

Penerapan Algoritma Dynamic Priority Scheduling Pada Sistem Pengaduan Tumpukan Sampah Berbasis Web

Nurfatzma Ayu^{*1}, Statiswaty², Sutardi³

Jurusan Informatika, Fakultas Teknik, Universitas Halu Oleo^{1,2,3}

*e-mail: nurfatzzmaayu@gmail.com¹

ABSTRAK

Permasalahan pengelolaan sampah di wilayah perkotaan seperti di Kelurahan Kambu, Kota Kendari semakin kompleks seiring dengan peningkatan volume sampah yang signifikan. Masyarakat sering kali kesulitan dalam menyampaikan laporan tumpukan sampah, sementara pihak Dinas Lingkungan Hidup menghadapi kendala dalam melakukan penanganan secara cepat karena terbatasnya armada pengangkut dan belum tersedianya sistem pengaduan yang terstruktur. Oleh karena itu, diperlukan sebuah sistem pengaduan yang mampu memproses laporan secara efisien, transparan, dan berbasis skala prioritas.

Penelitian ini bertujuan untuk mengembangkan sistem berbasis web yang mengelola pengaduan tumpukan sampah menggunakan Algoritma Dynamic Priority Scheduling untuk menentukan urutan penanganan berdasarkan tingkat prioritas. Sistem memprioritaskan laporan berdasarkan empat parameter, yaitu dampak lingkungan (bobot 4), volume sampah (bobot 3), waktu pengaduan (bobot 2), dan jarak lokasi (bobot 1) yang dihitung menggunakan formula Haversine. Data penelitian mencakup informasi lokasi (koordinat geografis), volume sampah, dampak lingkunga dengan 10 sub-kriteria, dan waktu pelaporan. Sistem dibangun menggunakan Laravel, PHP, MySQL, dan LeafletJS, Serta dirancang untuk masyarakat sebagai pelaporan dan admin Dinas Lingkungan Hidup sebagai dirancang untuk masyarakat sebagai pelapor dan admin Dinas Lingkungan Hidup sebagai pengelola pengaduan.

Pengujian sistem dilakukan melalui evaluasi Algoritma Dynamic Priority Scheduling dengan skenario implementasi perhitungan manual dan User Acceptance Testing (UAT) yang melibatkan 5 responden masyarakat dan 5 responden admin/petugas. Hasil pengujian menunjukkan sistem mampu menghasilkan urutan prioritas yang logis dan objektif dengan tingkat penerimaan sangat baik, yaitu 88.8% untuk masyarakat dan 93.6% untuk admin/petugas. Sistem ini di harapkan dapat meningkatkan efektivitas penanganan pengaduan, transparansi pelayanan publik, partisipasi aktif masyarakat dan responsivitas Dinas Lingkungan Hidup Kota Kendari dalam mengelola tumpukan sampah di Kelurahan Kambu secara berkelanjutan.

Kata kunci : *Dynamic Priority Scheduling, Pengaduan Sampah, Web, Prioritas Dinamis.*

ABSTRACT

The issue of waste management in urban areas, such as Kambu Subdistrict in Kendari City, has become increasingly complex due to the significant rise in waste volume. The community often faces difficulties in reporting waste pile-ups, while the Environmental Agency encounters obstacles in responding promptly because of limited transportation fleets and the absence of a structured complaint system. Therefore, a system is needed that can process waste reports efficiently, transparently, and based on a priority scale.

This research aims to develop a web-based system that manages waste pile-up complaints using the Dynamic Priority Scheduling Algorithm to determine the handling order based on priority levels. The system prioritizes reports according to four parameters: environmental impact (weight 4), waste volume (weight 3), report time (weight 2), and location distance (weight 1), calculated using the Haversine formula. The research data include geographical coordinates, waste volume, environmental impact with 10 sub-criteria, and report time. The system is developed using Laravel, PHP, MySQL, and LeafletJS, and is designed for two user roles the public as reporters and the Environmental Agency as complaint managers. System testing was carried out through the evaluation of the Dynamic Priority Scheduling Algorithm using manual calculation implementation scenarios and User Acceptance Testing (UAT) involving 5 public

respondents and 5 admin/staff respondents. The test results showed that the system was able to produce a logical and objective priority order with a very good level of acceptance 88.8% from the public and 93.6% from administrators/staff. This system is expected to improve the effectiveness of complaint handling, public service transparency, active community participation, and the responsiveness of the Kendari City Environmental Agency in managing waste pile-ups in Kambu Subdistrict sustainably.

Keywords : *Dynamic Priority Scheduling, Garbage Complaints, Web, Dynamic Priority.*

PENDAHULUAN

Sampah adalah salah satu masalah yang umumnya dihadapi kota-kota besar Indonesia, masalah ini muncul dengan tingginya tingkat urbanisasi yang pesat, volume sampah di perkotaan terus meningkat (Ferdiansa dkk., 2022). Sampah merupakan sisa hasil dari berbagai kegiatan manusia yang jumlahnya terus bertambah seiring dengan peningkatan populasi, ragam aktivitas, dan tingkat konsumsi. Namun, banyak masyarakat belum memahami pentingnya menjaga kebersihan lingkungan dan sering membuang sampah sembarangan di tempat-tempat seperti jalan raya, parit, dan bantaran kali. Jumlah penduduk dan tingkat pendapatan masyarakat juga mempengaruhi timbulnya sampah yang dihasilkan. Oleh karena itu, peningkatan jumlah sampah harus diimbangi dengan sistem pengelolaan yang baik untuk menciptakan lingkungan yang bersih dan nyaman (Suhendri Ergie, 2024).

Berdasarkan data Kementerian Lingkungan Hidup pada tahun 2024 dari 317 Kabupaten/Kota di Indonesia, total sampah yang dihasilkan mencapai 34,2 juta ton per tahun. Dari jumlah tersebut, hanya sekitar 4,5 juta ton (13,24%) yang berhasil dikurangi, dan 15,9 juta ton (46,51%) ditangani melalui proses pengelolaan. Artinya, hanya sekitar 59,74% sampah yang terkelola, sementara sisanya sebanyak 13,7 juta ton (40,26%) masih belum tertangani. Angka ini menunjukkan bahwa hampir setengah dari total sampah di Indonesia belum mendapatkan penanganan yang memadai.

Dinas Lingkungan Hidup Kota Kendari mencatat produksi sampah harian mencapai 243 ton pada awal tahun 2025 yang berasal dari berbagai sumber, mulai dari rumah tangga hingga tempat usaha. Jumlah tersebut menjadi tantangan serius bagi pemerintah daerah dalam menjaga kebersihan dan kelestarian lingkungan. Di Kota Kendari, khususnya di Kelurahan Kambu, permasalahan sampah menjadi salah satu persoalan lingkungan yang paling sering dikeluhkan masyarakat karena menimbulkan bau tidak sedap dan mengganggu kesehatan lingkungan sekitar.

Berdasarkan hasil wawancara dengan warga yang tinggal di sekitar area Tempat Pembuangan Sementara (TPS) di Kelurahan Kambu, diketahui bahwa sebagian besar dari mereka sering melihat tumpukan sampah yang tidak diangkut selama beberapa hari. Banyak warga yang mengeluhkan kondisi tersebut namun tidak mengetahui ke mana harus menyampaikan pengaduan. Sementara itu, berdasarkan hasil wawancara dengan Kepala Seksi Bidang Pengangkutan Sampah Dinas Lingkungan Hidup Kota Kendari, ditemukan bahwa pengangkutan sampah dilakukan setiap hari mulai pukul 04.00 hingga 06.00 pagi. Namun, jumlah armada pengangkut sampah mengalami penurunan signifikan dari 60 unit menjadi hanya 39 unit. Selain itu, mayoritas sopir armada kini sudah berusia lanjut, yang berdampak pada kecepatan dan efektivitas pengangkutan.

Pengaduan masyarakat terkait tumpukan sampah masih dilakukan secara manual melalui sambungan telepon, yang sering kali tidak terdokumentasi dengan baik dan menyulitkan proses tindak lanjut serta pemantauan laporan secara sistematis. Untuk mengatasi permasalahan tersebut, diperlukan sistem pengaduan yang tidak hanya menampung laporan masyarakat secara daring, tetapi juga mampu memprioritaskan penanganannya secara objektif. Sistem ini diharapkan menjadi media pelaporan yang efektif. Dengan adanya sistem ini, petugas kebersihan dapat mengetahui lokasi yang membutuhkan penanganan segera, sementara masyarakat dapat menyampaikan laporan dengan mudah tanpa harus datang ke instansi terkait.

Dynamic Priority Scheduling adalah pendekatan dinamis untuk algoritma penjadwalan prioritas. Dalam pendekatan dinamis, algoritma ini berfokus pada proses penentuan antrian berdasarkan aturan prioritas yang ditetapkan (Setyawati dan Maulachela, 2020). Penelitian Nasution (2023) berjudul *Customer Service Information System using Dynamic Priority Scheduling Algorithm at PT Sumatra Sistem Integrasi* menunjukkan bahwa Algoritma *Dynamic Priority Scheduling* terbukti efektif dalam menyelesaikan

permasalahan antrian layanan pelanggan di perusahaan tersebut. Penerapan Algoritma *Dynamic Priority Scheduling* memungkinkan sistem pengelolaan sampah menangani pengaduan berdasarkan tingkat urgensi, seperti volume, lokasi, waktu, dan dampak lingkungan. Pendekatan ini mempercepat respons pengangkutan dan meningkatkan efisiensi pengelolaan sampah secara keseluruhan (Labibah dan Pulungan, 2025).

Dengan diterapkannya sistem pengaduan tumpukan sampah berbasis Web yang terintegrasi dengan Algoritma *Dynamic Priority Scheduling* diharapkan proses pengelolaan pengaduan menjadi lebih cepat, objektif, dan terarah. Sistem ini tidak hanya meningkatkan efektivitas dalam penanganan laporan dari masyarakat, tetapi juga mendukung pengambilan keputusan oleh Dinas Lingkungan Hidup Kota Kendari.

Berdasarkan latar belakang yang telah diuraikan, serta untuk menjawab permasalahan pengelolaan pengaduan tumpukan sampah di Kelurahan Kambu, dibutuhkan sistem yang mampu mengelola laporan secara terstruktur, menentukan prioritas penanganan, serta menyajikan informasi lokasi secara visual. Oleh karena itu, penulis mengangkat topik penelitian berjudul **“Penerapan Algoritma Dynamic Priority Scheduling pada Sistem Pengaduan Tumpukan Sampah Berbasis Web”**.

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan melalui empat pendekatan utama yang saling melengkapi guna memperoleh data yang relevan dan komprehensif, yaitu studi literatur, survei lapangan, wawancara, dan kuesioner.

3.1.1 Studi Literatur

Studi literatur dilakukan untuk mengkaji teori, konsep, serta penelitian terdahulu yang relevan dengan topik penelitian, terutama terkait algoritma Dynamic Priority Scheduling serta penerapannya dalam konteks layanan publik, sistem antrean, dan pengelolaan pengaduan. Literatur yang dikaji meliputi jurnal ilmiah, buku, artikel, dan dokumentasi teknologi yang digunakan dalam pengembangan sistem, seperti Laravel, MySQL, dan PHP. Selain itu, penelitian-penelitian sebelumnya terkait sistem informasi pengaduan atau pelayanan publik berbasis web juga dianalisis secara mendalam sebagai pembanding dalam mengidentifikasi kelemahan sistem yang ada dan menemukan peluang perbaikan. Studi literatur ini menjadi landasan penting dalam merumuskan permasalahan, menentukan pendekatan metodologis, serta merancang solusi yang berbasis teknologi tepat guna (Avif Nurrohman et al., 2024; Masyfa et al., 2023).

3.1.2 Survei Lapangan

Survei lapangan dilakukan secara langsung di Kelurahan Kambu dan Dinas Lingkungan Hidup Kota Kendari. Tujuannya adalah untuk memperoleh data empiris terkait pengelolaan sampah di wilayah tersebut. Kegiatan survei mencakup observasi titik-titik penumpukan sampah, pengamatan terhadap jadwal dan keteraturan pengangkutan, serta identifikasi prosedur pengaduan manual yang digunakan masyarakat. Selain itu, interaksi dengan aparat kelurahan dan petugas kebersihan juga dilakukan guna menggali informasi terkait prosedur kerja dan hambatan teknis di lapangan. Data dari survei ini memberikan gambaran faktual yang digunakan sebagai dasar dalam perancangan sistem pengaduan sampah berbasis web.

