

Rancang Bangun Sistem Monitoring Kesuburan Tanah Berbasis Internet of Things Menggunakan ESP32 dan Sensor RS485 pada Budidaya Tanaman Cabai

Ryan Firmansyah^{1*}, Rudi Susanto², Herliyani Hasanah³, Wiji Lestari⁴, Triyono⁵

¹Teknik Informatika/Fakultas Ilmu
Komputer

Universitas Duta Bangsa)

^{1*}220103188@mhs.udb.ac.id

²Teknik Informatika/Fakultas Ilmu
Komputer

Universitas Duta Bangsa

² rudi_susanto@udb.ac.id

³Teknik Informatika/Fakultas Ilmu
Komputer

Universitas Duta Bangsa

³ herliyani_hasanah@udb.ac.id

⁴Teknik Informatika/Fakultas Ilmu
Komputer

Universitas Duta Bangsa)

⁴ wiji_lestari@udb.ac.id

⁵Teknik Informatika/Fakultas Ilmu
Komputer

Universitas Duta Bangsa

⁵ triyono@udb.ac.id

Abstrak— Keberhasilan budidaya tanaman cabai sangat bergantung pada kondisi tanah, sementara di lapangan pemantauannya masih banyak mengandalkan cara manual melalui pengambilan sampel secara berkala. Akibatnya, informasi mengenai kondisi lahan kerap terlambat diperoleh dan perubahan pada tanah tidak dapat terdeteksi secara langsung. Kondisi ini pada akhirnya membuat keputusan terkait pemupukan maupun irigasi menjadi kurang tepat waktu, sehingga dibutuhkan solusi berbasis Internet of Things (IoT) yang dapat memantau kondisi tanah secara otomatis dan berkesinambungan. Penelitian ini bertujuan membangun sistem monitoring kesuburan tanah berbasis IoT dengan memanfaatkan ESP32 dan sensor tanah RS485 untuk memperoleh data kondisi tanah secara real-time pada budidaya cabai. Tahapan yang diterapkan meliputi analisis kebutuhan, desain, implementasi, hingga pengujian sistem. Dalam pengembangannya, ESP32 berperan sebagai pengendali utama, sensor tanah RS485 7-in-1 sebagai perangkat pengumpul data, protokol MQTT sebagai media transmisi data, sementara Python Backend, MySQL, InfluxDB, REST API, dan dashboard web berfungsi sebagai komponen pengelolaan dan penyajian data. Evaluasi sistem dilakukan melalui tiga jenis pengujian, yaitu monitoring sensor, komunikasi MQTT, serta layanan REST API. Dari hasil pengujian, sistem terbukti mampu memantau tujuh parameter tanah secara real-time dengan seluruh komponennya terintegrasi dengan baik. Komunikasi data via MQTT berlangsung stabil dengan keberhasilan pengiriman mencapai 100%, layanan REST API mampu menyajikan data secara konsisten ke dashboard, dan dashboard berhasil menampilkan kondisi tanah secara real-time bagi pengguna. Dengan demikian, sistem yang dikembangkan terbukti dapat mendukung monitoring kondisi tanah secara real-time dan berfungsi sebagai media pemantauan pada budidaya cabai, sekaligus membuka peluang pengembangan lebih lanjut menjadi bagian dari sistem pendukung keputusan maupun penerapan pertanian presisi.

Kata kunci—Internet of Things, monitoring kesuburan tanah, ESP32, sensor RS485, budidaya cabai.

Abstract— The success of chili cultivation depends heavily on soil conditions, yet in practice, soil monitoring is still largely carried out manually through periodic sampling. This approach often causes delays in obtaining information about field conditions and makes it difficult to detect changes in the soil as they occur. Such limitations can, in turn, hinder timely decision-making regarding fertilization and irrigation, pointing to the need for an Internet of Things (IoT)-based solution capable of continuously and automatically monitoring soil conditions. Accordingly, this study focuses on developing an IoT-based soil fertility monitoring system built around an ESP32 microcontroller and an RS485 soil sensor to obtain real-time soil condition data for chili cultivation. The development process followed four stages, namely requirements analysis, system design, implementation, and testing. Within this system, the ESP32 functions as the main controller, the RS485 7-in-1 soil sensor serves as the data acquisition device, the MQTT protocol handles data transmission, while a Python-based backend, MySQL and InfluxDB databases, a REST API, and a web dashboard together manage and visualize the collected data. To assess its performance, the system was evaluated through three types of testing: sensor monitoring, MQTT communication, and REST API service. The evaluation results show that the system was able to monitor seven soil parameters in real time, with every component functioning in full integration. MQTT communication proved stable, achieving a 100% message delivery rate; the REST API consistently supplied data to the dashboard; and the dashboard successfully presented real-time soil condition information to users. These findings confirm that the developed system can effectively support real-time soil monitoring for chili cultivation, while also offering potential for further development into decision support systems and precision agriculture applications.

Keywords— Internet of Things, soil fertility monitoring, ESP32, RS485 sensor, chili pepper cultivation

I. PENDAHULUAN

Sebagai salah satu komoditas hortikultura bernilai ekonomi tinggi, cabai (*Capsicum*

annuum L.) menghadapi permintaan pasar yang terus bertumbuh, sejalan dengan peningkatan jumlah penduduk serta kebutuhan konsumsi rumah tangga dan industri di Indonesia [1].

