

Implementasi Smart Home Menggunakan MQTT Broker Lokal untuk Kontrol Lampu dan Monitoring Suhu Berbasis ESP8266

Dandi Yohananda Saputra Utama¹, Muawan Bisri², Nazril Abdillah³

¹²³ Universitas Indonesia Mandiri

dandiyohanandasaputra@uimandiri.ac.id, muawanbisri@uimandiri.ac.id, nazrilabdillah@uimandiri.ac.id

Abstrak

Perkembangan Internet of Things (IoT) mendorong penerapan sistem smart home untuk meningkatkan kenyamanan dan efisiensi pengendalian perangkat rumah tangga. Penelitian ini bertujuan mengimplementasikan sistem smart home menggunakan MQTT broker lokal untuk kontrol lampu dan monitoring suhu berbasis ESP8266. Sistem dirancang menggunakan arsitektur publish/subscribe dengan Mosquitto sebagai broker MQTT pada jaringan lokal (LAN), NodeMCU ESP8266 sebagai pengendali, sensor DHT22 sebagai pembaca suhu, dan Node-RED sebagai antarmuka pengguna. Pengujian dilakukan untuk mengevaluasi kinerja sistem berdasarkan parameter fungsionalitas, waktu tunda (delay), dan stabilitas koneksi. Hasil pengujian menunjukkan tingkat keberhasilan kontrol lampu sebesar 100%. Rata-rata delay komunikasi sebesar 90,5 ms pada QoS 0 dan 113,5 ms pada QoS 1. Pengujian stabilitas selama 24 jam menunjukkan sistem berjalan tanpa gangguan koneksi maupun kegagalan pengiriman data. Hasil penelitian menunjukkan bahwa penggunaan MQTT broker lokal mampu memberikan respons yang cepat, stabil, dan tidak bergantung pada koneksi internet. Sistem ini layak diterapkan pada lingkungan rumah tangga skala kecil sebagai solusi otomasi yang sederhana dan ekonomis.

Kata kunci: Smart Home, Internet of Things, MQTT, ESP8266, Broker Lokal.

Abstract

The development of the Internet of Things (IoT) has encouraged the implementation of smart home systems to improve comfort and efficiency in controlling household devices. This study aims to implement a smart home system using a local MQTT broker for lamp control and temperature monitoring based on ESP8266. The system was designed using a publish/subscribe architecture with Mosquitto as the MQTT broker within a Local Area Network (LAN), NodeMCU ESP8266 as the controller, DHT22 as the temperature sensor, and Node-RED as the user interface. System performance was evaluated based on functionality, communication delay, and connection stability. The experimental results show a 100% success rate in lamp control execution. The average communication delay was 90.5 ms for QoS 0 and 113.5 ms for QoS 1. A 24-hour stability test demonstrated continuous operation without connection loss or data transmission failure. The results indicate that the use of a local MQTT broker provides fast response, stable communication, and independence from internet connectivity. The proposed system is suitable for small-scale household automation applications as a simple and cost-effective solution.

Keywords: Smart Home, Internet of Things, MQTT, ESP8266, Local Broker.

1. Pendahuluan

Perkembangan Internet of Things (IoT) telah mendorong integrasi perangkat fisik dengan jaringan komunikasi untuk mendukung otomatisasi dan pertukaran data secara real-time. Dalam konteks rumah tangga, konsep *smart home* menjadi salah satu implementasi IoT yang berkembang pesat karena mampu meningkatkan kenyamanan, efisiensi energi, serta efektivitas pengendalian perangkat listrik (Mayub et al., 2025). Implementasi sistem *smart home* memungkinkan pengguna mengontrol dan memantau perangkat seperti lampu, kipas, dan sensor lingkungan melalui jaringan lokal maupun internet.