3.1.3 Wawancara

Wawancara bersifat semi-terstruktur dan dilakukan terhadap pihak-pihak yang terkait langsung dengan pengelolaan sampah, yaitu pejabat Dinas Lingkungan Hidup Kota Kendari, masyarakat Kelurahan Kambu, serta petugas kebersihan. Wawancara dengan pihak dinas bertujuan untuk memperoleh informasi kebijakan, strategi operasional, serta kondisi armada pengangkut sampah. Dari masyarakat, wawancara digunakan untuk memahami persepsi, keluhan, serta harapan terhadap sistem pengaduan. Sedangkan dari petugas lapangan, wawancara menggali kendala teknis dan operasional yang dihadapi. Informasi yang diperoleh bersifat kualitatif dan digunakan untuk melengkapi serta memvalidasi data dari hasil observasi dan kuesioner.

3.1.4 Kuesioner

Kuesioner disusun dalam bentuk pertanyaan terbuka dan tertutup yang ditujukan kepada masyarakat Kelurahan Kambu. Tujuannya adalah untuk memperoleh data kuantitatif terkait frekuensi penemuan tumpukan sampah, mekanisme pengaduan yang biasa dilakukan, tingkat kepuasan terhadap sistem yang ada,

serta kesiapan menggunakan sistem digital. Hasil dari kuesioner dianalisis untuk memperkuat temuan dari survei lapangan dan wawancara, serta untuk menggambarkan persepsi masyarakat sebagai pengguna akhir terhadap sistem yang akan dikembangkan.

3.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah Rational Unified Process (RUP). RUP merupakan kerangka kerja pengembangan perangkat lunak yang bersifat iteratif dan use-case driven, serta menekankan pada pemisahan fase pengembangan menjadi empat tahapan utama yaitu Inception, Elaboration, Construction, dan Transition (Fariz & Riswandha, 2024).

3.2.1 Inception (Permulaan)

Tahap ini bertujuan untuk mengidentifikasi permasalahan yang terdapat dalam sistem pengaduan sampah yang masih bersifat manual, serta menentukan kebutuhan sistem yang akan dikembangkan. Analisis dilakukan dengan fokus pada kelemahan sistem eksisting dan ruang lingkup sistem yang dirancang. Hasil dari tahap ini adalah definisi kebutuhan awal dan tujuan pengembangan sistem.

3.2.2 Elaboration (Perencanaan)

Pada tahap ini dilakukan perancangan sistem secara lebih rinci, termasuk pemodelan sistem menggunakan Unified Modeling Language (UML). Diagram yang digunakan meliputi use case diagram, activity diagram, sequence diagram, dan class diagram. Selain itu, dilakukan perancangan antarmuka pengguna yang mempertimbangkan aspek user friendly dan aksesibilitas. Dalam tahap ini juga ditentukan bagaimana algoritma Dynamic Priority Scheduling diterapkan untuk memprioritaskan laporan berdasarkan parameter tertentu (Fatmasari & Sauda, 2020; Masyfa et al., 2023).

3.2.3 Construction (Konstruksi)

Tahap konstruksi merupakan fase implementasi sistem. Sistem dibangun menggunakan framework Laravel, bahasa pemrograman PHP, dan basis data MySQL. Pada tahap ini juga diterapkan algoritma Dynamic Priority Scheduling untuk mengelola prioritas laporan secara dinamis, berdasarkan parameter seperti kepadatan sampah, lokasi, dan tingkat urgensi (Avif Nurrohman et al., 2024). Selanjutnya dilakukan uji coba fungsional terhadap sistem untuk memastikan kesesuaian antara rancangan dan implementasi.

3.2.4 Transition (Transisi)

Tahap transisi difokuskan pada penerapan sistem ke pengguna akhir. Dilakukan uji coba sistem secara menyeluruh melalui metode User Acceptance Testing (UAT) untuk mengevaluasi kesesuaian sistem terhadap kebutuhan pengguna (Aliyah et al., 2025). Selain itu, dilakukan sosialisasi dan pelatihan kepada calon pengguna dari kalangan masyarakat dan Dinas Lingkungan Hidup. Evaluasi terhadap sistem dilakukan untuk mengidentifikasi kelebihan dan kekurangan sebelum penerapan skala penuh.

3.3 Waktu dan Tempat Penelitian

3.3.1 Waktu Penelitian

Penelitian ini dilaksanakan dalam rentang waktu April hingga Oktober 2025, dengan tahapan meliputi pengumpulan data, perancangan sistem, implementasi, serta pengujian dan evaluasi.

3.3.2 Tempat Penelitian

Penelitian ini dilaksanakan di Kelurahan Kambu, Kota Kendari, Sulawesi Tenggara, sebagai lokasi lapangan utama, serta di Laboratorium Software Engineering, Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, sebagai tempat perancangan dan pengembangan sistem.

HASIL DAN PEMBAHASAN

3.4 Hasil

3.4.1 Analisis Kebutuhan Sistem

Pada tahap analisis kebutuhan sistem dalam pengembangan suatu sistem, fokus diberikan pada pemahaman dan dokumentasi kebutuhan dari komponen-komponen terkait dalam sistem tersebut. Tahap ini melibatkan identifikasi rincian atau struktur dalam pengembangan perangkat lunak, termasuk spesifikasi untuk masukan, keluaran, dan proses yang diperlukan untuk mengelola masukan guna menghasilkan keluaran sesuai dengan keinginan. Analisis kebutuhan sistem juga mencakup evaluasi kebutuhan fungsional dengan merancang sistem menggunakan *Unified Modeling Language* (UML), perancangan tampilan *interface* (antar muka), serta analisis kebutuhan non-fungsional yang mencakup persyaratan perangkat keras dan lunak yang diperlukan dalam pengembangan sistem.

3.4.1.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah spesifikasi mengenai proses-proses yang harus dilakukan oleh sistem agar dapat beroperasi sesuai dengan tujuan yang telah ditetapkan. Kebutuhan ini mencakup spesifikasi masukan, proses, dan keluaran yang diperlukan untuk mendukung fungsionalitas utama sistem pelaporan pengaduan sampah berbasis Web dan Algoritma Dynamic Priority Scheduling. Sistem yang dibangun harus mampu menerima pengaduan dari masyarakat, mengelola data pengaduan, menghitung prioritas penanganan secara otomatis, serta memberikan informasi status penanganan kepada pelapor. Adapun kebutuhan fungsional sistem adalah:

1. Pembuatan Pengaduan, masyarakat dapat mengisi data laporan berupa alamat, deskripsi kejadian, volume sampah, dampak lingkungan, serta mengunggah foto dokumentasi.
2. *Monitoring* Status Pengaduan, sistem harus menyediakan fitur untuk melihat status pengaduan yang telah dibuat (Menunggu, Diproses, Selesai) beserta riwayat perubahan status dengan catatan dan notifikasi kepada pelapor.
3. Pengelolaan Pengaduan oleh Petugas, petugas dapat melihat daftar laporan yang masuk secara detail, memverifikasi data, serta memperbarui status penanganan.
4. Penentuan Prioritas Dinamis, sistem secara otomatis menghitung skor prioritas pengaduan dengan Algoritma Dynamic Priority Scheduling berdasarkan kriteria volume, waktu pelaporan, jarak lokasi (dari deskripsi alamat), dan dampak lingkungan.
5. Penjadwalan Penanganan, sistem mengurutkan laporan berdasarkan hasil perhitungan prioritas dan menampilkan daftar eksekusi untuk petugas lapangan.
6. Pembaruan Status Pengaduan, petugas dapat memperbarui status pengaduan secara berkala sehingga masyarakat mengetahui progres penanganan.
7. Manajemen Data Pengguna, admin dapat mengelola data pengguna (masyarakat, petugas, admin), melakukan registrasi, *login*, dan pengaturan hak akses.
8. *Dashboard* Laporan, sistem menyediakan statistik jumlah laporan, status penanganan, serta distribusi kategori prioritas untuk mendukung evaluasi Dinas Lingkungan Hidup.

3.4.1.2 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan langkah penting dalam pembangunan aplikasi, di mana dilakukan penilaian terhadap sumber daya yang diperlukan. Proses analisis ini terbagi menjadi dua tahap utama, yaitu analisis kebutuhan perangkat keras dan analisis kebutuhan perangkat lunak. Tujuan dari analisis kebutuhan non fungsional adalah untuk mengidentifikasi dan memahami semua kebutuhan sumber daya yang diperlukan dalam pembangunan aplikasi, baik dari segi perangkat keras maupun perangkat lunak.

3.4.1.2.1 Kebutuhan Perangkat Keras (Hardware)

Untuk mendukung implementasi aplikasi yang telah dirancang, diperlukan perangkat keras dengan spesifikasi yang memadai guna memastikan proses pengembangan dan pengujian sistem dapat berjalan secara optimal. Perangkat utama yang digunakan dalam penelitian ini adalah sebuah laptop dengan merek Asus VivoBook S14, yang dilengkapi dengan prosesor AMD Ryzen 7 4700U untuk mendukung performa komputasi yang tinggi selama proses pengembangan. Laptop ini memiliki layar berukuran 14 inci, yang

cukup ideal untuk aktivitas pemrograman serta perancangan antarmuka pengguna. Selain itu, perangkat ini juga didukung oleh memori RAM sebesar 8 GB, yang memungkinkan eksekusi simultan berbagai aplikasi pengembangan seperti editor kode, server lokal, dan basis data tanpa mengalami penurunan kinerja. Spesifikasi perangkat keras tersebut dipilih untuk memastikan seluruh tahapan dalam proses pengembangan sistem, mulai dari penulisan kode hingga pengujian aplikasi berbasis web, dapat dilakukan secara efisien dan efektif.

3.4.1.2.2 Kebutuhan Perangkat Lunak (Software)

Dalam mendukung proses pengembangan sistem pengaduan tumpukan sampah berbasis web, penelitian ini memanfaatkan sejumlah perangkat lunak yang memiliki fungsi dan spesifikasi sesuai dengan kebutuhan teknis sistem. Sistem operasi yang digunakan adalah Windows 11 Home, yang berperan sebagai platform utama untuk menjalankan seluruh perangkat lunak pengembangan. Untuk pengelolaan lingkungan pengembangan lokal, digunakan Laragon versi 6.0, yang telah terintegrasi dengan berbagai komponen penting seperti server Apache, MySQL, dan PHP.

Sistem manajemen basis data yang digunakan adalah MySQL versi 8.0, yang berfungsi untuk menyimpan dan mengelola data laporan dari pengguna. Bahasa pemrograman server-side yang digunakan adalah PHP versi 8.3.11, sementara struktur halaman web dibangun menggunakan HTML versi 5. Untuk menambahkan elemen interaktivitas pada sisi klien, digunakan JavaScript versi 8.

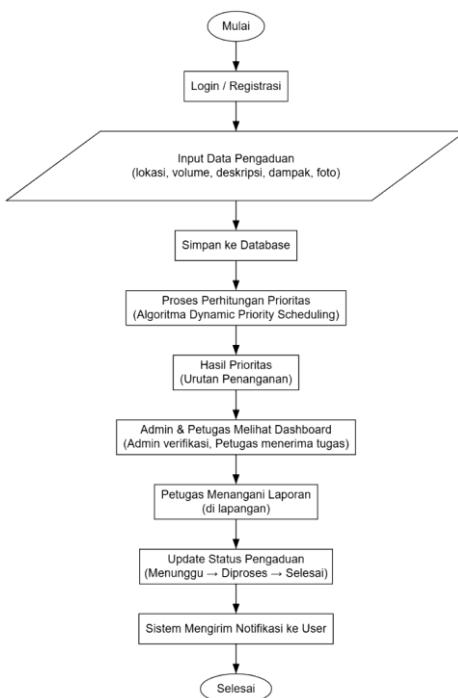
Dalam pengembangan aplikasi, digunakan framework Laravel versi 11 yang berbasis arsitektur Model-View-Controller (MVC), sehingga mempermudah dalam membangun aplikasi yang terstruktur dan terorganisir dengan baik. Seluruh proses penulisan kode dilakukan menggunakan Visual Studio Code versi terbaru, yang merupakan editor kode sumber modern dengan berbagai ekstensi pendukung untuk pengembangan web. Kombinasi perangkat lunak tersebut memungkinkan proses perancangan, pengembangan, dan pengujian aplikasi dilakukan secara optimal serta sesuai dengan kebutuhan sistem yang dirancang.

