Implementation of Self-Hosted IoT Ecosystem on NPK Soil Monitoring System

Aditya Bakti Nugroho[1], Faisal Candrasyah Hasibuan[1], Doan Perdana[2]
[1]Department of Computer Engineering, School of Electrical Engineering, Telkom University, Indonesia
[2]Department of Telecommunication Engineering, School of Electrical Engineering, Telkom University, Indonesia

Article Info

Article history:

Received September 17, 2025
Revised September 28, 2025
Accepted October 26, 2025

Keywords:

IoT
Self-hosted
MQTT
Node-RED
Personal Computer

ABSTRACT

In previous research, a tool has been made to detect soil NPK nutrient content with a screen display on the device. However, the reading results of the system cannot be monitored remotely. This research aims to develop a self-built IoT ecosystem to monitor the readings remotely. On the device side, a microcontroller can connect to the internet via a Wi-Fi network. The communication protocol selected is the MQTT protocol based on the pub-sub model. The software chosen to present the data is Node-RED. The service is self-hosted using a personal computer (PC). To be accessed from the internet, a tunneling service is used. The data presentation service obtained can be accessed remotely. Based on the test results, the MQTT protocol allows sending data only in the size of tens of bytes with an average delivery time of under one second. The data is presented in a dashboard that can be accessed via the internet with a browser.

Corresponding Author:

Aditya Bakti Nugroho
Department of Computer Engineering
School of Electrical Engineering, Telkom University
Bandung, Indonesia
Email: aditbn11@gmail.com

1. INTRODUCTION

Indonesia is an agricultural country [1]. However, successful farming requires careful consideration of several factors, including soil conditions. The quality of plants that grow in the soil is primarily determined by the nutrient content of the ground [2]. Soil fertility depends on several parameters: moisture, temperature, soil nitrogen, phosphorus, potassium (NPK) content, and soil pH value [3]. Nevertheless, soil fertility is challenging to detect directly. In prior research, a solution to the problem involved creating a soil nutrient monitoring system. This device detects the soil's nutrient content and displays it on the device's screen. In another study, soil NPK levels were identified through a mobile application based on soil images [4].

Soil monitoring systems play an important role in agricultural production. IoT-based soil condition monitoring is used in the process of maximizing crop yields by observing soil parameters and providing information that can be used by farmers remotely [5]. From the information obtained, farmers can conduct further analysis of the soil they manage [6]. After that, farmers can make decisions for their land based on previous information and analysis. In another study, it was mentioned that the excess of chemical fertilizers has become a concern in the Asian region [7].

This research aims to establish an IoT ecosystem enabling remote monitoring of soil NPK values. Several requirements must be met to meet this objective. The needs that must be addressed include the communication requirements of the system, data flow management by the server, and presentation of data. Two options exist for the server that manages the data stream from the embedded device and showcases it: to use a pre-existing

IoT service, such as Blynk, or to create a self-made ecosystem service. Blynk has been implemented in other IoT studies [9], [10]. With this solution, data is sent and stored in Blynk's cloud, and monitoring the data requires using Blynk's application [11]. This restricts the flexibility of the system. As an alternative, it is possible to build an open-source software ecosystem that uses low-cost components [12]. The server that handles this can rent a cloud server or use an on-premise server. In this research, an on-premise server will be used that runs on a personal computer.

The chosen solution is to transmit the data through an internet network connection to the server, with an on-premises server that runs on a personal computer which can then present the data in a dashboard that can be monitored via a web browser. For the embedded device to be able to connect to the Internet network, the microcontroller ESP32 is required, which can connect to a Wi-Fi network. MiFi is used to connect to the Internet through a 4G cellular network and broadcast it via Wi-Fi. Technical term abbreviations such as MQTT should always be explained when first used. The communication protocol chosen for machines, such as microcontrollers and servers, is the MQTT protocol based on the pub-sub model. The ngrok tunneling service allows the microcontroller to connect to the server through the internet [8].

