European Scholar Journal (ESJ)
Available Online at: https://www.scholarzest.com
Vol. 2 No.12, December 2021
ISSN: 2660-5562

ILS ROOM TEMPERATURE MONITORING AT JUANDA AIRPORT WITH ANDROID IOT BASED USING MESSAGE QUEUE TELEMETRY TRANSPORT PROTOCOL

Agung Santoso, Isturom Arif, Mochammad Masduki Rachmanto
Universitas Maarif Hasyim Latif
agung@dosen.umaha.ac.id, isturomarif@dosen.umaha.ac.id,
mochammad_masduki_rachmanto@gmail.com

Article history:

Received: 10th October 2021
Accepted: 11th November 2021
Published: 26th December 2021

Abstract:

Room temperature monitoring Instrument Landing System (ILS) is critical to keep the equipment working correctly. So that the airplane that will land can receive a good signal and the landing can be carried out perfectly. Remote monitoring must be done in real-time and continuously. Therefore data retrieval is also done wirelessly. So that only sensor equipment and transmission systems are on site.

By utilizing the Message Queue Telemetry Transport (MQTT), a simple message protocol that can publish and subscribe. The MQTT protocol is designed to be open and easy to implement, capable of handling thousands of remote clients with only one server. MQTT minimizes network bandwidth and device resource requirements. This approach makes the MQTT protocol very suitable for connecting machine to machine.

During the performance of the tool, an internet connection must be connected. The DHT11 sensor module retrieves the temperature value in the ILS chamber. Furthermore, the temperature value is sent by the NodeMCU to the MQTT server and displayed on the smartphone and the MQTT-Dash Android application.

Keywords: ILS, MQTT, DHT11, NodeMCU and Smartphones.

I.INTRODUCTION

Instrument Landing System (ILS) is an instrument landing aid (non-visual) that is used for aircraft pilots in carrying out aircraft landing procedures at an airport (airport) safely. Benefits of ILS [1] "Localizer Transmitters as right/left guides from the runway axis, Glide Slope Transmitters as landing angle guides on the runway axis and Marker Beacons (Inner, Middle, Outer Marker Beacons) located at a certain distance from the threshold as horizontal distance guides to the runway threshold.

Technicians never guard the ILS room at the airport. Technicians only check the condition of the machine once a week. Therefore, the temperature in the ILS room must always be relatively calm. Temperature monitoring is critical to keep the equipment working correctly. So that the airplane that will land can receive a good signal and the landing can be carried out perfectly.

This study proposes an ILS room temperature monitoring tool remotely in real-time. Data retrieval is carried out wirelessly so that only sensor equipment and transmission systems are on site. The Message Queue Telemetry Transport (MQTT) protocol is a simple message protocol that has publish and subscribe capabilities. The MQTT protocol is designed to be open and easy to implement, capable of handling thousands of remote clients with only one server. MQTT minimizes network bandwidth and device resource requirements. This approach makes the MQTT protocol suitable for connecting machine to machine (M2M), an essential aspect of the concept of the Internet of Things [2]. Internet of Things (IoT) is a network that connects various objects that have identifiers and IP addresses. IoT can communicate with each other and exchange information about itself and the environment it senses. IoT can use or produce services and work together to achieve a common goal [3]. IoT is a concept that aims to expand the benefits of the already widespread internet connectivity. IoT works by utilizing a programming algorithm that generates arguments where each argument command automatically produces an interaction between connected devices at any distance [4]. IoT is a concept that aims to expand the benefits of the already widespread internet connectivity. IoT works by utilizing a programming algorithm that generates arguments where each argument command automatically produces an interaction between connected devices at any distance [4]. IoT is a concept that aims to expand the benefits of the already widespread internet connectivity. IoT works by utilizing a programming algorithm that generates arguments where each argument command automatically produces an interaction between connected devices at any distance [4].

Initialize the input, and output pins, which the android device will send by converting the datagram sent into logic commands "LOW" and "HIGH". To activate and deactivate the system for each component and initialize the IP address to be the destination address for sending datagrams from the android device[5]

II.RESEARCH METHODS

Starting with the system's flow, the design of a system diagram consists of determining the components used. Create a hardware prototype. Program the sensor node and the monitor node for the last test. Figure 1 is the flow of the temperature monitoring system.

