JOURNAL OF INFORMATICS DEVELOPMENT

E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

Monitoring System Indoor Mushroom Cultivation via Telegram Bot

Rio Ridho Sandikyawan[1], Samsul Arifin[2]

[12]Computer Engineering, Faculty of Technology and Design, ASIA Institute of Technology and Business, Malang
Jln. Soekarno - Hatta – Rembuksari No. 1A, Malang, East Java, 65141

Corresponding Author: Rio Ridho Sandikyawan (rioridho.s@gmail.com)

ARTICLE INFO

Date of entry:
14 April 2024
Revision Date:
25 April 2024
Date Received:
30 April 2024

ABSTRACT

Mushroom cultivation in Indonesia has significant potential due to its tropical climate, which is ideal for growing various types of mushrooms. However, maintaining optimal environmental conditions, such as temperature and humidity, is crucial for successful cultivation. This study aims to design and develop an Internet of Things (IoT)-based monitoring system for indoor mushroom farming, utilizing NodeMCU and Telegram Bot for real-time data management. IoT is used in mushroom cultivation to monitor and manage environmental conditions in real-time, considering that mushroom growth is very sensitive to changes in temperature and humidity. The main challenges faced are ensuring the stability of environmental conditions and reducing communication delays in data transmission to maintain the quality and quantity of mushroom production. The system employs a DHT11 sensor connected to a NodeMCU 8266 microcontroller to monitor temperature and humidity. Data is transmitted to farmers via the Telegram app, allowing for remote monitoring and early warning alerts when environmental parameters exceed safe limits. Field testing and performance evaluations were conducted, comparing mushroom growth between crops cultivated with and without the monitoring system. The results show that mushrooms grown under the IoT-based system achieved better growth, with the system maintaining optimal conditions between 24°C to 27°C for temperature and 80% to 90% for humidity. Communication delays averaged 9 seconds, which impacted the successful rate of real-time monitoring. Overall, the system improved the control of environmental conditions and supported enhanced mushroom growth, demonstrating its effectiveness in optimizing cultivation practices. transmission rates and maintaining environmental parameters to further improve cultivation results.

Keywords: Indoor mushroom cultivation, IoT, Telegram Bot, DHT11, NodeMCU ESP8266

Cite this as: Sandikyawan, R. R., & Arifin, S. (2024). Monitoring System Indoor Mushroom Cultivation via Telegram Bot, *Journal of Informatics Development*, *2*(2), 45–56. https://doi.org/10.30741/jid.v2i2.1348

INTRODUCTION

Mushroom cultivation is one of the agricultural sectors with significant potential in Indonesia. With a supportive tropical climate, various types of mushrooms can be cultivated with promising yields. However, mushroom cultivation requires careful attention to environmental conditions such as temperature, humidity, and lighting to grow optimally and achieve maximum production.

Generally, the optimal temperature for oyster mushrooms is between 24°C and 27°C, with humidity levels ranging from 80% to 90% (Aji, 2018). Low temperatures can slow down metabolic processes and reduce crop yields. On the other hand, excessively high temperatures may promote the growth of unwanted fungi or pathogens. Moreover, unstable humidity can make mushroom fruiting bodies more susceptible to damage or contamination (Chong et al., 2023). In the current era of information technology, the application of technology-based monitoring systems can be an efficient and effective solution to overcome the challenges of indoor mushroom cultivation. An automated and real-time monitoring system can assist farmers in accurately and responsively monitoring environmental conditions, enabling timely intervention when necessary.

The Internet of Things (IoT) technology presents a promising option for implementation in mushroom cultivation monitoring systems. The definition of the Internet of Things is sensors that are connected to the internet and behave like the internet by creating open connections at all times, as well as sharing data freely and enabling unexpected applications, so that computers can understand the world around them and become part of human life (Yudhanto, Y., & Azis, A. 2019). With sensors connected to the internet, data on temperature, humidity, and other parameters can be collected and analyzed in real time. This system not only simplifies the monitoring process but can also provide early warnings if conditions deviate from the mushrooms' requirements.

