Design of Soil Humidity Monitoring System Using the Internet of Things Concept and MQTT

1st I Komang Agus Ady Aryanto
Department of Computer System
lITB STIKOM Bali
Denpasar, Indonesia
agus_ady@stikom-bali.ac.id

2nd Roy Rudolf Huizen
Department of Computer System
ITB STIKOM Bali
Denpasar, Indonesia
roy@stikom-bali.ac.id

3rd Kadek Yota Ernanda Aryanto
Department of Informatics
Ganesha University of Education
Singaraja, Indonesia
yota.ernanda@undiksha.ac.id

2020 International Conference on Smart Technology and Applications (ICoSTA) 978-1-7281-3083-5/20/$31.00 ©2020 IEEE 10.1109/ICoSTA48221.2020.1570611115

Abstract— **Agriculture has been a relevant field that attracted attention in recent years caused by the growing population, which has raised the problem of food sufficiency. Therefore, one of the solutions that this study offered is to improve production results through innovations by performing an automatic plantation's watering process that can be monitored remotely. The purpose of this study is to help farmers to be able to control the condition of the soil moisture, together with the automatic watering process. The hardware used in this research is the NodeMCU Board with Wi-Fi ESP-32, Relay, and Moisture Sensors. This research applied the concept of the Internet of Things (IoT), where each sensor module was assembled to be an object that has been programmed to carry out its tasks. A fuzzy-based module was developed to regulate the pump control, measure soil moisture, and control the water pump machine. Furthermore, each object was given its own identity to facilitate the communication process of every object. This process was connecting each object through the internet network by applying a lightweight communication protocol, the MQTT protocol. The data from sensors that represented the condition of each object were displayed on a graph in real-time. There were also tables of data that can be printed into a document file. This research was able to assist farmers in the process of watering their plants. Therefore, the harvesting process had increased, with only a little maintenance cost.**

Keywords— Internet of Things, Sensor, Web, Wi-Fi, Microcontroller, Smart farming, Soil moisture

I. INTRODUCTION

Agriculture is a field of great concern nowadays due to the growing population, which has raised the food needs problems to be fulfilled. Therefore, it is necessary to develop agricultural technology products that can solve the problems yet can also reduce the cost of production and maintenance. Several conditions have to be measured to provide correct data to provide the appropriate treatments [1]. Watering plants is one of the essential parts of keeping plants to grow [2][3]. It requires a regular watering control so that the land on the plant have ideal and balanced soil moisture condition. If it is too wet or dry, the plant may get withered and later died. These problems can be solved by the presence of a technology that can control the condition of the soil moisture to ensure the subtle growth of the plant.

This research was aimed to develop a prototype product using the internet of things (IoT) technology that focused on controlling and maintaining the soil moisture and humidity. The prototype was also equipped with the automatic watering system that worked based on the data given by the sensors. The prototype was connected to the internet; therefore, it can be controlled and maintained through a computer or mobile device.

This research implemented technologies such as microcontrollers, Wifi, web technology, database system, environmental sensors, IoT-enabled aquatic technology, and cloud technology. The prototype was designed to be easily used and operated with also an easy installation process and did not require ample space in its placing. The design includes both hardware, circuit, and controller design. Each device held its own unique identity that was used in the communication within the system or with other devices. The "things" on a device, like its definition, have a close relationship with the machine-to-machine (M2M) communication [4].

In this study, an IoT-based system that can be used to regulate the communication process using the MQTT protocol [5] has been designed and implemented. It worked using the publish-subscribe scheme utilizing the protocol. In the communication, the MQTT required the transport system that can deliver its commands byte streams from client to server or vice versa [6]. In addition, the built system applied the Mosquitto application to help the process of managing data among devices. The system was developed to be the solution for the agriculture problems in recent years to improve the quality of plants' growth.

II. MATERIALS AND METHODS

In this work, we have divided the development process into two phases. The first one was the design and development of the hardware required for the system. Secondly, the design and implementation of the software and the system. The process of the whole method is shown in Fig. 1.

The hardware development separated into two stages, the design and the development stage. The design stage started with the sketching of the proposed devices, including the shape, dimensions, and material. In this stage, the requirements were all defined to give a more precise direction of the development stage. The process continued to the development stage, where the complete hardware was built, consisting of a series of sensors and acoustic sensor boards. When the electronic device had been completed, all of the hardware implementations will be integrated.

