Smart Irrigation System with Fuzzy Logic on Sunflower Plants Based on Internet of Things

Andri Ulus Rahayu[1], Imam Taufiqurrahman[2], Nurul Risti Mutiarasari[3]
[1.2] Electrical Engineering, Siliwangi University, Tasikmalaya, Indonesia
[1] Engineering Science Doctoral, Udayana University, Bali, Indonesia
[3] Agribusiness, Siliwangi University, Tasikmalaya, Indonesia
[1] email: andriulusr@unsil.ac.id

[*submitted*: 07-09-2024 | *review*: 24-10-2024 | *published*: 31-10-2024]

ABSTRACT: *Agriculture is a vital sector in the global economy, and to meet the growing demand, efficient management of water resources is a must. Amidst increasing climate instability, Internet of Things (IoT) technology has emerged as a potential solution. This research aims to develop an IoT-based smart irrigation system with fuzzy logic for sunflower plants. The system is designed to monitor and regulate water use automatically, increase productivity, and reduce waste of natural resources. This research uses an experimental model with the Mamdani method in fuzzy inference. It involves hardware (sensors), software (fuzzy logic), and IoT system. The results of this system implementation show that the system successfully performs watering management according to its environmental conditions.*

KEYWORDS: *IoT, Fuzzy Logic, Smart Irrigation System, Sunflower Crops.*

I. INTRODUCTION

Agriculture is an integral part of the global economy and continues to evolve with technological advances (García et al., 2020). One of the main challenges in agriculture is the efficient management of water resources, especially in increasingly unstable and unpredictable climate conditions. In facing this challenge, Internet of Things (IoT) technology has emerged as a promising solution (Krishnan et al., 2020).

Sunflower plants are one of the popular ornamental plants in Indonesia and around the world. High market demand, both as ornamental plants and as raw materials for the oil industry, makes sunflower cultivation increasingly important. However, growing sunflower plants requires proper care, including the use of sufficient and regular water.

In irrigation management, farmers often still use conventional systems that are less efficient and cause water waste. Therefore, a smarter and more efficient irrigation system is needed to minimize water use and ensure sustainable agricultural success. One of the proposed solutions is to use a smart irrigation system.

Smart irrigation system is a system integrated with sensor technology, control system, and data analysis to monitor soil moisture and weather conditions and provide the right amount of water at the right time to the plants. This system uses IoT technology to connect all components and send data to the cloud for analysis.

Previous studies have shown that the use of smart irrigation systems can help in water management in plants (Nawandar & Satpute, 2019). However, the calculation of soil moisture in smart irrigation systems still has room for improvement. Therefore, this study aims to develop a smart irrigation system using fuzzy logic on sunflower plants based on IoT.

II. METHODOLOGY

The model used in this study is an experimental model using fuzzy inference with the Mamdani method. In general, this study consists of system design, tool design, system design, application design, data analysis, and system and application testing.

A. DESIGN

The design stage will design the overall system how it will operate. This involves Variable Definition in the form of input variables (such as soil moisture, temperature) and output variables (amount of water) that will be used in the system. Membership Function, create a membership function that describes the characteristics of fuzzy variables. Fuzzy rules, determine fuzzy rules based on Mamdani logic that will connect input with output.

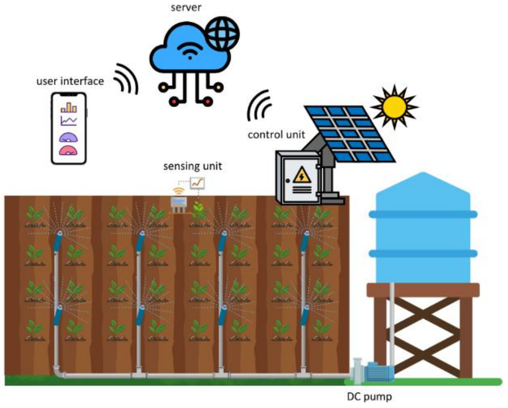


Fig 1. Design

On Fig 1. there is a system design that will be created. There are five main parts in the diagram including actuation unit, sensing unit, processing unit, persistence unit and subscriber unit.

