

Evaluasi Sistem Pemantauan Cuaca Berbasis IoT Menggunakan *Automatic Weather Station* (AWS) Pada Lahan Perkebunan

Pria Nur Islam^{1*}, Afu Ichsan Pradana², Dwi Hartanti³

^{1*,2,3} Program Studi Teknik Informatika, Universitas Duta Bangsa Surakarta

*Email: 22010384@mhs.udb.ac.id

Abstract

Extreme climate change requires accurate micro-climate monitoring infrastructure in the open agricultural sector to reduce risks in agribusiness operations. This research aims to implement an Internet of Things (IoT)-based Automatic Weather Station (AWS) prototype in Pakebun Outdoor and evaluate the Quality of Service (QoS) performance of outdoor telemetry networks using two protocols, namely MQTT and HTTP POST. The system was built by applying the Rapid Application Development (RAD) method, utilizing a dual middleware architecture on Node-RED to facilitate simultaneous data transmission to the InfluxDB time-series database. Validation was conducted through Black Box testing and continuous field QoS monitoring for 30.8 hours. The test results showed that both protocols were able to collect 5,768 data packets with a very low packet loss rate of 0.02%. However, MQTT demonstrated superiority with a lower average delay of 1,647.53 ms, while the HTTP POST transmission recorded 1,652.05 ms. This is due to the absence of repeated TCP handshake cycles in MQTT. Therefore, it can be concluded that the MQTT protocol is more reliable and efficient in terms of latency compared to HTTP POST for outdoor environmental telemetry purposes. The contribution of this research provides a new empirical basis for selecting the most effective communication protocols for the development of precision agriculture digitalization systems.

Keywords: Automatic weather station, internet of things, mqtt, quality of service, rapid application development

Abstrak

Perubahan iklim yang ekstrem memerlukan adanya infrastruktur untuk pemantauan cuaca mikro yang akurat dalam sektor pertanian terbuka, demi mengurangi risiko pada operasi agribisnis. Penelitian ini bertujuan untuk menerapkan prototipe *Automatic Weather Station* (AWS) berbasis *Internet of Things* (IoT) di Pakebun Outdoor dan melakukan evaluasi terhadap kinerja *Quality of Service* (QoS) jaringan telemetri luar ruangan menggunakan dua protokol, yaitu MQTT dan HTTP POST. Sistem dibangun dengan menerapkan metode *Rapid Application Development* (RAD), menggunakan arsitektur *middleware* ganda pada Node-RED untuk memfasilitasi transmisi data secara bersamaan ke basis data *time-series InfluxDB*. Validasi dilakukan melalui pengujian *Black Box* dan pemantauan QoS di lapangan selama 30,8 jam tanpa henti. Hasil pengujian menunjukkan bahwa kedua protokol dapat mengumpulkan 5.768 paket data dengan tingkat kehilangan paket yang sangat rendah, yaitu 0,02%. Namun, MQTT menunjukkan keunggulan dengan rata-rata delay yang lebih rendah, yaitu 1.647,53 ms, sementara transmisi HTTP POST mencatatkan 1.652,05 ms. Hal ini disebabkan oleh tidak adanya siklus *TCP handshake* yang berulang pada MQTT. Dengan demikian, dapat disimpulkan bahwa protokol MQTT lebih andal dan efisien dalam hal latensi dibandingkan HTTP POST untuk keperluan telemetri lingkungan luar ruangan. Kontribusi dari penelitian ini memberikan dasar empiris yang baru dalam memilih protokol komunikasi yang paling efektif bagi pengembangan sistem digitalisasi pertanian presisi.

Kata kunci: Stasiun cuaca otomatis, internet of things, mqtt, kualitas layanan, rapid application development

1. Pendahuluan

Sektor pertanian dan produksi makanan di dunia saat ini sedang menghadapi situasi yang sangat sulit karena perubahan iklim yang semakin cepat dan pemanasan global. Dengan perkiraan jumlah penduduk dunia yang akan meningkat, kebutuhan akan pangan diperkirakan naik hingga 60% pada tahun 2050. Perubahan cuaca ekstrim menjadi ancaman utama yang mengganggu stabilitas dalam bisnis pertanian [1]. Sifat dari komoditas hortikultura yang sangat peka terhadap perubahan suhu, tingkat sinar ultraviolet (UV), variasi intensitas cahaya, dan perubahan curah hujan menjadikannya bagian dari rantai pasok yang paling rentan terhadap kegagalan dalam produksi. Ketidakpastian cuaca di luar ruangan tidak hanya mengurangi jumlah hasil panen, tetapi juga merusak kualitas gizi dan menyebabkan kerusakan lebih awal di jalur distribusi makanan, yang pada akhirnya mengancam upaya untuk menghapus kelaparan di seluruh dunia (Nafhah et al., 2021).

Sebagai negara yang bergantung pada pertanian di daerah tropis, Indonesia mengalami kemungkinan kerugian ekonomi besar akibat perubahan iklim besar seperti El Niño Southern Oscillation (ENSO). Fenomena El Niño secara signifikan mengubah waktu tanam yang biasa, memperpanjang masa kekeringan, dan menyebabkan perubahan curah hujan lokal yang tidak terduga [1]. Penelitian terbaru menunjukkan bahwa ketidakpastian cuaca lokal dan perubahan curah hujan ekstrem di daerah pertanian di Indonesia dapat mengurangi produktivitas tanaman sayur hingga 48,9% (Nafhah et al., 2021). Saat ini, banyak petani masih bergantung pada informasi cuaca umum dari satelit atau ramalan tradisional yang kurang tepat [3]. Namun, validasi data cuaca yang diambil langsung dari alat di stasiun cuaca darat terbukti jauh lebih akurat dan memiliki sedikit bias akibat kondisi geografis dibandingkan dengan perkiraan dari satelit [4]. Oleh karena itu, pemasangan Automatic Weather Station (AWS) yang mandiri di

tingkat lahan pertanian menjadi teknologi yang sangat diperlukan untuk memastikan akurasi dalam mengelola risiko operasional dalam agribisnis [5].

