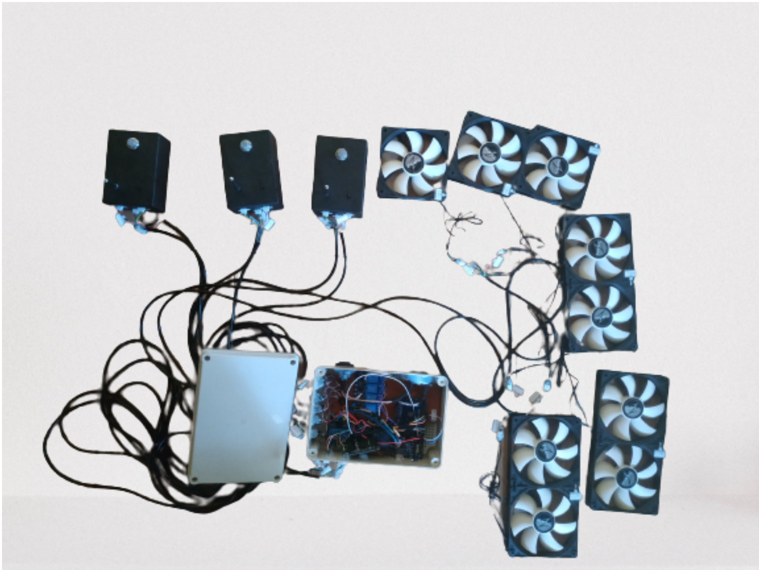

Figure 7. Hardware Implementation

3.2.Relational Database Implementation

Figure 8 is the Variable Initialization source code which is used to define functions for connecting firebase using the "Firebase_ESP_Client.h" library to make it easier to send data between the data received by the sensor.

```
void terimaData(){
  StaticJsonBuffer<1000> jsonBuffer;
  JsonObject& dataMega = jsonBuffer.parseObject(Serial);

  if (dataMega == JsonObject::invalid()) {
    jsonBuffer.clear();
    cekData = "Tidak Ada";
    return;
  }
  suhu1 = dataMega["suhuServer1"];
  asap1 = dataMega["asapServer1"];
  api1 =  dataMega["apiServer1"].as<String>();
  keadaan1 = dataMega["keadaanServer1"].as<String>();

  suhu2 = dataMega["suhuServer2"];
  asap2 = dataMega["asapServer2"];
  api2 =  dataMega["apiServer2"].as<String>();
  keadaan2 = dataMega["keadaanServer2"].as<String>();

  suhu3 = dataMega["suhuServer3"];
  asap3 = dataMega["asapServer3"];
  api3 =  dataMega["apiServer3"].as<String>();
  keadaan3 = dataMega["keadaanServer3"].as<String>();

  cekData = "Ada";

}
```

Figure 8. Source code Receiving Arduino Mega Data

In Figure 9 is the source code for receiving Arduino Mega data where every temperature, smoke and fire data from Arduino Mega will be sent to nodemcu. In this process the nodemcu will check whether the data from the arduino mega is

"Available" if there is data it will be displayed on the serial nodemcu, if not, the data will not be displayed.

```
void tampilSerial(){
    Serial.println(" ");
    Serial.println("KEADAAN RUANGAN SERVER 1");
    String dataSensorServer1 = "Suhu:" + String(suhu1) + "    Asap:" + asap1 + "    Api:" + api1;
    Serial.println(dataSensorServer1);
    Serial.println(keadaan1);

    Serial.println("====================================================");
    Serial.println("KEADAAN RUANGAN SERVER 2");
    String dataSensorServer2 = "Suhu:" + String(suhu2) + "    Asap:" + asap2 + "    Api:" + api2;
    Serial.println(dataSensorServer2);
    Serial.println(keadaan2);

    Serial.println("====================================================");
    Serial.println("KEADAAN RUANGAN SERVER 3");
    String dataSensorServer3 = "Suhu:" + String(suhu3) + "    Asap:" + asap3 + "    Api:" + api3;
    Serial.println(dataSensorServer3);
    Serial.println(keadaan3);
    Serial.println("====================================================");
    Serial.println(" ");
}
```

