

Implementasi *Internet of Things* untuk *Prototype* Pembersih Kaca Gedung Bertingkat Menggunakan *Raspberry Pi* dan Aplikasi Android

Maula Zikri Sinaga¹, Hari Toha Hidayat^{2*}, Guntur Syahputra³

^{1,2,3} *Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B. Aceh Medan Km.280 Buketrata 24301 INDONESIA*

¹maulazikrisinaga23@gmail.com

^{2*}haritoha@pnl.ac.id

³guntursyahputra@pnl.ac.id

Abstrak— Kebersihan merupakan aspek penting yang berdampak signifikan pada kesehatan manusia. Namun, banyak orang masih kurang peduli dalam menjaga kebersihan, termasuk kebersihan kaca pada gedung bertingkat yang sering diabaikan. Hingga kini, proses pembersihan kaca pada gedung bertingkat masih banyak dilakukan secara manual oleh manusia, yang tidak hanya berisiko tinggi terhadap keselamatan tetapi juga memerlukan biaya operasional yang besar dengan hasil yang kurang optimal. Penelitian ini mengembangkan sebuah sistem robot pembersih kaca gedung bertingkat berbasis *Raspberry Pi* yang terintegrasi dengan teknologi *Internet of Things (IoT)*. Robot ini dikendalikan melalui *smartphone Android* yang mengirimkan perintah ke *Raspberry Pi*, yang berfungsi sebagai prosesor untuk menggerakkan motor penggerak, mengatur pergerakan robot ke kiri, dan kanan, serta mengontrol pompa air untuk menyemprotkan cairan pembersih kaca. Robot akan bekerja sesuai dengan pola yang telah dirancang dan mengirimkan data status secara real-time ke *Firestore*, memungkinkan monitoring yang efisien dan berkelanjutan. Selain itu, robot ini dilengkapi dengan sensor hujan yang mampu mendeteksi tetesan air hujan, sehingga robot dapat menunda pembersihan hingga kondisi cuaca memungkinkan. Hasil pengujian menunjukkan bahwa sistem robot pembersih kaca ini mampu beroperasi dengan efisien dan efektif. Pengujian kualitas layanan (*QoS*) menunjukkan bahwa throughput sistem tetap stabil, dengan tingkat *packet loss* sebesar 0%, memastikan bahwa semua data perintah dan status antara perangkat *Android* dan robot dikirimkan serta diproses dengan cepat dan akurat. Dengan demikian, robot ini tidak hanya meningkatkan efisiensi dan keselamatan dalam proses pembersihan kaca gedung bertingkat, tetapi juga mengurangi biaya operasional secara signifikan.

Kata kunci— *Raspberry Pi*, *Smartphone*, Pembersih Kaca, *Internet of Things*, *QoS*, *Firestore*

Abstract— Cleanliness is an important aspect that has a significant impact on human health. However, many people still do not care about maintaining cleanliness, including the cleanliness of glass in high-rise buildings which is often ignored. Until now, the process of cleaning glass in high-rise buildings is still mostly done manually by humans, which not only poses a high risk to safety but also requires large operational costs with less than optimal results. This study developed a high-rise building glass cleaning robot system based on *Raspberry Pi* integrated with *Internet of Things (IoT)* technology. This robot is controlled via an *Android smartphone* that sends commands to the *Raspberry Pi*, which functions as a processor to drive the motor, regulate the movement of the robot to the left and right, and control the water pump to spray glass cleaning fluid. The robot will work according to the designed pattern and send real-time status data to *Firestore*, enabling efficient and sustainable monitoring. In addition, this robot is equipped with a rain sensor that can detect raindrops, so that the robot can postpone cleaning until weather conditions allow. The test results show that this glass cleaning robot system is able to operate efficiently and effectively. Quality of service (*QoS*) testing shows that the system throughput remains stable, with a packet loss rate of 0%, ensuring that all command and status data between the *Android device* and the robot are delivered and processed quickly and accurately. Thus, the robot not only improves the efficiency and safety of the glass cleaning process of high-rise buildings, but also significantly reduces operational costs.