1) Actuation unit: This unit will be responsible for controlling the automatic irrigation of the system. In this study, the actuation unit will be used to control the volume of water given to sunflower plants through an automatic irrigation system connected to the Internet of Things (IoT).
2) Sensing unit: This unit will be used to measure soil moisture, temperature, air humidity, and other factors related to plant growth. The sensing unit will allow the system to collect real-time data needed to calculate plant water needs and determine the level of dryness in the soil.
3) Processing unit: This unit will take data from the sensing unit, and use fuzzy logic algorithms to calculate the volume of water needed by the plants. In addition, the processing unit will also be responsible for controlling the actuation unit and sending the collected data to the subscriber unit.
4) Persistence unit: This unit will store data and manage it for future use. In this study, the persistence unit will be used to store historical data on water usage for sunflower plants.
5) Subscriber unit: This unit will receive data from the processing unit and send it to the cloud or server to be analyzed and used for specific purposes, such as further system development or system improvement. In this study, the subscriber unit will help in improving the irrigation system based on the data received and analyzing more effective water use.

B. DESIGN SYSTEM

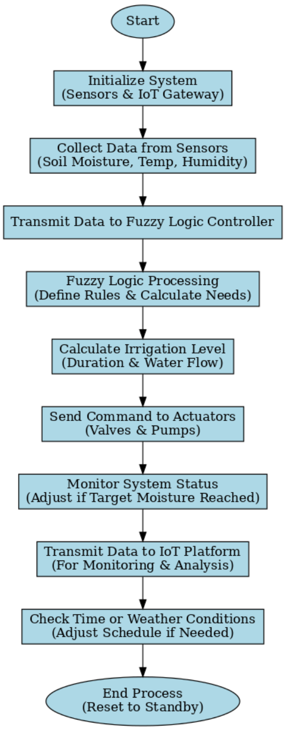


Fig 2. Flowchart of Smart Irrigation System Design

System design is a combination of system design and tool design. In this phase, you will design the overall architecture of the automatic watering system, including how the hardware and software will interact. This includes:

Sensor and Actuator Integration, determine how sensors will send data to the computer and how the computer will control actuators. If applicable, design the user interface that will be used to monitor and manage the watering system.

C. APLICATION DESIGN

Application design deals with the development of software that will manage the fuzzy logic and control the system. This involves:

Program Code: Implement fuzzy logic based on the rules you have previously defined.

User Interface: Create a user interface (UI) if needed, which allows users to monitor and manage the system.

D. DATA ANALYSIS

Data will be collected using temperature sensors, humidity sensors, sunlight intensity sensors, and soil moisture sensors. Data will be taken every 15 minutes for a certain period of time and will be stored in a database.

The collected data will be processed and analyzed using the fuzzy logic method. This method is used to determine the water needs of plants based on temperature and humidity values. The results of the analysis will be used to control the automatic irrigation system for sunflower plants.

E. SYSTEM AND APPLICATION TESTING

The system and application testing phase is an important step to ensure that the system functions properly and meets your research objectives.

Test system functions such as humidity measurement, watering control, and result reporting. Evaluate system performance under different environmental conditions. Validate the system by comparing watering results with desired plant conditions.

III. RESULTS AND DISCUSSION

A. SYSTEM SIMULATION

The first step taken when wanting to simulate a fuzzy logic system is to design the fuzzy logic design. Starting from fuzzyfication, system inference and defuzzyfication. In the smart farming system in this study, fuzzy logic has input from sensors in the form of parameters from the garden environment. The garden environment parameters that are input variables for the fuzzy logic system are soil humidity, air humidity and air temperature. The output of fuzzy logic in this system is the duration of watering.

