

Implementasi MQTT pada Sistem Monitoring Suhu dan Kelembapan Ruangan Berbasis IoT

Muhammad Zen^{1*}, Hafni², Muhammad Nurhidayat³

^{1,2,3}Fakultas Sains Komputasi dan Kecerdasan Digital, Program Studi Sistem Komputer, Universitas Pembangunan Panca Budi, Medan, Indonesia

Email: ^{1*}muhammadzen@dosen.pancabudi.ac.id, ²hafni@dosen.pancabudi.ac.id, ³muhammadhidayat0278@gmail.com

(*Email Corresponding Author: muhammadzen@dosen.pancabudi.ac.id)

Received: 25 Juni 2026 | Revision: 30 Agustus 2026 | Accepted: 23 September 2026

Abstrak

Pemantauan suhu dan kelembapan ruangan secara manual memiliki keterbatasan karena tidak menyediakan informasi kondisi lingkungan secara kontinu dan jarak jauh. Penelitian ini mengembangkan sistem monitoring berbasis Internet of Things (IoT) yang mengintegrasikan sensor DHT22, mikrokontroler ESP8266, LCD 16×2, jaringan Wi-Fi, protokol Message Queuing Telemetry Transport (MQTT), broker, dan dashboard web. Tahapan penelitian meliputi analisis kebutuhan, perancangan perangkat keras dan perangkat lunak, implementasi arsitektur publish/subscribe, serta pengujian pembacaan sensor, komunikasi MQTT, dan monitoring real-time. ESP8266 bertindak sebagai publisher yang mengirimkan data suhu dan kelembapan pada topik terpisah, sedangkan dashboard bertindak sebagai subscriber. Hasil implementasi menunjukkan sensor dapat membaca perubahan kondisi ruangan secara berkala, data berhasil ditampilkan pada LCD dan dashboard, dan komunikasi MQTT dapat berlangsung terus-menerus selama koneksi Wi-Fi tersedia. Mekanisme reconnect pada program memungkinkan pengiriman dilanjutkan setelah koneksi tersedia kembali. Pada dokumentasi pengujian, dashboard menampilkan suhu 37,10 °C dan kelembapan 63,00%, sedangkan pengujian perangkat memperlihatkan pembaruan nilai secara langsung. Hasil ini menunjukkan arsitektur ESP8266-DHT22-MQTT dapat digunakan sebagai solusi monitoring lingkungan berbiaya rendah dan mudah dikembangkan. Namun, laporan pengujian belum menyediakan pengukuran kuantitatif delay, packet loss, throughput, maupun galat sensor terhadap instrumen referensi, sehingga evaluasi QoS dan kalibrasi menjadi agenda pengembangan berikutnya.

Kata Kunci: DHT22, ESP8266, Internet of Things, MQTT, Monitoring Real-Time

Abstract

Manual room temperature and humidity monitoring is limited because it does not provide continuous and remote information about environmental conditions. This study develops an Internet of Things (IoT)-based monitoring system integrating a DHT22 sensor, ESP8266 microcontroller, 16×2 LCD, Wi-Fi network, Message Queuing Telemetry Transport (MQTT), broker, and web dashboard. The research stages include requirement analysis, hardware and software design, implementation of the publish/subscribe architecture, and testing of sensor acquisition, MQTT communication, and real-time monitoring. The ESP8266 acts as a publisher that sends temperature and humidity data through separate topics, while the dashboard acts as a subscriber. The implementation shows that the sensor can periodically detect room-condition changes, data can be displayed on both the LCD and dashboard, and MQTT communication can operate continuously while Wi-Fi connectivity is available. The reconnect mechanism allows data transmission to resume after the connection is restored. During documented testing, the dashboard displayed 37.10 °C and 63.00% relative humidity, while the device test showed direct value updates. These results indicate that the ESP8266-DHT22-MQTT architecture is suitable as a low-cost and extensible environmental monitoring solution. However, quantitative delay, packet loss, throughput, and reference-instrument error were not recorded; therefore, QoS evaluation and sensor calibration remain priorities for further work.

Keywords: DHT22, ESP8266, Internet Of Things, MQTT, Real-Time Monitoring

1. PENDAHULUAN

Pemantauan kondisi lingkungan ruangan merupakan kebutuhan penting pada ruang kerja, laboratorium, ruang server, ruang penyimpanan, dan fasilitas pelayanan yang memerlukan informasi suhu serta kelembapan secara berkala. Pemantauan manual hanya memberikan potret kondisi pada saat pengukuran dilakukan, sehingga perubahan di antara dua waktu pengamatan dapat terlewat. Internet of Things (IoT) menawarkan pendekatan yang menghubungkan sensor, perangkat komputasi tertanam, jaringan, dan aplikasi sehingga data lingkungan dapat dikumpulkan serta diakses secara jarak jauh. Arsitektur IoT modern juga menuntut protokol komunikasi yang sesuai dengan keterbatasan memori, prosesor, energi, dan bandwidth pada perangkat ujung [1], [2].