System Flow

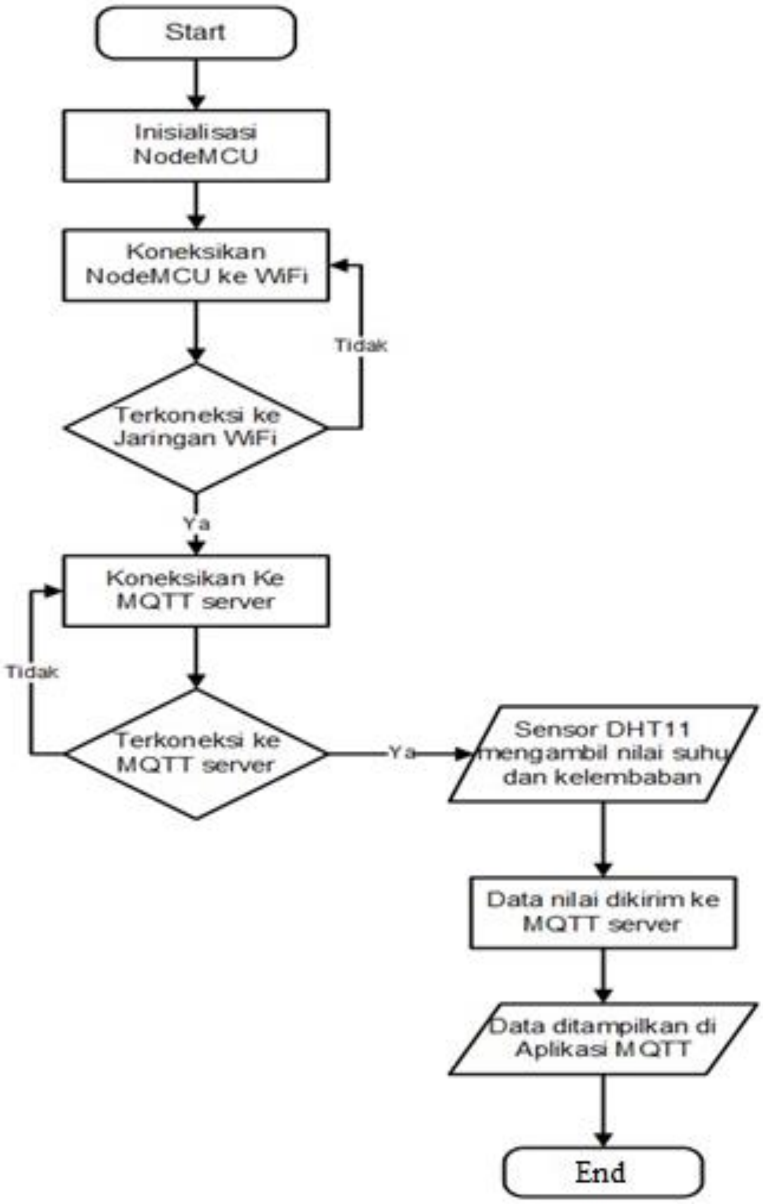

Figure 1. The flow of Temperature Monitoring System

Figure 1 describes the initialization of NodeMCU to get an internet connection via a wifi network. The connection to the MQTT server is also established. If the link is not yet connected, check the server's MQTT connection if the connection is already installed. Then the DHT11 sensor module will take the temperature value in the ILS room. After the sensor retrieves the data, the data will be sent to the MQTT server and then displayed in the Android-based MQTT application.

System Architecture

The system architecture describes the temperature monitoring system in the form of a complete design of the equipment prototype to be made.

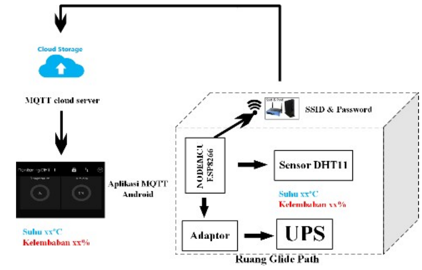

Figure 2. Temperature Monitoring System Architecture

The DHT11 temperature sensor reads the Glide Path room temperature. Then the data is processed by the NodeMCU ESP8266 and sent to the cloud via the SSID. It will also be displayed in the MQTT Dash app. The following is a detailed description of the flow of Figure 2:

1. NodeMCU: a microcontroller to send programs to the DHT11 sensor and receiver the data transmitted by the sensor. This module also functions as a sender of data from the DHT11 sensor to be sent to ANDROID.
2. Adapter: as a resource for NodeMCU.
3. DHT11 sensor: module to accommodate temperature value data in the ILS room.
4. UPS: backup battery power supply in case of power failure.
5. Cloud Used as data storage from NodeMCU.