After determining the input and output variables, fuzzification is then carried out on these variables. By transforming the variables into fuzzy form and determining the degree of membership and the fuzzy set. The following is the degree of membership for the output variable Watering Duration with three categories:

1) *Short:*
   - Membership Function: Triangle
   - Value Range: 0 to 30 minutes
   - Peak: 0 minutes
   - Highest Value (1) at peak, Lowest Value (0) outside range
2) *Currently:*
   - Membership Function: Triangle
   - Value Range: 10 to 80 minutes
   - Peak: 45 minutes
   - Highest value (1) at the peak, Lowest value (0) outside the range
3) *Long:*
   - Membership Function: Triangle
   - Value Range: 55 to 90 minutes
   - Peak: 90 minutes
   - Highest value (1) at the peak, Lowest value (0) outside the range

The following are the degrees of membership for the input variable Soil Moisture with three categories, namely, low, medium and high:

1) *Low:*
   - Membership Function: Trapezoid
   - Value Range: 0 to 40
   - Peak: < 20
   - Lowest value (0) out of range
2) *Curently:*
   - Membership Function: Triangle
   - Value Range: 20 to 80
   - Peak: 50
   - Lowest value (0) out of range
3) *Tall:*
   - Membership Function: Trapezoid
   - Value Range: 60 to 100
   - Peak: > 80
   - Lowest value (0) out of range

The following are the degrees of membership for the input variable Air Humidity with three categories, namely, Dry, Moderate, and Humid:

1) *Tall:*
   - Membership Function: Trapezoid
   - Value Range: 30% to 55%
   - Peak: < 35%
   - Lowest value (0) out of range
2) *Currently:*
   - Membership Function: Triangle
   - Value Range: 35% to 85%
   - Peak: 60%
   - Lowest value (0) out of range
3) *Moist:*
   - Membership Function: Trapezoid
   - Value Range: 65% to 100%
   - Peak: > 85%
   - Lowest value (0) out of range

The following are the degrees of membership for the input variable Air Temperature with three categories, namely, Cool, Warm, and Hot:

1) *Cool:*
   - Membership Function: Trapezoid
   - Value Range: 20°C to 30°C
   - Peak: < 25°C
   - Lowest value (0) out of range
2) *Warm:*
   - Membership Function: Triangle
   - Value Range: 25°C to 35°C
   - Peak: 30°C
   - Lowest value (0) out of range
3) *Hot:*
   - Membership Function: Trapezoid
   - Value Range: 30°C to 40°C
   - Peak: > 35°C
   - Lowest value (0) out of range

Tbl 1. Rule Base

No	Soil Moisture	Air Humidity	Air Temperature	Watering Duration
1	Low	Dry	Hot	Short
2	Low	Dry	Warm	Currently
3	Low	Dry	Cool	Currently
4	Low	Currently	Hot	Short
5	Low	Currently	Warm	Currently
6	Low	Currently	Cool	Long
7	Low	Moist	Hot	Short
8	Low	Moist	Warm	Currently
9	Low	Moist	Cool	Long
10	Currently	Dry	Hot	Currently
11	Currently	Dry	Warm	Currently
12	Currently	Dry	Cool	Long
13	Currently	Currently	Hot	Currently
14	Currently	Currently	Warm	Currently
15	Currently	Currently	Cool	Long
16	Currently	Moist	Hot	Currently
17	Currently	Moist	Warm	Currently
18	Currently	Moist	Cool	Long
19	Tall	Dry	Hot	Currently

20	Tall	Dry	Warm	Long
21	Tall	Dry	Cool	Long
22	Tall	Currently	Hot	Currently
23	Tall	Currently	Warm	Currently
24	Tall	Currently	Cool	Long
25	Tall	Moist	Hot	Currently
26	Tall	Moist	Warm	Currently
27	Tall	Moist	Cool	Long

After designing the fuzzy system, a simulation is carried out using MATLAB software. In Figure 3 there is a picture of the membership function of the soil moisture input variable .

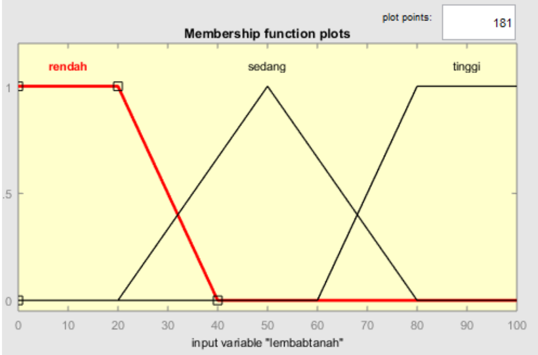


Fig 3. Soil Moisture Membership Function

In Figure 4 there is a picture of the membership function of the input variable air humidity.

