

Analisa QoS pada MQTT untuk website monitoring dan pengendalian pintu air

Vandy Achmad¹, Agus Khumaidi², Ryan Yudha Adhitya³, Dimas Pristovani Riananda⁴, Imam Sutrisno⁵
e-mail: vandyachmad@student.ppns.ac.id, aguskhumaidi@ppns.ac.id, ryanyudhaadhitya@ppns.ac.id,
dimaspristovani@ppns.ac.id, imam3jpg@yahoo.com

^{1,2,3,4,5}Prodi Teknik Otomasi, Politeknik Perkapalan Negeri Surabaya, Jalan Teknik Kimia, Keputih, Kec. Sukolilo Surabaya, Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 15 Agustus 2023

Direvisi 10 September 2023

Diterbitkan 30 September 2023

Kata kunci:

website
pintu air
MQTT

Keywords:

website
water gate
MQTT

ABSTRAK (9 PT)

Pintu air merupakan sistem yang digunakan untuk mengatur ketinggian air, mengalirkan air untuk diolah PDAM dan mengatur pengendalian debit air agar tidak terjadi banjir. Namun, pada pintu air jagir wonokromo, sistem yang digunakan adalah panel kendali yang dikendalikan oleh operator. Oleh karena itu penulis membuat *website* agar sistem menjadi lebih efisien. *Website* ini dapat memantau ketinggian air, posisi pintu air dan mengirim data untuk mengendalikan pintu air dari pc. Dengan menggunakan sistem ini, sistem pengendalian akan bekerja dengan lebih optimal. Pengujian QoS yang telah dilakukan menunjukkan performa yang bagus dari sisi *delay*, *throughput*, *jitter* dan *packet loss*. Berdasarkan pengujian *delay* mencapai 53.53 ms dan masuk ke dalam kategori bagus. Pengujian *throughput* mencapai 220 Kb/s dan dapat dikategorikan dalam indeks 4 atau sangat bagus. Pengujian *jitter* mencapai 0.17 ms dan masuk ke dalam kategori sedang. Sedangkan pengujian *packet loss* mencapai 0.24 % dan dapat dikategorikan dalam indeks 4 atau sangat bagus. Pengujian mengenai performa jaringan berdasarkan *QoS(Quality of Service)* tersebut terbukti meningkatkan performa operator dalam monitoring dan pengendalian pintu air.

ABSTRACT (9 PT)

Sluice gates are systems used to regulate water levels, drain water to be processed by PDAM and regulate water discharge control to prevent flooding. However, at the jagir wonokromo sluice gate, the system used is a control panel controlled by the operator. Therefore, the author created a website to make the system more efficient. This website can monitor the water level, the position of the sluice gate and send data to control the sluice gate from a PC. By using this system, the control system will work more optimally. QoS testing that has been done shows good performance of delay, throughput, jitter and packet loss. Based on delay testing, it reaches 53.53 ms and falls into the good category. Throughput testing reaches 220 Kb/s and falls into the very good category. Jitter testing reaches 0.17 ms and falls into the medium category. While packet loss testing reaches 0.24% and falls into a very good category. QoS (Quality of Service) testing is proven to improve operator performance in monitoring and controlling floodgates.

Penulis Korespondensi:

Vandy Achmad,
Prodi Teknik Otomasi,
Politeknik Perkapalan Negeri Surabaya,
Jl. Teknik Kimia, Keputih, Kec. Sukolilo, Surabaya, Jawa Timur, Indonesia, 60111.
Email: vandyachmad@student.ppns.ac.id
Nomor HP/WA aktif: 085736636375



1. PENDAHULUAN

Indonesia merupakan negara yang memiliki tingkat intensitas curah hujan yang cukup tinggi dan berpotensi dalam bencana banjir. Dalam mengatasi banjir dibuatlah sistem pintu air untuk mengatur agar volume sungai tidak melebihi batasan tertentu[1]. Aliran air yang meluap di pintu air jagir wonokromo dapat menyebabkan luapan air di kampung dan aliran sungai kecil di sekitarnya. Pengendalian air yang dilakukan selama beberapa tahun terakhir telah memiliki performa yang baik. Tetapi masih terdapat kekurangan di bagian panel kendali yang tidak memiliki sensor dan menggunakan pemodelan pengendalian yang lama. Pengendalian panel menggunakan sistem counter sehingga masih menggunakan indikator berupa lampu ac. Permasalahan tersebut diangkat oleh penulis dan dituangkan dalam penelitian ini. Penggunaan *website* yang telah diterapkan dalam penelitian ini dapat meningkatkan proses distribusi air menjadi lebih cepat dan mengatasi banjir menjadi lebih optimal. Penelitian ini memfokuskan pada proses pengendalian dari *website* agar respon pintu air menjadi lebih cepat dan mengurangi potensi terjadinya banjir.

