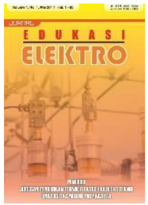
Volume 8, No. 2, November 2024, pages 149 - 160

JEE

Jurnal Edukasi Elektro

https://dx.doi.org/10.21831/jee.v8i2.76816



Automated Hydroponics System using the Internet of Things

Dinan Yulianto (✉), Abiema Febrian Nugraha, Faisal Fajri Rahami

Universitas Ahmad Dahlan, Yogyakarta, Indonesia

dinan.yulianto@tif.uad.ac.id

Abstract— The Indonesian government has instituted urban farming regulations to bolster food security and catalyze economic growth. However, implementing urban farming initiatives across various regions of Indonesia is expected to face challenges in the quality and availability of arable land, along with deficiencies in community knowledge. Hydroponics represents a sustainable agricultural approach proposed as a viable solution to address land quantity and quality limitations. This paper presents the comprehensive design and deployment of an automatic monitoring and control system tailored to hydroponic parameters using Internet of Things (IoT) technologies. This system integrates web technology with a NodeMCU microcontroller and sensor devices, such as DHT22 Sensor, SEN011 Sensor, TDS SEN0244 Sensor, DS18820 Sensor, and HC-SR04 Sensor. Web technology was successfully built to display eight hydroponic environmental data variables in real time, including nutrient levels, water pH, water level, water temperature, air temperature, air humidity, and pump performance status. The pH threshold value of water on a scale of 5.0 to 6.5 will automatically control the pH pump, the nutrient threshold value on a scale of 500 to 800 ppm will automatically control the nutrient pump, and the water height threshold value on a scale of 30 to 10 cm will automatically control the water pump. Through web technology, users can also intervene in system performance based on natural plant conditions. The entire system functionality was tested with 25 scenarios through a black box test approach, demonstrating that the hydroponic environment was monitored and controlled efficiently.

Keywords: hydroponics, internet of things, urban farming, system, web

Article submitted 2024-08-06. Resubmitted 2024-10-23. Final acceptance 2024-11-17. Final version published as submitted by the authors.

1 Introduction

Urbanization is rapidly transforming the distribution of the world's population; by 2022, estimates suggest that 56.2% of people will live in urban regions [1]. The ASEAN Sustainable Urbanization Forum reported a potential growth of 70 million civilians residing in ASEAN cities by 2025 [2]. The acceleration of urbanization has significant implications, mainly due to the increasingly limited agricultural land in urban areas, which is 20% compared to 1.3% in suburban areas and 0% in rural areas [3]. In response to urbanization's challenges, urban farming has emerged as a promising solution [4], [5], [6]. The practice of urban farming could increase the productivity and sustainability of agriculture in cities. Furthermore, urban farming encourages city residents to use unproductive land for small home gardens [7].

The positive impact of urban farming on the environment encourages people to gain knowledge and abilities in the agricultural ecosystem by optimizing the potential of the surrounding resources.

The Indonesian government has even developed urban farming programs in various regions to achieve food security and support economic growth [8]. Unfortunately, Indonesia's urban farming program will face considerable challenges, such as differences in the characteristics of crop fields in quality and quantity. An agricultural sector's essential characteristics include soil type, nutrients, irrigation flow, pest resistance [9], and public education and awareness.

The other perspective. In dense urban areas like Indonesia, conventional agricultural land is limited, water availability is limited, most of the land is potentially polluted or infertile, and the use of plant nutrients has the potential to produce health and environmental problems. Hydroponics, one of the urban farming techniques, has the potential to be implemented in Indonesian programs. Other techniques include vertical farming, indoor farming, aeroponics, aquaculture, and aquaponics [10].

Hydroponics is derived from two Greek words: hydro means water and ponos means labor [11]. Hydroponics is growing plants without soil using nutrient-rich solutions in solvent water [12]. A hydroponic technique can also be considered a flexible and appropriate technology for developing countries [12]. Hydroponics can solve the issues of limited land through alternative media for plants placed in restricted spaces, limited water media through recycling mechanisms, and infertile soil quality by providing plant nutrient solutions through water media. Hydroponic mechanisms related to water recycling and controlling the amount of plant nutrients can reduce potential health and environmental problems.

Previous research has implemented Internet of Things (IoT) technology into various hydroponic techniques through greenhouse media and vertical soilless farming. IoT technology interacts with every physical object via the internet network to access data and control the physical world remotely [13]. The first research was to build and test a micro-scale greenhouse system for lettuce plants [14]. The system automatically monitors and controls the nutrient solution's air temperature, humidity, and pH in the hydroponic growing medium [14]. The second research also focused on building an automatic system for controlling light and air humidity in lettuce plants [15]. The system can activate the pump when humidity reaches 60% and turn on the LED lights between 6 a.m. and 6 p.m. [15]. The third research automatically monitors and controls a hydroponic greenhouse environment against temperature, light, and humidity variables [16]. The system was built specifically for spinach plants, which realize misting activation action when the humidity threshold value is <55%, fan activation action when the temperature threshold value is >= 32oC, and light activation action from 11 a.m. to 3 p.m. [16]. Other research focuses on building automation to provide plant nutrition through a vertical hydroponic system. The system will control water circulation or drip irrigation to efficiently supply nutrients to plants on vertical shelves [17].