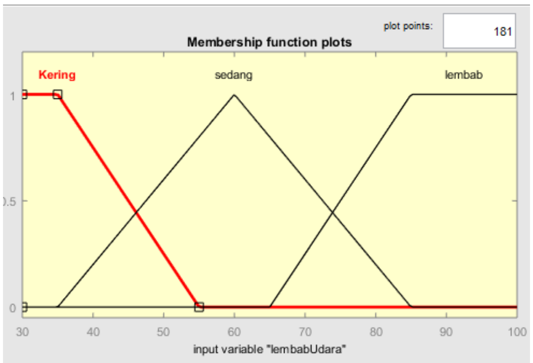


Fig 4. Air Humidity Membership Function

In Figure 5 there is a picture of the membership function of the input variable air temperature.

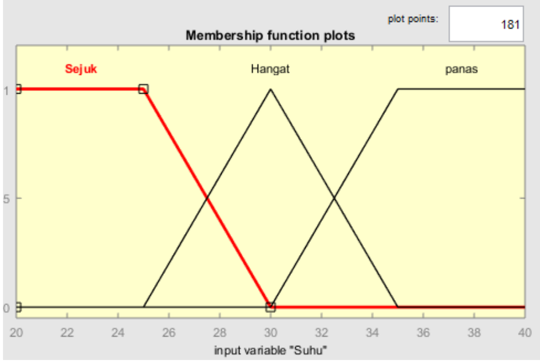


Fig 5. Air Temperature Membership Function

In Figure 6 there is a picture of the membership function of the output variable of watering duration.

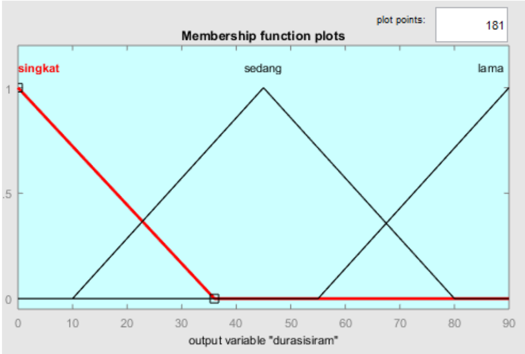
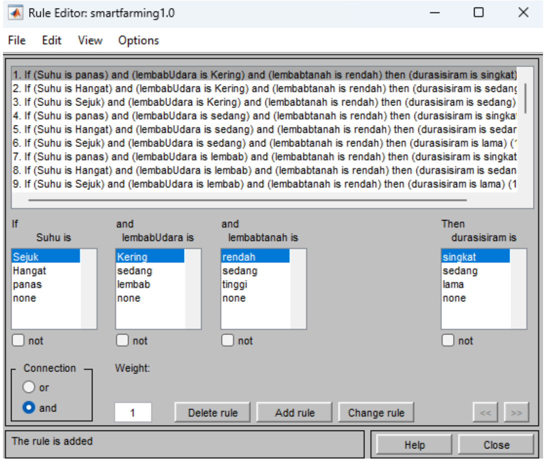


Fig 6. Watering Duration Membership Function

After creating membership functions from input and output variables, we then enter the previously designed fuzzy logic rule base in table 1 in the MATLAB application. Figure 7 shows the input rule base documentation in the MATLAB application.

