Integration of MQTT and Node-RED for IoT-Based Server Room Temperature Monitoring System

Yoanda Alim Syahbana[1,a)], Aditya Jati [2,b)]

[1]Computer Engineering Technology, Politeknik Caltex Riau, Pekanbaru, Indonesia
[2]Informatics Engineering, Politeknik Caltex Riau, Pekanbaru, Indonesia

[a)]Corresponding author: yoanda@pcr.ac.id
[b)] aditya20ti@mahasiswa.pcr.ac.id

Abstract. The absence of temperature monitoring in the server room of PT PLN (Persero) Unit Induk Distribusi Riau and Kepulauan Riau (UID RKR) poses a high risk to the stability of business processes at UID RKR. Without reliable temperature monitoring, servers can experience overheating, potentially causing equipment damage and data loss. This would result in disruptions to the electricity services provided by UID RKR. Based on these potential issues, this research implements an IoT device consisting of an ESP32 and a DHT22 temperature sensor connected to the Telegram application. The IoT device monitors the server room temperature in real-time and sends notification messages through the Telegram chatbot API if there is a drastic temperature change in the server room. The Information Technology Systems Division (DSTI) of UID RKR can also send commands to the chatbot to obtain device status. Communication between the IoT device and the chatbot is facilitated through the MQTT protocol. The integration of IoT devices, API, and the visual display of the monitoring system is developed using Node-RED, which strongly supports real-time data processing. The monitoring system has been tested, including initial connection tests, temperature monitoring functionality, Telegram API message delivery, and system reliability. The test results show that the system has worked 100% effectively and can reliably monitor the server room temperature 24/7 in real-time. Validation with DSTI has also been conducted, and DSTI acknowledges the benefits provided by the monitoring system.

Keywords: Temperature Monitoring, IoT, MQTT, Node-RED

INTRODUCTION

PT PLN (Persero) Unit Induk Distribusi Riau and Kepulauan Riau (UID RKR) is one of the business units of PT PLN (Persero), a state-owned enterprise engaged in the electricity sector. UID RKR is responsible for distributing electricity in the Riau and Riau Islands regions, including remote areas and small islands. Its role includes providing reliable and high-quality electricity for household, commercial, industrial, and public needs. They also maintain distribution networks and carry out repairs and infrastructure development for electricity. Other services include new electricity connections, power changes, and bill payments. UID RKR supports innovations such as renewable energy (RE) and smart grids. Facing geographical challenges, UID RKR continues to innovate in the fields of technology and management. These initiatives are in line with national energy transformation and the government's sustainable electricity targets.

PLN UID RKR has a server room that functions as a center for network traffic management and intranet data storage, which also serves as an internet access hub for its employees. This server room is monitored by the Information Technology Systems Division (STI) of PLN UID RKR. The current problem is that there are no employees available to directly monitor the server room temperature. This is because the temperature inside the server room is extremely cold, making it inadvisable to stay inside for long periods. However, this cold temperature can sometimes become hot, potentially causing the server to overheat, even emitting smoke.

The absence of temperature monitoring in the server room of PT PLN (Persero) Unit Induk Distribusi Riau and Kepulauan Riau (UID RKR) poses a significant risk to the stability of business processes at UID RKR. Without reliable temperature monitoring, servers may experience overheating, potentially leading to equipment damage and data loss. This would result in disruptions to the electricity services provided by UID RKR.

The Internet of Things (IoT) has begun to be implemented as part of Industry 4.0 [1]. IoT is one of the key elements of the Industry 4.0 revolution, which can be used to monitor temperature [2][3][4]. IoT devices that are continuously connected to the internet make remote monitoring easier. IoT-based temperature monitoring systems using the Message Queuing Telemetry Transport (MQTT) protocol can send data or messages in real-time [5].

This research applies an IoT solution to monitor the server room temperature directly. The IoT devices used include the ESP32 and the DHT22 temperature sensor connected to the Telegram application. These devices monitor the server room temperature in real-time and send notifications via the Telegram chatbot API if there is a significant temperature change. In addition, DSTI UID RKR can send commands through the chatbot to check device status, as implemented in [6]. Communication between the IoT device and the chatbot is regulated using the MQTT protocol. The integration of IoT devices, APIs, and the monitoring interface is developed using Node-RED, which supports real-time data processing.