3.4.2 Perancangan Sistem

Perancangan sistem merupakan tahap penting yang mencakup perencanaan alur kerja serta desain logika sistem yang akan dikembangkan agar dapat memenuhi tujuan penelitian. Dalam penelitian ini, sistem dirancang untuk membantu masyarakat melaporkan tumpukan sampah secara daring sekaligus memberikan dukungan bagi Dinas Lingkungan Hidup dalam menentukan urutan prioritas penanganan laporan. Sistem ini mengimplementasikan Algoritma Dynamic Priority Scheduling yang berfungsi untuk mengatur antrian pengaduan berdasarkan tingkat urgensi.

Algoritma ini mempertimbangkan empat parameter utama, yaitu volume sampah, waktu pengaduan, jarak lokasi pengaduan dari titik layanan terdekat, dan dampak lingkungan. Dengan adanya perhitungan skor prioritas dari setiap laporan, sistem dapat mengurutkan pengaduan sehingga tumpukan sampah yang paling mendesak dapat segera ditangani.

Alur kerja sistem dimulai dari masyarakat yang melaporkan pengaduan melalui web, kemudian sistem menyimpan data tersebut dalam basis data. Selanjutnya, Algoritma Dynamic Priority Scheduling akan menghitung nilai prioritas untuk setiap laporan. Hasil perhitungan digunakan untuk menyusun antrian penanganan, yang kemudian ditampilkan dalam *dashboard* admin. Dengan demikian, proses pengelolaan laporan tumpukan sampah dapat berlangsung secara lebih cepat, transparan, dan berbasis skala prioritas. Adapun alur kerja sistem dapat dilihat pada Gambar 1 berikut.



Gambar 1. Flowchart Alur Sistem

Flowchart pada Gambar 1 menggambarkan tahapan proses dalam penerapan Algoritma Dynamic Priority Scheduling. Berikut adalah tahapan utama sistem.

- Mulai, proses dimulai ketika pengguna (masyarakat atau petugas) mengakses sistem.
- Login/Registrasi, pengguna melakukan autentikasi dengan login, atau registrasi apabila belum memiliki akun. Hal ini memastikan hanya pengguna terdaftar yang dapat mengakses sistem.
- Input Data Pengaduan, masyarakat mengisi formulir pengaduan berupa informasi lokasi kejadian, volume sampah, deskripsi, dampak lingkungan, serta bukti foto.
- Simpan ke Database, Data yang di masukkan pengguna akan divalidasi dan kemudian disimpan ke dalam basis data MySQL untuk dikelola lebih lanjut.
- Proses Perhitungan Prioritas, sistem menjalankan Algoritma Dynamic Priority Scheduling untuk menghitung skor prioritas setiap laporan berdasarkan empat kriteria utama; volume sampah, waktu pengaduan, jarak lokasi dari titik layanan, serta dampak lingkungan.
- Hasil Prioritas, setelah sistem melakukan perhitungan dengan Algoritma Dynamic Priority Scheduling, setiap pengaduan akan memperoleh nilai prioritas tertentu. Nilai ini digunakan untuk menentukan urutan penanganan laporan secara objektif. Laporan dengan tingkat urgensi lebih tinggi, seperti volume sampah yang besar, waktu pengaduan yang lebih lama, lokasi yang dekat dengan armada, atau dampak lingkungan yang serius, akan ditempatkan pada urutan teratas. Dengan demikian, petugas dapat langsung mengetahui laporan mana yang harus ditangani terlebih dahulu.
- Admin dan Petugas melihat *dashboard*, pada tahap ini, sistem menampilkan data hasil prioritas ke dalam *dashboard*. Admin berperan sebagai pihak yang memverifikasi kebenaran data laporan yang masuk, termasuk memastikan lokasi dan deskripsi pengaduan sesuai dengan kondisi lapangan. Setelah proses verifikasi dilakukan, data laporan yang sudah valid ditampilkan dalam bentuk daftar urutan prioritas. Daftar ini kemudian menjadi acuan bagi petugas kebersihan dalam menjalankan tugasnya.
- Petugas Menangani Laporan, petugas kebersihan melaksanakan penanganan laporan sesuai instruksi sistem di lapangan.
- Update Status Pengaduan, setelah laporan ditangani, status pengaduan diperbarui di sistem (Menunggu, Diproses, Selesai).

- j. Sistem Mengirim Notifikasi, sistem secara otomatis mengirimkan notifikasi kepada masyarakat terkait perkembangan status laporannya.
- k. Selesai, proses pengaduan dianggap selesai setelah laporan ditangani dan status pengaduan telah diperbarui.

3.4.3 Analisis Perancangan UML

Unified Modeling Language (UML) adalah suatu tata baku yang telah menjadi standar industri untuk merancang, mendokumentasikan, dan visualisasi sistem perangkat lunak. Adapun *unified modeling language* yang akan diaplikasikan pada sistem yang akan dibangun ini terdiri dari *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

1. Use Case Diagram

Use Case Diagram menggambarkan interaksi antara pengguna dengan sistem pengaduan tumpukan sampah. Terdapat dua aktor utama, yaitu *User* (Masyarakat) dan *Admin/Petugas*. Masing-masing aktor memiliki peran dan hak akses yang berbeda sesuai fungsinya dalam sistem.

- a. Sebagai aktor utama dalam sistem, *user* atau masyarakat memiliki beberapa fungsi utama yang dapat diakses. Pertama, masyarakat perlu melakukan *login* menggunakan akun yang telah terdaftar untuk dapat masuk ke dalam sistem. Setelah itu, mereka dapat memanfaatkan fitur “Buat Pengaduan”, yaitu dengan mengisi formulir yang tersedia dengan detail informasi seperti lokasi, deskripsi masalah, volume sampah, serta melampirkan foto sebagai bukti pendukung. Selain itu, masyarakat juga dapat menggunakan fitur “Status Pengaduan” untuk memantau perkembangan laporan yang telah dikirim, mulai dari status menunggu, diproses, hingga selesai. Setelah selesai menggunakan sistem, masyarakat dapat melakukan “*Logout*” sebagai langkah keluar dari aplikasi untuk menjaga keamanan akun yang dimiliki.

- b. Sebagai pengelola sistem, admin atau petugas memiliki peran yang lebih luas dibandingkan masyarakat, karena bertanggung jawab dalam memverifikasi, mengatur, dan menindaklanjuti pengaduan yang masuk. Admin dapat melakukan penerimaan pengaduan dari masyarakat, kemudian sistem akan membantu mereka dalam menentukan prioritas pengaduan dengan menggunakan algoritma *Dynamic Priority Scheduling* sehingga laporan yang lebih mendesak dapat segera ditangani. Selanjutnya, admin berperan dalam menjadwalkan penanganan pengaduan sesuai dengan urutan prioritas yang dihasilkan. Untuk menjaga transparansi, admin juga dapat memperbarui status pengaduan sesuai perkembangan, seperti dari menunggu menjadi diproses atau selesai. Selain itu, admin memiliki akses untuk mengelola data pengguna dan data sistem, termasuk menambah, memperbarui, atau menghapus data pengguna, sehingga keamanan serta keteraturan sistem tetap terjaga. Dengan demikian, admin berfungsi sebagai penghubung utama antara laporan masyarakat dengan proses penanganan lapangan. Mengelola Data Pengguna dan Data Sistem, yaitu mengatur informasi pengguna (*user* dan admin lain) serta melakukan manajemen data di dalam sistem untuk menjaga keamanan dan keteraturan.

2. Activity Diagram

Activity Diagram menggambarkan berbagai aliran proses dalam sistem yang akan dirancang, mencakup bagaimana setiap aliran dimulai, kemungkinan keputusan yang dapat terjadi, dan bagaimana aliran tersebut berakhir. Adapun *Activity Diagram* sistem yang akan dibangun adalah sebagai berikut.

a. Activity Diagram Login (Admin dan User)

Activity diagram menggambarkan alur *activity* pengguna sistem, baik sebagai admin maupun *user*. Proses dimulai ketika pengguna mengakses halaman *login* dan memasukkan *username* dan *password*. Selanjutnya, sistem melakukan proses *verifikasi* dengan mencocokkan data yang dimasukkan terhadap data yang terdapat di basis data. Apabila data yang dimasukkan sesuai, maka sistem akan mengarahkan pengguna ke halaman utama (*dashboard*) sesuai dengan perannya. Namun, apabila terdapat ketidaksesuaian, maka sistem akan menampilkan pesan kesalahan dan meminta pengguna untuk mengulangi proses *login*.

b. *Activity Diagram Buat Pengaduan (User)*

Activity diagram ini menjelaskan proses pelaporan pengaduan tumpukan sampah oleh pengguna sistem. Proses diawali saat pengguna mengakses menu pelaporan pengaduan. Setelah itu, pengguna mengisi data laporan, yang terdiri dari lokasi kejadian, volume sampah, dampak lingkungan, dan informasi tambahan seperti foto dokumentasi. Setelah semua data diisi dengan lengkap, pengguna mengirimkan laporan tersebut kepada sistem. Sistem kemudian akan menyimpan data laporan ke dalam basis data dan memberikan notifikasi bahwa laporan berhasil dikirim.

c. *Activity Diagram Status Pengaduan (User)*

Proses dimulai saat pengguna memilih menu “Status Pengaduan” pada antarmuka sistem. Sistem akan mengambil data status laporan dari basis data, kemudian menampilkannya dalam bentuk daftar laporan yang mencakup informasi status terkini dari masing-masing laporan, seperti “Menunggu”, “Diproses”, atau “Selesai”.

d. *Activity Diagram Kelola Data Pengaduan (Admin)*

Activity diagram ini merupakan proses yang dilakukan oleh admin dalam mengelola data pengaduan yang masuk dari masyarakat. Setelah admin berhasil melakukan *login*, ia mengakses menu pengelolaan pengaduan. Sistem kemudian menampilkan seluruh laporan yang telah diterima dari pengguna. Admin dapat melihat detail setiap laporan, melakukan verifikasi informasi, dan apabila diperlukan dapat memperbarui data laporan. Setelah proses pengelolaan selesai, data diperbarui akan disimpan kembali ke dalam sistem.

e. *Activity Diagram Kelola Prioritas (Admin)*

Activity diagram proses perhitungan prioritas laporan pengaduan menggunakan Algoritma *Dynamic Priority Scheduling*. Proses dimulai saat admin mengakses menu prioritas pengaduan. Sistem mengambil seluruh data laporan berstatus “Menunggu” dari basis data, kemudian melakukan perhitungan prioritas berdasarkan parameter yang telah ditentukan, yaitu volume sampah, waktu pelaporan, jarak lokasi, dan dampak lingkungan. Hasil dari perhitungan ini adalah urutan prioritas laporan dari yang paling mendesak hingga yang kurang mendesak.

f. *Activity Diagram Kelola Data User (Admin)*

Activity diagram menggambarkan alur kerja admin dalam mengelola informasi pengguna sistem. Admin dapat melakukan penambahan, pengubahan, maupun penghapusan data *user* melalui antarmuka sistem. Setelah admin memilih salah satu tindakan, sistem akan menampilkan formulir data yang harus diisi atau diperbarui. Setelah data dimasukkan atau diedit, sistem menyimpan informasi tersebut ke dalam basis data. Proses ini dilakukan untuk menjaga integritas dan keamanan sistem serta memastikan bahwa hanya pengguna yang sah yang dapat menggunakan fitur dalam sistem.