Previous research has extensively explored the integration of Internet of Things (IoT) technology into hydroponic systems, focusing on automated monitoring and control mechanisms to optimize plant growth conditions. This study aims to build upon existing research by incorporating the Message Queuing Telemetry Transport (MQTT) protocol, a lightweight messaging protocol, to enhance communication efficiency and scalability within IoT-enabled hydroponic environments [18]. By leveraging MQTT, this research endeavors to achieve real-time data transmission, robust connectivity, and seamless interoperability across diverse IoT devices, thereby advancing the precision and reliability of environmental management in hydroponic farming.

2 Method

2.1 User and functional requirement

User requirement fulfillment is an essential prerequisite for successful product development in software development[19]. The term user requirement refers to the needs and expectations expressed by users regarding creating or developing a software system [20]. The domain of user requirements also includes descriptions of product attributes or characteristics important to user satisfaction [20]. The urgency of user requirements cannot be ignored because a deep understanding of user needs will ensure that the final product meets technical specifications, also can be used effectively and efficiently by targeted users. User requirements correlate with functional requirements, which describe the technical specifications the system must meet to meet user requirements [21]. This research will involve *Putra Kebun Hidroponik* partners, as a hydroponic practitioner who provides advice on user requirements.

2.2 Hardware and system architecture design

A framework for IoT-based food and agricultural system architecture was used for the system design [22]. The integration of hardware and software entities forms system architecture. This hardware architectural design uses the NodeMCU microcontroller as the control center for sensors and relays for monitoring and controlling. Each sensor is connected to a microcontroller to send data readings for each hydroponic parameter. A microcontroller controlling NodeMCU will automatically activate or deactivate relays based on real-time data analysis with threshold values for optimal hydroponic cultivation conditions. A block diagram of the system hardware architecture is shown in Figure 1.

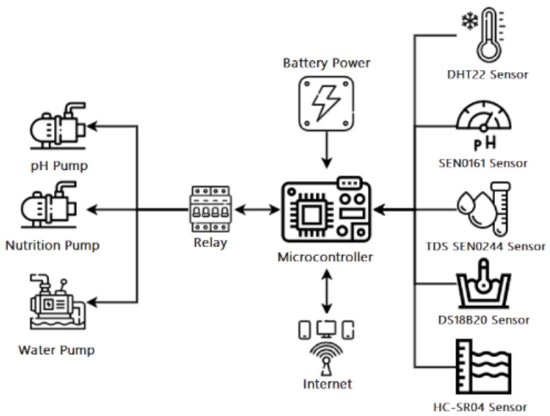


Fig. 1. Hardware architecture design

The results of the hardware architecture design become a reference for compiling the cable circuit design, which visualizes each component's integration. The cable circuit design result is shown in Figure 2.

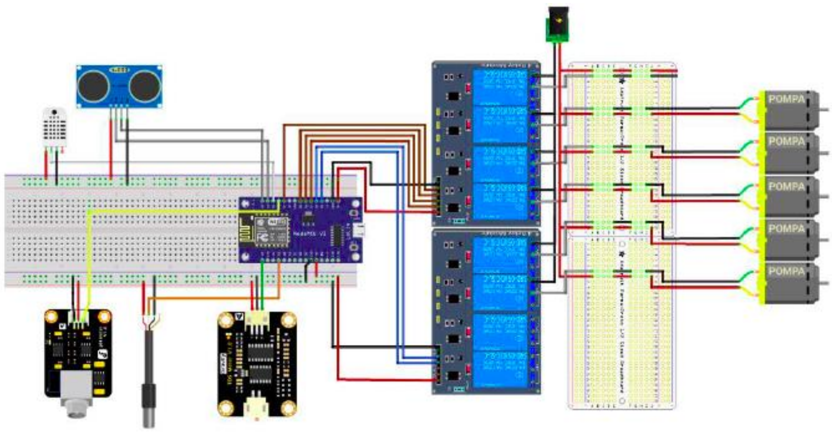


Fig. 2. Wiring circuit design

An architecture model describes the interactions and results between entities in a system. This system's architecture model is shown in Figure 3.

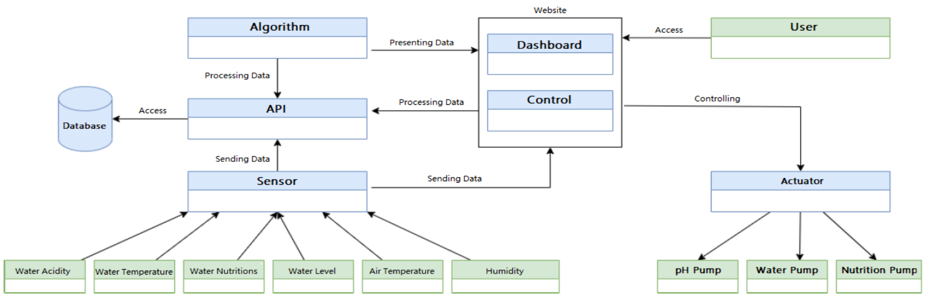


Fig. 3. System architecture design

Users can monitor the condition of the hydroponic medium on the Dashboard and operate the actuator via control from a computer, laptop, cellphone, or other gadget. Control is part of the web that will function as a control center for the actuator, while Dashboard is part of the web that will display the condition of the planting media and the status of the actuator in real-time. The sensor will periodically send data published on the MQTT protocol. The sensors in the system consist of a pH sensor, water temperature sensor, nutrient level sensor, water level sensor, temperature sensor, and air humidity sensor. The API will function as a portal to access the database, providing easy access to various entities' data. The database is the media that will be used to store all data in this system. The algorithms in the system will be split into two sections. The first is the algorithm for storing data in a database, and the second is the logic for automating actuator control.