Pada lapisan aplikasi IoT, Message Queuing Telemetry Transport (MQTT) banyak digunakan karena menerapkan pola publish/subscribe melalui broker. Pemisahan publisher dan subscriber membuat perangkat sensor tidak perlu mengetahui lokasi aplikasi penerima secara langsung. Kajian protokol IoT menunjukkan bahwa karakteristik protokol

aplikasi memengaruhi trafik, reliabilitas, bandwidth, dan konsumsi sumber daya [2]. Evaluasi sistem publish/subscribe juga menegaskan bahwa fitur broker, kebijakan pengiriman, dan konfigurasi Quality of Service (QoS) berpengaruh terhadap karakteristik layanan [3]. Dengan karakteristik tersebut, MQTT relevan untuk pengiriman data sensor berukuran kecil dan periodik pada sistem monitoring lingkungan.

Implementasi monitoring berbasis MQTT telah dikaji pada beragam skenario. Diono dkk. mengembangkan jaringan sensor menggunakan NodeMCU ESP8266 dengan MQTT, di mana node sensor berperan sebagai publisher dan sistem monitoring sebagai subscriber; data dari beberapa node dapat ditampilkan pada sistem monitoring [4]. Widjaja dkk. mengembangkan monitoring kondisi udara ruang kelas menggunakan ESP8266/ESP32, DHT22, gateway, basis data lokal, dan layanan cloud; rancangan tersebut memperlihatkan manfaat penyimpanan lokal sebagai penyangga ketika koneksi cloud tidak tersedia [5]. Hendinata dan Fikri mengembangkan monitoring suhu dan kelembapan budidaya jamur menggunakan NodeMCU ESP8266 dan dashboard web, yang menekankan kemudahan pemantauan kondisi lingkungan dari jarak jauh [6].

Penelitian yang lebih baru memperlihatkan kecenderungan evaluasi yang semakin kuantitatif. Surbakti dkk. mengimplementasikan DHT22, ESP32, dan MQTT untuk monitoring kualitas udara ruangan [7]. Puthiyidam dan Joseph menunjukkan bahwa ukuran pesan, jumlah klien, serta tingkat QoS memengaruhi latensi komunikasi MQTT [8]. Arifin dkk. mengevaluasi optimasi broker pada QoS 0, 1, dan 2 menggunakan metrik throughput, delay, jitter, response time, dan penggunaan CPU [9]. Utomo dan Izzaturrahmani mengintegrasikan MQTT dengan Node-RED dan melaporkan akurasi monitoring di atas 98,4% dengan galat rata-rata suhu 0,4 °C pada skenario pengujiannya [10]. Temuan-temuan tersebut menunjukkan bahwa keberhasilan fungsi real-time perlu dibedakan dari evaluasi performa jaringan dan akurasi sensor secara terukur.

Perkembangan pada 2025-2026 juga memperkuat kebutuhan terhadap ketahanan koneksi dan pencatatan data. Guda dkk. mengembangkan sistem ESP8266-DHT22-MQTT dengan mekanisme toleransi gangguan sensor, Wi-Fi, dan server serta kemampuan reconnect [11]. Sultan menggabungkan DHT22, MQTT QoS 1, Node-RED, peringatan real-time, dan analitik prediktif untuk monitoring lingkungan [12]. Wirasapta dan Pamungkas mengukur performa end-to-end jembatan BLE-Wi-Fi-MQTT pada monitoring suhu dan kelembapan [13], sedangkan Tomas dkk. mengintegrasikan ESP32, MQTT, Node-RED, dan MySQL untuk monitoring serta penyimpanan historis [14]. Arah penelitian tersebut menunjukkan bahwa sistem monitoring yang baik tidak hanya menampilkan nilai sensor, tetapi juga perlu menyediakan mekanisme pemulihan koneksi, histori data, dan metrik performa komunikasi.

Berdasarkan penelitian terkait tersebut, gap pada penelitian ini berada pada implementasi sederhana dan berbiaya rendah untuk lingkungan kantor desa dengan integrasi pembacaan lokal dan monitoring jarak jauh dalam satu alur. Sistem menggunakan DHT22 sebagai sensor, ESP8266 sebagai pengendali sekaligus antarmuka Wi-Fi, LCD 16×2 sebagai tampilan lokal, serta MQTT untuk meneruskan data ke dashboard web. Berbeda dari studi yang berfokus pada optimasi broker atau analitik lanjutan, penelitian ini menitikberatkan pada keterlaksanaan arsitektur end-to-end dan mekanisme reconnect pada perangkat. Di sisi lain, laporan eksperimen yang menjadi dasar artikel ini belum merekam delay, jitter, packet loss, throughput, maupun data pembandingan sensor secara numerik. Karena itu, klaim hasil dibatasi pada keberhasilan fungsional dan observasi kestabilan selama pengujian, bukan pada pembuktian QoS jaringan atau akurasi metrologis.