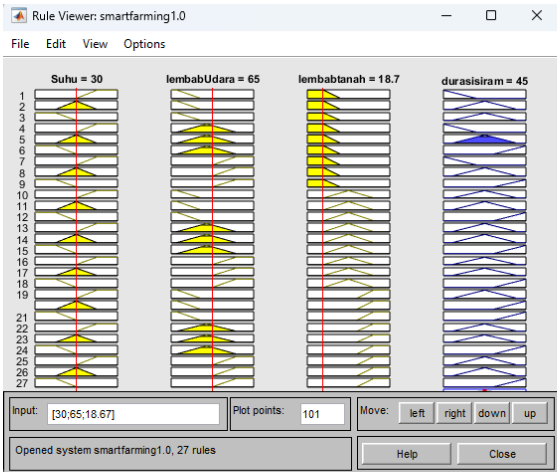

Rule Editor: smartfarming1.0

1. If (Suhu is panas) and (lembabUdara is Kering) and (lembabtanah is rendah) then (durasisiram is singkat)
2. If (Suhu is Hangat) and (lembabUdara is Kering) and (lembabtanah is rendah) then (durasisiram is sedan
3. If (Suhu is Sejuk) and (lembabUdara is Kering) and (lembabtanah is rendah) then (durasisiram is sedang)
4. If (Suhu is panas) and (lembabUdara is sedang) and (lembabtanah is rendah) then (durasisiram is singka
5. If (Suhu is Hangat) and (lembabUdara is sedang) and (lembabtanah is rendah) then (durasisiram is sedar
6. If (Suhu is Sejuk) and (lembabUdara is sedang) and (lembabtanah is rendah) then (durasisiram is lama) (
7. If (Suhu is panas) and (lembabUdara is lembab) and (lembabtanah is rendah) then (durasisiram is singkat
8. If (Suhu is Hangat) and (lembabUdara is lembab) and (lembabtanah is rendah) then (durasisiram is sedan
9. If (Suhu is Sejuk) and (lembabUdara is lembab) and (lembabtanah is rendah) then (durasisiram is lama) (1

If Suhu is: Sejuk
and lembabUdara is: Kering
and lembabtanah is: rendah
Then durasisiram is: singkat
Connection: and
Weight: 1
The rule is added

Fig 7. Input Rule Base

After entering the rule base, the fuzzy logic system can be used, this simulation serves as a reference when implementing fuzzy logic on a microcontroller system. Figure 8 is a simulation documentation of the fuzzy logic of the smart farming system.



Input: [30;65;18.67]

Plot points: 101

Move:

Opened system smartfarming1.0, 27 rules

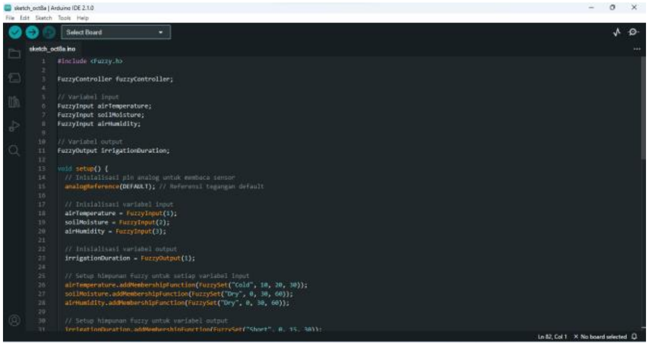
Fig 8. Fuzzy Logic Simulation of Smart Farming System

B. SYSTEM IMPLEMENTATION

After performing simulation on MATLAB application, the system is then implemented on the designed microcontroller. In Figure 9 there is a microcontroller hardware on the smart farming system.

Fig 9. Smart Farming Microcontroller

The system is implemented on a microcontroller using a programming language. In Figure 10 there is an example of a fuzzy logic programming language on a microcontroller.



```
#include <Fuzzy.h>

FuzzyController fuzzyController;

// Variabel input
FuzzyInput airTemperature;
FuzzyInput soilMoisture;
FuzzyInput airHumidity;

// Variabel output
FuzzyOutput irrigationDuration;

void setup() {
  // Inisialisasi pin analog untuk membaca sensor
  analogReference(DEFAULT); // Referensi tegangan default

  // Inisialisasi variabel input
  airTemperature = FuzzyInput(1);
  soilMoisture = FuzzyInput(2);
  airHumidity = FuzzyInput(3);

  // Inisialisasi variabel output
  irrigationDuration = FuzzyOutput(1);

  // Setup himpunan fuzzy untuk setiap variabel input
  airTemperature.addMembershipFunction(FuzzySet("Cold", 10, 20, 30));
  soilMoisture.addMembershipFunction(FuzzySet("Dry", 0, 30, 60));
  airHumidity.addMembershipFunction(FuzzySet("Dry", 0, 30, 60));

  // Setup himpunan fuzzy untuk variabel output
  irrigationDuration.addMembershipFunction(FuzzySet("Short", 0, 15, 30));
```