3. *Sequence Diagram*

Sequence diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antar objek dalam sebuah sistem berdasarkan urutan waktu. *Diagram* ini menyajikan alur komunikasi antar objek melalui pengiriman pesan yang terjadi dalam sebuah skenario tertentu, dengan memperlihatkan bagaimana objek-objek saling berinteraksi secara kronologis untuk mencapai suatu tujuan atau proses bisnis. Berikut ini merupakan *sequence diagram* dari sistem yang akan dibangun.

a. *Sequence Diagram Login (Admin dan user)*

Sequence diagram login (admin dan *user*) dari sistem yang akan dibangun merupakan interaksi antara pengguna dengan sistem saat melakukan *login*. Pengguna memasukkan *username* dan *password* melalui antarmuka sistem. Sistem kemudian mengirimkan permintaan verifikasi kepada basis data. Jika kredensial sesuai, sistem memberikan akses kepada pengguna untuk masuk ke dalam sistem dan menampilkan halaman utama. Jika tidak sesuai, sistem akan menampilkan pesan kesalahan.

b. *Sequence Diagram Melaporakan Pengaduan (User)*

Sequence diagram pelaporan pengaduan oleh pengguna sistem dimana pengguna mengakses form pengaduan, kemudian mengisi informasi terkait laporan yang meliputi lokasi, volume sampah, dampak lingkungan, dan mengunggah gambar. Setelah data dikirim, sistem melakukan validasi dan menyimpan data ke dalam basis data. Sistem kemudian memberikan notifikasi bahwa laporan telah berhasil dikirim.

c. *Sequence Diagram Status Pengaduan (User)*

Sequence diagram status pengaduan (*user*) menggambarkan proses pengguna dalam melihat status laporan yang telah dikirim. Pengguna mengakses menu status, lalu sistem mengambil data dari basis data dan menampilkannya dalam bentuk daftar yang mudah dipahami.

d. *Sequence Diagram Kelola Pengaduan (Admin)*

Sequence diagram ini menggambarkan aktivitas yang dilakukan oleh admin dalam mengelola laporan dari pengguna. Admin membuka menu pengelolaan laporan dan memilih laporan yang akan diperiksa. Sistem menampilkan data laporan secara lengkap. Admin dapat melakukan validasi atau pengeditan informasi, kemudian menyimpan pembaruan tersebut. Sistem akan menyimpan perubahan ke dalam basis data dan memberikan umpan balik berupa notifikasi bahwa data berhasil diperbarui.

e. *Sequence Diagram Kelola Prioritas (Admin)*

Sequence diagram ini menjelaskan proses saat admin menjalankan fitur penjadwalan prioritas. Admin memilih menu proses prioritas, kemudian sistem mengambil seluruh laporan yang belum diproses dan menghitung nilai prioritas menggunakan Algoritma *Dynamic Priority Scheduling*. Hasilnya ditampilkan dalam urutan prioritas untuk memudahkan admin menentukan penanganan laporan secara efisien dan objektif.

f. *Sequence Diagram Kelola Data User (Admin)*

Sequence diagram kelola data *user* yang dilakukan oleh admin. Admin dapat memilih untuk menambahkan, mengedit, atau menghapus data *user*. Setelah memilih aksi, sistem akan menampilkan formulir data yang dapat diisi atau diperbarui. Selanjutnya, data disimpan ke dalam basis data dan sistem memberikan konfirmasi bahwa proses berhasil dilakukan. Aktivitas ini diperlukan untuk memastikan bahwa data pengguna sistem senantiasa diperbarui dan sesuai dengan kebutuhan operasional.

4. Class Digaram

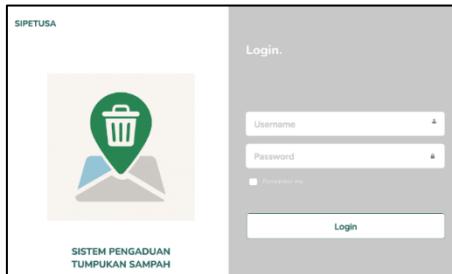
Diagram kelas adalah salah satu bentuk diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk merepresentasikan struktur dari suatu sistem perangkat lunak. Diagram ini menunjukkan berbagai kelas yang terdapat dalam sistem, termasuk atribut dan metode yang dimiliki oleh masing-masing kelas, serta relasi antar kelas seperti asosiasi, agregasi, komposisi, dan pewarisan.

3.4.4 Perancangan Interface

Perancangan *interface* merupakan bagian penting dalam pengembangan sebuah sistem karena berfungsi sebagai jembatan utama antara pengguna dengan aplikasi. Tujuannya adalah menghadirkan tampilan yang sederhana, mudah dipahami, serta nyaman digunakan oleh siapa pun. Dalam proses perancangannya, pendekatan yang digunakan berfokus pada pengguna (*user-centered design*), sehingga setiap elemen yang ditampilkan benar-benar menyesuaikan dengan kebutuhan, kebiasaan, dan preferensi mereka. Komponen visual seperti tombol, ikon, maupun menu dirancang konsisten agar mudah dikenali dan dipahami, sehingga pengguna tidak kesulitan dalam melakukan navigasi. Selain itu, setiap fitur diatur secara logis dan rapi agar bisa diakses dengan cepat tanpa membuat pengguna merasa rumit. Respon yang interaktif, tata letak yang tertata, serta sentuhan estetika yang menyenangkan semakin menambah kenyamanan ketika berinteraksi dengan sistem. Dengan desain yang berorientasi pada kenyamanan dan efisiensi, antarmuka tidak hanya memberikan pengalaman penggunaan yang lebih baik, tetapi juga membantu pengguna mencapai tujuan mereka secara lebih efektif dan menyenangkan.

1. Halaman Login (Admin dan User)

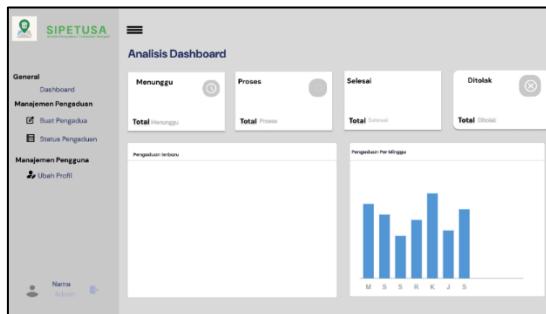
Halaman *login* dirancang agar dapat digunakan oleh dua jenis pengguna, yaitu masyarakat sebagai pelapor dan admin/petugas sebagai pengelola laporan. Pengguna diminta memasukkan *username* dan kata sandi untuk dapat masuk ke sistem. Tampilan *login* dibuat sederhana dengan fokus pada kolom input dan tombol masuk, sehingga memudahkan pengguna dalam mengakses sistem dengan cepat. Adapun perancangan *interface* halaman *login* (admin dan *user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 2 berikut.



Gambar 2. Halaman Login (Admin dan User)

2. Halaman Beranda (User)

Halaman beranda untuk pengguna berfungsi sebagai menu utama yang menampilkan navigasi menuju fitur penting, seperti melaporkan pengaduan, melihat status laporan, dan mengelola profil akun. Desain halaman ini menekankan pada kemudahan navigasi, sehingga masyarakat dapat langsung memilih layanan sesuai kebutuhan. Adapun perancangan *interface* halaman beranda (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 4 berikut.



Gambar 3. Halaman Beranda (User)

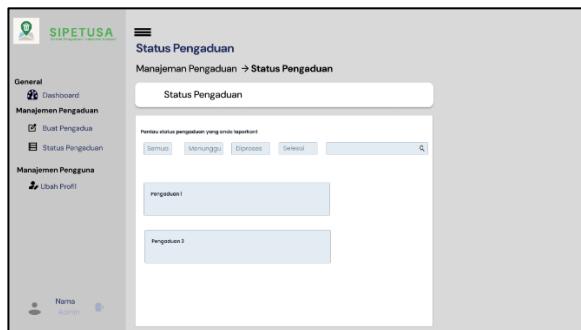
3. Halaman Buat Pengaduan (*User*)

Halaman ini menyediakan formulir bagi masyarakat untuk melaporkan tumpukan sampah. Formulir terdiri dari beberapa input, antara lain alamat lokasi, deskripsi pengaduan, volume sampah, dampak lingkungan, serta fitur unggah foto. Dengan tampilan *form* yang terstruktur, masyarakat dapat mengisi laporan secara sistematis dan jelas. Adapun perancangan *interface* halaman Buat pengaduan (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 5 berikut.

Gambar 4. Halaman Buat Pengaduan (User)

4. Halaman Status Pengaduan (*User*)

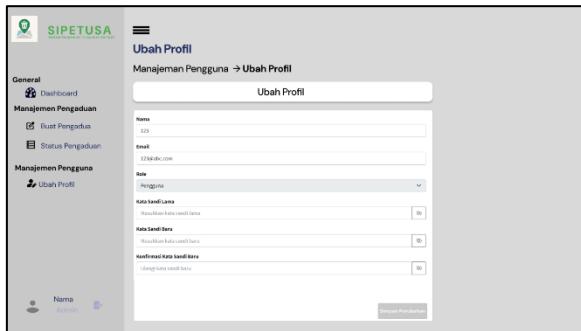
Halaman status pengaduan berfungsi untuk memantau perkembangan laporan yang telah dikirimkan oleh masyarakat. Status ditampilkan dalam bentuk tabel yang berisi keterangan apakah laporan masih menunggu, sedang diproses, atau sudah selesai. Fitur ini memberikan transparansi kepada masyarakat sehingga mereka mengetahui tindak lanjut dari laporan yang diajukan. Adapun perancangan *interface* halaman status pengaduan (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 6 berikut.



Gambar 5. Halaman Status Pengaduan (*User*)

5. Halaman Ubah Profil (*User*)

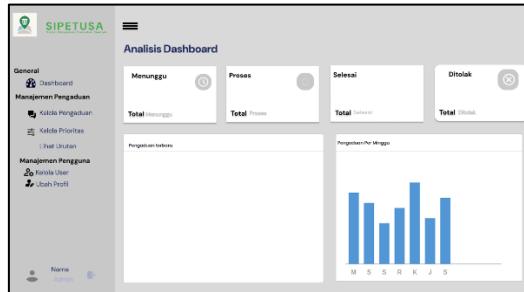
Pada halaman ini, pengguna dapat memperbarui data seperti nama, email, serta mengubah kata sandi apabila diperlukan. Tampilan *form* disusun sederhana dan jelas, dengan kolom input yang mudah diisi sehingga memudahkan proses pengeditan. Selain itu, tersedia fitur konfirmasi kata sandi baru untuk memastikan keamanan dan meminimalisir kesalahan saat penggantian *password*. Adapun halaman ubah profil (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 7 berikut.



Gambar 6. Halaman Ubah Profil (*User*)

6. Halaman Dashboard (Admin)

Halaman *dashboard* ditujukan bagi admin dan petugas sebagai pusat informasi utama. *Dashboard* menampilkan ringkasan jumlah laporan, status pengaduan, serta grafik distribusi laporan berdasarkan kategori prioritas. Tampilan ini membantu admin dalam memantau kondisi lapangan secara keseluruhan. Adapun perancangan *interface* halaman *dashboard* (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 8 berikut.



Gambar 7. Halaman Dashboard (Admin)

7. Halaman Kelola Pengaduan (Admin)

Halaman ini digunakan oleh admin untuk melihat daftar pengaduan yang masuk secara detail. Admin dapat memverifikasi laporan, menambahkan catatan, dan memperbarui status laporan. Data pengaduan ditampilkan dalam bentuk tabel dengan tombol aksi yang memudahkan proses pengelolaan laporan. Adapun perancangan *interface* halaman kelola pengaduan (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 9 berikut.

| No | Nama Pelapor | Lokasi | Waktu | Volume | Deskripsi | Status | Aksi |
|----|--------------|------------|-------|--------|------------------------------|------------|------|
| 1 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Pending | |
| 2 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Waiting | |
| 3 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Processing | |
| 4 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Rejected | |

Gambar 8. Halaman Kelola Pengaduan (Admin)

8. Halaman Lihat Urutan (Admin)

Halaman lihat urutan (admin/petugas) menampilkan daftar laporan pengaduan yang telah masuk ke dalam sistem dan diurutkan berdasarkan hasil perhitungan prioritas menggunakan algoritma *Dynamic Priority Scheduling*. Pada halaman ini, informasi laporan ditampilkan dalam bentuk tabel yang memuat data nama pelapor, lokasi, waktu pengaduan, volume sampah, deskripsi, serta status penanganan. Admin atau petugas dapat memantau urutan prioritas laporan untuk menentukan pengaduan mana yang harus ditangani lebih dahulu. Selain itu, tersedia menu aksi berupa ikon untuk memperbarui atau menghapus data pengaduan, sehingga proses pengelolaan dapat dilakukan dengan lebih mudah. Halaman ini juga dilengkapi fitur pencarian dan tombol *Export PDF* yang memungkinkan admin mencetak daftar laporan sebagai bahan laporan resmi. Adapun perancangan *interface* halaman lihat urutan (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 10 berikut.

| No | Nama Pelapor | Lokasi | Waktu | Volume | Deskripsi | Status | Aksi |
|----|--------------|------------|-------|--------|------------------------------|------------|------|
| 1 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Pending | |
| 2 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Waiting | |
| 3 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Processing | |
| 4 | Urg. Mewar | 2024-08-10 | 10:00 | 1000 | Tempat sampah di depan rumah | Rejected | |

Gambar 9. Halaman Lihat Urutan (Admin)

9. Halaman Kelola User (Admin)

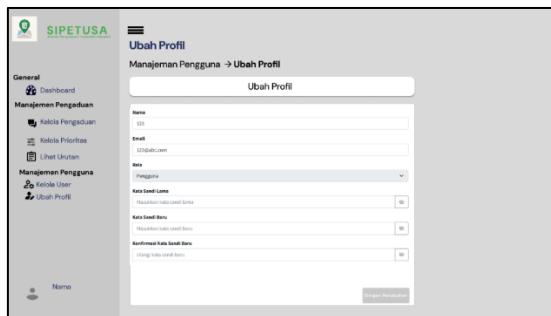
Halaman ini digunakan admin untuk mengelola data pengguna, baik masyarakat, petugas, maupun admin lainnya. Admin dapat menambah akun baru, memperbarui data, maupun menghapus pengguna yang tidak aktif. Adapun perancangan *interface* halaman proses prioritas (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 11 berikut.

| No | Nama | Email | Role | Status | Aksi |
|----|------|-------|------|----------------|------|
| 1 | | | | Belum ada User | |

Gambar 10. Halaman Kelola User (Admin)

10. Halaman Ubah Profil (Admin)

Halaman ubah profil (admin) merupakan fitur yang disediakan untuk memberikan keleluasaan kepada admin dalam mengelola informasi akun pribadinya. Pada halaman ini, admin dapat memperbarui data penting seperti nama, alamat email, serta melakukan perubahan kata sandi demi menjaga keamanan akun. Tampilan form disusun sederhana dan terstruktur dengan kolom input yang jelas, meliputi kolom kata sandi lama, kata sandi baru, dan konfirmasi kata sandi baru. Selain itu, disediakan tombol simpan perubahan agar admin dapat menyimpan data yang telah diperbarui. Adapun perancangan *interface* halaman kelola data *user* (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 12 berikut.



