

Pengembangan Sistem Antrian Digital Berbasis Website untuk Layanan Multi-Kelompok

Achmad Zidan Ramdani¹, Agus Prihanto²

^{1,2} S1 Teknik Informatika, Universitas Negeri Surabaya

achmad.21040@mhs.unesa.ac.id

agusprihanto@unesa.ac.id

Abstrak— Sistem antrian konvensional sering kali mengalami kendala signifikan dalam mengelola layanan multi-kelompok, seperti ketimpangan distribusi beban kerja antar loket dan ketidakpastian waktu tunggu pelanggan. Kondisi ini sering memicu penumpukan antrian yang tidak terurai serta ketidakadilan distribusi pelayanan yang berdampak negatif pada kepuasan pelanggan. Penelitian ini bertujuan merancang dan membangun sebuah sistem antrian digital berbasis website untuk menangani masalah tersebut dengan menerapkan model *Multi-Queue*. Pengembangan sistem dilakukan menggunakan framework Laravel serta integrasi teknologi *WebSocket* (*Laravel Reverb*) untuk memfasilitasi komunikasi data secara *real-time*. Model pengembangan yang diterapkan adalah *Waterfall* yang meliputi tahapan analisis, desain, implementasi, hingga pengujian. Evaluasi kualitas sistem dilakukan secara komprehensif melalui serangkaian pengujian kuantitatif. Pengujian fungsionalitas menggunakan metode *black-box testing* memastikan seluruh fitur berjalan sesuai rancangan. Evaluasi keadilan (*Fairness*) menggunakan metrik *Jain's Fairness Index* (JFI) menghasilkan nilai sempurna 1.0 pada skenario beban seimbang, serta membuktikan keberhasilan isolasi antrian di mana antrian layanan yang padat tidak menghambat antrian layanan yang sepi. Dari sisi respon sistem, pengujian mencatat rata-rata *Response Time* (TTFB) yang sangat responsif, yaitu 35-56 ms, jauh melampaui standar rekomendasi Google (< 800 ms). Audit kualitas teknis frontend menggunakan *Google Lighthouse* juga memberikan skor 100 pada aspek Performance. Penelitian ini menyimpulkan bahwa sistem antrian *Multi-Queue* berbasis web yang dikembangkan tidak hanya berhasil secara fungsional dalam mendigitalkan proses antrian, tetapi juga terbukti adil, responsif, dan memiliki standar kualitas kode yang optimal.

Kata Kunci— Sistem Antrian, *Multi-Queue*, *Response Time*, *Google Lighthouse*, *Websocket*.

I. PENDAHULUAN

Proses pelayanan pada instansi publik seperti rumah sakit, klinik, atau bank sangat bergantung pada efisiensi sistem antrian yang diterapkan. Ketidakseimbangan antara permintaan layanan dengan kapasitas fasilitas sering kali menyebabkan penumpukan antrian yang menghambat produktivitas. Permasalahan menjadi lebih kompleks pada instansi yang menerapkan layanan multi-kelompok (*multi-group services*), seperti bank yang memisahkan layanan *Teller* dan *Customer Service*. Sering terjadi ketimpangan distribusi di mana satu kelompok layanan mengalami kemacetan parah sementara kelompok lain kosong, yang berdampak langsung pada penurunan kepuasan pelanggan [1].

Modernisasi pengelolaan antrian melalui aplikasi berbasis website menjadi solusi strategis untuk mengatasi permasalahan

tersebut. Penerapan teknologi ini terbukti mampu memangkas waktu tunggu non-produktif dan memberikan transparansi informasi yang signifikan bagi pengguna [2]. Namun, sekadar mendigitalkan antrian tidak cukup tanpa strategi pengelolaan alur yang tepat. Tantangan utama dalam sistem multi-layanan adalah menjaga keadilan (*fairness*) agar pelanggan pada layanan yang cepat tidak terhambat oleh layanan yang lambat. Oleh karena itu, penerapan model *Multi-Queue* (Antrian Majemuk) diperlukan untuk mengisolasi antrian antar-kelompok layanan secara independen.