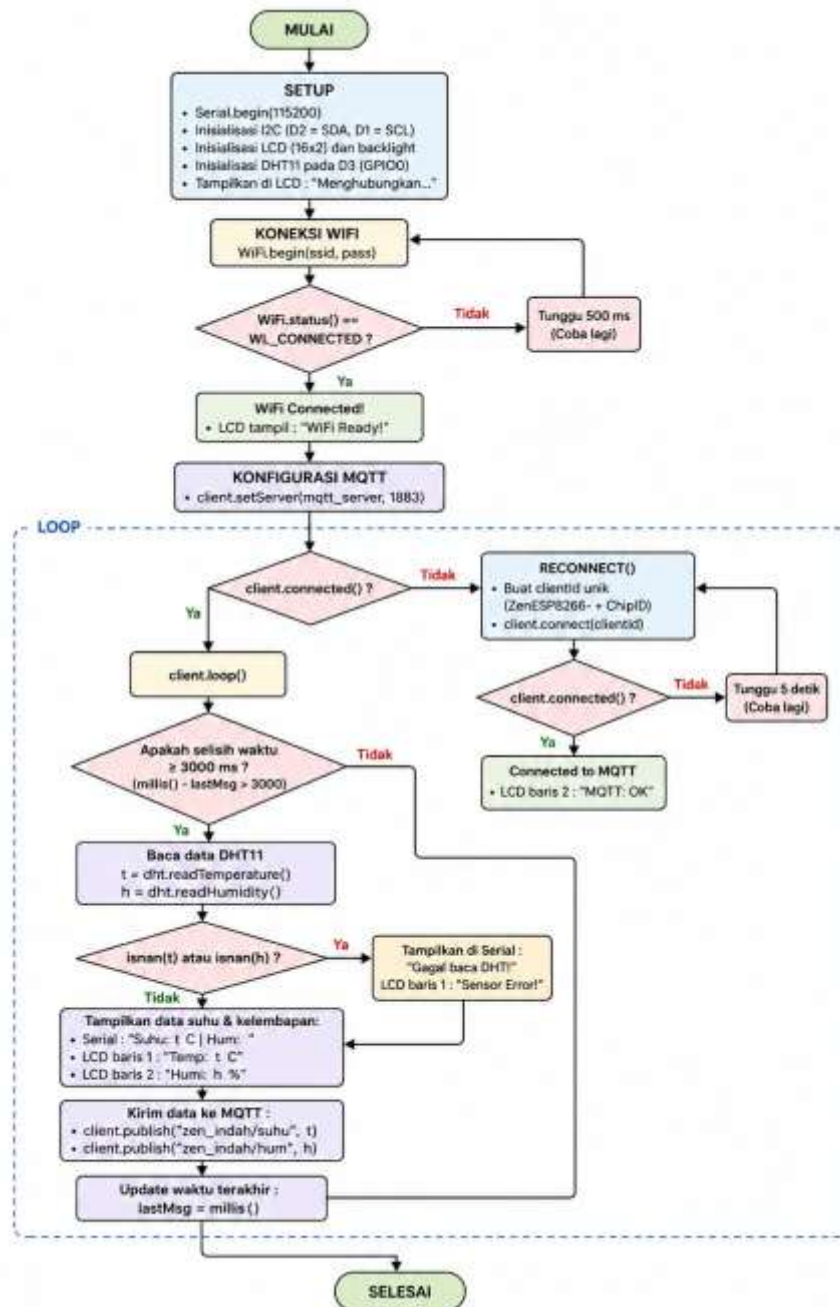
Tujuan penelitian adalah merancang dan mengimplementasikan sistem monitoring suhu dan kelembapan ruangan berbasis IoT menggunakan ESP8266 dan protokol MQTT, menguji aliran data dari sensor menuju LCD dan dashboard, serta mengamati kemampuan sistem melanjutkan komunikasi setelah koneksi tersedia kembali. Implementasi dilakukan di Kantor Desa Klambir V Kebon, Kecamatan Hamparan Perak, Kabupaten Deli Serdang. Hasil penelitian diharapkan menjadi dasar pengembangan monitoring lingkungan yang lebih terukur melalui penambahan basis data historis, notifikasi, pengujian QoS, keamanan komunikasi, dan kalibrasi sensor pada penelitian lanjutan.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian menggunakan pendekatan perancangan dan pengujian prototipe. Tahapan dimulai dari identifikasi kebutuhan monitoring di Kantor Desa Klambir V Kebon, dilanjutkan studi literatur mengenai IoT, ESP8266, DHT22, MQTT, dan dashboard web. Setelah kebutuhan ditetapkan, dilakukan perancangan perangkat keras, perancangan perangkat lunak, integrasi jaringan Wi-Fi dan broker MQTT, implementasi dashboard, kemudian pengujian fungsional. Data penelitian berupa pembacaan suhu dan kelembapan serta dokumentasi keberhasilan pengiriman dan penampilan data pada LCD dan dashboard.

