



ANALISIS *QUALITY OF SERVICE* INTEGRASI *MQTT-LORA* DENGAN HAMBATAN UNTUK MONITORING *AUTONOMOUS UNDERWATER VEHICLE*

Sartika Oktavian Puji Prameswari¹, Noorman Rinanto², Zindhu Maulana Ahmad Putra³ Dwi Sasmita Aji Pambudi⁴, Ryan Yudha Adhitya⁵

^{1,2,3,4,5} Politeknik Perkapalan Negeri Surabaya, Jl. Teknik Kimia Keputih, Surabaya, 60111, Indonesia

Oktavianprameswari25@student.ppns.ac.id¹, noorman.rinanto@ppns.ac.id², zindhu@ppns.ac.id³, dwi.sasmita@ppns.ac.id⁴, ryanyudhadhitya@ppns.ac.id⁵

Abstrak

Pertumbuhan teknologi dalam bidang robotika, khususnya robot bawah air, telah menciptakan peluang baru dalam eksplorasi dan aplikasi di berbagai sektor. Robot bawah air telah berevolusi dari bentuk robot yang dikendalikan secara *remote*, menjadi varian yang beroperasi secara mandiri (*Autonomous Underwater Vehicle*). GCS berperan sebagai monitor atau pusat kontrol yang akan memudahkan pengguna untuk melakukan pemantauan di darat, ketika AUV sedang berada di bawah air. Dengan fokus pada implementasi GCS menggunakan protokol MQTT pada penelitian ini, diharapkan dapat memberikan kontribusi pada pengembangan teknologi AUV yang dapat diandalkan dan efisien untuk berbagai tugas dan misi bawah air. Hasil penelitian menunjukkan bahwa integrasi MQTT dan LoRa mampu meningkatkan efisiensi komunikasi dan monitoring AUV secara signifikan dengan nilai rata-rata *delay* pengiriman dan penerimaan data hanya 0,02 detik, dengan nilai perbedaan *delay* antar pengiriman paket data sebesar 0,32 milidetik, dan jumlah banyaknya data yang dipindahkan dengan rentang waktu tertentu sebesar 0,304 Kb/s, serta jumlah paket data yang hilang adalah 0. Berdasarkan pengujian yang telah dilakukan, sensor HWT905, MS5837, dan Ublox M8-N juga menunjukkan tingkat akurasi yang memadai.

Kata kunci: Kendaraan Bawah Air Otonom, Sistem Kontrol Darat, LoRa, MQTT

Abstract:

The growth of technology in the field of robotics, especially underwater robots, has created new opportunities for exploration and application in various sectors. Underwater robots have evolved from robots that are controlled remotely, to variants that operate independently (Autonomous Underwater Vehicle). GCS acts as a monitor or control center which will make it easier for users to monitor on land, when the AUV is underwater. By focusing on the implementation of GCS using the MQTT protocol in this research, it is hoped that it can contribute to the development of reliable and efficient AUV

technology for various underwater tasks and missions. The research results show that the integration of MQTT and LoRa is able to significantly increase the efficiency of AUV communication and monitoring with an average value of delay for sending and receiving data of only 0.02 seconds, with a difference in delay between sending data packets of 0.32 milliseconds, and the amount of data transferred over a certain time span was 0.304 Kb/s, and the number of lost data packets was 0. Based on the tests that have been carried out, the HWT905, MS5837, and Ublox M8-N sensors also show an adequate level of accuracy.

Keywords: *Autonomous Underwater Vehicle, Ground Control System, LoRa, MQTT*

pendahuluan

Pertumbuhan teknologi dalam bidang robotika telah menciptakan peluang baru dalam eksplorasi dan aplikasi di berbagai sektor (Nurul Hidayat et al., 2024). Salah satu sektor yang terkena pengaruh dari berkembangnya teknologi ini yaitu sektor maritim. Luasnya samudera menyebabkan kurangnya pengetahuan yang mendalam tentang eksplorasi laut. Untuk melakukan inspeksi dan survey bawah laut memerlukan teknologi yang mampu mengeksplorasi laut dengan kedalaman tertentu. Pemanfaatan kendaraan tak berawak pada eksplorasi bawah air menjadi krusial. Salah satu kendaraan tak berawak yang dapat dimanfaatkan adalah *Autonomous Underwater Vehicle (AUV)* (Ar-Rasyid, 2022).

Autonomous Underwater Vehicle (AUV) merupakan kendaraan bawah air yang dirancang untuk beroperasi secara mandiri tanpa kendali langsung dari operator atau manusia (Prima et al., 2018). AUV dilengkapi dengan sistem navigasi, sensor, dan pemrograman yang memungkinkan AUV untuk menjalankan tugas-tugas tertentu secara *autonomous*. Dalam penerapannya, peran *Autonomous Underwater Vehicle (AUV)* di bidang eksplorasi maritim sangatlah penting (Qilei ZhangJinying, 2020). Dengan penambahan berbagai fitur pendukung, observasi laut pada kedalaman tertentu di bawah air sangat mungkin dapat dilakukan oleh AUV (Sherif MostafaY, 2021). Dalam rangka mengoptimalkan kinerja robot bawah air, penerapan Ground Control System (GCS) menjadi krusial. Selain itu, dengan memanfaatkan teknologi AUV yang bergerak secara otonom, bersamaan dengan penerapan GCS, tim pencarian (Tim SAR) dapat dengan mudah menemukan titik koordinat korban atau kapal karam beserta titik kedalaman objek tersebut berada.

