JIETA
Journal on Internet Engineering Technologies and Applications
Volume 1, **Number** 1, January 2024, pp. 45-52

Smart Home System Based on Internet of Things (IoT) for Monitoring and Controlling Room Temperature

Puji Aprilia Putri[1*], Herdian Saputra[2]
[1]Politeknik Negeri Padang, Indonesia, [2]Akademiki Komunitas Negeri Aceh Barat
[1]pujiaprilia711@gmail.com, , [2]herdian.saputra@aknacehbarart.ac.id

open access

***Corresponding Author**

Article History:
Submitted: 24-03-2024
Accepted: 24-04-2024
Published: 01-05-2024

Keywords:
Smart Home, Internet of Things (IoT), Temperature Monitoring, DHT22 Sensor, Node-Red, ESP32 Microcontroller, MQTT, Raspberry Pi

ABSTRACT

This research developed an Internet of Things (IoT)-based Smart Home system for monitoring and controlling room temperature using the DHT22 sensor. The system utilizes an ESP32 microcontroller and Node-RED platform to process real-time temperature data transmitted via the MQTT protocol to a Raspberry Pi server. The DHT22 sensor measures the room temperature, and the data is sent to the ESP32 for processing and visualization by Node-RED. The Raspberry Pi serves as the control center, enabling temperature monitoring and control through a mobile application. The system automatically controls a fan based on room temperature, where the fan turns on when the temperature exceeds 30°C and turns off when the temperature drops below 30°C. Additionally, users can manually monitor the temperature and control the fan via the mobile application. Test results show that the system functions well, making it easier for users to control the fan.

INTRODUCTION

Indonesia is a relatively hot country due to its location along the equator, resulting in two seasons: the dry season and the rainy season. In addition to weather factors, room temperature also increases based on the number of people in the room. When the number of occupants exceeds the room's capacity, proper air circulation is needed to prevent the room from becoming uncomfortably hot (Adiyoga, A., 2023). Therefore, fans are needed in homes to lower room temperature and achieve an ideal indoor climate, as excessive heat can reduce productivity during activities. Currently, fans are mostly operated manually. When it feels hot, the fan is turned on by pressing or turning a switch. Fans are often left running because people forget to turn them off. Fans contain electronic components such as a motor to move the blades, and if left running continuously, these components can wear out quickly. High mobility often causes people to neglect or forget to turn off fans due to fatigue from daily activities. This necessity has supported the emergence of a new trend among millennials, namely the Internet of Things (IoT) (Adiyoga, A., 2023).

IoT represents one of the latest technological advancements of this century, rapidly evolving with internet connectivity that simplifies many daily activities (Nuramal, A., 2020). Kusumawardhani and Sudrajat (2019) emphasize that non-ideal indoor temperatures can affect human productivity and comfort, especially in tropical climates like Indonesia.

Given the explanation of these issues, this study aims to develop and build a prototype system for monitoring and controlling room temperature based on the Internet of Things (IoT) using Node-RED software as a control medium and Telegram for notifications. This system is expected to facilitate real-time temperature monitoring and control.

LITERATURE REVIEW

2.1 *Internet of Things* (IoT)

Internet of Things (IoT) is a concept that enables various devices to connect and communicate with each other through the internet. IoT offers the ability to connect devices, collect data, and allow real-time data transmission and analysis, playing a crucial role in remote monitoring and control systems (Gubbi et al., 2013). In its application to smart homes, IoT provides energy efficiency and occupant comfort by allowing devices to be automatically managed and controlled (Miorandi et al., 2012). Atzori et al. (2010) also stated that IoT can support applications that utilize real-time data to enhance productivity and comfort. Moreover, Hossain and Patwary (2021) demonstrated that using IoT in home automation systems can optimize energy management and enhance home security with the integration of smart devices.

2.2 Room Temperature Monitoring System

IoT-based room temperature monitoring has been widely researched and implemented by various scientists. Rahman et al. (2020) developed a temperature monitoring system using a DHT22 sensor connected to an ESP32 microcontroller, which accurately detects temperature and humidity. The data from this sensor is then transmitted to an IoT platform using the MQTT protocol, known for its lightweight and efficient characteristics for long-distance communication (Lema et al., 2017).