Kondisi tanah tempat tanaman tumbuh menjadi salah satu penentu utama keberhasilan budidaya cabai. Beberapa parameter tanah seperti pH, kelembaban, suhu, electrical conductivity (EC), serta kandungan nitrogen (N), fosfor (P), dan kalium (K) berperan penting dalam menentukan kesuburan lahan dan pertumbuhan tanaman [2]. Namun, dalam praktiknya, pengukuran kondisi tanah pada budidaya cabai masih banyak dilakukan secara manual melalui pengambilan sampel yang dilakukan secara berkala, sehingga hasil pengukuran tidak dapat dipantau secara terus-menerus dan perubahan kondisi tanah sering kali tidak terdeteksi secara langsung [3], [4]. Kondisi ini menyebabkan informasi mengenai kondisi tanah menjadi terlambat diperoleh, sehingga petani mengalami kesulitan dalam mengetahui perubahan kondisi lahan secara cepat dan proses pengambilan keputusan terkait pemupukan maupun irigasi menjadi kurang tepat waktu [5], [6]. Untuk itu, dibutuhkan sistem monitoring yang dapat mengamati kondisi tanah secara langsung (real-time), sehingga informasi kondisi lahan dapat diperoleh dengan lebih cepat dan berkesinambungan.

Berbagai penelitian telah memanfaatkan teknologi Internet of Things (IoT) untuk mendukung kegiatan monitoring pada bidang pertanian, seperti monitoring kelembaban tanah, monitoring suhu, monitoring pH, monitoring kondisi lingkungan, hingga pengembangan sistem berbasis mikrokontroler ESP32 [2]. Penelitian-penelitian tersebut menunjukkan bahwa pemanfaatan sensor dan mikrokontroler dalam sistem monitoring pertanian sudah banyak diterapkan dan terbukti mampu membantu proses pemantauan kondisi lahan secara otomatis. Namun, sebagian besar sistem yang dikembangkan masih memiliki keterbatasan, di antaranya hanya memonitor satu atau dua parameter tanah saja seperti kelembaban tanah dan debit air tanpa melibatkan parameter penting lainnya seperti pH, EC, maupun kandungan hara N, P, dan K [7]. Selain itu, beberapa sistem juga belum menggunakan sensor multi-parameter secara terintegrasi, hanya menampilkan data secara lokal melalui LCD atau notifikasi aplikasi perpesanan, serta belum memiliki dashboard berbasis web yang dapat diakses secara real-time [4]. Di sisi lain, beberapa penelitian yang telah

menerapkan protokol komunikasi MQTT dan dashboard web umumnya masih berfokus pada parameter lingkungan umum (seperti suhu udara, kelembaban udara, dan intensitas cahaya) dan belum secara khusus mengintegrasikan sensor tanah multi-parameter beserta mekanisme penyimpanan data historis melalui basis data maupun REST API dalam satu arsitektur yang utuh [8]. Berdasarkan penelitian-penelitian tersebut, masih terdapat peluang untuk mengembangkan sistem monitoring yang mampu mengintegrasikan sensor tanah multi-parameter, komunikasi MQTT, penyimpanan data, REST API, dan dashboard web ke dalam satu arsitektur yang terintegrasi sehingga proses monitoring dapat dilakukan secara real-time. Dengan demikian, pengembangan sistem monitoring tanah yang terintegrasi menjadi diperlukan satu platform yang mampu menjalankan proses akuisisi, transmisi, penyimpanan, hingga visualisasi data secara real-time.

Merujuk pada permasalahan dan celah penelitian (research gap) yang telah dipaparkan, penelitian ini mengusulkan sebuah sistem monitoring kesuburan tanah berbasis IoT yang menyatukan proses akuisisi, transmisi, penyimpanan, dan visualisasi data dalam satu arsitektur terpadu. Sistem ini dirancang dengan memanfaatkan ESP32 sebagai mikrokontroler utama, sensor tanah RS485 7-in-1 sebagai perangkat akuisisi data yang mampu mendeteksi parameter pH, EC, NPK, suhu, dan kelembaban tanah secara bersamaan [9], protokol MQTT sebagai media komunikasi data antarperangkat, InfluxDB dan MySQL sebagai media penyimpanan data historis, serta Flask REST API sebagai layanan pertukaran data yang menghubungkan perangkat keras dengan antarmuka pengguna [10]. Data yang telah terkumpul kemudian ditampilkan melalui dashboard berbasis web sebagai media visualisasi utama bagi pengguna. Dengan arsitektur tersebut, sistem yang dikembangkan mampu membaca parameter tanah secara real-time, mengirimkan data melalui jaringan internet, menyimpan data secara historis, serta menampilkan data pada dashboard web yang dapat diakses kapan saja [10]. Kemampuan ini diharapkan dapat mempermudah proses monitoring kondisi tanah, menyediakan informasi kondisi lahan secara real-time dan

berkelanjutan, mendukung pengelolaan budidaya cabai secara lebih efektif, serta menjadi sumber data yang dapat dimanfaatkan sebagai dasar pengembangan sistem pendukung keputusan (decision support system) pada penelitian selanjutnya [9].

Tujuan dari penelitian ini adalah membangun sistem monitoring kesuburan tanah berbasis Internet of Things (IoT) yang dapat menjalankan proses akuisisi, transmisi, penyimpanan, dan visualisasi data kondisi tanah secara real-time. Fokus penelitian mencakup pemantauan kondisi tanah, pembacaan parameter melalui sensor, komunikasi data, penyimpanan, dan penyajian data pada dashboard berbasis web. Melalui sistem ini, diharapkan proses monitoring tanah dapat berlangsung secara real-time dan menghasilkan data yang berguna sebagai dasar pengambilan keputusan dalam budidaya tanaman cabai.

II. METODOLOGI PENELITIAN

Pendekatan Prototype digunakan dalam penelitian ini untuk membangun sistem monitoring kesuburan tanah berbasis IoT. Tahapan pengembangannya mencakup perancangan arsitektur sistem, desain perangkat keras, implementasi perangkat lunak pada ESP32 beserta backend, hingga pengujian terhadap sistem yang dibangun. Melalui metodologi ini, diharapkan dihasilkan sistem yang mampu melakukan akuisisi, transmisi, penyimpanan, serta visualisasi kondisi tanah secara real-time.

