MQTT Protocol-Based ESP-32 Smarthome with Multi-sensor Recognition

1st Febriansyah Dwicahya Makatita
Department of Electrical Eng. Education
Universitas Pendidikan Indonesia
Bandung, Indonesia
febrimakatita@upi.edu

2nd Nurul Fahmi Arief Hakim
Department of Electrical Eng. Education
Universitas Pendidikan Indonesia
Bandung, Indonesia
nurulfahmi@upi.edu

*Corresponding author: nurulfahmi@upi.edu
Received: January 27, 2024; **Accepted:** May 17, 2024

***Abstract*—Rapid technological advancements are accompanied by a progressive increase in sophistication. Nonetheless, an issue that must not be overlooked in light of this progress pertains to the energy sources employed. One possible resolution to this issue involves the implementation of an automation system within the domestic setting. Therefore, the objective of this study was to develop an automation system for use in a smarthome or residential setting. In addition to this, the objective of this study is to assess the efficacy of the smarthome application and streamline the system in comparison to prior investigations. Following a literature assessment, the research methodology consists of system design, system development, and system testing. Based on the conducted research, it was determined that the temperature sensor employed exhibited a commendable accuracy rate of 98.12%. Positive outcomes were also obtained through the implementation of the MQTT protocol in this study; the MQTT Dashboard application exhibited an average delay of 0.725 seconds, while the node-red application demonstrated a delay of 0.67 seconds. It can be concluded from these results that the designed and implemented system functions as intended. Additionally, these results indicate that the implementation of a smarthome system can streamline the management of routinely utilized electronic devices. In addition to this, the implementation of the sensors yields favorable outcomes, thereby establishing their dependability for the regulation of automated systems. With any luck, this study will offer a comprehensive understanding of the potential of smarthomes as a sustainable alternative that will inspire greater attention to energy conservation in the home environment.**

***Keywords*— Smarthome, ESP32, IoT, MQTT**

I. Introduction

Rapid technological advancements have a profound impact on numerous facets of existence. Nevertheless, in light of the swift advancements in technology, an additional factor that warrants consideration is the electrical energy source, which serves as the power supply for the aforementioned technologies. Indonesia's documented electricity consumption for the year 2022 amounts to 83,813.09 MW. PLN provides 42,495.10 MW of this total, while non-PLN suppliers (IO, EBTKE, IPP, PPU) contribute 41,317.99 MW. This value increased by approximately 11.07% compared to the previous year [1]. It is evident from the aforementioned data that PLN is the primary provider of electricity; therefore, it merits special consideration, given that the majority of its electricity is generated from coal [1]. Given the inherent finite nature of coal as a fuel, it is evident that this energy source is not suitable for extended periods of reliance. Moreover, the utilization of coal contributes to environmental degradation by generating carbon emissions, which pose significant risks to the surrounding ecosystem.

One potential resolution to this issue involves the implementation of automation on readily accessible and frequently utilized electronic devices, including but not limited to lighting, fans, and similar devices. By implementing an automation system on these devices, it is possible to prevent inefficient operation, such as neglecting to deactivate the lights during the day or operating the fan while the room temperature remains at a comfortable level [2].

The application of automation principles to the residence is referred to as "smarthome." The level of interaction between occupants and systems is minimized in a smarthome [3]. Internet of Things (IoT) technology is also incorporated into smarthome systems; this facilitates human control of electronic devices by allowing for far-reaching manipulation [4]. IoT is a technology that enables devices in an integrated system to communicate via the internet. Therefore, the two devices are capable of communicating so long as both the control and controlled devices have an internet connection.

Prior studies investigating the implementation of IoT in smart homes through the utilization of MySQL as an intermediary demonstrated a commendable application in regulating the power state of the lighting [5]. In addition, blynk has been employed in prior investigations as a mediator between systems and the internet [6]. The MIT App Inventor, an intermediary application utilized in prior investigations, has been purposefully developed for installation on Android devices as well [7]. Prior studies have also implemented bot functionalities within the Telegram platform [8]. The web has been utilized in prior research to facilitate the reading and monitoring process, thereby offering consumers an enhanced user experience [9]-[11].

Prior studies have established that the potential for the development of smarthomes is substantial. Utilizing multi-sensor recognition and the MQTT communication protocol are two methods for enhancing the control of electronic devices. The utilization of the MQTT protocol is justified by the relatively low volume of internet traffic, which effectively mitigates the design latency time of the system. Additionally, the research will involve the integration of multiple sensors into the system it designs. These sensors include the KY-038 sensor for sound detection, the Light Dependent Resistance (LDR) sensor for light monitoring, and the DHT22 sensor for temperature control. In the interim, the apparatus under control consists of fans and lighting that are managed through a 4-channel Relay Module. The article highlights the ESP32's versatility and MQTT's efficacy in creating a scalable, efficient smart home ecosystem.