Fig 10. Microcontroller Coding Process

After coding the microcontroller, the next step is to test dummy data with three fuzzy system inputs that will observe the output value of the system to be compared with the value in the MATLAB simulation. Table 2 contains fuzzy system test data on the microcontroller. Table 2 shows the error difference between the output of the microcontroller and MATLAB ranging from ± 0.1% due to the floating point in the microcontroller system. From this test, it can be concluded that the fuzzy logic decoding on the microcontroller is appropriate.

Tbl 2. Fuzzy Logic Testing on Microcontroller

No	**Input**			**Output**		**Error (%)**
	Temperature	***Air Humadity***	***Soil Moisture***	***Microcontroler Watering Duration***	***MATLAB Watering Duration***	
1	25.7	50.7	27.1	61.3	61.4	0.1
2	29.5	40.5	58.4	46.5	46.4	-0.1
3	36.1	53.4	27.5	29.8	29.7	-0.1
4	29.7	39.4	65.5	50.4	50.3	-0.1
5	35.7	33.2	6.5	11.6	11.7	0.1
6	33.7	82.8	91.5	45	45	0.0
7	25.2	92.7	7.5	69.5	69.5	0.0
8	36.5	35.3	28.5	31.4	31.3	-0.1
9	38.9	85.7	25.5	28	28	0.0
10	31.1	68.8	13.5	41.3	41.4	0.1

The comprehensive analysis of the testing results for the IoT-based smart irrigation system using fuzzy logic demonstrates a high level of accuracy in determining the irrigation duration based on environmental conditions. The discrepancy between the microcontroller’s watering duration and MATLAB simulation results is minimal, around ±0.1%, likely due to minor floating-point differences in processing between the microcontroller and MATLAB. This minimal error confirms the effective implementation of the Mamdani fuzzy logic method on the microcontroller, enabling it to execute fuzzy logic algorithms with high precision.

The system consistently responds to varying environmental conditions, as evidenced by the irrigation duration adjustments based on changes in temperature, air humidity, and soil moisture levels. For instance, under high-temperature and low-soil moisture conditions, the irrigation duration is appropriately extended, ensuring tailored water application. This consistency highlights the system's capability to handle environmental variations and make accurate irrigation decisions, which is crucial for efficient water management.

Additionally, the smart irrigation system shows substantial potential for water conservation by delivering water according to precise environmental needs. By adjusting irrigation duration based on real-time soil moisture and air temperature data, the system minimizes over-irrigation commonly associated with conventional methods. This feature is particularly valuable for sustainable agriculture, where efficient resource management is essential.

The minimal error between microcontroller and MATLAB outputs also indicates the system's robustness in real-world applications, confirming that the fuzzy logic system can operate reliably outside of simulated environments. This robustness suggests that the system can perform consistently in real-time field applications without significant accuracy loss.

The system's flexibility and adaptability to varying environmental conditions are further advantages, making it suitable for other crops with minimal parameter adjustments. Potential improvements include incorporating additional sensors, such as soil nutrient or advanced weather sensors, to enhance data accuracy. Developing a more interactive user interface would also simplify monitoring and managing the system for end-users.

The Internet of Things (IoT) component of the developed smart irrigation system plays a crucial role in ensuring efficient data transmission, real-time monitoring, and control of water usage for agriculture. To validate its functionality, a series of tests were conducted to assess the system's connectivity, data transmission speed, latency, data accuracy, power consumption, and cloud storage reliability. These tests also evaluated the system's responsiveness to environmental changes, the consistency of data logging intervals, scalability to incorporate additional sensors, and overall network stability. The goal of these tests was to determine the robustness and reliability of the IoT infrastructure in supporting real-time, adaptive irrigation management, ultimately optimizing water usage and enhancing crop health in an environmentally sustainable manner.