A. Analisis Kebutuhan Sistem

Analisis Kebutuhan dilakukan untuk menentukan kebutuhan perangkat keras dan perangkat lunak.

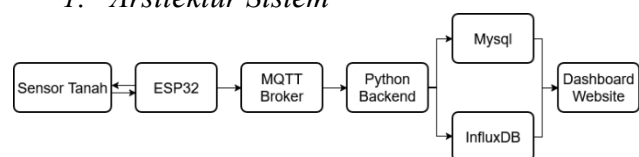
1. Kebutuhan perangkat keras terdiri dari:
 - a. ESP32 dipilih sebagai mikrokontroler karena telah dilengkapi dengan modul Wi-Fi yang memungkinkan data ke server melalui jaringan internet.
 - b. Sensor Tanah RS485 Sebagai unit utama untuk mendeteksi kondisi tanah.
 - c. Konverter TTL to RS485 sebagai jembatan komunikasi antara ESP32 dan Sensor Tanahnya

menggunakan Protokol Modbus RTU.

2. Kebutuhan perangkat keras terdiri dari:
 - a. Arduino IDE sebagai lingkungan Pengembangan Program.
 - b. Python sebagai backend yang menerima dan mengelola data hasil monitoring.
 - c. MQTT Broker Sebagai media komunikasi data
 - d. Mysql dan InfluxDB sebagai basis penyimpanan.
 - e. Flask Rest API sebagai media layanan data.
 - f. Dashboard Web sebagai media Visualisasi hasil monitoring.

B. Desain

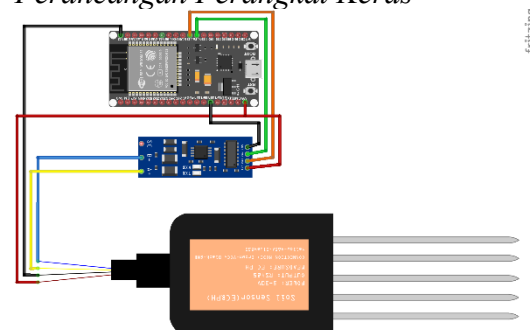
1. Arsitektur Sistem



Gambar 1. Diagram Arsitektur Sistem

Arsitektur sistem yang dikembangkan terdiri atas sensor tanah RS485, ESP32, MQTT Broker, Python Backend, MySQL, InfluxDB, dan dashboard website. Sensor tanah mengirimkan data hasil pengukuran ke ESP32 melalui komunikasi RS485. Selanjutnya, ESP32 memproses data dan mengirimkannya ke MQTT Broker menggunakan jaringan Wi-Fi. Python Backend menerima data dari MQTT Broker, kemudian menyimpannya ke MySQL dan InfluxDB. Selanjutnya, dashboard website mengambil data melalui backend untuk menampilkan informasi kondisi tanah secara real-time.

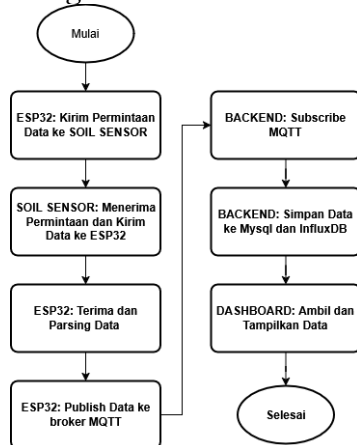
2. Perancangan Perangkat Keras



Gambar 2. Diagram Perangkat Keras

Perangkat keras pada sistem monitoring terdiri atas sensor tanah RS485 multi-parameter, ESP32, modul konverter RS485 ke TTL, dan modul MicroSD. Tujuh parameter kondisi tanah meliputi suhu, kelembaban, pH, electrical conductivity (EC), nitrogen (N), fosfor (P), dan kalium (K) diukur menggunakan sensor tanah yang terpasang pada sistem ini. Karena sensor menggunakan protokol komunikasi RS485 Modbus, diperlukan modul konverter RS485 ke TTL agar komunikasi dengan ESP32 dapat dilakukan melalui antarmuka UART. ESP32 berfungsi sebagai pengendali utama sistem yang bertugas mengirim permintaan pembacaan data kepada sensor, menerima data hasil pengukuran, melakukan parsing data, memberikan timestamp, menyimpan data ke MicroSD sebagai media pencadangan, serta mengirimkan data ke MQTT Broker melalui jaringan Wi-Fi. Seluruh komponen dihubungkan sesuai dengan rancangan perangkat keras yang ditunjukkan pada Gambar 3 sehingga proses akuisisi data dapat dilakukan secara stabil dan berkelanjutan.

3. Alur Logika Sistem



Gambar 3. Diagram Alur Logika Siste,

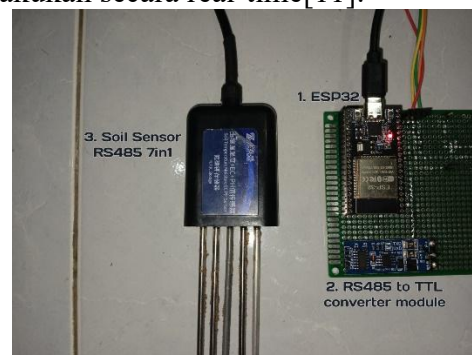
Alur kerja sistem dimulai ketika ESP32 mengirimkan permintaan pembacaan data ke sensor tanah melalui komunikasi RS485. Sensor kemudian mengirimkan hasil pengukuran ke ESP32 untuk diproses. Setelah proses parsing selesai, ESP32 mempublikasikan data ke MQTT Broker melalui jaringan Wi-Fi. Selanjutnya, Python Backend menerima data dari MQTT Broker, memprosesnya, kemudian menyimpannya ke dalam MySQL dan InfluxDB. Dashboard monitoring mengambil data melalui

backend dan menampilkan kondisi tanah secara real-time kepada pengguna.