Alur perangkat lunak disusun agar perangkat melakukan inisialisasi serial, I2C, LCD, sensor, dan Wi-Fi sebelum konfigurasi MQTT. Pada loop utama, koneksi MQTT diperiksa terlebih dahulu. Jika terputus, fungsi reconnect membuat client ID dan mencoba menghubungkan perangkat kembali. Setelah koneksi tersedia, sensor dibaca secara periodik, nilai diperiksa, ditampilkan pada serial dan LCD, kemudian dipublikasikan ke topik MQTT. Pendekatan ini mencegah proses publish dijalankan ketika client belum terhubung dan memberi mekanisme pemulihan dasar saat koneksi terputus.



Gambar 1. Alur program monitoring suhu dan kelembapan berbasis MQTT

Gambar 1 memperlihatkan urutan inisialisasi, koneksi Wi-Fi, konfigurasi broker pada port 1883, pengecekan status client, pembacaan sensor periodik, penampilan lokal, publish data, dan mekanisme reconnect. Pada dokumentasi program, interval pembacaan ditetapkan sekitar 3 detik. Diagram pada laporan menggunakan label DHT11 pada beberapa blok, sedangkan perangkat, uraian metode, dan implementasi penelitian menggunakan DHT22; artikel ini mengikuti perangkat DHT22 yang dinyatakan pada rancangan dan hasil implementasi.

2.2 Arsitektur Sistem dan Perangkat

Arsitektur sistem terdiri atas lapisan sensing, edge processing, communication, dan application. DHT22 membaca suhu dan kelembapan, ESP8266 memproses data serta menyediakan konektivitas Wi-Fi, LCD 16×2 menampilkan data lokal, sedangkan MQTT mengirimkan data ke broker untuk diteruskan kepada dashboard sebagai subscriber. Struktur ini mengikuti pola publisher-broker-subscriber yang lazim pada MQTT [3], [4]. Pemisahan tersebut membuat dashboard tidak berkomunikasi langsung dengan sensor dan memungkinkan pengembangan subscriber lain tanpa mengubah proses akuisisi pada perangkat.

Tabel 1. Komponen utama sistem

Komponen	Fungsi
DHT22	Akuisisi suhu dan kelembapan ruangan
ESP8266/NodeMCU	Pemrosesan data, koneksi Wi-Fi, dan publisher MQTT
LCD 16×2 I2C	Tampilan lokal nilai suhu dan kelembapan
Broker MQTT	Perantara distribusi pesan publish/subscribe
Dashboard web	Subscriber dan visualisasi data real-time
Catu daya 5 V	Sumber daya rangkaian prototipe

2.3 Implementasi MQTT dan Akuisisi Data

ESP8266 dikonfigurasi sebagai publisher. Data suhu dipublikasikan pada topik zen_indah/suhu dan data kelembapan pada topik zen_indah/hum. Dashboard berlangganan topik tersebut dan menampilkan nilai yang diterima. Pemisahan topik memudahkan pengelolaan variabel pada sisi subscriber. Pada program, `client.loop()` dijalankan selama client terhubung dan fungsi `reconnect` dipanggil ketika koneksi MQTT terputus. Mekanisme `reconnect` ini sejalan dengan kebutuhan ketahanan konektivitas pada perangkat single-core yang juga menjadi fokus studi Guda dkk. [11].

Pengumpulan data dilakukan melalui observasi lokasi, eksperimen perangkat, pembacaan sensor secara periodik, dan dokumentasi tampilan sistem. Analisis bersifat deskriptif terhadap keberhasilan pembacaan, pengiriman, penerimaan, dan visualisasi data. Laporan penelitian menyebut rencana membandingkan sensor dengan alat ukur pembanding, tetapi tidak menyertakan tabel pasangan data referensi-sensor atau perhitungan galat. Oleh karena itu, artikel ini tidak menghitung akurasi, MAE, RMSE, atau persentase error agar tidak menghasilkan angka yang tidak didukung data eksperimen.