Urgensi penggunaan stasiun cuaca otomatis ini terlihat dari unit bisnis urban farming Pakebun Outdoor yang dikelola oleh PT Penta Digital Nusantara di kawasan Cileunyi, Jawa Barat. Sebagai perusahaan agribisnis modern, Pakebun memerlukan informasi tentang iklim yang tepat untuk memastikan kualitas produk pangan tetap terjaga. Selama ini, pengawasan terhadap kondisi cuaca di lahan masih dilakukan secara manual yang rentan terhadap kesalahan manusia. Ketidacukupan informasi cuaca mikro di kawasan Pakebun mengakibatkan keputusan operasional sering kali terlambat diambil [6]. Perubahan digital melalui sistem telemetri IoT menjadi solusi penting bagi Pakebun untuk menciptakan sistem pemantauan yang cepat, jelas, dan otomatis [7].

Meskipun banyak studi telah meneliti desain perangkat AWS yang didukung oleh IoT, masih ada kekurangan dalam penelitian mengenai cara meningkatkan efisiensi software stack di bagian backend server maupun frontend aplikasi [8]. Penelitian tentang stasiun cuaca sebelumnya cenderung menggunakan database relasional biasa seperti PostgreSQL atau MySQL, yang sering kali menghadapi masalah kinerja dan membutuhkan waktu lama untuk menjalankan kueri saat mencoba mengelola kecepatan penulisan data telemetri yang tinggi [9]. Di samping itu, visualisasi data cuaca yang canggih sering kali bergantung pada platform luar seperti Grafana, yang membutuhkan banyak sumber daya, atau terpaksa menggunakan protokol HTTP POST standar pada gerbang sensor nirkabel [1]. Dalam pengujian telemetri nirkabel yang ada, protokol HTTP POST menunjukkan bahwa cara ini tidak efisien karena memerlukan proses jabat tangan yang berulang serta membawa beban header yang besar setiap kali mengirim data, yang mengakibatkan

pemborosan bandwidth jaringan dan penggunaan daya baterai yang tinggi [10]. Sebaliknya, pengujian perbandingan parameter latensi menunjukkan bahwa protokol MQTT dengan sistem publish/subscribe dapat memberikan respons pengiriman data yang jauh lebih cepat, serta menghemat energi perangkat di luar ruangan secara signifikan dibandingkan HTTP [11].

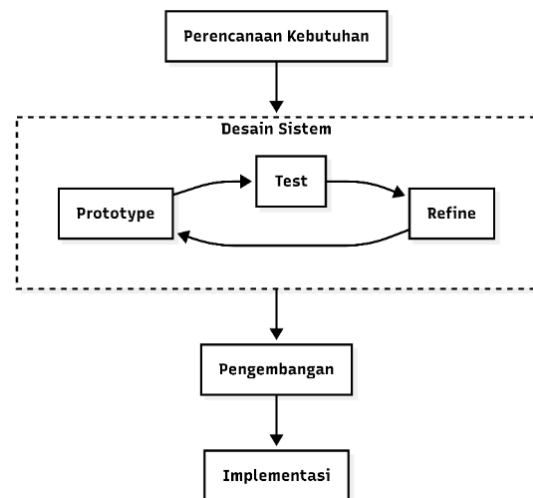
Untuk mengatasi masalah tersebut, studi ini menyajikan inovasi berupa arsitektur AWS IoT yang menghubungkan AWS WH65LP dengan pengontrol ESP32 tanpa perlu menggunakan kontainer yang berat seperti yang diusulkan oleh [12]. Kebaruan dalam desain arsitektur ini diperkuat dengan penggunaan sistem penyimpanan ganda yang menggabungkan modul Jam Real Time (RTC) dan kartu MicroSD sebagai penyimpanan cadangan lokal untuk mengurangi risiko kehilangan data. Selain itu, terdapat juga keselarasan antara Node-RED dengan InfluxDB yang berfungsi sebagai Basis Data time series di sisi server, yang telah terbukti bisa mencatat data 20,95 kali lebih cepat dibandingkan dengan RDBMS berdasarkan penilaian [9]. Dari segi metodologi, keseluruhan proses pembuatan sistem dari desain infrastruktur telemetri hingga visualisasi data menggunakan metode Rapid Application Development (RAD) untuk memastikan integrasi perangkat keras dan perangkat lunak yang cepat dan berulang [13]. Integritas data telemetri yang disimpan dijamin melalui prosedur Kontrol Kualitas (Quality Control/QC) berdasarkan validasi waktu sesuai standar [14]. Oleh sebab itu, penelitian ini bertujuan untuk mewujudkan model fisik sistem pemantauan cuaca dengan delapan parameter di kawasan pertanian Pakebun Outdoor, memastikan bahwa sistem berfungsi dengan baik melalui pengujian Black Box, serta menilai secara kuantitatif perbandingan kehandalan kinerja jaringan antara protokol MQTT dan HTTP POST berdasarkan metrik seperti keterlambatan, kehilangan paket, dan kepadatan data nyata

dari hasil uji lapangan yang dilakukan secara langsung.

2. Metode Penelitian

2.1. Desain Penelitian dan Diagram Alir

Model pengembangan sistem *Rapid Application Development* (RAD) berfungsi sebagai kerangka kerja utama dalam studi ini untuk mengatur seluruh proses penggabungan perangkat keras di tingkat lapangan serta arsitektur perangkat lunak *server* dengan cara yang bertahap. Proses pelaksanaan penelitian disusun secara teratur ke dalam empat fase yang saling terkait. Urutan pelaksanaan dari tahap-tahap ini ditampilkan secara visual pada Gambar 1.