3 Result and Discussion

3.1 User and functional requirement

Analyzing user and functional requirements is central to developing a system oriented towards hydroponic cultivation solutions. This process began with systematically observing the hydroponic cultivation practices carried out by our partner, Putra Kebun. The observation results successfully identified various challenges and obstacles in the cultivation process, such as nutrient management, environmental control, and plant health monitoring. These observations provided direct insight into the partner's operational dynamics and revealed urgent needs for improvement and innovation.

Table 1. User and functional requirement

Code	User Requirement	Code	Functional Requirement
UR-1	Users have privacy regarding the use of the system related to hydroponic environmental management data.	FR-1	The system is able to authenticate (log in) users.
		FR-2	The system is able to validate (log out) users
UR-2	Users have access to real-time hydroponic environmental/plant health data.	FR-3	The system is able to display data from sensors and NodeMCU in real-time
UR-3	Users have access to automatic hydroponic environmental management efforts.	FR-4	The system is able to manage actuator devices automatically
UR-4	Users have access to manual hydroponic environmental management efforts.	FR-5	The system is able to manage actuator devices manually
UR-5	Users can access the status of hydroponic environmental management data by the system.	FR-6	The system is able to display the status (automatic) of hydroponic environment management
		FR-7	The system is able to display the status (manual) of hydroponic environment management

An in-depth interview with Putra Kebun's owner was conducted to explore hopes and aspirations related to developing a system that can support the efficiency and effectiveness of hydroponic cultivation. In the interview session, the owner conveyed various specific needs, such as the ability to monitor plant conditions in real time, automatic irrigation settings, and analytical features that can provide valuable information for decision-making. Combining data from observations and interviews, we formulated clear and measurable functional requirements, which served as the basis for system design. The results of identifying user and functional requirements are shown in Table 1.

3.2 Hardware implementation

The results of the hardware architecture design (Figure 1) and cable circuit design (Figure 2) are realised by integrating each component, as shown in Figure 4.

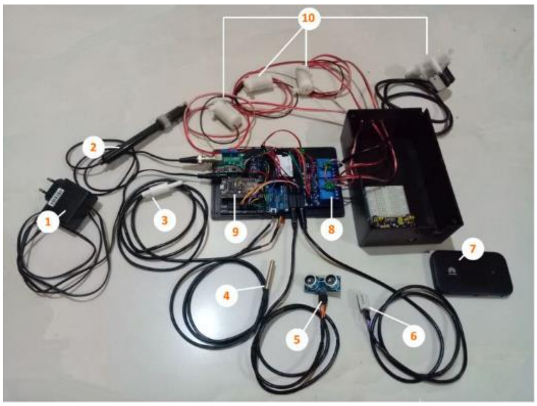


Fig. 4. System architecture design

In the hardware architecture of this hydroponic environmental monitoring and control system, several main components work synergistically to ensure optimal ecological management. This system utilizes a 12 Volt DC adapter as the main power supply (Number 1), which provides the energy needed to operate all devices. The sensors used in this system include the SEN0161 pH sensor (Number 2), the SEN0244 TDS sensor (Number 3), and the DS18B20 water temperature sensor (Number 4), which collectively measure critical parameters such as pH, Total Dissolved Solids (TDS), and water temperature, which provide essential data for monitoring the hydroponic quality.

The HC-SR04 ultrasonic sensor (Number 5) detects distance and water level, while the DHT22 sensor (Number 6) monitors humidity and air temperature. The data collected from all these sensors is then sent to the NodeMCU microcontroller (Number 9), which acts as the control and data processing centre. This microcontroller uses the MQTT protocol to transmit data to the server, which can be accessed and displayed via a web interface.

In order to control various external devices, the system is equipped with a relay (Number 8) that allows control of actuators (Number 10), which may include pumps or lighting systems in a hydroponic environment. The mobile modem (Mifi) (Number 7) is a network connector, providing the internet connectivity needed for real-time data communication. Consequently, the system monitors environmental conditions continuously and allows automatic control to maintain ideal conditions.

The components of the IoT-based hydroponic systems consume electrical energy, as shown in Table 2.