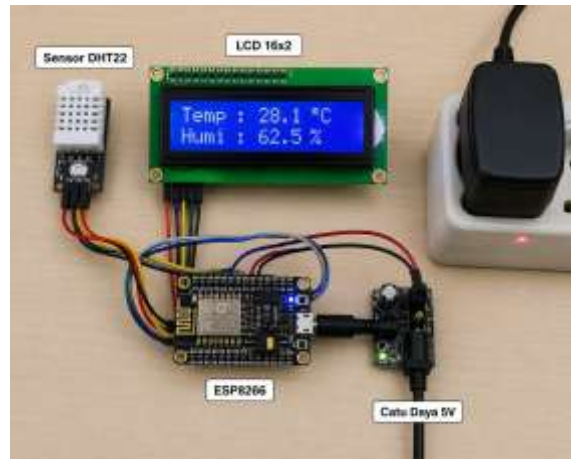
2.4 Skenario Pengujian

Pengujian dibagi menjadi empat skenario: (1) pengujian integrasi perangkat untuk memastikan DHT22, ESP8266, LCD, dan catu daya berfungsi; (2) pengujian pembacaan sensor untuk mengamati perubahan nilai suhu dan kelembapan; (3) pengujian komunikasi MQTT untuk memastikan publisher dapat mengirim pesan ke broker dan dashboard menerima pesan sebagai subscriber; serta (4) pengujian monitoring real-time untuk memastikan nilai pada dashboard diperbarui selama perangkat terhubung. Selain itu diamati perilaku `reconnect` setelah Wi-Fi tersedia kembali. Parameter kuantitatif jaringan belum direkam pada laporan, sehingga evaluasi komunikasi dinyatakan sebagai hasil fungsional dan observasional.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Implementasi Perangkat

Prototipe berhasil mengintegrasikan sensor DHT22, LCD 16×2, ESP8266, dan catu daya 5 V. Sensor menjadi sumber data, ESP8266 melakukan pembacaan dan komunikasi jaringan, sedangkan LCD menyediakan informasi lokal tanpa harus membuka dashboard. Dokumentasi perangkat memperlihatkan seluruh komponen terhubung dalam satu rangkaian dan LCD dapat menampilkan nilai suhu serta kelembapan. Keberadaan tampilan lokal penting karena pengguna masih dapat melihat nilai terakhir yang dibaca pada perangkat meskipun akses dashboard terganggu.



Gambar 2. Implementasi DHT22, LCD 16×2, ESP8266, dan catu daya

Pada Gambar 2, LCD mendokumentasikan pembacaan 28,1 °C dan 62,5% RH. Nilai ini digunakan sebagai bukti fungsi tampilan lokal, bukan sebagai nilai rata-rata eksperimen. Penggunaan ESP8266 relevan untuk prototipe IoT berbiaya rendah karena modul menyediakan Wi-Fi terintegrasi dan telah digunakan pada berbagai sistem monitoring lingkungan [4]-[6]. Namun, untuk implementasi operasional jangka panjang, rangkaian perlu ditempatkan pada enclosure yang memadai dan tata kabel perlu dibuat lebih permanen untuk mengurangi risiko sambungan longgar.

3.2 Hasil Pembacaan Sensor DHT22

Pengujian DHT22 menunjukkan sensor dapat memberikan data suhu dan kelembapan secara berkala dan perubahan nilai dapat diteruskan ke ESP8266. Pada dokumentasi pengujian lain, LCD menampilkan suhu 36,90 °C dan kelembapan 64,00% RH. Perbedaan nilai terhadap dokumentasi pada Gambar 2 menunjukkan bahwa sistem dapat memperbarui hasil pembacaan ketika kondisi sensor berubah. Akan tetapi, variasi tersebut tidak dapat digunakan untuk menilai akurasi karena tidak tersedia pembacaan simultan dari instrumen referensi yang terkalibrasi.



Gambar 3. Dokumentasi pengujian pembacaan suhu dan kelembapan

Keterbatasan ini penting karena penelitian terkait telah menunjukkan manfaat evaluasi kuantitatif sensor. Utomo dan Izzaturrahmani, misalnya, membandingkan pembacaan sensor pada kondisi berbeda dan melaporkan metrik akurasi serta galat [10]. Studi monitoring HVAC juga menggunakan deviasi pengukuran untuk menilai kesesuaian sistem [15]. Dengan demikian, tahap berikutnya sebaiknya menggunakan termohigrometer referensi, beberapa titik kondisi, dan pengulangan untuk menghitung bias, MAE, RMSE, serta rentang error suhu dan kelembapan.

3.3 Hasil Komunikasi MQTT

Komunikasi MQTT berjalan sesuai pola publish/subscribe yang dirancang. ESP8266 mengirimkan nilai suhu dan kelembapan ke broker menggunakan topik yang telah ditentukan, kemudian dashboard menerima dan menampilkan data. Selama koneksi internet tersedia, pengiriman dilakukan berulang. Laporan pengujian juga mencatat bahwa ketika koneksi Wi-Fi terputus dan kemudian tersedia kembali, perangkat dapat melanjutkan proses pengiriman melalui mekanisme

reconnect tanpa konfigurasi ulang manual. Perilaku ini menunjukkan bahwa fungsi pemulihan koneksi dasar telah bekerja pada tingkat aplikasi.