Gambar 1. Diagram Alir Tahapan *Rapid Application Development*

Menurut alur yang ditunjukkan pada Gambar 1, proses pengembangan tidak berlangsung secara lurus, tetapi menerapkan proses yang berulang selama tahap perancangan sistem yang terdiri dari siklus pembuatan model awal (*prototype*), pengujian fungsi (*test*), dan perbaikan desain (*refine*). Siklus visual ini dipilih berdasarkan prinsip pengembangan sistem yang terintegrasi yang dibuat oleh [13] untuk mengatasi masalah integrasi perangkat keras secara cepat di lapangan.

2.2. Lokasi Penelitian dan Bahan Perangkat Utama

Implementasi fisik, pemasangan alat, dan pengumpulan data telemetri jaringan dilaksanakan secara langsung di lingkungan terbuka (*outdoor*) di kawasan kebun Pakebun *Outdoor*. Lokasi penelitian ini terletak di Kavling Mekar Murni, Bl. B No.10, Cimekar, Kecamatan Cileunyi, Kabupaten Bandung, Jawa Barat 40623.

Standarisasi penulisan metode dipenuhi dengan menghilangkan penyebutan peralatan kecil yang umum ditemukan di laboratorium, sehingga penjelasan difokuskan hanya pada perangkat utama yang digunakan di lapangan serta tumpukan perangkat lunak *backend server* yang mencakup:

- Instrumen Sensor Lapangan: Satu unit stasiun cuaca otomatis *Automatic Weather Stasion* jenis *WH65LP* yang sesuai dengan standar industri untuk mengumpulkan berbagai parameter meteorologi sekaligus.
- Perangkat Keras Pemroses dan Telemetri: Mikrokontroler *ESP32*, modul konverter serial *HW-0519* (dari *RS485* ke *TTL*), modul Jam *Real Time (RTC)* jenis *DS1307*, dan kartu *MicroSD* untuk penyimpanan tambahan lokal.
- Arsitektur *Server Cloud*: *Platform* pemrograman berbasis alur *Node-RED* yang berfungsi sebagai penghubung, *broker MQTT* pentarium. id, dan basis data yang menyimpan data dalam bentuk *time series* yaitu *InfluxDB*.

2.3. Prosedur Pelaksanaan dan Tahapan Siklus RAD

Tahapan pelaksanaan percobaan dan pembangunan sistem dilakukan dengan cara yang teratur melalui empat langkah utama dalam siklus RAD yang dijelaskan sebagai berikut:

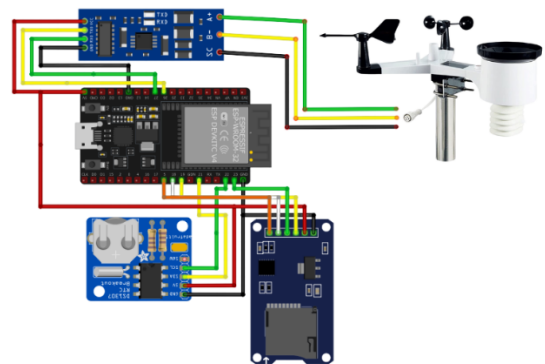
- Fase Perencanaan Kebutuhan

Pada tahap ini, dilakukan penentuan dan analisis teknis mengenai masalah yang disebabkan oleh perubahan jaringan internet tanpa kabel di tempat terbuka Pakebun *Outdoor*. Di awal tahap ini,

dilakukan penetapan spesifikasi sistem untuk secara otomatis mengumpulkan delapan parameter cuaca mikro dari sensor *AWS WH65LP*, yaitu suhu udara ($^{\circ}\text{C}$), kelembapan udara (%), arah angin ($^{\circ}$), kecepatan angin (m/s), kecepatan hembusan maksimum (m/s), curah hujan akumulatif (mm), intensitas cahaya (lux), dan radiasi ultraviolet (UV).

- Fase Desain Sistem

Di tahap ini, dilakukan pemodelan alur data dan perancangan diagram rangkaian elektronika (*wiring diagram*). Rincian tentang rencana penyambungan elemen elektronik di lapangan disampaikan secara menyeluruh pada Gambar 2. Sinyal serial dari sensor *AWS WH65LP* terhubung ke pin serial mikrokontroler *ESP32* melalui modul konverter *HW-0519* untuk mengubah sinyal menjadi *TTL*. Jaringan telemetri ini dirancang dengan efisien tanpa memerlukan proses kontainerisasi yang berat sebagaimana dikritik oleh [12]. Pada tahap ini juga, sistem penyimpanan keamanan *Dual Path* dimodelkan, di mana data cuaca yang dikumpulkan akan diperoleh dengan penanda waktu dari *RTC DS1307* sebelum disimpan di *MicroSD* lokal dan dikirim ke *server cloud* secara bersamaan. Jika dalam pengujian purwarupa terjadi masalah pada pembacaan data, proses penyempurnaan akan dilakukan segera sebelum melanjutkan ke tahap berikutnya.