Keywords— *Raspberry Pi*, *Smartphone*, *Glass Cleaner*, *Internet of Things*, *QoS*, *Firestore*

I. PENDAHULUAN

Pembersihan adalah hal terpenting yang harus diperhatikan seiring berjalannya waktu. Profesi cleaning service memegang peranan yang sangat penting dalam menjaga kebersihan kantor, mulai dari membersihkan lantai dan ruangan hingga membersihkan kaca. Pada gedung perkantoran bertingkat, karena lokasi kerja yang tinggi, petugas kebersihan mempunyai risiko lebih tinggi jika kaca bangunan dibersihkan secara manual.[1] Pada saat ini, gondola masih digunakan untuk membersihkan kaca pada gedung-gedung bertingkat. Gondola atau keretgantungan merupakan alat penunjang yang membantu pekerja yang bekerja diluar gedung bertingkat. Penggunaan gondola dalam membersihkan kaca bagian luar di ketinggian akan sangat

berisiko bagi pekerja. Meskipun gondola yang digunakan sudah memenuhi syarat Keselamatan dan Kesehatan Kerja (K3), namun pekerjaan ini masih dapat mengancam nyawa pekerjanya[2].

Di era modern ini, banyak pekerjaan yang memerlukan penggunaan teknologi canggih, salah satunya adalah robot prototipe pembersih kaca untuk gedung bertingkat. Robot ini dirancang untuk mempercepat proses pembersihan, terutama pada gedung-gedung tinggi, sekaligus meringankan beban pekerjaan manusia dalam bidang kebersihan.[3] Dalam upaya mengatasi permasalahan pembersihan kaca pada gedung bertingkat, terciptalah sebuah prototipe robot pembersih kaca berbasis *RaspberryPi* yang dikontrol melalui perangkat *Android*. Teknologi ini bekerja dengan menggunakan water pump yang mengalirkan cairan pembersih, yang kemudian disapu menggunakan wiper yang digerakkan oleh motor

servo.[4] Raspberry Pi berfungsi sebagai media pengolah data, memberikan perintah kepada motor penggerak untuk bergerak ke kiri dan ke kanan. Selain itu, robot ini dilengkapi dengan sensor hujan yang akan mendeteksi tetesan air hujan, sehingga dapat bekerja secara otomatis ketika kondisi cuaca memungkinkan seluruh sistem akan dimonitoring melalui smartphone Android. Dengan dilakukan penelitian tersebut diharapkan dapat memberikan inovasiteknologi baru, mengurangi resiko terjadinya kecelakaan kerja dan meningkatkan efisiensi waktu dan biaya dalam proses perawatan gedung bertingkat.[5] Pada umumnya proses pembersihan kaca pada gedung bertingkat menggunakan tenaga konvensional yaitu oleh tenaga kerja manusia. Proses ini relative sederhana namun memiliki risiko kecelakaan kerja. Oleh karena itu, diperlukan sistem mekanis yang dapat menggantikan manusia diantaranya dengan menggunakan robot pembersih kaca.[6]

A. State of the Art

Penelitian tentang penerapan *Internet of Things* (IoT) sudah banyak dikerjakan oleh para peneliti, namun hingga sekarang IoT masih menjadi bahan penelitian yang menarik karena masih begitu banyak permasalahan yang belum ditemukan solusi yang optimal, namun metode untuk itu telah dikembangkan dan menghasilkan sesuatu yang relevan dengan penelitian yang menggunakan *Raspberry Pi* dan Android. Penelitian ini mengadopsi metode MQTT (*Message Queuing Telemetry Transport*), yaitu protokol komunikasi dengan konsep *publish/subscribe* dan dirancang khusus untuk komunikasi antar perangkat berdaya rendah. Protokol MQTT berjalan di atas TCP/IP dan ideal untuk digunakan di *Internet of Things* (IoT).[2]