III. HASIL DAN PEMBAHASAN

A. Implementasi Sistem

Implementasi sistem dilakukan dengan mengintegrasikan seluruh komponen perangkat keras dan perangkat lunak menjadi sebuah sistem monitoring kesuburan tanah berbasis Internet of Things (IoT). Sistem yang dibangun terdiri atas ESP32 sebagai pengendali utama, modul konverter RS485 ke TTL sebagai media komunikasi antara mikrokontroler dan sensor, sensor tanah RS485 7-in-1 sebagai perangkat akuisisi data, serta dashboard monitoring sebagai antarmuka untuk menampilkan hasil pengukuran. Integrasi antar komponen tersebut memungkinkan proses pembacaan, pengiriman, penyimpanan, dan visualisasi data kondisi tanah dilakukan secara real-time[11].



Gambar 4. Rangkaian Perangkat Keras

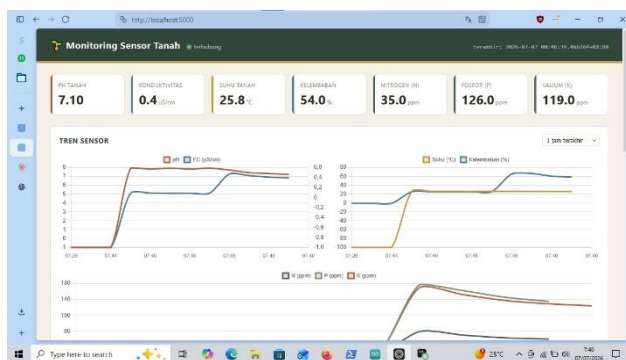
Gambar 4 memperlihatkan implementasi perangkat keras yang terdiri atas ESP32, modul konverter RS485 ke TTL, dan sensor tanah RS485 7-in-1. ESP32 berperan sebagai pengendali utama yang melakukan komunikasi dengan sensor menggunakan protokol Modbus RTU melalui antarmuka RS485[9]. Data hasil pembacaan sensor selanjutnya diproses oleh ESP32 sebelum dikirimkan ke MQTT Broker melalui jaringan Wi-Fi untuk diproses lebih lanjut oleh sistem backend.

Implementasi sistem dilakukan pada tanaman cabai dengan menempatkan sensor tanah secara langsung pada media tanam, sebagaimana ditunjukkan pada Gambar 5. Penempatan sensor pada area perakaran bertujuan agar data yang diperoleh mampu merepresentasikan kondisi tanah di sekitar akar tanaman sehingga hasil monitoring mencerminkan kondisi lingkungan tumbuh secara aktual.

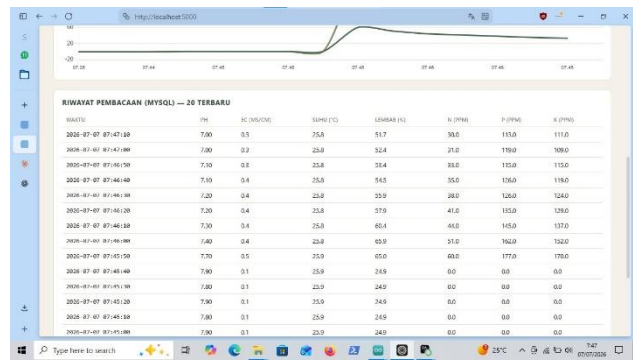


Gambar 5. Implementasi pada Tanaman Cabai

Sensor tanah RS485 7-in-1 digunakan untuk memperoleh tujuh parameter kondisi tanah, yaitu pH, kelembapan tanah, suhu tanah, Electrical Conductivity (EC), nitrogen (N), fosfor (P), dan kalium (K)[9]. Seluruh data hasil pengukuran dikirimkan ke backend menggunakan protokol MQTT untuk disimpan pada basis data sebelum ditampilkan melalui dashboard monitoring. Sebagai media visualisasi, sistem dilengkapi dengan dashboard monitoring berbasis web yang menyajikan hasil pengukuran secara real-time, seperti ditunjukkan pada Gambar 6. Dashboard menampilkan informasi kondisi tanah dalam bentuk kartu informasi, grafik tren, dan tabel riwayat pembacaan, sehingga pengguna dapat memantau nilai parameter terkini maupun perubahan data dari waktu ke waktu.



(a)



(b)

Gambar 6. (a) Dashboard Card dan Grafik (b) Dashboard Tabel History

Berdasarkan implementasi yang telah dilakukan, dashboard berhasil menampilkan seluruh parameter hasil pembacaan sensor, meliputi pH, EC, suhu tanah, kelembapan tanah, nitrogen, fosfor, dan kalium. Selain menampilkan nilai terkini, dashboard juga menyediakan grafik historis dan riwayat pembacaan sensor yang tersimpan pada basis data. Dengan demikian, sistem yang dikembangkan mampu mendukung proses monitoring kondisi tanah secara berkelanjutan melalui penyajian informasi yang mudah dipahami dan dapat diakses secara real-time.

B. Pengujian Sistem

1. Hasil Monitoring Sensor

Pengujian sistem dilakukan untuk mengevaluasi kemampuan sistem dalam membaca dan menampilkan data kondisi tanah secara real-time. Sensor tanah RS485 berhasil mengukur tujuh parameter, yaitu pH, kelembapan tanah (soil moisture), suhu tanah (soil temperature), Electrical Conductivity (EC), nitrogen (N), fosfor (P), dan kalium (K). Data hasil pengukuran kemudian dikirim oleh ESP32 melalui MQTT ke backend dan ditampilkan pada dashboard monitoring. Sebagian data hasil monitoring ditunjukkan pada Tabel 1.