Ground Control System (GCS) merupakan sistem kontrol yang digunakan untuk mengatur atau memantau dan berinteraksi dengan *Autonomous Underwater Vehicle* dari daratan. GCS berperan sebagai monitor atau pusat kontrol yang akan memudahkan pengguna untuk melakukan pemantauan di darat, ketika AUV sedang berada di dalam air (Ramdani, 2019). Selain itu, GCS juga memudahkan pengguna untuk mengoptimalkan kinerja, mengubah jalur misi, dan merespon secara *real-time* terhadap kondisi AUV.

Dalam penelitian ini, terdapat tantangan dalam pengimplementasian GCS untuk AUV di bawah air yang kompleks. Teknologi LoRa menjadi pertimbangan untuk

mendukung komunikasi jarak jauh dalam pengimplementasi GCS pada AUV. LoRa (Low Range) merupakan teknologi nirkabel jarak jauh yang terbukti efektif dalam hal mentransmisikan data pada rentang jarak yang signifikan (Astutik, 2023). Menurut Bhawiyuga, et al (2019), sistem LoRa telah berhasil mengirim data sensor *node* ke *gateway* menggunakan komunikasi *LoRa* dengan jarak hingga 400 m (Mu'ammad, 2020). Sayangnya, penerapan dalam air beresiko menghadapi tantangan lain seperti, absorpsi sinyal oleh air dan kecepatan transisi data yang terbatas, sehingga dibutuhkan metode komunikasi tambahan yang dapat memastikan kehandalan dan kecepatan pertukaran informasi (Arifin et al., 2023).

Oleh karena itu, metode yang digunakan dalam penelitian ini adalah *Message Queing Telemetry Transport* (MQTT). Pemanfaatan MQTT sebagai protokol komunikasi antara AUV dan GCS menjadi kunci dalam mendukung pertukaran informasi secara *real-time*. Penerapan MQTT memungkinkan AUV untuk mengirimkan data ke GCS dengan cepat dan tingkat akurasi yang tinggi, sehingga operator secara instan dapat memantau perkembangan misi dan segera melakukan intervensi jika diperlukan. Dengan meminimalkan latensi komunikasi, MQTT tidak hanya meningkatkan efisiensi operasional tapi juga mendukung respon cepat terhadap perubahan kondisi di lapangan.

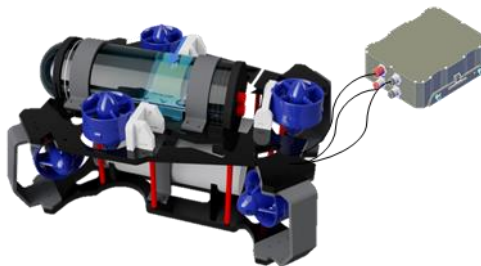
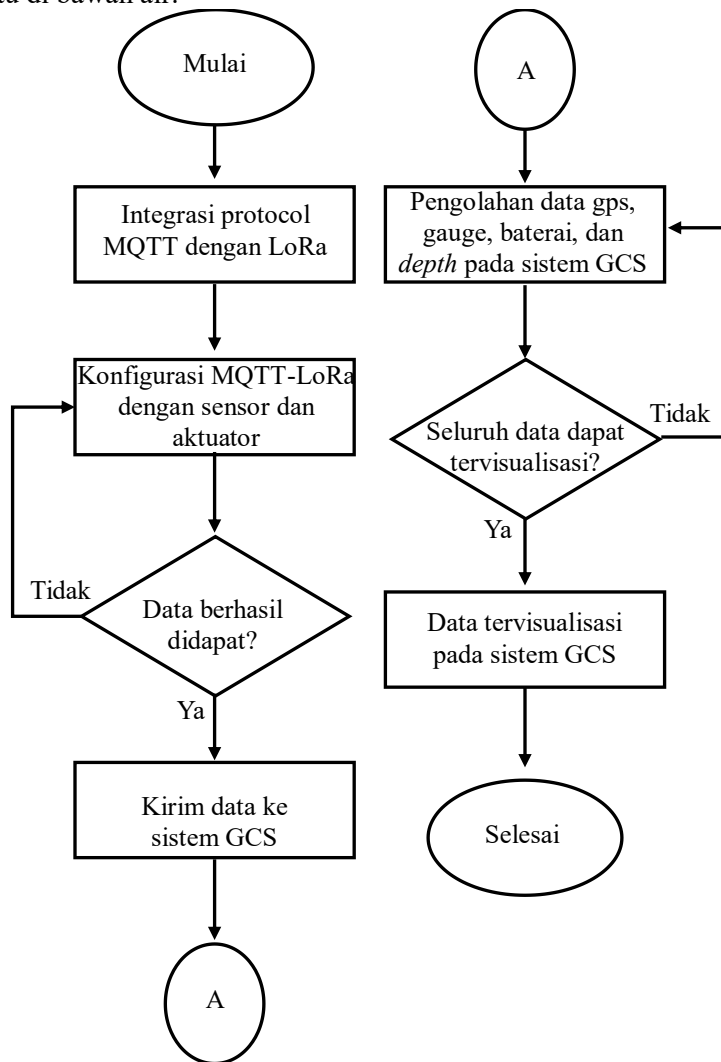
Pemilihan MQTT sebagai protokol didasari oleh kecepatan pengiriman dan penerimaan data, serta rendahnya penggunaan bandwidth yang dimiliki MQTT dibanding dengan HTTP menjadi keunggulan utama untuk keperluan monitoring jarak jauh terlebih dengan kondisi di bawah air yang sangat mungkin memiliki bandwidth yang terbatas. Selain itu, latensi yang rendah dalam pertukaran pesan sangat penting untuk keperluan monitoring secara *real-time*. Model *Publish* dan *Subscribe* memudahkan monitoring dengan sistem kerja dimana *Subscriber* selalu menerima pembaruan data pada satu topik khusus yang menghubungkan *Publisher* dan *Subscriber* tanpa meminta (*pulling data*) seperti HTTP. Dengan ini, pemilihan MQTT sebagai protokol dalam penelitian ini merupakan keputusan terbaik yang diambil oleh penulis.