With this proposed solution, a fully owned IoT ecosystem that affordable can be made. It is hoped that by creating this system, a fully owned IoT ecosystem will be more accessible to more users. The user can fully control and own the entire IoT ecosystem from the embedded system to the server that manages and serves the data.

2. THEORETICAL OVERVIEW

Soil nutrients are divided into two types, namely macro elements and microelements. Macro nutrients are needed in plants, consisting of nitrogen, phosphorus, potassium [13]. In addition, some micronutrients are only required for plants consisting of boron, copper, zinc, ferrous, molybdenum, manganese, chlorine, sodium, cobalt, silicon, and nickel. Each of these elements plays a role in plant growth.

There is a study on IoT networks for soil monitoring. That paper deals with the propagation of electromagnetic waves according to the electrical characteristics of the soil and the different compositions of the layers that make up the soil [14].

There is a study about soil monitoring techniques. It is using soil and imaging spectroscopy, based on visible-near-infrared and short-wave infrared (400-2500 nm) spectral reflectance, has been shown to be a proven method for quantitative prediction of key soil surface properties [15].

There is also a study that discusses about IoT framework. This paper provides a comprehensive study of the recent IoT based soil properties monitoring system. This paper tries to find out the current state of IoT (sensors, communication technology, attached hardware, cloud platforms, user interface, power system, challenges and benefits) in remote soil monitoring [5]. For accessing the system to the Internet, there are study that use tunnel technology. The paper using NGROK for smart cities to control devices remotely [8].

3. METHOD

3.1 System Realization

The system realization process involves implementing a system based on previously described theoretical foundations. The system realization process consists of each subsystem's realization, described below.

3.1.1 Realization of The Embedded System

The realization of the embedded system in the form of a device is done by attempting to turn on and test the device plugged into the ground. The device must be able to perform its main functionality, connect to the network, and send data to the server. The main functionality is to detect the nutrient state of the soil and display it on its OLED display. Furthermore, the device can connect to the internet through the Wi-Fi network provided by MiFi. When connected to the network, the device must send its data to the server successfully.

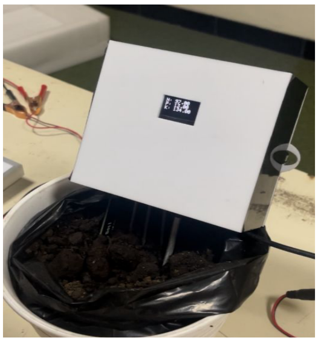


Figure 1. The device, when active, is plugged into the ground

3.1.2 System Server

The server is a personal computer (PC) with an Intel Core2Duo E8400 processor. The PC uses 4GB RAM and 80GB HDD storage. The computer is then installed with Linux Ubuntu server 22.04 OS. To access and manage the computer used as a server, another computer is needed to connect using SSH. This computer connects to a network through a LAN cable that utilizes a fiber-based home ISP for internet access.

Once the server computer has the operating system installed, the next stage is to install the required software. Firstly, Docker is installed as a platform to run container applications. In addition, Docker Compose was established to assist in the installation of containers. After installing Docker and Docker Compose, the next step is installing the necessary software to manage the data flow and presentation in the IoT monitoring system. Docker Compose configuration determines the internal container network and the settings for each container image used. The primary images used in this system are Node-RED and the Mosquitto MQTT broker.