At the software stage, this work was carried out through analysis before entering into software design. Later, after the software design was complete, the activity continued with testing before it was integrated with the developed hardware. When the software was running well, its integration with the built hardware can be done. Otherwise, an improvement over

978-1-7281-3083-5/20/$31.00 ©2020 IEEE

the software was made, running on phase two of the software development. The system testing was conducted in cycles until the hardware and software could be integrated and run properly.

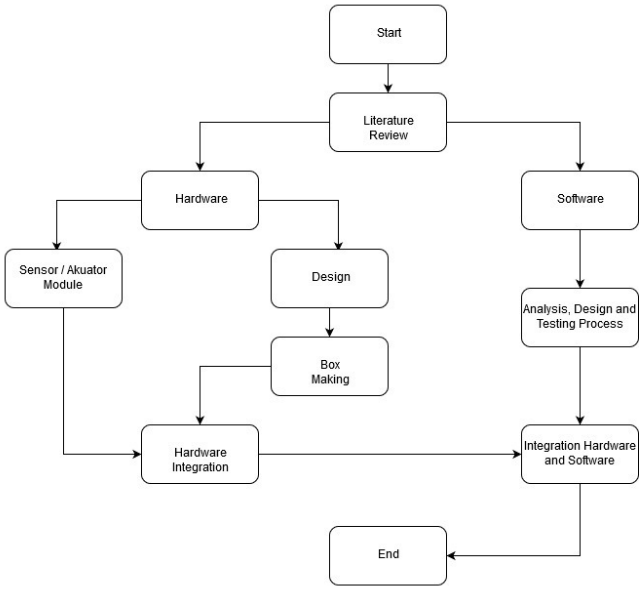

Fig. 1. The research methods

A. Hardware Design

In this study, several modules were used to support the manufacture of tools for agricultural systems. These modules were put together to form an object that had their respective tasks. The controller in each object used a NodeMCU ESP32 board. This module was a drinking system with a chip base from Espressif products, namely ESP8266, with version 32 [7]. The NodeMCU Board provides firmware that is intended to provide a platform design in innovating the manufacture of embedded systems that have Wi-fi communication capabilities. The ESP32 hardware module was used as the main module for the media where the program was embedded as well as controlling all activities involving mechanical processes in the proposed system. Therefore, through this module, the existing devices were able to communicate with servers and other devices. The humidity sensors and relay actuators were integrated into the module to control and monitor all activity conditions in the intended area; in this case, the agricultural land. In addition, communication technology on this device used 802.11 based technology, or better known as Wi-Fi. By using an internet network, it allows the system to be implemented and integrated quickly and easily on other internet network systems. Fig. 2 shows the design of the hardware used in this research.

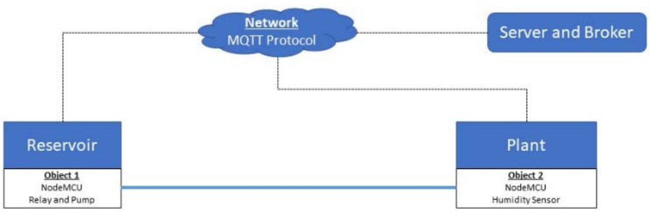

Fig. 2. The design of hardware

Communication technology on this system used 802.11 or better known as Wi-Fi. By using an internet network, allowing the system to be implemented and integrated quickly and easily with other systems. The power supply of the device used a 5V 2A voltage source. This voltage is a standard voltage and easy to find on any device that is compatible with a USB power supply.

B. Software Design

In the software module, there was a web-based interface program. Through this interface, the clients can monitor the management of all activities of each object. The interface was created using modern web languages. That was intended so that such technology could be compatible with a variety of devices with various features, making it easier for users to use the system. The design and development of software of the prototype are shown in Fig. 3.