The discussion of this research is divided into four main sections. Beginning with the introduction, the first part presents the background and context of the research. The second part explains the application of MQTT and Node-RED integration in the IoT-based server room temperature monitoring system. Testing results and analysis are presented in the third section. Finally, the fourth section summarizes the conclusions based on the implementation results.

METHODS

The system architecture design for the IoT-based room temperature monitoring system for PLN UID RKR's server room is shown in Figure 1. This architectural design refers to literature studies from several previous research, such as in [7]-[13]. This research involves several integrated components to facilitate the monitoring of the server room temperature. The DHT22 sensor measures the temperature in the room and then sends the data to the NodeMCU ESP32. The DHT22 sensor was selected due to its wide measurement range, covering 0 to 100% for humidity and -40 degrees Celsius to 125 degrees Celsius for temperature [14]. The data is then sent to the MQTT broker through the MQTT protocol, which enables communication between devices and the dashboard. The MQTT broker acts as an intermediary to manage and distribute data to various systems. Node-RED, as a web-based development platform, receives data from MQTT and sends notification requests to the Telegram API as implemented in [15]. The Telegram API processes this request and sends a message to a designated Telegram group, providing real-time temperature information. Additionally, the temperature data is also displayed directly on an I2C LCD for local monitoring. MySQL serves as the database management system that stores temperature data for analysis and reporting. With MySQL, the collected data can be periodically stored and accessed for trend analysis or report generation, ensuring the data remains managed and available for long-term reference. Node-RED does not have an official mobile application, but its web interface can be accessed via a browser on mobile devices, allowing for remote monitoring and management. Public MQTT brokers, such as broker.hivemq.com [16], used in this research, allow communication between the ESP32 and other systems without requiring private server configuration.

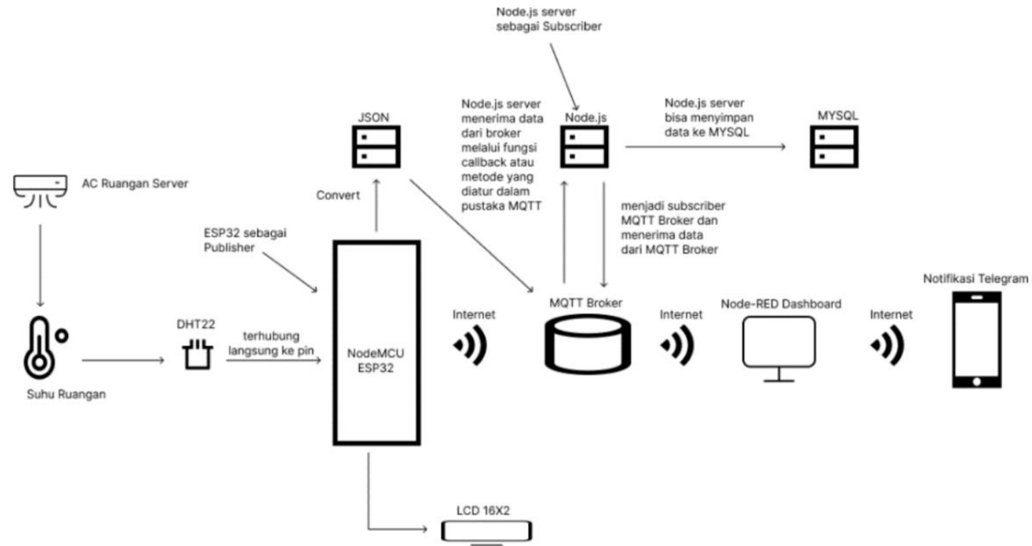

FIGURE 1. System architecture

Figure 2 shows the steps for temperature data collection and notification delivery to Telegram in a Data Flow Diagram (DFD). The DFD starts with the process of collecting temperature data from the DHT22 sensor installed in the server room. This temperature data is then forwarded to the processing stage in the ESP32. This process is aimed at processing and presenting the data before it is stored in the database. Afterward, the processed data is sent to the notification delivery process to Telegram. At this stage, the system uses the Telegram API to send notifications to a specified user or Telegram group about the temperature conditions in the server room.