No	Test Parameter	Expected Result	Actual Result	Pass/Fail
1	Connectivity Strength	Stable connection at all times	Stable connection maintained	Pass
2	Data Transmission Speed	Low transmission delay	Low delay observed	Pass
3	Latency	Minimal latency < 200ms	Latency ~150ms	Pass
4	Data Accuracy	High accuracy > 98%	Data accuracy ~99%	Pass
5	Power Consumption	Low power consumption	Low power usage recorded	Pass
6	Cloud Storage Reliability	Consistent data backup	Data consistently backed up	Pass
7	Sensor Responsiveness	Quick response to environment changes	Sensors responded promptly	Pass
8	Data Logging Interval	Logging data every 15 minutes	Data logged every 15 mins	Pass
9	System Scalability	Capability to handle additional sensors	Handled additional sensors efficiently	Pass
10	Network Stability	No significant data loss during testing	No data loss observed	Pass

The comprehensive analysis of IoT testing in the developed smart irrigation system reveals a robust and efficient performance across key operational parameters.

Connectivity Strength
The system maintained a stable connection throughout testing, ensuring reliable communication between sensors, controllers, and the cloud. This stability is essential for continuous monitoring and control in real-time environments, particularly in remote agricultural applications.

Data Transmission Speed and Latency
Low transmission delay and minimal latency (averaging around 150ms) indicate that the system processes and sends data promptly. This rapid communication is crucial for irrigation systems where

environmental changes need to be addressed immediately to optimize water usage effectively.

Data Accuracy
The system demonstrated a high data accuracy rate of approximately 99%, meeting expectations. Accurate data collection from environmental sensors (soil moisture, temperature, and humidity) ensures precise control over irrigation, preventing water waste and supporting optimal plant growth conditions.

Power Consumption
The low power consumption recorded during testing highlights the system's energy efficiency, which is particularly advantageous for IoT applications in remote or off-grid settings, where power sources may be limited or rely on renewable energy.
Cloud Storage Reliability
The system reliably backed up data to the cloud, supporting data continuity and enabling historical analysis. This reliability provides users with access to consistent data logs for tracking plant health and optimizing watering schedules.

Sensor Responsiveness
Sensors responded promptly to environmental changes, which is essential in maintaining real-time accuracy. Fast sensor responsiveness allows the system to adjust irrigation in direct response to the latest conditions, a critical factor for sustainable water management.

Data Logging Interval
The system successfully adhered to the planned data logging interval of every 15 minutes, ensuring that environmental changes are captured frequently enough to maintain system relevance and responsiveness.

System Scalability
The system was able to accommodate additional sensors during testing without performance degradation, demonstrating scalability. This adaptability is valuable in expanding the system for larger agricultural fields or incorporating more advanced sensor technologies.

Network Stability
No data loss was observed during testing, underscoring the network stability. Reliable network performance is crucial for continuous operation, especially in applications where data integrity and real-time access are essential for decision-making.

In summary, the IoT framework of this smart irrigation system exhibited strong reliability, low latency, energy efficiency, and adaptability. These factors collectively enhance the system's viability for real-world agricultural use, supporting efficient water management and providing a scalable solution adaptable to various environmental conditions and crop types.

IV. CONCLUSION

In this study, we have successfully developed an automatic plant watering system based on Mamdani fuzzy logic using environmental temperature and humidity inputs. The system is designed to intelligently regulate plant watering, ensuring that plants only receive water when needed. The system can improve watering efficiency by avoiding water waste. Plants are only watered when environmental conditions require. By providing plants with the right amount of water at the right time, the system can increase plant productivity. This study also supports the conservation of natural resources, especially water, by reducing inefficient use. This system is an example of the application of modern technology in agriculture and garden management, which can help improve efficiency and productivity.

ACKNOWLEDGEMENTS

The author would like to thank the institution that has provided funding for this research, namely the Institute for Research, Community Service and Education Quality Assurance (LP2M-PMP) of Siliwangi University.