B. Internet of Things (IoT)

Internet of Things, atau disingkat IoT, adalah sebuah konsep yang memperluas manfaat dari koneksi internet yang selalu terhubung. Fungsi serupa dapat mengembangkan data, remote control, dan lainnya, termasuk objek dunia nyata. Contohnya termasuk makanan, elektronik, barang koleksi, peralatan, termasuk barang-barang rumah tangga, semua terhubung dan aktif melalui sensor tertanam ke jaringan global dan lokal. Istilah IoT (*Internet of Things*) diciptakan pada tahun 1999 oleh Kevin Ashton, direktur eksekutif MIT Auto ID Centre. Mereka juga menemukan sistem identifikasi RFID (*Radio Frequency Identification*) berdasarkan perangkat global. Penemuan ini disebut sebagai lompatan besar dalam commercialising IoT.[10]

C. Raspberry Pi

Raspberry Pi atau disingkat Raspi adalah komputer mini yang dirancang dan dikembangkan di Inggris oleh *Raspberry Pi Foundation*. Pada tahun 2006 ide komputer mini ekonomis untuk anak-anak muncul. Ide muncul ketika ilmuwan komputer Eben Upton, bersama dengan Rob Mullins, Jack Lang, dan Alan Mycroft, memperhatikan bahwa banyak mahasiswa baru yang melarikan diri dari aspek teknis komputasi. Hal ini terutama disebabkan oleh program pembelajaran sekolah yang lebih menekankan penggunaan komputer daripada memahaminya[7]

D. Sensor Hujan

Rain Sensor adalah jenis perangkat yang dapat mendeteksi tetesan hujan. Sensor hujan digunakan sebagai

switch yang digerakkan berdasarkan curah hujan. Sensor ini bekerja layaknya sebuah saklar. Ketika sensor terkena tetesan air pada permukaan bantalan sensor maka modul sensor akan membaca data dari bantalan sensor untuk memproses dan mengubah data tersebut menjadi output analog atau digital. Pada prinsipnya, papan sensor mencakup satu set jejak tembaga terbuka yang bekerja bersama seperti resistor variabel atau potensiometer. Resistansi papan sensor akan bervariasi sesuai dengan jumlah air yang jatuh di permukaannya. Oleh karena itu, hambatan di sini berbanding terbalik dengan jumlah air. Output dari sensor ini tergantung pada resistansinya.[8]

E. Pompa Air

Pompa adalah suatu alat yang digunakan untuk memindahkan suatu zat cair dari suatu tempat ke tempat lain melalui pipa (saluran) dengan menambahkan energi pada cairan yang dipindahkan secara terus menerus. Pompa bekerja berdasarkan prinsip perbedaan tekanan antara bagian hisap dan bagian distribusi. Perbedaan tekanan tersebut merupakan akibat dari suatu mekanisme seperti putaran roda impeller yang menyebabkan sisi hisap nyaris vakum.[10]

F. Motor Dc

Motor DC merupakan motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar).[3]

G. Firebase

Firestore Realtime Database merupakan platform Database yang digunakan pada aplikasi realtime. Ketika terjadi perubahan data, maka aplikasi yang terhubung dengan firebase akan memperbaharui secara otomatis melalui setiap device (perangkat) baik website ataupun mobile. Firebase mempunyai library (pustaka) yang lengkap untuk sebagian besar platform web dan mobile. Firebase dapat digabungkan dengan framework lain seperti node, java, javascript, dan lain-lain. [6]

H. MQTT

MQTT (*Message Queuing Telemetry Transport*) adalah protokol yang beroperasi di atas stack TCP/IP dan dirancang khusus untuk mesin-mesin (M2M) dan tidak memiliki alamat khusus. Tidak memiliki alamat khusus ini seperti arduino, raspi atau perangkat lain yang tidak memiliki alamat khusus. Sistem kerja MQTT menerapkan data *Publish* dan *Subscribe*[9].

