

MQTT와 Node-RED를 기반한 MongoDB로 저장 하는 스마트 팜 시스템 구현

박흥진

Based on MQTT and Node-RED Implementation of a Smart Farm System that stores MongoDB

Hong-Jin Park

요약 농촌의 인구 감소, 고령화에 따른 농촌 인력 부족, 기후 변화에 따른 병해충 증가로 인해 어려움을 겪고 있는 농업에 생산성을 높이고, 농산물의 품질을 향상시키는 기술 중 하나가 IoT를 활용한 스마트 팜 기술이다. 기존의 IoT를 활용한 스마트 팜은 단순히 농장을 모니터링하거나, 스마트 식물 재배기 구현, 온실 자동 개폐 시스템 등이 있다. 본 논문은 사물 인터넷의 산업 표준 프로토콜인 MQTT와 사물 인터넷의 대표적인 개발 미들웨어인 Node-RED를 기반한 스마트 팜 시스템을 구현한다. 먼저, 아두이노 센서들로 데이터를 추출하고, MQTT 프로토콜을 사용하여 IoT 디바이스에서 데이터를 수집하고 전송한다. 그런 다음, Node-RED를 사용하여 MQTT 메시지를 처리하고 데이터를 저장하기 위해 대표적인 NoSQL인 MongoDB에 실시간으로 센싱 데이터를 저장한다. 농장 관리자는 본 스마트 팜 시스템을 통해 컴퓨터나 모바일 폰을 이용하여 시간과 공간에 제약없이 언제 어디서든 실시간으로 스마트 팜에 센싱 정보를 확인할 수 있다.

Abstract Smart farm technology using IoT is one of the technologies that can increase productivity and improve the quality of agricultural products in agriculture, which is facing difficulties due to the decline in rural population, lack of rural manpower due to aging, and increase in diseases and pests due to climate change. Smart farms using existing IoT simply monitor farms, implement smart plant growers, and have automatic greenhouse opening and closing systems. This paper implements a smart farm system based on MQTT, an industry standard protocol for the Internet of Things, and Node-RED, a representative development middleware for the Internet of Things. First, data is extracted from Arduino sensors, and data is collected and transmitted from IoT devices using the MQTT protocol. Then, Node-RED is used to process MQTT messages and store the sensing data in real time in MongoDB, a representative NoSQL, to store the data. Through this smart farm system, farm managers can use a computer or mobile phone to check sensing information on the smart farm in real time, anytime, anywhere, without restrictions on time and space.

Key Words : Arduino UNO WiFi Rev.2, Node-RED, MongoDB, MQTT, Smart Farm System

1. 서론

스마트 팜은 정보통신 기술을 접목하여 온도,

습도, 이산화탄소 등의 작물의 생육 환경을 최적의 상태로 관리하는 농업 방식이다. 정밀 농업이

This work was supported by Korea Institute of Planning and Evaluation for Technology in Food, Agriculture and Forestry(IPET) and Korea Smart Farm R&D Foundation(KosFarm) through Smart Farm Innovation Technology Development Program, funded by Ministry of Agriculture, Food and Rural Affairs(MAFRA) and Ministry of Science and ICT(MSIT), Rural Development Administration(RDA)(421038-03)

* Dept. of Computer Science and Engineering, Sangji University (hjpark1@sangji.ac.kr)

Received September 21, 2023

Revised October 07, 2023

Accepted October 17, 2023

라고도 하는 스마트 팜은 농촌 인구 감소와 고령화에 따른 농촌 인력 부족, 기후 변화에 따른 작부 체계의 변화와 병해충에 증가 등으로 인해 어려움을 겪고 있는 농업의 생산성을 높이고, 농산물의 품질의 향상시키는 등 농업 전반에 경쟁력 제고에 기여할 수 있는 기술이다[1-4]. 농림축산식품부에 따르면 2021년 전 세계 스마트 팜 시장 규모는 148억 달러이고 앞으로 지속적으로 증가하여 2025년에는 220억 달러로 연평균 9.8% 시장이 확대될 것으로 예측하고 있다[5,6]. 각 나라의 스마트 팜 기술 혁신과 그에 맞는 정책이 뒷받침되어 경쟁력이 향상되고 있는 만큼 스마트 팜 시장의 성장세도 가파르게 증가 될 예정이다. 2050년까지 전 세계 인구가 97억 명에 달할 것으로 예상됨에 따라 증가하는 식량 수요를 충족하기 위해 지속 가능하고 효율적인 스마트 농업이 절실히 필요하다. 이를 위한 국내외 농업 분야에 빅데이터, 인공지능, 로봇, 사물인터넷 등의 적용에 관한 연구가 활발히 진행되고 있다.