II. Methods

A. System architecture

At this point, the process of designing the hardware and software components of the system under study commences. As shown in Figure 1, the block diagram of the complete system will be constructed. The block diagram delineates the system into three distinct processes: input, processing, and output. The input section comprises three distinct categories of sensors: light sensors, sound sensors, and temperature sensors. The light sensor employed is an LDR sensor, which measures the light intensity in the environment under examination. Additionally, the LDR sensor is a category of sensors whose resistance can vary in response to light or conditions such as darkness [12]. The sound sensor employed is the KY-038 sensor, which is designed to identify acclaim in the testing environment. Receiving sound, the KY-038 sensor will deliver voltage feedback. This sensor is capable of transmitting a maximum of 3.5 volts in response to a powerful sound [13]. In addition, a temperature sensor is present to ascertain the ambient temperature of the chamber undergoing examination. The utilized sensor is the DHT-22, which is said to provide 18% more accurate humidity and 4% more precise temperature measurements than its predecessor, the DHT-22 [14].

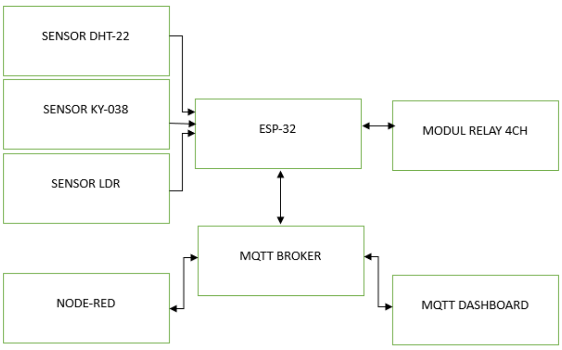

Fig 1. System block diagram

The concept of the IoT is illustrated in Figure 2. An ESP-32 microcontroller is implemented in the process section; it processes sensor input before forwarding it to the actuator and the internet. Espressif manufactures the ESP-32 microcontroller, which is equipped with Wi-Fi and Bluetooth modules to enable wireless internet connections. The IoT concept can be implemented when the system is capable of being connected to the internet. The IoT, which combines internet network connectivity with perpetually connected machines, devices, or other physical objects, acquires and processes data in real time for subsequent processing and execution [15]. Put simply, IoT refers to a control paradigm wherein the internet serves as an intermediary platform facilitating real-time communication among various devices.

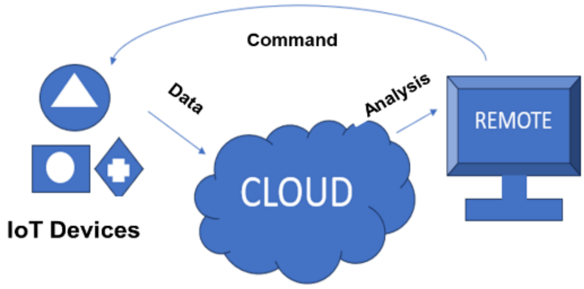

Fig 2. The Internet of Things conception

This investigation makes use of the MQTT (Message Queuing Telemetry Transport) communication protocol. A graphical representation of the MQTT concept is presented in Figure 3. The MQTT communication protocol is a Machine to Machine (M2M) communication protocol comprised of the following components: topic, subscriber, publisher, broker, and client. The publisher is a device element that functions as a message sender to the merchant regarding topics arranged by the publisher. Brokers serve as intermediaries or providers of data that publishers and subscribers have the ability to access. A subscriber, conversely, is a device that utilizes a broker to access or read data pertaining to a pre-established subject matter as supplied by the publisher. Clients are devices that possess the capability to function as both publishers and subscribers. The speed at which data is processed and the simplicity of integrating additional devices that utilize this protocol for communication are two notable benefits of this protocol [16].

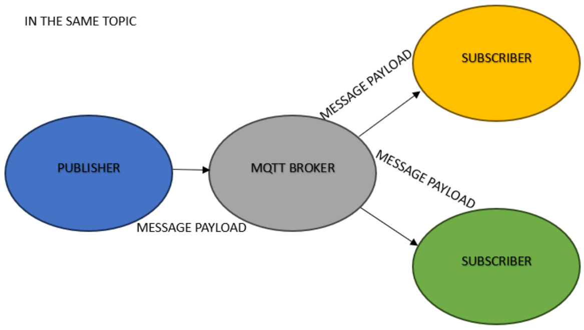

Fig 3. MQTT communication protocol concept

B. Maintaining the Integrity of the Specifications

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

The software utilized in this study as a client is Node-red and MQTT Dashboard. MQTT Dashboard is an Android application that provides users with the capability to both subscribe to and publish MQTT topics. This application is utilized to enable the monitoring and control processes to be executed on an Android smartphone, thereby streamlining the process. As a web server, Node-red is a platform utilized to retrieve and display data. The node-red platform has the capability to conduct monitoring through the implementation of various supplied interface components [17]. Thus, the process of monitoring the system can be executed with relative ease. An actuator is comprised of a four-channel relay module located in the output section. The purpose of this relay is to regulate control devices, including fans and lighting. By utilizing an electromagnetic principle to move the switch contacts, relays enable higher voltage electricity to be controlled by low voltage electricity [18].