Berbagai strategi teknis telah diterapkan untuk mengoptimalkan sistem antrian digital. Penggunaan metodologi pengembangan seperti Scrum sering dipilih untuk memastikan manajemen pengembangan perangkat lunak yang terstruktur dan adaptif [3]. Dari sisi algoritma, pendekatan *Multilevel Queue* telah diimplementasikan pada platform *mobile* guna meningkatkan efisiensi pendaftaran [4]. Selain itu, simulasi menggunakan perangkat lunak seperti Arena juga kerap digunakan untuk memodelkan sistem antrian dan mencari konfigurasi ideal tanpa risiko gangguan operasional [5]. Terdapat pula pengembangan sistem antrian berbasis web yang memanfaatkan penyimpanan lokal (*localStorage*) untuk efisiensi sumber daya server, meskipun hal ini memiliki keterbatasan dalam sinkronisasi data antar perangkat secara *real-time* [6].

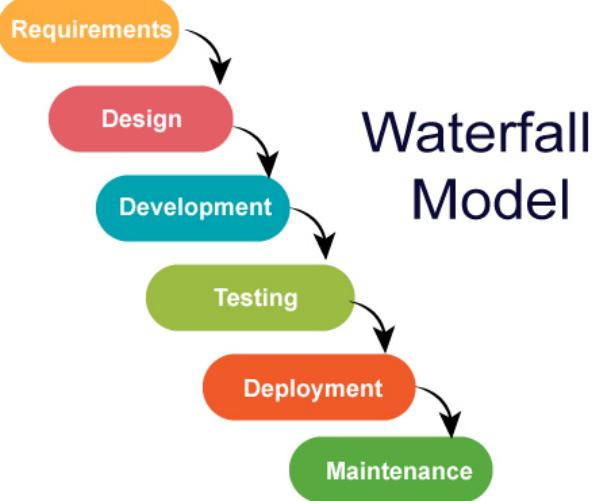
Melengkapi pendekatan-pendekatan tersebut, penelitian ini mengembangkan sebuah sistem antrian berbasis website yang mengintegrasikan model *Multi-Queue* dengan teknologi *WebSocket* (*Laravel Reverb*). Integrasi ini bertujuan untuk memastikan komunikasi data yang stabil dan *real-time* antar perangkat, mengatasi keterbatasan sinkronisasi pada penyimpanan lokal, serta menyediakan solusi aplikasi siap pakai yang berbeda dengan pendekatan simulasi semata. Fokus utama pengembangan ditekankan pada performa teknis dan pengukuran kualitas sistem secara kuantitatif.

Penelitian ini bertujuan untuk merancang bangun sistem antrian digital dengan model *Multi-Queue* dan melakukan evaluasi mendalam terhadap kualitas perangkat lunak. Evaluasi tidak hanya terbatas pada fungsionalitas, tetapi juga mencakup pengukuran tingkat keadilan distribusi layanan (*fairness*) menggunakan *Jain's Fairness Index*, performa waktu tanggap (*response time*), serta kualitas teknis kode menggunakan standar *Google Lighthouse*.

II. METODE PENELITIAN

Penelitian ini menerapkan model pengembangan perangkat lunak *Waterfall* yang bersifat sistematis dan sekuensial.

Pendekatan ini dipilih karena kebutuhan sistem untuk layanan antrian multi-kelompok telah terdefinisi dengan jelas. Tahapan pengembangan meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian.



Gbr. 1 Tahapan Pengembangan Perangkat Lunak Model Waterfall.

A. Analisis Kebutuhan

Tahap ini difokuskan pada identifikasi kebutuhan fungsional untuk menangani layanan multi-kelompok (*multi-group services*). Berdasarkan observasi, sistem dirancang untuk memfasilitasi tiga aktor utama dengan spesifikasi kebutuhan sebagai berikut:

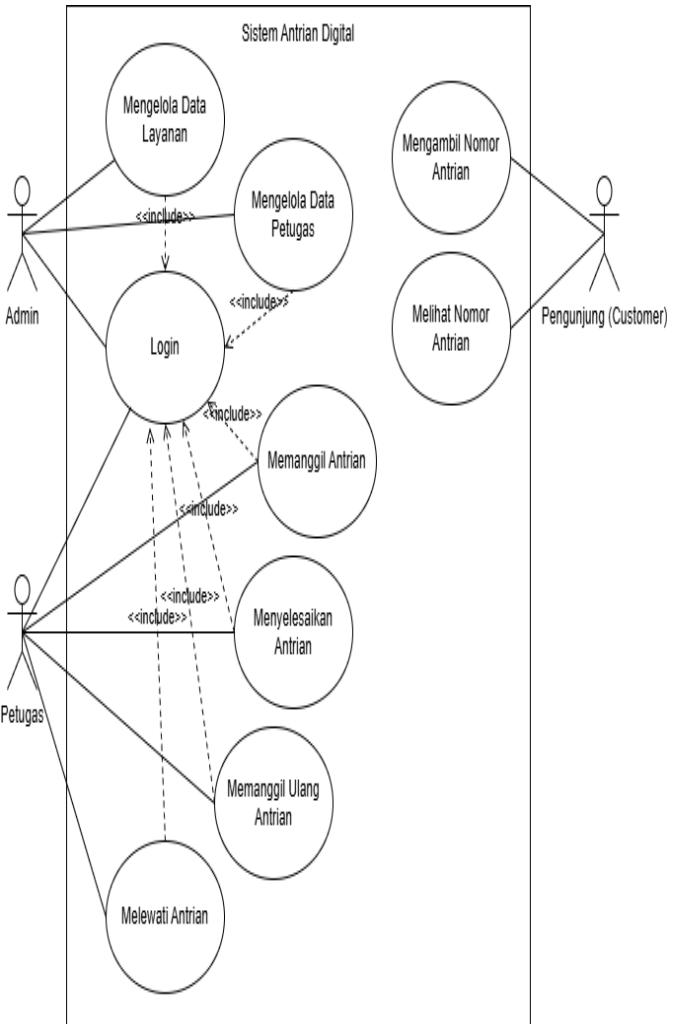
1. Pengunjung dapat mengambil nomor antrian berdasarkan layanan dan memantau status di monitor publik.
2. Petugas dapat memanggil antrian sesuai urutan layanan, memanggil ulang, dan memperbarui status antrian.
3. Admin mampu mengelola data layanan, data petugas, dan memantau antrian secara keseluruhan.

B. Perancangan Sistem

Tahap perancangan menerjemahkan hasil analisis kebutuhan menjadi spesifikasi teknis yang siap diimplementasikan. Perancangan sistem ini meliputi pemodelan interaksi, alur proses, struktur data, dan arsitektur komunikasi.

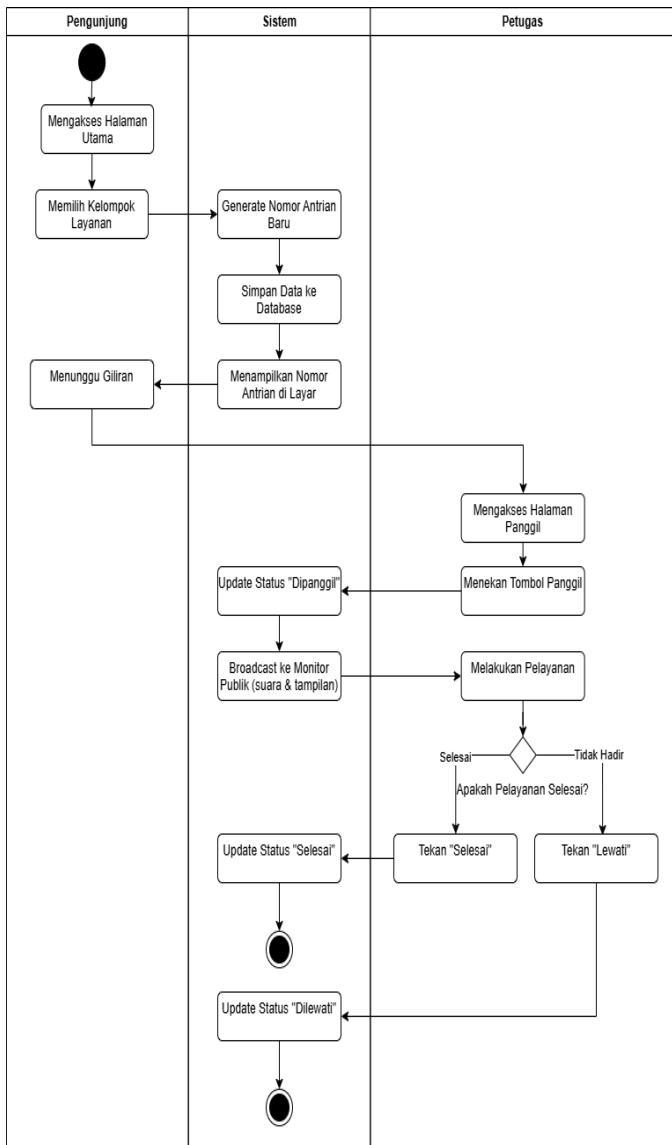
1) *Perancangan Interaksi (Use Case):* Interaksi sistem dirancang melibatkan tiga aktor dengan hak akses spesifik.