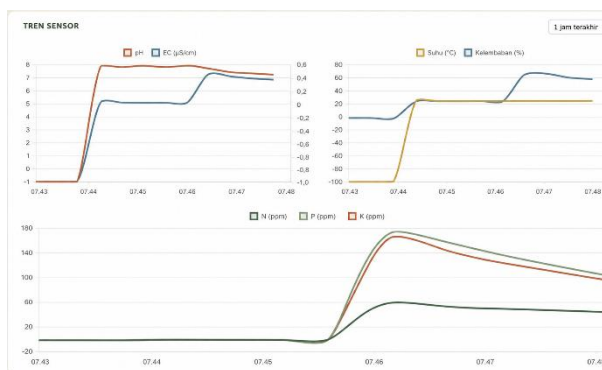
Tabel 1. Tabel Hasil Monitoring

Waktu	pH	Moisture	Temp	EC	N	P	K
2026-07-07 07:45:50	7.7	65	25.9	0.468	60	177	170
2026-07-07 07:46:00	7.4	65.9	25.8	0.435	51	162	152
2026-07-07 07:46:10	7.3	60.4	25.8	0.405	44	145	137
2026-07-07 07:46:20	7.2	57.9	25.8	0.389	41	135	129
2026-07-07 07:46:30	7.2	55.9	25.8	0.361	38	126	124
2026-07-07 07:46:40	7.1	54.5	25.8	0.362	35	126	119
2026-07-07 07:46:50	7.1	53.4	25.8	0.342	33	115	115
2026-07-07 07:47:00	7	52.4	25.8	0.332	31	119	109
2026-07-07 07:47:10	7	51.7	25.8	0.331	30	113	111
2026-07-07 07:47:20	7	51.2	25.8	0.333	27	122	102

Berdasarkan Tabel 1, sistem berhasil melakukan pembacaan terhadap seluruh parameter tanah secara kontinu selama proses pengujian. Nilai pH yang diperoleh berada pada rentang 7,0–7,7, sedangkan suhu tanah relatif stabil pada kisaran 25,8–25,9°C. Nilai kelembapan tanah (soil moisture) mengalami perubahan dari 65,0% menjadi 49,7%, yang diikuti dengan perubahan nilai Electrical Conductivity (EC) dari 0,468 mS/cm menjadi 0,309 mS/cm. Selain itu, sensor juga berhasil membaca perubahan konsentrasi nitrogen, fosfor, dan kalium pada setiap interval pengukuran. Dengan hasil tersebut, terbukti bahwa sensor memiliki kemampuan untuk mengenali perubahan kondisi tanah secara terus-menerus sepanjang proses monitoring berlangsung.

Perubahan nilai pada setiap parameter menunjukkan bahwa proses akuisisi dan transmisi data berjalan dengan baik[11]. Data hasil pembacaan sensor berhasil diterima oleh backend melalui protokol MQTT, kemudian disimpan pada basis data dan diproses untuk ditampilkan pada dashboard monitoring. Hal ini menunjukkan bahwa sistem mampu melakukan monitoring kondisi tanah secara real-time tanpa mengganggu proses pengiriman maupun penyimpanan data.

Guna mempermudah pengamatan, hasil monitoring turut divisualisasikan melalui grafik pada dashboard monitoring[12]. pada Gambar 7. Visualisasi tersebut memudahkan pengguna dalam mengamati perubahan setiap parameter tanah dari waktu ke waktu tanpa harus membaca data numerik secara langsung.



Gambar 7. Grafik Visualisasi

Berdasarkan hasil implementasi dan pengujian, sistem monitoring yang dikembangkan berhasil mengintegrasikan proses akuisisi data menggunakan sensor tanah RS485, transmisi data melalui MQTT, penyimpanan data pada MySQL dan InfluxDB, serta penyajian informasi melalui dashboard web secara real-time. Integrasi antar komponen tersebut memungkinkan proses monitoring kondisi tanah dilakukan secara berkelanjutan, sehingga sistem dapat dimanfaatkan sebagai media pemantauan pada budidaya tanaman cabai dan menjadi landasan bagi pengembangan sistem pendukung keputusan pada penelitian selanjutnya.

2. Pengujian MQTT

Pengujian komunikasi MQTT dilakukan untuk mengukur performa transmisi data antara ESP32 sebagai *publisher* dan Python Backend sebagai *subscriber*. Pengujian dilakukan dengan mengirimkan 100 pesan secara berurutan menggunakan QoS level 1 dan interval pengiriman 0,2 detik[13]. Setiap pesan diberi nomor urut dan *timestamp*, kemudian nilai *delay* dihitung berdasarkan selisih waktu antara waktu pengiriman dan waktu penerimaan pesan. Selain itu, pengujian juga digunakan untuk menghitung persentase *packet loss* selama proses transmisi.

Tabel 2. Pengujian MQTT

Parameter	Nilai
Jumlah pesan dikirim	100
Jumlah pesan diterima	100
Packet loss	0 pesan (0%)
Delay rata-rata	0,083 ms
Delay minimum	0,000 ms
Delay maksimum	2,690 ms
Durasi total pengujian	± 19,26 detik

Berdasarkan Tabel 2, seluruh 100 pesan yang dikirim berhasil diterima oleh subscriber dengan tingkat keberhasilan 100% dan packet loss sebesar 0,00%. Selain itu, diperoleh nilai delay rata-rata di bawah 1 ms, yang menunjukkan bahwa proses komunikasi menggunakan protokol MQTT pada lingkungan pengujian berjalan dengan sangat baik. Hasil tersebut dipengaruhi oleh skenario

pengujian yang dilakukan pada jaringan lokal, di mana publisher, subscriber, dan MQTT Broker berada pada host yang sama sehingga proses transmisi tidak dipengaruhi oleh hambatan jaringan nirkabel, jarak komunikasi, maupun interferensi sinyal. Sebagian besar data pengujian menunjukkan nilai delay sebesar 0,00 ms, yang bukan berarti proses transmisi berlangsung tanpa waktu tunda, melainkan dipengaruhi oleh keterbatasan resolusi pencatatan waktu (timestamp) yang digunakan dalam pengukuran. Sementara itu, beberapa pesan yang memiliki delay hingga 2,69 ms diduga disebabkan oleh variasi penjadwalan proses (thread scheduling) pada sistem operasi maupun proses internal Python selama pengujian[14].