Penelitian mengenai performa *website* telah banyak diaplikasikan untuk berbagai macam penelitian. Ana dkk [2] telah meneliti mengenai performa *website* tersebut dengan menggunakan protokol *http*. Pengujian terhadap waktu pengiriman dan penerimaan data memiliki rata-rata delay sebesar 2 detik. Hal tersebut menunjukkan performa pengiriman *website* sangat bagus. Penelitian lain mengenai *website* juga dilakukan oleh gilang dkk[3] juga melakukan penelitian mengenai monitoring *water level* pada pintu air menggunakan *Internet Of Things*. Sistem pada penelitian tersebut akan bekerja dalam dua kondisi yakni *manual* dan otomatis. Kedua pengujian pada kondisi tersebut menunjukkan rata-rata *error* untuk *delay* yang cukup kecil yaitu 0.15 %. *Delay* tersebut terjadi karena ketidakstabilan pada jaringan.

Untuk mengatasi ketidakstabilan jaringan dan tetap mempertahankan performa. Maka, penulis menggunakan protokol komunikasi MQTT. Protokol Komunikasi MQTT dapat bekerja pada jaringan yang buruk dan tetap optimal dalam mengirimkan data. Penelitian mengenai performa MQTT telah dilakukan oleh Hilal dkk [4] pada simulasi bendungan. Sistem tersebut bekerja untuk mengirimkan data sensor pada sistem HMI SCADA sebagai sistem monitoring dan pengendalian. Performa penerimaan data MQTT memiliki rata-rata *error* kurang dari 0.10 % dan menunjukkan hasil yang optimal. Selain itu performa dalam pengiriman instruksi pengendalian dengan menggunakan MQTT memiliki delay sebesar 0.84 detik dan masuk dalam kategori bagus untuk parameter *delay*. Penelitian dengan menggunakan analisa *Quality of Service* dilakukan oleh Soma dkk [5] pada sebuah sistem mengenai pengendalian pintu air yang dilakukan terhadap tambak ikan bandeng dengan menggunakan konsep *Internet of Things*. Performa mengenai *delay* didapatkan pada 0.298 detik. Sedangkan *throughput* memiliki performa 2.294 bits/s. Kedua parameter tersebut masuk dalam kategori bagus untuk *throughput* dan *delay*.

Berdasarkan uraian diatas, penulis akan membuat sistem *monitoring* dan pengendalian banjir yang dapat mengatur terbuka dan tertutupnya pintu air. Pengembangan *website* yang dapat melakukan *monitoring* dan pengendalian dapat meningkatkan performa dari sisi pengendalian. Penggunaan sensor ultrasonik untuk mendeteksi ketinggian air dan sensor laser untuk mendeteksi posisi pintu air dapat meningkatkan akurasi dari pengendalian pintu air. Aktivitas untuk pemantauan ketinggian air dapat dilakukan hanya dengan layar pc sehingga *operator* tidak perlu memantau secara *manual*. Proses pengendalian untuk menaikkan pintu air dan menurunkan pintu air dapat dilakukan hanya dengan menekan tombol di *website* sesuai posisi pintu air yang diinginkan.

Dengan menggunakan *website* operator dapat memantau ketinggian air pada bagian hulu, hilir dan posisi pintu air. Selain itu, ketika memasukkan nilai debit air. Maka, pintu air akan bergerak secara otomatis sesuai dengan sistem yang telah ditentukan. Sistem *website* ini akan diterapkan pada rancang bangun sistem yang mengambil referensi dari pintu air jagir wonokromo. Data yang dimasukkan juga merupakan hasil *skala* dan dapat diimplementasikan pada pintu air secara aktual. Sistem yang dibuat akan menggunakan protokol komunikasi MQTT dan menggunakan hosting agar sistem dapat diakses di perangkat manapun. Sistem ini akan dioptimalkan untuk penggunaan pada pc karena monitoring dan pengendalian pintu air akan lebih efektif dengan menggunakan pc. Analisa *Quality of Service* akan menunjukkan performa dari pengiriman dan penerimaan data MQTT pada *website*.