II. METODOLOGI PENELITIAN

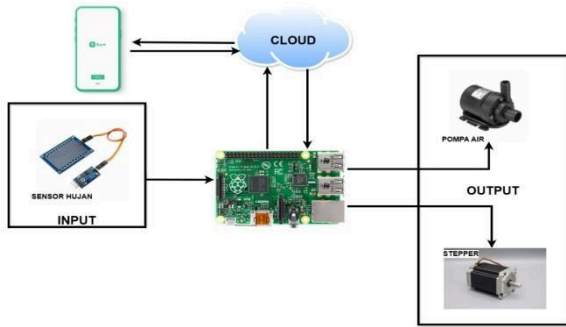
A. Data dan Pengumpulan Data

Sumber data dalam penelitian ini menggunakan data primer. Setelah alat dirangkai, dilakukan pengujian untuk memperoleh data yang dibutuhkan.

B. Rancangan Sistem (software/hardware)

Perancangan sistem digunakan untuk menjelaskan gambaran sistem perancangan yang akan dibuat. Blok

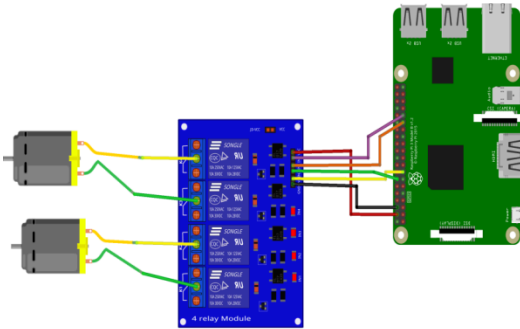
diagram sistem secara keseluruhan menjelaskan bagaimana penerapan *Internet of Things* pada robot pembersih kaca gedung bertingkat dapat bekerja. Blok diagram atau gambaran perancangan sistem dapat dilihat pada gambar 3.2 berikut:



Gambar 1 Blok Diagram Sistem

C. Perancangan Relay 4 Channel dan DC Motor GearBox

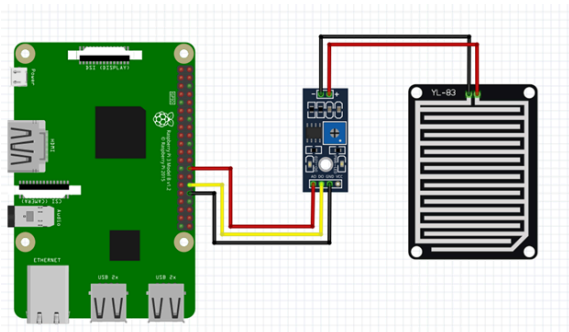
Relay yang digunakan dengan tipe relay 4 channel *Power supply* range from 5V~7.5V dan motor yang digunakan *Gear Box* untuk menggerakkan *wipper* pembersih kaca yang menyediakan torsi tinggi dan kontrol kecepatan yang presisi, Gear box memiliki 2 Pin. Rangkaian yang dirancang dapat dilihat pada gambar 3.2 berikut.



Gambar 2 Blok Diagram DC Motor GearBox

D. Perancangan Sensor Hujan

Sensor yang digunakan adalah sensor hujan dengan tegangan operasi antara 3,3 hingga 5V. Sensor ini memiliki 3 pin. Rangkaian yang dirancang dapat dilihat pada Gambar 3.3 berikut.



Gambar 3 Blok Diagram Sensor hujan

E. Teknik Pengujian

Tahapan pengujian akan dilakukan setelah data terkumpul pada saat pengujian alat.

1. Memastikan robot pembersih kaca jendela gedung merespons secara tepat terhadap berbagai perintah yang dikirimkan melalui aplikasi Android.
2. Menguji apakah stepper motor dan pompa air berfungsi dengan baik sesuai dengan perintah yang diberikan melalui *Raspberry Pi*.

III. HASIL DAN PEMBAHASAN

A. Hasil Pengujian

Tahap pengujian sistem salah satu tahap untuk meletakkan sistem supaya siap untuk dioperasikan. Beberapa aktifitas secara berurutan berlangsung pada tahap ini, melakukan pengujian dari setiap *performa response* pada robot, pengujian perancangan sistem.