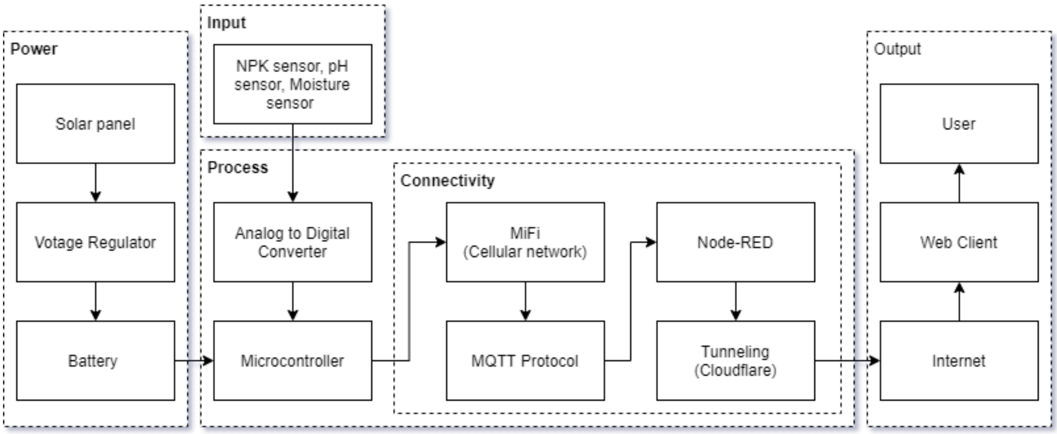
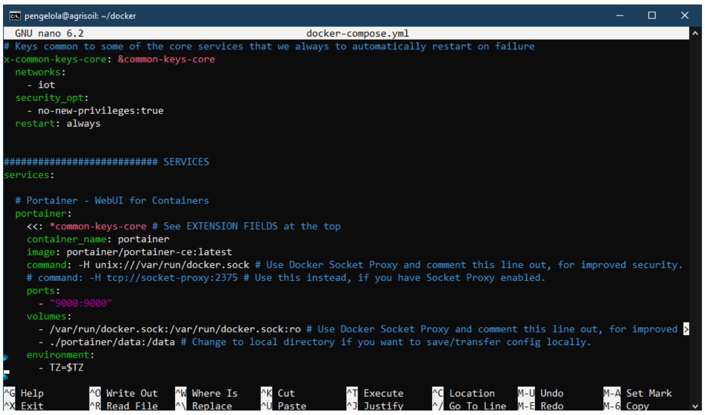