Salah satu protokol komunikasi yang banyak digunakan dalam sistem IoT adalah Message Queuing Telemetry Transport (MQTT). MQTT merupakan protokol ringan berbasis *publish/subscribe* yang dirancang untuk perangkat dengan sumber daya terbatas dan jaringan dengan bandwidth rendah (Susanto et al., 2018). Karakteristik tersebut menjadikan MQTT sesuai untuk diterapkan pada mikrokontroler seperti ESP8266 dalam sistem otomasi rumah. Beberapa penelitian di Indonesia menunjukkan bahwa MQTT efektif digunakan untuk komunikasi data sensor dan pengendalian perangkat secara real-time dengan konsumsi sumber daya yang relatif kecil (Gunawan Mardianto et al., 2023).

Meskipun demikian, sebagian besar implementasi sistem *smart home* masih menggunakan broker MQTT berbasis cloud. Ketergantungan terhadap layanan cloud dapat menimbulkan beberapa kendala, seperti latensi komunikasi, ketergantungan terhadap koneksi internet, serta potensi risiko keamanan data. Penelitian terkait implementasi broker MQTT lokal sebagai pusat komunikasi dalam jaringan lokal masih relatif terbatas, khususnya dalam pengujian performa delay dan stabilitas sistem pada lingkungan rumah tinggal.

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan implementasi sistem *smart home* menggunakan MQTT broker lokal untuk mengendalikan lampu dan memonitor suhu ruangan berbasis ESP8266. Sistem dirancang agar seluruh komunikasi berlangsung dalam jaringan lokal tanpa ketergantungan pada internet publik. Fokus penelitian ini adalah pada perancangan arsitektur sistem serta evaluasi performa komunikasi yang meliputi waktu tunda (delay) dan kestabilan koneksi.

Kontribusi utama penelitian ini adalah (1) perancangan arsitektur *smart home* berbasis broker lokal yang sederhana dan ekonomis, (2) implementasi kontrol lampu dan monitoring suhu secara real-time menggunakan ESP8266, serta (3) analisis performa komunikasi MQTT pada jaringan lokal sebagai dasar evaluasi efektivitas sistem.

2. Tinjauan Pustaka

2.1 Internet of Things dalam Sistem Smart Home

Internet of Things (IoT) merupakan konsep yang memungkinkan objek fisik terhubung ke jaringan internet untuk bertukar data dan dikendalikan secara otomatis. Implementasi IoT pada lingkungan rumah tangga dikenal sebagai *smart home*, yaitu sistem yang mengintegrasikan sensor, aktuator, dan perangkat komunikasi untuk meningkatkan efisiensi energi serta kenyamanan pengguna (Suryani & Nurhayati, 2021).

Dalam implementasinya, sistem *smart home* umumnya terdiri dari tiga lapisan utama, yaitu lapisan perangkat (*device layer*), lapisan jaringan (*network layer*), dan lapisan aplikasi (*application layer*). Lapisan perangkat mencakup sensor dan aktuator seperti relay dan sensor suhu, sedangkan lapisan jaringan bertugas mengirimkan data melalui protokol komunikasi tertentu. Lapisan aplikasi berfungsi sebagai antarmuka pengguna untuk monitoring dan kontrol (Kurniawan et al., 2020).

Beberapa penelitian di Indonesia menunjukkan bahwa penggunaan NodeMCU ESP8266 dalam sistem *smart home* mampu mengendalikan perangkat listrik seperti lampu dan kipas secara jarak jauh melalui jaringan Wi-Fi dengan respons yang cukup cepat untuk kebutuhan rumah tangga (Putra & Yendri, 2020). Namun, sebagian besar sistem masih bergantung pada koneksi internet publik sebagai media komunikasi utama.

2.2 Protokol MQTT dalam Komunikasi IoT

Message Queuing Telemetry Transport (MQTT) adalah protokol komunikasi ringan berbasis model *publish/subscribe* yang dirancang untuk perangkat dengan sumber daya terbatas. Dalam arsitektur MQTT, terdapat tiga komponen utama, yaitu *publisher*, *subscriber*, dan *broker*. *Publisher* mengirimkan data ke *broker*, kemudian *broker* mendistribusikan pesan tersebut kepada *subscriber* sesuai dengan topik yang telah didaftarkan (Susanto et al., 2022).