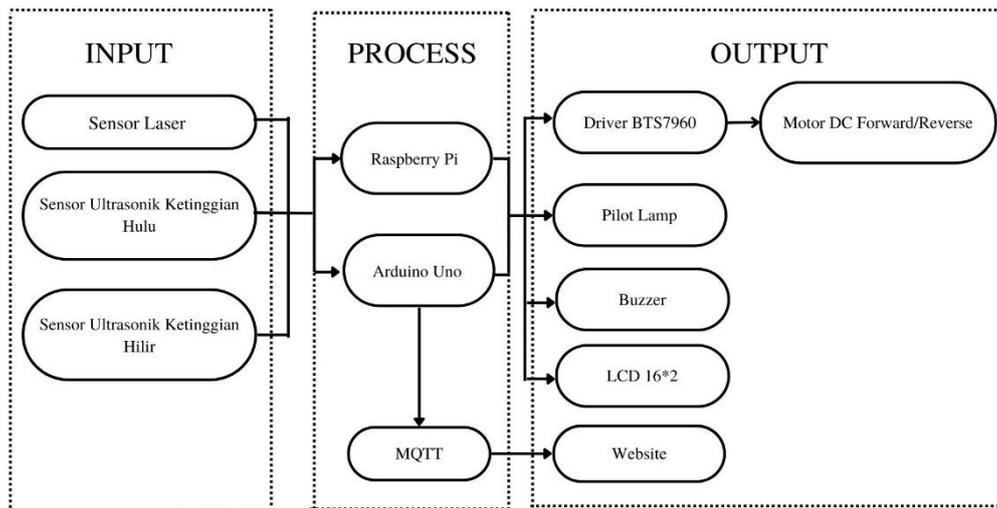
2. METODE PENELITIAN

MQTT atau kepanjangan dari (Message Queue Telemetry Protocol) dapat diartikan sebagai protokol komunikasi yang menggunakan metode *publish* dan *subscribe* untuk mengirim dan menerima data. Sistem ini bekerja dengan menggunakan broker sehingga pengirim dan penerima data harus melalui broker dahulu. Broker akan meneruskan data menuju ke *publisher* atau *subscriber*. MQTT memiliki keunggulan yaitu dapat bekerja pada jaringan yang buruk, memiliki paket data yang ringan, dapat bekerja secara asinkron dan mudah diimplementasikan dalam berbagai penggunaan[6].

Raspberry Pi 3 Model B adalah sebuah *microprocessor* yang digunakan untuk berbagai macam penggunaan. Diantaranya, untuk sistem Internet of Things, pengembangan perangkat keras, pengembangan aplikasi dan sebagai server kecil dalam basis data. Raspberry Pi menggunakan sistem operasi linux berbasis debian yang diberi nama raspbian. Beberapa fitur raspberry pi secara *hardware* dapat digunakan untuk pin i2c, pin spi, pwm(*pulse width modulation*), pin audio dan pin kamera[7]. Qos atau kepanjangan dari (*Quality of Service*) digunakan untuk mengecek kemampuan sistem atau jaringan dan memastikan kualitas dari sebuah jaringan pada sebuah *website*. Terdapat beberapa parameter seperti *delay*, *packet loss*, *throughput* dan *jitter* yang dapat digunakan untuk menunjukkan performa dari jaringan pada *website*. [8].

2.1 Perancangan Sistem

Perancangan sistem dapat diartikan sebagai referensi dalam membuat sebuah alat yang ada pada penelitian. Penelitian ini berfokus pada pembuatan sistem untuk monitoring dan pengendalian pintu air.