In order to facilitate the programming process, a programming flow or flowchart is generated once the design is complete. Figure 4 depicts the flowchart that was designed. Initially, the microcontroller will attempt to establish a connection to the Wi-Fi address specified in the program. The illumination of the Wi-Fi indicator LED occurs when the microcontroller is connected. In the absence of a connection, the LED will flicker until the microcontroller is linked. Following its Wi-Fi connection, the microcontroller will attempt to establish a connection with the MQTT broker that was additionally specified in the program. Once the device has established a connection with the MQTT broker, the MQTT indicator LED will illuminate; in the absence of a connection, the LED will deactivate and the connection attempt will resume. Once the microcontroller has established connections with the Wi-Fi and MQTT brokers, the sensor reading data obtained from the microcontroller will be transmitted to the MQTT broker. This enables the user to access the sensor readings through the MQTT client, which is concurrently connected to the same broker as the microcontroller. The subsequent phase involves the user entering commands to activate the automatic and manual modes. Activating auto mode grants complete control of the actuator to sensor readings; conversely, activating manual mode grants complete control of the actuator to the user, with the sensor serving solely as a monitoring device.

In automatic mode, sensor values are used for both monitoring and controlling. When the light sensor detects darkness, the microcontroller activates the lights via a relay. If the light sensor detects illumination, it deactivates the lights. Similarly, the temperature sensor provides data to the microcontroller, which activates the fan when the room is hot and deactivates it when the room is chilly, both through relay commands. Upon the user's instruction to enter manual mode, the sensor's value will exclusively serve the purpose of monitoring. The actuator's control process shall be exclusively established through user input. User input is derived from two distinct sources: applause sounds and virtual buttons integrated within the MQTT interface application. Before offering applause as input, one must initially engage clap mode. As soon as applaud mode is activated, the sound sensor will record the user's sound input. A command to activate the lamp will be transmitted by the microcontroller through a relay if the count of acclaim interpreted is an odd number. Conversely, in the event that an even number of applause signals are detected, the microcontroller will transmit a deactivation command to the lights through a relay.

In the absence of clap mode and while the system is operating in manual mode, the control process will be wholly driven by user input via the virtual buttons displayed on the MQTT dashboard. The payload format comprises the command form displayed on the MQTT Dashboard. When the user issues a command to activate a device, the payload transmitted contains the value 1. If, on the other hand, the user instructs the device to be turned off, the payload transmitted will contain zero. The implementation of this logic is intended to decrease the data capacity or throughput in order to mitigate the potential for delays. which requires a considerable amount of effort. In response to a user's instruction to activate the light, the microcontroller will receive the message and activate the light via a relay. Conversely, in the event that the user instructs the lamp to cease operation, the microcontroller shall perceive the communication and transmit the directive to the lamp by means of a relay. The same applies to the ventilation control procedure. The microcontroller will receive the user's command to activate the fan and, upon receiving the message, will activate the fan via a relay. Conversely, in response to the user's instruction to deactivate the fan, the microcontroller will intercept the message and transmit the directive to the fan via a relay.

C. Testing of systems

The assessment procedure will be executed from two perspectives: software and hardware. In general, the functionality of all systems will be verified by utilizing the MQTT Dashboard application on an Android smartphone to attempt to turn the lights on and off. The following parameters will be taken into account: latency time, reading accuracy level, and the system's utilization of an IP address from the internet.

1.) Hardware verification

Hardware testing is conducted in the following phases:

- Verify that the microcontroller is capable of connecting to the provided Wi-Fi and operating normally.
- Turn it on and verify that the LEDs function as intended.
- Evaluate the temperature sensor readings in comparison to sensors of industrial grade.
- Observe the sensor reading value in response to applause.
- Examine the light sensor reading value in various weather conditions, including nighttime, cloudy, sunny, and overcast.
- Determine if the relay alters in response to the microcontroller's instruction.

2.) Software verification:

Software testing is also conducted in the following phases:

- Observe the sensor reading values transmitted to MQTT by the microcontroller.
- Employ the MQTT Dashboard to transmit commands to the microcontroller and observe the elapsed time between the transmission of the application and its reception from the microcontroller.