Integrasi MQTT dengan implementasi LoRa pada penelitian ini memungkinkan eksplorasi dan operasi yang lebih andal dan responsif di dalam air. Oleh karena itu, penulis memutuskan untuk melakukan penelitian yang membahas tentang implementasi Ground Control System pada Autonomous Underwater Vehicle untuk misi pencarian dan penyelamatan dengan menggunakan MQTT sebagai protokol dan LoRa sebagai komunikasi antar mikrokontroller untuk pengiriman data.

Metode Penelitian

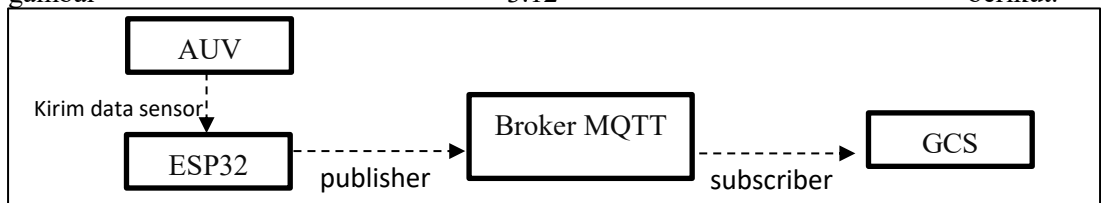
Pada penelitian ini, sistem GCS dirancang dengan menggunakan protokol MQTT dengan komunikasi LoRa. Sistem GCS akan dirancang dengan mengintegrasikan MQTT dengan LoRa untuk pertukaran data yang kemudian akan ditampilkan pada GCS sebagai sistem kontrol dan monitoring pada AUV. Keterbaruan yang ada pada tugas akhir ini adalah penggunaan MQTT sebagai protokol dengan komunikasi LoRa yang akan meningkatkan efektivitas dan efisiensi adanya GCS untuk kegiatan *controlling* dan *monitoring* AUV. Dalam penelitian, peneliti mencoba mengintegrasikan MQTT dengan LoRa sebagai protokol komunikasi yang akan menghubungkan antara GCS dan AUV, sehingga operator dapat mengetahui kondisi AUV ketika berada di bawah air secara

real-time dan dapat memberikan manuver dengan cepat terhadap AUV ketika terjadi sesuatu di bawah air.



Gambar 1. Desain Prototipe AUV Floating Balloon

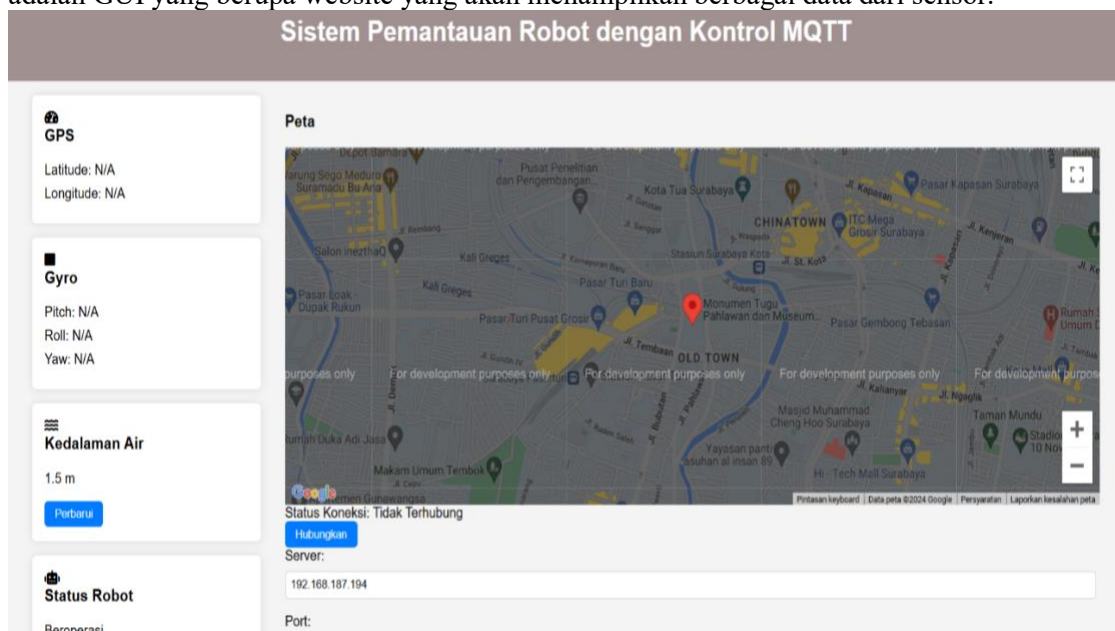
Perencanaan desain skema MQTT akan membahas alur pengiriman data dari sensor hingga tervisualisasi pada GUI. Alur pengiriman data MQTT akan dibahas pada gambar 3.12 berikut.