B. Pengujian Qos

Pada pengujian ini alat dan sistem dilakukan pengujian untuk dapat mengetahui fungsi dan tujuan dari penelitian tercapai sesuai dengan yang diharapkan. Pada pengujian alat dan sistem ini dipakai teknik pengujian dengan menggunakan *Wireshark* yang mana pengujian berfokus untuk seberapa waktu merespon saat pengiriman perintah dari *android* ke alat.

C. Pengujian Delay

Pada pengujian ini dilakukan perhitungan untuk mengetahui durasi waktu (*delay*) yang didapat selama pengiriman data menggunakan *wireshark* dilakukan pengujian dengan menekan tombol otomatis, dan didapatkan hasil total *delay* dan rata-rata *delay* sebagai berikut.

Tabel 1
Pengujian 1 Delay Otomatis

Delay	Total = 81.421.729 Second/detik
	Rata-Rata = 0,433094303 Second/ detik

Pada tabel 4.1 dilakukan pengujian 1 secara otomatis, pada pengujian ini dilakukan perhitungan untuk durasi waktu selama pengiriman data secara otomatis ke *firebase* dan ke robot pembersih kaca. Total *delay* yang didapat dalam pengiriman data secara otomatis Total = 81.421.729 Second/detik dan rata-rata *delay* yang didapatkan selama proses mengirimkan data yaitu Rata-Rata = 0.433094303 Second/detik.

Tabel . 2
Pengujian 2 Delay Otomatis

Delay	Total = 120.318.289 Second/detik
	Rata-Rata = 8594163,5 Second/ detik

Pada tabel 2 dilakukan pengujian 2 secara otomatis, pada pengujian ini dilakukan perhitungan untuk durasi waktu

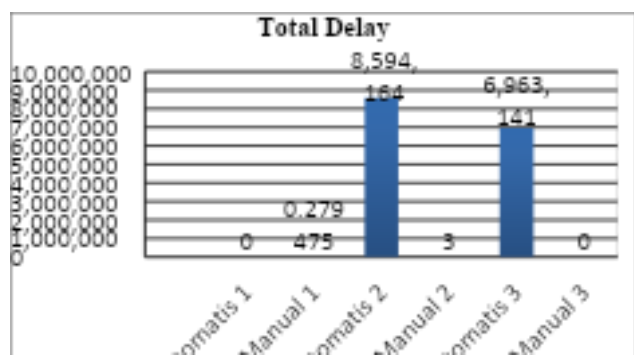
selama pengiriman data secara otomatis ke *firebase* dan ke robot pembersih kaca. Total *delay* yang didapat dalam pengiriman data secara otomatis Total = 120.318.289 *Second/detik* dan rata-rata *delay* yang didapatkan selama proses mengirimkan data yaitu Rata-Rata = 8594163,5 *Second/detik*.

Tabel 3
Penguujian 3 *Delay* Otomatis

Delay	Total = 69.631.413 <i>Second/detik</i>
	Rata-Rata = 6963141,3 <i>Secound/detik</i>

Pada tabel 3 dilakukan penguujian 3 secara otomatis, pada penguujian ini dilakukan perhitungan untuk durasi waktu selama pengiriman data secara otomatis ke *firebase* dan ke robot pembersih kaca. Total *delay* yang didapat dalam pengiriman data secara otomatis Total = 69.631.413 *Second/detik* dan rata-rata *delay* yang didapatkan selama proses mengirimkan data yaitu Rata-Rata = 6963141,3 *Second/detik*.

Dari grafik ini, dapat disimpulkan bahwa ada variasi yang signifikan dalam total *delay* antara metode otomatis dan manual. Metode otomatis kedua dan ketiga memiliki *delay* yang sangat tinggi dibandingkan dengan metode lainnya, sementara metode manual secara umum cenderung memiliki total *delay* yang sangat rendah atau bahkan tidak ada *delay* sama sekali. Hal ini menunjukkan bahwa metode manual lebih konsisten dalam menghindari *delay* dibandingkan dengan metode otomatis dalam konteks yang ditampilkan pada grafik ini.