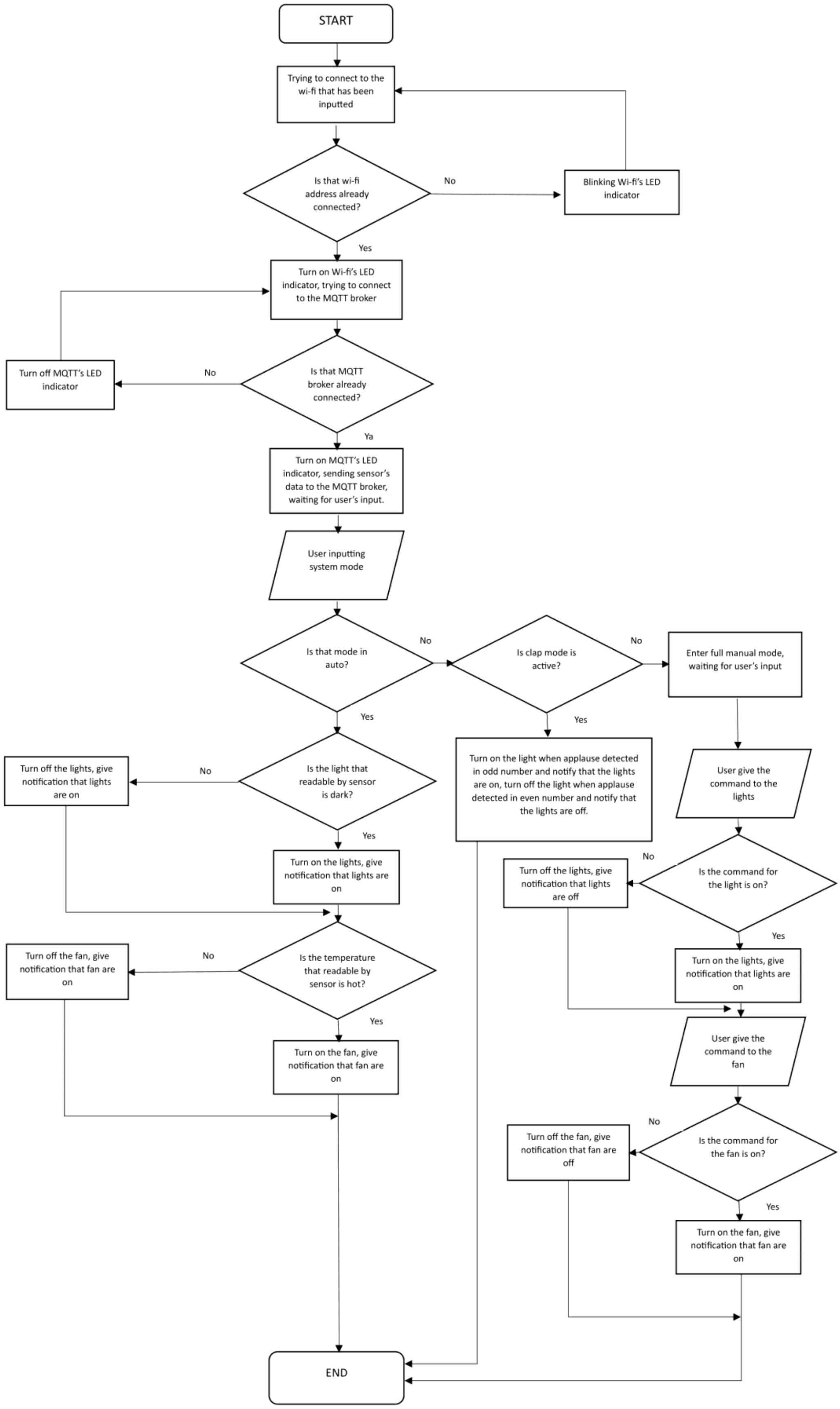

Fig 4**.** Flowchart programming system

III. Results and discussion

Illustrative instances of the device being implemented on a miniature dwelling will be presented in Figure 6 and Figure 7. The findings presented in this study are data obtained from the direct testing of the designed system, encompassing both hardware and software components. Based on the acquired data, an evaluation can be conducted to determine whether the designed system operates as anticipated. In order to derive more definitive conclusions, the analysis procedure will make minimal modifications to previously cited research. The devised devices will be illustrated in Figure 5. Prototypes of the designed devices are tested immediately on a miniature home so that their functionality can be observed firsthand.

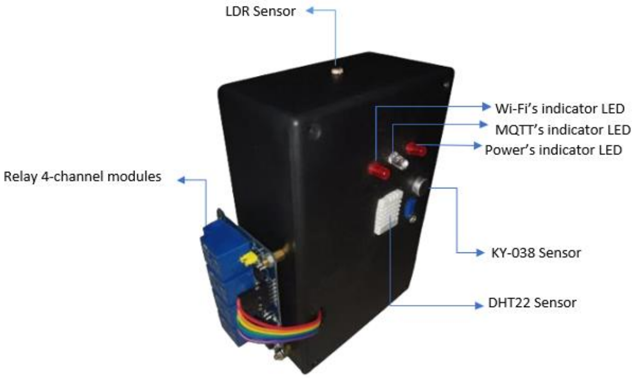

Fig. 5. System devices

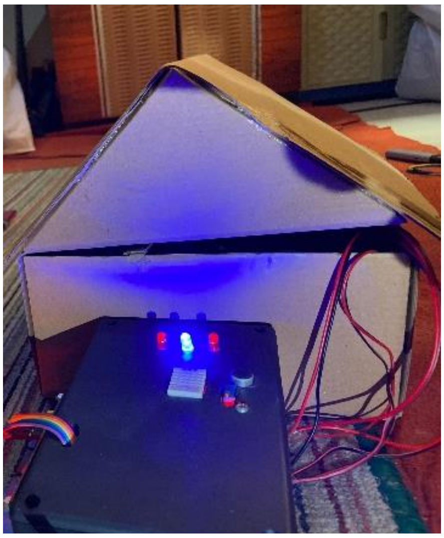

Fig. 6. Miniature house

Fig. 7. Validation the device on a miniature

A. *Hardware evaluation*

The outcomes derived from the evaluation of the hardware integrated within the system design are outlined below.