IoT(Internet of Things) 기술을 접목한 스마트 팜 관련 기존 연구로는 단순히 농장 환경을 모니터링하거나, 아두이노와 센서를 이용하여 스마트 IoT 온실 개발이나, 가정에서 손쉽게 재배가 가능한 스마트 식물 재배기 구현 관련 연구, 자동으로 워터 펌프를 통해 물을 공급하는 시스템 관련 연구가 진행되고 있다. 또한, 단순히 온실 자동 개폐 시스템을 설계하거나, 스마트 팜 시스템을 설계하는 논문이 있다.

본 논문은 사물 인터넷이나 M2M(Machine-to-Machine)에서 사용되는 산업 표준 프로토콜인 MQTT(Message Queueing Telemetry Transport)와 사물 인터넷의 대표적인 개발 미들웨어인 Node-RED를 기반으로 스마트 팜 시스템을 구현한다. 본 논문에 IoT 기반의 스마트 팜 구축하기 위해 아두이노 우노 와이파이 Rev. 2 기반하였다. 본 논문에서는 여러 개의 스마트 팜 관련 센서를 연결하여 IoT 시스템을 구성하기 위해 MQTT가 사용한다. MQTT에 연결된 여러 센서 디바이스를 모니터링하고 작동

을 통제하기 위해서 Node-RED를 사용한다. Node-RED는 각 센서 디바이스가 보내는 메시지를 처리하고 그 결과를 대시보드에 표시할 수 있다. 또한 각 센서 디바이스에서 생성된 센싱 데이터를 Node-RED를 통해 MongoDB에 저장하였다. MongoDB는 NoSQL 데이터베이스로 정형화되지 않는 센싱 데이터를 저장하는 데 적합하다.

본 논문의 구성은 다음과 같다. 2장에서는 IoT를 활용하여 스마트 팜 관련 기존 연구를 살펴본다. 3장에서는 본 논문을 구현하기 위해 기반 연구를 살펴본다. 4장에서는 아두이노 우노 와이파이 Rev.2 기반으로 MQTT와 Node-RED, MongoDB를 이용하여 스마트 팜 시스템을 설계하고 5장에서 시스템을 구현한다. 마지막 6장에서 논문의 결론을 맺는다.

2. 기존 관련 연구

IoT를 활용하여 스마트 팜을 위해 관련된 연구는 단순히 농장 환경 정보를 모니터링한다. ESP32 NodeMCU 보드에 온도 센서, 습도 센서, 토양 센서 등을 연결하여 모바일 폰으로 농장 환경을 모니터링한다[7,8].

아두이노를 활용하여 가정에서 관상용 식물이나 채소를 손쉽게 재배가 가능한 스마트 식물 재배기를 개발하였다[9]. ON, OFF, AUTO 버튼을 통해 사용자는 워터 펌프를 제어할 수도 있다. 아두이노를 기반으로 온도, 습도, 토양 습도에 따른 물주기 관리하는 스마트 미니 온실이라는 식물 재배 환경 제어 시스템을 개발하거나[10], PCB를 제작하여 온도 센서, 조도 센서, 습도 센서 등을 설치하여 스마트 팜 재배를 위한 복합환경 제어 시스템을 개발하거나[11], 아두이노와 라즈베리파이를 이용하여 온도 센서, 습도 센서, 토양 센서를 통해 스마트 팜을 모니터링하고 모바일 어플리케이션을 통해 실시간으로 제어하는 시스템을 개발하였다[12].

MQTT를 이용한 스마트 팜 연구는 DC 24V로 동작하는 모터와 함께 WiFi 네트워크를 사용

하여 MQTT 프로토콜을 이용한 스마트 팜 자동 온실 개폐 시스템을 설계하거나[13], MQTT와 ESP 모듈을 활용하여 스마트 팜 네트워크 시스템을 설계하였다[14].

MQTT와 Node-RED를 연계한 연구 개발한 시스템은 설비 시스템의 상태를 측정하기 위해 NodeMCU(Micro Controller Unit)을 기반으로 MQTT와 Node-RED를 이용한 설비 모니터링 시스템을 구현하였다. NodeMCU는 와이파이 기능이 내장되어 있으며, 이를 이용하여 센싱된 데이터를 MQTT 프로토콜을 이용하여 무선으로 로컬 서버로 전송하여 로컬 서버에 Node-RED를 이용하여 센서 노드에서 수집된 정보를 실시간적으로 모니터링 할 수 있다[15]. 또 다른 논문에서는 제진형 배전반의 진동 및 환경 데이터 수집을 위한 사물인터넷 시스템을 구현하였다. Kepler ESP 하드웨어 플랫폼에 MQTT와 Node-RED를 이용하여 진동 데이터를 측정하고 실시간으로 모니터링 하고 하였다[16].