Figure 9. Source code Displaying Data

In Figure 10, the source code displays the data sent to the Arduino Mega containing temperature, smoke, fire sensor data and the state of the server room. In this study the authors use as many as 3 tools where each server is placed 1 tool.

```
void kirimFirebase(){
    if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis > 15000 || sendDataPrevMillis == 0)){
    sendDataPrevMillis = millis();

    //server 1
    Firebase.RTDB.setFloat(&fbdo, "Server1/suhu", suhu1);
    Firebase.RTDB.setInt(&fbdo, "Server1/asap", asap1);
    Firebase.RTDB.setString(&fbdo, "Server1/api", api1);
    Firebase.RTDB.setString(&fbdo, "Server1/keadaan", keadaan1);
    //server 2
    Firebase.RTDB.setFloat(&fbdo, "Server2/suhu", suhu2);
    Firebase.RTDB.setInt(&fbdo, "Server2/asap", asap2);
    Firebase.RTDB.setString(&fbdo, "Server2/api", api2);
    Firebase.RTDB.setString(&fbdo, "Server2/keadaan", keadaan2);
    //server 1
    Firebase.RTDB.setFloat(&fbdo, "Server3/suhu", suhu3);
    Firebase.RTDB.setInt(&fbdo, "Server3/asap", asap3);
    Firebase.RTDB.setString(&fbdo, "Server3/api", api3);
    Firebase.RTDB.setString(&fbdo, "Server3/keadaan", keadaan3);
```

Figure 10. Source code Sending Data to firebase

3.3.Implementation of Hardware Test Results

In system testing, the aim is to find out whether the system can work well or not. The test was carried out during working hours on July 28, 2022, which is to find out conditions related to temperature, smoke, and fire on server devices. Testing is carried out with 2 schemes, namely:

a. On the second day of system testing using real data where testing is carried out on Thursdays at 9.10 - 13.02 to find out whether the system can work well or not. The following in table 2 is the result of testing conditions a.

Table 2. System test results with condition A

Test Time	Server Type	Temperature	Smoke	Fire	Fuzzy Condition	Cooling Fan	Exhaust Fan	Electric current
01.00 (Hour)	Server 1	25.56 °C	153 ppm	Not Fire	Safe	Off	Off	On
	Server 2	24.69 °C	223 ppm	Not Fire	Safe	Off	Off	On

Test Time	Server Type	Temperature	Smoke	Fire	Fuzzy Condition	Cooling Fan	Exhaust Fan	Electric current
	Server 3	24. 50 °C	211 ppm	Not Fire	Safe	Off	Off	On
	Server 1	24. 63 °C	153 ppm	Not Fire	Safe	Off	Off	On
02.00 (Hour)	Server 2	24. 75 °C	223 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	224 ppm	Not Fire	Safe	Off	Off	On
	Server 1	24. 31 °C	152 ppm	Not Fire	Safe	Off	Off	On
03.00 (Hour)	Server 2	24. 69 °C	222 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	215 ppm	Not Fire	Safe	Off	Off	On
	Server 1	24. 25 °C	150 ppm	Not Fire	Safe	Off	Off	On
04.00 (Hour)	Server 2	24. 81 °C	222 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	215 ppm	Not Fire	Safe	Off	Off	On
05.00 (Hour)	Server 1	24. 25 °C	150 ppm	Not Fire	Safe	Off	Off	On

b. The system testing this time uses simulation data to determine whether the system can work properly or not by using certain conditions, namely turning on the fan as a cooler when it detects an abnormal temperature, turning on the fan as a smoke exhaust when the sensor detects smoke and turning off the current. When the fire sensor detects a fire, the test is carried out on Thursday at 9.10 – 13.02. The following in table 3 is the result of testing the schematic system b.

Table 3. System test results with condition B

Test Time	Server Type	Temperature	Smoke	Fire	Fuzzy Condition	Cooling Fan	Exhaust Fan	Electric current
	Server 1	24. 00 °C	156 ppm	Not Fire	Safe	Off	Off	On
01.00 (Hour)	Server 2	24. 63 °C	228 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	217 ppm	Not Fire	Safe	Off	Off	On