Integrating an IoT-based monitoring system with a mobile or web application is a strategic step in supporting the digitalization of the agricultural sector, especially in indoor mushroom cultivation. This technology allows farmers to monitor environmental conditions such as temperature, humidity, light, and CO2 levels in real time from a distance. In addition, farmers can also receive notifications or early warnings if conditions occur that are not in accordance with predetermined parameters. With this feature, farmers can take quick action to address potential problems before they affect production.

This system can also be equipped with automation functions, such as ventilation, watering, or lighting settings, which are directly triggered by sensor data. This not only increases operational efficiency but also reduces reliance on manual labor, which is often a constraint in mushroom cultivation. In addition, historical data collected by the system can be analyzed to identify patterns or trends that can be used to improve cultivation practices. For example, farmers can understand how certain changes in environmental parameters affect mushroom growth and then use this insight to optimize future cultivation conditions.

In the long term, this system is expected to contribute to creating a more sustainable and technology-based agricultural business model. The use of IoT technology in mushroom cultivation not only increases productivity and quality of harvests, but also allows for the development of a more environmentally friendly and energy-efficient approach. Therefore, this study focuses on the design and development of an IoT-based monitoring system that is not only technologically relevant, but also has a positive impact on farmers and the agricultural sector as a whole. The results of this study are expected to be the basis for further innovation in the application of technology in agriculture, especially in Indonesia, and support efforts to improve national food security.

E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

METHOD

The data obtained from the node will be processed by Wemos D1 to run the automation of the DC water pump and exhaust fan, then the data will be sent to the database to be stored using the MQTT protocol (Kristiyanti, D. R. et al., 2022). The Fuzzy Logic method will map the temperature and humidity conditions expressed in membership values and degrees of truth so that the system can make its own decisions (Saputra, C. et al., 2022). The system that has been created uses a DHT 22 sensor to measure the temperature and humidity in the room, then the results will be displayed via a 20x4 LCD layer offline and via the blynk application on a smartphone as a monitoring medium and notification of sensor reading results remotely (Dirgayusari, A. M. et al. 2021).

A. General System Description

The system design is developed according to the previously defined requirements engineering to facilitate the implementation stage and subsequent phases. The system to be built collects data from temperature and humidity sensors connected to a microcontroller. Microcontrollers work based on the program embedded in them, and the program is made according to the desired application (Dharmawan, H. A. 2017). The data obtained by the sensors will be read by a NodeMCU 8266 microcontroller. There are pins specifically for SPI (Serial Peripheral Interface) and PWM (Pulse Width Modulation) communication which are not available in version 0.9, ESP8266 uses Wifi2.4 GHz, supports WPA/WPA2 (Setiawan, Y. 2017). The data received by the NodeMCU 8266 from the sensors will then be processed and sent to the client via the internet, allowing the client to monitor the conditions in real time through the Telegram application. The system design that will be implemented in this study can be seen in Figure 1.

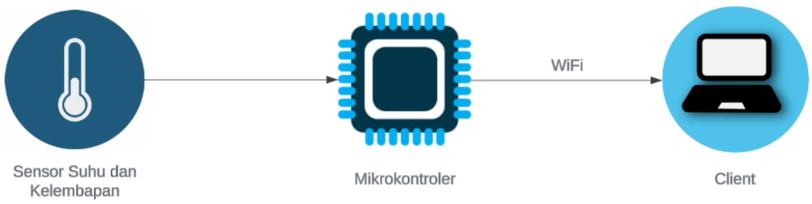

Figure 1. System Overview
Source : Research

B. System Design

The temperature and humidity sensor used in this study is the DHT11 sensor, which is connected to a NodeMCU 8266 microcontroller. The design process begins with the hardware installation, where the DHT11 sensor is attached to the NodeMCU 8266 by connecting the necessary pinouts. The NodeMCU 8266 is programmed using the Arduino IDE. Arduino evolved into a platform because it became the choice and reference for many practitioners (Djuandi, F. 2011).