기존 IoT 활용한 스마트 팜 연구는 단순히 농장 환경을 모니터링하거나, 가정용이나 관상용으로 식물을 재배 가능한 식물 재배기 개발에 중점이 되어 연구거나, MQTT 만을 이용하여 스마트 팜이 연구 개발되고 있어서 MQTT 프로토콜과 MQTT를 제어하고 모니터링 하기위한 Node-RED를 연계 한 스마트 팜 시스템을 연구 개발되지 않았다. MQTT 프로토콜은 사물 인터넷이나 M2M에 사용되는 산업 분야의 표준 프로토콜이다. 또한, Node-RED는 IBM사에서 개발한 사물인터넷 분야의 대표적인 개발용 미들웨어 도구이다. 본 논문은 아두이노 우노 와이파이 Rev.2을 기반으로 MQTT와 Node-RED를 이용하여 스마트 팜을 개발하며, 센싱 데이터를 저장하기 위해 NoSQL 데이터베이스 중 MongoDB를 사용하여 센싱 데이터를 실시간적으로 저장하여 처리한다.

3. 기반 연구

3.1 아두이노 우노 와이파이 Rev.2

아두이노 우노 와이파이 Rev.2는 기능적으로 아두이노 우노 R3와 동일하지만 와이파이 및 기타 향상된 기능이 추가된 모듈이다[17,18]. 아두이노 우노 와이파이 Rev.2는 와이파이 네트워크에 대한 액세스를 제공하거나 액세스 포인트 역할을 할 수 있는 통합 TCP/IP 프로토콜 스택이 포함된 SoC(System on Chip)이다. 개발 언어로는 C언어를 사용하며 아두이노 우노 통합 개발 환경을 사용한다. 또한 다양한 실드 모듈에 추가하여 사용할 수 있다. 아두이노 우노 와이파이 Rev.2는 [그림 1]에서 보는 것 처럼 ATmega4809 마이크로컨트롤러가 사용되며 48KB의 플래시 메모리가 있고 입력 전압은 6-20V이다. 802.11b/g/n을 지원하는 와이파이 모듈이 내장되어 있다. 14개의 디지털 핀과 6개의 아날로그 핀으로 구성되어 있어 다양한 센서와 액추에이터를 연결할 수 있다. 또한 아두이노 우노와 동일한 핀 맵을 사용하므로, 기존의 아두이노 우노 보드와 호환이 가능하다.

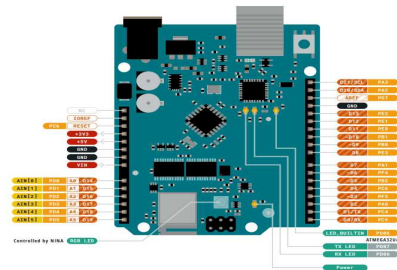


그림 1. 아두이노 우노 와이파이 Rev. 2 제품의 핀 배치
 Fig. 1. Pinout of arduino UNO WiFi Rev. 2 product

3.2 MQTT

MQTT는 사물인터넷과 같이 제한된 대역폭을 갖는 통신환경에 적합하도록 개발된 푸시 기술

(Push Technology) 기반의 메시지 전송 프로토콜이다. MQTT는 TCP/IP 프로토콜 위에서 작동하지만 메모리 용량이 적은 기기에서도 구현이 가능하고 통신 부하도 적어서 널리 사용되는 사물 인터넷 통신 프로토콜이다. MQTT는 사물 인터넷 산업 관련 국제 표준을 담당하고 있는 OASIS(Organization for the Advancement of Structured Information Standards)에서 관리하고 있다[19,20].

