윤혜원 / 김동준 / 이미란 / 원채희

<https://1drv.ms/w/s!ArXTeNifb8tHipIJ5sLHbLWoDgi_9w?e=fVOvwb>

**Thesis Purpose: LoRa-MQTT based Smart Farm**

**Abstract-hyewon**

1. **INTRODUCTION - Hye Won**

**2. Related Work**

1. **LoRa - Ran**

**-Previous Work**

1. **MQTT -chaehee**

**-Previous Work**

* 1. existing studies와 우리의 차이점

LoRa 와 MQTT 기술의 특징 및 이래서 얘네를 together 써보고자 한다

증거

**3) LoRa and MQTT-dj**

**3. System Design-hyewon**

* 3.1 System Model
* 3.2 System Process
* 3.3 System Implementation Setup

-Components Required - ran, chaehee

-Draginio LoRa Shield

-LoRa Technology

-MQTT

-TTN

-Cayenne

**-sw**

**4. Results and Conclusion-dj**

**-LoRa-MQTT**

-limitation:

-Wifi is needed

-Cloud Computing is needed

**Reference**

[**https://ieeexplore.ieee.org/abstract/document/5750779**](https://ieeexplore.ieee.org/abstract/document/5750779)

**Abstract**:

Smart Farms are the future of Agriculture. They are expected to increase the quantity and quality of agricultural products with low-cost and high convenience. There has been a lot of studies on how to design a Smart Farm using Internet of Things(IoT). However, farmers are skeptical of running smart farms, due to its high cost, difficult management and potential data leakage to other competitors. Therefore, this study proposes a LoRa and MQTT based Smart Farm that has low-cost setup and maintenance fee, small battery consumption, high security, and a friendly user-interface for the farmer to conveniently manage the farm. The prototype uses two Arduino Board with Dragino LoRa Hat, in which one is connected to VH400 soil moisture sensor, and DHT11 temperature and humidity sensor, and the other connected to a solenoid valve irrigation actuator. There is a Raspberry Pi Irrigation Node, which uses Open Weather API to get 5 days forecast data. Lastly, a web based application where the farmer can see all of the data and manually activate the actuator.

1. **Introduction: (smart farm)**

According to the Food and Agriculture Organization of the United Nations, the definition of “Smart Farming is a farming management concept using modern technology to increase the quantity and quality of agricultural products.”

The key factors of smart farms are the following:

1) low cost on installation and maintenance

2) convenience on management

3) efficient data transmission

4) high security and reliability

There are various type of farming, such as crop, livestock, fish and dairy. However, this study focuses on crop farming. In crop farming, agriculture data that are vital are soil moisture, temperature, humidity and weather. Soil moisture, temperature, and humidity can be easily collected by installing sensors in farms. Past, current, and future Weather forecast could be collected from weather API(Application Programming Interface)s. There are many open source APIs that use Big Data technology that provide these information. All of the data could be stored on databases via the internet, which is useful for data analyzation. Analyzing data will help the optimization and modification of the surrounding environment and predict when the crops need water. Also, by implementing the datas in a web based application, farmers can conveniently manage their farms anywhere at anytime.

Many papers suggest using LoRa as the networking technology and LoRaWAN as the protocol, because of its low cost, one of the key factors of smart farms. Also, LoRa could communicate in long distances. The energy consumption is low compared to other networking technologies, so farmer’s don’t have to go to the farm and change batteries often.

However, there weren’t any studies which focused on the messaging protocols for smart farms. The size of the data received from sensors in smart farms is very small. TCP/IP, a high weight protocol only increases the number of waste packages. MQTT is a messaging protocol that is extremely simple and lightweight. Data transmission could be guaranteed and each topics could be assigned with passwords, which supports high security. Hence, this study proposes a smart farm based on LoRa and MQTT.

1. **Related Work:**
2. **LoRa**

When applying wireless wide area network technology, such as LTE, to the IoT devices for smart farm, it is necessary to go through a complicated process such as a gateway to access IoT devices. However, if LPWAN is used, it can be connected directly to IoT devices without going through a complicated process. As a result, LPWAN like LoRa, which can transfer less data battery-efficiently, is well suited for IoT smart farm. LoRa network is short for Long Range, which is a long-range wireless communications system. LoRa transmits small amounts of data to the IoT devices in a battery-efficient way, which makes it a fit for our study of sending small amounts of data from sensors over long distances.