Gambar 1: Perancangan Sistem

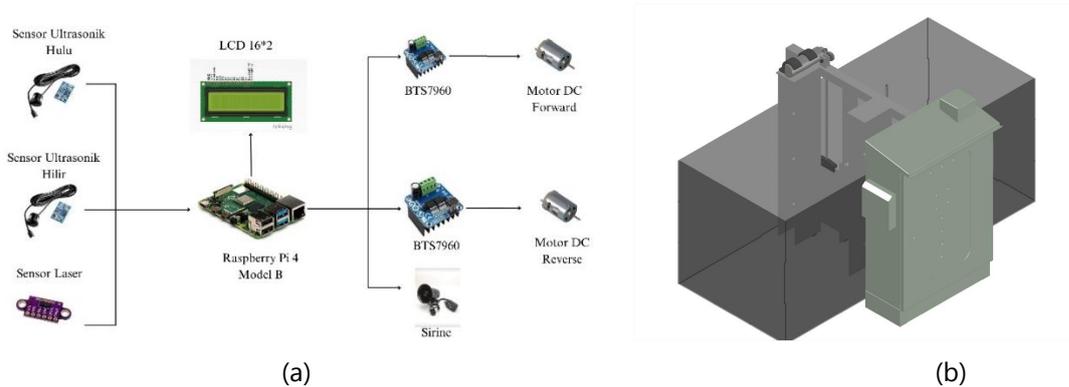
Gambaran tentang rancangan dan referensi mengenai rancangan sistem ditunjukkan oleh Gambar 1 mengenai perancangan sistem. Sensor ultrasonik akan digunakan untuk mendeteksi ketinggian air pada bagian hulu dan hilir. Sedangkan sensor laser akan mendeteksi posisi pintu air. Data tersebut akan dikirimkan pada Arduino uno untuk diproses dan dikirimkan pada raspberry pi. Raspberry pi akan bertindak untuk mengirim data menggunakan MQTT pada *website* dan menerima data dari *website*. Output dari sistem ini akan menggerakkan motor dc melalui driver motor BTS7960. Selain itu sistem akan menyalakan pilot lamp dan membunyikan buzzer. Sistem akan menampilkan data juga pada lcd.

2.2 Perancangan Hardware

Dalam membuat sebuah sistem elektrik dan mekanik agar berjalan sesuai sistem. Maka diperlukan sebuah rancangan berupa *hardware* untuk sistem secara elektrik dan mekanik. Pembuatan sistem elektrik akan diletakkan pada panel. Pembuatan *hardware* mekanik memiliki ukuran 70 cm × 30 cm × 37 cm. Sistem mekanik akan terdiri atas



pintu air, sistem katrol, motor dc, sensor ultrasonik, sensor laser. dan lain-lain. Desain dari sistem mekanik dibuat dengan menggunakan 3D. Perancangan Elektrik ditunjukkan oleh gambar 2(a). Sedangkan desain mekanik *hardware* pada gambar 2(b).



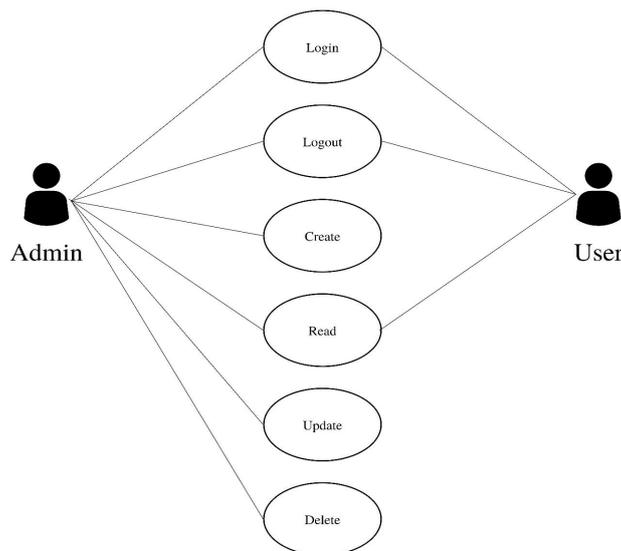
Gambar 2: (a) Perancangan Elektrik *Hardware*, (b) Desain Mekanik *Hardware*

2.3 Perancangan *Software*

Dalam merancang perangkat lunak atau dapat juga diartikan sebagai *software* ini sangat diperlukan untuk menunjukkan cara kerja atau langkah dari sistem secara *software*. Perancangan perangkat lunak akan terdiri dari diagram *use case* untuk menunjukkan langkah kerja sistem. Selain itu akan ditunjukkan mengenai relasi *database* untuk menunjukkan fitur-fitur yang dapat digunakan oleh pengguna. Arsitektur protokol *website* juga akan dijelaskan pada bagian ini.

a. Diagram *Use Case*

Use Case digunakan untuk menunjukkan urutan langkah-langkah yang terkait untuk menjelaskan cara kerja dari sebuah *website*[9]. *User* atau pengguna dari *website* harus register terlebih dahulu untuk dapat melakukan login. Terdapat dua jenis level pada sistem yaitu user biasa dan admin. *User* biasa dapat mengakses fitur berupa login, *logout* dan monitoring. *User* biasa dapat mengakses data level ketinggian air, kondisi pintu air dan client yang sedang mengakses *website*. Sedangkan Admin dapat melakukan *login*, *logout*, *create*, *read*, *update* dan *delete*. Admin dapat mengakses monitoring dan kendali pada *website*. Admin juga dapat menghapus, mengupdate dan membuat tambahan tampilan pada *website*. Gambar 3 menunjukkan gambar diagram *use case*.