2.3 Automatic Temperature Control

Automatic temperature control is a vital feature in smart home systems, aimed at maintaining occupant comfort by automatically adjusting room temperature. Research by Chen et al. (2019) showed that using temperature sensors connected to an automatic control system allows devices like fans or air conditioners to be turned on or off based on specific temperature thresholds. This system is highly beneficial in keeping room temperature optimal without human intervention. Kumar and Singh (2020) added that although this system offers convenience, challenges such as device interoperability and data security must be addressed to achieve effective implementation.

2.4 Use of MQTT in IoT Systems

Message Queuing Telemetry Transport (MQTT) is a communication protocol often used in IoT applications due to its lightweight design and capability for real-time data transmission (Light, 2017). MQTT uses a publish/subscribe model, allowing IoT devices to communicate efficiently while reducing bandwidth consumption. Research by Jaramillo et al. (2019) showed that MQTT is highly suitable for IoT applications requiring continuous data communication, such as temperature monitoring systems. Gupta and Verma (2021) also emphasized that integrating MQTT with Node-RED provides a flexible solution for managing sensor data and automation. Zhao and Zhang (2019) further added that using MQTT in energy management can enhance efficiency by enabling real-time data analysis, thereby making temperature control more effective.

METHODS

3.1 Block Diagram

In creating a device, the block diagram is an essential aspect of the design and development process, making it easier to construct the device. The system's working principle involves monitoring and controlling room temperature. This IoT-based smart home system monitors and controls room temperature using the DHT22 sensor and the ESP32 microcontroller. The process begins with the DHT22 sensor, which measures the room temperature, and the collected data is sent to the ESP32 microcontroller. The ESP32 processes the received data and determines whether the room temperature has exceeded the set threshold. If the temperature exceeds the defined limit, the ESP32 sends a signal to the relay to activate the fan, thus allowing automatic temperature control.

Additionally, the ESP32 connects to a Wireless Access Point, enabling communication with Raspberry Pi and other platforms like Telegram. The Raspberry Pi further processes the data and displays temperature information through the Node-RED interface, which can be accessed using devices such as smartphones or laptops. Telegram is used to send real-time notifications to users about the room temperature status, allowing users to monitor room conditions anytime and anywhere. With this system, room temperature can be monitored and controlled automatically, making the room more comfortable and energy-efficient. The block diagram of the proposed device can be seen in Fig 1.