Sarker et al. Demonstrated the benefits of applying LoRa technology to the IoT. Furth

Then, the data is transmitted to the gateway based on LoRa, and the Wi-Fi module of the ESP32 on the gateway is used to upload the data to the cloud server of the Internet through the MQTT protocol, which is convenient for the client to access and call the resulting data information at any time.

ermore, they have demonstrated that using an integration of cloud and edge computing is more appropriate than just using cloud computing in the LoRa IoT environment, reducing waste of resources.

Muangprathub et al. proposed a system to control environmental factors in crop fields. And they implemented web application and mobile application, it allowed for automatic and manual control. However, they used only wifi of the NodeMCU, which has a voltage of 135-215 mA. Our study is using LoRa network to develop a more efficient system in terms of power consumption.

Ko et al. demonstrated through experiments that LoRa was the most efficient among many LPWAN protocols, and found the optimal PHY configuration setting on the LoRa network. They also found that the variation of tree farms was bigger than the variation of open area and that the RSSI's Variation was more consistent in the tree farm than an open area.

~~Kodal et al. proposed a smart IoT farm model based on LoRa. Similar to our study, the system uploaded data to the cloud using the MQTT protocol, but only by monitoring sensor data.~~

1. **MQTT**

“MQTT stands for MQ Telemetry Transport. It is a publish/subscribe, extremely simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. The design principles are to minimise network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery.”

MQTT is suitable for the message exchange in building automation scenarios, with real-time constraints because it is able to support reliable message transmission[1], with timely alarm notification [2]. MQTT provides three QoS(Quality of Service) levels, using a publish/subscribe communication pattern. With MQTT, IoT devices publish or send information about a given topic to MQTT message broker. Topic is the standard that publishers and subscribers work. Topics are organized hierarchically using forward slashes (/) and efficiently manage a large number of sensor devices.

Both publisher and subscriber act as clients to the broker. The publisher connects to the broker server to publish the topic, and the subscriber connects to the broker server to subscribe the topic. More than one publisher and subscriber can connect to a broker to publish or subscribe to a topic. Multiple clients can also subscribe to one topic.

1. **LoRa MQTT**

Spinsante et al. proposed a system for building automation services using LoRa and MQTT. This study verified that LoRa suits well in indoor scenarios, and the use of MQTT fulfilled the real-time requirements of the building automation service.

Huang et al. proposed a marine wireless sensor network monitoring system based on LoRa and MQTT. This study proved that even though the sensor packages are in outdoor water environment, the data from sensors don’t affected significantly if the whole set of equipment is within a certain coverage of the wireless signal.

Kodal et al. proposed a smart IoT farm model based on LoRa. Similar to our study, the system uploads data to the cloud using the MQTT protocol, but this system only has monitoring part.

[1] H. C. Hwang, J. Park, and J. G. Shon, “Design and implementation of

a reliable message transmission system based on mqtt protocol in iot,”

Wireless Personal Communications, vol. 91, no. 4, pp. 1765–1777, 2016.

[2] A. DelCampo, E. Gambi, L. Montanini, D. Perla, L. Raffaeli, and

S. Spinsante, “Mqtt in aal systems for home monitoring of people

with dementia,” in 2016 IEEE 27th Annual International Symposium

on Personal, Indoor, and Mobile Radio Communications (PIMRC), Sept

2016, pp. 1–6.

[3] Al-Hatmi Razan, “Internet of Things: Survey and open issues of

MQTT protocol,” Proceedings - 2017 International Conference on

Engineering and MIS, ICEMIS 2017, vol.2018-January, pp.1-6,

January 29, 2018.

## **A.** **Required Components**

**TABLE I. Required Components**

|  |  |
| --- | --- |
| **Parameter** | **Sensors / Value** |
| **Temperature and Humidity** | **DHT11** |
| **Soil Moisture** | **VH400** |
| **Solenoid Valve** | **0~0.8MPa** |
| **Relay Module** | **High level trigger** |

**DHT11**

|  |  |
| --- | --- |
| **Operating Voltage** | **3.5V to 5.5V.** |
| **Operating current** | **0.3mA (measuring) 60uA (standby)** |
| **Output** | **Serial data.** |
| **Temperature Range** | **0°C to 50°C.** |
| **Humidity Range** | **20% to 90%** |
| **Resolution** | **Temperature and Humidity both are 16-bit.** |
| **Accuracy** | **±1°C and ±1%** |