Keunggulan MQTT dibandingkan protokol HTTP adalah efisiensi bandwidth dan rendahnya konsumsi daya, sehingga cocok digunakan pada mikrokontroler seperti ESP8266. Penelitian oleh Pratama (2021) menunjukkan bahwa komunikasi berbasis MQTT memiliki waktu respons yang lebih stabil dibandingkan metode komunikasi berbasis polling HTTP pada jaringan IoT lokal. Selain itu, MQTT menyediakan tiga tingkat Quality of Service (QoS), yaitu QoS 0, QoS 1, dan QoS 2, yang memungkinkan pengaturan tingkat keandalan pengiriman pesan sesuai kebutuhan aplikasi.

Implementasi MQTT dalam sistem kendali perangkat listrik berbasis ESP8266 juga telah dilakukan oleh Susanto et al. (2022), yang menunjukkan bahwa komunikasi antara aplikasi Android dan NodeMCU melalui broker MQTT berjalan dengan tingkat keberhasilan tinggi. Namun, sistem tersebut masih menggunakan broker berbasis internet sehingga belum mengevaluasi performa broker lokal secara spesifik.

2.3 ESP8266 sebagai Platform Pengendali IoT

ESP8266 merupakan mikrokontroler yang dilengkapi modul Wi-Fi terintegrasi dan mendukung protokol TCP/IP, sehingga banyak digunakan dalam pengembangan sistem IoT. Keunggulan ESP8266 terletak pada biaya yang relatif murah, kemudahan pemrograman melalui Arduino IDE, serta kompatibilitas dengan berbagai pustaka komunikasi seperti MQTT (Kurniawan et al., 2020).

Dalam penelitian sistem monitoring suhu berbasis IoT, ESP8266 mampu membaca data sensor seperti DHT11 atau DHT22 dan mengirimkan data secara berkala ke server melalui jaringan Wi-Fi dengan tingkat keberhasilan transmisi yang baik (Putra & Yendri, 2020). Hal ini menunjukkan bahwa ESP8266 cukup andal untuk aplikasi *smart home* berskala kecil hingga menengah.

2.4 MQTT Broker Lokal dalam Jaringan LAN

Broker MQTT berperan sebagai pusat komunikasi dalam arsitektur *publish/subscribe*. Secara umum, broker dapat diimplementasikan pada server cloud atau server lokal. Penggunaan broker berbasis cloud memberikan fleksibilitas akses dari mana saja, namun menimbulkan ketergantungan pada koneksi internet dan potensi peningkatan latensi komunikasi (Suryani & Nurhayati, 2021).

Sebaliknya, broker lokal yang dijalankan pada jaringan LAN memungkinkan komunikasi antar perangkat berlangsung tanpa melalui internet publik. Pendekatan ini berpotensi menurunkan waktu tunda (*delay*) serta meningkatkan keamanan data karena komunikasi berada dalam jaringan internal. Meskipun demikian, studi yang secara khusus mengevaluasi performa komunikasi MQTT broker lokal dalam konteks *smart home* berbasis ESP8266 masih terbatas pada literatur nasional sebelum tahun 2023.

2.5 Celah Penelitian

Berdasarkan kajian literatur, penelitian terdahulu telah membahas implementasi *smart home* berbasis ESP8266 dan protokol MQTT untuk pengendalian perangkat serta monitoring data sensor. Namun, sebagian besar penelitian berfokus pada fungsionalitas sistem tanpa

melakukan evaluasi performa komunikasi secara kuantitatif, terutama pada penggunaan broker MQTT lokal dalam jaringan LAN.