Gambar 2. Skema Interkoneksi Rangkaian Elektronika Sistem *AWS IoT*

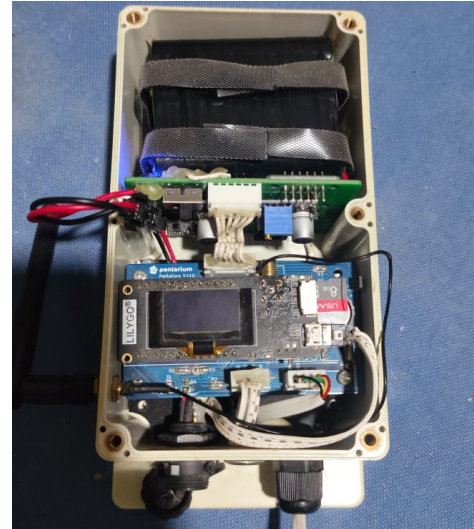
c. Fase Pengembangan

Tahap ini dimulai dengan melaksanakan integrasi secara mekanis dan kalibrasi fisik pada elemen sensor yang berada di luar ruangan. Bentuk fisik dari alat pengukur cuaca otomatis (*Automatic Weather Station*) jenis *WH65LP* ditampilkan secara lengkap sebelum proses pemasangan pada tiang infrastruktur pendukung, seperti yang terlihat di Gambar 3.



Gambar 3. Realisasi Fisik Unit Sensor Lapangan *AWS WH65LP*

Rangkaian elektronik dibuat dalam bentuk fisik dengan menggunakan kotak panel tahan air (*Waterproof*) untuk melindunginya di lapangan. Susunan sirkuit dan hubungan antar komponen elektronik yang ada di dalam kotak panel setelah perakitan dijelaskan dengan rinci pada Gambar 4. Selaras dengan pemasangan sirkuit fisik, penulisan kode program pada *chip ESP32* diselesaikan menggunakan *software Arduino IDE* untuk mengatur cara membaca data secara serial, pengaturan penyimpanan lokal, serta pengiriman data menggunakan berbagai protokol (*MQTT* dan *HTTP POST*). Di sisi *server*, *platform Node-RED* diatur untuk mengolah data yang diterima dan mengirimkannya ke basis data *time-series InfluxDB*. Basis data ini dipilih karena memiliki kecepatan penulisan yang dapat berjalan 20,95 kali lebih cepat dibandingkan dengan sistem *RDBMS SQL* biasa merujuk pada evaluasi [9].



Gambar 4. Tata Letak Sirkuit dan Komponen Elektronika pada Panel *Box*

d. Fase Implementasi

Prototipe fisik *AWS* telah dipasang sepenuhnya di lokasi operasional Pakebun *Outdoor*. Dokumentasi tentang keadaan pemasangan tiang untuk infrastruktur pendukung serta lokasi penempatan alat telemetri di area uji lapangan ditampilkan dengan jelas pada Gambar 5. Fungsionalitas mekanis dan kelistrikan diuji secara menyeluruh menggunakan metode *Black Box Testing* pada tingkatan pengujian sistem (*system testing*). Uji ini sangat penting dilakukan di lapangan untuk mendeteksi masalah pada sistem lebih awal, karena perangkat *IoT* memiliki keterikatan yang kuat (*strong coupling*) antara *hardware* dan *software* di luar ruangan seperti yang dijelaskan dalam peta metodologi pengujian [15]. Integritas aliran data telemetri yang masuk ke *server* basis data dijamin melalui prosedur Kontrol Kualitas (*Quality Control/QC*) yang berbasis validasi waktu (*timestamp*) merujuk pada standar [14].



Gambar 5. Kondisi Instalasi Operasional Perangkat AWS di Lapangan Pakebun Outdoor

2.4. Metode Pengolahan Data dan Tolok Ukur Kinerja Jaringan (QoS)

Setelah semua alat fisik berfungsi dengan baik di lapangan, dilakukan pengujian performa jaringan telemetri untuk menilai dan membandingkan seberapa dapat diandalkannya pengiriman paket data tanpa kabel antara protokol *MQTT* dan *HTTP POST*. Catatan *log* waktu pengiriman direkam dari sisi pengirim (*node* sensor *ESP32*) sampai tersimpan di sisi penerima (*server database InfluxDB*). Uji kualitas layanan (*Quality of Service/QoS*) secara menyeluruh (*end-to-end*) dalam lingkungan terbuka ini sesuai dengan kerangka pengujian sistem *IoT* menurut [15] untuk menilai pengaruh perubahan konektivitas luar ruangan terhadap metrik *delay* dan *packet loss*, serta parameter kerapatan data nyata dari pengujian di lapangan. Analisis data statistik dari pengamatan dihitung secara kuantitatif berdasarkan metrik parameter telemetri milik [10] dan [11].

Perhitungan rata-rata nilai keterlambatan jaringan ditentukan menggunakan rumus yang tertera pada Persamaan (1).

$$\text{Delay} = \frac{\sum_{i=1}^n (\text{Treceive}, i - \text{Tsend}, i)}{n} \quad (1)$$

Dalam Persamaan (1), Delay mengindikasikan rata-rata waktu tunda dalam pengiriman data, yang diukur dalam milidetik (ms). Notasi $T_{\text{receive},i}$ menggambarkan waktu

(*timestamp*) ketika paket data cuaca ke-*i* berhasil diterima dan dicatat di sisi server basis data, sedangkan $T_{\text{send},i}$ menggambarkan waktu saat paket data ke-*i* dikirim oleh mikrokontroler *ESP32* di lapangan. Variabel *n* menunjukkan total jumlah paket data cuaca yang berhasil dikirim dan disimpan di basis data selama periode pengujian.