The sensor will be controlled by the NodeMCU 8266 by initiating the input port to enable communication between the two, with the execution of source code from the Arduino IDE. To allow the client to connect to the device created on the NodeMCU 8266, source code is written for WiFi connection and Bot Token handling. If the WiFi username and password or Bot Token are incorrectly inputted, the connection to the device will fail. Once successfully connected, the NodeMCU 8266 will start reading the DHT11 sensor. The working principle is to utilize capacitive changes in the position of the dielectric material between the two plates, the shift in the position of one of the plates and the area of the plates facing each other (Budi, K. S., & Pramudya, Y. 2017).

After completing the sensor reading, the data will be further processed to determine if the values are still within safe limits. If the temperature is below 24°C or above 27°C, or if the humidity is below 80% or above 90%, the system will send a warning message along with the data to the client.

Otherwise, the data transmitted will reflect the sensor readings without any alerts. The system workflow design can be seen in the main process illustrated in Figure 2.

The sub-processes in the system workflow flowchart involve the NodeMCU 8266 reading values from the DHT11 sensor, followed by checking these values against predefined parameters. The results are classified as follows:

- Temperature is considered safe if it falls between 24°C and 27°C.
- Temperature is deemed too cold if it is below 24°C.
- Temperature is categorized as too hot if it exceeds 27°C.
- Humidity is deemed safe if it falls between 80% and 90%.
- Humidity is classified as too dry if it is below 80%.
- Humidity is considered too humid if it exceeds 90%.

The NodeMCU then sends the classified data, including the temperature and its classification as well as the humidity and its classification, to the client. The sensor reading workflow can be seen in the sub-processes illustrated in Figure 3.