Gambar 3: Diagram *Use Case*



b. Relasi Database

Desain dari database digunakan untuk basis data dari tabel MySQL.[10] Desain ini menjelaskan bahwa terdapat 4 jenis tabel pada database MySQL. Tabel pertama digunakan untuk menjelaskan mengenai login. Tabel mengenai login terdiri atas 4 variabel yakni *id*, *username*, *password* dan *id_level*. Tabel kedua digunakan untuk mengatur level yang terdiri atas dua hal yakni, *admin* dan *user_biasa*. Sedangkan tabel ketiga digunakan untuk menyimpan device yang mengakses website. Tiap device yang mengakses website akan disimpan pada bagian ini. Tabel keempat digunakan untuk menyimpan data dari sensor dan pengendalian pintu air.

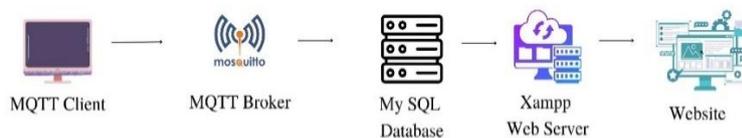
Pada dasarnya, pengguna terdiri atas dua jenis yaitu, *admin* dan *user* biasa. Sistem untuk monitoring dapat diakses oleh user biasa sesuai yang ditunjukkan oleh tabel keempat. User biasa dapat juga mengunduh laporan mengenai ketinggian air hulu, ketinggian air hilir dan posisi pintu air secara *real-time*. Sedangkan *admin* dapat mengakses seluruh data, mengubah data dan mengendalikan sistem. Gambar 4 menunjukkan gambar relasi database.



Gambar 4: Relasi Database

c. Arsitektur Protokol Website

Arsitektur Protokol Website digunakan untuk menunjukkan proses pengiriman data pada website. Bagian ini akan menjelaskan cara kerja pengiriman data dari client hingga tampil pada website[11]. MQTT Client dapat bertindak sebagai publisher atau subscriber. Raspberry pi akan bertindak sebagai pengirim atau publisher ketika mengirim data menuju website. Sedangkan akan bertindak sebagai penerima atau subscriber ketika menerima data dari website. Website akan bertindak sebagai publisher untuk mengirim data menuju raspberry pi. Ketika website menerima data dari Raspberry Pi maka website tersebut bertindak sebagai subscriber. Raspberry Pi akan mengirim data sensor enuju MySQL Database dengan perantara MQTT Broker. Data tersebut akan ditampilkan pada website di bagian monitoring. Website bekerja dengan menggunakan hosting sehingga sistem dapat diakses dari pc manapun. Gambar 5 menunjukkan arsitektur protokol website.



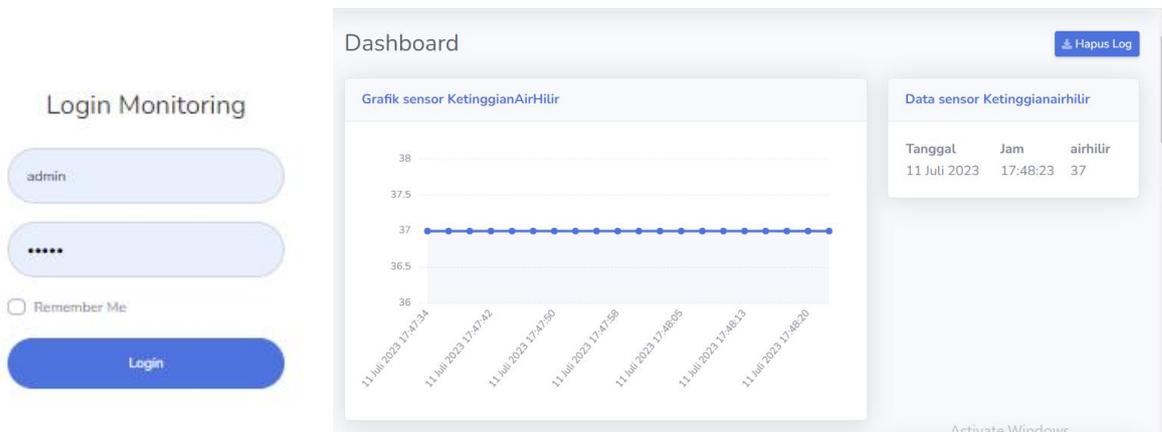
Gambar 5: Arsitektur Protokol Website

d. Tampilan Antarmuka Website

Tampilan antarmuka website digunakan untuk menampilkan halaman website berupa 4 tampilan utama. Tampilan pertama adalah halaman login. Sebelum memasuki tampilan utama website sebuah halaman login harus diisi terlebih dahulu. Syarat untuk memasuki website adalah mengisi dua hal yakni *username* dan *password*. Lalu setelah user berhasil masuk ke dalam website. User akan masuk ke tampilan dashboard untuk monitoring dan