Sementara itu, persentase tingkat kegagalan atau hilangnya data telemetri akibat fluktuasi sinyal internet di area perkebunan terbuka dihitung menggunakan rumus *packet loss* pada Persamaan (2).

$$\text{Packet Loss} = \left(\frac{P_{\text{sent}} - P_{\text{received}}}{P_{\text{sent}}} \right) \times 100\% \quad (2)$$

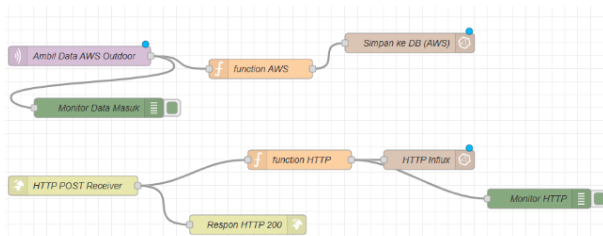
Berdasarkan Persamaan (2), *Packet Loss* menunjukkan persentase paket data yang tidak sampai atau tidak berhasil sampai ke *server* dalam bentuk persen (%). Notasi P_{sent} adalah singkatan dari jumlah seluruh paket data cuaca yang dikirim oleh perangkat mikrokontroler *ESP32* dari ladang pertanian, sementara P_{received} adalah total paket data yang sebenarnya berhasil diterima, lolos dari pengecekan, dan disimpan dengan baik di *server* basis data *time-series InfluxDB*.

3. Hasil Penelitian

3.1. Realisasi Arsitektur Perangkat Lunak Server (Middleware)

Implementasi arsitektur perangkat lunak di lingkungan server cloud dilakukan dengan memanfaatkan platform pemrograman berbasis alur visual yakni *Node-RED*. Rancangan logika aliran data dikembangkan secara terpisah melalui dua pipa penerimaan yang berfungsi secara mandiri dan bersamaan. Pendekatan arsitektur yang terpisah ini diterapkan untuk memastikan keabsahan pengujian komparatif secara menyeluruh, mulai dari pengiriman perangkat keras di tingkat lapangan, sekaligus meningkatkan efektivitas dari tumpukan perangkat lunak di *backend server* seperti yang diulas oleh [8]. Proses penggabungan kluster server dengan endpoint eksternal dirancang untuk mendukung transformasi digital melalui sistem

telemetri *IoT* yang cepat dan otomatis sesuai dengan standar yang ditetapkan oleh [7], serta untuk menjamin konsistensi dari topologi sistem integrasi perangkat secara holistik, sebagaimana dijelaskan oleh [16]. Diagram aliran logika untuk pemrosesan data cuaca mikro yang komprehensif di platform Node-RED dapat dilihat pada Gambar 6.



Gambar 6. Aliran Logika Pemrosesan Data Telemetri AWS pada Node-RED

Berdasarkan desain yang terlihat pada Gambar 6, sistem *middleware* terbagi menjadi dua kelompok utama untuk pengolahan yang tidak saling mengganggu satu sama lain:

a. Kluster Telemetri *MQTT*

Alur kerja ini dimulai dengan node input yang menggunakan *MQTT Subscribe*, yang disetel untuk secara pasif mendengarkan topik pengiriman dari lapangan lewat broker nirkabel pentarium. id. Setelah paket data diterima dengan baik, *payload* kemudian diproses oleh blok fungsi *AWS* untuk dipecah menjadi parameter cuaca yang berbeda sebelum disimpan ke dalam database melalui blok Simpan ke *DataBase (AWS)*. Proses penanganan *payload JSON* yang cepat ini menggunakan skema manajemen memori yang ringan untuk *ESP32*, dengan pengoptimalan fungsi pemrosesan *string* berdasarkan waktu.

b. Kluster Telemetri *HTTPS POST*

Secara terpisah, sebuah node input dengan tipe *HTTP In* diatur sebagai alamat *URL server* untuk menerima permintaan *POST* yang dikirimkan oleh mikrokontroler di lapangan. Untuk menghindari terjadinya masalah koneksi pada unit mikrokontroler akibat keterlambatan status di bagian hulu, *node HTTP Response* langsung disambungkan untuk memberikan umpan balik secara langsung (Respon *HTTP 200 OK*). Data yang

diterima kemudian diteruskan ke blok fungsi *HTTP* sebelum disimpan ke dalam basis data melalui blok *HTTP Influx*.

Untuk membuat proses pengawasan eksperimen lebih mudah, setiap kluster dilengkapi dengan komponen pemantau internal, yaitu Pemantauan Data Masuk pada jalur *MQTT* dan Pemantauan *HTTP* pada jalur *HTTP POST*.

3.2. Realisasi Antarmuka Dashboard Monitoring Sistem AWS

Tahap visualisasi data diwujudkan dalam bentuk antarmuka pengguna yang berbentuk *dashboard web* responsif untuk pemantauan. *Dashboard* ini berperan sebagai alat untuk merepresentasikan data meteorologi temporal yang secara dinamis diambil dari *database InfluxDB*, sehingga tidak perlu bergantung pada *platform* eksternal seperti *Grafana* yang dapat membebani sumber daya sistem [1]. Dengan penerapan antarmuka otomatis ini, fungsi pengawasan kondisi cuaca lahan yang sebelumnya dilakukan secara manual dapat digantikan, yang sering kali berisiko terjadi kesalahan manusia [6]. Desain visualisasi yang instan ini selaras dengan prinsip kemudahan dalam menerapkan *IoT* untuk meningkatkan efisiensi pemantauan harian [17]. Desain visual dan tata letak panel untuk pemantauan parameter cuaca mikro pada aplikasi *dashboard* ditampilkan dalam Gambar 7.



Gambar 7. Tampilan Antarmuka Aplikasi Dashboard Monitoring Parameter Cuaca AWS

3.3. Pengujian Kinerja Jaringan Telemetry (*Quality Service*)

Pengujian kualitas layanan jaringan atau *Quality of Service (QoS)* dilakukan secara bersamaan untuk menganalisis keandalan transmisi data antara protokol *MQTT* dan *HTTP POST*, dengan fokus pada parameter delay dan kehilangan paket. Pengumpulan *dataset* telemetry berlangsung secara terus-menerus dan dalam waktu nyata selama 30,8 jam tanpa henti, dimulai dari 19 Juni 2026 pukul 04:15:45 UTC hingga 20 Juni 2026 pukul 11:08:16 UTC.