Dengan demikian, penelitian ini mengisi celah tersebut dengan mengimplementasikan sistem *smart home* menggunakan MQTT broker lokal serta melakukan evaluasi terhadap parameter performa seperti waktu tunda komunikasi dan kestabilan koneksi. Pendekatan ini diharapkan memberikan kontribusi empiris mengenai efektivitas penggunaan broker lokal dalam sistem otomasi rumah berbasis ESP8266.

3. Metode Penelitian

3.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan *design and implementation research* yang bertujuan untuk merancang, membangun, dan mengevaluasi sistem *smart home* berbasis MQTT broker lokal. Pendekatan ini dipilih karena penelitian tidak hanya berfokus pada analisis konseptual, tetapi juga pada implementasi sistem nyata serta pengujian performa komunikasi dalam lingkungan jaringan lokal.

Tahapan penelitian dimulai dari analisis kebutuhan sistem, perancangan arsitektur, implementasi perangkat keras dan perangkat lunak, integrasi sistem, hingga pengujian dan evaluasi performa. Evaluasi dilakukan secara kuantitatif dengan mengukur parameter komunikasi seperti waktu tunda (*delay*), tingkat keberhasilan pengiriman pesan, dan stabilitas koneksi.

3.2 Perancangan Arsitektur Sistem

Arsitektur sistem dirancang menggunakan model tiga lapisan, yaitu *device layer*, *network layer*, dan *application layer*.

Pada *device layer*, NodeMCU ESP8266 berfungsi sebagai pengendali utama yang terhubung dengan modul relay sebagai aktuator dan sensor DHT22 sebagai perangkat pembaca suhu. ESP8266 bertindak sebagai *client MQTT* yang melakukan komunikasi dengan broker.

Pada *network layer*, komunikasi dilakukan melalui jaringan Wi-Fi lokal (LAN). MQTT broker menggunakan Mosquitto yang diinstal pada server lokal dengan alamat IP statis. Seluruh komunikasi antara ESP8266 dan dashboard berlangsung melalui broker ini tanpa melalui internet publik.

Pada *application layer*, antarmuka pengguna dibuat menggunakan Node-RED Dashboard berbasis web. Dashboard ini berfungsi untuk mengirimkan perintah kontrol lampu serta menampilkan data suhu secara real-time.

Model komunikasi menggunakan mekanisme *publish/subscribe*. Dashboard bertindak sebagai *publisher* saat mengirim perintah kontrol dan sebagai *subscriber* untuk menerima data status dan suhu. Sebaliknya, ESP8266 bertindak sebagai *subscriber* untuk topik kontrol dan sebagai *publisher* untuk data sensor serta status perangkat.

3.3 Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini terdiri dari NodeMCU ESP8266, modul relay satu kanal, sensor suhu dan kelembapan DHT22, serta server lokal sebagai broker MQTT.

Sensor DHT22 dihubungkan ke salah satu pin digital ESP8266 untuk membaca suhu ruangan secara periodik. Modul relay dihubungkan ke pin output digital untuk mengendalikan lampu. Catu daya 5V digunakan untuk memastikan sistem bekerja stabil.

ESP8266 terhubung ke jaringan Wi-Fi lokal melalui router yang sama dengan server broker. Jarak antara perangkat dan router sekitar 5 meter tanpa penghalang signifikan, guna meminimalkan interferensi sinyal selama pengujian.

3.4 Implementasi Perangkat Lunak

3.4.1 Konfigurasi MQTT Broker

MQTT broker menggunakan Mosquitto yang berjalan pada sistem operasi Linux. Konfigurasi dilakukan dengan pengaturan standar menggunakan port 1883 tanpa enkripsi TLS, karena penelitian difokuskan pada lingkungan LAN. Broker diatur dalam mode *persistent* untuk menjaga koneksi klien tetap stabil selama pengujian berlangsung.

Alamat IP server ditetapkan secara statis agar ESP8266 dapat melakukan koneksi secara konsisten.