1. *LDR sensor*

The LDR sensor was evaluated in four locations, with three experiments conducted in each location condition. The four locations under consideration are nighttime (dark), cloudy, sunny, and overcast. The outcomes of the tests are shown in Table 1. The acquired data comprises the sensor readings and the test conditions. Equation (1) will be utilized to convert this data to lumen units. The outcomes derived from the evaluation of the hardware integrated within the system design are outlined below. The value 1504 is displayed on the ADC reference after calibration with a reference value of 1350 lumens. The lumen units are utilized to convert the LDR sensor readings via the aforementioned equation.

$$Lumen = \frac{reference\ value}{ADC\ reference}\ x\ sensor\ output \quad (1)$$

TABLE I. A Record of LDR Sensor Readings

No	Conditions	Value sensor (test 1)	Value sensor (test 2)	Value sensor (test3)
1	sunny	1729.8 lm	1730.5 lm	1750.3 lm
2	cloudy	1434.3 lm	1437 lm	1431.6 lm
3	overcast	78 lm	71.8 lm	69.1 lm
4	nighttime (dark)	26.9 lm	25.1 lm	31.4 lm

2. *LDR sensor*

The LDR sensor was evaluated in four locations, with three experiments conducted in each location condition. The four locations under consideration are nighttime (dark), cloudy, sunny, and overcast. The outcomes of the tests are shown in Table 1. The acquired data comprises the sensor readings and the test conditions. Equation (1) will be utilized to convert this data to lumen units.

3. *KY-038 Sensor*

Input for testing the KY-038 sensor consisted of applauding sounds, and the corresponding readings were observed on the serial monitor integrated within the Arduino IDE. Four repetitions of the experiment yielded the results displayed in Figure 8-10. As shown in Figure 8, the highest possible reading value is 90 when acclaim is heard. The utmost recorded value in Figure 9, elicited by applause, is 87. The maximum reading value in Figure 10, which corresponds to the sound of acclaim, is 117. Therefore, it can be deduced that the KY-038 sensor will yield a reading value ranging from 80 to 120 when calibrated to detect acclaim.

Fig. 8. The first stimulus for the sound sensor was applauding.

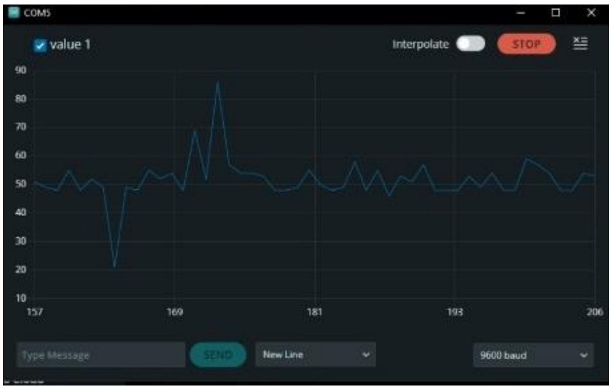

Fig. 9. The second stimulus for the sound sensor was applauding.

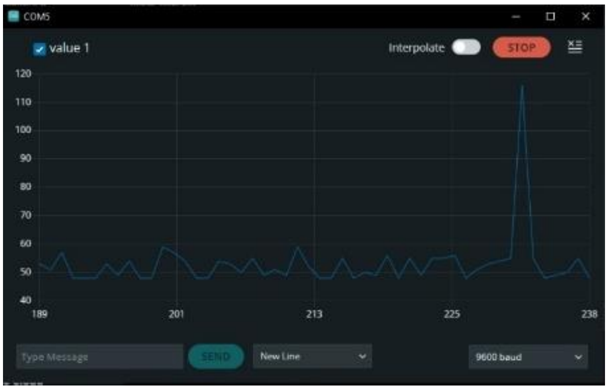

Fig. 10. The third stimulus for the sound sensor was applauding

B. *Software validation*

The process of software testing involves the interconnection of two applications. This experiment was conducted under two distinct conditions. In the initial condition, both the control device and the controlled device are assigned identical Wi-Fi IP addresses. Subsequently, under the second condition, these two devices are assigned distinct Wi-Fi IP addresses.