Test Time	Server Type	Temperature	Smoke	Fire	Fuzzy Condition	Cooling Fan	Exhaust Fan	Electric current
02.00 (Hour)	Server 1	35. 00 °C	389 ppm	Not Fire	Dangerous	On	On	On
	Server 2	24. 63 °C	227 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	218 ppm	Not Fire	Safe	Off	Off	On
03.00 (Hour)	Server 1	24. 00 °C	155 ppm	Not Fire	Safe	Off	Off	On
	Server 2	24. 63 °C	226 ppm	Not Fire	Safe	Off	Off	On
	Server 3	24. 50 °C	500 ppm	Fire	Dangerous	Off	On	Off
04.00 (Hour)	Server 1	23. 94 °C	155 ppm	Not Fire	Safe	Off	Off	On
	Server 2	32. 63 °C	226 ppm	Not Fire	Standby	On	Off	On
	Server 3	24. 50 °C	225 ppm	Not Fire	Safe	Off	Off	On
05.00 (Hour)	Server 1	24. 00 °C	155 ppm	Not Fire	Safe	Off	Off	On
	Server 2	24. 63 °C	225 ppm	Fire	Dangerous	Off	Off	Off
	Server 3	24. 50 °C	211 ppm	Not Fire	Safe	Off	Off	On

In table 2 the results of testing the system condition a and table 3 the results of testing the condition system b can be analyzed that the system is working well which is where supervision is carried out for 1-2 minutes the temperature, smoke and fire sensors do not experience drastic changes in value and when the value of the sensor temperature that has not reached > 30 °C then the cooling fan of the server device is "Off" which means it is not on. when the smoke value has not reached > 400 then the fan as a smoke exhaust does not turn on and when the fire sensor does not detect a fire then the electric current is still alive. In table 3, the smoke sensor when the system is turned on has a high value and the longer the system is turned on, the smoke sensor will experience a change in value, because the smoke sensor takes time to read and get a fairly accurate value for smoke levels in condition a. In condition b using simulation data where the data is data created to test whether the temperature, smoke and fire sensors can function and provide action according to certain parameters where the temperature sensor parameter > 30 °C gives the server device cooling fan action "On" , the smoke sensor > 400 ppm gives the smoke exhaust fan action "On" and the fire sensor when it detects a fire then provides the action of turning off the electric current. In table 2 the results of testing condition b where in the test of 2 servers 1 temperature sensor, the smoke sensor that was tested was increasing the temperature to 35 °C where the action of the cooling fan of the server device was "Live", increasing smoke levels reaching 389 ppm which the action “On” exhaust fan and fuzzy system display “Danger” room condition. In test 3 for server 3, smoke and fire sensors were tested. The smoke sensor was tested, namely increasing the smoke level to 500 ppm which is where the smoke exhaust fan action is "Live". The fire sensor is tested by intentionally giving a fire where the action of the circuit breaker will issue "Off" data. fuzzy system displays the room condition data "Danger".

3.4. 1st Server Device Monitoring

Server 1 device monitoring can be done via a mobile application. The application can display temperature, smoke, fire values in realtime. Figure 11 is the result of the server 1 device monitoring display.

Figure 11. Results of 1st Server Device Monitoring

Figure 11 is a monitoring display for server 1 which has a function to view sensor readings on server 1 device. To open this feature the user can select the "Server 1" menu on the main page.

3.5. 2nd Server Device Monitoring

Server 2 device monitoring can be done via the mobile application. The application can display temperature, smoke, fire values in realtime. In figure 12 is the result of the server 2 device monitoring display.

Figure 12. Monitoring Results of the 2nd Server Device

Figure 12 is a monitoring display for server 2 which has a function to view sensor readings on server device 2. To open this feature the user can select the "Server 2" menu on the main page, then the application will display the values of the sensors that have been installed on the server. server device 2.

3.6. 3rd Server Device Monitoring

Server 3 device monitoring can be done via a mobile application. The application can display temperature, smoke, fire values in realtime. In Figure 13 is the result of the monitoring of server 3 devices.

Figure 13. Monitoring Results of the 3rd Server Device

In figure 13 is a monitoring display for server 3 which has a function to view sensor readings on server 3 devices. To open this feature the user can select the "Server 3" menu on the main page, then the application will display the values of the sensors that have been installed on the server. server device 3.

3.7. Sensor Notification Testing Detect

The mobile application provides a notification feature to users so that users can find out that the sensor detects a temperature value > 30, the smoke sensor > 300, the fire sensor detects a fire, then a notification will be displayed on the application. In Figure 14 is a notification display on a mobile application.

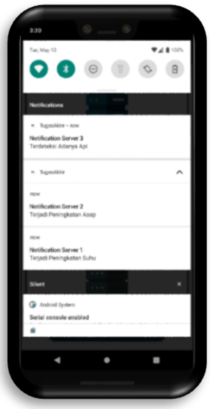

Figure 14. Sensor Notification Test Results Detect

In Figure 14 is a notification display that has a function to notify the default hardware or each sensor detects an increase in temperature, smoke and fire are detected in the server room, the notification will be displayed on the application.