Figure 1 Overall system design plan chart

Once the desired container is successfully launched, validation is performed by accessing the server over a local network. It was observed that the service can run and can be accessed from the local network. To be accessed via the internet, Cloudflare's Tunneling service is used. The Tunneling service allows the system to be accessed from the internet without exposing the public IP address of the server. The tunneling service is also paired with DNS so users can visit it easily. For MQTT connection, because it is not supported by Cloudflare tunneling service, Ngrok is used as an alternative tunneling service for MQTT protocol connection via the internet.



```
pengelola@agrisoil: ~/docker
  GNU nano 6.2                                  docker-compose.yml
# Keys common to some of the core services that we always to automatically restart on failure
x-common-keys-core: &common-keys-core
  networks:
    - iot
  security_opt:
    - no-new-privileges:true
  restart: always


########################### SERVICES
services:

  # Portainer - WebUI for Containers
  portainer:
    <<: *common-keys-core # See EXTENSION FIELDS at the top
    container_name: portainer
    image: portainer/portainer-ce:latest
    command: -H unix:///var/run/docker.sock # Use Docker Socket Proxy and comment this line out, for improved security.
    # command: -H tcp://socket-proxy:2375 # Use this instead, if you have Socket Proxy enabled.
    ports:
      - "9000:9000"
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock:ro # Use Docker Socket Proxy and comment this line out, for improved >
      - ./portainer/data:/data # Change to local directory if you want to save/transfer config locally.
    environment:
      - TZ=$TZ

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

Figure 2 part of the configuration for the installed service

4. RESULTS AND DISCUSSION

4.1 Experimental Results

4.1.1 Data Integrity

Device testing for intact and reliable data transmission to the server. The data used for this test can be calibrated data or uncalibrated data. This test's primary focus and attention is the integrity and reliability of communication between machines. The test is carried out by programming the microcontroller to send numerical data to the server, and the tester monitors the data entering the server. Table 1 describes the data integrity test results. The test results showed that all data was sent intact.

Table 1 Data integrity validation

Data sent by the device					**Data received by the server**				
N	**P**	**K**	**pH**	**Moist**	**N**	**P**	**K**	**pH**	**Moist**
100	31	110	6	60	100	31	110	6	60
98	25	113	6	58	98	25	113	6	58
90	20	102	6	52	90	20	102	6	52
82	17	131	6	55	82	17	131	6	55
112	17	121	6	55	112	17	121	6	55

4.1.2 System Response Time

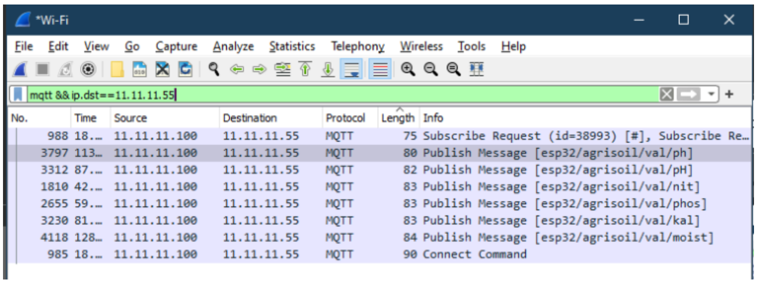
System response determines the time it takes to send data between machines. Using a method like the previous test, the microcontroller is programmed to send numerical data to the server, and the tester monitors the data entering the server by calculating the time it takes. Time counting is done using a stopwatch. Table 2 describes the test results. The results were obtained for the data transmission length, with average results below one second.

Table 2 Data delivery duration table

Test No.	Data Transmission Duration (in Second)
1	0,9
2	0,8
3	0,9
4	0,7
5	1,1

4.1.3 Size of Transmitted Data Packets

The size of data packets determines the bandwidth used in sending data. Data is sent using the MQTT protocol. This test is carried out using the Wireshark software tool. The packet size was 75 to 84 bytes (including connection request packets). Figure 8 shows the size of the data packets.

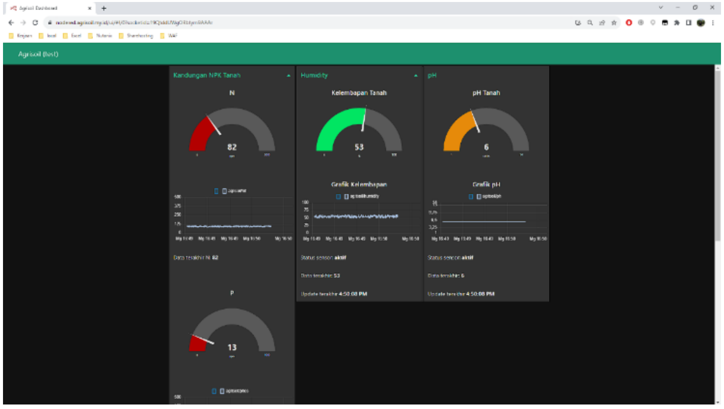

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
mqtt && ip.dst==11.11.11.55
No.     Time  Source        Destination   Protocol  Length Info
    988 18.…  11.11.11.100  11.11.11.55   MQTT          75 Subscribe Request (id=38993) [#], Subscribe Re…
   3797 113…  11.11.11.100  11.11.11.55   MQTT          80 Publish Message [esp32/agrisoil/val/ph]
   3312 87.…  11.11.11.100  11.11.11.55   MQTT          82 Publish Message [esp32/agrisoil/val/pH]
   1810 42.…  11.11.11.100  11.11.11.55   MQTT          83 Publish Message [esp32/agrisoil/val/nit]
   2655 59.…  11.11.11.100  11.11.11.55   MQTT          83 Publish Message [esp32/agrisoil/val/phos]
   3230 81.…  11.11.11.100  11.11.11.55   MQTT          83 Publish Message [esp32/agrisoil/val/kal]
   4118 128…  11.11.11.100  11.11.11.55   MQTT          84 Publish Message [esp32/agrisoil/val/moist]
    985 18.…  11.11.11.100  11.11.11.55   MQTT          90 Connect Command
```

Figure 3 MQTT data packet capture results that are received by the server

4.1.4 Application Interface

Data sent to the server must be accessible to users to use the information. Testing is done by visiting a web address with a browser on a device connected to the internet. The results obtained data can be displayed with gauge charts and line graphs. The following picture is the result of testing the display of data presentation on the web.