Hasil tersebut konsisten dengan implementasi MQTT pada jaringan sensor oleh Diono dkk. [4] dan monitoring berbasis Node-RED oleh Utomo dan Izzaturrahmani [10]. Meski demikian, istilah 'cepat', 'ringan', dan 'stabil' dalam laporan belum disertai data numerik. Penelitian performa MQTT menunjukkan bahwa latensi dipengaruhi ukuran pesan, jumlah klien, dan QoS [8], sementara konfigurasi broker juga dapat memengaruhi delay, jitter, response time, throughput, dan penggunaan CPU [9]. Karena penelitian ini belum merekam timestamp publisher-subscriber atau packet capture, kesimpulan yang dapat dipertahankan adalah bahwa pesan berhasil diteruskan dan dashboard diperbarui selama skenario uji, bukan bahwa sistem telah memenuhi ambang QoS tertentu.

Tabel 2. Ringkasan hasil pengujian fungsional

Aspek	Hasil observasi	Status
Pembacaan DHT22	Nilai suhu dan kelembapan terbaca dan berubah sesuai kondisi pengujian	Berfungsi
Tampilan LCD	Nilai suhu dan kelembapan ditampilkan pada LCD 16×2	Berfungsi
Koneksi Wi-Fi	ESP8266 dapat terhubung ke jaringan selama akses tersedia	Berfungsi
Publish MQTT	Data suhu dan kelembapan dikirim melalui topik terpisah	Berfungsi
Dashboard subscriber	Data diterima dan ditampilkan secara real-time	Berfungsi
Reconnect	Pengiriman dapat dilanjutkan setelah koneksi tersedia kembali	Berfungsi
QoS kuantitatif	Delay, jitter, packet loss, throughput belum direkam	Belum dievaluasi
Akurasi metrologis	Data instrumen referensi dan galat belum tersedia	Belum dievaluasi

3.4 Monitoring Data Secara Real-Time

Dashboard web berhasil menampilkan variabel suhu dan kelembapan sebagai dua indikator yang dapat diakses melalui perangkat yang terhubung ke internet. Dokumentasi pengujian memperlihatkan suhu 37,10 °C dan kelembapan 63,00% RH. Karena nilai pada dashboard berasal dari pesan MQTT yang diterbitkan ESP8266, keberhasilan tampilan tersebut mengonfirmasi rantai data sensing-processing-publish-broker-subscribe-visualization telah tersambung secara end-to-end.



Gambar 4. Dashboard monitoring suhu dan kelembapan secara real-time

Visualisasi sederhana pada Gambar 4 sesuai untuk kebutuhan pemantauan kondisi terkini. Namun, dashboard belum memperlihatkan grafik historis, timestamp, status koneksi, maupun alarm ambang. Tomas dkk. mengintegrasikan MQTT, Node-RED, dan MySQL sehingga data real-time juga dapat disimpan sebagai histori [14], sedangkan Sultan menambahkan peringatan dan analitik prediktif [12]. Pengembangan serupa akan meningkatkan nilai operasional sistem karena pengguna tidak hanya mengetahui kondisi saat ini, tetapi juga dapat meninjau tren dan menerima notifikasi ketika parameter melewati batas yang ditetapkan.

3.5 Analisis terhadap Penelitian Terkait

Dibandingkan penelitian terkait, kontribusi sistem ini terletak pada integrasi sederhana ESP8266-DHT22-LCD-MQTT-dashboad untuk lingkungan kantor desa serta mekanisme reconnect yang sudah tertanam pada program. Sistem Diono dkk. [4] menggunakan beberapa node dan Node-RED; Widjaja dkk. [5] menambahkan gateway dan penyimpanan lokal; Guda dkk. [11] menguji toleransi kegagalan secara lebih sistematis; sedangkan studi 2026 mulai menekankan pengukuran end-to-end dan penyimpanan historis [13], [14]. Posisi penelitian ini lebih tepat sebagai prototipe fungsional yang membuktikan keterhubungan seluruh komponen dengan biaya dan kompleksitas rendah.