4. CONCLUSION

Based on the system testing that has been carried out, the conclusions obtained in the study are:

1. The monitoring and control system for temperature and internet-based fire server devices is integrated with a mobile application using the DS18B20 sensor for temperature input, MQ2 sensor for smoke level input and flame sensor for fire input. This makes the main parameters as inputs to be processed on the Arduino mega board and then the fuzification process continues to provide an output state of the server device is safe or not. In fuzzy logic calculations, it is done by forming a linguistic variable whose input is a number that is mapped within a certain range. Then testing using matlab software to determine the results of the fuzzification calculation which has a total of 18 rules. After the rules are determined, the process of calculating the maximum value of each of the specified rules can be determined so that the bias towards one rule can be determined and will issue an output to determine whether the server device is safe or not. Fuzification results from the data obtained on the DS18B20 and MQ2 sensors have 3 fuzzy variables, and the Flame sensor has 2 fuzzy variables. The DS18B20 sensor has 3 variables, namely cold with a range of values (0-10 °C), normal with a range of values (10-25 °C), hot with a range of values (25-40 °C). The MQ2 smoke sensor has 3 variables, namely thin with a range of values (0-25 ppm), medium with a range of values (25-50 ppm), thick with a range of values (50-100 ppm). The flame sensor has 2 variables, namely Yes with a range of values (1), None with a range of values (0). From the results of the fuzification calculation, there are 18 rules that have 3 final outputs, namely Safe, Alert, and Danger. Determination of whether the output is Safe, Standby, and Danger must each meet one of 18 rules.
2. Devices designed for automatic fire control and surveillance systems server devices using temperature, smoke and fire sensors, NodeMCU functions to transmit data from existing sensor readings to be sent to the firebase database. The mobile application that has been created functions to retrieve data on the

firebase database to be able to display information related to temperature, smoke levels, fire and the state of the server device is safe or not to the server admin. In the system made there is an automatic control where when there is an increase in temperature in the range (> 30°C) the system will turn on the fan automatically to lower the temperature. When the system detects smoke in the range (>300 ppm), the system will turn on the exhaust fan to remove the smoke confined to the server device. When the system detects a fire in range (1), the system will cut off the electrical current of the server device to reduce the possibility of a short circuit spreading to the device and/or other rooms in the server room.

3. On the results of testing the hardware and software systems can function properly. In hardware testing, the DS18B20 sensor works well which can detect the temperature value on the server device. The DS18B20 sensor error percentage when compared to conventional temperature measuring instruments has an average error of 1.76%. The MQ2 sensor and fire sensor can function properly which can detect smoke levels and the presence of fire on the server device. In the automatic control that has been made, it can run according to its function where when there is an increase in temperature in the range (>30°C) the cooling fan is on and in the range (<30°C) it can be turned off. There is an increase in smoke, the exhaust fan can run in the range (>300 ppm) and in the range (<300 ppm) it can turn off and if a fire is detected in range (1) it can turn off the current on the server device. In testing the application software can display temperature data, smoke levels and detect the presence or absence of fire and can send notifications when there is an increase in temperature, smoke and detect fire in the mobile application.

REFERENCES

[1] S. W. Mustika, R. S. Wardani, and D. B. Prasetio, "PENILAIAN RISIKO KEBAKARAN GEDUNG BERTINGKAT Sika," *J. Kesehat. Masy. Indones.*, vol. 13, no. 1, pp. 18–25, 2018.

[2] Y. Efendi, "Internet Of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile," *J. Ilm. Ilmu Komput.*, vol. 4, no. 2, pp. 21–27, 2018, doi: 10.35329/jiik.v4i2.41.

[3] A. Setiawan, M. C. Untoro, M. Praseptiawan, A. Afriansyah, and I. F. Ashari, "Sistem Kendali Tanaman Hidroponik dengan Internet of Things di Kelompok Tani Hidroponik Tulang Bawang Barat," *J. Pendidik. dan Konseling*, vol. 4, no. 2, pp. 3880–3885, 2022, [Online]. Available: https://core.ac.uk/download/pdf/322599509.pdf.

[4] I. F. Ashari and A. Ayuningtyas, "PACKAGE RECEIVER BOX BASED ON IOT USING FUZZY MAMDANI AND MOBILE APPLICATION," *J. Ilmu Pengetah. dan Teknol. Komput.*, vol. 8, no. 1, pp. 56–64, 2022, doi: 10.33480/jitk.v8i1.2982.In.