- a) Pengunjung: Mengambil tiket antrian dan memantau status di monitor.
- b) Petugas: Memanggil antrian, melewati antrian (*skip*), dan menyelesaikan layanan.
- c) Admin: Mengelola data master (layanan dan petugas) serta melihat *dashboard* statistik.



Gbr. 2 Use Case Diagram Sistem Antrian Digital

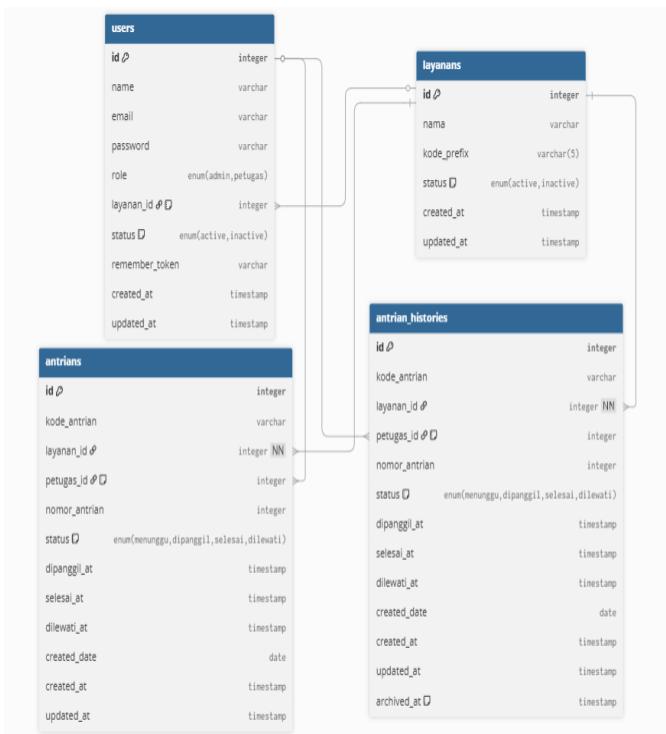
2) *Perancangan Alur Proses (Activity Diagram):* Alur kerja operasional dirancang untuk menjamin urutan *First-In-First-Out* (FIFO). Proses dimulai saat pengunjung memilih jenis layanan, sistem menghasilkan nomor antrian, hingga petugas memanggil nomor tersebut yang secara otomatis memperbarui tampilan di monitor publik.



Gbr. 3 Diagram Aktivitas Proses Operasional Antrian

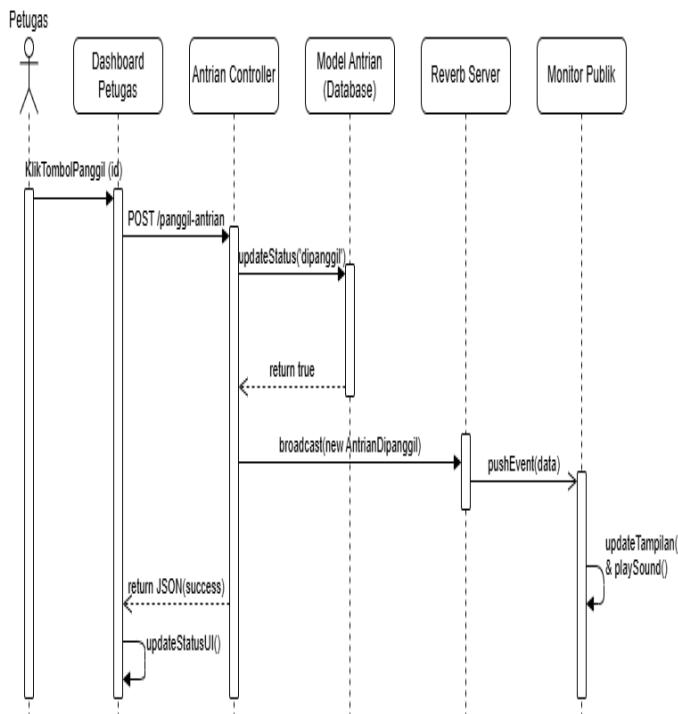
3) *Perancangan Basis Data (ERD)*: Struktur penyimpanan data menggunakan basis data relasional yang terdiri dari empat entitas utama untuk menjaga integritas data:

- users: Menyimpan data autentikasi dan peran (Admin/Petugas).
- layanans: Menyimpan kategori layanan untuk memisahkan antrian.
- antrians: Menyimpan data transaksi antrian yang sedang berjalan.
- antrian_histories: Menyimpan riwayat antrian yang telah selesai untuk keperluan arsip.