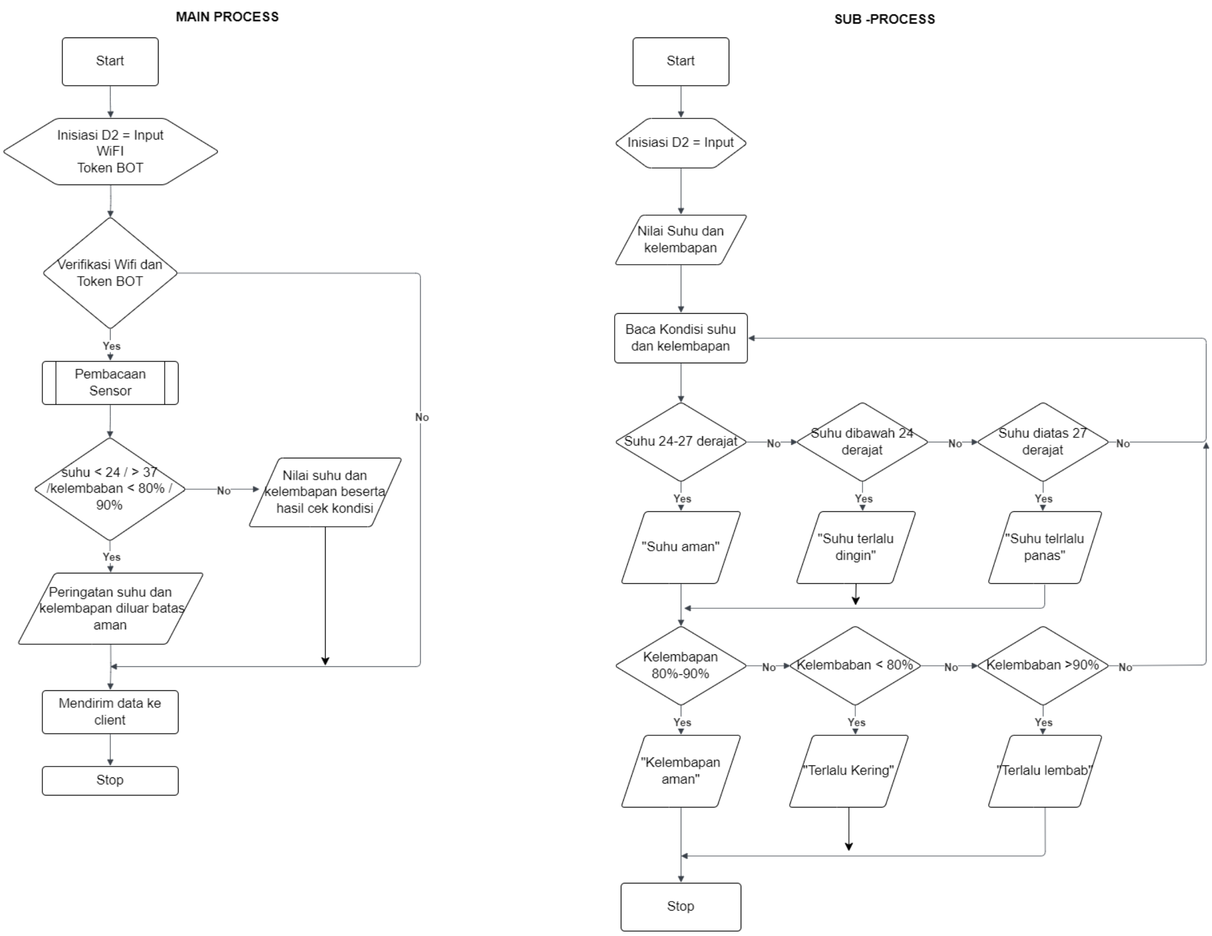

Figure 2. Flowchart diagram of sensor readings
Source : Research

JOURNAL OF INFORMATICS DEVELOPMENT
E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

C. Aplication Design

The design of the client application will utilize Telegram, specifically leveraging the Telegram Bot feature. Telegram Bot is a robot that can be used or set according to user needs, can respond to messages or send messages, this feature is easy to use and very helpful for IoT implementation (Saputra, C. et al., 2022). When the client first accesses the chat bot, they must enter the command "/start" to initiate the connection with the NodeMCU. To connect with the NodeMCU, the client requires an internet connection and a matching Bot Token. If these conditions are not met, the client will not be able to connect to the NodeMCU. In C++, new concepts such as classes with their properties such as inheritance and overloading are added. One of the most fundamental differences with the C language is the support for the concept of object-oriented programming (OOP) (Hanief, S., Jepriana, I. W., & Kom, S. 2020).

Once connected, the system will display the message "/stop" to terminate the data transmission process. Data will continue to be received at the specified intervals as long as the client does not input the "/stop" command. When the client enters the "/stop" command, data transmission will cease, and the message "data sending stopped" will appear. Additionally, the client can type the command "/status" to retrieve data from the NodeMCU and display it with a single request and the client will also receive an automatic warning message if the temperature or humidity falls outside the safe range. The workflow design of the client application can be seen in Figure 3.