Tabel 3. Perbandingan penelitian terkait dan posisi penelitian

Studi	Platform/Protokol	Fokus utama
[4] Diono dkk., 2021	ESP8266, MQTT, Node-RED	Monitoring beberapa sensor node
[5] Widjaja dkk., 2022	ESP8266/ESP32, DHT22, MQTT	Monitoring indoor dengan gateway dan penyimpanan
[7] Surbakti dkk., 2024	ESP32, DHT22, MQTT	Monitoring kualitas udara ruangan
[10] Utomo & Izzaturrahmani, 2025	MQTT, Node-RED	Monitoring dan evaluasi akurasi sensor
[11] Guda dkk., 2025	ESP8266, DHT22, MQTT	Fault tolerance dan reconnect
[13] Wirasapta & Pamungkas, 2026	ESP32, BLE, MQTT, EMQX	Pengukuran performa end-to-end
Penelitian ini	ESP8266, DHT22, LCD, MQTT, dashboard	Integrasi end-to-end dan monitoring kantor desa

Gap yang masih tersisa adalah validasi kuantitatif. Untuk meningkatkan kekuatan eksperimen, penelitian lanjutan perlu mengukur latency end-to-end dengan timestamp pada publisher dan subscriber, packet delivery ratio/packet loss, throughput, jitter, serta pengaruh QoS 0, 1, dan 2. Pengujian juga perlu dilakukan pada durasi panjang dan beberapa kondisi kualitas Wi-Fi. Dari sisi sensor, kalibrasi terhadap instrumen referensi harus menghasilkan tabel pasangan data dan metrik error. Selain itu, penggunaan autentikasi broker dan TLS perlu dipertimbangkan karena keamanan merupakan tantangan penting pada implementasi MQTT dan arsitektur IoT [16], [17].

4. KESIMPULAN

Penelitian berhasil merancang dan mengimplementasikan sistem monitoring suhu dan kelembapan ruangan berbasis Internet of Things menggunakan sensor DHT22, ESP8266, LCD 16×2, jaringan Wi-Fi, protokol MQTT, broker, dan dashboard web. Arsitektur publish/subscribe dapat mengalirkan data dari sensor menuju dashboard, sedangkan LCD menyediakan tampilan lokal pada perangkat. Pengujian fungsional menunjukkan pembacaan sensor dapat diperbarui secara periodik, data dapat dipublikasikan melalui topik suhu dan kelembapan, dashboard dapat menerima serta menampilkan nilai secara real-time, dan mekanisme reconnect memungkinkan proses komunikasi dilanjutkan ketika koneksi tersedia kembali. Dokumentasi eksperimen memperlihatkan beberapa kondisi pembacaan, termasuk tampilan dashboard 37,10 °C dan 63,00% RH, yang digunakan sebagai bukti keberhasilan fungsi visualisasi dan bukan sebagai statistik keseluruhan pengujian. Dengan demikian, prototipe menunjukkan bahwa kombinasi ESP8266-DHT22-MQTT dapat menjadi dasar solusi monitoring lingkungan berbiaya rendah untuk ruang kerja atau fasilitas sejenis. Namun, laporan eksperimen belum menyertakan data numerik delay, jitter, packet loss, throughput, QoS, maupun pasangan pengukuran sensor terhadap instrumen referensi. Oleh sebab itu, penelitian ini belum dapat menyatakan nilai performa jaringan atau akurasi metrologis secara kuantitatif. Pengembangan berikutnya perlu menambahkan basis data historis, notifikasi ambang, pengujian QoS 0-2, pencatatan timestamp end-to-end, uji kestabilan jangka panjang, kalibrasi DHT22 terhadap termohigrometer referensi, serta autentikasi dan enkripsi komunikasi MQTT. Penguatan tersebut akan mengubah prototipe fungsional menjadi sistem monitoring yang dapat dievaluasi secara objektif dari aspek akurasi, reliabilitas, performa, dan keamanan.

REFERENCES

- [1] M. Lombardi, F. Pascale, and D. Santaniello, "Internet of Things: A General Overview between Architectures, Protocols and Applications," *Information*, vol. 12, no. 2, p. 87, 2021. doi: <https://doi.org/10.3390/info12020087>
- [2] C. Bayılmış, M. A. Ebleme, Ü. Çavuşoğlu, K. Küçük, and A. Sevin, "A survey on communication protocols and performance evaluations for Internet of Things," *Digital Communications and Networks*, vol. 8, no. 6, pp. 1094–1104, 2022. doi: <https://doi.org/10.1016/j.dcan.2022.03.013>