Table 2. Energy consumption specifications of each component

No	Component	Voltage	Amperre	Watt	Pcs	Active Mode (Watt)	Idle Mode (Watt)
1	Sensor pH SEN0161	5.00	0.005	0.025	1	0.025	0.025
2	Sensor TDS SEN0244	5.00	0.01	0.05	1	0.05	0.05
3	Sensor Suhu Air DS18B20	3.30	0.0015	0.00495	1	0.00495	0.00495
4	Sensor Ultrasonik HC-SR04	5.00	0.015	0.075	1	0.075	0.075
5	Sensor DHT22	3.30	0.0025	0.00825	1	0.00825	0.00825
6	Relay	5.00	0.10	0.50	4	2.00	0.00
7	Mikrokontroler NodeMCU	3.30	0.30	0.99	1	0.99	0.99
8	Submarline and Solenoid Pump	12.00	1.00	12.00	4	48.00	0.00
Total						51.1532	1.1532

Table 2 details each component's energy consumption, including specific microcontrollers, sensors, and actuators. Analysis of the use of electrical energy by each system component identifies varying power requirements in idle and active modes. In idle mode, the power is generally lower when the device is not actively operating but remains connected and ready for action. The pump's maximum power at idle was measured at 1.1532 Watts. This condition shows that even though the pump is inactive, the system still requires energy to maintain monitoring and control functions. The power requirement significantly increases when components run continuously, such as pumps supplying nutrients or sensors collecting data. Each component that functions integratively consumes 51.1532 Watts. The system consumes considerable electricity while actively functioning to ensure optimal plant growth.

3.3 Software implementation

In the context of developing an Internet of Things (IoT) technology system for hydroponic cultivation, the data interaction model and system architecture implemented rely on cable media as the primary means of communication and utilize the JavaScript Object Notation (JSON) format and the Message Queuing Telemetry Transport (MQTT) protocol. JSON is an efficient format for storing and sending data, thus facilitating the exchange of information between various components in the system. This JSON implementation is carried out on every data-sending activity in the system, which allows the delivery of information in a structured and easy-to-understand manner. Data in JSON format is taken from the database using the Postman application, which tests and verifies communication between clients and servers. This form of network communication between clients and servers uses the HTTP protocol, the standard in data transfer on the web. JSON data taken from the database using the Postman application and the form of network communication between the client and server using the HTTP protocol is shown in Figure 5.

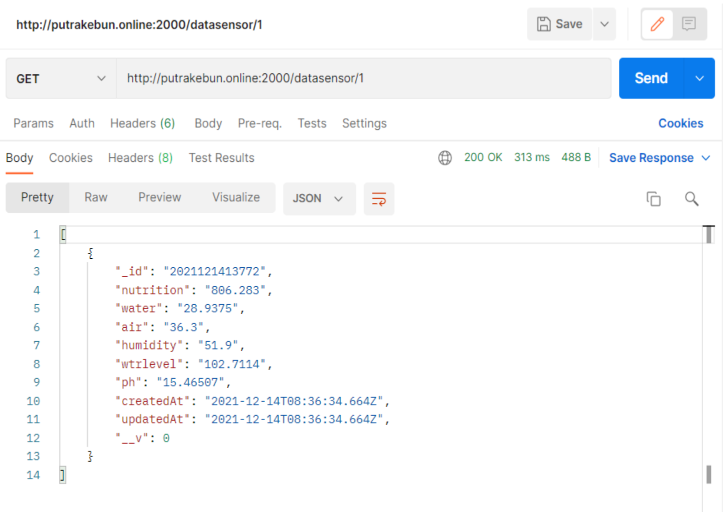


Fig. 5. JSON implementation

Message Queuing Telemetry Transport (MQTT) is a connectivity protocol specifically designed for communication between microcontrollers and applications in the context of Internet of Things (IoT) systems. This protocol offers an efficient and lightweight method for sending messages in environments with limited bandwidth and high latency, making it very suitable for applications that require remote monitoring and control, such as in hydroponic cultivation. This study used the MQTT Broker from Eclipse Mosquitto (EMQ), which was operated on an Ubuntu 20.04-based server. Using EMQ as an MQTT broker provides advantages in terms of scalability and reliability and the ability to handle many simultaneous connections with minimal latency. Data delivery trials using the microcontroller's MQTT are shown in Figure 6.

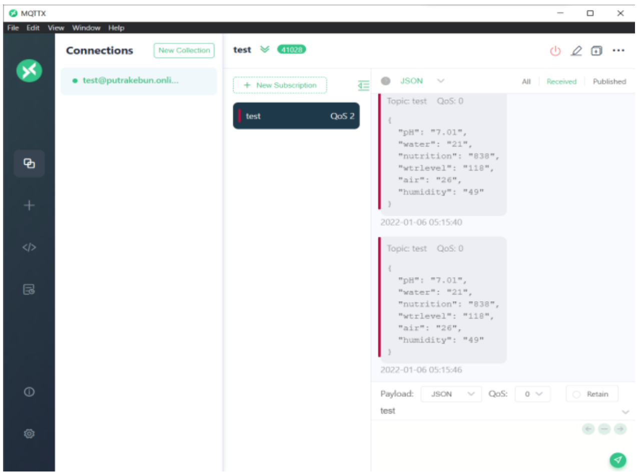


Fig. 6. Data transfer using the MQTT protocol

This study's proposed system data storage model will implement an infrastructure consisting of a web server, cloud-based data storage, and a Representational State Transfer (REST API)-based application programming interface. The web server is a service provider for the application being built, providing an efficient data management and distribution environment. In this case, the web server is implemented using the Nginx service installed on a server with the Ubuntu 20.04 operating system. Nginx was chosen because of its superior ability to handle simultaneous requests with low latency and ease of configuration and scalability, which are very important for data-oriented applications in hydroponic cultivation. In addition, using the REST API allows flexible interaction between system components, providing ease in accessing and manipulating data stored in the cloud. The result of the Nginx implementation as a web server is shown in Figure 7.