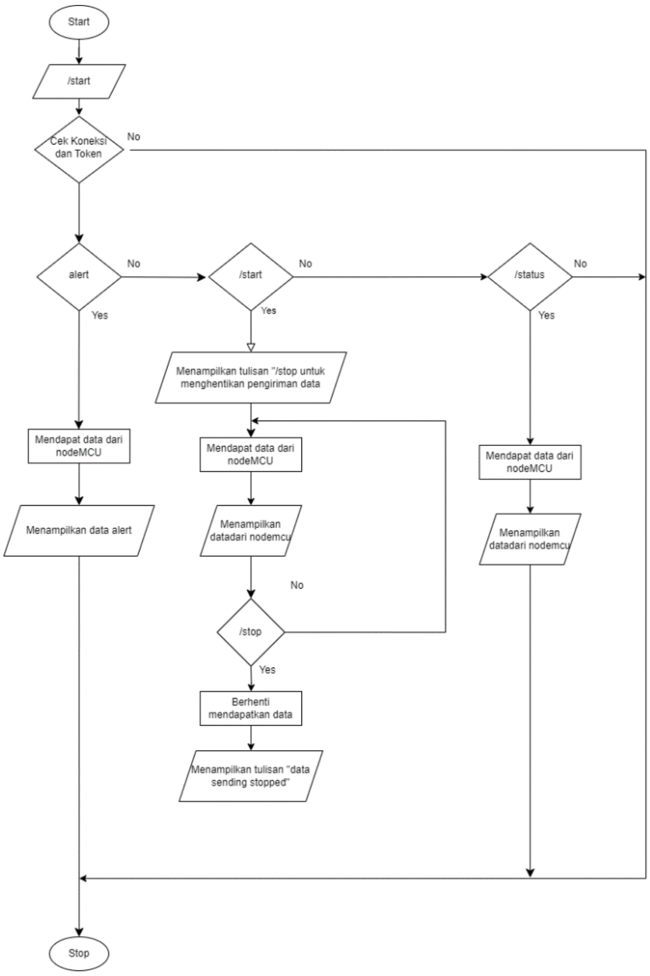

Figure 3. Flowchart user application diagram
Source : Research

JOURNAL OF INFORMATICS DEVELOPMENT
E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

RESULTS AND DISCUSSION

A. Functional System Testing

The first test conducted is the sensor testing. This test is performed to determine whether the system is capable of obtaining temperature and humidity data from the DHT11 sensor. The results of the sensor testing can be seen in Figure 4.

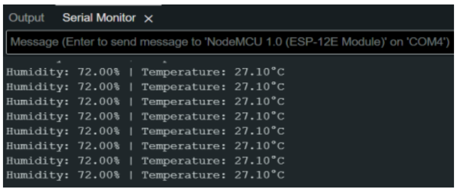

Figure 4. ***Sensor Testing Results***
Source : Data Processing, 2024

The second test is the NodeMCU testing. This test aims to verify that the system can successfully obtain and process temperature and humidity data according to the specified parameters. The results of the NodeMCU testing can be seen in Figure 5.

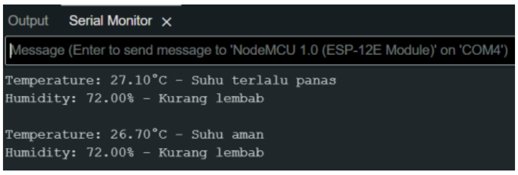

Figure 5. NodeMCU Testing Results
Source : Data Processing, 2024

The final test in the functional system testing is the client testing. This test is used to confirm that the Telegram Bot can receive and display temperature and humidity readings along with the processed data from the NodeMCU. The results of the client testing can be seen in Figure 6.

Figure 6. Client Testing Results
Source : Data Processing, 2024

B. System Performance Testing

System performance testing is conducted in several parts. The first part is command testing. The purpose of this test is to evaluate whether the client can send the commands “/start,” “/status,” and “/stop” to the NodeMCU and receive results as per the system design.

The “/start” command to NodeMCU through the Telegram Bot chat is intended to display data continuously received from the NodeMCU. The results of this test can be seen in Figure 7.

E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

Figure 7. /start Command Results
Source : Data Processing, 2024

The "/status" command through Telegram chat prompts NodeMCU to send only one data reading and the processed data based on the safe parameter limits. The results of this test can be seen in Figure 8.

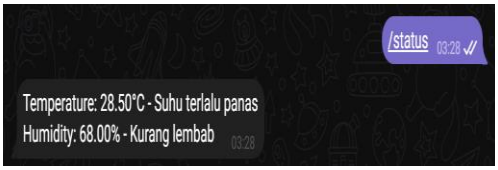

Figure 8. /status Command Result
Source : Data Processing, 2024

The "/stop" command to NodeMCU is used to halt data transmission from NodeMCU to the client. Upon successful execution, a message "Data sending stopped" will appear. The results of this test can be seen in Figure 9.