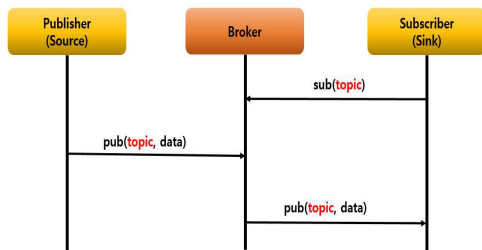


그림 2. MQTT 동작 구조
Fig. 2. Example of MQTT operation

MQTT는 발행(Publish)/구독(Subscribe) 패턴을 사용한다. 발행자는 메시지를 발행하고, 해당 메시지를 구독하고 있는 수신자는 해당 메시지를 수신한다. MQTT 브로커(Broker)를 중심으로 통신을 하는데 이는 발행자와 수신자 사이에서 메시지를 라우팅하고 필요한 경우에 메시지를 저장하며, MQTT 클라이언트는 MQTT 브로커에 연결되어 메시지를 발행하고 수신하는 구조이다. MQTT의 모든 메시지들은 토픽(Topic)이라는 형태의 메시지 정보와 실제 데이터 값(Payload)으로 구성이 된다. 토픽은 '/'기호를 사용하여 계층구조로 형성할 수 있다. [그림 2]로 예로 들면 MQTT의 발행 노드에서 생성된 메시지가 MQTT 브로커를 거쳐 구독 노드에 전달된다. 구독 노드는 주기적인 체크 방식을 이용하여 브로커에 있는 토픽을 조회하여 전달 받는다.

MQTT는 높은 확장성으로 수천 개의 디바이스와 통신할 수 있고, 또한 TCP/IP 연결을 통해 데이터를 전송하므로 대역폭 효율성이 높고, 메

시지 기반 프로토콜이므로, 애플리케이션이나 디바이스 사이의 데이터 흐름을 간편하게 관리할 수 있다. 그리고 높은 가용성을 보장하며, 장애 발생 시 재연결이 가능하다. 그리고 SSL/TLS 기반으로 보안 연결을 지원하므로, 데이터의 안전성과 보안성을 보장받을 수 있다.

3.3 Node-RED

Node-RED는 IBM사에서 개발한 사물인터넷 분야의 대표적인 개발용 미들웨어 도구이다 [21,22]. [그림 3]은 Node-RED의 프로그램 예제를 나타낸다. Node-RED는 흐름(flow) 기반으로 작동하며, 간단한 드래그-앤-드롭(Drag-and-drop)을 통해 다양한 기능을 구현할 수 있다. Node-RED는 다양한 노드들이 이미 구현되어 있어 사용자가 이들 노드를 조합하면 간단한 사물인터넷 애플리케이션을 빠르게 만들 수 있다. 예를 들어, HTTP 요청 노드와 JSON 분석 노드, 데이터베이스 연동 노드 등을 조합하여 RESTful API를 구현할 수 있다. 또한, Node-RED는 많은 플랫폼과 디바이스와 연동할 수 있다. 예를 들어, MQTT 브로커와 연동하여 MQTT 메시지를 처리하거나, AWS(Amazon Web Services), Azure, Google Cloud Platform과 같은 클라우드 서비스와도 연동할 수 있다. Node-RED는 웹 브라우저에서 작동하며, 다양한 플랫폼에서 실행할 수 있다. 또한, Node-RED의 노드는 Node.js로 구현되어 있어, 사용자가 직접 노드를 만들거나 기존 노드를 수정하여 필요한 기능을 추가할 수 있다. 또한 여러 라이브러리가 있어 팔레트(Pallet)에서 설치를 하여 노드를 추가 적으로 사용할 수 있다. 또한, Node-RED에는 편리한 GUI 기능으로 대시보드(Dashboard) 기능이 있으며 여기에는 다양한 GUI용 도구들을 제공하고 있어 사용자는 이를 이용하여 기존에 설계된 다양한 제어용 데이터 흐름을 보기 쉽게 실시간으로 볼 수 있다.