Gambar 11. Halaman Ubah Profil (Admin)

3.4.5 Pengujian User Acceptance Testing

Pengujian dilakukan menggunakan pendekatan *User Acceptance Testing* (UAT) untuk memastikan bahwa sistem telah memenuhi kebutuhan dan harapan pengguna. UAT mencakup pengujian terhadap fungsionalitas, antarmuka, serta kemudahan penggunaan sistem. Tujuannya adalah untuk memastikan bahwa setiap fitur, tombol, dan tampilan bekerja dengan baik serta sesuai dengan ekspektasi pengguna akhir.

Dalam pengujian User Acceptance Testing (UAT), sejumlah pertanyaan disusun untuk mengevaluasi sistem dari berbagai aspek, yaitu fungsionalitas, usability, kompatibilitas, performa sistem, kesesuaian proses bisnis, dan aksesibilitas.

Dari aspek fungsional, pengguna diminta menilai apakah sistem berhasil menerima dan menyimpan laporan pengaduan secara lengkap, termasuk alamat, deskripsi, volume, dan foto. Selain itu, ditanyakan pula apakah sistem mampu menghitung prioritas pengaduan berdasarkan parameter seperti volume, waktu, jarak, dan dampak lingkungan dengan menggunakan algoritma Dynamic Priority Scheduling. Evaluasi juga mencakup keakuratan tampilan urutan prioritas laporan di dashboard admin, kemampuan admin atau petugas dalam memperbarui status laporan (Menunggu, Diproses, Selesai), serta kejelasan riwayat status yang ditampilkan kepada pelapor.

Pada aspek usability, pertanyaan difokuskan pada kemudahan penggunaan sistem, seperti apakah tampilan halaman pelaporan mudah dipahami oleh pengguna baru, apakah menu navigasi seperti lapor, status, dan dashboard tersusun logis serta mudah diakses, dan apakah instruksi penggunaan aplikasi mudah diikuti. Penilaian juga dilakukan terhadap kejelasan pesan error saat input tidak lengkap dan kenyamanan interaksi saat pengguna mengakses laporan atau status pengaduan.

Untuk aspek kompatibilitas, dievaluasi apakah aplikasi dapat diakses melalui berbagai jenis browser seperti Chrome, Firefox, dan Edge, serta kompatibel baik di perangkat komputer maupun smartphone. Selain itu, dinilai apakah tampilan peta pengaduan terlihat rapi di perangkat mobile maupun desktop, apakah sistem tetap dapat digunakan meskipun koneksi internet lambat, dan apakah koordinat GPS lokasi laporan terbaca dengan benar oleh sistem.

Pada aspek performa sistem, pengguna diminta menilai kecepatan sistem dalam menampilkan daftar laporan dan sejauh mana algoritma penentuan prioritas mempengaruhi performa sistem secara keseluruhan. Evaluasi juga mencakup kelancaran interaksi pengguna dengan peta pengaduan, khususnya saat memperbesar atau memperkecil tampilan, serta efisiensi penggunaan memori dan daya perangkat saat aplikasi digunakan dalam jangka waktu lama.

Aspek kesesuaian proses bisnis mencakup pertanyaan tentang sejauh mana sistem membantu menentukan laporan yang harus diprioritaskan penanganannya berdasarkan kriteria yang telah ditentukan. Pengguna juga diminta menilai apakah perubahan data seperti volume atau dampak secara otomatis menghasilkan perubahan urutan prioritas, apakah sistem dapat memberikan estimasi urutan penanganan yang logis bagi petugas kebersihan, dan apakah sistem mendukung kebutuhan operasional Dinas Lingkungan Hidup, khususnya dalam menentukan prioritas harian. Selain itu, relevansi dan keterterapan informasi prioritas di lapangan juga menjadi bagian dari penilaian.

Terakhir, dalam aspek aksesibilitas, pengguna mengevaluasi apakah tombol navigasi dan menu memiliki ukuran yang cukup besar untuk diakses dengan mudah, apakah ikon pada peta (seperti titik laporan sampah) dilengkapi dengan keterangan yang jelas, serta apakah ukuran teks pada aplikasi dapat dibaca dengan mudah oleh berbagai kalangan. Antarmuka sistem juga dinilai dari sisi kemudahan penggunaan oleh masyarakat umum, khususnya pengguna non-teknis.

3.5 Pembahasan

3.5.1 Implementasi Sistem

Implementasi sistem dilakukan berdasarkan hasil perancangan yang telah dibahas pada bab sebelumnya. Sistem pengaduan tumpukan sampah ini dibangun berbasis web untuk memberikan kemudahan bagi masyarakat dalam melaporkan permasalahan sampah sekaligus membantu pihak Dinas Lingkungan Hidup dalam menentukan prioritas penanganan laporan secara objektif. Arsitektur sistem mencakup pengelolaan data pengaduan, data pengguna, perhitungan skor prioritas, serta pengurutan laporan berdasarkan tingkat urgensi menggunakan algoritma Dynamic Priority Scheduling. Dalam implementasinya, sistem ini menggunakan framework Laravel (PHP) untuk mengelola backend, MySQL sebagai basis data, serta LeafletJS yang dipadukan dengan HTML, CSS, dan JavaScript untuk menghadirkan tampilan antarmuka yang interaktif dan responsif.

3.5.2 Implementasi Data

Data yang digunakan dalam penelitian ini terdiri dari data laporan pengaduan masyarakat serta data kriteria prioritas yang menjadi dasar penerapan algoritma. Data laporan mencakup informasi berupa alamat lokasi, deskripsi permasalahan, volume sampah, waktu pelaporan, dampak lingkungan, serta dokumentasi foto. Keseluruhan data tersebut disimpan dalam basis data MySQL dan menjadi masukan utama bagi sistem dalam menjalankan proses prioritisasi. Sementara itu, data kriteria prioritas ditetapkan berdasarkan empat parameter utama yaitu volume sampah, waktu pengaduan, jarak lokasi dari titik layanan, dan dampak lingkungan. Setiap parameter diberi bobot tertentu yang ditentukan melalui pertimbangan kebutuhan lapangan. Nilai prioritas laporan kemudian dihitung secara otomatis oleh sistem menggunakan persamaan Algoritma *Dynamic Priority Scheduling*, sehingga laporan dengan skor tertinggi akan ditempatkan pada urutan penanganan terdepan. Dengan integrasi kedua jenis data tersebut, sistem dapat memberikan hasil prioritas yang lebih akurat dan sesuai dengan kondisi nyata di lapangan

3.5.3 Implementasi Data

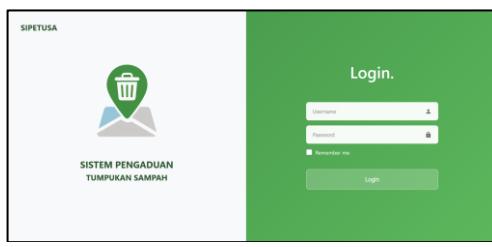
Data yang digunakan dalam penelitian ini terdiri dari data laporan pengaduan masyarakat serta data kriteria prioritas yang menjadi dasar penerapan algoritma. Data laporan mencakup informasi berupa alamat lokasi, deskripsi permasalahan, volume sampah, waktu pelaporan, dampak lingkungan, serta dokumentasi foto. Keseluruhan data tersebut disimpan dalam basis data MySQL dan menjadi masukan utama bagi sistem dalam menjalankan proses prioritisasi. Sementara itu, data kriteria prioritas ditetapkan berdasarkan empat parameter utama yaitu volume sampah, waktu pengaduan, jarak lokasi dari titik layanan, dan dampak lingkungan. Setiap parameter diberi bobot tertentu yang ditentukan melalui pertimbangan kebutuhan lapangan. Nilai prioritas laporan kemudian dihitung secara otomatis oleh sistem menggunakan persamaan Algoritma *Dynamic Priority Scheduling*, sehingga laporan dengan skor tertinggi akan ditempatkan pada urutan penanganan terdepan. Dengan integrasi kedua jenis data tersebut, sistem dapat memberikan hasil prioritas yang lebih akurat dan sesuai dengan kondisi nyata di lapangan.

3.5.4 Implementasi Interface

Antarmuka sistem dirancang untuk memberikan pengalaman penggunaan yang sesuai dengan kebutuhan dua jenis pengguna, yaitu masyarakat sebagai pelapor dan admin/petugas sebagai pengelola laporan. Pada sisi masyarakat, sistem menyediakan halaman utama untuk melakukan *login*, serta formulir pengaduan yang memfasilitasi pengisian detail laporan termasuk alamat, deskripsi masalah, volume sampah, dampak lingkungan, dan foto. Setelah pengaduan dikirimkan, masyarakat dapat memantau status laporannya melalui halaman khusus yang menampilkan informasi perkembangan penanganan. Pada sisi admin dan petugas, sistem dilengkapi dengan *dashboard* utama yang menyajikan ringkasan jumlah laporan, status penanganan, serta daftar prioritas pengaduan. Admin juga dapat mengelola data pengguna, memverifikasi laporan, dan memperbarui status pengaduan. Salah satu fitur inti yang disediakan adalah halaman analisis prioritas, di mana admin dapat melihat hasil perhitungan algoritma *Dynamic Priority Scheduling* dan urutan eksekusi laporan. Dengan rancangan antarmuka ini, baik masyarakat maupun petugas dapat menggunakan sistem dengan mudah sesuai kebutuhan masing-masing.

1. Halaman Login (Admin dan User)

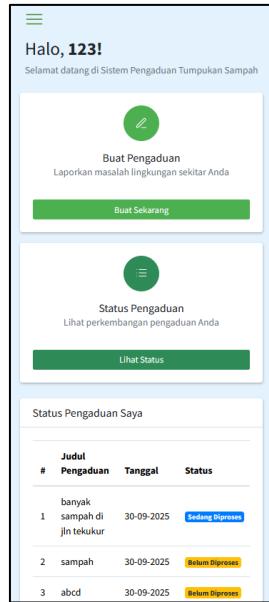
Adapun perancangan *interface* halaman *login* (admin dan *user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 13 berikut.



Gambar 12. Halaman Login (Admin dan User)

2. Halaman Beranda (*User*)

Adapun perancangan *interface* halaman beranda (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 14 berikut.



Gambar 13. Halaman Beranda (User)

3. Halaman Buat Pengaduan (*User*)

Adapun perancangan *interface* halaman Buat pengaduan (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 15 berikut.

Gambar 14. Halaman Buat Pengaduan (User)

4. Halaman Status Pengaduan (*User*)

Adapun perancangan *interface* halaman status pengaduan (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 16 berikut.