Gbr. 4 Desain Entity Relationship Diagram (ERD)

4) *Perancangan Arsitektur Real-Time (Sequence Diagram)*: Untuk kebutuhan respon cepat (*low latency*), arsitektur komunikasi dirancang menggunakan teknologi *WebSocket*. Saat petugas menekan tombol panggil, server memicu *event* yang disiarkan (*broadcast*) secara instan ke seluruh klien (Monitor Publik) tanpa memerlukan refresh halaman manual.



Gbr. 5 Diagram Sekuens Arsitektur Komunikasi Real-Time

C. Implementasi

Sistem diimplementasikan menggunakan *framework* Laravel (PHP) sebagai *backend*. Fitur kunci pada tahap ini adalah integrasi Laravel Reverb sebagai server WebSocket. Teknologi ini memungkinkan server untuk melakukan *broadcast event* secara instan ke seluruh klien tanpa *refresh* halaman [7]. Logika *Multi-Queue* diterapkan pada *controller* untuk memastikan isolasi antrian antar-kelompok layanan.

D. Metode Pengujian

Pengujian dilakukan untuk memvalidasi kualitas perangkat lunak melalui empat metode evaluasi kuantitatif:

1. Pengujian fungsionalitas menggunakan metode *Black Box Testing* untuk memastikan setiap fitur input-output berjalan sesuai rancangan.
2. Evaluasi keadilan (*Fairness*) untuk mengukur tingkat pemerataan layanan antar kelompok menggunakan *Jain's Fairness Index* (JFI) yang diperkenalkan oleh Jain et al. [8]. Nilai JFI dihitung menggunakan persamaan (1):

$$J(x_1, x_2, \dots, x_n) = \frac{(\sum_{i=1}^n x_i)^2}{n \cdot \sum_{i=1}^n x_i^2} \quad (1)$$

3. Evaluasi performa (*response time*) untuk mengukur kecepatan respon server menggunakan parameter *Time to First Byte* (TTFB) dengan target standar optimasi web, nilai TTFB yang baik harus berada di bawah 800 ms [9].
4. Kualitas teknis, yaitu audit otomatis menggunakan *Google Lighthouse* untuk mengukur skor

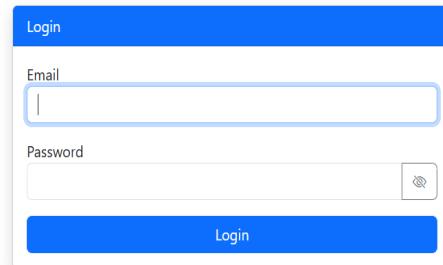
Performance, Accessibility, dan Best Practices sesuai standar pengembangan web modern [10].

III. HASIL DAN PEMBAHASAN

A. Tinjauan Antarmuka Sistem

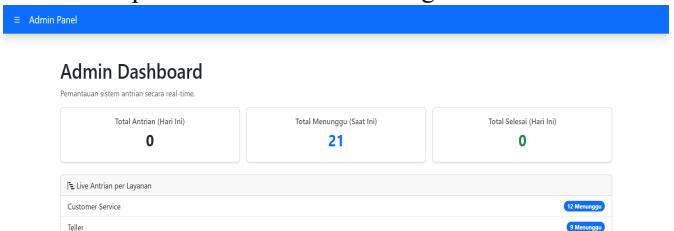
Sistem antrian digital berhasil dikembangkan dengan antarmuka berbasis web yang responsif. Tampilan utama sistem dibagi berdasarkan peran pengguna:

- 1) *Halaman Login*: Berfungsi sebagai gerbang keamanan untuk memverifikasi hak akses Admin dan Petugas sebelum masuk ke sistem.