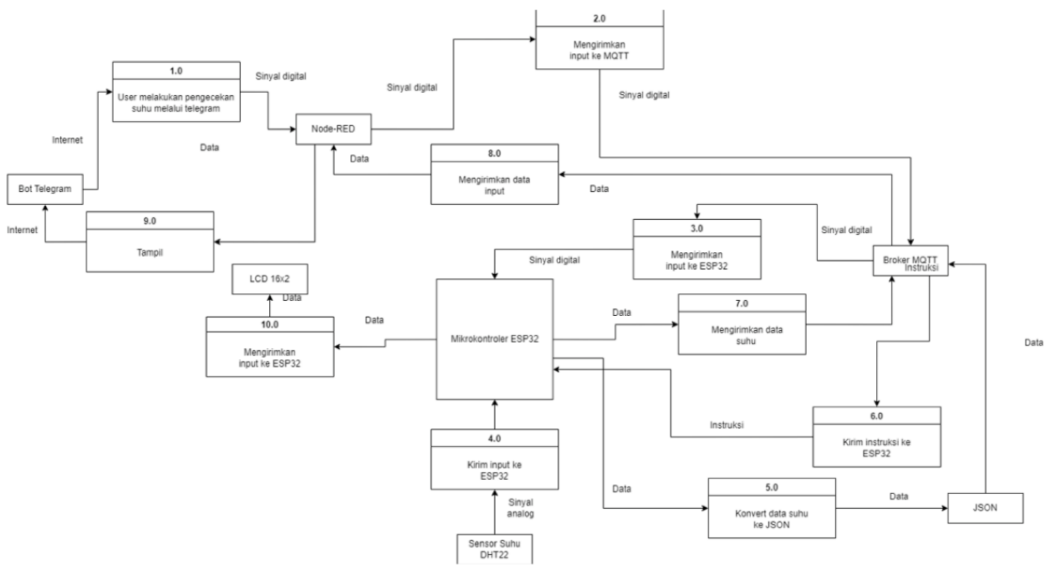

FIGURE 2. Data Flow Diagram

In the creation of the IoT device for monitoring server room temperature, the temperature is measured and transmitted by the DHT22 temperature sensor, then forwarded to the NodeMCU ESP32, which also functions as a Wi-Fi module to connect the device to the MQTT broker via PubSubClient. PubSubClient is a library used for the MQTT protocol on the ESP32. The collected temperature data is then connected to Node-RED. For public access, additional configuration is required to ensure data security in Node-RED and the Telegram API, which operate within the same network. The temperature data is also displayed on the Node-RED dashboard and an LCD.

Figure 3 shows the setup of the DHT22 sensor connected to a breadboard, reading temperature data, which is then displayed in real-time on the LCD. This setup also utilizes the Telegram API for remote monitoring via Telegram messages. Users can receive immediate alerts if the temperature exceeds a set threshold, as long as there is an internet connection. The ESP32 serves as a bridge between the DHT22 sensor and various platforms, including Telegram and Node-RED, through the MQTT protocol.

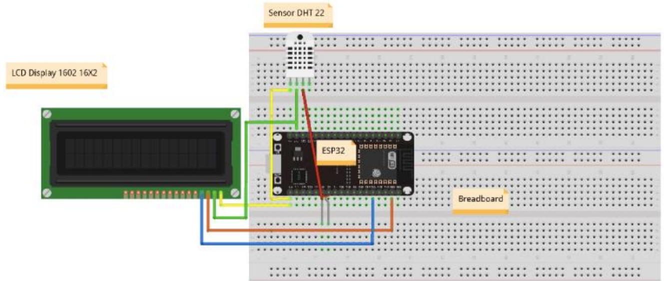

FIGURE 3. ESP32, DHT22 sensor, and LCD schematic

The data collected is sent in real-time via MQTT to the broker, which can be accessed through Node-RED for processing and display on the dashboard. Node-RED functions as a data visualization and IoT workflow management tool, allowing users to monitor temperature through the dashboard and create graphs. Figure 4 shows the Node-RED flow. This flow is designed to connect various devices and services through programmable blocks or nodes. In this research, the Node-RED flow is responsible for managing the data received from the DHT22 sensor connected to the ESP32. Node-RED periodically collects temperature and humidity data and sends it to the MQTT broker.