1. *MQTT Dashboard*

The process of verifying the MQTT Dashboard involves attempting to transmit messages to the microcontroller from the MQTT Dashboard. The microcontroller then presents the sensor measurement results on the MQTT Dashboard. The outcomes of the tests are detailed in Tables 3 and 4. Ten trials were conducted for each condition during the testing phase. Consider the latency time when conducting this MQTT test via two distinct Wi-Fi IP addresses. The experiments conducted indicate that the data transmitted by the control device was error-free and arrived at a relatively brief delay of 0.726 seconds for an alternative IP address compared to 0.732 seconds for the same IP address. The MQTT Dashboard application is visually represented in Figure 11 and Figure 12. A display of lamp conditions and two switches, labeled "on" and "off," are present on lamps 1 and 2. Similar to fan 1, fan 2 also features a condition display and two switches labeled "on" and "off." The monitoring tab displays three distinct weather conditions pertaining to the vicinity of the house: sunny, cloudy, and gloomy. In addition to the aforementioned, a temperature reading in degrees Celsius is also presented. Aside from the clap mode button, the all tiles tab features auto and manual controls for adjusting the system mode, as well as a clap mode button for entering clap mode.

TABLE II. MQTT DASHBOARD TESTING BETWEEN THE CONTROLLED AND CONTROL DEVICES USING THE SAME WI-FI IP ADDRESS

No	Controlled device	Condition	Delay value 1	Delay value 2
1	Auto/manual button	Good	1,09 second	0,9 second
2	Lamp 1	Good	0,53 second	0,61 second
3	Lamp 2	Good	0,57 second	0,57 second
4	Fan 1	Good	0,62 second	0,66 second
5	Fan 2	Good	0,9 second	0,8 second

TABLE III. MQTT DASHBOARD TESTING BETWEEN THE CONTROLLED AND CONTROL DEVICES USING DIFFERENT WI-FI IP ADDRESS

No	Controlled device	Condition	Delay value 1	Delay value 2
1	Auto/manual button	Good	0.89 second	0.95 second
2	Lamp 1	Good	0.56 second	0.6 second
3	Lamp 2	Good	0.58 second	0.79 second
4	Fan 1	Good	0.53 second	0.56 second
5	Fan 2	Good	0.7 second	0.7 second

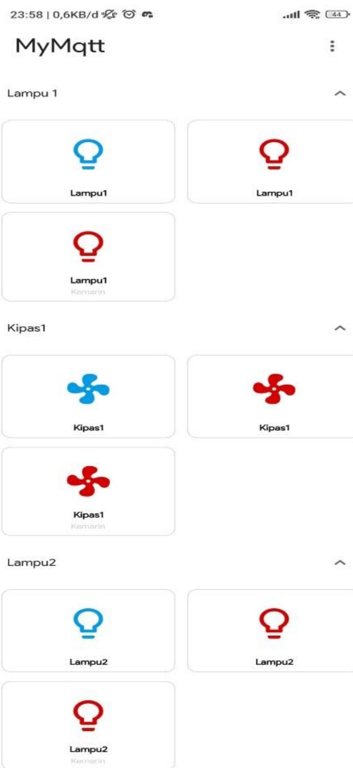

Fig. 11. MQTT dashboard main view for lamp and fan

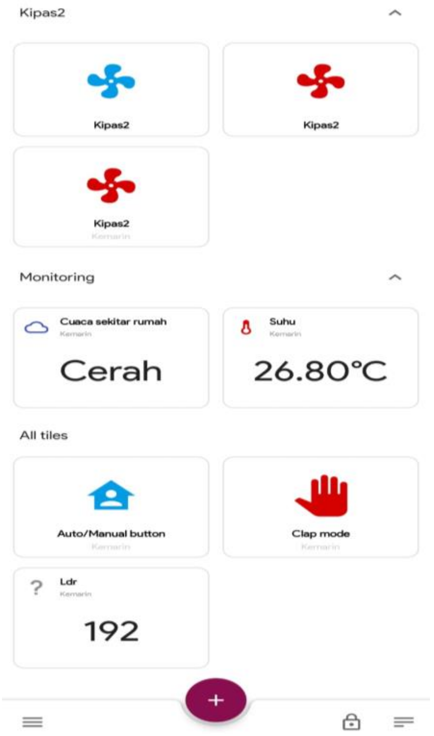

Fig. 12. MQTT dashboard main view for weather, button, LDR, and Clap mode

2. Node-red validation

The sensor reading results from the microcontroller are retrieved for testing purposes on node-red. These results are subsequently presented on the node-red webserver. Considerable research effort is devoted to investigating the latency time, as well as the Wi-Fi IP address of both the control and controlled devices. The outcomes of the conducted experiments are presented in Tables 5 and 6. For each condition, eight trials were conducted as trials. The presented data consists of the controlled device, the status of the data transmission (success or failure), and the duration of the delay that transpired. The tests conducted indicated that the data transmitted by the control device was error-free and arrived at its destination with a relatively brief delay. On average, the delay for devices using the same IP address was 0.82 seconds, while for devices using separate IP addresses, it was 0.67 seconds.