Gambar 15. Halaman Status Pengaduan (User)

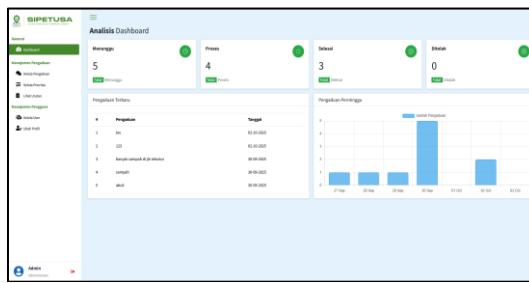
5. Halaman Ubah Profil (*User*)

Adapun perancangan *interface* halaman ubah profil (*user*) dari sistem yang akan dibangun dapat dilihat pada Gambar 17 berikut.

Gambar 16. Halaman Ubah Profil (User)

6. Halaman *Dashboard* (Admin)

Adapun perancangan *interface* halaman *dashboard* (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 18 berikut.



Gambar 17. Halaman Dashboard (Admin)

7. Halaman Kelola Pengaduan (Admin)

Adapun perancangan *interface* halaman kelola pengaduan (admin) dari sistem yang akan dibangun dapat dilihat pada 19 berikut.

Gambar 18. Halaman Kelola Pengaduan (Admin)

8. Halaman Proses Prioritas

Adapun perancangan *interface* halaman proses prioritas (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 20 berikut.

| No | Nama Pengguna | Jenis | Value | Volume | Sampaikan Langsung |
|----|---------------|-------|--------|--------|--------------------|
| 1 | Toto 1 | 1.00 | 10.000 | 2 | 2 |
| 2 | Lili 1 | 1.00 | 10.000 | 2 | 2 |
| 3 | Lili 2 | 1.00 | 10.000 | 2 | 2 |
| 4 | Lili 3 | 1.00 | 10.000 | 2 | 2 |
| 5 | Lili 4 | 1.00 | 10.000 | 2 | 2 |

Gambar 19. Halaman Proses Prioritas (Admin)

9. Halaman Lihat Urutan (Admin)

Adapun perancangan *interface* halaman kelola status pengaduan (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 21 berikut.

| No | Nama Pengguna | Target | Lahan | Volume | Status | Jenis | Sampaikan Langsung | Prioritas |
|----|---------------|---------------------|-------------------|--------|--------|-------|--------------------|-----------|
| 1 | Lili 1 | 2023-09-01 10:00:00 | 4.00000, 10.00000 | 2 | 10.000 | 4.00 | 2 | 100.000 |
| 2 | Lili 1 | 2023-09-01 10:00:00 | 4.00000, 10.00000 | 2 | 10.000 | 4.00 | 2 | 100.000 |
| 3 | Lili 2 | 2023-09-01 10:00:00 | 4.00000, 10.00000 | 2 | 10.000 | 4.00 | 2 | 100.000 |
| 4 | Lili 3 | 2023-09-01 10:00:00 | 4.00000, 10.00000 | 2 | 10.000 | 4.00 | 2 | 100.000 |
| 5 | Toto 1 | 2023-09-01 10:00:00 | 4.00000, 10.00000 | 2 | 10.000 | 4.00 | 2 | 100.000 |

Gambar 20. Halaman Lihat Urutan (Admin)

10. Halaman Halaman Kelola User (Admin)

Adapun perancangan *interface* halaman proses prioritas (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 22 berikut.

| No | Nama | Email | Role | Status | Aktif |
|----|--------|-----------------|---------------|--------|-------|
| 1 | Lili 1 | lili1@gmail.com | Administrator | Aktif | |
| 2 | Lili 2 | lili2@gmail.com | Administrator | Aktif | |
| 3 | Toto 1 | toto1@gmail.com | Administrator | Aktif | |

Gambar 21. Halaman Kelola User (Admin)

11. Halaman Ubah Profil (Admin)

Adapun perancangan *interface* halaman kelola data *user* (admin) dari sistem yang akan dibangun dapat dilihat pada Gambar 23 berikut.

Ubah Profil
Manajemen Pengguna —> Ubah Profil

Ubah Profil

Name
Admin

Email
admin@site.com

Role
Administrator

Kata Sandi Lama
Masukkan kata sandi lama

Kata Sandi Baru
Masukkan kata sandi baru

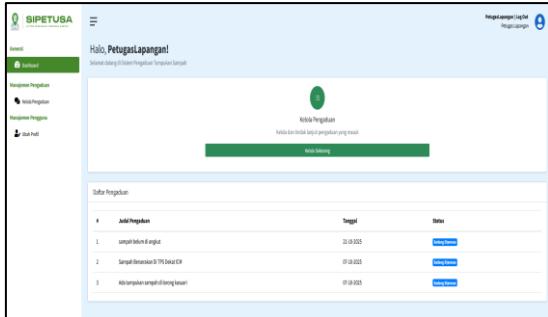
Konfirmasi Kata Sandi Baru
Ulangi kata sandi baru

Simpan Perubahan

Gambar 22. Halaman Ubah Profil (Admin)

12. Halaman *Dashboard* (Petugas Lapangan)

Adapun perancangan *interface* halaman *dashboard* (petugas lapangan) dari sistem yang akan dibangun dapat dilihat pada Gambar 24 berikut.



Gambar 23. Halaman Dashboard (Petugas Lapangan)

13. Halaman Daftar Pengaduan (Petugas Lapangan)

Adapun perancangan *interface* halaman daftar pengaduan (petugas lapangan) dari sistem yang akan dibangun dapat dilihat pada Gambar 25 berikut.

| Daftar Pengaduan | | | | | | | | |
|---|--------------|--------------------------------|------------|----------|-------------------|--|-------------------------|------------------------------|
| Manajemen Pengaduan — Aktivitas Pengaduan | | | | | | | | |
| No | Nama Pelapor | Lokasi | Tanggal | Volumen | Dampak Lingkungan | Dekripsi | Aksi | Status |
| 1 | S.B. Purnomo | 410702, 22233100 | 01-01-2025 | Besar | Tinggi | Sampah sudah berserakan dan merusak jalur jalan | <button>Simpan</button> | Belum Diketahui |
| 2 | Ahmad Pust | 41030919700303, 22132034012050 | 01-01-2025 | Sebagian | Tinggi | ada tumpukan sampah di lorong Kasuari, sampahnya sudah beberapa hari bertumpuk | <button>Simpan</button> | Belum Diketahui |
| 3 | Bach Hartono | 41030919700307, 22132034012051 | 01-01-2025 | Besar | Tinggi | Sampah sudah berserakan dan sudah 2 hari tidak diangkut | <button>Simpan</button> | Belum Diketahui |

Gambar 24. Halaman Dashboard (Petugas Lapangan)

3.5.5 Skenario Implementasi Algoritma Dynamic Priority Scheduling

Algoritma *Dynamic Priority Scheduling* diterapkan pada sistem untuk mengatur urutan penanganan laporan berdasarkan tingkat urgensi. Proses perhitungan prioritas dilakukan dengan mempertimbangkan empat parameter utama, yaitu volume sampah, waktu pelaporan, jarak lokasi, dan dampak lingkungan. Setiap parameter diberikan bobot tertentu yang menjadi faktor penentu skor akhir. Semakin besar volume sampah dan semakin lama laporan dibiarkan tanpa penanganan, maka nilai prioritasnya akan semakin tinggi. Demikian pula, lokasi yang lebih dekat dengan titik layanan akan memperoleh skor tambahan karena lebih mudah dijangkau petugas, sedangkan laporan dengan dampak lingkungan serius juga memperoleh bobot tinggi.

Hasil perhitungan dari algoritma ini menghasilkan urutan eksekusi laporan yang ditampilkan pada *dashboard* admin, di mana laporan dengan nilai tertinggi ditempatkan pada posisi paling atas. Dengan demikian, sistem dapat membantu Dinas Lingkungan Hidup dalam menentukan langkah penanganan yang lebih cepat, objektif, dan efisien sesuai kondisi di lapangan. Adapun skenario implementasinya dapat dijelaskan sebagai berikut.

1. Data Awal Pengaduan

Untuk mendemonstrasikan alur kerja algoritma secara lengkap, digunakan dataset yang terdiri dari sepuluh pengaduan sampah. Setiap pengaduan memiliki identitas pelapor, serta judul pengaduan yang mencerminkan lokasi dan permasalahan yang dilaporkan. Misalnya, pengaduan dengan ID P001 diajukan oleh A.F. mengenai adanya tumpukan sampah di lorong Kasuari. Selanjutnya, P002 dilaporkan oleh S.R. terkait sampah berserakan di sekitar area ICM. P003 berasal dari B.H., yang menyampaikan bahwa sampah belum diangkut di wilayahnya. Kemudian, F.S. pada pengaduan P004 melaporkan sampah yang menumpuk di Jalan Lastirtada.

Pengaduan berikutnya, yaitu P005, berasal dari A.W., yang menyampaikan adanya sampah berserakan di tempat pembuangan sementara (TPS) dekat Warung Kopi 46. P006 diajukan oleh D.L., terkait sampah liar di TPS persimpangan Perdos. Pada pengaduan P007, H.P. melaporkan bahwa TPS di depan Klik Haji Lapato sudah penuh. Sementara itu, N.A. sebagai pelapor P008 menyampaikan bahwa sampah menggunung di dekat SDN 51. Pengaduan P009 dari I.R. menyebutkan bahwa TPS mengalami overflow di Lorong Kerisma 2. Terakhir, R.K. dalam laporan P010 menginformasikan adanya sampah yang berceceran kembali di sekitar area ICM.

Seluruh data ini kemudian digunakan sebagai masukan awal dalam penerapan algoritma untuk menghitung dan mengurutkan prioritas penanganan berdasarkan parameter-parameter yang telah ditentukan sebelumnya.

Tabel 1. Data Awal Pengaduan

| Koordinat | Waktu Masuk | Volume Sampah | Sub-Kriteria Dampak Lingkungan | Total Bobot Dampak |
|---|-------------|---------------|--|--------------------|
| -4.011855978741201, 122.5327193381608 | 6:30 | Besar (3) | Dekat sekolah, Bau menyengat, Menarik hama | 7 |
| -4.006326513620645, 122.53036704626898 | 7:15 | Sedang (2) | Menghalangi jalan, Sampah berserakan | 5 |
| -4.000472106996737, 122.52728354675273 | 6:00 | Besar (3) | Dekat RS/Puskesmas, Bau menyengat, Mencemari saluran, Menarik hama | 9 |
| -4.004532901535692, 122.5290585697096 | 7:45 | Sedang (2) | Dekat pemukiman padat, Bau menyengat | 4 |
| -4.0094664493886425, 122.53119047647972 | 8:00 | Kecil (1) | Genangan air | 1 |
| -4.011970115111293, 122.53083677944659 | 6:45 | Besar (3) | Menghalangi jalan, Dekat pasar, Sampah berserakan | 7 |
| -4.013589347953546, 122.5306438749707 | 7:00 | Sedang (2) | Dekat dengan rumah sakit/puskesmas, Bau menyengat | 5 |
| -4.01489769569965, 122.53253413557624 | 6:15 | Besar (3) | Dekat sekolah, Menghalangi jalan, Menarik hama | 8 |
| -4.014691148302068, 122.53249257930145 | 7:30 | Sedang (2) | Dekat pemukiman padat, Mencemari saluran | 4 |
| -4.007566164950778, 122.53106311576741 | 8:15 | Kecil (1) | Sampah berserakan | 2 |

2. Penentuan Kriteria dan Bobot Prioritas

Berdasarkan perancangan sistem yang telah dijelaskan pada subbab sebelumnya, terdapat empat kriteria utama yang digunakan dalam perhitungan prioritas, yaitu volume sampah, waktu pengaduan, jarak lokasi, dan dampak lingkungan. Setiap kriteria memiliki bobot yang mencerminkan tingkat kepentingannya dalam menentukan urgensi penanganan. Kriteria dan bobot yang digunakan dapat dilihat pada Tabel 4 berikut.