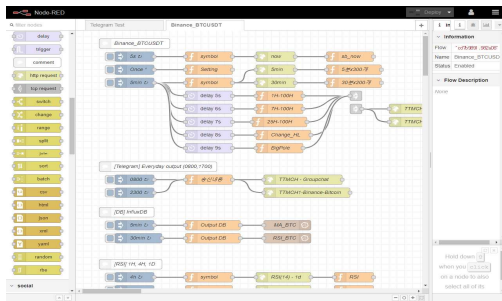


그림 3. Node-RED 샘플 프로그래밍 화면
Fig. 3. Sample programming screen of Node-RED

3.4 MongoDB

10gen이라는 회사에서 개발한 MongoDB는 NoSQL 데이터베이스 중 가장 많이 사용되는 하나로, 분산형 데이터베이스 관리 시스템(RDBMS)이다[23,24]. 관계형 데이터베이스와 달리 스키마가 고정되어 있지 않고, 필요한 데이터를 동적으로 추가하거나 수정할 수 있어 유연성이 높다. 또한, 분산형 구조로 되어 있어 대규모 데이터 처리에도 쉽다. MongoDB는 JSON(JavaScript Object Notation) 형태의 도큐먼트(Document)를 사용하여 데이터를 저장하며, 이를 콜렉션(Collection)으로 묶어서 관리한다. 각각의 도큐먼트는 고유한 Object ID를 가지며, 필드(Field)를 가진다. 이러한 구조는 JSON과 매우 유사하므로 개발자들이 쉽게 데이터를 이해하고 다룰 수 있다. MongoDB는 쿼리 기능을 지원하여 데이터를 검색하고 분석할 수 있다. 다양한 프로그래밍언어에서 사용할 수 있는 API를 제공한다. MongoDB는 대용량 데이터 처리, 실시간 데이터 분석, 클라우드 기반 애플리케이션 등 다양한 분야에서 활용된다. 이 MongoDB는 MongoDB compass를 사용해 설치한다.

4. 스마트 팜 시스템 설계

4.1 전체 시스템의 구성

[그림 4]는 전체 시스템의 구성도이다.

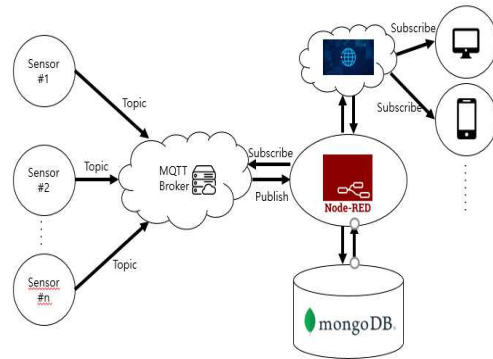


그림 4. 시스템 구성도
Fig. 4. System configuration diagram

[그림 4]에서 보는 것처럼 스마트 팜에 다양한 센서를 MQTT 브로커 서버에 연결한 후 통신한다. 아두이노 센서를 “Publisher”하여 MQTT 브로커에 데이터를 넘긴다. 그 후 Node-RED에서 MQTT 브로커를 구독하여 MQTT에서 Node-RED에게 발행한다. 그리고 Node-RED에 아두이노 센서 데이터를 보낸다. 보낸 데이터는 Node-RED에서 “localhost:8080” 주소로 PC와 스마트폰으로 원격을 통해 시각화하여 볼 수 있다. 그리고 센서 데이터를 Node-RED의 노드로 MongoDB에 연결하여 센싱 데이터로 저장한다.

4.2 센서 노드의 구성

센서 노드의 역할은 연구실 내의 개별 설비에 장착이 되어 지정된 측정 주기마다 다양한 센서들의 값을 읽고 처리하여 별도로 지정된 전송 주기마다 WiFi 기반의 MQTT 서버를 통해 스마트 팜 연구실 PC로 전송하는 역할을 한다. 센서들이 실시간으로 MQTT 클라이언트 서버로 전송되어 메시지를 보내며 Node-RED의 센서 노드에는 발행(publisher)과 구독(subscriber) 기능이 설치된다. [그림 5]의 우측은 습도(outTopic_h), 온도(outTopic_t), 이산화탄소(outTopic_co2), 조도(outTopic_c)로 설정해 두어 토픽을 가져와 설정한 노드이다. 좌측은 센서 데이터들을 토픽으로 발행하여 출력하면서

JSON 형태로 변환하여 MongoDB에 연결하고 데이터를 저장하는 방식의 노드이다.

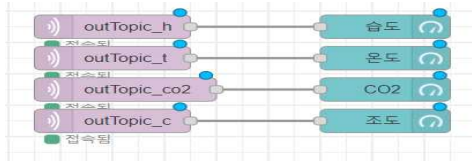


그림 5. Node-RED 센서 노드
Fig. 5. Node-RED sensor node

여기서 사용한 센서들은 CM_1107 이산화탄소, DHT 온습도, LCD, RGB_LED, 조도 센서 등을 사용하였다. 이를 사용하여 센서 측정 주기마다 정상 범위에서 벗어나면 RGB_LED 색으로 나타내어 나쁨, 보통, 좋음으로 색을 구분하게 해줬다. 이 부분은 실시간으로 볼 수 있게 하였다.