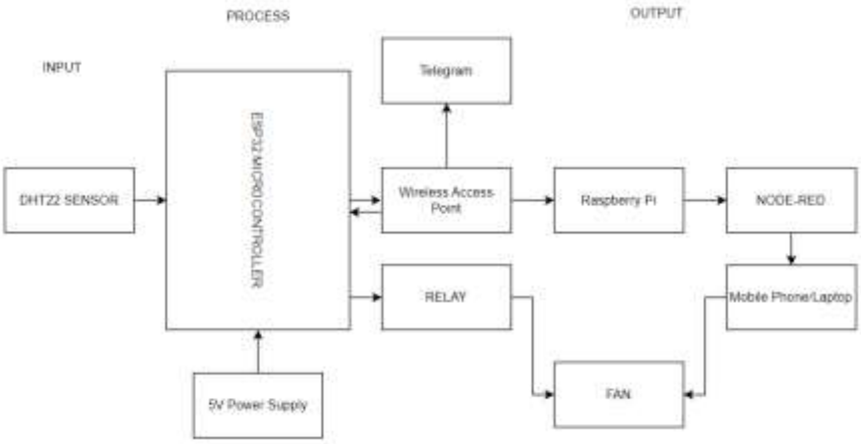

Fig 1. Block Diagram

3.2 *Flowchart*

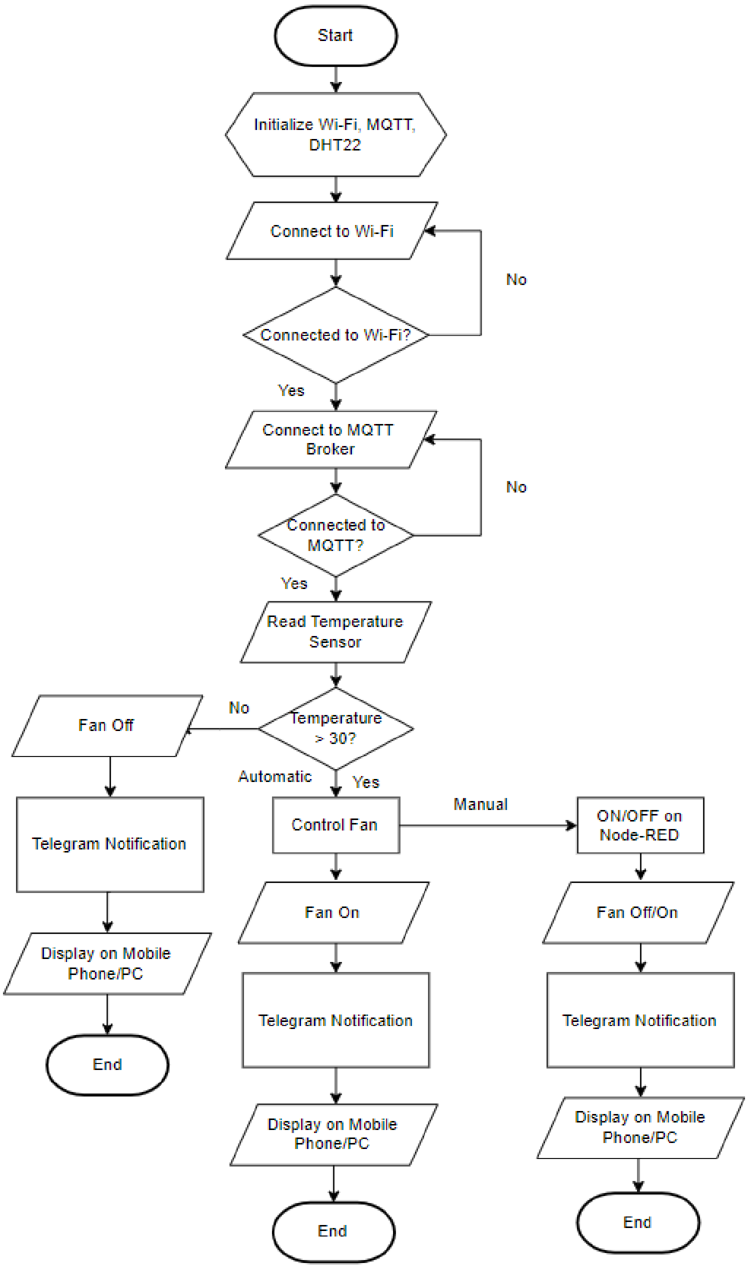

Fig 2. Flowchart

Figure 2 shows the flowchart of the device, which begins by initializing all the components, libraries, and sensors that will be used. The process starts with the initialization of Wi-Fi, MQTT, and the DHT22 sensor. The ESP32 then attempts to connect to the Wi-Fi network; if the connection fails, the system will stop until a successful connection is made. Once connected to Wi-Fi, the ESP32 connects to the MQTT broker for data exchange. If the connection to MQTT fails, the system will also stop until the connection is successful. After all connections are established, the ESP32 reads the temperature data from the DHT22 sensor.

The system then checks if the room temperature exceeds 30°C. If it does, the ESP32 will automatically turn on the fan and send a notification via Telegram, informing the user that the fan is on. This status can also be viewed on user devices like smartphones or PCs. In addition to the automatic mode, users have the option to manually control the fan through Node-RED, allowing them to set the fan status (ON/OFF) as desired. In both modes, whether automatic or manual, the system will continue to send notifications via Telegram about the fan's status. This information is also displayed on user devices, enabling easy real-time and remote monitoring of room temperature.

RESULT

4.1 Temperature Detection and Telegram Notification

In this study, the IoT system was tested for detecting room temperature and sending notifications via Telegram, which were then used to automatically control the fan. When the room temperature exceeds 30°C, the system sends a notification indicating that the temperature has crossed the threshold, and the fan automatically turns on.
Examples of Telegram notifications received during testing include:

- "Temperature exceeds 30°C: Fan ON."
- "Temperature drops below 30°C: Fan OFF."