Perlu diperhatikan bahwa hasil pengujian ini merepresentasikan kondisi terbaik (best-case scenario) pada jaringan lokal dan belum sepenuhnya menggambarkan kondisi implementasi di lapangan. Pada penggunaan sebenarnya, ESP32 berkomunikasi melalui jaringan Wi-Fi menuju MQTT Broker sehingga nilai delay dan potensi packet loss dapat dipengaruhi oleh kualitas sinyal, kondisi jaringan, serta jarak antara perangkat dan access point. Oleh karena itu, hasil pengujian ini digunakan sebagai baseline performa komunikasi MQTT pada sistem yang dikembangkan, sedangkan pengujian pada lingkungan operasional nyata dapat dilakukan pada penelitian selanjutnya untuk memperoleh evaluasi performa yang lebih representatif.

Pengujian REST API dilakukan untuk mengukur waktu respons (response time) layanan dalam melayani permintaan data dari sisi klien (dashboard). Pengujian dilakukan dengan mengirimkan 100 permintaan HTTP GET secara berturut-turut pada masing-masing dari tiga endpoint, yaitu /api/sensor/latest, /api/sensor/history, dan /api/sensor/series. Waktu respons dihitung sejak permintaan dikirim hingga seluruh isi respons diterima oleh klien. Selain itu, pengujian juga digunakan untuk mengevaluasi keberhasilan layanan dalam memproses setiap permintaan yang diberikan.

Tabel 3. Pengujian Rest API

Endpoint	Rata-rata(ms)	Min(ms)	Maks(ms)	Sukses
/api/sensor/latest	2.092,31	2.042,23	2.415,51	100/100
/api/sensor/history	2.115,77	2.073,64	2.177,89	100/100
/api/sensor/series	2.087,16	2.044,19	2.167,10	100/100
Gabungan (300 request)	2.098,41	2.042,23	2.415,51	100/100

Berdasarkan Tabel 3, seluruh 300 permintaan yang dikirimkan ke ketiga endpoint berhasil diproses dengan HTTP Status 200, sehingga tingkat keberhasilan layanan mencapai 100% tanpa ditemukan error maupun timeout. Meskipun demikian, waktu respons rata-rata pada seluruh endpoint berada pada kisaran 2,1 detik, yang menunjukkan bahwa layanan REST API telah berfungsi dengan baik dari sisi fungsional, namun masih memiliki performa yang relatif lambat untuk aplikasi monitoring real-time. Kesamaan waktu respons pada ketiga endpoint, meskipun menggunakan sumber data yang berbeda (MySQL dan InfluxDB), mengindikasikan bahwa keterlambatan lebih dipengaruhi oleh overhead sistem yang bersifat tetap dibandingkan kompleksitas proses pengambilan data. Kondisi tersebut diduga disebabkan oleh proses pembukaan koneksi baru ke basis data pada setiap permintaan (connection overhead) serta proses resolusi alamat localhost pada sistem operasi yang menambah waktu sebelum koneksi berhasil dibangun.

Perlu diperhatikan bahwa hasil pengujian ini diperoleh menggunakan Flask Development Server pada lingkungan pengembangan, sehingga belum sepenuhnya merepresentasikan performa layanan pada lingkungan produksi. Selain itu, mekanisme pembukaan koneksi baru pada setiap permintaan dan penggunaan alamat localhost berpotensi menambah waktu respons secara konsisten. Oleh karena itu, hasil pengujian ini menunjukkan bahwa layanan REST API telah berfungsi dengan baik dari sisi fungsional, namun masih memerlukan optimasi performa melalui penerapan connection pooling,

penggunaan WSGI server produksi seperti Unicorn, serta konfigurasi alamat host yang lebih optimal[15], [16]. Pengujian lanjutan setelah proses optimasi diperlukan untuk memperoleh waktu respons yang lebih sesuai dengan kebutuhan aplikasi monitoring real-time.

3. Pembahasan

Berdasarkan hasil implementasi, terlihat bahwa sistem monitoring yang dibangun berhasil menyatukan berbagai komponen — mulai dari akuisisi data lewat sensor tanah RS485 7-in-1, transmisi data melalui MQTT, penyimpanan pada MySQL dan InfluxDB, sampai penyajian informasi melalui dashboard berbasis web. Hasil pengujian pada Tabel 2 menunjukkan bahwa seluruh 100 pesan berhasil diterima tanpa packet loss dengan delay rata-rata di bawah 1 ms, sedangkan pengujian REST API pada Tabel 3 menunjukkan tingkat keberhasilan layanan mencapai 100% untuk seluruh permintaan pada ketiga endpoint. Kedua hasil tersebut memperkuat bahwa jalur komunikasi data sensor hingga penyajian pada antarmuka berjalan secara stabil, sehingga proses monitoring kondisi tanah dapat dilakukan secara otomatis dan berkesinambungan, memungkinkan pengguna memperoleh informasi kondisi tanah secara real-time.