Tabel 2. Bobot Prioritas

| Kriteria | Bobot | Skala Penilaian |
|-------------------|-------|--|
| Volume Sampah | 3 | Besar=3, Sedang = 2, Kecil 1 |
| Waktu Laporan | 2 | Selisih waktu laporan masuk ke dalam jam |
| Jarak Lokasi | 1 | 1/ jarak (Dalam kilometer) |
| Dampak Lingkungan | 4 | Tinggi = 3, Sedang = 2, Rendah = 1 |

Bobot dampak lingkungan memiliki nilai tertinggi (4) karena menyangkut aspek kesehatan masyarakat dan keselamatan publik. Volume sampah memiliki bobot 3 karena menentukan kapasitas armada yang dibutuhkan. Waktu pengaduan memiliki bobot 2 untuk memastikan laporan yang lebih lama tidak terabaikan. Jarak lokasi memiliki bobot terendah (1) namun tetap diperhitungkan untuk efisiensi distribusi armada.

a. Konversi Nilai Dampak Lingkungan

Sistem melakukan kategorisasi dampak lingkungan berdasarkan total bobot dari sub-kriteria yang dipilih oleh pengguna. Kategorisasi dampak lingkungan dapat dilihat pada Tabel 5 dan Tabel 6 berikut.

Tabel 3. Sub Kriteria Dampak Lingkungan

| No | Sub-Kriteria | Bobot Nilai |
|----|------------------------------------|-------------|
| 1. | Dekat dengan sekolah/kampus | 3 |
| 2. | Dekat dengan rumah sakit/puskesmas | 3 |
| 3. | Dekat dengan tempat ibadah | 3 |
| 4. | Menghalangi akses jalan | 3 |
| 5. | Bau menyengat mengganggu warga | 2 |
| 6. | Menarik Hama (lalat, tikus, kecoa) | 2 |
| 7. | Sampah berserakan ke jalan | 2 |
| 8. | Dekat dengan pasar tradisional | 2 |
| 9. | Mencemari saluran air/got | 2 |
| 10 | Dekat dengan pemukiman padat | 2 |

Tabel 4. Bobot Sub-Kriteria Dampak Lingkungan

| ID Laporan | Total Bobot Sub-Kriteria | Kategori Dampak | Nilai Untuk Algoritma (Di) |
|------------|--------------------------|-----------------|----------------------------|
| P001 | 7 | Sedang | 2 |
| P002 | 5 | Sedang | 2 |
| P003 | 9 | Tinggi | 3 |
| P004 | 4 | Sedang | 2 |
| P005 | 1 | Rendah | 1 |
| P006 | 7 | Sedang | 2 |
| P007 | 5 | Sedang | 2 |
| P008 | 8 | Tinggi | 3 |
| P009 | 4 | Sedang | 2 |
| P010 | 2 | Rendah | 1 |

Berdasarkan Tabel 7 sistem mengkategorikan dampak lingkungan menjadi tiga tingkat: Rendah (bobot 1-3), Sedang (bobot 4-7), dan Tinggi (bobot ≥ 8). Nilai kategori ini kemudian digunakan sebagai input parameter Di dalam rumus perhitungan prioritas.

b. Perhitungan Jarak Menggunakan Formula Haversine

Sistem menghitung jarak antara lokasi pengaduan dengan pusat pengangkutan sampah (Pool Armada Dinas Lingkungan Hidup Kota Kendari) menggunakan formula Haversine. Koordinat titik referensi pusat pengangkutan berada pada latitude -4.014240 dan longitude 122.525267. Formula Haversine yang digunakan adalah sebagai berikut:

$$\alpha = \sin^2 \left(\frac{\Delta \text{lat}}{2} \right) + \cos(\text{Lat}_1) \times \cos(\text{Lat}_2) \times \sin^2 \left(\frac{\Delta \text{lon}}{2} \right) \quad (5.1)$$

$$C = 2 \times \arctan 2(\sqrt{\alpha}; \sqrt{1 - \alpha}) \quad (5.2)$$

$$\text{jarak} = R \times C$$

Di mana R adalah jari-jari bumi

Contoh perhitungan jarak untuk P001;

Koordinat P001: (-4.011698, 122.532634)

Koordinat Pusat Pengangkutan: (-4.014240, 122.525267)

$$\Delta \text{lat} = -4.014240 - (-4.011698) = -0.002542 \text{ o} = -0.0000444 \text{ radian}$$

$$\Delta \text{lon} = 122.525267 - 122.532634 = -0.007367 \text{ o} = -0.0001286 \text{ radian}$$

$$\alpha = \sin^2 (-0.0000222) + \cos(-4.011698^\circ) \times \cos(-4.014240^\circ) \times \sin^2 (-0.0000643)$$

$$\alpha = 0.000000000493 + (0.9978 \times 0.9978 \times 0.0000000041) = 0.0000000089$$

$$C = 2 \times \arctan 2(\sqrt{0.000000089}, \sqrt{0.9999999911}) = 0.0001886$$

$$jarak = 6371 \times 0.0001886 = 0.82 \text{ km}$$

Tabel 5. Jarak Lokasi Pengaduan dari Pusat Pengangkutan

| ID Laporan | Koordinat | Jarak (km) |
|------------|-----------------------|------------|
| P001 | -4.011698, 122.532634 | 0.82 |
| P002 | -4.007582, 122.531089 | 0.88 |
| P003 | -4.006326, 122.530367 | 0.95 |
| P004 | -4.000472, 122.527283 | 1.54 |
| P005 | -4.004532, 122.529058 | 1.08 |
| P006 | -4.009466, 122.531190 | 0.66 |
| P007 | -4.011970, 122.530836 | 0.61 |
| P008 | -4.013589, 122.530643 | 0.56 |
| P009 | -4.014897, 122.532534 | 0.73 |
| P010 | -4.007582, 122.531089 | 0.88 |

c. Perhitungan Waktu Tunggu Pengaduan

Tabel 6. Perhitungan Waktu Tunggu Pengaduan

| ID Laporan | Waktu Masuk | Waktu Pemrosesan | Waktu Tunggu dalam Jam |
|------------|-------------|------------------|------------------------|
| P001 | 6:30 | 10.00 | 3.5 |
| P002 | 7:15 | 10.00 | 2.75 |
| P003 | 6:00 | 10.00 | 4.0 |
| P004 | 7:45 | 10.00 | 2.25 |
| P005 | 8:00 | 10.00 | 2.0 |
| P006 | 6:45 | 10.00 | 3.25 |
| P007 | 7:00 | 10.00 | 3.0 |
| P008 | 6:15 | 10.00 | 3.75 |
| P009 | 7:30 | 10.00 | 2.5 |
| P010 | 8:15 | 10.00 | 1.75 |

Berdasarkan Tabel 5.8, pengaduan P003 memiliki waktu tunggu terlama (4.0 jam), sedangkan P010 memiliki waktu tunggu tersingkat (1.75 jam).

3. Perhitungan Nilai Prioritas Menggunakan Algoritma Dynamic Priority Scheduling

Setelah seluruh parameter diperoleh, sistem menghitung nilai prioritas untuk setiap pengaduan menggunakan rumus Algoritma *Dynamic Priority Scheduling* sebagai berikut.

$$P_i = (V_i \times W_v) + (W_i \times W_w) + \left(\frac{1}{J_i} \times W_j \right) + (D_i \times W_d) \quad (5.3)$$

Keterangan:

- P_i = Prioritas untuk pengaduan i .
- V_i = Volume sampah pada pengaduan i , yang bernilai 1 (kecil), 2 (sedang), atau 3 (besar).
- W_i = Waktu pengaduan pada pengaduan i (dalam satuan jam atau menit sejak laporan dibuat).
- J_i = Jarak lokasi pengaduan i (dalam satuan meter atau kilometer dari pusat pengangkutan sampah).
- D_i = Skor dampak lingkungan.
- W = Bobot untuk masing-masing faktor (volume sampah, waktu pengaduan, jarak lokasi, dan dampak lingkungan).

Perhitungan Prioritas untuk P003:

Data P003:

$V_i = 3$ (Volume Besar)

$W_i = 4.0$ jam

$J_i = 0.95$ km

$D_i = 3$ (Dampak Tinggi)

$$P_{003} = (3 \times 3) + (4.0 \times 2) + \left(\frac{1}{0.95} \times 1 \right) + (3 \times 4)$$

$$P_{003} = 9 + 8 + 1.058 + 12 = 30.053$$

Hasil perhitungan prioritas untuk seluruh pengaduan disajikan pada Tabel 10 berikut.

Tabel 7. Hasil Perhitungan Prioritas

| ID | V_i | W_i (Jam) | J_i (km) | D_i | $(V_i \times W_v)$ | $(W_i \times W_v)$ | $(1/J_i \times W_j)$ | $(D_i \times W_v)$ | Nilai Prioritas (P_i) |
|------|-------|-------------|------------|-------|--------------------|--------------------|----------------------|--------------------|---------------------------|
| P001 | 3 | 3.5 | 0.82 | 2 | 9 | 7.0 | 1.220 | 8 | 25.220 |
| P002 | 2 | 2.75 | 0.88 | 2 | 6 | 5.5 | 1.136 | 8 | 20.636 |
| P003 | 3 | 4.0 | 0.95 | 3 | 9 | 8.0 | 1.053 | 12 | 30.053 |
| P004 | 2 | 2.25 | 1.54 | 2 | 6 | 4.5 | 0.649 | 8 | 19.149 |
| P005 | 1 | 2.0 | 1.08 | 1 | 3 | 4.0 | 0.926 | 4 | 11.926 |
| P006 | 3 | 3.25 | 0.66 | 2 | 9 | 6.5 | 1.515 | 8 | 25.015 |
| P007 | 2 | 3.0 | 0.61 | 2 | 6 | 6.0 | 1.639 | 8 | 21.639 |
| P008 | 3 | 3.75 | 0.56 | 3 | 9 | 7.5 | 1.786 | 12 | 30.286 |
| P009 | 2 | 2.5 | 0.73 | 2 | 6 | 5.0 | 1.370 | 8 | 20.370 |
| P010 | 1 | 1.75 | 0.88 | 1 | 3 | 3.5 | 1.136 | 4 | 11.636 |

Tabel 8. Urutan Eksekusi Penanganan Pengaduan Berdasarkan Prioritas

| Urutan | ID Laporan | Nama Pelapor | Lokasi | Nilai Prioritas | Status |
|--------|------------|--------------|--------------------------|-----------------|----------|
| 1 | P008 | NA | Dekat SDN 51 Kendari | 30.286 | Menunggu |
| 2 | P003 | BH | Tps Dekat Icm | 30.053 | Menunggu |
| 3 | P001 | AH | Lorong Kasuari | 25.220 | Menunggu |
| 4 | P006 | DL | Persimpangan Perdos | 25.015 | Menunggu |
| 5 | P007 | HP | Depan Klinik Haji Lapato | 21.639 | Menunggu |
| 6 | P002 | SR | Dekat ICM | 20.636 | Menunggu |
| 7 | P009 | IR | Lorong Kerisma 2 | 20.370 | Menunggu |
| 8 | P004 | FS | Jalan Lastirtada | 19.149 | Menunggu |
| 9 | P005 | AW | Dekat Warkop 46 | 11.926 | Menunggu |
| 10 | P010 | RK | Dekat ICM | 11.636 | Menunggu |

Berdasarkan hasil perhitungan yang disajikan pada Tabel 11 pengaduan P008 memperoleh nilai prioritas tertinggi (30.286) dan menjadi urutan pertama dalam eksekusi penanganan. Hal ini disebabkan oleh kombinasi dari empat faktor, yaitu: (1) volume sampah yang besar (3), (2) waktu tunggu yang cukup lama (3.75 jam), (3) jarak lokasi yang paling dekat dari pusat pengangkutan (0.56 km), dan (4) dampak lingkungan tinggi (3). Kontribusi terbesar pada nilai prioritas P008 berasal dari parameter dampak lingkungan (12 poin), waktu tunggu (7.5 poin), dan faktor jarak yang sangat dekat (1.786 poin).

Pengaduan P003 berada pada urutan kedua dengan nilai prioritas 30.053. Meskipun memiliki waktu tunggu terlama (4.0 jam) dan dampak lingkungan tinggi (3), jarak lokasi P003 sedikit lebih jauh (0.95 km) dibandingkan P008, sehingga nilai prioritasnya sedikit lebih rendah. Hal ini menunjukkan bahwa kombinasi seluruh kriteria secara bersamaan menentukan urutan prioritas, bukan hanya satu kriteria dominan.

Sebaliknya, pengaduan P010 memperoleh nilai prioritas terendah (11.636) karena memiliki volume sampah kecil (1), waktu tunggu tersingkat (1.75 jam), dan dampak lingkungan rendah (1). Hal ini menunjukkan bahwa sistem berhasil membedakan tingkat urgensi pengaduan secara objektif berdasarkan kriteria yang telah ditetapkan.