Gambar 2 Skema MQTT

(sumber: dokumentasi pribadi, 2023)

pada gambar 3 tersebut terlihat bahwa skema MQTT melibatkan peran *publisher*, *broker*, dan *subscriber*. Pada penelitian ini, esp32 memiliki peran sebagai *publisher* yang akan menerima data dari sensor yang sudah diolah oleh STM dan data dari GPS modul yang kemudian akan diteruskan ke broker *Mosquitto* dengan topik tertentu yang telah *unsubscribe* oleh *subscriber*. Peran *subscriber* dalam penelitian ini adalah GUI yang berupa website yang akan menampilkan berbagai data dari sensor.



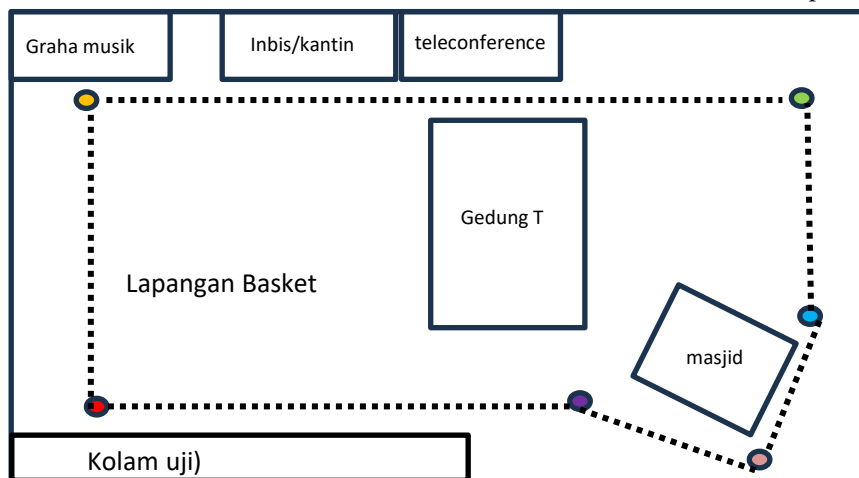
Gambar 3 Desain Tampilan GCS

Pada gambar 4 merupakan rancangan tampilan GCS yang akan dibuat pada sistem ini. Data sensor yang akan ditampilkan pada website yakni posisi robot dari sensor gauge, status kedalaman air dari sensor depth, posisi robot dari modul gps, status baterai robot dari sensor arus dan tegangan, serta status power robot (Saputra, 2022).

Hasil dan Pembahasan

Integrasi MQTT dan LoRa

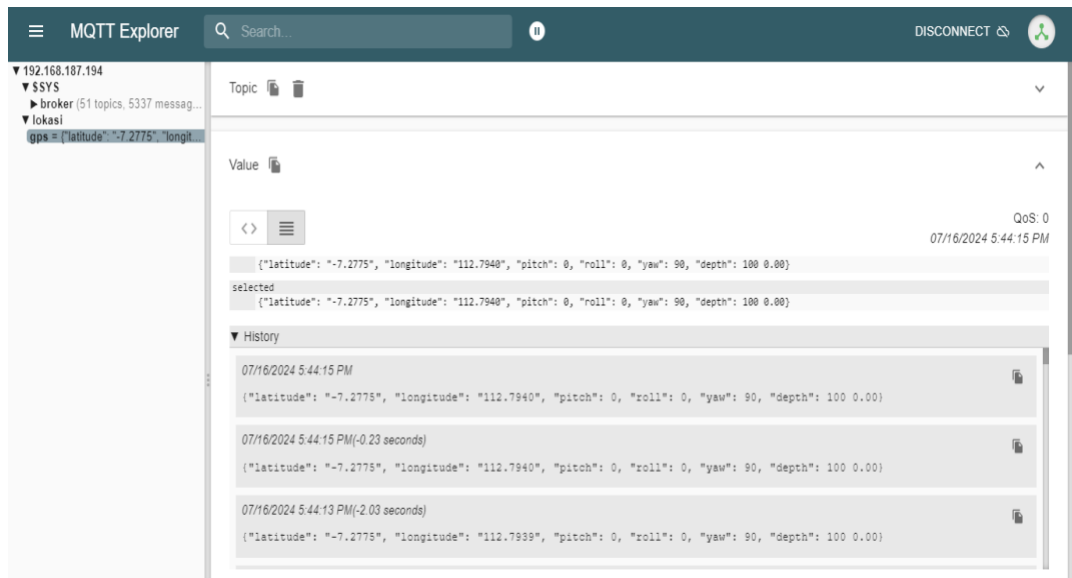
Pengujian pada integrasi MQTT dengan LoRa dilakukan dengan skenario yang sama saat pengujian LoRa. Pengujian integrasi ini dilakukan dengan komunikasi yang dilakukan oleh LoRa pada *floating box* yang mengirimkan data-data sensor dari AUV ke LoRa pada *box* darat. Kemudian, data tersebut akan dikirimkan ke server mqtt.