pengendalian pintu air. Gambar 6(a) menunjukkan halaman *login*. Sedangkan, gambar 6(b) menunjukkan *dashboard*.



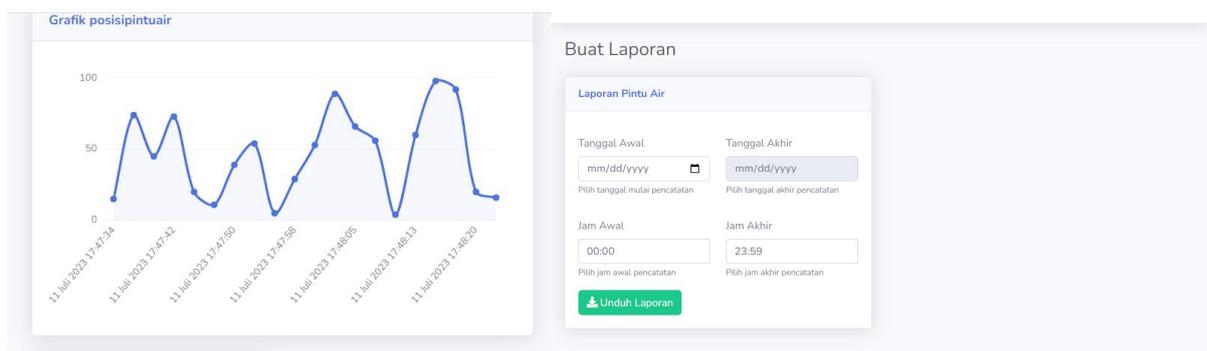
(a) (b)
Gambar 6: (a) Tampilan *website* untuk halaman *Login*, (b) Tampilan *website* untuk *dashboard*

Tampilan *monitoring* digunakan untuk menampilkan antarmuka menu grafik dan data dari ketinggian air hulu, ketinggian air hilir dan posisi pintu air. Gambar 7(a) menunjukkan antarmuka menu ketinggian hilir. Sedangkan gambar 7(b) menunjukkan ketinggian hulu.



(a) (b)
Gambar 7: (a) antarmuka menu ketinggian hilir , (b) antarmuka menu ketinggian hulu

Antarmuka posisi pintu air ditunjukkan oleh gambar 8(a). Sedangkan, antarmuka mengenai menu laporan ditunjukkan pada gambar 8(b). Menu laporan digunakan untuk menampilkan laporan ketinggian air hulu, ketinggian air hilir dan posisi pintu air yang dapat diunduh berupa pdf.



(a) (b)
Gambar 8: (a) antarmuka menu posisi pintu air , (b) antarmuka menu laporan



3. HASIL DAN PEMBAHASAN

3.1 Pengujian Perangkat Lunak

Penulis telah melakukan penelitian yang akan mengimplementasikan protokol *MQTT* yang digunakan untuk menerima data sensor dan data aktuator dan sebagai penghubung antara *raspberry pi* dengan *website*. Perancangan *website* menggunakan *Visual Studio Code*. Bahasa yang digunakan untuk membuat tampilan adalah kombinasi antara HTML, CSS dan PHP. Styling Website menggunakan bootstrap untuk format tabel, grafik dan menu[12]. Integrasi antara *website* dan *raspberry pi* menggunakan *MQTT* untuk mengirim data sensor dan menerima data debit air dari *website*. Gambar 6 menunjukkan proses integrasi *raspberry pi* yang dilakukan pada *website*.

```
9  port = 1003
10
11 # Konfigurasi URL API untuk menyimpan data
12 api_url = "https://pintuairvandy.monitoringonline.net/fromdatabase.php"
13
14 # Inisialisasi objek serial
15 ser = serial.Serial('/dev/ttyUSB0', 115200)
16
17 # Fungsi untuk mengirimkan data melalui MQTT
18 def send_data_mqtt(data):
19     # Mengirimkan data ke MQTT broker
20     formatted_data = json.dumps(data)
21     client.publish("pintuair/monitoring123", formatted_data)
```