The DHT11 sensor is a calibrated digital signal. The temperature measurement range of this sensor is 0-50°C, and the relative humidity measurement range is 20-90%. The DHT11 sensor requires a power supply of 3 to 5.5 Volt DC. Accuracy for relative humidity is ± 4%, and accuracy for temperature is ± 2°C. The physical form of the DHT11 sensor[6]

Hardware

In the hardware, various components are used, including the NodeMCU ESP8266, the DHT11 sensor module and the Male To Female jumper cable. Using NodeMCU ESP8266 because it functions as a versatile wifi module[7]

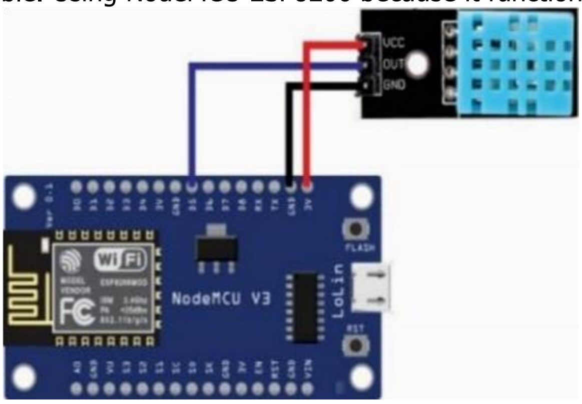

Figure 3 System build

Figure 3 system circuit consists of PIN D5 on the NodeMCU connected to Pin S (output) on the DHT11 sensor module. The GND pin is connected to the G (Ground) pin of the DHT11 sensor. Pin 3V is connected to Pin V (voltage source) on the DHT11 sensor.

MQTT Dash Configuration

MQTT Dash turns intelligent phones into clients. ServerMQTT broker serves as a setting for running applications. The server must be named, given the widget's address, port, and call, as shown in Figure 4

Figure 4 MQTT Broker Address Setting and Topic Naming

The server name in the MQTT-Dash configuration is Monitoring DHT11. Address MQTT-Dash at broker.mqtt-dashcard.com with port 1883. Humidity widget name, Name : Humidity, Topic (sub) : DHT11Humi, postfix : % (optional) and progress color : Blue (optional). Widget Temperature, Name : Temperature, topic (sub) : DHT11Temp, Postfix : c (optional) and Progress color : Red (optional)[7].

III. RESULTS AND TESTS

The temperature monitoring prototype is shown in Figure 5. There is 1 NodeMCU Baseboard, 1 ESP8266 NodeMCU, 1 breadboard, 1 DHT11 sensor. The DHT11 sensor is directly connected to the NodeMCU according to the specified PIN.

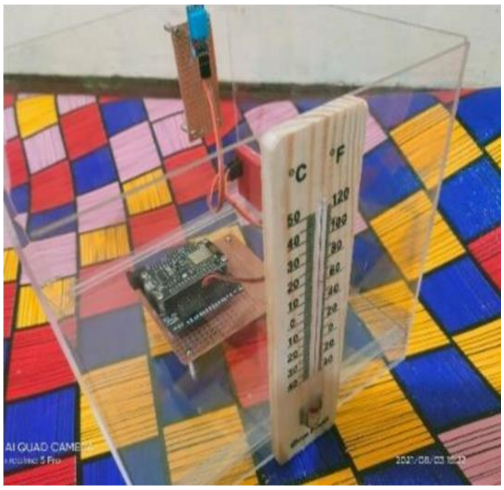

Figure 5. Temperature Monitoring Prototype

Test

This system testing is done to determine whether this tool can run well. To measure the percentage level of the error difference between the temperature value from the MQTT Dash application and the temperature value displayed by the serial monitor, the following formula is used.

$$\text{error} \frac{\text{selisih suhu}}{\text{suhu asli}} 100 \quad (1)$$

The original temperature is the temperature on the room thermometer. The temperature difference is the original temperature value minus the temperature value displayed by the MQTT Dash application. The error is obtained by dividing the difference in temperature divided by the original temperature.