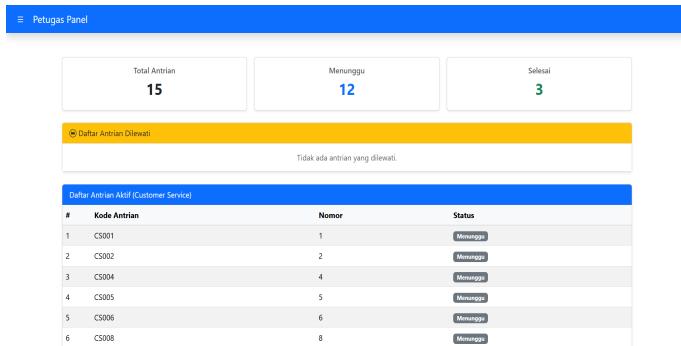
Gbr. 6 Antarmuka Halaman Login

- 2) *Dashboard Admin*: Menyajikan statistik harian dan menu pengelolaan data master (layanan dan petugas) serta pemantauan antrian secara global.



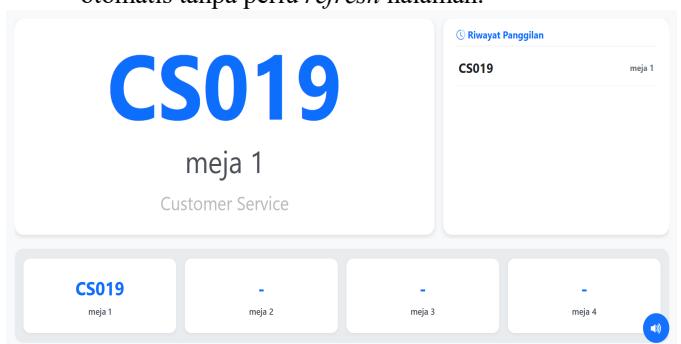
Gbr. 7 Dashboard Admin

- 3) *Dashboard Petugas*: Antarmuka untuk informasi antrian sesuai kelompok layanan yang ditugaskan.



Gbr. 8 Dashboard Petugas

- 4) *Monitor Publik*: Tampilan layar di ruang tunggu yang memperbarui status pemanggilan secara *real-time* dan otomatis tanpa perlu *refresh* halaman.



Gbr. 9 Tampilan Monitor Publik

B. Pengujian Fungsionalitas

Berdasarkan pengujian *Black Box* terhadap fitur-fitur krusial seperti *Login*, *Ambil Antrian*, dan *Panggil Antrian*, seluruh skenario pengujian memberikan hasil sesuai harapan. Hasil uji fungsional disajikan pada Tabel 1.

Tabel I
 Hasil Uji Fungsionalitas

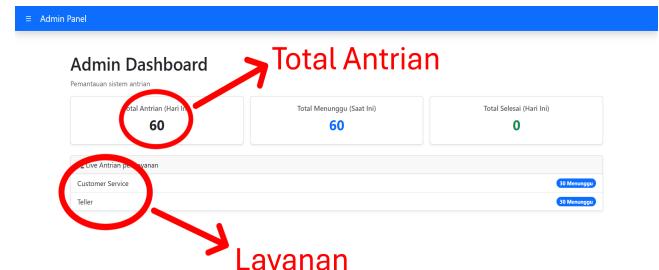
No	Skenario Uji	Output yang Diharapkan	Hasil
1.	Login Admin	Admin berhasil masuk ke dashboard admin	Berhasil
2.	Login Petugas	Petugas berhasil masuk ke dashboard petugas	Berhasil
3.	Login gagal	Sistem menampilkan pesan error login	Berhasil
4.	Tambah layanan	Data layanan tersimpan di database dan tampil di daftar layanan	Berhasil
5.	Hapus layanan	Data layanan terhapus dari database	Berhasil
6.	Edit Layanan	Data layanan di database dan tabel berhasil diperbarui.	Berhasil

No	Skenario Uji	Output yang Diharapkan	Hasil
7.	Tambah petugas	Akun petugas tersimpan di database	Berhasil
8.	Edit Petugas	Data petugas di database berhasil diperbarui	Berhasil
9.	Ambil Antrian (Customer)	Customer mendapat nomor (misal A001) dan data tersimpan di tabel antrians.	Berhasil
10.	Panggil Antrian (Petugas)	Sistem mengambil antrian, status antrian berubah menjadi dipanggil dan tampilan Monitor Publik menampilkan nomor antrian.	Berhasil
11.	Selesaikan Antrian	Status antrian berubah menjadi Selesai.	Berhasil
12.	Lewati Antrian	Status antrian berubah menjadi dilewati.	Berhasil
13.	Panggil Ulang Antrian	Status antrian berubah menjadi dipanggil dan tampilan Monitor Publik menampilkan nomor antrian.	Berhasil