Once the data is received by the MQTT broker, the Node-RED flow processes the information and forwards it to other nodes for further analysis or specific actions. If the room temperature exceeds the predetermined threshold, a node in this flow will trigger the sending of a notification to the Telegram API to alert users about significant temperature changes. By using Node-RED, the data flow from the sensor to the user becomes more structured and efficient, enabling real-time and automated monitoring of the server room temperature.

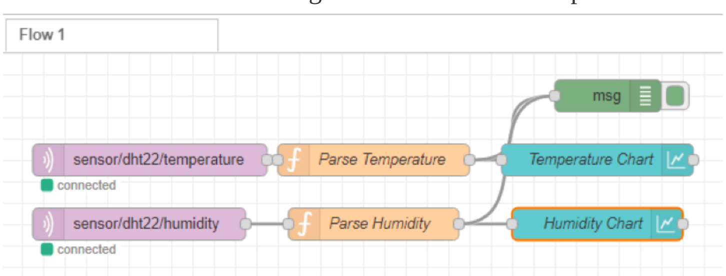

FIGURE 4. Node-RED flow for DHT22 sensor data

The testing in this research ensures that the ESP32 can connect properly to the Wi-Fi network. This involves configuring the connection correctly. Once the device is connected, it is important to maintain a stable connection so that the sensor data communication is not interrupted. Additionally, message delivery testing to the Telegram API was also conducted to ensure that the API is configured correctly and that a valid API token is used for communication. This testing includes sending test messages after the IoT device detects temperature changes to ensure the messages are accurately received by the specified Telegram group or recipient.

Temperature monitoring functionality testing was conducted by placing the DHT22 sensor in various locations to test the accuracy of the temperature readings and to monitor whether the temperature data is correctly sent from the sensor to the MQTT broker and displayed on the Node-RED dashboard. This testing also includes system availability tests by running the IoT device continuously to ensure uninterrupted operation, as well as simulating emergency scenarios to ensure fast and accurate temperature detection. Finally, remote monitoring testing was conducted to ensure that temperature data can be accessed from different locations via the Node-RED dashboard, which is accessed over the internet.

RESULTS AND DISCUSSION

The IoT device for server room temperature monitoring using the Telegram API and MQTT protocol, developed for the Information Technology Division of PT PLN (Persero) Unit Induk Distribusi Riau and Kepulauan Riau, has been successfully implemented. Figure 5 shows the IoT device in operation during monitoring.

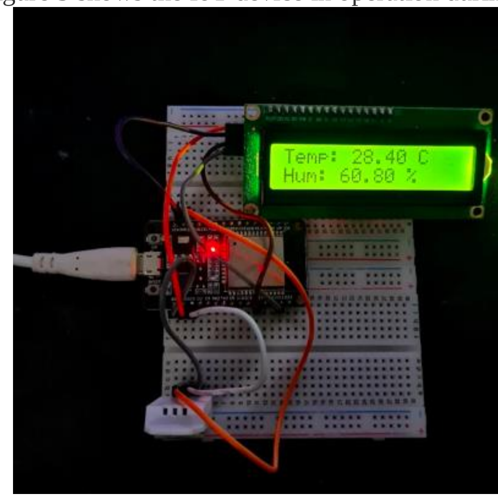

FIGURE 5. IoT device implementation

Figure 6 shows the Node-RED dashboard that provides real-time monitoring of temperature and humidity. The temperature is displayed in degrees Celsius (°C), while humidity is shown as a percentage (%). This dashboard presents information collected from the DHT22 sensor through the ESP32 and the MQTT broker, allowing users to view graphs of temperature and humidity, as well as the current status of the server room environment. This interface facilitates monitoring of room conditions without the need to access the hardware directly.