Illustrate a visual representation of the Node-red application in Figure 13. The household condition monitoring tab displays the current state of every controlled device. The controlled devices comprise the light 1, light 2, fan 1, and fan 2 conditions. Sensor readings are displayed on the weather monitoring tab. The sensor readings encompass both the ambient temperature and the weather conditions, which are categorized as follows: sunny, cloudy, or gloomy.

TABLE IV. NODE-RED TESTING BETWEEN THE CONTROLLED AND CONTROL DEVICES USING THE IDENTICAL WI-FI IP ADDRESS

No	Controlled device	Condition	Delay value 1	Delay value 2
1	Lamp 1	Good	0.56 second	0.52 second
2	Lamp 2	Good	0.72 second	0.55 second
3	Fan 1	Good	0.67 second	0.78 second
4	Fan 2	Good	1.2 second	1 second

TABLE V. NODE-RED TESTING BETWEEN THE CONTROLLED AND CONTROL DEVICES USING THE IDENTICAL WI-FI IP ADDRESS

No	Controlled device	Condition	Delay value 1	Delay value 2
1	Lamp 1	Good	0.67 second	0.62 second
2	Lamp 2	Good	0.76 second	0.89 second
3	Fan 1	Good	0.62 second	0.8 second
4	Fan 2	Good	0.8 second	0.76 second

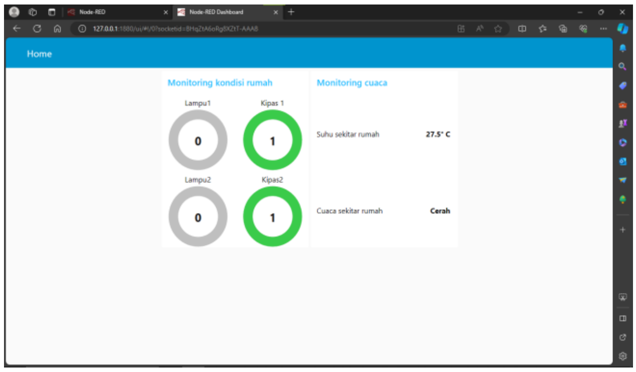

Fig. 13. Node-red dashboard

C. *Analysis*

As determined by verifying the system on the device, the temperature sensor exhibited an error rate of 1.98%, corresponding to an accuracy of 98.12%. It was subsequently determined, through testing, that the latency time between the controlled and control devices was not significantly impacted by the Wi-Fi IP addresses that were utilized. Subsequent to analysis, the determinants of delay time are data size and throughput, with delay being directly proportional to throughput [19]. In addition to this, the speed of the internet provider also exerts an impact, as illustrated in Tables 3, 4, 5, and 6. In the experiments conducted in Tables 3 and 5, the internet connection was supplied by the cellular card provider on the smartphone. Conversely, in the tests conducted in Tables 4 and 6, a reputable Wi-Fi router provider offering 5G speed provided the internet connection. The average delay time for Table 3 is 0.732 seconds; Table 4 reveals that the actual delay time is 0.726 seconds. The discrepancy in latency between the two tables amounts to 0.006 seconds. Table 5 presents the mean delay time as 0.82 seconds, whereas Table 6 presents the mean delay time as 0.67 seconds. The discrepancy between the two tables amounts to 0.15 seconds. Connection delays involving the ESP32 microcontroller can result from hardware or software issues, network configuration, or signal strength. Distant routers or physical obstructions often weaken Wi-Fi signals, substantially hindering connection times. Network congestion, caused by numerous devices competing for bandwidth, also impedes connections. Outdated router configurations or restricted DHCP address pools contribute to delays, as do inefficiencies in the ESP32's firmware or connection code, and authentication delays on secure networks. Additionally, unstable or inadequate power supplies and channel interference from overlapping Wi-Fi signals can exacerbate these issues. Ensuring robust Wi-Fi signals, minimizing network congestion, maintaining updated firmware, and optimizing connection code can help mitigate these delays. However, significant latency can lead to packet loss or data failure, diminishing the system's overall efficacy. The findings of this study also serve as a supplement, simplification, and analysis of elements not previously addressed in [20], [21].

IV. Conclusion

The system functions as intended, with both the hardware and software operating efficiently. The accuracy of the DHT22 temperature readings transmitted was 98.12%. The average latency observed on an internet connection where both the control and controlled devices utilize the same Wi-Fi IP address is 0.732 seconds for the MQTT Dashboard application and 0.82 seconds for the node-red application. When an internet connection is established between the control device and the controlled device using a distinct Wi-Fi IP address, the MQTT Dashboard application experiences an average delay of 0.725 seconds, while the node-red application experiences a delay of 0.67. For the next step, the automation-based smarthome concept can be added with other sensors, such as infrared or RFID.