Gambar 4. Grafik *Delay*

D. Penguujian *Throughput*

Pada bagian ini dilakukan penguujian menggunakan *wires hark* yang menghasilkan keluaran *response throughput*. Hasil penguujian sistem *throughput* rata-rata dari pengukuran jenis lalu lintas dilakukan penguujian dengan secara otomatis dan Manual menggunakan *wireshark* dan didapatkan hasil jumlah *throughput* sebagai berikut.

Tabel 4
Penguujian 1 *Throughput* Otomatis Pada *Wireshark*

<i>Throughput</i>	
Jumlah Bytes:TimeSpan	2,256Kb/s

Pada tabel 4.7 dilakukan penguujian 1 dengan perintah otomatis, pada penguujian ini dilakukan perhitungan hasil penguujian sistem *throughput* rata-rata dari pengukuran jenis lalu lintas saat mengirim data perintah otomatis ke *firebase* dan ke robot pembersih Jendela. Jumlah *Average bits/s* yang didapat dalam pengiriman data otomatis yaitu 2,256Kb/s.

Measurement	Captured	Displayed	Marked
Packets	188	188 (100.0%)	—
Time span, s	81.422	81.422	—
Average pps	2.3	2.3	—
Average packet size, B	122	122	—
Bytes	22970	22970 (100.0%)	0
Average bytes/s	282	282	—
Average bits/s	2256	2256	—

Gambar 5 Penguujian *Throughput* Pada *Wireshark*

Pada penguujian untuk hasil dari *throughput* yang sudah dihitung dari jumlah *bytes*, *times pan*, dan hasil *bytes*. Pada hasil by tes rata-rata pengukuran dari lalu lintas setiap penguujian berbeda-beda. Pada *throughput* mengukur jumlah data yang dapat ditransfer per satuan waktu, menunjukkan sistem mampu mengelola data perintah dan status secara efisien. Pada gambar 4.7 diatas yaitu hasil penguujian *throughput* langsung menggunakan *wireshark*, jumlah *throughput* saat pengiriman transmisi data, jumlah hasilnya 277 *Averagebits/s*. Hasil nilai *Averagebits/s* dari hasil jumlah *Bytes:Time Span.Throughput* yang stabil memastikan bahwa semua data yang diperlukan untuk mengendalikan robot dapat dikirim dan diproses dengan cepat.

E. Penguujian *Packet Loss*

Pada bagian ini dilakukan penguujian menggunakan *wireshrak* yang menghasilkan rata-rata hilangnya paketdata yang seharusnya dikirimkan melalui jaringan, ketika terjadi *packetloss* beberapa paket data tersebut tidak sampai ketujuan. Pada penguujian ini tingkat kehilangan paket (*packetloss*) berada pada tingkat yang rendah, yang menunjukkan bahwa semua data yang dikirim dari perangkat *Android* ke robot dan sebaliknya berhasil mencapai tujuan tanpa kehilangan yang signifikan.

Berdasarkan data penguujian *QoS* dalam tabel diatas, dapat disimpulkan bahwa kualitas layanan *QoS* secara keseluruhan baik. Berikut yaitu analisis detailnya tabel4.7 penguujian *QoS*.

Tabel 5
Penguujian *QoS*

Penguujian	QoS	Nil ai yang did apat	Indek s	Kategori
Otomatis	<i>Throughput</i>	2,256Kb/s	4	SangatBagus
Penguji an1	<i>Packet Loss</i>	0%	4	SangatBagus
	<i>Delay</i>	81	4	SangatBagus

		ms		
Manual Pengujian 1	Throughput	6,78 2Kb/s	4	SangatBagus
	Packet Loss	0%	4	SangatBagus
	Delay	93 ms	3	Bagus
Otomatis Pengujian2	Throughput	277 Kb/s	4	SangatBagus
	Packet Loss	0%	4	SangatBagus
	Delay	228 ms	4	SangatBagus
Manual Pengujian2	Throughput	20 Kb/s	4	SangatBagus
	Packet Loss	0%	4	SangatBagus
	Delay	250 ms	4	SangatBagus
Otomatis Pengujian3	Throughput	360 Kb/s	4	SangatBagus
	Packet Loss	0%	4	SangatBagus
	Delay	820 ms	1	Jelek
Manual Pengujian3	Throughput	16 Kb/s	4	SangatBagus
	Packet Loss	0%	4	SangatBagus
	Delay	201 ms	3	Bagus