Shell

```
Sent data to MQTT broker: {"posisi_pintuair": 7.8, "ketinggianairhulu": 25.0, "ketinggianairhulu": 26.0}
Data sent to API successfully
API Response: Data saved to database
```

Gambar 6: Proses Integrasi Raspberry Pi dengan *website*

3.2 Pengujian QoS Website

Pengujian pada *website* ini bias dikatakan pengujian dari sisi *subscriber*. Hal tersebut dikarenakan pengujian ini berfokus pada proses penerimaan data dari *raspberry pi*. Pengujian pada QoS website akan digunakan untuk menguji 4 parameter yaitu *delay*, *throughput*, *jitter* dan *packet loss*. Kecepatan pengiriman data ditentukan oleh *throughput*. Sedangkan waktu jeda antara data yang tampil dan yang diterima ditentukan oleh *delay*. *Jitter* menentukan trafik jaringan. *Packet Loss* sendiri menentukan seberapa banyak data yang hilang dalam jaringan. Berdasarkan tabel 1 dapat diketahui bahwa *delay* menunjukkan kategori bagus yaitu 53.53 ms. *Jitter* dapat dikategorikan pada indeks 4 dan dapat dikatakan sebagai kategori sangat bagus yaitu 0.17 ms. Sedangkan *throughput* masuk ke dalam kategori sedang yaitu 220 Kb/s. Lalu, *packet loss* termasuk kategori sangat bagus dengan persentase 0.24%.

TABEL I : PENGUJIAN QoS WEBSITE

| No | Parameter | Nilai parameter | Kategori |
|----|-------------|-----------------|--------------|
| 1 | Delay | 53.53 ms | Bagus |
| 2 | Jitter | 0.17 ms | Sangat Bagus |
| 3 | Throughput | 220 Kb/s | Sedang |
| 4 | Packet loss | 0.24 % | Sangat Bagus |

3.3 Pengujian QoS Thonny ide

Pengujian *quality of service* juga dilakukan dari sisi Thonny ide yang ada di *raspberry pi* untuk menguji pengiriman data dan *MQTT* ketika bertindak sebagai *publisher*. Pengujian juga dilakukan dari sisi *publisher* untuk menunjukkan hasil kinerja dari *MQTT* dari segi pengiriman data. Data yang dikirimkan menggunakan *raspberry pi* akan dianalisa lalu lintas jaringannya dan diuji 4 parameter dengan cara yang sama dengan pengujian *website*. *Delay* dapat dikatakan untuk masuk sebagai kategori sangat bagus yaitu 10.03 ms. *Jitter* dapat dikatakan untuk masuk sebagai kategori sangat bagus yaitu 33.47 ms. Lalu, *throughput* termasuk dalam kategori sedang yaitu 328 kb/s. Sedangkan, *packet loss* masuk ke dalam kategori sangat bagus yaitu 0.54 %.



TABEL 2 : PENGUJIAN QOS SUBSCRIBER

| No | Parameter | Nilai parameter | Kategori |
|----|-------------|-----------------|--------------|
| 1 | Delay | 10.03 ms | Sangat Bagus |
| 2 | Jitter | 33.47 ms | Sangat Bagus |
| 3 | Throughput | 328 Kb/s | Sedang |
| 4 | Packet loss | 0.54 % | Sangat Bagus |

4. KESIMPULAN

Peneliti telah melakukan penelitian dan dapat menarik kesimpulan mengenai *quality of service* yang menunjukkan performa yang bagus dari sisi *website* dan dari sisi *publisher*. Terdapat 4 parameter atau variabel dalam menentukan *quality of service* seperti *delay*, *packet loss*, *throughput* dan *jitter* menunjukkan performa yang bagus. *Delay* pada sisi subscriber menunjukkan nilai 53.53 ms dan termasuk kategori sangat bagus. Sedangkan pada Thonny ide menunjukkan nilai 10.03 dan termasuk kategori sangat bagus juga. *Jitter* pada *website* menunjukkan nilai 0.17 ms dan termasuk ke dalam kategori sangat bagus. Lalu ,pada *website* nilai *jitter* mencapai 33.37 ms dan masuk ke dalam kategori sangat bagus. Pada saat pengujian *throughput* nilai pada sisi *website* adalah 220 Kb/s. Sedangkan nilai pada sisi thonny ide adalah 328 Kb/s. Kedua sisi tersebut menunjukkan kategori sedang. Pada pengujian *packet loss* kedua sisi menunjukkan kategori sangat bagus dengan rincian 0.24 % pada sisi *website* dan 0.54 % pada sisi *subscriber*. Hal tersebut menunjukkan bahwa *website* telah layak dan optimal untuk digunakan dalam monitoring dan pengendalian pintu air.