**(** [**https://components101.com/dht11-temperature-sensor**](https://components101.com/dht11-temperature-sensor) **)**

**VH400**

|  |  |
| --- | --- |
| **Power consumption** | **< 13mA** |
| **Supply voltage** | **3.5V to 20 VDC.** |
| **Power on to Output stable** | **400ms** |
| **Output Impedance** | **10K ohms** |
| **Operational temperature** | **-40ºC to 85ºC** |
| **Accuracy at 25°C** | **2%** |
| **Output** | **0 to 3V related to moisture content** |
| **Shell color** | **Red** |
| **Voltage Output Curves** | [**Curves**](https://www.vegetronix.com/Curves)**,** [**Piecewise linear equations**](https://www.vegetronix.com/Products/VH400/VH400-Piecewise-Curve.phtml) |
| **Certifications** | [**CE Declaration of Conformity**](https://www.vegetronix.com/Products/VH400/VH400-RoHS-Declaration.pdf) |

**(**[**https://www.vegetronix.com/Products/VH400/**](https://www.vegetronix.com/Products/VH400/)**)**

**To get soil information from the farm, the DHT11 sensor received soil humidity and temperature, and the VH400 sensor received Soil moisture. In the transmitter device, the LoRa/GPS shield of Dragino was attached to the Arduino, and Raspberry Pi was used as a gateway to receive data from the transmitter. Edge computing nodes for the irrigation used Raspberry Pi. An actuator-operated node was attached to Arduino to LoRa/GPS shield of Dragino and was operated by the relay module of the high-level trigger and the solenoid valve.**

## **B.** **System Architecture**

**Fig. 1. Architecture of Smart Farm based on LoRa and MQTT**

**This work aims to design a smart farm that fulfills all the key factors of smart farms mentioned in page 1, and allow farmers to conveniently manage their farm by using web based application. The components are sensors, gateway, cloud, irrigation node, actuators, and web based application, as shown in Fig. 1.**

**The first component is sensors. The sensors for soil moisture, temperature, humidity, are connected to an Arduino, which has a LoRa Shield on top. These sensors obtain data from the soil and air. The sensors used in this project are shown in Table 1.**

**The second component is a LoRa gateway. This gateway was built by using a raspberry pi with a Lora/GPS Hat, and registered in TTN(The Things Network) Cloud. It receives sensor datas by Lora RF, LoraWAN, and sends it to the TTN Cloud by UDP using single\_chan\_pkt\_fwd packet forwarder.**

**The third component is TTN Cloud. “TTN is about enabling low power** [**Devices**](https://www.thethingsnetwork.org/docs/devices/) **to use long range** [**Gateways**](https://www.thethingsnetwork.org/docs/gateways/) **to connect to an**

**open-source, decentralized** [**Network**](https://www.thethingsnetwork.org/docs/network/) **to exchange data**

**with** [**Applications**](https://www.thethingsnetwork.org/docs/applications/)**.” When the cloud receives data from the packet**