C. Evaluasi Keadilan (Fairness)

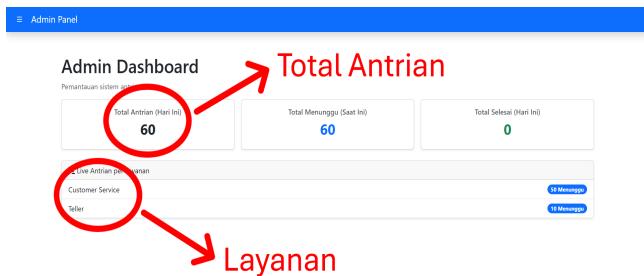
Evaluasi kinerja algoritma *Multi-Queue* dilakukan melalui simulasi beberapa kondisi beban kerja:

1. Skenario beban seimbang, pada kondisi ini beban antrian dan jumlah petugas diatur seimbang antar layanan, sistem menghasilkan nilai JFI sempurna yaitu 1.0.



Gbr. 10 Konfigurasi Dashboard Admin untuk Skenario Beban Seimbang

2. Skenario beban tidak seimbang, pada kondisi ini beban antrian dibuat timpang (salah satu layanan sangat padat), sistem tetap berjalan stabil. Meskipun nilai JFI turun menjadi 0.69, hal ini membuktikan keberhasilan isolasi antrian. Pelanggan pada layanan yang sepi tetap terlayani dengan cepat tanpa terhambat oleh antrian panjang di layanan lain, sesuai tujuan model *Multi-Queue*.



Gbr. 11 Konfigurasi Dashboard Admin untuk Skenario Beban Tidak Seimbang

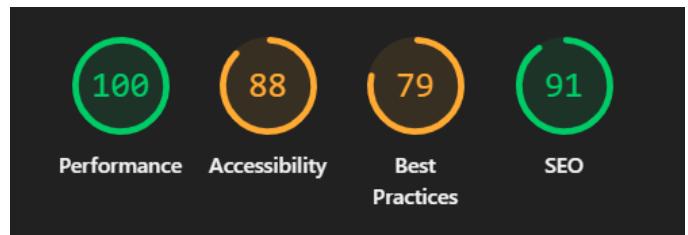
D. Evaluasi Keadilan (Fairness)

- 1) *Response Time (TTFB)*: Pengujian terhadap aksi transaksional utama menunjukkan rata-rata waktu respon server yang sangat cepat, berkisar antara 35 ms hingga 56 ms. Hasil ini jauh melampaui standar performa web yang baik (< 800 ms) [9].

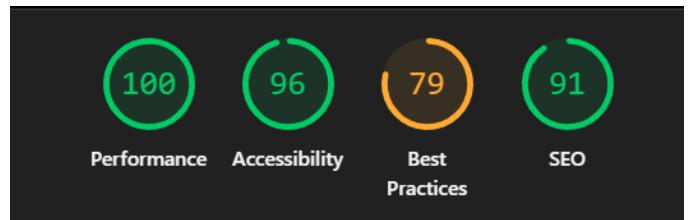
Tabel II
 Hasil Pengujian Waktu Tanggap Server

No	Skenario Uji	Aksi yang Diukur	Target NFR	Hasil Rata-rata (ms)
1.	Panggil Antrian	Waktu <i>request POST</i> (petugas memanggil antrian) hingga server merespons.	< 800 ms	56.75 ms
2.	Selesaikan Antrian	Waktu <i>request POST</i> (petugas menyelesaikan antrian) hingga server merespons.	< 800 ms	55.04 ms
3.	Ambil Antrian Baru	Waktu <i>request POST</i> (customer membuat antrian) hingga server merespons.	< 800 ms	35.23 ms

- 2) *Kualitas Kode (Google Lighthouse)*: Audit kualitas teknis dilakukan pada dua antarmuka utama. Halaman Monitor Publik mendapatkan skor 100 untuk aspek *Performance*, memvalidasi efisiensi implementasi WebSocket [7]. Sementara itu, Halaman Pengambilan Antrian mencatatkan skor Aksesibilitas 96, menandakan desain yang ramah pengguna sesuai standar web [10].