DAFTAR PUSTAKA

- [1] H. M. Imaaduddin, K. Kuntjoro, I. Sa'ud, D. Harijanto, and D. Indriyani, "Pemanfaatan Kolam Tampung Sebagai Upaya Mitigasi Banjir Kawasan Pada Sub Sistem Medokan Ayu Kota Surabaya," *J. Apl. Tek. Sipil*, vol. 20, no. 2, p. 243, 2022, doi: 10.12962/j2579-891x.v20i2.12519.
- [2] A. Hasanah, S. A. Saptari, and D. Lestari, "Sistem Deteksi Banjir Dan Pintu Air Otomatis Menggunakan Raspberry Pi 3 Berbasis Website," *InfoTeklar J. Nas. Inform. dan Teknol. Jar.*, vol. 4, no. 2, pp. 250–254, 2020, [Online]. Available: <https://jurnal.uisu.ac.id/index.php/infotekjar/article/view/2286>
- [3] G. A. Saputra and R. N. Rohmah, "Monitoring Water Level Dan Pengendalian Pintu Bendungan Berbasis Iot (Internet of Things)," *Emit. J. Tek. Elektro*, vol. 22, no. 1, pp. 83–91, 2022, doi: 10.23917/emit.v22i1.16726.
- [4] A. Sahi, "Aplikasi Test Potensi Akademik Seleksi Saringan Masuk Lp3I Berbasis Web Online Menggunakan Framework Codeigniter," *Tematik*, vol. 7, no. 1, pp. 120–129, 2020, doi: 10.38204/tematik.v7i1.386.
- [5] S. Sunarya, R. Munadi, and ..., "Rancang Bangun Sistem Kendali Pintu Air Pada Tambak Ikan Bandeng (Chanos-Chanos) Berbasis IoT," *eProceedings ...*, vol. 7, no. 3, pp. 9191–9198, 2020, [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/viewFile/14083/13823>
- [6] B. P. Dessyanto, "Protokol Jaringan dalam Internet of Things," *LPPM UPN "Veteran" Yogyakarta*, p. 105, 2020, [Online]. Available: <http://eprints.upnyk.ac.id/27440/1/buku-protokol-jaringan-dalam-iot-dan-sertifikat-haki.pdf>
- [7] L. Setiyani, "Perancangan dan Implementasi IoT (Internet of Things) pada Smarthome Menggunakan Raspberry Pi Berbasis Android," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 10, no. 2, pp. 459–466, 2019.
- [8] A. R. Maulana, H. Walidainy, M. Irhamsyah, F. Fathurrahman, and A. Bintang, "Analisis Quality of Service (Qos) Jaringan Internet Pada Website E-Learning Universitas Syiah Kuala Berbasis Wireshark," *J. Komputer, Inf. Teknol. dan Elektro*, vol. 6, no. 2, pp. 27–30, 2021, doi: 10.24815/kitektro.v6i2.22284.
- [9] T. Arianti, A. Fa'izi, S. Adam, and Mira Wulandari, "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (Unified Modelling Language)," *J. Ilm. Komput. ...*, vol. 1, no. 1, pp. 19–25, 2022, [Online]. Available: <https://journal.polita.ac.id/index.php/politati/article/view/110/88>
- [10] Ismai, "Perancangan Basis Data Sistem Informasi Perwira Tugas Belajar (Sipatubel) Pada Kementerian Pertahanan," *Senamika*, vol. 1, no. 2, pp. 222–233, 2020.
- [11] D. I. Saputra, G. M. Karmel, and Y. B. Zainal, "Perancangan Dan Implementasi Rapid Temperature Screening Contactless Dan Jumlah Orang Berbasis Iot Dengan Protokol Mqtt," *J. Energy Electr. Eng.*, vol. 2, no. 1, pp. 20–30, 2020, doi: 10.37058/jeee.v2i1.2147.
- [12] M. Y. Putra, "Responsive Web Design Menggunakan Bootstrap Dalam Merancang Layout Web," *Inf. Syst. Educ. Prof.*, vol. 5, no. 1, pp. 61–70, 2020.