**forwarder it uplinks to the web based application. Also, TTN Cloud is**

**integrated with Cayenne, an administrator interface.**

**The fourth component is a**

**irrigation node. This node is a raspberry pi which runs a Open Weather api to**

**get 5 days forecast.**

**The fifth component is**

**actuators. This actuator is a solenoid valve that is connect to an Arduino.**

**This valve is turned on automatically when the soil moisture is under 200.**

**(soil moisture range). Also, it can be turned on/off by the farmer’s decision.**

**The sixth component is a web**

**based application. The backend development was done by using Node JS, frontend**

**HTML/CSS, database MySQL version 5.6. The web application's design was**

**RWD(responsive web design), in order to meet the farmer's need to use the**

**application on various devices such as computers and smartphones. It displayed**

**the datas received from the sensors; temperature, humidity, soil moistures, and**

**whether the irrigation node was turned on/off. Also, the farmer can turn**

**on/off the actuator by choice, by clicking on a button.**

## **C.** **MQTT Diagram**

**Fig. 2. MQTT Diagram of Controlling Actuator Automatically - Sending data to irrigation node**

**Fig. 3. MQTT Diagram of Controlling Actuator Automatically- sending data to actuator**

**Fig. 4. MQTT Diagram of Controlling Actuator by Farmer’s Decision**

**“MQTT is a Client Server publish/subscribe messaging transport protocol.” There are three MQTT diagrams that is used in this architecture. The diagrams for MQTT when the actuator are activated automatically due to the soil moisture, as shown in Fig. 2. and 3. First, when sending the sensor data to the irrigation node, as shown in Fig 2. TTN (The Things Network) Cloud acts as a MQTT Broker. Irrigation node for edge computing act as a subscriber for “dev/test” topic. After appending data from the Open Weather API, it sends it to the Gateway as shown in Fig 3. The irrigation node, the raspberry pi with Mosquitto Broker installed, is the publisher of “agiot” and act as the MQTT Broker. The Gateway is the subscriber of the topic “agiot”, which receives the sensor data and weather data in a string.**

**An actuator could be activated**

**when the farmer clicks on the button in the web based application, as shown in**

**Fig 4. Web based application is a MQTT publisher of the topic “smartfarm”. The**

**gateway, the subscriber of “smartfarm” receives the farmer’s decision and sends**

**it to the actuator by LoRa. The broker for this diagram is “Eclipse Mosquitto,**

**an open source (EPL/EDL licensed) message broker that implements the MQTT**

**protocol versions 5.0.”**

## **D.** **Software Description**

**Fig. 5. Login Page**

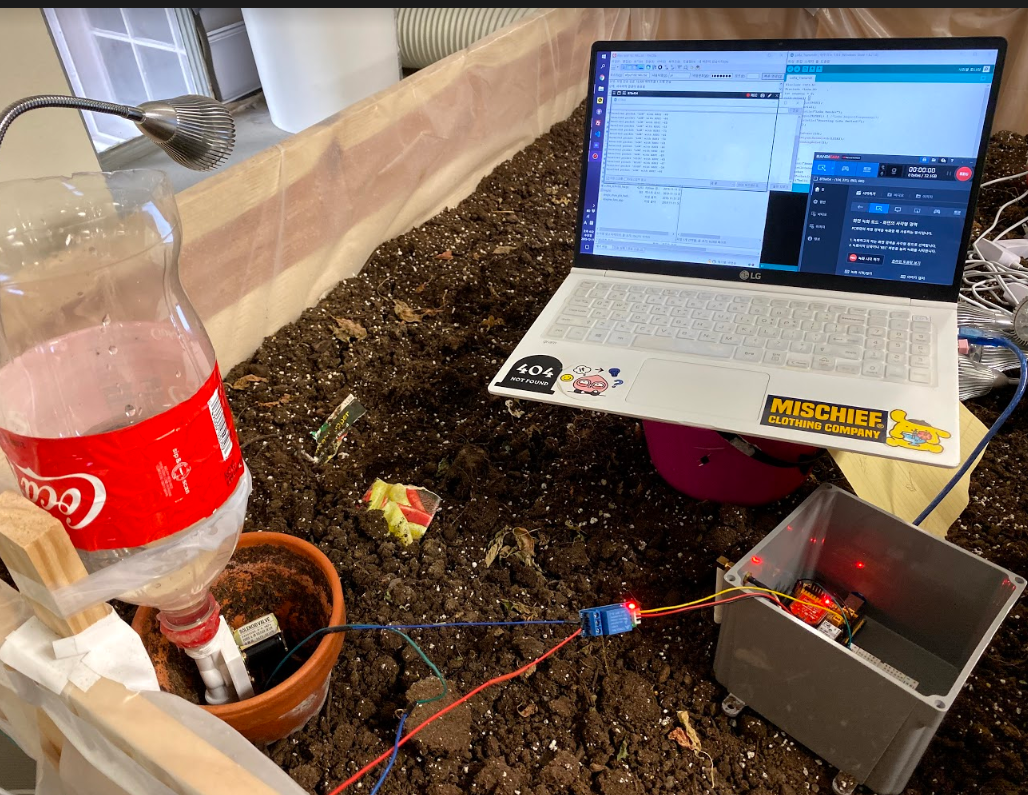
**Fig. 6. My Page**

**Fig. 7. Adding farm**

**Fig. 8. My Farm Page**

**After the farmer creates an account in the web based application, the page renders to a login page, as shown in Fig. 5. When successfully login, the farmer’s my page appears, as shown in Fig. 6. In my page, the farmer can see their person information and the farms he/she has registered. If the farmer wants to add more farms there is an adding farm button which links to farm form page, as shown in Fig 7. In this page the farm inserts information like farm’s name, location, and the type networking it uses such as LoRa or APRS. If the farmer clicks on the “Go” button of one typical farm listed in the “My Farm List”, the page renders to My Farm Page, as shown in Fig 8. In this page the farmer can see the farm’s information that the farm inserted when submitting the form. Also, temperature, humidity, soil moisture data which was sent straightly from the sensors installed at the farm. There is an irrigation “ON/OFF” button, which shows whether the actuator is turned “ON/OFF” in the farm. When the farmer clicks on the button, the actuator can be activated.**