Gbr. 12 Skor Audit Performa Halaman Monitor Publik



Gbr. 13 Skor Audit Aksesibilitas Halaman Pengambilan Antrian

IV. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem antrian digital berbasis *website* yang mampu menangani kompleksitas layanan multi-kelompok (*multi-group services*) secara efektif. Penerapan model *Multi-Queue* terbukti berhasil mengisolasi antrian antar-layanan, sehingga kepadatan pada satu jenis layanan tidak menghambat kinerja layanan lainnya. Berdasarkan hasil pengujian *fairness* menggunakan *Jain's Fairness Index (JFI)*, sistem mencapai nilai indeks 1.0 pada kondisi beban seimbang, yang mengindikasikan tingkat keadilan sempurna dalam distribusi pelayanan.

Dari aspek performa teknis, integrasi teknologi *WebSocket (Laravel Reverb)* berhasil memfasilitasi pembaruan data secara *real-time* dengan latensi yang sangat rendah. Pengujian waktu tanggap (*response time*) server mencatatkan rata-rata 35 ms hingga 56 ms, jauh melampaui standar performa web yang direkomendasikan (< 800 ms). Selain itu, skor audit *Google Lighthouse* yang mencapai nilai sempurna (100) pada aspek kinerja antarmuka memvalidasi bahwa sistem ini tidak hanya fungsional, tetapi juga sangat responsif dan ringan bagi pengguna akhir.

V. SARAN

Berdasarkan hasil implementasi dan evaluasi, terdapat beberapa saran untuk pengembangan sistem di masa mendatang:

1. Sistem dapat dikembangkan lebih lanjut dengan integrasi mesin tiket kios (Kiosk) yang memungkinkan pengguna mencetak tiket antrian secara fisik..
2. Fitur laporan admin dapat dikembangkan menjadi analisis tren mingguan atau bulanan yang menyajikan grafik jam sibuk serta rekomendasi otomatis terkait penambahan atau pengurangan jumlah petugas aktif di layanan tertentu.

UCAPAN TERIMA KASIH

Segala puji syukur kepada Tuhan Yang Maha Esa atas rahmat dan hidayahnya sehingga artikel ini dapat terselesaikan. Serta ucapan terima kasih kepada orang tua, dosen pembimbing, serta teman-teman yang senantiasa memberikan bantuan dan dukungannya.

REFERENSI

- [1] J. Prasetya, D. P. Prayogi, A. Haris, dan S. Badri, "Analisis Efektivitas Sistem Pelayanan Pasien Rawat Jalan BPJS dengan Pendekatan Model Antrian," *Progress Conference*, vol. 4, no. 1, hal. 491–502, 2021.
- [2] R. Kurniati dan J. Jaroji, "Sistem Antrian Multi Channel Rumah Sakit Berbasis Web," *Jurnal Inovtek Polbeng*, vol. 3, no. 2, hal. 140-146, 2018.
- [3] G. Germecca, N. A. Wardhani, dan M. M. Dewi, "Implementasi Sistem Informasi Antrian Berbasis Website dengan Metodologi Scrum," *JOISM*, vol. 5, no. 2, hal. 233-238, 2024.
- [4] A. Priambudi, Y. F. Wijaya, dan U. Darusalam, "Aplikasi Mobile Pendaftaran Pasien Klinik Berbasis Algoritma Multilevel Queue dan FIFO," *Jurasik*, vol. 9, no. 1, 2024.
- [5] N. W. Aulia dan Hidayat, "Usulan Perbaikan Sistem Pelayanan Pada Antrian Loket Pembayaran Pajak Kendaraan Bermotor Menggunakan Simulasi Software Arena," *JUSTI*, vol. 5, no. 3, 2024.
- [6] A. Dinanti, J. R. Gultom, I. Z. Muchtar, dan A. B. Hutabarat, "Perancangan dan Implementasi Website Sistem Antrian Bank Berbasis Queue," *Teknik: Jurnal Ilmu Teknik dan Informatika*, vol. 5, no. 2, 2025.
- [7] Laravel, "Laravel Reverb Documentation," 2025. [Online]. Tersedia: <https://laravel.com/docs/reverb>.
- [8] R. Jain, D. M. Chiu, dan W. R. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems," *DEC Research Report TR-301*, 1984.
- [9] J. Wagner dan B. Pollard, "Time to First Byte (TTFB)," web.dev, 2023. [Online]. Tersedia: <https://web.dev/articles/ttfb>.
- [10] Google Developers, "Lighthouse Overview," Chrome for Developers, 2024. [Online]. Tersedia: <https://developer.chrome.com/docs/lighthouse/overview>.