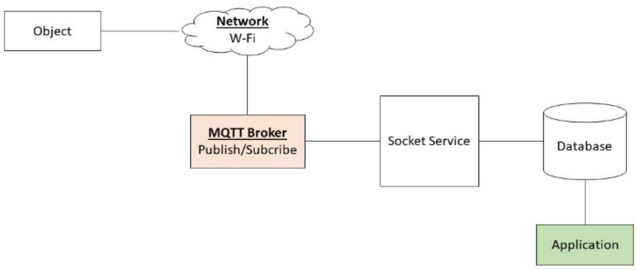

Fig. 3. The design and development of the software used in this research

The communication management process on the server was managed using an MQTT broker. It handled the process of sending and publishing data using the MQTT protocol. Message Queuing Telemetry Transport (MQTT) is a protocol that runs on TCP / IP. This protocol can be implemented in many situations, especially in the IoT area. In addition, MQTT also uses the term “topic,” which serves as a filter for the broker in sending messages to each client. In other words, it becomes a benchmark by the client to subscribe; therefore, interference between other objects does not occur in communication. Broker has a critical task since all communication processes done by objects should be connected to it first and then taken by the client through the help of socket services for further analysis, either stored in a database or displayed into a web page.

C. System Design

The work process of the measurement of soil moisture is shown in Fig. 4. The workflow was started from the tool, checking the condition of the internet network. Network configuration on the device was done by embedding the library in the microcontroller through the syntax created. The primary network configuration setting was the SSID and Password on the wi-fi network that you will use later. If the device was successfully connected to the internet, then the tool could send measurement data to the broker by publishing a topic in accordance with the identity of the device.

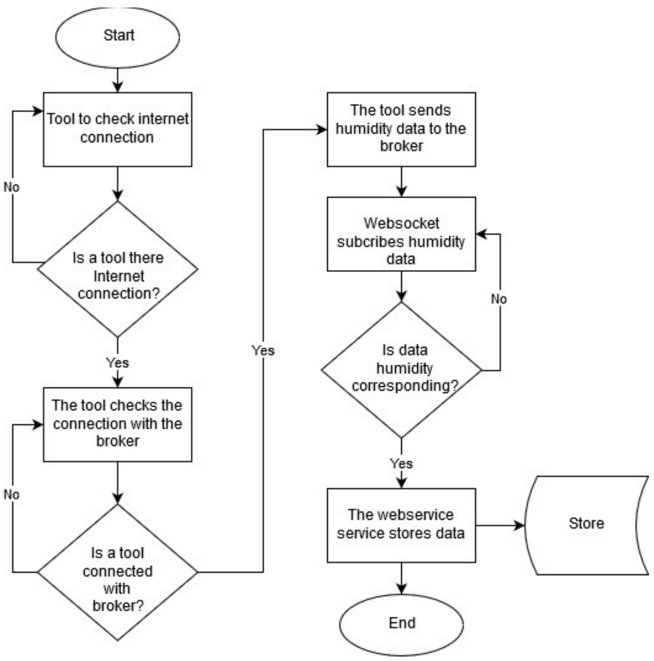

Fig. 4. The soil humidity measurement workflow

The resulting value of this moisture measurement was accommodated by a broker where all activities related to sensor data were collected. Later, it could be taken by the client. The data retrieval process in the broker by this client used a web socket service. The services were developed using current web programming languages. This web socket file worked in real-time to retrieve data from the broker by determining the topic on the subscriber. The data are taken then stored directly in the database using a web service. In addition, data could actually also be directly processed directly on web pages without the need to be stored again in a database.

The process flow for the settings of the water pump and watering system was designed to be automatically run. The flow can be seen in Fig.5.

In this research, a fuzzy-based module was used in order to regulate the pump control. In figure 5, it was shown that the pumping process was initiated by the given setpoint inputted by the user from the web page of the application. The setpoint value is the value required as the reference value by the fuzzy system. It was set by the user according to the amount of water that was needed for the plants. The value was sent to the broker and later delivered to the controlling system. The sensor has continuously measured the humidity of the soil and from the difference between the obtained value and the setpoint value resulting in the current error value. The system calculated the speed error value from the change of error values divided by the sampling time. The resulting of the fuzzy calculation then used by the actuator to regulate the intensity of the watering based on the measured condition of the soil. The process stopped when the error value has reached the value 0.

III. RESULTS

The systems were given unique identities and could communicate with other systems or devices using those identities. Each object was equipped with the NodeMCU ESP-12E module as a controller of the sensor or its actuator. This controller module was embedded with wi-fi as a communication medium over its internet wireless network. With the wireless network, communication media provides the advantages of the system that can be integrated with other objects that have an internet network. The flow of the system can be seen in Fig. 6.