# **II.** **Prototype**



<Prototype 환경>

The prototype was designed before installation on the farm. Since the Web server operated on a local server, the on/off state of the actuator was changed from the desktop through the Web application. From the web application, MQTT transmits an actuator status value of 0 and 1 to the Gateway. The receiver node that is connected to the actuators that finally received the data operates the actuators. The prototyping environment was conducted in a small garden 1.2 m \* 2.0 m in the K-SW Square of Purdue University.

III. Conclusion

**This paper presented a LoRa and MQTT based smart farm system monitoring condition of soil and air. The data from sensors are transferred over LoRa and it is sent directly to TTN. From TTN, data is transferred by MQTT protocol. The signal for actuator node is sent by LoRa. The farmers can directly access the monitoring result on the Web interface and control the actuator. Under excessively dry condition for the crop, the edge logic for irrigation can decide turning on or off the actuator by referring the data which is from the sensors and weather data from weather API. The signal is sent over LoRa to the actuator node. However, because of intervention of the TTN, necessity of edge computing logic is faded. Since the quality of internet connection at outdoor is not stable, and irrigation needs to be available for any time, entire ecosystem if required to keep controlling the irrigation system even though internet connection is lost. For future work, building the entire ecosystem running without internet connection could guarantee irrigation system to keep the good condition for crops.**

##### **Acknowledgment**

##### **This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the National Program for Excellence in SW(R7116-16-1014) supervised by the IITP (Institute for Information & communications Technology Promotion). References**

**[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)***

**[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.**

**[3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.**

**[4] K. Elissa, “Title of paper if known,” unpublished.**

**[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.**

**[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].**

**[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.**

**3. System Design**

1. component required

|  |  |
| --- | --- |
| **Parameter** | **Value** |
| Temperature and Humidity | DHT11 |
| Soil moisture | VH400 |
| Solenoid Valve | 0~0.8MPa |
| Relay Module | High level trigger |

To get soil information from the farm, the DHT11 sensor received soil humidity and temperature, and the VH400 sensor received Soil moisture. In the transmitter device, the LoRa/GPS shield of Dragino was attached to the Arduino, and Raspberry Pi was used as a gateway to receive data from the transmitter. Edge computing nodes for the irrigation used Raspberry Pi. An actuator-operated node was attached to Arduino to LoRa/GPS shield of Dragino and was operated by the relay module of the high-level trigger and the solenoid valve.

1. System Design

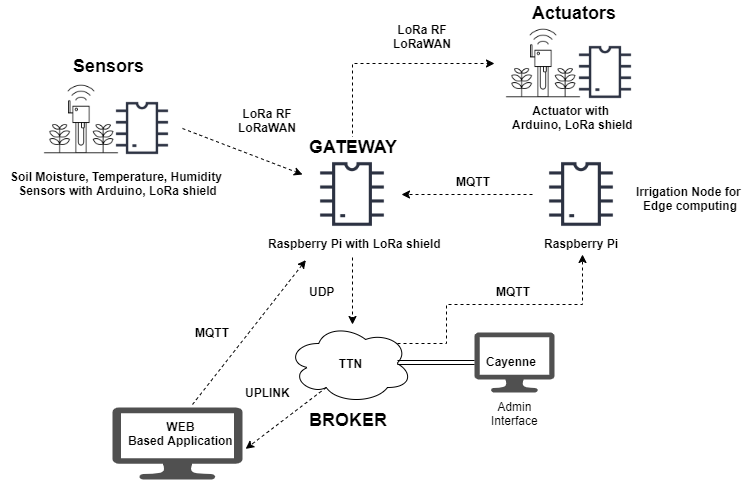


Fig. 1. Architecture of Smart Farm based on LoRa-MQTT

This work aims to design a smart farm that fulfills all the key factors of smart farms mentioned in page 1, and allow farmers to conveniently manage their farm by using web based application. The components are sensors, gateway, cloud, irrigation node, actuators, and web based application, as shown in Fig. 1.