Jika dibandingkan dengan penelitian-penelitian terdahulu, cakupan integrasi pada sistem yang dikembangkan di penelitian ini terbilang lebih menyeluruh. Sebagai contoh, penelitian Ariwibowo dkk. [1] hanya berfokus pada pemantauan kelembapan tanah dan debit air tanaman cabai menggunakan ESP32, tanpa melibatkan protokol transmisi standar seperti MQTT atau dashboard monitoring berbasis web.. Penelitian Ashar dkk. [2] memang menggunakan sensor tanah RS485 7-in-1 yang serupa dengan penelitian ini, namun sistem dirancang untuk beroperasi secara offline tanpa koneksi internet dan hanya menampilkan data melalui Human Machine Interface (HMI) lokal, sehingga belum mendukung transmisi data jarak jauh maupun penyimpanan pada basis data cloud. Pada sisi arsitektur REST API,

penelitian Bramantyo dkk. [3] mengembangkan integrasi perangkat IoT, basis data PostgreSQL, dan dashboard website untuk pemantauan kualitas udara lahan pertanian, dengan menambahkan protokol WebSocket (Socket.io) agar pembaruan data dapat dikirim secara push real-time ke antarmuka tanpa perlu klien melakukan permintaan berulang. Pendekatan tersebut berbeda dari penelitian ini yang masih mengandalkan REST API berbasis polling murni, sehingga waktu respons yang diperoleh (rata-rata 2,1 detik) sepenuhnya bergantung pada frekuensi permintaan dari sisi klien. Sementara itu, penelitian Wardihani dkk. [5] telah memanfaatkan protokol MQTT untuk pemantauan dan pengendalian parameter greenhouse, tetapi arsitekturnya berfokus pada pengendalian aktuator melalui skema gateway-server dan belum melibatkan REST API maupun kombinasi basis data relasional dan time-series. Sistem yang dikembangkan pada penelitian ini melengkapi kekurangan tersebut dengan mengintegrasikan sensor RS485 7-in-1, komunikasi MQTT, backend berbasis Python, dua jenis basis data (MySQL dan InfluxDB), serta REST API dalam satu arsitektur yang utuh, sehingga memberikan cakupan parameter tanah yang lebih lengkap dan fleksibilitas pengelolaan data yang lebih baik dibandingkan penelitian-penelitian tersebut.

Penggunaan MQTT sebagai protokol komunikasi memberikan proses pertukaran data yang ringan dan efisien, terbukti dari hasil pengujian yang menunjukkan delay rata-rata hanya 0,083 ms tanpa adanya packet loss. Pemisahan fungsi penyimpanan menggunakan MySQL untuk data relasional dan InfluxDB untuk data time-series memungkinkan pengelolaan data yang lebih optimal sesuai karakteristik masing-masing jenis data. Selain itu, arsitektur modular yang memisahkan lapisan akuisisi data, backend, basis data, dan dashboard memudahkan pengembangan sistem apabila di masa mendatang diperlukan penambahan sensor baru maupun integrasi dengan sistem pendukung keputusan.

Penelitian ini masih memiliki beberapa keterbatasan. Pengujian komunikasi MQTT

dilakukan pada jaringan lokal, di mana publisher, subscriber, dan MQTT Broker berada pada host yang sama, sehingga hasil yang diperoleh merepresentasikan kondisi terbaik (best-case scenario) dan belum sepenuhnya menggambarkan kondisi komunikasi pada jaringan Wi-Fi di lingkungan budidaya yang sebenarnya. Selain itu, pengujian REST API masih menggunakan Flask Development Server dengan waktu respons rata-rata sekitar 2,1 detik, yang belum menggambarkan performa optimal ketika diimplementasikan pada server produksi, mengingat keterlambatan tersebut diduga lebih dipengaruhi oleh overhead pembukaan koneksi baru pada setiap permintaan dibandingkan kompleksitas proses pengambilan data itu sendiri. Mekanisme pembaruan data pada dashboard yang masih bergantung pada permintaan berulang (polling) dari sisi klien juga berpotensi menambah beban pada server dibandingkan mekanisme push berbasis WebSocket.

IV. KESIMPULAN

Tujuan penelitian ini tercapai dengan berhasilnya pengembangan sistem monitoring kesuburan tanah berbasis IoT yang memanfaatkan mikrokontroler ESP32 dan sensor tanah RS485 untuk mendukung budidaya tanaman cabai. Berbagai komponen sistem — akuisisi data melalui sensor, komunikasi data via protokol MQTT, penyimpanan data pada backend berbasis MySQL dan InfluxDB, hingga visualisasi lewat dashboard berbasis web — berhasil disatukan dalam satu arsitektur yang utuh. Perpaduan perangkat keras dan lunak ini memungkinkan proses monitoring kondisi tanah berjalan secara real-time serta mudah diakses pengguna. Secara umum, sistem yang dikembangkan mampu melakukan pembacaan tujuh parameter kondisi tanah secara bersamaan, mengirimkan data dengan baik melalui jaringan komunikasi yang digunakan, menyediakan layanan backend yang berjalan sebagaimana mestinya, serta menyajikan informasi kondisi tanah kepada pengguna

secara berkelanjutan melalui dashboard yang informatif. Penelitian ini memberikan kontribusi berupa sistem monitoring kondisi tanah yang terintegrasi dan modular, sehingga dapat digunakan sebagai media pemantauan lahan budidaya cabai secara berkelanjutan serta menjadi dasar pengembangan pengelolaan pertanian yang lebih efisien dan berbasis data. Untuk pengembangan selanjutnya, penelitian ini dapat dilanjutkan dengan melakukan optimasi performa layanan backend, pengujian sistem pada lingkungan budidaya yang lebih luas dan kondisi jaringan yang lebih beragam, serta integrasi sistem monitoring dengan sistem pendukung keputusan guna menghasilkan rekomendasi pemupukan secara otomatis bagi petani.