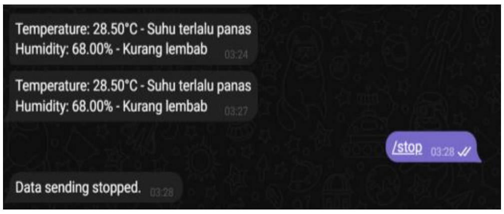

Figure 9. /stop Command Result
Source : Data Processing, 2024

The testing involves evaluating the system's ability to send warning messages when temperature or humidity values fall outside the safe ranges. Specifically, the NodeMCU sends warning messages to the user in the following scenarios:

- Temperature Below 24°C: The system sends a warning if the temperature drops below 24°C.
- Temperature Above 27°C: The system sends a warning if the temperature exceeds 27°C.
- Humidity Below 80%: The system sends a warning if the humidity falls below 80%.
- Humidity Above 90%: The system sends a warning if the humidity exceeds 90%.

These warning messages are automatically transmitted to the client via the Telegram Bot when the sensor readings fall outside the defined safe parameters. The results of these tests, including screenshots or logs of the warning messages, are illustrated in the corresponding figures 10.

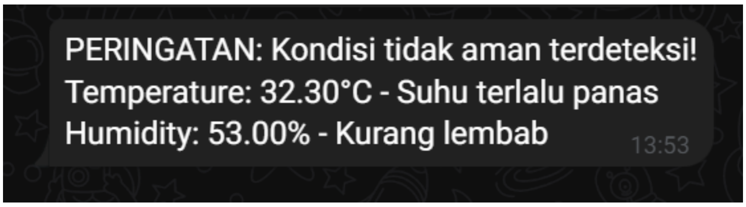

Figure 10. Result of Warning Message
Source : Data Processing, 2024

The last part of the system performance testing is the successful rate testing. This test involves sending data from NodeMCU to the client over 30 minutes with intervals of 3 seconds, 3 minutes, and 5 minutes. The successful rate is influenced by delays in communication between NodeMCU and the Telegram Bot, which is why the successful rate cannot reach 100%. After conducting tests for 30 minutes with the specified intervals (30 seconds, 3 minutes, and 5 minutes), the average delay was found to be 9 seconds. The results of the successful rate testing can be seen in Figure 11.

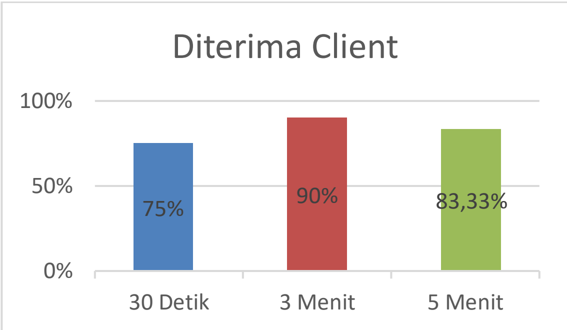

Figure 11. Successful Rate Testing Results
Source : Data Processing, 2024

C. Field Testing

The field testing was conducted to evaluate the performance of the developed system in real-world conditions. This testing involved monitoring temperature and humidity in a mushroom cultivation location and making necessary adjustments if issues were identified. The test involves displaying the temperature and humidity conditions in the morning, afternoon, and evening, and then showing the current conditions or initial conditions before watering and after watering.

The temperature and humidity readings in the mushroom chamber during the morning between 7 am to 12 pm are provided in the table 1.

Table 1. Morning Conditions

Temperature	Humidity
21.90°C	81.00%
22.20°C	79.00%
22.60°C	75.00%
23.40°C	73.00%
24.80°C	71.00%

Source : Data Processing, 2024

The temperature and humidity readings in the mushroom chamber during the afternoon between 12 pm to 5 pm are provided in the table 2.

JOURNAL OF INFORMATICS DEVELOPMENT
E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

Table 2. Afternoon Conditions

Temperature	Humidity
28.90°C	68.00%
28.50°C	68.00%
27.60°C	71.00%
26.70°C	72.00%
26.20°C	74.00%

Source : Data Processing, 2024

The temperature and humidity readings in the mushroom chamber during the evening between 5 pm to 10 pm are provided in the table 3.