Hasil perhitungan juga menunjukkan bahwa parameter dampak lingkungan memiliki pengaruh paling signifikan terhadap nilai prioritas, sesuai dengan bobot tertinggi yang diberikan (4). Hal ini sejalan dengan tujuan sistem untuk memprioritaskan penanganan tumpukan sampah yang berpotensi menimbulkan gangguan kesehatan masyarakat dan keselamatan publik. Faktor jarak juga memberikan kontribusi yang cukup signifikan, terutama untuk lokasi yang sangat dekat dengan pusat pengangkutan, seperti terlihat pada kasus P008 yang memiliki komponen jarak sebesar 1.786 poin.

4. Kode Algoritma Dynamic Priority Scheduling

```
public function prosesPrioritas()
{
    $Wv = 3;
    $Ww = 2;
    $Wj = 1;
    $Wd = 4;
    $pengaduans = Pengaduan::where ('status', 'Menunggu')->get();
    $bobotKriteria = [
        ['no' => 1, 'bobot' => 3],
        ['no' => 2, 'bobot' => 3],
        ['no' => 3, 'bobot' => 3],
        ['no' => 4, 'bobot' => 3],
        ['no' => 5, 'bobot' => 2],
        ['no' => 6, 'bobot' => 2],
        ['no' => 7, 'bobot' => 2],
        ['no' => 8, 'bobot' => 2],
        ['no' => 9, 'bobot' => 2],
        ['no' => 10, 'bobot' => 2],
    ];
    $totalBobot = array_sum(array_column($bobotKriteria, 'bobot'));
    if ($totalBobot <= 3) {
        $kategoriDampak = 'Rendah';
        $nilaiDi = 1;
    } elseif ($totalBobot >= 4 && $totalBobot <= 7) {
        $kategoriDampak = 'Sedang';
        $nilaiDi = 2;
    } else {
        $kategoriDampak = 'Tinggi';
        $nilaiDi = 3;
    }
    foreach ($pengaduans as $pengaduan) {
        $Vi = $pengaduan->volume;
        $Wi = $pengaduan->created_at->diffInMinutes(now()) / 60;
        $Ji = $pengaduan->jarak > 0 ? $pengaduan->jarak : 1;
        $Di = $nilaiDi;
        $Pi = ($Vi * $Wv) + ($Wi * $Ww) + ((1 / $Ji) * $Wj) + ($Di * $Wd);
        $pengaduan->prioritas = $Pi;
        $pengaduan->save();
    }
}
```

```

return redirect()->route('lihat.urutan')->with('success', 'Proses prioritas
berhasil dijalankan!');
}
public function export_pdf()
{
$pengaduan = Pengaduan::with('user')
->orderByRaw("CASE
WHEN status = 'Menunggu' THEN 1
WHEN status = 'Diproses' THEN 2
WHEN status = 'Selesai' THEN 3
ELSE 4
END")
->get();
$filename = 'Laporan Pengaduan_ ' . now()->format('Y-m-d_H-i-s') . '.pdf';
$pdf = Pdf::loadView('export_pdf', compact('pengaduan'))
->setPaper('a4', 'landscape');
return $pdf->download($filename);
}
}

```

3.5.6 User Acceptance Testing (UAT)

Pengujian User Acceptance Testing (UAT) merupakan tahap akhir dalam pengembangan perangkat lunak yang bertujuan untuk memastikan bahwa sistem yang dikembangkan telah sesuai dengan kebutuhan dan harapan pengguna. Penilaian hasil UAT dalam penelitian ini mengacu pada kriteria tingkat penerimaan pengguna terhadap sistem dalam bentuk persentase. Rentang nilai diklasifikasikan ke dalam lima kategori, yaitu 20,00% hingga 36,00% dikategorikan sebagai *tidak baik*, 36,01% hingga 52,00% sebagai *kurang baik*, 52,01% hingga 68,00% sebagai *cukup*, 68,01% hingga 84,00% sebagai *baik*, dan 84,01% hingga 100% sebagai *sangat baik*. Untuk mendapatkan nilai akhir, setiap jawaban responden diberi bobot menggunakan skala Likert, yaitu *sangat setuju (SS)* dengan skor 5, *setuju (S)* dengan skor 4, *netral (N)* dengan skor 3, *tidak setuju (TS)* dengan skor 2, dan *sangat tidak setuju (STS)* dengan skor 1. Seluruh skor kemudian diakumulasi dan dikonversi ke dalam bentuk persentase untuk menentukan kategori penerimaan sistem.

Pengujian UAT dilakukan terhadap dua kelompok pengguna, yaitu admin atau petugas dari Dinas Lingkungan Hidup dan pengguna umum dari masyarakat Kelurahan Kambu. Untuk kategori admin/petugas, jumlah responden sebanyak lima orang dengan 29 pertanyaan yang mencakup aspek fungsional, usability, kompatibilitas, performa sistem, serta kesesuaian proses bisnis. Hasil dari jawaban responden menunjukkan bahwa sebanyak 97 respon diberikan dengan pilihan *sangat setuju*, 41 dengan *setuju*, dan 2 dengan *netral*. Tidak terdapat jawaban *tidak setuju* maupun *sangat tidak setuju*. Setelah dilakukan perhitungan, total skor yang diperoleh adalah 655 dari skor maksimal 700. Nilai persentase akhir adalah 93,6%, yang berada dalam kategori *sangat baik*. Hal ini menunjukkan bahwa sistem dinilai sangat memadai dalam membantu tugas admin dalam menangani laporan pengaduan sampah secara efisien dan sistematis.

Untuk kelompok pengguna dari masyarakat umum, jumlah responden juga sebanyak lima orang dengan total 28 pertanyaan yang difokuskan pada kemudahan penggunaan, fungsionalitas, aksesibilitas, serta kompatibilitas sistem. Respon yang diterima terdiri dari 70 jawaban *sangat setuju*, 61 jawaban *setuju*, dan 9 jawaban *netral*, tanpa ada jawaban *tidak setuju* maupun *sangat tidak setuju*. Perhitungan skor menunjukkan bahwa total nilai yang diperoleh adalah 621 dari skor maksimal 700, dengan persentase akhir sebesar 88,8%. Nilai ini juga termasuk dalam kategori *sangat baik*.

Berdasarkan hasil UAT dari kedua kategori pengguna, sistem pengaduan tumpukan sampah berbasis web yang menggunakan algoritma Dynamic Priority Scheduling berhasil memperoleh penilaian sangat baik dengan skor 88,8% dari masyarakat dan 93,6% dari admin/petugas. Temuan ini menunjukkan bahwa sistem tidak hanya memenuhi kebutuhan fungsional, tetapi juga mampu meningkatkan kenyamanan

penggunaan bagi masyarakat serta mendukung efektivitas kerja Dinas Lingkungan Hidup Kota Kendari dalam memprioritaskan dan menangani laporan tumpukan sampah di wilayah Kelurahan Kambu.

SIMPULAN

Berdasarkan hasil penelitian dan pembahasan mengenai penerapan Algoritma Dynamic Priority Scheduling pada sistem pengaduan tumpukan sampah berbasis web di Kelurahan Kambu, Kota Kendari, dapat disimpulkan sebagai berikut.

1. Sistem pengaduan tumpukan sampah berbasis web telah berhasil dibangun menggunakan framework Laravel, PHP, MySQL, dan LeafletJS dengan metode pengembangan *Rational Unified Process* (RUP). Sistem ini menyediakan platform digital yang memudahkan masyarakat dalam melaporkan pengaduan tumpukan sampah secara terstruktur melalui formulir yang mencakup lokasi kejadian, deskripsi permasalahan, volume sampah, dampak lingkungan, dan dokumentasi foto. Bagi admin dan petugas Dinas Lingkungan Hidup, sistem menyediakan fitur pengelolaan pengaduan yang komprehensif, termasuk *dashboard* monitoring, pembaruan status, dan manajemen pengguna.
2. Implementasi Algoritma Dynamic Priority Scheduling terbukti efektif dalam mengatur urutan penanganan pengaduan tumpukan sampah secara dinamis dan objektif. Algoritma ini menghitung nilai prioritas berdasarkan empat parameter utama dengan pembobotan yang telah ditetapkan: dampak lingkungan (bobot 4), volume sampah (bobot 3), waktu pengaduan (bobot 2), dan jarak lokasi (bobot 1). Hasil perhitungan menghasilkan urutan eksekusi laporan yang logis sesuai tingkat urgensi, di mana pengaduan dengan nilai prioritas tertinggi ditempatkan pada posisi teratas untuk penanganan segera. Sistem secara otomatis menghitung jarak menggunakan formula Haversine dan mengonversi nilai dampak lingkungan berdasarkan sub-kriteria yang dipilih pengguna, sehingga menghasilkan prioritas yang akurat dan sesuai dengan kondisi lapangan.
3. Hasil pengujian *User Acceptance Testing* (UAT) menunjukkan tingkat penerimaan yang sangat baik dari kedua kategori pengguna, dengan perolehan nilai 88.8% untuk pengguna masyarakat dan 93.6% untuk admin/petugas, yang keduanya masuk dalam kategori "Sangat Baik". Penilaian positif ini mencakup aspek fungsionalitas sistem, kemudahan penggunaan (usability), kompatibilitas, performa sistem, kesesuaian proses bisnis, dan aksesibilitas. Hasil ini mengindikasikan bahwa sistem telah memenuhi kebutuhan dan ekspektasi pengguna akhir, mampu meningkatkan efektivitas dan transparansi dalam pengelolaan pengaduan tumpukan sampah, serta mendukung pengambilan keputusan yang lebih cepat dan objektif bagi Dinas Lingkungan Hidup Kota Kendari dalam menentukan prioritas penanganan laporan di Kelurahan Kambu

DAFTAR PUSTAKA

- Aliyah, A., Hartono, N., Muin, A. A., 2025, Penggunaan User Acceptance Testing (UAT) pada Pengujian Sistem Informasi Pengelolaan Keuangan dan Inventaris Barang, *Jurnal Sains dan Teknologi Informasi*, 3, 1, 84-100.
- Avif Nurrohman, Ismail Abdurrozzak Zulkarnain, K. N., 2024, *PENERAPAN ALGORITMA DYNAMIC PRIORITY SCHEDULING UNTUK SISTEM PENJADWALAN ADAPTIF*, 20, 2, 12–21.
- Erwin Ferdiansa, Febriasyah, A. D. A., 2022, *Kata kunci : Smart Living , Pengaduan Sampah, Jarak Terdekat, Makassar, Haversine Formula*, 3, 2, 1–13.
- Fariz, M., dan Riswandha, M. N., 2024, Implementasi Metode Rational Unified Process(Rup) Untuk Rancang Bangun Sistem Informasi Pengajuan Cuti Menggunakan Framework Laravel, *Josiat*, 1, 2, 105–115.
- Fatmasari, F., dan Sauda, S., 2020, Pemodelan Unified Modeling Language Sistem Informasi Enterprise Resource Planning, *Jurnal Media Informatika Budidarma*, 4, 2, 429. <https://doi.org/10.30865/mib.v4i2.2022>
- Labibah, L., dan Pulungan, M. A., 2025, *Evaluasi Pengelolaan Sampah Menggunakan Pemodelan Sistem Dinamis di Kabupaten Ponorogo*, 10, 1, 24–32.

- Masyfa, F. H., Kartikasari, D. P., dan Tibyani, -, 2023, Penjadwalan dan Pelaporan Menggunakan Dynamic Priority Scheduling dan Geolocation untuk Keamanan Lingkungan, *Techno.Com*, 22, 1, 195–206. <https://doi.org/10.33633/tc.v22i1.7132>
- Nasution A.F.M, Suendri, dan Muliani Harahap, A., 2023, Customer Service Information System Using Dynamic Priority Scheduling Algorithm At PT Sumatra Sistem Integrasi, *Journal of Information Systems and Technology Research*, 2, 1, 25–37. <https://doi.org/10.55537/jistr.v2i1.324>
- Setyawati, R., dan Maulachela, A. B., 2020, Penerapan Algoritma Dynamic Priority Scheduling pada Antrian Pencucian Mobil, *JTIM: Jurnal Teknologi Informasi dan Multimedia*, 2, 1, 29–35. <https://doi.org/10.35746/jtim.v2i1.85>
- Suhendri Ergie, J., 2024, *Aplikasi Pengaduan Tumpukan Sampah dengan Algoritma Analytical Hierarchy Process (AHP) untuk Menentukan Prioritas Layanan Hierarchy Process (AHP) untuk menentukan prioritas penanganan pengaduan . AHP dan menangani aduan masyarakat dengan lebih efektif*, 4.