[5] G. Santoso *et al.*, "Rancang Bangun Sistem Monitoring Suhu dan Kelembapan pada Ruang Server Berbasis IoT (Internet Of Things)," *J. Teknol. Technoscientia*, vol. 11, no. 2, pp. 186–193, 2019, [Online]. Available: https://ejournal.akprind.ac.id/index.php/technoscientia/article/view/1248.

[6] I. F. Ashari, M. D. Satria, and M. Idris, "Parking System Optimization Based on IoT using Face and Vehicle Plat Recognition via Amazon Web Service and ESP-32 CAM," *Comput. Eng. Appl. J.*, vol. 11, no. 2, pp. 137–153, 2022, doi: 10.18495/comengapp.v11i2.409.

[7] M. A. Ulinnuha, "Rancang Bangun Sistem Monitoring Suhu dan Kelembaban Ruang Server ITN Malang Berbasis web," *J. Mhs. Tek. Inform.*, vol. 1, no. 1, pp. 732–738, 2017.

[8] A. P. Ratnasari and F. A. Dwiyanto, "Metode Steganografi Citra Digital," *Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 2, no. 2, p. 52, 2020, doi: 10.30872/jsakti.v2i2.3300.

[9] I. F. Ashari, "Sistem Monitoring dan Kontrol Budidaya Ikan Nila Berbasis IoT dengan

Bioflok (Studi kasus: Kelompok Budidaya Ikan Sadewa Mandiri, Pringsewu)," *Suluah Bendang*, vol. 22, no. 2, pp. 417–427, 2022, doi: 10.24036/sb.02680.

[10] S. Bahri, "Sistem Keamanan Ruang Server Menggunakan Teknologi Rfid Dan Password," *J. Elektum*, vol. 15, no. 1, pp. 11–18, 2018, [Online]. Available: https://jurnal.umj.ac.id/index.php/elektum/article/view/2117.

[11] E. B. Raharjo, S. Marwanto, and A. Romadhona, "Rancangan Sistem Monitoring Suhu Dan Kelembapan Ruang Server," *Teknika*, vol. 6, no. 2, pp. 61–68, 2019.

[12] M. Irfan, L. P. Ayuningtias, and J. Jumadi, "Analisa Perbandingan Logic Fuzzy Metode Tsukamoto, Sugeno, Dan Mamdani (Studi Kasus : Prediksi Jumlah Pendaftar Mahasiswa Baru Fakultas Sains Dan Teknologi Uin Sunan Gunung Djati Bandung)," *J. Tek. Inform.*, vol. 10, no. 1, pp. 9–16, 2018, doi: 10.15408/jti.v10i1.6810.

[13] S. Batubara, "Analisis Perbandingan Metode Fuzzy Mamdani Dan Fuzzy Sugeno Untuk Penentuan Kualitas Cor Beton Instan," *It J. Res. Dev.*, vol. 2, no. 1, pp. 1–11, 2017, doi: 10.25299/itjrd.2017.vol2(1).644.

[14] S. Widaningsih, "Perbandingan Metode Data Mining Untuk Prediksi Nilai Dan Waktu Kelulusan Mahasiswa Prodi Teknik Informatika Dengan Algoritma C4,5, Naïve Bayes, Knn Dan Svm," *J. Tekno Insentif*, vol. 13, no. 1, pp. 16–25, 2019, doi: 10.36787/jti.v13i1.78.

[15] S. Amalia, R. Andari, and R. Syukriansyah, "Studi Pemodelan Sistem Pengontrolan Suhu Ruangan Berbasis Logika Fuzzy Sugeno," *J. Sains dan Teknol. J. Keilmuan dan Apl. Teknol. Ind.*, vol. 20, no. 2, p. 175, 2020, doi: 10.36275/stsp.v20i2.287.

[16] H. Ubaya and K. Exhaudi, "Monitoring Temperatur dan Kelembaban Ruang Server Berbasis Web Telegram," *Generic*, vol. 12 No 2, pp. 27–31, 2020.

[17] B. I. Gunawan and U. Y. Oktiawati, "JURNAL RESTI Sistem Pemantau dan Pengendali Suhu Ruang Server Menggunakan Fuzzy," *J. Resti*, vol. 4, no. 10, pp. 1–9, 2020.