Table 2 System test results

Day to	O'clock	Application MQTT		room thermometer		Error	Accuracy
1	20.56	22 °c	71 %	22.5 °c	71%	2.2%	97.8%
	21.04	24 °c	82 %	23°c	81%	4.3%	95.7%
	21.32	21 °c	78 %	20°c	78%	5%	95%
	22.08	19 °c	77 %	18.9 °c	77%	0.5 %	99.5%
	22.30	19 °c	77 %	18.4 °c	76%	3.2%	96.8%
2	19.22	23 °c	60 %	24°c	61%	4.1%	95.9%
	20.39	24 °c	67 %	24.7 °c	66%	2.8%	97.2%
	21.09	24 °c	66 %	25°c	64%	4%	96%
	21.18	24 °c	69 %	24.3°c	67%	1.2%	98.8%
	22.00	21 °c	72 %	20.8°c	72%	1%	98%
3	06.20	25 °c	55 %	25°c	55%	0%	100%
	06.26	25 °c	54 %	24.9°c	54%	0.5 %	95.5%
	06.37	25 °c	54 %	24.6°c	54%	1.6%	98.4%
	06.49	25 °c	55 %	24.5°c	55%	2%	98%
	06.51	25 °c	57 %	24.5 °c	55%	2%	98%
4	06.06	24 °c	58 %	24.4 °c	58%	1.6%	98.4%
	06.20	24 °c	56 %	23.9 °c	57%	0.4%	99.6%
	06.29	24 °c	57 %	23.8 °c	57%	0.8%	99.2%
	06.33	24 °c	58 %	23.7 °c	57%	1.3%	98.7%
	06.40	24 °c	58 %	23.7 °c	57%	1.3%	98.7%
Average accuracy						**97.76%**	

From table 2 above, it can be concluded that the tool works well, but several factors cause the resulting value to be different from that in the room thermometer. The first factor is the unstable internet network connected to the NodeMCU. The data transmission process often experiences delays, resulting in the value in the MQTT application not being the same as the value on the thermometer.

The second factor The thermometer used does not guarantee that the resulting value is accurate because when testing on the 3rd and 4th day using a thermometer that has been calibrated, the value generated by the application and the thermometer is almost the same and has a slight difference.

IV. CONCLUSIONS AND SUGGESTIONS

Remote temperature monitoring can be concluded that the prototype shows maximum results. The prototype is very suitable for monitoring the temperature of a room so that the equipment is not damaged due to overheating. The main component of the prototype is the DHT11 sensor which takes the temperature value in the room. Then the temperature value is sent by NodeMCU to the MQTT server. The MQTT protocol was chosen because it could minimize network bandwidth and device resource requirements. Next, the temperature value is displayed on the smartphone.

REFERENCES

1. Feti Fatonah, 2014, Method of Measuring Equipment Localizer in Airport (Case Study Sam Ratulangi Airport Manado), https://media.neliti.com/media/publications/234194-metode-pengukuran-peralatan-localizer-di-697c6bf7.pdf
2. Totok Budioko.2016. SISTEM MONITORING SUHU JARAK JAUH BERBASIS INTERNET OF THINGS MENGGUNAKAN PROTOKOL MQTT, Seminar Riset Teknologi Informasi (SRITI). STMIK AKAKOM YOGYAKARTA
3. Fujiama Diapoldo Silalahi, Jarot Dian, Nuris Dwi Setiawan, 2021*. Implementasi Internet Of Things (Iot) Dalam Monitoring Suhu Dan Kelembaban Ruang Produksi Obat Non Steril Menggunakan Arduino Berbasis Web*. Jurnal JUPITER, Vol. 13 No. 2 Bulan Oktober, Tahun 2021 , Hal. 62 – 68.
4. Sukandar Sawidin, Deitje S. Ponggoh, Ali Ramschie,2018, Rancang Bangun Sistem Kontrol Temperatur Dan Kelembaban Ruangan Dengan Android, Politeknik Negeri Bandung, https://jurnal.polban.ac.id/proceeding/article/view/1073/875.
5. A Najmurrokhman1,*, Kusnandar2, Amrulloh3, 2018, PROTOTIPE PENGENDALI SUHU DAN KELEMBABAN UNTUK COLD STORAGE MENGGUNAKAN MIKROKONTROLER ATMEGA328 DAN SENSOR DHT11, Jurnal Teknologi, Universitas Muhamadiyah Jakarta, 10(1).
6. Mochamad Fajar Wicaksono,2017, IMPLEMENTASI MODUL WIFI NODEMCU ESP8266 UNTUK SMART HOME,Jurnal Teknik Komputer UNIKOM, 6(1)
7. D.Robotics. 2020, Juli 30. *DHT11 Humidity & Temperature Sensor.* https://datasheet4u.com/datasheetparts/DHT11-datasheet.php?id=785590.