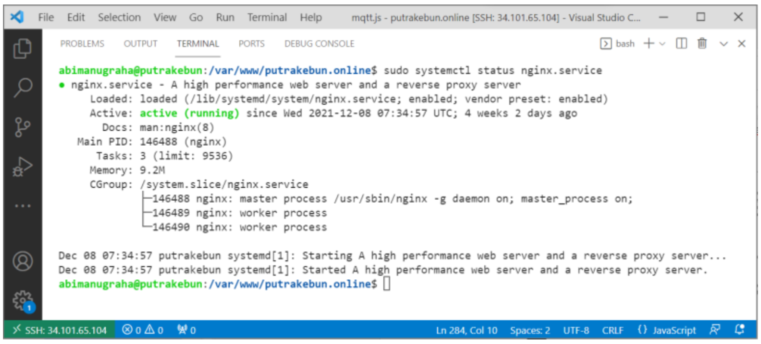


Fig. 7. Nginx implementation

Cloud-based data storage is an essential service provider in the IoT technology system for hydroponic cultivation. In this study, cloud data storage was implemented by utilizing the NoSQL database from MongoDB, which was operated on an Ubuntu 20.04-based server. The selection of MongoDB as a database management system was based on its ability to handle unstructured and semi-structured data and the ease of scalability it offers. MongoDB enables large-scale data management with high efficiency and provides flexibility in data storage and retrieval, which is very important in applications that require real-time data monitoring and analysis. Using MongoDB, the system can store various data types, from environmental parameters to analysis results, in an easy-to-access and managed format. In addition, integrating cloud-based data storage with other system components ensures data availability and security, which are crucial aspects in modern hydroponic cultivation management. The results of this cloud data storage implementation, which show the database operations that have been running on the server, are shown in Figure 8.

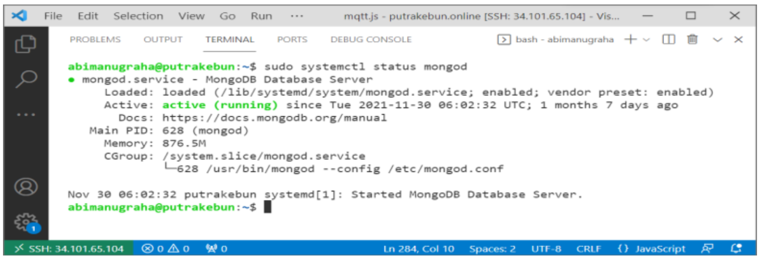


Fig. 8. Cloud data storage implementation

In this study, cloud technology will be implemented, and the Google Cloud Platform will be used as the central infrastructure. Google Cloud Platform provides various computing services that enable efficient data processing, scalable storage, and optimal resource management. The configuration of virtual machine instances (VM Instances) implemented on the Google Cloud Platform will enable the effective management of complex workloads and support system operations in real time. By utilizing cloud capabilities, this system can automate the data-based decision-making process and improve collaboration between IoT devices integrated into the hydroponic ecosystem. The result of the VM Instance configuration that has been running on the Google Cloud Platform is shown in Figure 9.

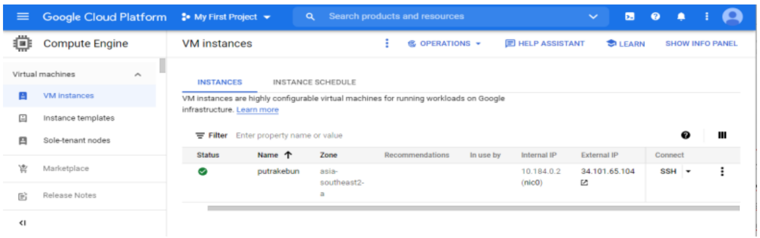


Fig. 9. VM Instance configuration

In developing the hydroponic system, the algorithm and REST API were carefully created using Node.js technology, which allows it to operate in real time in a cloud environment. This real-time capability ensures that the system can immediately respond to dynamic changes in environmental conditions, such as temperature, humidity, and nutrient levels, thereby improving the overall efficiency of plant growth. In an effort to facilitate the operation and management of the system to be always on, the system implements PM2 as a robust process manager designed explicitly for Node.js. PM2 ensures that the algorithm and API remain active and responsive and provides comprehensive monitoring features, allowing users to track real-time performance metrics. Using PM2, users can easily manage application processes, enable automatic restarts in case of failure, and optimize resource usage. The monitoring results obtained through PM2 are visually shown in Figure 10.