4.3 MQTT 메시지

[그림 6]은 센서 노드와 MQTT 서버 사이에 사용되고 있는 MQTT 메시지(Topic)이다. 해당 센서 데이터를 MQTT 브로커 클라이언트를 이용해서 메시지를 보내는 형태이다. 아두이노 코드에서 습도, 온도, 이산화탄소, 조도 센서 등을 MQTT 메시지 형태로 보내는 것이다. 이 데이터 메시지들을 Node-RED에서 MQTT 메시지 노드를 사용하여 연결하고 사용하는 형태이다. [그림 6]에서는 센서 데이터를 변수에 넣고 문자열로 형태를 바꿔준다. 그 후 JSON 형태로 MQTT 브로커 서버로 넘겨준다. 즉, 온도 데이터를 가져와 실수형 변수인 "t" 변수에 넣고 문자형 형태로 바꾸어 JSON 형태로 토픽을 주어 MQTT 브로커 클라이언트로 보낸다.

```
float t = dht.readTemperature(); //온도
float h = dht.readHumidity(); //습도
uint8_t ret = cm1106_i2c.measure_result();
float cds = analogRead(A0);

String tStr = String(t);
String hStr = String(h);
String co2Str = String(cm1106_i2c.co2);
String cStr = String(cds);

client.publish("outTopic_t", tStr.c_str());
client.publish("outTopic_h", hStr.c_str());
client.publish("outTopic_co2", co2Str.c_str());
client.publish("outTopic_c", cStr.c_str());
delay(10000);
```

그림 6. MQTT 메시지 정의(topic)
Fig. 6. Definition of MQTT message(topic)

4.4 MongoDB 설계

MongoDB[4]는 센서 데이터를 저장, 삽입하기 위해 구현을 하였다. 센서 데이터의 일정 주기를 시간 텀을 주고 삽입하는 방식이다. MongoDB에 연결할 때 접속이 성공하면 MQTT 브로커에서 "Connected successfully to server" 메시지를 보내준다. 또한 MongoDB에 데이터 삽입이 성공하면 "Inserted a document into the documents." 메시지를 보내준다.

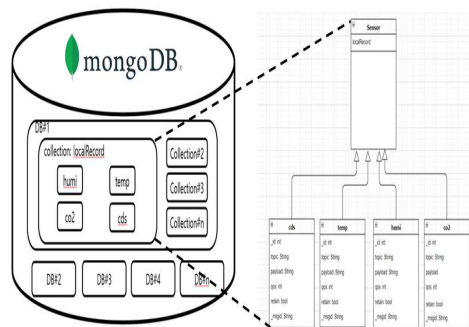


그림 7. MongoDB 시스템 구성도
Fig. 7. MongoDB System Diagram

[그림 7]은 MongoDB의 구조도로 Database 이름은 "Sensor"로 지정하였고, collection 이름

을 “localRecord”로 구현하였다. “Sensor” 데이터베이스 localRecord 안에 Co2, humi, temp, cds의 실시간 데이터를 삽입하여 저장한다. 데이터를 실시간으로 저장을 한 후 데이터의 추이를 분석할 수 있다.

5. 구현

[그림 8]은 본 논문의 모든 기능이 구현된 시스템의 사진이다. 아두이노 센서 CM_1107, 온습도, LCD, RGB_RED, 조도, Arduino UNO WiFi Rev.2 모듈 등의 모습으로 구현되어 있다.

우측 아래에는 Node-RED로 MQTT 서버 토픽 노드를 이용하여 습도, 온도, 이산화탄소, 조도를 시각화하였고, 이 센서들의 데이터를 JSON 형태로 변화함으로써 MongoDB에 삽입하는 노드를 보여주고 있다. 그리하여 노트북과 핸드폰 유무선 기기로 서버와 시각화를 모두 구현하였다.

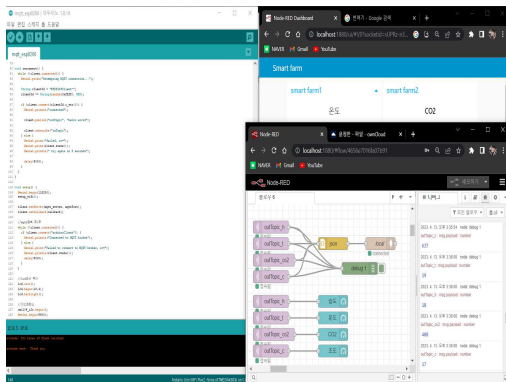


그림 8. 데이터 임의의 변화에 따른 결과 검증
Fig. 8. Verification of results following random changes in data

[그림 9]는 실시간 데이터를 가지고 와서 MongoDB에 삽입해 저장된 모습이다.