Data terakhir N: 82

Status sensor: aktif

Data terakhir: 53

Update terakhir 4:50:08 PM

Status sensor: aktif

Data terakhir: 6

Update terakhir 4:50:08 PM

Figure 4 Web display showing data sent by the devices.

Data transmission and display testing have also been carried out in the field. Data is sent to the server via the internet and monitored via the internet by visiting the web address in the device's browser. Two kinds of machines are used: laptops and smartphones with Android OS. The test results show that the system can display data sent via the Internet. The following is documentation of the testing process.

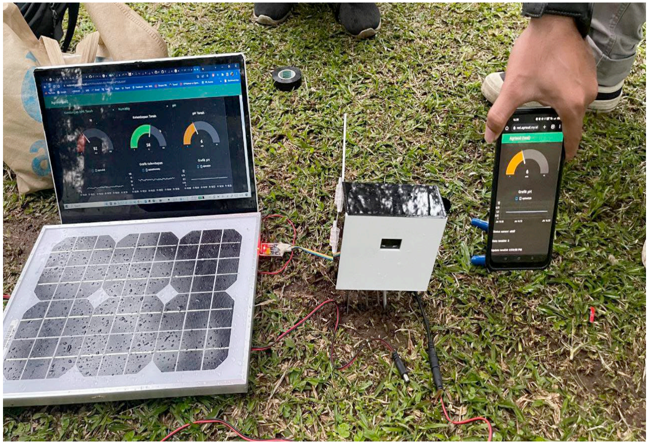


Figure 5 Testing the web presentation directly in the field.

4.2 Discussion

Based on the test results, this system has both strengths and weaknesses. One of its major advantages is that it can be implemented on inexpensive and readily available hardware. At the same time, it is scalable through service containerization. Additionally, this system can serve as a practical solution in basic IoT ecosystems, particularly for non-mission-critical systems. This system can be easily achieved by a broad range

of users using regular hardware and a home internet connection. It is reliable due to data integrity testing and efficient thanks to the MQTT protocol. The system's flexibility allows for a variety of data transmission media and frequencies. This is due to the small size of the data packets sent over the Internet.
The limitations of this system are as follows: It is implemented on a home internet connection, which may not be always available. In addition, its use of consumer-grade hardware may have an impact on its reliability. Another weakness of the system is that it needs to be set up and maintained by an experienced user.

5. CONCLUSION

From the test results, this system can operate to read the state of soil NPK and send the data so that it can be monitored remotely. The system can send data to the server via the internet using the MQTT protocol. The amount of data sent in one transmission ranges from 75-84 bytes per value. It was also found that the average data transmission length was just under one second. The server can run MQTT Broker and NodeRED services to manage and present the data flow to the user. These services are made possible using Docker. Data monitoring can be accessed from the internet using tunneling services.

REFERENCES

[1] F. R. Moeis, T. Dartanto, J. P. Moeis, and M. Ikhsan, “A longitudinal study of agriculture households in Indonesia: The effect of land and labor mobility on welfare and poverty dynamics,” *World Dev Perspect*, vol. 20, p. 100261, Dec. 2020, doi: 10.1016/J.WDP.2020.100261.

[2] J. Aliparo *et al.*, “IoT-based Assessment and Monitoring of NPK Content and Fertility Condition of Soil,” *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, vol. 2022-November, 2022, doi: 10.1109/TENCON55691.2022.9978040.

[3] M. Phute, J. Madake, and S. Bhatlawande, “Precision Farming Based Soil Fertility Assessment Techniques,” *ACCESS 2021 - Proceedings of 2021 2nd International Conference on Advances in Computing, Communication, Embedded and Secure Systems*, pp. 170–172, 2021, doi: 10.1109/ACCESS51619.2021.9563284.

[4] R. Sumiharto and R. Hardiyanto, “NPK Soil nutrient measurement prototype based on local binary pattern and back-propagation,” *Proceedings - 2018 IEEE International Conference on Internet of Things and Intelligence System, IOTAIS 2018*, pp. 23–28, Jan. 2019, doi: 10.1109/IOTAIS.2018.8600858.