Pemilihan periode pengamatan yang panjang ini sengaja dilakukan untuk

menangkap variasi nyata dari kualitas konektivitas jaringan nirkabel di kawasan terbuka Pakebun *Outdoor*, baik pada saat lalu lintas jaringan mencapai puncaknya maupun ketika terjadi perubahan kondisi cuaca dari siang hingga malam. Seluruh catatan *log* waktu pengiriman paket dari node lapangan ke server cloud berhasil dikumpulkan dengan tepat, dan hasil analisis data statistik dicatat dengan rinci dalam Tabel 1.

Tabel 1. Perbandingan Metrik Kinerja Kualitas Layanan Jaringan Telemetry

Protokol	Total Data Berhasil Diserap	Rata-Rata Delay (ms)	Packet Loss (%)
<i>MQTT</i>	5.678	1.647,53	0,02
<i>HTTP POST</i>	5.678	1.652,05	0,02

Sumber: Data primer yang diolah (2026)

Analisis metrik yang tertera pada Tabel 1 memberikan gambaran mengenai kinerja dua protokol telemetry yang beroperasi di atas kluster *middleware* yang terisolasi. Dari total 5.769 paket unik yang berhasil dicatat secara keseluruhan dalam server database selama periode pengujian, masing-masing protokol memperlihatkan tingkat keberhasilan yang sangat tinggi dalam menerima paket, dengan keduanya berhasil mengamankan sebanyak 5.768 paket. Penurunan satu paket pada setiap protokol bukan disebabkan oleh kegagalan sistem selama penggunaan, melainkan akibat dari pergeseran waktu tunggu pada inisiasi jabat tangan jaringan yang memakan waktu 16 detik di awal dan akhir proses pengambilan data. Perekaman data pertama untuk jalur *HTTP POST* terjadi pada pukul 04:15:45 UTC ketika koneksi *MQTT* masih dalam proses inisiasi sesi subscribe. Sementara itu, kluster *MQTT* secara konsisten menuntaskan perekaman data terakhir hingga pukul 11:08:16 UTC, pada saat sesi transmisi *HTTP POST* dinyatakan selesai oleh mikrokontroler.

Kondisi ini mengarah pada penghitungan tingkat persentase kehilangan paket yang serupa dan sangat kecil, yaitu 0,02% untuk kedua pihak yang terlibat. Meskipun demikian, perbedaan dalam kinerja yang dapat dijelaskan

secara ilmiah tetap muncul dalam parameter latensi jaringan. Berdasarkan pengamatan, protokol *MQTT* menunjukkan rata-rata waktu keterlambatan pengiriman yang lebih efektif, yakni 1.647,53 ms. Angka latensi ini mengungguli performa transmisi pada jalur *HTTP POST* yang memerlukan rata-rata waktu tunda sedikit lebih lama, yaitu 1.652,05 ms. Perbedaan tunda waktu sebesar 4,52 ms ini menegaskan bahwa struktur pengiriman data berkala milik *MQTT* mampu mempercepat proses data cuaca mikro secara nyata di Pakebun *Outdoor*. sejalan dengan penilaian efisiensi latensi dari protokol *IoT* dalam jaringan terbatas yang telah dibuktikan oleh (Sifawa et al., 2026).

3.4. Analisis Mekanisme Transmisi Jaringan

Penurunan kinerja dan waktu tunda yang meningkat pada protokol *HTTP POST* dipengaruhi oleh karakteristik arsitektur transaksional yang mendasarinya. Setiap kali unit mikrokontroler *ESP32* mengirimkan delapan parameter cuaca mikro dari lapangan, protokol *HTTP POST* harus membangun koneksi *TCP* baru melalui proses jabat tangan *TCP*, sambil menambahkan beban header yang besar untuk setiap sesi pengiriman. Proses pembentukan koneksi ini menyebabkan penggunaan *bandwidth* yang tinggi dan secara kumulatif meningkatkan latensi dalam jaringan telemetri. Detail dari batasan siklus jabat tangan dalam protokol *TCP/IP* ini selaras dengan kajian matematis yang dilakukan oleh [10].

Karakteristik jaringan di lingkungan perkebunan luar ruangan, yang sering mengalami redaman sinyal karena vegetasi yang lebat, semakin memperburuk kendala dalam pengiriman data. Hal ini dipetakan dalam teori atenuasi luar ruangan yang diungkapkan oleh [19], yang juga didukung oleh evaluasi infrastruktur nirkabel di area pedesaan terbuka oleh [20]. Saat konektivitas internet di area terbuka perkebunan tidak stabil, jeda waktu tunggu selama siklus *HTTP POST* berdampak langsung pada kehilangan

paket data yang tidak berhasil divalidasi oleh log server. Hal ini menyebabkan peningkatan angka kegagalan paket data, seperti yang dianalisis dalam log koneksi nirkabel jarak jauh oleh [21]. Fenomena kegagalan ini secara logis menjelaskan mengapa jumlah paket yang berhasil diterima oleh node *HTTP In* pada Gambar 6 tercatat lebih rendah pada awal jangka waktu eksperimen.

Sebaliknya, keunggulan performa *MQTT* berasal dari penerapan model *publish/subscribe* yang beroperasi melalui koneksi *TCP* tunggal yang persisten (*keep-alive*). Node *ESP32* di lapangan hanya perlu melakukan satu kali inisiasi jabat tangan di awal sesi operasional, setelah itu paket data cuaca mikro dikirim secara berkala dalam bentuk payload pesan yang ringan dengan ukuran header yang minimal. Pola komunikasi yang efisien ini secara signifikan mengurangi waktu pemrosesan dalam jaringan telemetri nirkabel, menghasilkan waktu tunda rata-rata yang lebih rendah dan mengurangi risiko kegagalan transmisi data di area Pakebun *Outdoor* [11].