Table 3. Evening Conditions

Temperature	Humidity
25.30°C	77.00%
24.80°C	73.00%
24.60°C	74.00%
26.70°C	72.00%
26.20°C	70.00%

Source : Data Processing, 2024

The results of the temperature and humidity observations, along with the parameter conditions in real-time from before watering to after watering are shown in Figure 12

Figure 12. Watering Test Results
Source : Data Processing, 2024

Maintaining optimal room conditions, with a temperature between 24°C and 27°C and humidity between 80% and 90%, is essential to minimize the risk of mushrooms not growing properly or even

JOURNAL OF INFORMATICS DEVELOPMENT
E-ISSN : 2963-0568, P-ISSN : 2963-055X
Available online at:
https://ejournal.itbwigalumajang.ac.id/index.php/jid

spoiling. The test involves one baglog of mushrooms on a 1x4 meter rack equipped with the monitoring system and one baglog on a rack without the monitoring system. The farmer performs at least one watering session during the afternoon. This setup allows for a comparison of mushroom growth between the rack with the monitoring system and the rack without it.

Table 4. Data Baglog with System Monitoring

	Before	After
Length	7,4 cm	8,0 cm
Widht	6,5 cm	7,3 cm
Height	9,0 cm	10,0 cm

Source : Data Processing, 2024

The table indicates that the mushroom baglog equipped with the monitoring system showed growth with dimensions of 0.6 cm in length, 0.8 cm in width, and 1 cm in height. This growth reflects the system's effectiveness in maintaining optimal environmental conditions, which contributes to better mushroom development. The corresponding image of the mushroom baglog with the monitoring system is provided in the figure 13.

Figure 13 Baglog with System Monitoring
Source : Data Processing, 2024

Table 5. Data Baglog without System Monitoring

	Before	After
Length	4,5 cm	5,5 cm
Widht	3,5 cm	4,5 cm
Height	7,5 cm	8,0 cm

Source : Data Processing, 2024

The table shows that the mushroom baglog without the monitoring system exhibited growth with dimensions of 0.5 cm in length, 0.5 cm in width, and 0.5 cm in height. This growth highlights the impact of the absence of a monitoring system, as the lack of consistent environmental control may result in less optimal mushroom development compared to those with monitoring. The corresponding image of the mushroom baglog with the monitoring system is provided in the figure 14.

Figure 14 Baglog without System Monitoring
Source : Data Processing, 2024

Based on the above testing, it can be concluded that the monitoring system helps mushroom farmers maintain optimal room conditions within safe limits and enhances mushroom growth.

CONCLUSION

This study evaluated the performance and effectiveness of a mushroom cultivation monitoring system incorporating NodeMCU and Telegram Bot for real-time data management. The findings indicate that the system effectively integrates the DHT11 sensor with the NodeMCU, accurately measuring and processing temperature and humidity for reliable monitoring. The system demonstrates robust functionality, providing users with real-time data access via Telegram Bot, though occasional communication delays averaging 9 seconds suggest the need for optimizing data transmission intervals. Field testing confirmed the system's practical effectiveness, as mushrooms cultivated with the monitoring system exhibited superior growth compared to those grown without it, with optimal environmental conditions identified as a temperature range of 24°C to 27°C and humidity between 80% and 90%. Future work should focus on optimizing data transmission intervals and maintaining these optimal conditions to further enhance system performance and mushroom development.

REFERENCES

Aji, N. (2018). Pengatur Suhu dan Kelembaban Kumbung Jamur Otomatis. ELINVO (Electronics, Informatics, and Vocational Education, 50.

Budi, K. S., & Pramudya, Y. (2017). PENGEMBANGAN SISTEM AKUISISI DATA KELEMBABAN DAN SUHU DENGAN MENGGUNAKAN SENSOR DHT11 DAN ARDUINO BERBASIS IOT. PROSIDING SEMINAR NASIONAL FISIKA (E-JOURNAL), 6,SNF2017-CIP. https://doi.org/10.21009/03.SNF2017.02.CIP.07

Chong, J. L., Chew, K. W., Peter, A. P., Ting, H. Y., & Show, P. L. (2023). Internet of things (IoT)-Based environmental monitoring and control system for home-based mushroom cultivation. Biosensors, 13(1), 98. https://doi.org/ 10.3390/bios13010098

Dharmawan, H. A. (2017). Mikrokontroler: konsep dasar dan praktis. Universitas Brawijaya Press. Available at books.google.com [Akses 20 Juli 2024]

Dirgayusari, A. M., & Sudiarsa, I. W. (2021). Implementasi sistem monitoring dan kontrol suhu kelembaban ruang budidaya jamur berbasis IoT. Jurnal Sistem Informasi Dan Komputer Terapan Indonesia (JSIKTI), 4(2), 78-89.