4.2 Fan Control via Node-RED

This system is also equipped with a dashboard on the Node-RED platform that allows users to monitor room temperature and fan status in real-time, as well as manually control the fan if needed. Figure 1 shows the Node-RED dashboard used during system testing.

The dashboard features two main components:

- Fan Status: The status of the fan (On or Off) is visually displayed with colored icons. The fan automatically turns on when the room temperature exceeds 30°C and turns off when the temperature drops below this limit.
- Manual Control: Users can turn the fan on or off manually by pressing the buttons available on the dashboard, although the automatic system remains active.

Room temperature and fan status can be monitored directly through Node-RED, providing users with flexibility in managing the system. The data from the experimental results can be seen in Table 1.

Table 1. Results of Fan Control Experiment

No. Percobaan	Deteksi Suhu ruang (°C)	Notifikasi *Telegram*	Kipas angin
Percobaan 1	Suhu 28°C	Suhu dibawah 30° Kipas angin MATI	MATI
Percobaan 2	Suhu 29°C	Suhu dibawah 30° Kipas angin MATI	MATI
Percobaan 3	Suhu 30°C	Suhu dibawah 30° Kipas angin MATI	MATI
Percobaan 4	Suhu 31°C	Suhu melebihi 30° kipas angin MENYALA	MENYALA
Percobaan 5	Suhu 33°C	Suhu melebihi 30° kipas angin MENYALA	MENYALA

The results from five experiments showed that the fan remained in the OFF state when the temperature was below 30°C, while the fan automatically turned ON when the temperature exceeded the 30°C threshold. The display of the fan status and manual control on the Node-RED dashboard can be seen in Fig 3.

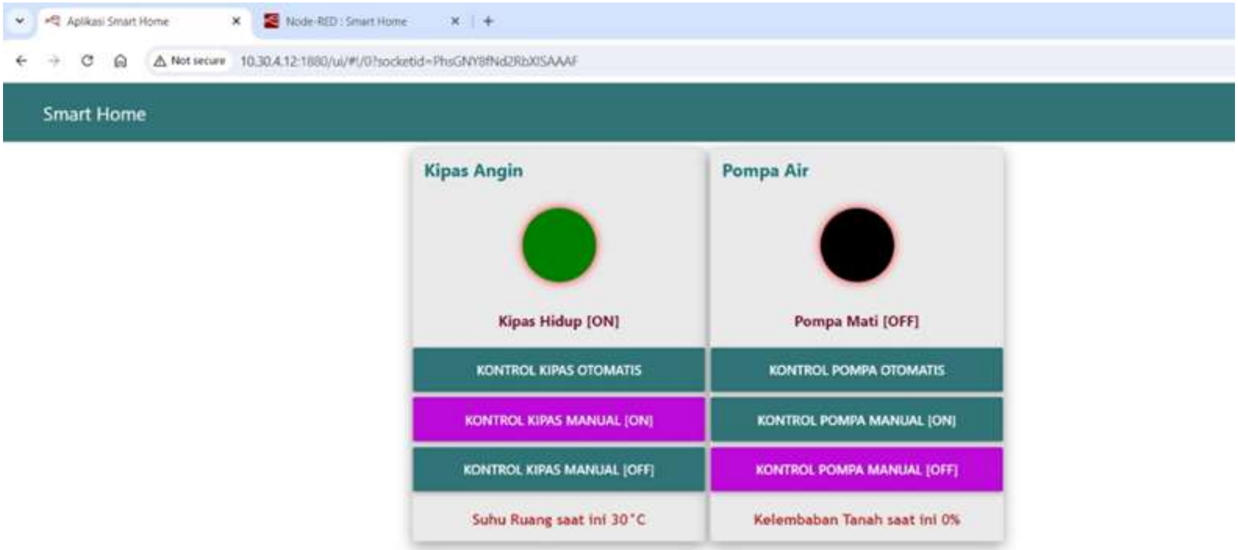

Fig 3. Node-RED Dashboard Display for Monitoring and Controlling the Fan.