Gambar 4. Denah Pengujian Integrasi MQTT-LoRa



Gambar 5. Maps Pengujian Integrasi MQTT-LoRa



Gambar 6. MQTT Explorer

Data-data yang telah diterima oleh LoRa darat, akan diterima oleh MQTT server dan dapat dilihat melalui MQTT Explorer dengan menyambung pada topik yang sesuai. Kemudian, data yang diterima dapat dilihat di kolom *value* beserta *delay* pengiriman pesan. Seluruh data pengujian integrasi ini dapat dilihat pada tabel berikut (Purnama, 2018).

Tabel 1. Pengujian Integrasi MQTT-LoRa

Pengujian Integrasi MQTT-LoRa					
No.	Jarak (meter)	Rata-rata Panjang Karakter Dikirim	Rata-rata Panjang Karakter Diterima	Delay	
1	55	75	75	0,2	
2	75	75	75	0,22	
3	90	74	74	0,16	
4	95	74	74	0,83	
5	110	74	74	0,54	
Rata-rata <i>delay</i>				0,39	

Berdasarkan tabel pengujian integrasi LoRa dengan MQTT diatas, pengujian dilakukan pada berbagai jarak mulai dari 55 cm hingga 110 cm. Pada setiap jarak, rata-rata panjang karakter data yang dikirimkan adalah sekitar 75 karakter, dan hasil

pengujian menunjukkan rata-rata panjang karakter yang diterima juga mendekati 75 karakter, dengan sedikit variasi.

Konsistensi data yang dikirim dan diterima terlihat sangat baik. Pada jarak 55 cm hingga 110 cm, rata-rata panjang karakter yang dikirim adalah antara 74 hingga 75 karakter. Demikian juga, rata-rata panjang karakter yang diterima juga tetap konsisten antara 74 hingga 75 karakter. Hal ini menunjukkan bahwa sistem LoRa dan MQTT berhasil menjaga integritas data yang dikirimkan pada berbagai jarak tanpa adanya kehilangan data yang signifikan (Bachri, 2022).

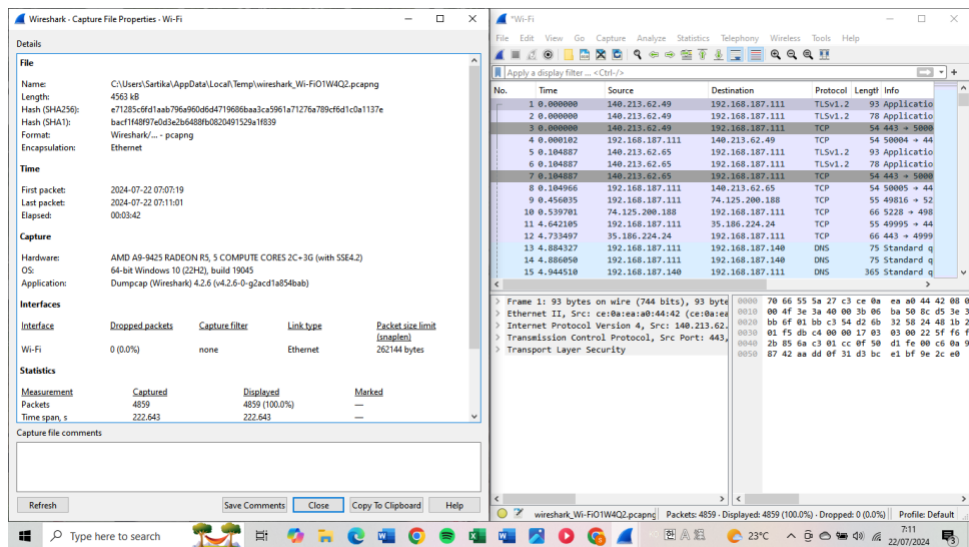
Namun, analisis *delay* pengiriman data menunjukkan peningkatan seiring dengan bertambahnya jarak. Pada jarak 55 m, *delay* tercatat sebesar 0,2 detik, sedangkan pada jarak 95 m, *delay* meningkat hingga 0,83 detik. Sedangkan pada jarak 110 meter, *delay* menurun pada angka 0,54. Rata-rata *delay* untuk semua jarak yang diuji adalah 0,39 detik. Peningkatan *delay* ini dapat disebabkan oleh beberapa faktor, seperti penurunan kekuatan sinyal, interferensi lingkungan, dan waktu yang dibutuhkan untuk pengiriman ulang paket data yang hilang.

Efisiensi sistem LoRa-MQTT tetap menunjukkan kinerja yang baik dalam menjaga panjang karakter data yang dikirim dan diterima, meskipun terdapat peningkatan *delay* seiring dengan bertambahnya jarak. Hal ini penting untuk memastikan bahwa data yang dikirimkan dari sensor dapat diterima secara utuh oleh sistem penerima.