In addition to offering data visualization, the Node-RED dashboard also allows users to set desired temperature thresholds and receive notifications if the temperature exceeds the specified limits. This feature provides users with additional control to maintain stable conditions in the server room.

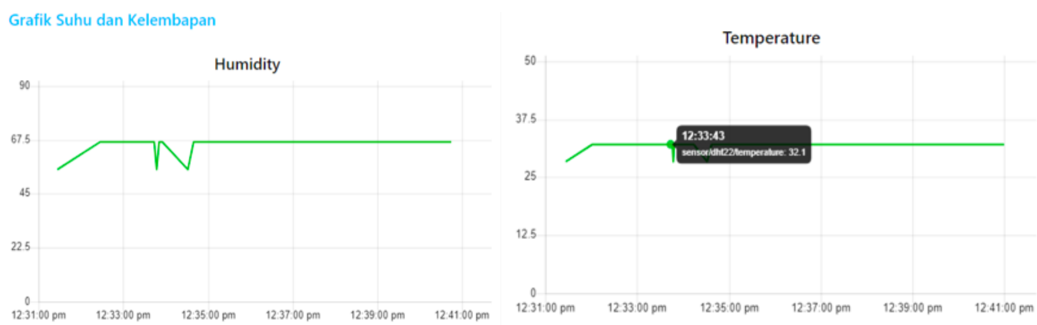

FIGURE 6. Node-RED dashboard

The IoT device has also successfully undergone a series of tests with highly satisfactory results. Out of a total of 60 tests conducted, covering various functional aspects, all were successful with a 100% success rate. Initial testing confirmed that the ESP32 device and other IoT components could connect to the Wi-Fi network and publish temperature data to the MQTT broker, with all 10 tests successfully demonstrating good connection management. The API integration testing for Telegram ensured proper API configuration and effectiveness in sending notification messages, with all 10 tests also successful.

Temperature monitoring functionality tests, which included placing the DHT22 sensor in various locations, also yielded satisfactory results, with all 10 tests successful, proving the accuracy and stability of temperature monitoring. Availability and stability testing confirmed that the system operates well over the long term and can effectively respond to temperature changes and message requests, with all 10 tests successful. Additionally, emergency scenario testing and remote monitoring demonstrated that the system can reliably provide emergency notifications and temperature monitoring from a distance, with all 10 tests also successful.

CONCLUSIONS

The implementation of MQTT and Node-RED integration for server room temperature monitoring using the Telegram API, developed for the Information Technology Division of PT PLN (Persero) Unit Induk Distribusi Riau and Kepulauan Riau, has been successfully tested with highly satisfactory results. Out of a total of 60 tests covering various functional aspects, all tests were successful with a 100% success rate. The test results indicate that the developed IoT system has achieved the research objectives and can effectively monitor the server room temperature, allowing for quick responses to temperature changes and preventing damage to the servers. This research is expected to have a positive impact on maintaining the stability and performance of the server room at PLN Unit Induk Distribusi Riau and Kepulauan Riau.

REFERENCES

[1] A. Gilchrist, *Industry 4.0: The Industrial Internet of Things*. Boca Raton, FL: CRC Press, 2016.

[2] B. Adi dan B. R. Saputra, "System Control dan Monitoring Suhu pada SUB DISTRIBUTION PANEL (SDP) Berbasis IoT Menggunakan Sensor DHT22 untuk Mengurangi Terjadinya Overheat pada PT.XYZ," *Jurnal Teknik Elektro*, pp. 67–72, 2022.

[3] F. Akbar dan Sugeng, "Implementasi sistem monitoring Suhu dan kelembaban ruangan penyimpanan obat berbasis Internet of Things (IoT)," *Jurnal Sosial Dan Teknologi (SOSTECH)*, vol. 1, no. 9, pp. 1021–1028, 2021.