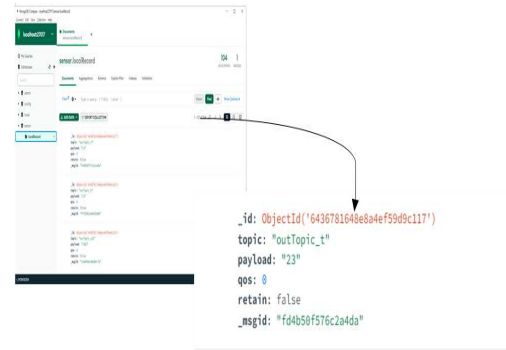


그림 9. MongoDB에 저장된 실시간 데이터
Fig. 9. Real-time data stored in MongoDB

토픽으로 온도(outTopic_t), 습도(outTopic_h), 이산화탄소(outTopic_co2), 조도(outTopic_c) 실시간 데이터를 MQTT 브로커 클라이언트 서버에서 받아와 MongoDB에서 출력한다. topic은 아두이노에서 MQTT 서버를 구독해서 MQTT 서버에서 발행하여 Node-RED로 데이터를 보내 MongoDB로 삽입하는 것이다. payload는 실제 데이터로 센서에서 데이터를 추출하여 topic으로 내보내 MongoDB로 삽입되는 것이다. id는 유일함을 보존하기 위함으로 유일성 즉 primary key이다. 스마트 팜 연구실에서 Node-RED 대시보드를 통해 특정 센서 노드의 습도를 조정할 경우, 이 내용이 MQTT 서버를 통해 해당 센서 노드로 전송되고 이에 따라 센서 노드에서의 습도 측정치가 바뀌는 걸 볼 수 있다.

[그림 10]은 본 논문에서 구현된 스마트 팜 정보를 웹 페이지 스마트 폰으로 실시간으로 나타내고 있다. 가장 먼저 스마트 팜에 있는 온도 센서와 습도 센서, 조도 센서, Co2 센서에서 측정된 센싱 값이 웹 서버로 전송된다. 웹 서버에서는 센서로부터 전송받은 센싱 데이터를 MySQL 데이터베이스에 저장한다. 데이터베이스에 저장된 데이터를 PHP로 통해 웹 브라우저로 전송된다. 사용자는 컴퓨터나 스마트 폰을 이용하여 시간과 공간에 관계없이 언제든지 실시간적으로 스마트 팜에 센싱 정보를 확인할 수 있다. 스마트 팜에서 사용된 센서가 정상적인 값은 웹이나 스

마트 폰으로 실시간적으로 확인할 수 있다.

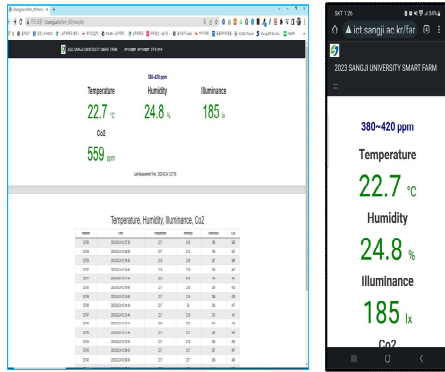


그림 10. 다양한 Node-RED 대시보드 화면과 프로그래밍 화면

Fig. 10. Various Node-RED dashboard & programming screen

6. 결론

6차 산업이라고 불리는 ICT 기반의 지능형 스마트 팜 기술은 식량 안보에 관련되어 있는 매우 중요한 기술 중 하나이다. 농산물의 생산량 증가는 물론, 노동시간 감소를 통한 농업 환경을 개선 시키는 기술이다.

본 논문은 기존 스마트 팜 개발 논문과 다르게 사물 인터넷에 널리 사용되고 있는 MQTT 프로토콜과 Node-RED를 이용하여 스마트 팜을 설계 및 구현하였다. 온도 센서, 습도 센서, 조도 센서, Co2 센서 등을 이용하여 MQTT 브로커 서버로 토픽과 페이로드를 전송하였다. 농장 관리자는 컴퓨터나 스마트 폰을 이용하여 시간과 공간에 제약 없이 언제 어디서나 실시간적으로 스마트 팜의 정보를 확인할 수 있다.

추후로 데이터에서 수집된 데이터를 빅데이터 기술이나 딥러닝 기술을 적용하여 식물 재배 등에 적용할 수 있게 해도 좋은 연구일 것이다. 본 연구에서 더 보강하여 진정한 지능형 스마트 팜 모니터링 제어 시스템으로의 발전을 위한 연구가 필요하다.