Berikut penjelasan *QoS* berdasarkan tabel 4.7 yang diberikan : Perintah Secara Otomatis Pengujian 1 :

1. *Throughput* : Nilai *throughput* untuk perintah otomatis pada pengujian 1 yaitu 2,256 Kb/s, yang masuk dalam kategori Sangat Bagus. Hal ini menunjukkan bahwa transfer data suara berjalan dengan lancar dan tanpa hambatan.
2. *Packet Loss*: Nilai *packet loss* untuk perintah suara pada pengujian 1 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer,
3. *Delay* : Nilai *delay* untuk pengujian otomatis pada Pengujian 1 adalah 81 ms, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan bahwa waktu yang dibutuhkan data untuk berpindah dari satu titik ke titik lain dalam jaringan relatif singkat

Pengujian Manual 1 :

1. *Throughput*: Nilai *throughput* untuk pengujian manual pada Pengujian 1 adalah 6,782 Kb/s, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan bahwa transfer data selama pengujian otomatis berjalan dengan lancar.
2. *Packet Loss*: Nilai *packet loss* untuk pengujian manual pada Pengujian 1 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer.
3. *Delay*: Nilai *delay* untuk pengujian manual pada Pengujian 1 adalah 93 ms, yang masuk dalam kategori

"Sangat Bagus". Hal ini menunjukkan bahwa waktu yang dibutuhkan data untuk berpindah dari satu titik ke titik lain dalam jaringan relatif singkat.

Secara Otomatis Pengujian 2 :

1. *Throughput* : Nilai *throughput* untuk perintah otomatis pada pengujian 2 yaitu 277 Kb/s, yang masuk dalam kategori Sangat Bagus. Hal ini menunjukkan bahwa transfer data suara berjalan dengan lancar dan tanpa hambatan.
2. *Packet Loss*: Nilai *packet loss* untuk perintah suara pada pengujian 2 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer,
3. *Delay* : Nilai *delay* untuk pengujian otomatis pada Pengujian 2 adalah 228 ms, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan

Pengujian Manual 2 :

1. *Throughput*: Nilai *throughput* untuk pengujian manual pada Pengujian 2 adalah 20 Kb/s, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan bahwa transfer data selama pengujian otomatis berjalan dengan lancar.
2. *Packet Loss*: Nilai *packet loss* untuk pengujian manual pada Pengujian 2 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer.
3. *Delay*: Nilai *delay* untuk pengujian manual pada Pengujian 2 adalah 250 ms, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan bahwa waktu yang dibutuhkan data untuk berpindah dari satu titik ke titik lain dalam jaringan relatif singkat.

Secara Otomatis Pengujian 3 :

1. *Throughput* : Nilai *throughput* untuk perintah otomatis pada pengujian 3 yaitu 360 Kb/s, yang masuk dalam kategori Sangat Bagus. Hal ini menunjukkan bahwa transfer data suara berjalan dengan lancar dan tanpa hambatan.
2. *Packet Loss*: Nilai *packet loss* untuk perintah suara pada pengujian 3 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer,
3. *Delay*: Nilai *delay* untuk pengujian otomatis pada Pengujian 3 adalah 820 ms, yang masuk dalam kategori "Jelek". Hal ini menunjukkan adanya penundaan yang signifikan pada pengujian ini, yang mungkin disebabkan oleh faktor eksternal seperti kemacetan jaringan atau gangguan dari luar

Pengujian Manual 3 :

1. *Throughput*: Nilai *throughput* untuk pengujian manual pada Pengujian 3 adalah 16 Kb/s, yang masuk dalam kategori "Sangat Bagus". Hal ini menunjukkan bahwa transfer data selama pengujian otomatis berjalan dengan lancar.
2. *Packet Loss*: Nilai *packet loss* untuk pengujian manual pada Pengujian 3 adalah 0%, yang merupakan nilai terbaik. Hal ini menunjukkan bahwa tidak ada paket data yang hilang selama transfer.
3. *Delay*: Nilai *delay* untuk pengujian manual pada

Pengujian 3 adalah 201 ms, yang masuk dalam kategori “Bagus”. Hal ini menunjukkan bahwa waktu yang dibutuhkan data untuk berpindah dari satu titik ke titik lain dalam jaringan relatif singkat.