Djuandi, F. (2011). Pengenalan arduino. E-book. www. tobuku, 24. [Access 20 Juli 2024]

Enterprise, J. (2017). Otodidak pemrograman JavaScript. Jakarta: Elex Media Komputindo.

Hanief, S., Jepriana, I. W., & Kom, S. (2020). Konsep Algoritme dan Aplikasinya dalam Bahasa Pemrograman C++. Penerbit Andi.

Hidayatulloh, M. F., Arifin, S., & Islamiyah, M. (2022). Rancang Bangun Sistem Pengabutan Air Untuk Ruangan Kumbung Jamur. POSITIF: Jurnal Sistem dan Teknologi Informasi, 8(1), 18-25.

Islam, T., Zakaria, Z., Hamidin, N., & Ishak, M. A. B. M. (2016). Optimization of humidifying procedure in controlled environment for indoor cultivation of Pleurotus pulmonarius. African Journal of Biotechnology , 15(45), 2578-2586

Kristiyanti, D. R., Wijayanto, A., & Aziz, A. (2022). Sistem Monitoring Suhu dan Kelembaban pada Budidaya Jamur Tiram Berbasis Internet of Things Menggunakan MQTT dan Telegram BOT. Adopsi Teknologi Dan Sistem Informasi (ATASI), 1(1), 61 - 73. https://doi.org/10.30872/atasi.v1i1.60.

Islamiyah, M., Arifin, S. (2023). Control System Of Chicken Egg Incubator Using Fuzzy Logic Method. Jurnal POSITIF: Jurnal Sistem dan Teknologi Informasi, 9(2), 123-129.

Prasetyo, S., & Abdullah, S. (2021). Rancang Bangun Penyiram Tanaman Otomatis Berbasis Internet of Things Menggunakan NodeMCU dan Telegram. Jurnal RESTIKOM: Riset Teknik Informatika dan Komputer, 3(2), 51-59.

Purwono Prasetyawan, Muhammad Rizqi Rahmatullah, Raizummi Fil'aini, & Ilham Firman Ashari. (2024). SISTEM PENGKABUTAN OTOMATIS UNTUK OPTIMALISASI PRODUKTIVITAS KUMBUNG JAMUR KEPUNG SETO. Prosiding Seminar Nasional Terapan Riset Inovatif (SENTRINOV), 9(3), 342 - 350. Retrieved from https://proceeding.isas.or.id/index.php/sentrinov/article/view/1484

Saputra, C., Setiawan, R., & Arvita, Y. (2022). Penerapan Sistem Kontrol Suhu dan Monitoring Serta Kelembaban pada Kumbung Jamur Tiram Berbasis Iot Menggunakan Metode Fuzzy Logic. Jurnal Sains dan Informatika, Volume 8, Nomor 2. https://doi.org/10.34128/jsi.v8i2.504

Setiawan, Y. (2017). TA: Rancang Bangun Pemantauan dan Penjadwalan Alat Pemberi Pakan Ikan Otomatis Secara Jarak Jauh (Doctoral dissertation, Institut Bisnis dan Informatika Stikom Surabaya).

Sofwan, A., Wafdulloh, Y., Akbar, M. R., & Setiyono, B. (2020). Sistem pengaturan dan pemantauan suhu dan kelembapan pada ruang budidaya jamur tiram berbasis iot (internet of things). Transmisi, 22(1), 1-5.

Yudhanto, Y., & Azis, A. (2019). Pengantar Teknologi Internet of Things (IoT). UNSPress. Available at books.google.com [Access 20 Juli 2024]