[4] D. R. Kristiyanti, A. Wijayanto, and A. Aziz, "Sistem Monitoring Suhu dan Kelembaban pada Budidaya Jamur Tiram Berbasis Internet of Things Menggunakan MQTT dan Telegram BOT," *Adopsi Teknologi Dan Sistem Informasi (ATASI)*, vol. 1, no. 1, pp. 61–73, 2022.

[5] A. Saepudin, "Teknologi Internet Of Things Dalam Proses Monitoring Suhu dan Kelembaban Di Gudang Penyimpanan Bahan Kulit," *Jurnal Teknik Informatika Dan Sistem Informasi*, vol. 9, no. 4, pp. 2712–2719, 2022.

[6] S. Mamay and B. Saputro, "Implementasi remote monitoring pada virtual private server berbasis Telegram Bot API (studi kasus Politeknik TEDC Bandung)," *Jurnal Sistem Komputer Dan Kecerdasan Buatan*, vol. 4, no. 2, pp. 104–111, 2021.

[7] F. Vinola, A. Rakhman, and Sarjana, "Sistem Monitoring dan Controlling Suhu Ruangan Berbasis Internet of Things," *Jurnal Teknik Elektro Dan Komputer*, vol. 9, no. 2, pp. 117–126, 2020.

[8] M. F. A. Safii, S. Raharjo, and U. Lestari, "Analisis Quality of Service Protokol MQTT dan HTTP pada Penerapan Sistem Monitoring Suhu Berbasis NODEMCU (Studi Kasus Ruang Server Kampus 3 IST AKPRIND Yogyakarta)," *Jurnal JARKOM*, vol. 7, no. 1, pp. 11–19, 2019.

[9] D. I. Saputra, M. G. Karmel, and Y. B. Zainal, "Perancangan dan Implementasi Rapid Temperature Screening Contactless dan Jumlah Orang Berbasis IOT Dengan Protokol MQTT," *Journal of Energy and Electrical Engineering (JEEE)*, vol. 2, no. 1, pp. 20–30, 2020.

[10] D. Hermanto, A. N. Salim, and I. P. Putra, "Sistem Monitoring Suhu dan Kelembapan Udara Menggunakan Protokol MQTT Berbasis WEMOS D1 MINI," *Applicable Innovation of Engineering and Science Research*, pp. 111–116, 2021.

[11] I. Wardhana, V. A. Isnaini, R. P. Wirman, R. Syafitri, and A. Nasuha, "Rancang Bangun Alat Pengukur Suhu Real Time Laboratorium Menggunakan Protokol MQTT Berbasis Internet of Things," *Jurnal Teori Dan Aplikasi Fisika*, vol. 09, no. 01, pp. 39–46, 2021.

[12] Y. Yudhistira, "Perekam Data Intensitas Cahaya, Suhu, dan Kelembapan Udara Berbasis IoT (Internet of Things) Menggunakan ESP32 dan MQTT (Message Queuing Telemetry Transport)," *Jurnal Sistem Informasi Dan Teknologi Peradaban (JSITP)*, vol. 2, no. 2, pp. 45–48, 2021.

[13] I. Harjanto, "IoT Gateway Menggunakan Protokol MQTT pada Perangkat Kendali Berbasis Modbus-RTU," *Jurnal Ilmiah Teknosains*, vol. 6, no. 1, pp. 12–19, 2020.

[14] K. Diantoro, R. Rahmadewi, and Ibrahim, "Implementasi sensor MQ 4 dan sensor DHT 22 pada sistem kompos pintar berbasis IoT (SIKOMPI)," *Electrician – Jurnal Rekayasa Dan Teknologi Elektro*, vol. 14, no. 3, pp. 84–94, 2020.

[15] N. Fernando, Humaira, and E. Asri, "Monitoring jaringan dan notifikasi dengan telegram pada dinas komunikasi dan informatika kota Padang," *Jitsi*, vol. 1, no. 4, pp. 121–126, 2020. "broker.hivemq.com," [Online]. Available: https://broker.hivemq.com.