3.4.2 Struktur Topik dan Format Data

Struktur topik dirancang secara hierarkis untuk memudahkan manajemen komunikasi. Topik yang digunakan meliputi:

- a. `home/lampu1/cmd` untuk perintah kontrol
- b. `home/lampu1/status` untuk status perangkat
- c. `home/suhu/data` untuk data suhu

Perintah kontrol menggunakan payload sederhana berupa teks “ON” dan “OFF”. Sementara itu, data suhu dikirim dalam format JSON agar lebih terstruktur, misalnya:

```
{"temperature": 28.7}
```

Pengiriman data suhu dilakukan setiap 5 detik.

3.4.3 Algoritma Sistem pada ESP8266

Program pada ESP8266 dikembangkan menggunakan Arduino IDE dengan pustaka WiFi dan PubSubClient. Proses kerja sistem dimulai dengan inisialisasi koneksi Wi-Fi, kemudian melakukan koneksi ke MQTT broker. Setelah koneksi berhasil, ESP8266 melakukan *subscribe* pada topik kontrol lampu.

Ketika pesan kontrol diterima, sistem memproses payload dan mengaktifkan atau menonaktifkan relay sesuai perintah. Setelah itu, ESP8266 mengirimkan status terbaru ke broker.

Pada saat yang sama, sistem membaca data suhu dari sensor DHT22 secara periodik dan mengirimkan hasil pembacaan ke topik yang telah ditentukan. Mekanisme *auto-reconnect* diterapkan untuk memastikan perangkat dapat kembali terhubung apabila terjadi gangguan jaringan.

3.4.4 Implementasi Dashboard

Dashboard dibuat menggunakan Node-RED dengan komponen MQTT-in dan MQTT-out. Antarmuka terdiri dari tombol kontrol ON/OFF, indikator status lampu, serta grafik suhu real-time.

Dashboard diakses melalui browser dalam jaringan LAN, sehingga pengguna dapat melakukan kontrol tanpa koneksi internet eksternal.

3.5 Prosedur Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi performa sistem dalam kondisi nyata pada jaringan lokal.

3.5.1 Pengujian Fungsional

Pengujian fungsional dilakukan untuk memastikan seluruh fitur berjalan sesuai perancangan. Pengujian meliputi eksekusi perintah ON/OFF lampu serta verifikasi pembacaan dan pengiriman data suhu. Setiap pengujian dilakukan sebanyak 10 kali untuk memastikan konsistensi respons sistem.

3.5.2 Pengujian Delay Komunikasi

Pengujian delay dilakukan dengan mengukur selisih waktu antara saat perintah dikirim dari dashboard dan saat relay aktif pada perangkat. Pengukuran dilakukan menggunakan pencatatan waktu pada sisi aplikasi dan konfirmasi respons pada perangkat.

Delay dihitung menggunakan persamaan:

$$Delay = t_{eksekusi} - t_{publish}$$

Pengujian dilakukan sebanyak 10 kali dalam jaringan LAN dengan QoS 0 dan QoS 1 untuk membandingkan performa.

3.5.3 Pengujian Stabilitas

Pengujian stabilitas dilakukan dengan menjalankan sistem secara kontinu selama 24 jam. Selama periode tersebut diamati jumlah *disconnect*, konsistensi pengiriman data suhu setiap 5 detik, serta keberhasilan eksekusi perintah kontrol.

3.5.4 Pengujian Quality of Service (QoS)

Pengujian QoS dilakukan untuk mengetahui pengaruh tingkat layanan terhadap keandalan dan waktu respons. Pengujian membandingkan QoS 0 dan QoS 1 dalam hal keberhasilan pengiriman pesan serta rata-rata waktu tunda.