Kesimpulannya, pengujian integrasi LoRa dengan MQTT menunjukkan bahwa sistem ini mampu menjaga integritas data yang dikirimkan pada berbagai jarak, dengan rata-rata panjang karakter data yang diterima mendekati panjang karakter data yang dikirim. Meskipun terdapat peningkatan *delay* pengiriman data seiring dengan bertambahnya jarak, sistem LoRa-MQTT tetap dapat dianggap efisien untuk kinerja sistem.

Pengujian QoS pada Website

Pengujian *quality of service* pada website ini dilakukan dengan menggunakan *software wireshark* untuk menghitung *packet loss* yang terjadi pada website. Pada tahap ini, QoS yang terdiri dari *throughput*, *delay*, *jitter*, dan *packet loss* akan dianalisis dari website yang telah dibuat. Pengujian ini akan menunjukkan bagaimana efisiensi dan performa website MQTT dalam menerima data. Data yang akan dianalisis merupakan data yang dikirim dari LoRa pada box darat (Danial et al., 2024). Gambar berikut adalah tampilan aplikasi *wireshark* saat menganalisis QoS website.



Gambar 7. Pengambilan data Wireshark.

Berdasarkan pengambilan data selama 20 menit, total data yang berhasil terkumpul yaitu 3630 data. Dari 3630 data tersebut, diambil 10 paket data sebagai sampel yang tertera di tabel berikut

Tabel 2. Sampel Pengujian QoS

No.	Jarak (m)	Time(ms)	Protocol	Length	Info
1	55	36.7437	MQTT	72	Publish Message [sensor/pitch]
2	55	88.353	MQTT	162	Publish Message [sensor/roll]
3	75	123.289	MQTT	180	Publish Message [sensor/pitch]
4	75	168.9155	MQTT	180	Publish Message [sensor/pitch]
5	90	210.3028	MQTT	88	Publish Message [sensor/pitch]
6	90	251.7076	MQTT	146	Publish Message [sensor/yaw]
7	95	282.5501	MQTT	180	Publish Message [sensor/pitch]
8	95	342.9034	MQTT	180	Publish Message [sensor/pitch]
9	110	454.5611	MQTT	180	Publish Message [sensor/pitch]
10	110	501.1761	MQTT	72	Publish Message [sensor/pitch]

Pada pengujian ini, nilai *throughput*, *delay*, *jitter*, dan *packet loss* akan dihitung berdasarkan parameter-parameter yang telah ditentukan sebelumnya. Dalam melakukan pengukuran ini, digunakan Wireshark, sebuah alat analisis jaringan yang dapat menangkap dan memeriksa paket data yang dikirim dan diterima dalam jaringan. Wireshark menjadi perangkat untuk menangkap lalu lintas jaringan secara detail, mencatat setiap paket data yang melewati jaringan.

Penggunaan wireshark juga membantu menunjukkan performa dari QoS. Untuk pengujian *delay* dan *jitter*, dapat dilihat pada tabel di bawah ini.

Tabel 3. Sampel data *delay* dan *jitter Quality of Service*

no.	Time(ms)	<i>Delay</i> (s)	<i>Jitter</i> (s)
1	36.7437	0.316093	-0.073638
2	88.353	0.34936	0.149297
3	123.289	0.456265	-0.042567
4	168.9155	0.413873	0.105798
5	210.3028	0.414048	-0.400731
6	251.7076	0.308425	0.782064
7	282.5501	0.603533	-0.836529
8	342.9034	0.116577	0.471368
9	454.5611	0.46615	-0.002649
10	501.1761	0.344355	-0.327505
total		3.788679	-0.175092
rata-rata		0.3788679	-0.0175092

Tabel 3. menampilkan sampel pengujian data QoS yang diambil secara acak. Data ini mencerminkan variasi dalam waktu, *delay*, dan *jitter* yang dialami oleh paket data selama transmisi. Analisis ini dibutuhkan untuk memahami kinerja sistem komunikasi dan memastikan bahwa sistem dapat beroperasi dengan efisien dan andal dalam situasi nyata (Tarigan et al., 2023).

Berdasarkan Tabel 4. tersebut dapat dilihat bahwa total rata-rata *delay* adalah 0,3788679 detik dan *jitter* sebesar -0,0175092 detik. Hasil ini menunjukkan bahwa *delay* dan *jitter* yang diukur berada dalam kisaran yang dapat diterima, dan menunjukkan bahwa ada stabilitas yang cukup tinggi dalam transmisi data. Berdasarkan persamaan 5. dapat dihitung nilai *delay* pada keseluruhan sistem sebagai berikut:

$$Delay = \frac{Time\ span}{Total\ Paket\ dikirim}$$

$$Delay = \frac{1208}{4577} = 1207,83\ s = 20,13\ ms$$

Delay adalah waktu yang dibutuhkan oleh paket data untuk berpindah dari pengirim ke penerima. *Delay* yang diukur dalam penelitian ini menunjukkan waktu rata-rata dari beberapa paket data yang dikirimkan melalui protokol MQTT. Dalam penelitian ini, *delay* dihitung menggunakan data dari wireshar, dengan hasil *delay* sebesar 20,13 ms, yang tergolong dalam kategori sangat bagus. *Delay* dapat dipengaruhi oleh beberapa faktor, termasuk jarak antara pengirim dan penerima, kecepatan transmisi, dan kualitas jaringan.