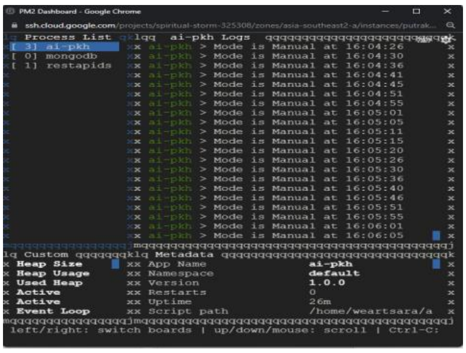


Fig. 10. P2M dashboard

The final result of the software creation is a web-based technology that displays graphical information on the hydroponic environment and pump activation status in real time. The hydroponic environmental condition data displayed includes nutrient levels, water pH, water height, water temperature, air temperature, and air humidity. Every data obtained will be automatically studied and compared with the threshold value to perform optimal hydroponic environmental control automation. The result of the Dashboard interface is shown in Figure 11.

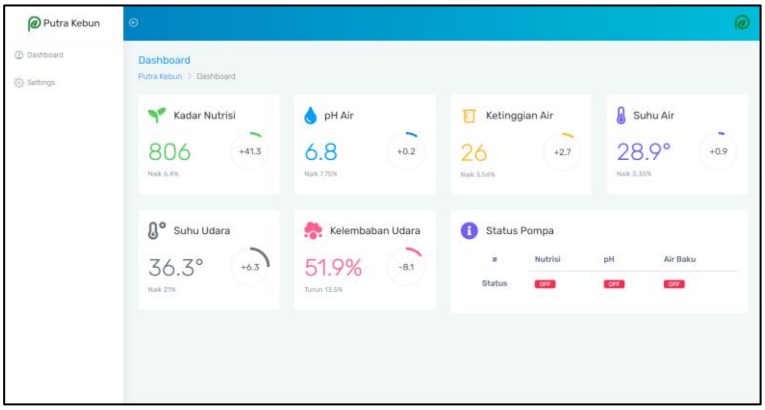


Fig. 11. Dashboard interface

Automating hydroponic environmental control requires determining threshold values related to water pH variables, water level, and nutrient levels. Attributes that must be added to each variable include the maximum threshold value, minimum threshold value, water control frequency, and waiting time duration. The result of the Settings display is shown in Fig 12.

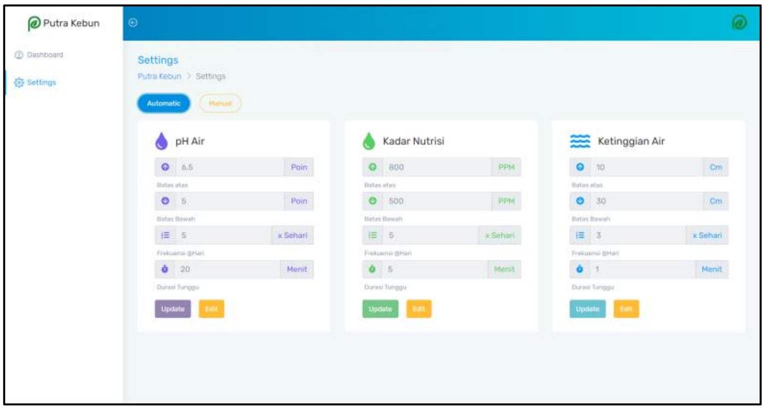


Fig. 12. Configuration interface

3.4 Functional system evaluation

Functional testing of the web-based hydroponic environmental monitoring and control system was carried out using a black-box test approach, which assesses the system's ability to meet predetermined functionality without considering its internal mechanisms [23]. In this context, the cultivation business actors act as test respondents, providing practical and contextual evaluations of system performance. The testing instruments systematically validate each functional element of the system (Table 1), ensuring that each feature and function operates according to predetermined specifications and effectively supports hydroponic environmental monitoring and control activities. The interpretation of the black-box test is shown in Table 3.