UCAPAN TERIMA KASIH

The author would like to thank the Directorate of Research and Community Service, Directorate General of Research and Development, Ministry of Higher Education, Science and Technology and Universitas Duta Bangsa Surakarta for funding with contract number 285/C3/DT.05.00/PL-BARU/2026, 042/LL6/AL.04.03/PL-BARU/2026, 066/UDB.LPPM/A.34-HK/II/2026..

REFERENSI

- [1] Liska, I. Novita, and S. Masithoh, "ANALISIS NILAI TUKAR PETANI CABAI (*Capsicum Annum* L.) DAN FAKTOR - FAKTOR YANG MEMPENGARUHINYA PADA MASA PANDEMI COVID-19," *J. Agribisains*, vol. 9, no. 1, pp. 61–67, Apr. 2023, doi: 10.30997/jagi.v9i1.8257.
- [2] W. Ifa Susuek Anselmus Talli, J. Dedy Irawan, and F. Xavierius Ariwibisono, "RANCANG BANGUN SISTEM MONITORING KUALITAS TANAH UNTUK TANAMAN CABAI BERBASIS IOT (INTERNET OF THINGS)," *jati*, vol. 7, no. 4, pp. 2428–2435, Dec. 2023, doi: 10.36040/jati.v7i4.7540.
- [3] K. Khanal, G. Ojha, S. Chataut, and U. K. Ghimire, "IoT-Based Real-Time Soil Health Monitoring System for Precision Agriculture," vol. 11, no. 07, 2024.
- [4] M. Riyadi, "Sistem Cerdas Untuk Monitoring Pengukuran Suhu Dan Kelembapan Tanah Pada Tanaman Cabai Berbasis Internet Of Things (IOT) Menggunakan Aplikasi Telegram," *JTE*, vol. 14, no. 2, p. 105, Jun. 2023, doi: 10.22441/jte.2023.v14i2.008.
- [5] W. Wijayanti, K. Santa, and S. C. Kumajas, "Sistem Kontrol Dan Monitoring Kelembapan Dan pH Tanah Tanaman Cabai Rawit Berbasis Internet of Things (IoT) Menggunakan Metode Fuzzy Mamdani," vol. 14, 2025.
- [6] M. N. K. Hidayatullah, A. A. Maulana, A. R. I. Ghifari, A. Ambari, and U. N. Jadid, "Pengembangan Aplikasi Mobile Berbasis IoT untuk Monitoring Kualitas Tanah dan Kebutuhan Nutrisi Tanaman Cabe," 2025.

- [7] M. R. Ariwibowo, L. A. Setiawan, and A. Iman, "SISTEM PEMANTAU KELEMBAPAN TANAH DAN DEBIT AIR PADA TANAMAN CABAI MENGGUNAKAN ESP32 BERBASIS INTERNET OF THINGS," *JITET*, vol. 13, no. 2, Apr. 2025, doi: 10.23960/jitet.v13i2.6382.
- [8] Eni Dwi Wardihani et al., "Pemantauan dan Pengendalian Parameter Greenhouse Berbasis IoT Dengan Protokol MQTT," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 13, no. 1, pp. 38–43, Feb. 2024, doi: 10.22146/jnteti.v13i1.8564.
- [9] I. A. A. Ashar et al., "INTEGRASI SENSOR KESUBURAN TANAH BERBASIS MODBUS RS485 PADA SISTEM MONITORING PERTANIAN PRESISI BERBASIS DATA," vol. 4, no. 4, 2025.
- [10] S. B. Bramantyo, I. N. Dewi, I. M. Reza, F. O. Saputra, and Z. A. Hasibuan, "PENGEMBANGAN ARSITEKTUR REST API UNTUK INTEGRASI DATA REAL-TIME PADA WEBSITE PEMANTAUAN KUALITAS UDARA LAHAN PERTANIAN," *Transmisi J.Ilm. Tek. Elekt.*, vol. 27, no. 1, pp. 49–56, Jan. 2025, doi: 10.14710/transmisi.27.1.49-56.
- [11] D. Berlian and L. O. Sari, "Smart Plant Monitoring Sistem Suhu dan Kelembaban Tanah Pada Tanaman Cabai Berbasis IoT Menggunakan Protokol MQTT".s
- [12] K. S. Lestari, R. A. P. Parhando, and M. W. Septiano, "Perancangan Antarmuka IOT Admin Dashboard Untuk Sistem Pemantauan Lingkungan Cerdas Menggunakan Pendekatan User Centered Design," vol. 2, no. 1, 2026.
- [13] N. Indrasari, A. S. Rohman, T. Genarsih, A. Dwinanda, and S. L. Putri, "ANALISIS PERFORMA PROTOKOL MQTT PADA SISTEM MONITORING GAS BERBAHAYA MENGGUNAKAN SENSOR MQ135 TERINTEGRASI ANDROID," no. 1.
- [14] T. D. Putra and R. Purnomo, "Analisis Algoritma Round Robin pada Penjadwalan CPU: Algoritma Scheduling CPU," *JITIKA*, vol. 15, no. 2, pp. 85–90, Sep. 2021, doi: 10.32815/jitika.v15i2.481.
- [15] I. K. D. A. S. Nanda, K. A. Seputra, and K. T. Dermawan, "Perbandingan Performa RESTful API Menggunakan Framework Python Flask dan Go Gin pada Pengembangan Aplikasi Alumni Circle," *RIGGS*, vol. 5, no. 1, pp. 12113–12119, Apr. 2026, doi: 10.31004/riggs.v5i1.7664.
- [16] T. Purwanto, "Analisa Perbandingan Kinerja Rest Api Dengan Framework Flask, Laravel, Dan Express Js," vol. 3, no. 4, 2023.