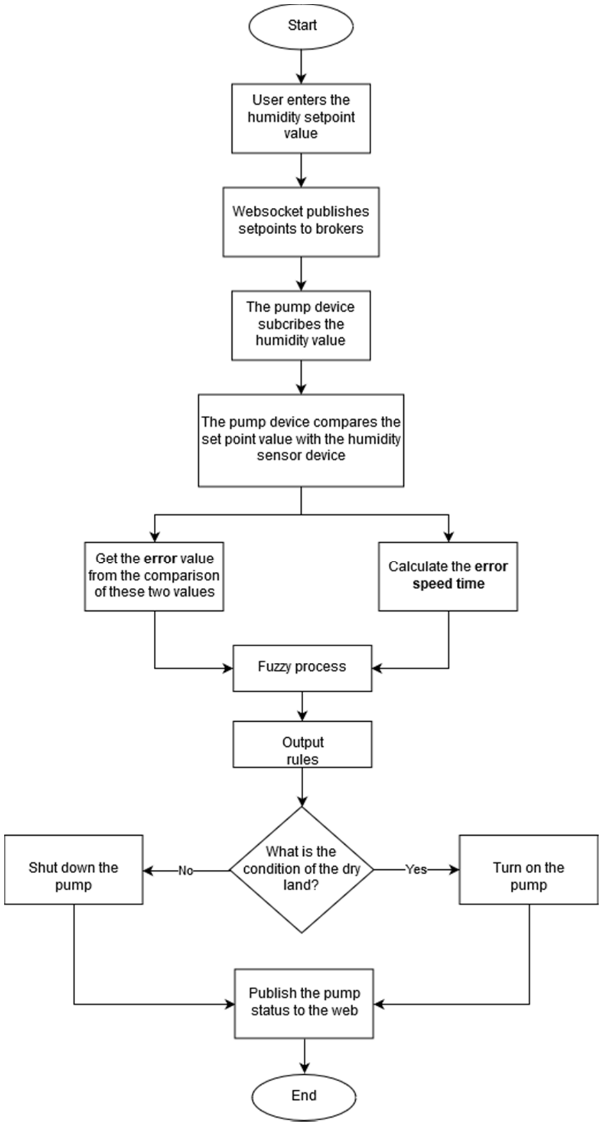

Fig. 5. Pump control process

Each of these objects can send and receive data through the Broker system by using the MQTT protocol as data transfer. MQTT can be implemented with a small bandwidth so that the data transfer process becomes lighter than using a standard protocol, and data transfer can be done in real-time. Important terms used in this protocol, such as Publish (send), Subscribe (accept), Topic, and Messages.

In giving orders, each mode was able to communicate with each other without human intervention. Therefore, the module continuously made requests and transmit data where this communication is processed at the Broker. Therefore, a particular configuration is needed on the broker's side according to system requirements.

Furthermore, on the client's side, data can be received by using Web sockets that have been configured according to the broker's identity, such as port, username, password, and broker's identification. The data received will be converted first before being displayed on a web page. Through this web page access, users see and control the condition of soil moisture. The developed web interface is responsive, which means that the web display can be adjusted according to the screen resolution of the device used to access it.

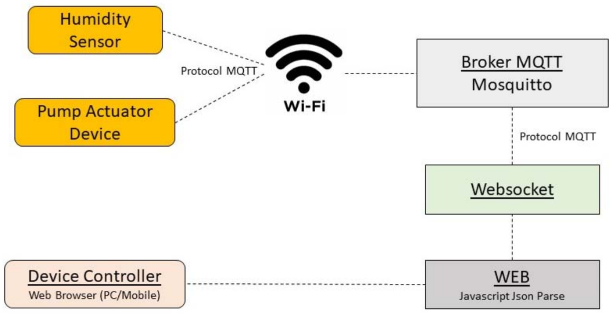

Fig. 6. The workflow of the proposed system

A. Fuzzy Logic Controller

There were two inputs used in this work in order to get the desired output. The first input (error variable) had three member functions, while the second input (speed error variable) had two member functions. There was one output obtained by the system. The basic rules of the system can be seen in TABLE I below.

TABLE I. BASIC RULES OF THE PROPOSED SYSTEM

		Input 1		
		Dry	**Good**	**Wet**
Input 2	**Dry-Speed**	Very Large	Medium	Small
	Wet-Speed	Large	Medium	Very Small

Meanwhile, Fig. 7 shown the adjacent of the membership functions to each of the inputs and output variables.