REFERENCES

- [1] Godwin Idoje, Tasos Dagiuklas, Muddesar Iqbal, "Survey for smart farming technologies: Challenges and issues", Computers & Electrical Engineering, pp. 72-78, Vol. 92, June 2021
- [2] Athanasios T. Balafoutis, "Smart Farming Technologies - Description, Taxonomy and Economic Impact", Precision Agriculture: Technology and Economic Perspectives, pp 21-77, 16 Nov., 2017
- [3] Faisal Karim Shaikh; Sarang Karim; Sherali Zeadally; Jamel Nebhen, "Recent Trends in Internet-of-Things-Enabled Sensor Technologies for Smart Agriculture", IEEE Internet of Things Journal, Vol. 9, pp. 25583-24598, 01 Dec. 2022
- [4] Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Rajiv Suman, "Enhancing smart farming through the applications of Agriculture 4.0 technologies", International Journal of Intelligent Networks, Vol. 3, pp. 150-164, Sep. 2022
- [5] <https://www.mafra.go.kr/home/5281/subview.do>
- [6] https://now.k2base.re.kr/portal/issue/ovsealssued/view.do?poliIssueId=ISUE_000000000000993&menuNo=200046&pageIndex=
- [7] Jash Doshi, Tirthkumar Patel, Santosh kumar Bharti, "Smart Farming using IoT, a solution for optimally monitoring farming conditions", Procedia Computer Science, Vol. 160, pp. 746-751, 2019
- [8] Jan Bauer, Nils Aschenbruck, "Design and implementation of an agricultural monitoring system for smart farming", IoT Vertical and Topical Summit on Agriculture, 08-09, May, 2018
- [9] Se-hoon Lee, Ha-Rin Lee, Han-Bi Kim, "Development of Smart IoT Greenhouse for Home Customized Plant Growing Environment", Korea Society of Computer Information, Vol. 30, No 2, pp. 487-488, Jul. 2022
- [10] Sung Jin Kim, Hyun Sik Lee, Eun Seo Kim, Jong Chan Park, "Smart Plant Growers Using IOT Technology", Korea Society of

Computer Information, Vol. 31, No. 1, pp. 141-144, Jan. 2023

[11] Jeong-Min Rho, Jae-Yeon Kang, Kyeong-Yeon Kim, Yu-Jin Park, Ki-Sok Kong, "IoT-based Smart Greenhouse System", Journal of The Korea Society of Computer and Information, Vol. 25, No. 11, pp. 1-8, Nov. 2020

[12] Chang-Hun Lee and Seung-Gook Hwang, "Design of Environmental Control System for Smart Farm Cultivation", The Journal of Korean Institute of Intelligent Systems, Vol. 32, No. 2, pp. 110-116, Apr. 2022

[13] Kyu-Ho Kim, Hwa-Ju Cho, Gang-Su Kim, Young-Hyung Kim, "Design of The Green House Automatic Switchgear System for Smart Farm", Korea Institute of Information Information, pp. 98-100, 2021

[14] Lim Sang Min, Yoe Hyun, An Yu Ra, Jeong Joo Won, Lee Meong Hun,E, "MQTT, ESP-NOW communication based smart farm network system design", The Korean Instuitute of Communications and Information Sciences, pp. 93-95, 2023

[15] Oh Sechun, Tae-Hyung Kim, Kim, Young Gon, "Implementation of factory monitoring system using MQTT and Node-RED", The Journal of Internet, Broadcasting and Communication, Vol. 18, No. 4, pp. 211-218, 2018

[16] Byeong-Yeong Lee, Young-Dong Lee, "Implementation of IoT System for Wireless Acquisition of Vibration and Environmental Data in Distributing Board", The Journal of Convergence Signal Processing, Vol. 22, No. 4, pp. 199-205, 2021

[17] <https://store.arduino.cc/products/arduino-uno-wifi-rev2>

[18] <http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega4808-4809-Data-Sheet-DS40002173A.pdf>

[19] <https://www.joinc.co.kr/w/man/12/MQTT/Tutorial>

[20] <https://aws.amazon.com/ko/what-is/mqtt/>

[21] <https://nodered.org/>

[22] <https://nodered.org/docs/>

[23] <https://www.mongodb.com/try/download/shell>

[24] <https://www.mongodb.com/press/10gen-announces-company-name-change-mongodb-inc>

저자약력

박 홍 진 (Hong-Jin Park)

[정회원]



- 1993년 2월 : 원광대학교 컴퓨터 공학과(공학사)
- 1995년 8월 : 중앙대학교 컴퓨터 공학과(공학석사)
- 2001년 8월 : 중앙대학교 컴퓨터 공학과(공학박사)
- 2001년 8월 ~ 현재 : 상지대학교 컴퓨터공학과 교수

〈관심분야〉 스마트 팜, 분산 시스템, 자연어처리, 딥러닝