IV. KESIMPULAN

Adapun yang dapat disimpulkan setelah melakukan penelitian mengenai sistem robot pembersih kaca gedung bertingkat dengan kendali otomatis dan manual berbasis Android dapat meningkatkan kenyamanan dan kemudahan pengguna. Secara keseluruhan, berdasarkan data yang dikumpulkan, pengujian manual menunjukkan performa yang lebih baik dalam hal delay dan throughput, namun dengan sedikit packet loss pada beberapa pengujian. Pengujian otomatis, meskipun umumnya stabil, memperlihatkan fluktuasi besar dalam delay pada beberapa skenario.

REFERENSI

- [1] Budi Utami Fahnun, & Reza Pangestu. (2022). Sistem Remote Kontrol Pada Robot Mobil Via Web Berbasis Raspberry Pi. *Jurnal Ilmiah Teknik*, 1(2), 143–153. <https://doi.org/10.56127/juit.v1i2.204>
- [2] Chitra Lekha, B. “Rancang Bangun Robot Pembersih Kaca Otomatis Berbasis Mikrokontroler ARM STM32 Dengan Sensor Proximity,” *Indones. J. Eng. Technol.*, vol. 2, no. 1, pp. 24–29, 2019, [Online]. Available:
- [3] Eko, I., & Rifky, M. F. (2015). *Berbasis Arduino Dan Interfacing*. 6(1), 28–37.
- [4] Herry Sufyan Hadi, Arief Abdurrakhman, & Bambang Sampurno. (2021). Rancang Bangun Sistem Monitoring Cairan Pembersih Pada Robot Pembersih Kaca Berbasis Mikrokontroler ATmega 8535. *J-Eltrik*, 1(1), 12. <https://doi.org/10.30649/je.v1i1.12>
- [5] Husna, R Nasir, M and H. T. Hidayat, “Rancang Bangun Prototype Jemuran Berbasis Iot (Internet Of Things),” *J. Teknol. Rekayasa Inf. dan Komput.*, vol. 3, no. 2, pp. 2581–2882, 2020.
- [6] Ilham Firman Maulana. (2020). Penerapan Firebase Realtime Database pada Aplikasi E-Tilang Smartphone berbasis Mobile Android. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(5), 854–863. <https://doi.org/10.29207/resti.v4i5.2232>
- [7] M. Nasir and Dkk, “Sistem Monitoring Absensi Perkuliahan dengan Menggunakan RFID Berbasis Raspberry Pi,” *Proceeding Semin. Nas. Politek. Negeri Lhokseumawe*, vol. 3, no. 1, 2019.
- [8] Normah, Rifai, B., Vambudi, S., & Maulana, R. (2022). Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis SMOTE. *Jurnal Teknik Komputer AMIK BSI*, 8(2), 174–180. <https://doi.org/10.31294/jtk.v4i2>
- [9] Sampurno, B., Abdurrakhman, A., Hadi, H. S., Joestiono, H., & Saudzah, A. F. (2019). Distance Control System to Increase Precision And Stability of Glass Cleaning Robot. *IPTEK Journal of Proceedings Series*, 0(3), 1. <https://doi.org/10.12962/j23546026.v2019i3.5831>
- [10] Umam, K., Haryanto, H., & Alfita, R. (2019). Rancang Bangun Robot Pembersih Kaca Otomatis Berbasis Mikrokontroler ARM STM32 Dengan Sensor Proximity. *Indonesian Journal of Engineering and Technology (INAJET)*, 2(1), 24–29. <https://doi.org/10.26740/inajet.v2n1.p24-29>