The input error variables were range between 0 and 100. The range was used in order to cover all of the possible error values. If the error given is a negative value, it means that the setpoint value is lower than the actual value indicating that the soil is dry. Meanwhile, when the value is positive, then the soil is wet.

The speed error was calculated based on the differences between the two error values at a specific time. In this research, the system used one second as the sampling time; therefore, the range of membership function of the values of the error and the speed error can be regulated.

Furthermore, the output values were also ranged between the value of 0 and 100. However, these values were not the final values of the system that were delivered to the controlling actuator. The values informed the system regarding the changes that had to be done based on the current condition.

B. Hardware implementation

The hardware that was used in the prototype consisted of controller, mechanical, and data processing parts as well as the NodeMCU ESP-12E, relay actuator, humidity sensor, and power supply that contains the power converter board. The NodeMCU ESP-12E module facilitated the communication between the device and the server. Communication technology on this system uses 802.11 or better known as Wi-Fi. By using an internet network. It allows the system to be implemented and integrated quickly and easily on other internet network systems. With a small size, this module was easy to place and did not require a large location. The power supply of the device uses a 5V 2A voltage source. This voltage is a standard voltage and is easily found on all devices that are compatible with USB power supplies.

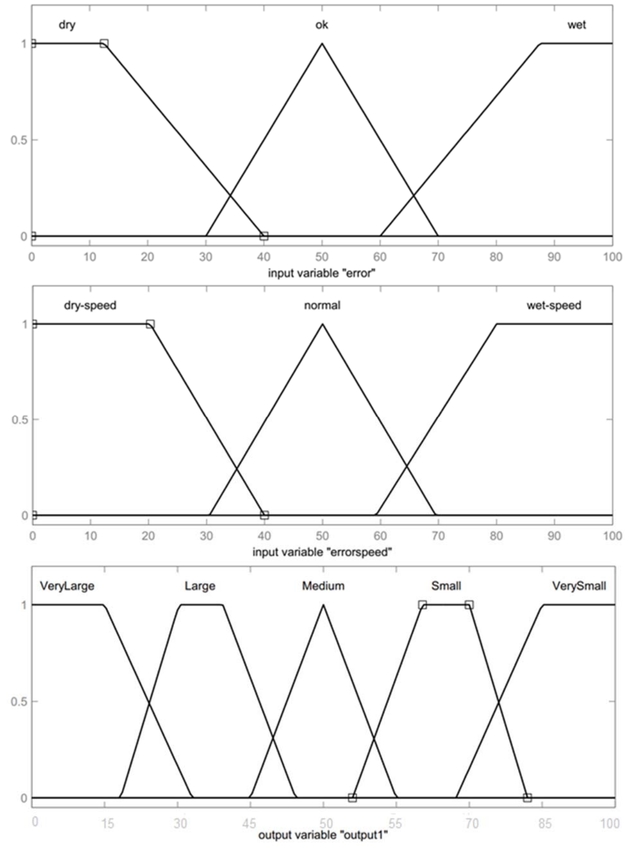

Fig. 7. Membership function adjacent to input and output variables

This module is equipped with sensors to measure soil moisture conditions. These sensors are connected to NodeMCU as well as the main process in processing data from sensor measurements. Each sensor is integrated with NodeMCU through existing pins adjusted according to the signal from the sensor, as shown in Fig. 8.

The water control module was used to control the solenoid valve device, which was related to electricity, as well as managing the water on / off for watering. This module circuit consisted of relay components that were connected to the NodeMCU, as shown in Fig. 9.

C. Software implementation

The software design is one of an important part of this work because all the processes contained in this tool were organized based on the program stored in the microcontroller and web interface. The web interface was created using a programming language consisting of Javascript, CSS, PHP, and Wiring. The web interface was designed to be responsive in order to be able to be accessed using various devices. It provided better accessibility of the system and easier to be accessed or used by the users. In addition, the implementation of firmware on the NodeMCU ESP-12E controller module and the creation of software for brokers and sockets that handle data traffic processes with the MQTT protocol. The general implementation process is as follows.