3.5. Komparasi Temuan dengan Riset Terdahulu

Temuan mengenai tren kinerja telemetri luar ruangan ini menambah dan memperluas hasil penelitian empiris yang dilakukan oleh [9]. Mereka menekankan bahwa arsitektur data *time-series* seperti *InfluxDB* memerlukan saluran data *streaming* yang efisien, seperti *MQTT*, untuk menjaga stabilitas kapasitas tulis server. Penelitian ini menunjukkan bahwa kecepatan pemrosesan data deret waktu pada server *InfluxDB* diperoleh berkat optimasi pengindeksan temporal yang efektif, yang juga didukung oleh [16]. Hasil dari uji lapangan ini juga menguatkan pandangan [12] yang menyoroti pentingnya menghilangkan tumpukan komputasi virtual yang berat dalam proses di tingkat *node* sensor. Selain itu, pengembangan stasiun cuaca berbasis mikrokontroler yang terjangkau telah terbukti handal secara fungsional, sebagaimana dibuktikan oleh [20], serta integrasi sensor

mikro untuk agrikultur yang dilaporkan oleh [21].

Metrik kuantitatif terkait dengan keandalan, seperti rendahnya kehilangan paket dan delay yang diperoleh dari sistem *AWS*, konsisten dengan perbandingan empiris latensi pada telemetri nirkabel yang dicatat oleh [10], serta analisis mengenai efisiensi penggunaan daya perangkat *IoT* nirkabel oleh [11]. Dalam konteks beban jaringan, pengurangan ukuran header metadata pada payload *MQTT* terbukti efektif dalam menghindari fragmentasi paket di lapisan transport, sebuah masalah interferensi yang sebelumnya sudah dianalisis secara mendalam oleh [18].

Secara keseluruhan, dengan mempertahankan efisiensi di lapisan transportasi data tanpa penerapan kontainerisasi, prototipe *AWS* berhasil menjaga integritas data meteorologi secara berkelanjutan meskipun diuji dalam kondisi lingkungan luar yang dinamis, sesuai dengan metodologi validasi sistem *IoT* yang ditetapkan oleh [15]. Keberhasilan dalam pengujian multi-protokol secara paralel melalui arsitektur pipa *server* yang terisolasi ini memberikan kontribusi teoretis baru dalam pengembangan model telemetri pertanian presisi yang ekonomis, selaras dengan peta jalan digitalisasi pertanian terbuka yang diusulkan oleh [22] dan kerangka kerja sistem iklim agribisnis terbuka yang diajukan oleh [13].

4. Kesimpulan

Berdasarkan hasil analisis dan pengujian yang dilakukan, penelitian ini berhasil merancang dan mengembangkan prototipe stasiun cuaca otomatis berbasis *Internet of Things (IoT)* yang dapat secara efektif memantau delapan parameter iklim mikro di area Pakebun *Outdoor*. Melalui metode *Rapid Application Development (RAD)* serta pengujian *Black Box*, arsitektur *middleware* dengan dua saluran pada *platform Node-RED* telah terbukti mampu mengolah data dengan stabil. Selain itu, hasil pengujian kualitas layanan jaringan (*QoS*) selama 30,8 jam

menunjukkan bahwa protokol *MQTT* memiliki keunggulan dalam hal keandalan transmisi dibandingkan dengan *HTTP POST* dalam konteks telemetri luar ruangan. Kedua protokol tersebut berhasil memproses 5.768 paket data dengan tingkat kehilangan paket yang sangat rendah, yakni 0,02%. Namun, *MQTT* menunjukkan waktu tunda rata-rata yang lebih baik, yaitu 1.647,53 ms, daripada *HTTP POST* yang mencapai 1.652,05 ms, yang disebabkan oleh adanya pengulangan dalam inisiasi koneksi di setiap sesi pengiriman.

5. Saran

Untuk penerapan praktis di lapangan, studi ini merekomendasikan penggunaan protokol *MQTT* dengan model *publish/subscribe* dalam infrastruktur pertanian terbuka untuk memastikan pengiriman data *time-series* yang cepat, konsisten, dan efisien dalam penggunaan bandwidth. Di sisi lain, untuk pengembangan penelitian ke depan, dianjurkan untuk melakukan uji banding dengan cakupan topologi jaringan nirkabel yang lebih luas (seperti *LoRaWAN*) serta menggabungkan sistem manajemen energi berbasis panel surya untuk menjaga agar perangkat *AWS* di kawasan perkebunan tetap beroperasi secara mandiri.

6. Daftar Pustaka

- [1] N. Islam, M. M. Rashid, F. Pasandideh, B. Ray, S. Moore, and R. Kadel, "A review of applications and communication technologies for internet of things (IoT) and unmanned aerial vehicle (uav) based sustainable smart farming," *Sustainability (Switzerland)*, vol. 13, no. 4, pp. 1–20, Feb. 2021, doi: 10.3390/su13041821.
- [2] A. Nafhah, A. Kahar Bakti, A. Muhtar, N. Zahra Wirantri, and D. Rimantho, "Prosiding Seminar Nasional dalam Rangka Dies Natalis Ke-35 Politeknik Pertanian Negeri Pangkajene Kepulauan. 'Smart Agriculture in Providing Food to Prevent Stunting' Pangkep, 11 Oktober 2023 Seminar Nasional dalam Rangka Dies Natalis ke-35 Politeknik Pertanian Negeri Pangkajene Kepulauan Tahun 2023