DISCUSSION

Various tests and measurements have been carried out on both the hardware and software of this IoT-based smart home system. The results align with the initial design, where the DHT22 sensor successfully monitors the room temperature with good accuracy. This sensor provides accurate input to the ESP32, which then processes the data and automatically activates the fan when the room temperature reaches 30°C.

In the first test, the room temperature was increased to 30°C, and the system automatically turned on the fan. This is shown in Fig 4, where the Node-RED dashboard indicates "Fan On [ON]." Additionally, a notification was sent to the Telegram app with the message "Current Room Temperature 30°C, Fan On," ensuring that users always receive the latest updates on the room's condition and fan status.

Fig 4. Node-Red Dashboard Output Display on First Test

Subsequently, testing was conducted to manually activate the fan through the Node-RED dashboard, even though the room temperature had not yet reached 30°C. The results of this test demonstrated that the system can be controlled manually with ease, as shown in Fig 5. The fan was successfully turned on manually, and the dashboard indicator displayed the status "Fan On [ON]."

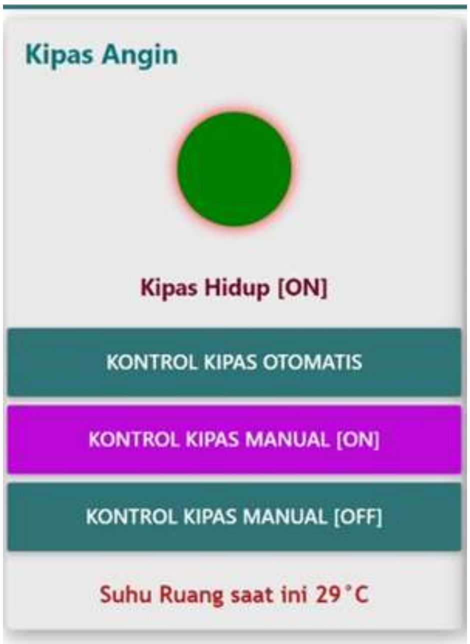

Fig 5. Node-Red Dashboard Output Display in Manual Testing ON

In the third test, the fan was turned off manually even though the room temperature had exceeded 30°C. Fig 6 shows the output display on the Node-RED dashboard when the fan was manually turned off, where the indicator changed to "Fan Off [OFF]." This demonstrates that the system provides flexibility in control, both automatically and manually.

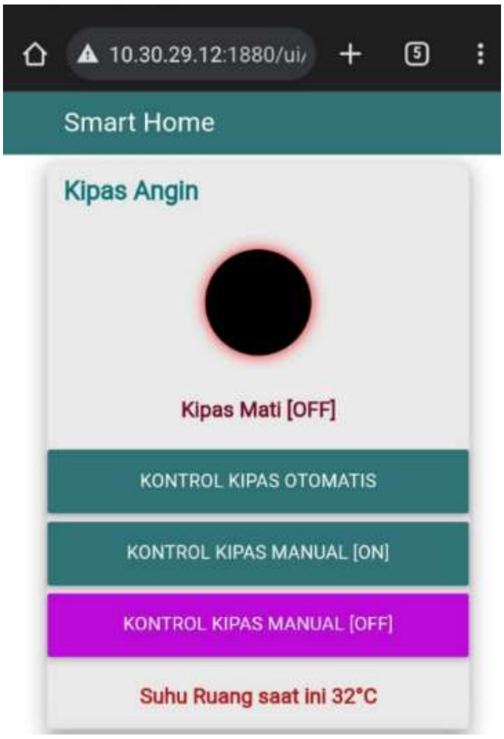

Fig. 6 Node-Red Dashboard Output Display on Manual Testing OFF

In addition to manual and automatic control, notifications from the system are also sent to Telegram in a timely manner. Fig 7 shows the notifications received in the Telegram app when the fan is turned on or off, whether automatically or manually. This confirms that the notification system functions effectively and responsively, which is one of the advantages of using the MQTT protocol.

Fig 7. Telegram Notification Testing

Although the test results indicate good system performance, there are several limitations that need to be considered. First, the system is highly dependent on internet connectivity. If the internet connection is unstable, communication between the ESP32 and Node-RED, as well as the sending of notifications to Telegram, may experience delays. Another limitation is that the testing environment is restricted to one room under ideal conditions, so implementing this system in more extreme environments requires further testing.