Table 3. Interpretation of the black-box test

Test ID	Funtion Name	Description	Expected Result	Final Result
1	[FR-1] Login	*Requirement: username = pkh-4DM; password = adm-Pkh1*		
		1. Write the registered username	The username text appears	ok
		2. Write the registered password	The *** character appears	ok
		3. Click the log in button	The dashboard display appears	ok
2	[FR-3] Data Moni-toring	*Requirement: Test ID 1 has been completed*		
		4. Click the dashboard menu	Cards appear for each parameter, namely pH, nutrient content, water level, water temperature, air tem-perature, air humidity and water pump status, pH and nutrients	ok
			Parameter values change every <5 sec-onds	ok
3	[FR-6] Automatic Mode Status	*Requirement: Test ID 1 has been completed*		
		5. Click the control menu	Automatic and manual buttons appear	ok
		6. Click the automatic button	Confirm alert appears	ok
		7. Click "Ok"	Automatic button turns blue	ok
			pH, nutrient, and water level cards ap-pear	ok
4	[FR-7] Manual Mode Status	*Requirement: Test ID 1 has been completed*		
		8. Click the control menu	Automatic and manual buttons appear	ok
		9. Click the manual button	Confirm alert appears	ok
		10. Click "Ok"	The manual button turns orange	ok
			The pump status card displays pH, nu-trients, and water level.	ok
5	[FR-4] Setting	*Requirement: Test ID 1 and 3 have been completed*		
		11. Click the edit button on one of the cards (pH, nutrients, water level)	Confirm alert appears	ok
		12. Click "Ok"	Setting form opens	ok
		13. Write the upper threshold value	Number display appears	ok
		14. Write the lower threshold value	Number display appears	ok
		15. Write the frequency	Number display appears	ok
		16. Write the duration time	Number display appears	ok
		17. Click "Update"	Confirm alert appears	ok
		18. Click "Ok"	Success message appears	ok
		Requirement: Settings canceled		
		19. Click "Cancel"	Confirm alert appears	ok
		20. Click "Ok"	The form is disabled	ok
6	[FR-4] Manual Ac-tuator Set-ting	*Requirement: Test ID 1 and 4 have been completed*		
		21. Click the switch according to the actuator you want to activate / deactivate	Confirm alert appears	ok
		22. Write the active/deactivate duration	Number display appears	ok
		23. Click "Ok"	Switch changes to on/off	ok
7	[FR-5] Automatic Actuator Setting	*Requirement: Test ID 1, 3 and 5 have been completed*		
		24. Click the dashboard menu and ob-serve until the parameter value is out-side the specified conditions	There is an automatic change in pump status according to the specified condi-tions	ok
8	[FR-2] Logout	*Requirement: Test ID 1 has been completed*		
		25. Click "Log out"	The login screen appears	ok

Test results will be interpreted by referring to the percentage of success calculated based on formula 1. The analysis of 25 test scenarios showed that 100% of the system's functionality was declared valid, indicating that all features operated flawlessly according to the established requirements.

$$Eligibility\ Percentages = \frac{Test\ Score}{Maximum\ Score} \quad (1)$$

4 Conclusion

The automated hydroponic mechanisms system was successfully built by implementing the MQTT protocol. Each hardware component of the system is integrated via the internet network with responsive web technology. Testing functional aspects of web technology using a black-box test approach consisting of 8 test cases and 25 scenarios to identify successful hardware-software integration. Web technology allows users to monitor and control every parameter in real-time using hardware in a hydroponic environment. The results of this research will be used in future research to test the sensor accuracy and reliability of MQTT protocol services. Further studies are also essential to discuss aspects of energy management to improve overall system efficiency. Developers can design more efficient and sustainable systems by monitoring power consumption in active and idle modes.

5 Acknowledgment

The authors thank the Research Institute and Community Service at Universitas Ahmad Dahlan for awarding the PD-236/SP3/LPPM-UAD/VII/2022 grant number.

6 References

[1] E. T. Yuniarsih, M. Salam, M. H. Jamil, and A. Nixia Tenriawaru, "Determinants determining the adoption of technological innovation of urban farming: Employing binary logistic regression model in examining Rogers' framework," *J. Open Innov. Technol. Mark. Complex.*, vol. 10, no. 2, p. 100307, Jun. 2024, doi: 10.1016/j.joitmc.2024.100307.

[2] D. H. Vo, A. T. Vo, and C. M. Ho, "Urbanization and renewable energy consumption in the emerging ASEAN markets: A comparison between short and long-run effects," *Heliyon*, vol. 10, no. 9, p. e30243, May 2024, doi: 10.1016/j.heliyon.2024.e30243.

[3] E. B. Nchanji and Y. K. Nchanji, "Urban farmers coping strategies in the wake of urbanization and changing market in Tamale, Northern Ghana," *Land use policy*, vol. 121, p. 106312, Oct. 2022, doi: 10.1016/j.landusepol.2022.106312.

[4] A. Drottberger, Y. Zhang, J. W. H. Yong, and M.-C. Dubois, "Urban farming with rooftop greenhouses: A systematic literature review," *Renew. Sustain. Energy Rev.*, vol. 188, p. 113884, Dec. 2023, doi: 10.1016/j.rser.2023.113884.

[5] R. Sharma, S. Wahbeh, B. Sundarakani, I. Manikas, and M. Pachayappan, "Enhancing domestic food supply in the UAE: A framework for technology-driven urban farming systems," *J. Clean. Prod.*, vol. 434, p. 139823, Jan. 2024, doi: 10.1016/j.jclepro.2023.139823.

[6] G. Califano, A. Crichton-Fock, and C. Spence, "Consumer perceptions and preferences for urban farming, hydroponics, and robotic cultivation: A case study on parsley," *Futur. Foods*, vol. 9, p. 100353, Jun. 2024, doi: 10.1016/j.fufo.2024.100353.

[7] S. A. Salim, S. H. Salim, and F. M. Razali, "Structured Literature Reviews (SLR) of Urban Farming for Improving Economic Status of Urban Residents," *Int. J. Sustain. Constr. Eng. Technol.*, vol. 12, no. 5, pp. 1–8, 2021.

[8] A. Abdillah, I. Widianingsih, R. A. Buchari, and H. Nurasa, "Implications of urban farming on urban resilience in Indonesia: Systematic literature review and research identification," *Cogent Food Agric.*, vol. 9, no. 1, Dec. 2023, doi: 10.1080/23311932.2023.2216484.