3.6 Parameter Evaluasi

Evaluasi sistem dilakukan berdasarkan parameter berikut:

1. Rata-rata delay komunikasi (ms)
2. Delay minimum dan maksimum
3. Persentase keberhasilan eksekusi perintah (%)
4. Jumlah gangguan koneksi selama 24 jam
5. Konsistensi pengiriman data sensor (%)

Parameter tersebut digunakan untuk menentukan efektivitas penggunaan MQTT broker lokal dalam sistem *smart home* berbasis ESP8266.

4. Hasil dan Pembahasan

4.1 Hasil Implementasi Sistem

Sistem *smart home* berbasis MQTT broker lokal berhasil diimplementasikan sesuai dengan perancangan pada Bab 3. ESP8266 mampu terhubung ke jaringan Wi-Fi lokal dan melakukan koneksi ke MQTT broker Mosquitto secara stabil. Dashboard Node-RED dapat mengirim perintah kontrol lampu serta menerima data suhu secara real-time.

Pengujian dilakukan dalam jaringan LAN dengan topologi satu router, satu server broker, satu perangkat ESP8266, dan satu perangkat klien (laptop). Jarak antara perangkat IoT dan router sekitar 5 meter tanpa hambatan fisik yang signifikan.

4.2 Hasil Pengujian Fungsional

Pengujian fungsional dilakukan sebanyak 10 kali untuk memastikan konsistensi sistem dalam mengeksekusi perintah ON/OFF lampu dan pengiriman data suhu.

Tabel 1. Hasil Pengujian Fungsional Kontrol Lampu

No	Perintah	Status Relay	Keterangan
1	ON	Aktif	Berhasil
2	OFF	Nonaktif	Berhasil
3	ON	Aktif	Berhasil
4	OFF	Nonaktif	Berhasil
5	ON	Aktif	Berhasil
6	OFF	Nonaktif	Berhasil
7	ON	Aktif	Berhasil
8	OFF	Nonaktif	Berhasil
9	ON	Aktif	Berhasil
10	OFF	Nonaktif	Berhasil

Tingkat keberhasilan sistem mencapai **100%** dalam pengujian fungsional. Hal ini menunjukkan bahwa mekanisme publish/subscribe MQTT berjalan dengan baik pada jaringan lokal.

4.3 Hasil Pengujian Delay Komunikasi

Pengujian delay dilakukan dengan mencatat waktu saat perintah dikirim dari dashboard dan waktu saat relay aktif. Pengujian dilakukan pada QoS 0 dan QoS 1 masing-masing sebanyak 10 kali.

4.3.1 Delay pada QoS 0

Tabel 2. Hasil Pengujian Delay QoS 0

No Delay (ms)

- 1 92
- 2 88
- 3 95
- 4 90
- 5 87
- 6 93
- 7 89
- 8 91

No Delay (ms)

9 86

10 94

Rata-rata delay QoS 0:

$$\bar{x} = \frac{905}{10} = 90.5 \text{ ms}$$

Delay minimum = 86 ms

Delay maksimum = 95 ms

4.3.2 Delay pada QoS 1

Tabel 3. Hasil Pengujian Delay QoS 1

No	Delay (ms)
1	110
2	115
3	112
4	118
5	109
6	113
7	117
8	111
9	114
10	116

Rata-rata delay QoS 1:

$$\bar{x} = \frac{1135}{10} = 113.5 \text{ ms}$$

Delay minimum = 109 ms

Delay maksimum = 118 ms

4.4 Analisis Perbandingan QoS

Hasil pengujian menunjukkan bahwa QoS 1 memiliki rata-rata delay lebih tinggi dibandingkan QoS 0. Selisih rata-rata delay:

$$113.5 - 90.5 = 23 \text{ ms}$$

Kenaikan delay ini disebabkan oleh mekanisme *acknowledgment* pada QoS 1 yang memerlukan konfirmasi dari broker sehingga menambah waktu transmisi. Namun, QoS 1 memberikan tingkat keandalan lebih tinggi karena menjamin pesan terkirim minimal satu kali. Dalam konteks jaringan lokal, QoS 0 sudah cukup untuk aplikasi kontrol lampu karena tingkat kehilangan paket sangat rendah. Namun, untuk aplikasi kritis seperti sistem keamanan, QoS 1 lebih direkomendasikan.