- Analisis Dampak Perubahan Iklim Terhadap Produksi Hortikultura di Kabupaten Enrekang Analyzing climate change impacts on horticultural production in Enrekang Regency.”
- [3] M. Santo Gitakarma *et al.*, “Pemantauan Lingkungan Pertanian Dengan Menggunakan Weather Station Berbasis Iot: Solusi Efektif Untuk Petani Di Era Digital,” *Jurnal Komputer dan Teknologi Sains (KOMTEKS)*, vol. 4, no. 1, 2025.
- [4] J. Ikhsan, A. K. Sari, A. Hairani, and A. N. Hidayah, “Correlation Analysis Between Measured Rain Data with Satellite at Rainfall Station in Merapi,” *Semesta Teknika*, vol. 27, no. 1, pp. 107–117, May 2024, doi: 10.18196/st.v27i1.22180.
- [5] D. Solihudin, O. Nurdiawan, R. Kurniawan, and C. L. Rohmat, “OPTIMIZATION IOT TECHNOLOGY IN WEATHER STATIONS FOR IMPROVE AGRICULTURAL SUCCESS DURING EL NIÑO ERA,” *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, vol. 10, no. 2, pp. 315–323, Nov. 2024, doi: 10.33480/jitk.v10i2.5851.
- [6] D. Rusdianto and A. Mustopa, “Analisis Pemanfaatan Sistem Informasi Inventory Barang Di Pakebun (Unit Bisnis Pt Penta Digital Nusantara),” 2023.
- [7] Zahraturrehmi and T. F. Suardi, “Integrasi Iot Dalam Rantai Nilai Agribisnis: Telaah Literatur Terhadap Inovasi Dan Efisiensi Sistem Pangan,” 2025.
- [8] A. Rizzardi, S. Sicari, and A. Coen-Porisini, “Towards rapid modeling and prototyping of indoor and outdoor monitoring applications,” *Sustainable Computing: Informatics and Systems*, vol. 41, Jan. 2024, doi: 10.1016/j.suscom.2023.100951.
- [9] E. Simanjuntak and N. Surantha, “Multiple time series database on microservice architecture for IoT-based sleep monitoring system,” *J. Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40537-022-00658-4.
- [10] S. Bagas Bhaskoro, H. Supriyanto, B. B. Aji, B. Pamungkas, and P. M. Bandung, “Perbandingan Performansi Latency Protokol Komunikasi Http Dan Mqtt Pada Internet Of Things,” *Jurnal Teknologi Terapan*], vol. 8, no. 2, 2022, [Online]. Available: <https://github.com/moscajs/mosca>
- [11] H. J. Jara Ochoa, R. Peña, Y. Ledo Mezquita, E. Gonzalez, and S. Camacho-Leon, “Comparative Analysis of Power Consumption between MQTT and HTTP Protocols in an IoT Platform Designed and Implemented for Remote Real-Time Monitoring of Long-Term Cold Chain Transport Operations,” *Sensors*, vol. 23, no. 10, May 2023, doi: 10.3390/s23104896.
- [12] H. Hugeng, D. Trisnawarman, and A. I. Y. Huntarso, “Enhanced IoT Solution System for Smart Agriculture in Indonesia,” *Green Intelligent Systems and Applications*, vol. 3, no. 2, Nov. 2023, doi: 10.53623/gisa.v3i2.325.
- [13] D. Agustin, “Design Smarthome Application with Rapid Application Development (RAD) Method Based on Hybrid Mobile,” *Jurnal Teknologi Informasi dan Pendidikan*, vol. 16, no. 1, pp. 86–96, Jun. 2023, doi: 10.24036/jtip.v16i1.698.
- [14] B. S. Patro and P. P. Bartakke, “Quality Control (QC) and Quality Assurance (QA) Procedures for Meteorological Data from Automatic Weather Stations,” in *2025 4th International Conference on Range Technology, ICORT 2025*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/ICORT64008.2025.11115447.
- [15] V. G. Ferreira, C. G. Herrera, S. Souza, R. R. Dos Santos, and P. S. L. De Souza, “Software Testing applied to the Development of IoT Systems: preliminary results,” in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Sep. 2023, pp. 113–122. doi: 10.1145/3624032.3624049.
- [16] T. Domínguez-Bolaño, V. Barral, C. J. Escudero, and J. A. García-Naya, “An IoT system for a smart campus: Challenges and solutions illustrated over several real-world use cases,” *Internet of Things (Netherlands)*, vol. 25, Apr. 2024, doi: 10.1016/j.iot.2024.101099.
- [17] F. Susanto, N. Komang Prasiani, and P. Darmawan, “Implementasi Internet Of Things Dalam Kehidupan Sehari-Hari,” Online, 2022. [Online]. Available:

- <https://jurnal.std-bali.ac.id/index.php/imagine>
- [18] H. A. Sifawa, A. B. Tambawal, and S. M. S/Tudu, "Performance Evaluation of IoT Protocols in Resource-Constraints Smart Home Networks," *International Journal of Science for Global Sustainability*, vol. 12, no. 1, pp. 48–61, Mar. 2026, doi: 10.57233/ijsgs.v12i1.1012.
- [19] S. D. Harsono, N. Salma, and R. Ardinal, "Implementation Automatic Packet Reporting System (APRS) for Weather Station," 2022. [Online]. Available: www.aprs.fi
- [20] B. Zamili *et al.*, "Rancang Bangun Alat Ukur Data Iklim Automatic Weather Station (AWS)," 2023, doi: 10.55180/aei.v2i1.766.
- [21] Muhamad Fajar and Adhitia Erfina, "Rancang bangun sistem monitoring curah hujan berbasis internet of things," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 4, no. 1, pp. 42–49, May 2023, doi: 10.37859/coscitech.v4i1.4502.
- [22] A. Reza Thayyib *et al.*, "SISTEMASI: Jurnal Sistem Informasi IoT Based Monitoring System using MQTT Protocol on Tortilla Chips Cutting Machine," 2023. [Online]. Available: <http://sistemasi.ftik.unisi.ac.id>