CONCLUSION

The IoT-based system designed using the DHT22 sensor and ESP32 module successfully monitors and controls room temperature in real-time via the Node-Red dashboard, where the Raspberry Pi functions as a gateway to connect all components. This system automatically activates the fan when the room temperature exceeds the threshold of 30°C, aiming to maintain temperature stability in the room. Apart from that, this system also successfully sends real-time notifications to users via the Telegram platform, ensuring users always have the latest information about temperature conditions. Performance measurements carried out show that the fan works effectively in responding to temperature changes, so this system can be considered efficient and reliable in automatically monitoring and controlling room temperature. This integration of IoT technology provides an innovative solution to create a comfortable and controlled environment.

REFERENCES

Adiyoga, A. (2023). Sistem kipas angin otomatis dengan sensor suhu dan sensor ultrasonik berbasis Arduino. Jawa Tengah. Retrieved July 12, 2024, from https://researchgate.net/publication/370460072_Sistem_Kipas_Angin_Otomatis_Dengan_Sensor_Suhu_dan_Sensor_Ultrasonik_Berbasis_Arduino.

Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. Computer Networks, 54(15), 2787-2805. https://doi.org/10.1016/j.comnet.2010.05.010

Chen, T., Chen, Y., Chen, C., & Huang, P. (2019). Intelligent air conditioning system for smart home with temperature monitoring and automatic control. Sensors, 19(17), 3797. https://doi.org/10.3390/s19173797

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645-1660. https://doi.org/10.1016/j.future.2013.01.010

Gupta, P., & Verma, P. (2021). IoT-based real-time temperature and humidity monitoring system using MQTT and Node-RED. International Journal of Advanced Research in Computer and Communication Engineering, 10(3), 50-55. https://doi.org/10.17148/IJARCCE.2021.10310

Hossain, M. S., & Patwary, M. N. (2021). Design and implementation of IoT-based smart home automation system. Journal of Network and Computer Applications, 175, 102877. https://doi.org/10.1016/j.jnca.2020.102877.

Jaramillo, A., Mendieta, J., & Contreras, J. (2019). MQTT-based smart home system for real-time data monitoring. IEEE Latin America Transactions, 17(12), 1975-1980. https://doi.org/10.1109/TLA.2019.8964703

Kumar, A., & Singh, P. (2020). A survey on Internet of Things: Applications and challenges. International Journal of Computer Applications, 975, 39-45. https://doi.org/10.5120/ijca2020920697.

Kusumawardhani, C. S., & Sudrajat, S. (2019). The influence of room temperature on the comfort and productivity of workers in a tropical climate. International Journal of Energy Economics and Policy, 9(4), 123-129. https://doi.org/10.32479/ijeep.7461

Lema, D., Yatnalkar, A., & Patel, M. (2017). Lightweight MQTT for IoT applications in energy management. Proceedings of the IEEE 3rd World Forum on Internet of Things (WF-IoT), 845-850. https://doi.org/10.1109/WF-IoT.2017.8355493

Light, R. (2017). Mosquitto: Server and client implementation of the MQTT protocol. Journal of Open Source Software, 2(13), 265. https://doi.org/10.21105/joss.00265

Miorandi, D., Sicari, S., De Pellegrini, F., & Chlamtac, I. (2012). Internet of Things: Vision, applications and research challenges. Ad Hoc Networks, 10(7), 1497-1516. https://doi.org/10.1016/j.adhoc.2012.02.016

Nuramal, A. (2020). Pemanfaatan Internet of Things untuk monitoring suhu di BPPT-MEPPO. Bengkulu. https://www.scribd.com/document/740438050/13274-Article-Text-28260-32818-10-20201027.

Rahman, M., Alam, K., & Khan, M. (2020). IoT-based temperature and humidity monitoring system using ESP32 and DHT22. International Journal of Computer Science and Network Security, 20(6), 62-67.

Zhao, Y., & Zhang, W. (2019). The role of IoT in energy management systems: A case study of smart buildings. Energy Reports, 5, 123-131. https://doi.org/10.1016/j.egyr.2019.09.002.