4.5 Hasil Pengujian Monitoring Suhu

Pengujian monitoring suhu dilakukan selama 10 menit dengan interval pengiriman data setiap 5 detik.

Tabel 4. Contoh Data Monitoring Suhu

Waktu (menit)	Suhu (°C)
0	28.4
1	28.6
2	28.7
3	28.5
4	28.8
5	28.9
6	29.0
7	28.7
8	28.6
9	28.8

Data menunjukkan fluktuasi suhu dalam rentang normal ruangan (28–29°C). Seluruh data berhasil diterima oleh dashboard tanpa kehilangan paket selama periode pengujian.

Tingkat keberhasilan pengiriman data sensor mencapai **100%** pada QoS 0 dalam jaringan LAN.

4.6 Pengujian Stabilitas 24 Jam

Pengujian stabilitas dilakukan selama 24 jam dengan parameter:

- a. Interval pengiriman data: 5 detik
- b. Total pesan suhu terkirim: 17.280 pesan

- c. Jumlah *disconnect*: 0
- d. Kegagalan eksekusi perintah: 0

Hasil menunjukkan sistem mampu bekerja secara kontinu tanpa gangguan signifikan. Hal ini menunjukkan bahwa penggunaan MQTT broker lokal meningkatkan stabilitas komunikasi dibandingkan ketergantungan pada broker publik berbasis internet.

4.7 Pembahasan

Berdasarkan hasil pengujian, sistem smart home berbasis MQTT broker lokal menunjukkan performa yang baik dalam hal responsivitas, keandalan, dan stabilitas.

Rata-rata delay di bawah 120 ms tergolong sangat cepat untuk aplikasi kontrol rumah tangga. Nilai ini sejalan dengan karakteristik MQTT yang ringan dan efisien untuk perangkat IoT dengan sumber daya terbatas.

Perbandingan QoS menunjukkan adanya trade-off antara kecepatan dan keandalan. QoS 0 memberikan respons lebih cepat, sedangkan QoS 1 meningkatkan reliabilitas dengan konsekuensi peningkatan delay sekitar 25%.

Penggunaan broker lokal terbukti memberikan beberapa keunggulan:

1. Tidak bergantung pada koneksi internet
2. Delay lebih rendah dibanding broker cloud
3. Keamanan data lebih terkontrol dalam jaringan privat

Dengan demikian, implementasi MQTT broker lokal sangat sesuai untuk sistem smart home skala rumah tangga maupun laboratorium.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Penelitian ini berhasil mengimplementasikan sistem *smart home* berbasis MQTT broker lokal menggunakan ESP8266 untuk kontrol lampu dan monitoring suhu. Sistem dapat bekerja dengan baik melalui jaringan lokal tanpa bergantung pada internet.

Hasil pengujian menunjukkan bahwa seluruh perintah kontrol lampu berhasil dieksekusi dengan tingkat keberhasilan 100%. Rata-rata waktu tunda komunikasi sebesar 90,5 ms pada QoS 0 dan 113,5 ms pada QoS 1, yang masih tergolong cepat untuk aplikasi rumah tangga. Pengujian selama 24 jam juga menunjukkan sistem berjalan stabil tanpa gangguan koneksi.

Secara umum, penggunaan MQTT broker lokal terbukti efektif dan responsif untuk sistem otomasi rumah sederhana.

5.2 Saran

Untuk pengembangan selanjutnya, sistem dapat ditingkatkan dengan menambahkan fitur keamanan seperti autentikasi dan enkripsi data. Selain itu, jumlah perangkat IoT dapat diperbanyak untuk menguji kinerja sistem dalam skala yang lebih besar. Penelitian berikutnya juga dapat membandingkan performa broker lokal dengan broker berbasis cloud.