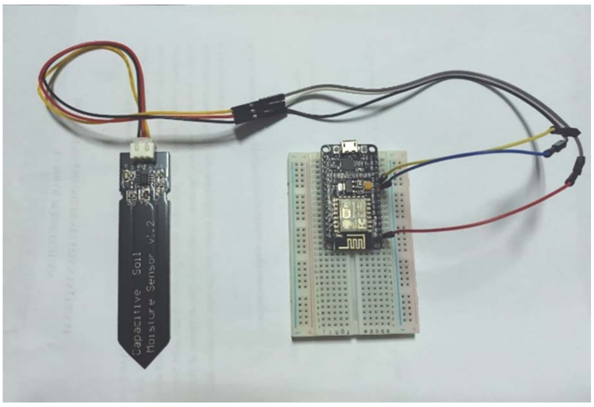

Fig. 8. The prototype design of the humidity sensor module

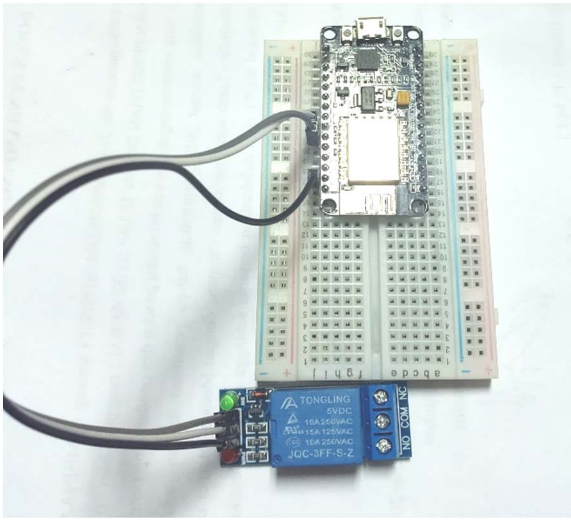

Fig. 9. The prototype design of the relay module

1) Implementation of MQTT web services

The application of the MQTT protocol in this study uses the Mosquitto broker service that supports the web socket. Data broker communication with the web socket utilized the OpenHAB service. The MQTTWS library was built using the javascript in order to be able to display the data on a web page and use the following configuration of the MQTT set with the value of host, port, path, and ID. The system stored the value of connectivity success states, including the timeout, SSL, and session. It later displayed the payload strings of the received messages.

2) Web implementation

The login page function is to carry out the account validation process to grant access rights to the main page. Therefore, only authorized users that can access the systems and prevents unauthorized actions or modules from being allowed in modifying the system.

The main page interface is shown in Fig. 10. In this page, it is equipped with graphs to make it easy for users to view the history of previous data. Moisture value using Pie graphs. In addition, the controlling process uses a button that is User-Friendly designed to facilitate the user in seeing the On or Off status of the water pump device.

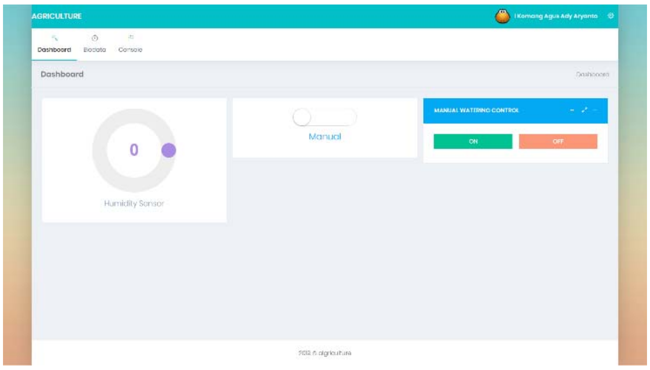

Fig. 10. The main page of the proposed monitoring system

The implemented system was able to collect data regarding the soil condition, especially related to its humidity, frequently. The data are collected through the whole year, which means the characteristics of the soil can be observed based on the changes from one season to another. In Indonesia mainly, there are two main seasons that are dry season that occurs between April until September and the rainy season from October until March. The data obtained from this work are expected to provide the base of further processing using related technology, such as big data technology. However, there are many issues that have to be addressed to ensure better data acquisition and further processing to be able to produce a meaningful result [8][9]. A scheme of data sharing could be beneficial in order to achieve better and more valid data [10]. Therefore, further research should be done for this purpose.

For further development, there are several issues that have to be addressed, as well. The watering system itself can be done in a more efficient manner involving robotic technology and automation [11]. Meanwhile, the weather has a significant influence on the farming process and production itself. Monitoring to the condition of the weather, however, should not spend too many resources [12].