The first component is sensors. The sensors for soil moisture, temperature, humidity, are connected to an Arduino, which has a LoRa Shield on top. These sensors obtain data from the soil and air. The sensors used in this project are shown in Table 1.

The second component is a LoRa gateway. This gateway was built by using a raspberry pi with a Lora/GPS Hat, and registered in TTN(The Things Network) Cloud. It receives sensor datas by Lora RF, LoraWAN, and sends it to the TTN Cloud by UDP using single\_chan\_pkt\_fwd packet forwarder.

The third component is TTN Cloud. “TTN is about enabling low power [Devices](https://www.thethingsnetwork.org/docs/devices/) to use long range [Gateways](https://www.thethingsnetwork.org/docs/gateways/) to connect to an open-source, decentralized [Network](https://www.thethingsnetwork.org/docs/network/) to exchange data with [Applications](https://www.thethingsnetwork.org/docs/applications/).” When the cloud receives data from the packet forwarder it uplinks to the web based application. Also, TTN Cloud is integrated with Cayenne, an administrator interface.

The fourth component is a irrigation node. This node is a raspberry pi which runs a Open Weather api to get 5 days forecast.

The fifth component is actuators. This actuator is a solenoid valve that is connect to an Arduino. This valve is turned on automatically when the soil moisture is under 200. (soil moisture range). Also, it can be turned on/off by the farmer’s decision.

The sixth component is a web based application. The backend development was done by using Node JS, frontend HTML/CSS, database MySQL version 5.6. The web application's design was RWD(responsive web design), in order to meet the farmer's need to use the application on various devices such as computers and smartphones. It displayed the datas received from the sensors; temperature, humidity, soil moistures, and whether the irrigation node was turned on/off. Also, the farmer can turn on/off the actuator by choice, by clicking on a button.

1. MQTT Diagram

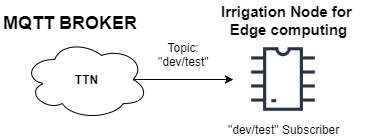


Fig. 2. MQTT Diagram of Controlling Actuator Automatically - Sending data to irrigation node

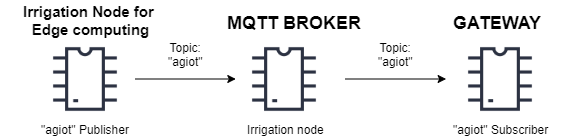


Fig. 3. MQTT Diagram of Controlling Actuator Automatically- sending data to actuator

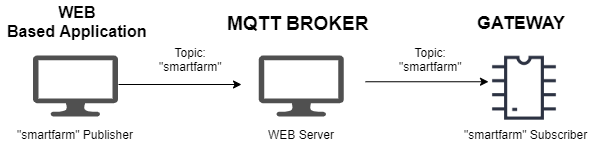


Fig. 4. MQTT Diagram of Controlling Actuator by Farmer’s Decision

“MQTT is a Client Server publish/subscribe messaging transport protocol.” There are two mqtt diagrams that is used in this architecture. The first diagram is a MQTT Diagram when the actuator are activated automatically due to the soil moisture, as shown in Fig. 2. and 3. First, when sending the sensor data to the irrigation node as shown in Fig 2., TTN (The Things Network) Cloud acts as a MQTT Broker. Irrigation node for edge computing act as a subscriber for “dev/test” topic. After appending datas from the Open Weather Api, it sends it to the Gateway as shown in Fig 3. The irrigation node, the raspberry pi with Mosquitto Broker installed, is the publisher of “agiot” and act as the MQTT Broker. The Gateway is the subscriber of the topic “agiot”, which receives the sensor data and weather data in a string.

Fig 4. is a MQTT Diagram when the actuator are activated when the farmer click on the button in the web based application. Web based application is a MQTT publisher of the topic “smartfarm”. The gateway, the subscriber of “smartfarm” receives the farmer’s decision and sends it to the actuator by LoRa. The broker for this diagram is “Eclipse Mosquitto, an open source (EPL/EDL licensed) message broker that implements the MQTT protocol versions 5.0.”

논문 작성시 추가할 것들:

1. 평가 (MQTT vs TCP/IP)