Berdasarkan persamaan 4.3, nilai *jitter* dapat dihitung sebagai berikut :

$$Jitter = \frac{total\ variasi\ delay}{total\ paket\ yang\ diterima}$$

$$Jitter = \frac{1207,83}{4577} = 0,32 \text{ ms}$$

Jitter adalah variasi *delay* antar paket data. *Jitter* dihitung sebagai rata-rata dari variasi *delay* antar paket (Palaha et al., 2021). Dalam penelitian ini, *jitter* dihitung dengan mengukur variasi *delay* dari setiap paket data. Hasil perhitungan *jitter* sebesar 0,32ms, yang tergolong dalam kategori sangat bagus dengan index 4. Adanya *jitter* yang sangat rendah ini dapat meningkatkan efisiensi dan konsistensi kualitas komunikasi *real-time* antara GCS dan AUV.

Berdasarkan persamaan 6, dapat diketahui nilai *throughput* pada sistem ini dengan perhitungan sebagai berikut:

$$\begin{aligned} \text{Throughput} &= \frac{\text{Paket data diterima}}{\text{Lama pengamatan}} \\ \text{Throughput} &= \frac{4577}{1208,29} = 0,038 \text{ byte/s} \\ \text{Throughput} &= 0,038 \times 8 = 0,304 \text{ Kbyte/s} \\ \text{Throughput} &= 30\% \end{aligned}$$

Throughput adalah jumlah data yang berhasil dikirimkan dan diterima dalam satuan waktu tertentu. *Throughput yang tinggi menunjukkan bahwa sistem mampu mentransmisikan data dengan cepat dan efisien.*

Dari perhitungan *throughput* ini, jumlah bytes paket data yang diterima adalah 22075433 dengan lama pengamatan 1208,29 detik. Sehingga, besar *throughput* 30% Kb/s dan tergolong pada kategori sedang. Dengan ini, dapat diketahui bahwa sistem mampu mentransmisikan data dengan kecepatan yang stabil dan cukup cepat.

Berdasarkan persamaan 4.4, jumlah paket data yang hilang selama proses pengiriman dan penerimaan data pada sistem dapat dihitung sebagai berikut:

$$\begin{aligned} \text{Packet Loss} &= \frac{(\text{paket data dikirim} - \text{paket data diterima}) \times 100\%}{\text{paket data dikirim}} \\ \text{Packet Loss} &= \frac{4577 - 4577}{4577} = 0 \\ \text{Packet Loss} &= 0 \end{aligned}$$

Packet loss adalah jumlah paket data yang hilang selama proses transmisi. *Packet loss* yang tinggi mengindikasikan masalah dalam jaringan atau sistem komunikasi, yang dapat memengaruhi kualitas dan kehandalan transmisi data. Dalam penelitian ini, hasil perhitungan *packet loss* yang sebesar 0% menunjukkan bahwa tidak ada paket data yang hilang selama proses transmisi data. Hasil perhitungan ini juga menunjukkan bahwa *packet loss* dalam sistem ini memasuki kategori sangat bagus dengan index 4. Dengan ini, sistem monitoring memiliki parameter *packet loss* yang sangat baik.

Secara keseluruhan, hasil perhitungan *delay* dan *jitter* menunjukkan bahwa sistem ini memiliki performa komunikasi yang sangat baik, dengan waktu *delay* yang rendah, dan *jitter* yang minimal, dapat membantu sistem untuk memastikan data dapat dikirimkan dan diterima secara *real-time* dengan kualitas tinggi. Hasil perhitungan *throughput* yang tinggi memastikan bahwa data dapat dikirimkan dengan cepat dan efisien, sementara *packet loss* yang nol menunjukkan bahwa sistem komunikasi sangat handal dan tidak ada kehilangan data selama transmisi.

Berdasarkan pengujian dengan 4 parameter tersebut, dapat disimpulkan bahwa pengimplementasian *Ground Control System* pada *Autonomous Underwater Vehicle (AUV)* telah terbukti dapat berfungsi dengan baik, didukung oleh nilai *delay* yang rendah, *jitter* yang tergolong pada kategori sangat bagus, *throughput* yang baik, dan tidak adanya *packet loss*, membuat sistem ini dapat diimplementasikan pada robot dan dapat meningkatkan efisiensi AUV dalam menjalankan misi bawah air (Rusman & Suwardoyo, 2022).

kesimpulan

Efektivitas Integrasi MQTT-LoRa: Kombinasi protokol MQTT dan teknologi LoRa terbukti efektif dalam mendukung komunikasi jarak jauh dengan konsumsi daya yang rendah untuk aplikasi monitoring AUV. MQTT memungkinkan pengelolaan pesan yang efisien, sementara LoRa menyediakan jangkauan komunikasi yang luas.

Hambatan fisik dan kondisi lingkungan bawah air, seperti kedalaman, tekanan air, dan keberadaan penghalang alami, berpengaruh signifikan terhadap performa sinyal LoRa. Hambatan ini dapat menyebabkan penurunan kekuatan sinyal, peningkatan latensi, dan potensi kehilangan data.