In this work, the data transmissions were tested using the MQTT protocol. The tests were done in an interval of 60 seconds or one minute, where the transferred data were stored into the database. The ability of the system to deliver data within the given time was observed, both the data storing and the data transmitted from the devices to the servers, as shown in Fig. 11.

The average of 2192 data was transferred every minute, where over 2000 data were able to be transmitted using the proposed system. It was mainly caused by the ability of the MQTT protocol itself that can transmit data in small size so that the time required for data transfer can be reduced.

IV. CONCLUSION

From the work that had been conducted, it is shown that the prototype was able to control the humidity and watering system with the very least of human interaction. The system was successfully provided the information regarding the condition of the soil, especially its humidity or moisture, and delivered it to the provided web-based system.

The system interface was developed based on the web with graphic and table design so that it was easier to see the history of previous humidity data. Each object is designed to send / receive values into the broker in real-time using the MQTT protocol that can reduce the size of the data packet as small as possible so that the data transfer process becomes lighter than using the standard protocol.

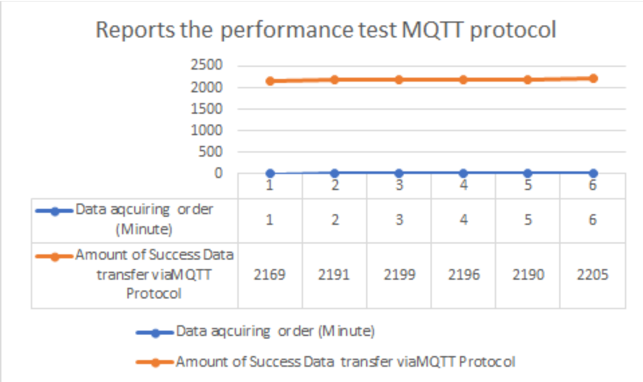

Fig. 11. Reports page of the performance test using the MQTT protocol

REFERENCES

[1] Chetan Dwarkani M ; Ganesh Ram R ; Jagannathan S ; R. Priyatharshini, "Smart farming system using sensors for agricultural task automation", 2015 IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR), 2015

[2] Gutiérrez, Villa-Medina, Nieto-Garibay, Porta-Gándara,"Automated Irrigation System Using a Wireless Sensor Network and GPRS Module", IEEE Instrumentation and Measurement Society, Vol 63, 2014

[3] K.Wardana, Crisnapati, KAA Aryanto, Krisnawijaya, Suranata,"IoT-based Drip Irrigation Monitoring and Controlling System using NodeMCU and Raspberry Pi", Atlantis Press, 2018

[4] Ovidiu Vermesan, Peter Friess, "Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers, 2013

[5] Burange, A. W., & Misalkar, H. D, "Review of Internet of Things in Development of Smart Cities with Data Management & Privacy", 2015

[6] R. J. Cohn, "MQTT Version 3.1.1", 2014

[7] Rajesh Singh,"Getting Started for Internet of Things with LaunchPad and Esp8266 (River Publishers in Information Science and Technology)", RIVER PUBL. 2019

[8] Rodriguez, D, de Voil, P, Rufino, M.C., Odendo, M., van Wijk, M.T, "To mulch or to munch? Big modeling of big data", Agricultural Systems, Volume 153, Pp 32-42, 2017

[9] Sjaak Wolfert, Lan Ge, Cor Verdouw, Marc-Jeroen Bogaardt, "Big Data in Smart Farming – A review", Agricultural Systems, Volume 153, Pages 69-80, 2017

[10] Leanne Wiseman, Jay Sanderson, Airong Zhang, Emma Jakku, "Farmers and their data: An examination of farmers' reluctance to share their data through the lens of the laws impacting smart farming", NJAS - Wageningen Journal of Life Sciences, Volumes 90–91, 2019

[11] Vaishali Lohchab ; Manish Kumar ; Gaurav Suryan ; Vishal Gautam ; Rohit Kumar Das, "A Review of IoT based Smart Farm Monitoring", 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018

[12] Sonam Tenzin ; Satetha Siyang ; Theerapat Pobkrut ; Teerakiat Kerdcharoen, "Low-cost weather station for climate-smart agriculture", 2017 9th International Conference on Knowledge and Smart Technology (KST), 2017