References

[1] Dirjen Gatrik "STATISTIK KETENAGALISTRIKAN TAHUN 2022", 2023 [Online]. Available: https://gatrik.esdm.go.id. Accessed on: Dec.31, 2023

[2] A. D. Fakhrudin, N. F. A. Hakim, and A. H. S. Budi, "Implementasi Protokol TCP dan UDP pada Sistem Monitoring dan Otomasi Rumah Jamur Berorientasi WSN," *TELKA - Telekomunikasi Elektronika Komputasi dan Kontrol*, vol. 9, no. 2, pp. 130–144, Nov. 2023.

[3] A. Natarajan, V. Krishnasamy, and M. Singh, "Occupancy detection and localization strategies for demand modulated appliance control in Internet of Things enabled home energy management system," *Renewable and Sustainable Energy Reviews*, vol. 167, p. 112731, Oct. 2022.

[4] S. Maghrisya, A. M. Ridwan, M. R. Effendi, N. F. Arief Hakim, R. P. Sari and R. Ramdani, "Prototype of Glucose Meter with Non-Invasive Method Based on IoT," *2023 9th International Conference on Wireless and Telematics (ICWT)*, Solo, Indonesia, 2023, pp. 1-5.

[5] J. A. Hassan and B. H. Jasim, "Design and implementation of internet of things-based electrical monitoring system," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 6, pp. 3052–3063, Dec. 2021.

[6] S. Garg, A. Yadav, S. Jamloki, A. Sadana, and K. Tharani, "IoT based home automation," *Journal of Information and Optimization Sciences*, vol. 41, no. 1, pp. 261–271, Jan. 2020.

[7] R. Damayanti and M. Mabe Parenreng, "Rancang Bangun Smart Home Berbasis Internet of Things," *JASENS*, vol. 1, no. 2, pp. 5-09, 2020.

[8] W. W. Anggoro and I. R. Widiasari, "Perancangan dan Penerapan Kendali Lampu Ruangan Berbasis IoT (Internet of Things) Android," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 3, pp. 1596-1606, 2021.

[9] S. R. Hidayati and I. Nirmala, "Smart Lamp: Kendali dan Monitor lampu Berbasis Internet of Things," *JUPITER*, vol. 14, no. 2, pp. 507-515, 2022.

[10] D. Susilo, C. Sari and G. w. Krisna, "Sistem Kendali Lampu pada Smart Home Berbasis IoT (Internet of Things)," *ELECTRA: Electrical Engineering Articles*, vol. 2, no. 1, pp. 23-30, 2021.

[11] F. Z. Rachman, "SMART HOME BERBASIS IOT," SNITT, pp. 369-374, 2017.

[12] Md. W. Rahman et al., "The architectural design of smart blind assistant using IoT with deep learning paradigm," *Internet of Things*, vol. 13, p. 100344, Mar. 2021.

[13] N. A. Zen, S. Indriyanto, I. Permatasari, J. Sasmita, and L. Yuliantini, "Preliminary study of using KY-038 sensor based on Arduino Uno and labview to determine the pulse rate," *Jurnal Fisika dan Aplikasinya*, vol. 16, no. 2, p. 87, 2020.

[14] Y. A. Ahmad, T. Surya Gunawan, H. Mansor, B. A. Hamida, A. Fikri Hishamudin and F. Arifin, "On the Evaluation of DHT22 Temperature Sensor for IoT Application," *2021 8th International Conference on Computer and Communication Engineering (ICCCE)*, Kuala Lumpur, Malaysia, 2021.

[15] Zulfahmi Syahputa, M. Z. "Testing Real-Time Applications on Windows 10 IOT Using the Nyquist Theory". *Journal of Physics* , 012066, 2019.

[16] S. Wagle, ''Semantic data extraction over MQTT for IoTcentric wireless sensor networks,'' in *Proc. Int. Conf. Internet Things Appl.* (IOTA), pp. 227–232, 2016.

[17] Mulyono, S., Qomaruddin, M., & Syaiful Anwar, M., "Penggunaan Node-RED pada Sistem Monitoring dan Kontrol Green House berbasis Protokol MQTT" *Jurnal Transistor Elektro Dan Informatika (TRANSISTOR EI)*, vol. 3 no.1, pp. 31–44, 2018.

[18] R. B. Ghoderao, S. Raosaheb Balwe, P. S. Chobe, D. B. Pardeshi and P. William, "Smart Charging Station for Electric Vehicle with Different Topologies," 2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2022, pp. 243-246.

[19] Permatasari elza Fitria dwi, Putrada aji gautama, Abdurohman Maman, " Analisis perbandingan performansi MQTT dan HTTP pada platform Node-Red" *e-proceeding of engineering*, vol. 6 no.2, pp 8559-8569, 2019.

[20] K. Okokpujie, D. Jacinth, G. A. James, I. P. Okokpujie and A. A. Vincent, "An IoT-Based Multimodal Real-Time Home Control System for the Physically Challenged: Design and Implementation," *Information Dynamics and Applications*, vol. 2, no. 2, pp. 90-100, 2023.

[21] M. Li et al., "Smart home: Architecture, technologies and systems," *Procedia Computer Science*, vol. 131, pp. 393–400, 2018.