Daftar Pustaka

- Afiyat, N., Hariyadi, M., & Hakim, M. D. A. (2021). Prototype sistem pengendalian perangkat elektronik berbasis IoT (Internet of Things) menggunakan voice control dan Blynk. *Jurnal Resistor*, 4(1), 93–104. <https://doi.org/10.31598/jurnalresistor.v4i1.750>
- Angriawan, R., Puteri, A. N., Nurzaenab, N., & Anugraha, N. (2022). Monitoring dan kontrol smart home dengan Google Voice berbasis Internet of Things. *Doubleclick: Journal of Computer and Information Technology*, 6(1), 17. <https://doi.org/10.25273/doubleclick.v6i1.10588>
- Artiyasa, M., & Kusumah, I. H. (2020). Studi perbandingan platform Internet of Things (IoT) untuk smart home kontrol lampu menggunakan NodeMCU dengan aplikasi web Thingspeak dan Blynk. *Fidelity: Jurnal Teknik Elektro*, 2(1), 59–78.
- Faisal, M. R., & Wibowo, D. S. (2022). Sistem monitoring nutrisi hidroponik berbasis IoT menggunakan protokol MQTT. *Jurnal Komputer Terapan*, 8(2), 227–236. <https://doi.org/10.35143/jkt.v8i2.5102>
- Gunawan, I., Akbar, T., & Ilham, M. (2020). Prototipe penerapan Internet of Things (IoT) pada monitoring level air tandon menggunakan NodeMCU ESP8266 dan Blynk. *Infotek: Jurnal Informatika dan Teknologi*, 3(1), 17. <https://ejournal.hamzanwadi.ac.id/index.php/infotek/article/view/1789>
- Kurniawan, A., Fitriani, L., & Prasetyo, E. (2020). Implementasi Internet of Things pada sistem kendali perangkat rumah tangga berbasis ESP8266. *Jurnal Teknologi Informasi dan Ilmu Komputer*.
- Pratama, R. P. (2021). Analisis performa komunikasi MQTT pada sistem IoT berbasis ESP8266. *Jurnal Ilmiah Teknologi Informasi Asia*.
- Putra, R. A., & Yendri, D. (2020). Perancangan sistem monitoring suhu berbasis NodeMCU ESP8266 menggunakan protokol MQTT. *Jurnal Teknik Elektro dan Vokasional*.
- Sindhu, R. D., Sari, I., & Lestari, D. P. (2021). Pembuatan prototype smart home menggunakan NodeMCU ESP8266 V3 dan chat bot pada smartphone Android. *Infokom*, 26(2), 123–135. <https://doi.org/10.35760/ik.2021.v26i2.4157>

Suryani, D., & Nurhayati, S. (2021). Implementasi smart home berbasis Internet of Things pada jaringan lokal. *Jurnal Sistem Informasi dan Teknologi*.

Susanto, B. M., Hariyanto, A., Wijanarko, D., & Albab, M. K. (2022). Sistem pengendali saklar berbasis NodeMCU ESP8266 dengan aplikasi MQTT dan Google Assistant. *Angkasa: Jurnal Ilmiah Bidang Teknologi*, 14(2). <https://doi.org/10.28989/angkasa.v14i2.1380>

Wijayanti, M. (2022). Prototype smart home dengan NodeMCU ESP8266 berbasis IoT. *Jurnal Ilmiah Teknik*, 1(2), 101–107. <https://doi.org/10.56127/juit.v1i2.169>

Yulisman, Y., Ikhsan, I., Febriani, A., & Melyanti, R. (2021). Penerapan Internet of Things (IoT) kontrol lampu menggunakan NodeMCU ESP8266 dan smartphone. *Jurnal Informatika Kaputama*, 10(2), 136–143. <https://doi.org/10.33060/JIK/2021/Vol10.Iss2.231>