- [3] A. Lazidis, K. Tsakos, and E. G. M. Petrakis, "Publish-Subscribe approaches for the IoT and the cloud: Functional and performance evaluation of open-source systems," *Internet of Things*, vol. 19, p. 100538, 2022. doi: <https://doi.org/10.1016/j.iot.2022.100538>
- [4] M. Diono, H. Azwar, and W. Khabzli, "Sistem Monitoring Jaringan Sensor Node Berbasis Protokol MQTT," *Jurnal Elektro dan Mesin Terapan*, vol. 7, no. 2, pp. 120–126, 2021. doi: <https://doi.org/10.35143/elementer.v7i2.5232>
- [5] M. Widjaja, D. K. Halim, and R. Andarini, "The Development of an IoT-based Indoor Air Monitoring System Towards Smart Energy Efficient Classroom," *Ultima Computing: Jurnal Sistem Komputer*, vol. 14, no. 1, pp. 28–35, 2022. doi: <https://doi.org/10.31937/sk.v14i1.2565>
- [6] L. K. Hendinata and A. I. R. Fikri, "Development of IoT-Based Temperature and Relative Humidity Monitoring System for Mushroom Cultivation House," *Jurnal Ecotipe*, vol. 10, no. 1, pp. 95–102, 2023. doi: <https://doi.org/10.33019/jurnalecotipe.v10i1.3919>
- [7] H. B. Surbakti, J. G. A. Ginting, S. Romadhona, M. B. Ginting, and K. Ni'amah, "Sistem Monitoring Kualitas Udara Ruangan dengan Protokol MQTT Berbasis Internet of Things," *Jurnal SINTA: Sistem Informasi dan Teknologi Komputasi*, vol. 1, no. 3, pp. 129–137, 2024. doi: <https://doi.org/10.61124/sinta.v1i3.25>
- [8] J. J. Puthiyidam and S. Joseph, "Internet of Things Network Performance: Impact of Message and Client Sizes and Reliability Levels," *ECTI Transactions on Electrical Engineering, Electronics, and Communications*, vol. 22, no. 1, 2024. doi: <https://doi.org/10.37936/ecti-ee.2024221.252941>
- [9] S. Arifin, A. W. Nugraha, F. S. Mukti, and S. Jatmika, "MQTT Broker Optimization: Comparative Analysis of Round Robin and Least Response Time," *Jurnal Nasional Teknik Elektro*, vol. 13, no. 3, pp. 127–136, 2024. doi: <https://doi.org/10.25077/jnte.v13n3.1260.2024>
- [10] E. B. Utomo and N. Izzaturrahmani, "Temperature and Humidity Monitoring System Using Node-Red Based on MQTT Protocol," *Journal of Electrical Engineering and Computer*, vol. 7, no. 1, pp. 40–48, 2025. doi: <https://doi.org/10.33650/jeeecom.v7i1.10658>
- [11] C. A. Guda, J. Hasan, S. Islam, M. S. Rahman, and M. H. Bhuyan, "Design, Simulation, and Implementation of an IIoT-based Temperature and Humidity Monitoring System with Single-Core Infinite Loop Prevention and Fault Tolerance for Multi-Sensor and Connectivity Failures," *Journal of Innovation and Technology*, vol. 2025, no. 2, pp. 1–13, 2025. doi: <https://doi.org/10.61453/joit.v2025no31>
- [12] M. M. Sultan, "Enhanced IoT-Based Environmental Monitoring System Using MQTT with Real-Time Alerts and Predictive Analytics," *IAR Journal of Engineering and Technology*, vol. 6, no. 1, pp. 1–5, 2025. doi: <https://doi.org/10.47310/iarjet.2025.v06i01.003>
- [13] A. H. Wirasapta and T. D. Pamungkas, "Real-Time IoT Temperature Monitoring Using Xiaomi Mijia BLE, ESP32, MQTT, and EMQX," *Journal of Applied Electrical Engineering*, vol. 10, no. 1, pp. 23–28, 2026. doi: <https://doi.org/10.30871/jaee.v10i1.12012>
- [14] M. Tomas, R. Budiman, Baharuddin, R. Nofendra, H. Andre, and E. Alfionita, "Development and Performance Evaluation of a Real-Time Environmental Monitoring System Using ESP32, MQTT, Node-RED, and MySQL Database," *Jurnal Andalas: Rekayasa dan Penerapan Teknologi*, vol. 6, no. 1, pp. 47–59, 2026. doi: <https://doi.org/10.25077/jarpet.v6i1.143>
- [15] M. F. Habillah, "Design of an IoT-Based Temperature and Humidity Monitoring System for Efficient HVAC Data Collection in Airport," *Journal of Airport Engineering Technology*, vol. 5, no. 2, 2025. doi: <https://doi.org/10.52989/jaet.v5i2.196>
- [16] Rachit, S. Bhatt, and P. R. Ragiri, "Security trends in Internet of Things: a survey," *SN Applied Sciences*, vol. 3, art. 121, 2021. doi: <https://doi.org/10.1007/s42452-021-04156-9>
- [17] H. Kurdi and V. Thayananthan, "Authentication mechanisms for IoT system based on distributed MQTT brokers: review and challenges," *Procedia Computer Science*, vol. 194, pp. 132–139, 2021. doi: <https://doi.org/10.1016/j.procs.2021.10.066>