[5] C. Bepery, M. S. S. Sozol, M. M. Rahman, M. M. Alam, and M. N. Rahman, “Framework for Internet of Things in Remote Soil Monitoring,” *ICCIT 2020 - 23rd International Conference on Computer and Information Technology, Proceedings*, Dec. 2020, doi: 10.1109/ICCIT51783.2020.9392658.

[6] P. Saha, V. Kumar, S. Kathuria, A. Gehlot, V. Pachouri, and A. S. Duggal, “Precision Agriculture Using Internet of Things and Wireless Sensor Networks,” pp. 519–522, Jun. 2023, doi: 10.1109/ICDT57929.2023.10150678.

[7] K. Moe, A. Z. Htwe, T. T. P. Thu, Y. Kajihara, and T. Yamakawa, “Effects on NPK Status, Growth, Dry Matter and Yield of Rice (Oryza sativa) by Organic Fertilizers Applied in Field Condition,” *Agriculture 2019, Vol. 9, Page 109*, vol. 9, no. 5, p. 109, May 2019, doi: 10.3390/AGRICULTURE9050109.

[8] K. Praghash, V. D. S. Eswar, J. Y. Roy, A. Alagarsamy, and S. Arunmetha, “Tunnel Based Intra Network Controller Using NGROK Framework for Smart Cities,” *Proceedings of the 5th International Conference on Electronics, Communication and Aerospace Technology, ICECA 2021*, pp. 39–43, 2021, doi: 10.1109/ICECA52323.2021.9676036.

[9] M. Sheth and P. Rupani, “Smart Gardening Automation using IoT with BLYNK App,” *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, vol. 2019-April, pp. 266–270, Apr. 2019, doi: 10.1109/ICOEI.2019.8862591.

[10] P. Serikul, N. Nakpong, and N. Nakjuatong, “Smart Farm Monitoring via the Blynk IoT Platform : Case Study: Humidity Monitoring and Data Recording,” *International Conference on ICT and Knowledge Engineering*, vol. 2018-November, pp. 70–75, Jan. 2019, doi: 10.1109/ICTKE.2018.8612441.

[11] M. Artiyasa, I. Himawan Kusumah, A. Suryana, Edwinanto, A. D. W. Muhammad Sidik, and A. Pradiftha Junfithrana, “Comparative Study of Internet of Things (IoT) Platform for Smart Home Lighting Control Using NodeMCU with Thingspeak and Blynk Web Applications,” *FIDELITY : Jurnal Teknik Elektro*, vol. 2, no. 1, pp. 1–6, 2020, doi: 10.52005/fidelity.v2i1.103.

[12] S. Matuska, P. Brida, and I. Bridova, “Implementation of a Self-hosted Internet of Things Solution on Personal Computer,” *Proceedings of the 2023 27th International Conference Electronics, ELECTRONICS 2023*, 2023, doi: 10.1109/IEEECONF58372.2023.10177497.

[13] R. Uchida and J. A. Silva, “Plant Nutrient Management in Hawaii’s Soils,” in *Plant Nutrient Management in Hawaii’s Soils, Approaches for Tropical and Subtropical Agriculture*, 2000, pp. 31–55.

[14] A. Slalmi, H. Chaibi, R. Saadane, A. Chehri, G. Jeon, and H. Kharraz Aroussi, “Energy-efficient and self-organizing Internet of Things networks for soil monitoring in smart farming,” *Computers & Electrical Engineering*, vol. 92, p. 107142, Jun. 2021, doi: 10.1016/J.COMPELECENG.2021.107142.

[15] S. Chabrillat, E. Ben-Dor, J. Cierniewski, C. Gomez, T. Schmid, and B. van Wesemael, “Imaging Spectroscopy for Soil Mapping and Monitoring,” *Surv Geophys*, vol. 40, no. 3, pp. 361–399, May 2019, doi: 10.1007/S10712-019-09524-0/TABLES/2.