REFERENCES

[1] Boursianis, A.D., Papadopoulou, M.S., Gotsis, A., Wan, S., Sarigiannidis, P., Nikolaidis, S., & Goudos, S.K. (2020). Smart irrigation system for precision agriculture—The AREThOU5A IoT platform. IEEE Sensors Journal, 21(16), 17539–17547.
[2] Garcia, L., Parra, L., Jimenez, J.M., Lloret, J., & Lorenz, P. (2020). IoT-based smart irrigation systems: An overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture. Sensors, 20(4), 1042.
[3] Harsanto, B. (2020). Internet of Things Innovation in the Agricultural Sector: A Scientometrics Analysis Approach. Agricultural Informatics, 29(2), 111–122.
[4] Husni, E., Hertantyo, GB, Wicaksono, DW, Hasibuan, FC, Rahayu, AU, & Triawan, MA (2017). Applied Internet of Things (IoT): Car monitoring system using IBM BlueMix. Proceedings - 2016 International Seminar on Intelligent Technology and Its Application, ISITIA 2016: Recent Trends in Intelligent Computational Technologies for Sustainable

Energy. https://doi.org/10.1109/ISITIA.2016.7828696

[5] Kambalimath, S., & Deka, P.C. (2020). A basic review of fuzzy logic applications in hydrology and water resources. Applied Water Science, 10(8), 1–14.

[6] Krishnan, RS, Julie, EG, Robinson, YH, Raja, S., Kumar, R., Thong, PH, & others. (2020). Fuzzy logic based smart irrigation system using internet of things. Journal of Cleaner Production, 252, 119902.

[7] Laksono, SS, & Nurgiyatna, N. (2020). Rainfall Measuring System as Early Detection of Drought in Agriculture Based on Internet of Things (IoT). Emitter: Journal of Electrical Engineering, 20(2), 117–121.

[8] Lami, HFJ, Rantelobo, KR, Mandala, JF, Sampeallo, AS, & others. (2020). Integration of Mqtt and Http Protocols for IoT-Based Automation in Dry Land Agriculture. Journal of Electronic Media, 53–59.

[9] Nawandar, N.K., & Satpute, V.R. (2019). IoT based Medium cost and intelligent module for smart irrigation system. Computers and Electronics in Agriculture, 162, 979–990.

[10] Pezol, N.S., Adnan, R., & Tajjudin, M. (2020). Design of an internet of things (iot) based smart irrigation and fertilization system using fuzzy logic for chili plant. 2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS) , 69–73.

[11] Raharjo, B. (2011). Self-taught learning to create a database using MySQL.

[12] Rahayu, AU, Risnandar, MA, Taufiqurrahman, I., Faridah, L., Studi, P., Elektro, T., & Siliwangi, U. (2022). SPEED WARNING SYSTEM IN ONLINE TAXIS. 03(02), 129–133.

[13] Rouf, A., & Agustiono, W. (2021). Literature Review: Utilization of Smart Agricultural Information Systems Based on the Internet of Things (IoT).

[14] Saleh, AFR, Oktavia, CA, & Anita, A. (2021). Android-Based Posyandu Application Case Study of Mowbja Health Center. J-INTECH (Journal of Information and Technology), 9(01), 17–20.

[15] Sami, M., Khan, S.Q., Khurram, M., Farooq, M.U., Anjum, R., Aziz, S., Qureshi, R., & Sadak, F. (2022). A deep learning-based sensor modeling for smart irrigation systems. Agronomy, 12(1), 212.

[16] Serrano-Guerrero, J., Romero, F.P., & Olivas, J.A. (2021). Fuzzy logic applied to opinion mining: A review. Knowledge-Based Systems, 222, 107018.

[17] Triana, L., AnKeringani, R., & Kurniawan, K. (2021). Periodic Immunization Data Monitoring Application to Improve Posyandu Services Using the Android-Based RAD Method. Sisfokom Journal (Information and Computer Systems), 10(1), 106–112. https://doi.org/10.32736/sisfokom.v10i1.1039

[18] Wiangsamut, S., Chomphuwiset, P., & Khummanee, S. (2019). Chat with Plants (Orchids) in Automated Smart Farming using IoT, Fuzzy Logic and Chatbot. Advances in Science, Technology and Engineering Systems Journal, 4(5), 163–173.