Tingkat QoS memainkan peran penting dalam memastikan keandalan komunikasi. QoS 0 menawarkan transmisi cepat tetapi kurang andal, QoS 1 memberikan keseimbangan antara kecepatan dan keandalan, sedangkan QoS 2 menjamin pengiriman data yang paling andal tetapi dengan potensi peningkatan latensi.

Untuk memastikan monitoring AUV yang efisien, perlu adanya penyesuaian tingkat QoS berdasarkan kondisi spesifik di lapangan. Implementasi QoS yang lebih tinggi dapat diadopsi di daerah dengan hambatan lingkungan yang lebih besar untuk menjaga kehandalan data.

Integrasi MQTT-LoRa perlu didukung dengan strategi pengelolaan hambatan yang baik, seperti penempatan antena yang strategis dan kalibrasi sistem yang rutin. Hal ini akan membantu dalam meminimalkan dampak hambatan lingkungan dan menjaga performa sistem yang optimal.

Daftar Pustaka

- Ar-Rasyid, M. N. (2022). *Persaingan Pengembangan Autonomous Weapon Systems Amerika Serikat Dan China Di Era Kontemporer*.
- Arifin, Z., Ariantini, M. S., Sudipa, I. G. I., Chaniago, R., Dwipayana, A. D., Adhicandra, I., Ariana, A. A. G. B., Yulianti, M. L., Rumata, N. A., & Alfiah, T. (2023). *GREEN TECHNOLOGY: Penerapan Teknologi Ramah Lingkungan Berbagai Bidang*. PT. Sonpedia Publishing Indonesia.
- Astutik, L. Y. (2023). Studi Kinerja Jaringan LoRa: Optimalisasi dan Analisis Efisiensi Komunikasi Nirkabel. *Jurnal Sistem Komputer Dan Informatika (JSON)*, 5(2), 268–277.
- Bachri, S. F. R. (2022). *Implementasi Protocol LoRaWAN pada Wireless Sensor*.

Network untuk Sistem Kompos Pintar dengan modul komunikasi LoRa. Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta.

- Danial, M. M., Imansyah, F., & Sujana, I. (2024). Inovasi Teknologi Lora Dalam Sistem Pemantauan Kualitas Air Sungai Untuk Mendukung Pertumbuhan Budidaya Tambak Ikan Yang Berkelanjutan. *Jurnal Abdi Insani*, 11(3), 2241–2251.
- Mu'ammam, M. F. (2020). Implementasi Pendekatan Saintifik Kurikulum 2013 Pada Mata Pelajaran Sejarah Kebudayaan Islam Di Man 2 Jakarta. Jakarta: FITK UIN Syarif Hidayatullah Jakarta.
- Nurul Hidayat, S. E., Inong Oskar, S. T., Khalid, I. A., Utama, F. Y., Fikri, M. A., Yunus, M. P., Susanti, N. A., Kurniawan, I. W. D., Ramadani, A. H., & Sakti, A. M. (2024). Eksplorasi Proses Manufaktur Untuk Masa Depan Teknologi Dan Produksi. *Cendikia Mulia Mandiri*.
- Palaha, F., Ermawati, E., Machdalena, M., & Arya, E. H. (2021). Analisa Traffic Data Esp8266 Pada Kontrol Dan Monitoring Daya Lisrik Menggunakan Aplikasi Blynk Berbasis Arduino Nano. *Jurnal Nasional Komputasi Dan Teknologi Informasi*, 4(6), 480–489.
- Prima, A. C., Mardiyanto, R., & ST MT, P. D. (2018). Rancang Bangun Underwater Remotely Operated Vehicle Untuk Memantau Nilai Ph Pada Badan Air. *Skripsi]* Institut Teknologi Sepuluh Nopember.
- Purnama, D. A. (2018). Pendekatan Metode Sustainable Value Stream Mapping Menggunakan Integrasi Fuzzy-AHP dan Valsat untuk Meningkatkan Produktivitas (Studi Kasus: CV. Sogan Batik Rejodani).
- Ramdani, F. (2019). Pembangunan Aplikasi Ground Control Station Pada Unmanned Aerial Vehicle (Uav) Berbasis Internet Of Things. Universitas Komputer Indonesia.
- Rusman, A. D. P., & Suwardoyo, U. (2022). Penerapan Sistem Informasi Berbasis IT Pengolahan Data Rekam Medis untuk Peningkatan Pelayanan di Rumah Sakit. Penerbit Nem.
- Saputra, B. (2022). Robot Lengan Spot Welding Pada Batrai Menggunakan Sensor Infrared. *Politeknik Negeri Sriwijaya*.
- Tarigan, W. J., Poddala, P., Prawitasari, P. P., Ikhyannuddin, S. A., Susanti Budiastuti, S. E., Sri Ambarwati, S. E., SM, M., Widyaningsih Azizah, S. E., Nurlina Aris, S. E., & Nugroho, T. P. (2023). Analisis Dan Desain Sistem Akuntansi. *Cendikia Mulia Mandiri*.
- Wisduanto, R. G., Bhawiyuga, A., & Kartikasari, D. P. (2019). Implementasi Sistem

Akuisisi Data Sensor Pertanian Menggunakan Protokol Komunikasi LoRa. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 3(3), 2201–2207.