[9] D. Jo and G. J. Kim, "IoT + AR: pervasive and augmented environments for 'Digi-log' shopping experience," *Human-centric Comput. Inf. Sci.*, vol. 9, no. 1, pp. 1–17, 2019, [Online]. Available: file:///C:/Users/D. Yulianto/Downloads/s13673-018-0162-5.pdf.

[10] A. K. Ng and R. Mahkeswaran, "Emerging and Disruptive Technologies for Urban Farming: A Review and Assessment," *J. Phys. Conf. Ser.*, vol. 2003, no. 1, p. 012008, Aug. 2021, doi: 10.1088/1742-6596/2003/1/012008.

[11] J. Jones, *Complete Guide for Growing Plants Hydroponically*. CRC Press, 2014.
[12] S. Khan, A. Purohit, and N. Vadsaria, "Hydroponics: current and future state of the art in farming," *J. Plant Nutr.*, vol. 44, no. 10, pp. 1515–1538, Jun. 2021, doi: 10.1080/01904167.2020.1860217.
[13] H. Kopetz and W. Steiner, "Internet of Things," in *Real-Time Systems*, Cham: Springer International Publishing, 2022, pp. 325–341.
[14] N. V. Nikolov, A. Z. Atanasov, B. I. Evstatiev, V. N. Vladut, and S.-S. Biris, "Design of a Small-Scale Hydroponic System for Indoor Farming of Leafy Vegetables," *Agriculture*, vol. 13, no. 6, p. 1191, Jun. 2023, doi: 10.3390/agriculture13061191.
[15] R. E. Putri, N. U. Lestari, Ifmalinda, F. Arlius, I. Putri, and A. Hasan, "Monitoring and Controlling System of Smart Mini Greenhouse Based on Internet of Things (IoT) for Spinach Plant (Amaranthus sp.)," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 14, no. 1, pp. 131–136, Feb. 2024, doi: 10.18517/ijaseit.14.1.18408.
[16] M. A. Rahman, N. R. Chakraborty, A. Sufiun, S. K. Banshal, and F. R. Tajnin, "An AIoT-based hydroponic system for crop recommendation and nutrient parameter monitorization," *Smart Agric. Technol.*, vol. 8, p. 100472, Aug. 2024, doi: 10.1016/j.atech.2024.100472.
[17] A. Shrivastava, C. K. Nayak, R. Dilip, S. R. Samal, S. Rout, and S. M. Ashfaque, "Automatic robotic system design and development for vertical hydroponic farming using IoT and big data analysis," *Mater. Today Proc.*, vol. 80, pp. 3546–3553, 2023, doi: 10.1016/j.matpr.2021.07.294.
[18] S. N. Deshpande and R. M. Jogdand, "A novel scheduling algorithm development and analysis for heterogeneous IoT protocol control system to achieve SCADA optimization: a next generation post covid solution," *Int. J. Inf. Technol.*, vol. 15, no. 4, pp. 2123–2131, Apr. 2023, doi: 10.1007/s41870-023-01267-x.
[19] O. Rexfelt and E. Rosenblad, "The progress of user requirements through a software development project," *Int. J. Ind. Ergon.*, vol. 36, no. 1, pp. 73–81, Jan. 2006, doi: 10.1016/j.ergon.2005.08.002.
[20] K. Wiegers and J. Beatty, *Software Requirements*, 3rd ed. Redmond, Washington: Microsoft Press, 2013.
[21] A. T. Bahill and A. M. Madni, "Discovering System Requirements," in *Tradeoff Decisions in System Design*, Cham: Springer International Publishing, 2017, pp. 373–457.
[22] C. Verdouw, H. Sundmaeker, B. Tekinerdogan, D. Conzon, and T. Montanaro, "Architecture framework of IoT-based food and farm systems: A multiple case study," *Comput. Electron. Agric.*, vol. 165, p. 104939, Oct. 2019, doi: 10.1016/j.compag.2019.104939.
[23] S. Nidhra and J. Dondeti, "Black Box and White Box Testing Techniques – A Literature Review," *Int. J. Embed. Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012.

7 Authors

Dinan Yulianto serves as an Assistant Professor in the Department of Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Indonesia. He received his (B.Eng.) from the Universitas Ahmad Dahlan and his (M.Eng.) from the Universitas Gadjah Mada. He is currently studying in the Professional Engineering Program at Universitas Muhammadiyah Yogyakarta. He is a member of Persatuan Insinyur Indonesia, Asosiasi Pendidikan Tinggi Informatika dan Komputer, and The International Association of Engineers (email: dinan.yulianto@tif.uad.ac.id).

Abiema Febrian Nugraha received his (B.Cs.) from the Universitas Ahmad Dahlan. He currently has a professional career as a full-stack programmer in a multinational company (email: abiema1700018148@webmail.uad.ac.id).

Faisal Fajri Rahani serves as an Assistant Professor in the Department of Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Indonesia. He received his (B.Sc.) and (M.Cs.) from the Universitas Gadjah Mada. He is a member of the Asosiasi Program Studi Informatika - Perguruan Tinggi Muhammadiyah / 'Aisyiyah and The International Association of Engineers (email: faisal.fajri@tif.uad.ac.id).