

PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI TICKETING MAINTENANCE BERBASIS WEB DI RST WIJAYAKUSUMA PURWOKERTO

Didik Prasetyo, Lutfiati Nur Afifah, Ridzal Achmad Fauzi

Informasika, Sistem Informasi, Universitas Bina Sarana Informatika, Indonesia

Email: didikprasetyo835@gmail.com, lutfiatinurafifah477@gmail.com, ridzalachmad22@gmail.com

Informasi	Abstract
Volume : 3 Nomor : 10 Bulan : Oktober Tahun : 2026 E-ISSN : 3062-9624	<i>The rapid development of information technology has encouraged hospitals to implement information systems that improve the effectiveness and efficiency of healthcare services, including facility and equipment maintenance management. At RST Wijayakusuma Purwokerto, facility damage reporting is still conducted manually through verbal communication, instant messaging applications, and handwritten records, resulting in delays in maintenance handling, difficulties in monitoring work progress, and inadequate documentation of maintenance history. This study aims to design and implement a Web-Based Maintenance Ticketing Information System at RST Wijayakusuma Purwokerto. The research employed the Research and Development (R&D) method using the Waterfall software development model, which consists of requirements analysis, system design, implementation, testing, and maintenance. Data were collected through observation, interviews, and literature review. The system was designed using Unified Modeling Language (UML), implemented using the Laravel framework with MySQL as the database management system, and tested using the Black Box Testing method. The research resulted in a web-based information system with three types of users: administrators, technicians, and users (reporters). The system provides features for damage reporting, ticket management, technician assignment, work status updates, maintenance monitoring, user management, division management, and maintenance report generation. The implementation of the system supports a more effective, efficient, transparent, and well-documented maintenance management process, facilitates real-time monitoring of maintenance activities, and contributes to improving service quality at RST Wijayakusuma Purwokerto.</i> Keyword: information system, maintenance ticketing, hospital, Waterfall, Laravel.

Abstrak

Perkembangan teknologi informasi mendorong rumah sakit untuk menerapkan sistem informasi yang mampu meningkatkan efektivitas dan efisiensi pelayanan, termasuk dalam pengelolaan maintenance fasilitas dan peralatan. Di RST Wijayakusuma Purwokerto, proses pelaporan kerusakan masih dilakukan secara manual melalui komunikasi lisan, aplikasi pesan instan, dan pencatatan manual sehingga menyebabkan keterlambatan penanganan, kesulitan monitoring, serta kurang optimalnya dokumentasi riwayat perbaikan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan Sistem Informasi Ticketing Maintenance Berbasis Web di RST Wijayakusuma Purwokerto. Metode penelitian yang digunakan adalah Research and Development (R&D) dengan model pengembangan perangkat lunak Waterfall yang meliputi analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Pengumpulan data dilakukan melalui observasi, wawancara, dan studi pustaka. Perancangan sistem menggunakan Unified Modeling Language (UML), implementasi sistem

menggunakan framework Laravel dengan basis data MySQL, serta pengujian dilakukan menggunakan metode Black Box Testing. Hasil penelitian menghasilkan sistem informasi berbasis web yang memiliki tiga jenis pengguna, yaitu admin, teknisi, dan user (pelapor). Sistem menyediakan fitur pelaporan kerusakan, pengelolaan tiket, penugasan teknisi, pembaruan status pekerjaan, monitoring pekerjaan, manajemen pengguna, manajemen divisi, serta pembuatan laporan maintenance. Implementasi sistem ini mampu mendukung pengelolaan maintenance secara lebih efektif, efisien, transparan, dan terdokumentasi sehingga memudahkan monitoring proses perbaikan serta meningkatkan kualitas pelayanan di RST Wijayakusuma Purwokerto.

Kata Kunci : *sistem informasi, ticketing maintenance, rumah sakit, Waterfall, Laravel.*

A. PENDAHULUAN

Perkembangan teknologi informasi telah membawa dampak signifikan di berbagai sektor, termasuk bidang kesehatan. Rumah sakit sebagai institusi pelayanan kesehatan dituntut memanfaatkan teknologi informasi untuk meningkatkan efektivitas, efisiensi, dan kualitas pelayanan. Salah satu penerapan yang banyak digunakan adalah sistem informasi berbasis web, yang mampu mengintegrasikan pengelolaan data secara terpusat, mempercepat penyampaian informasi, serta mendukung pengambilan keputusan secara tepat. Sistem ini juga meningkatkan kualitas pelayanan melalui proses bisnis yang lebih cepat, akurat, dan terdokumentasi dengan baik.

Salah satu kegiatan yang berperan penting dalam kelancaran operasional rumah sakit adalah proses maintenance terhadap fasilitas, peralatan medis, sarana pendukung, dan infrastruktur teknologi informasi (Jurnal et al., 2026). Proses maintenance yang masih dilakukan secara manual sering menimbulkan berbagai permasalahan, seperti keterlambatan penanganan kerusakan, kesulitan memonitor status pekerjaan teknisi, kurangnya dokumentasi riwayat perbaikan, hingga lambatnya penyusunan laporan. Kondisi ini dapat menghambat pelayanan rumah sakit karena kerusakan fasilitas tidak segera ditangani secara optimal. Penelitian sebelumnya menunjukkan bahwa digitalisasi proses maintenance melalui sistem informasi berbasis web mampu meningkatkan efektivitas pengelolaan data, mempermudah penjadwalan pekerjaan, serta mengurangi risiko kehilangan data historis perawatan.

Berdasarkan observasi di RST Wijayakusuma Purwokerto, pelaporan kerusakan fasilitas masih dilakukan melalui komunikasi lisan, aplikasi pesan instan, maupun pencatatan manual. Akibatnya, informasi kerusakan tidak terdokumentasi dengan baik, penugasan teknisi kurang terkoordinasi, dan pimpinan mengalami kesulitan memonitor status penyelesaian pekerjaan maintenance. Hal ini juga memperlambat penyusunan laporan dan berpotensi menimbulkan kesalahan pencatatan. Temuan tersebut sejalan dengan kebutuhan akan sistem terintegrasi yang mendukung proses pelaporan, penugasan, monitoring, dan pelaporan maintenance.

Beberapa penelitian terdahulu menunjukkan bahwa penerapan sistem informasi helpdesk atau ticketing berbasis web mampu meningkatkan efektivitas pengelolaan pengaduan dan maintenance. Deffan (2022) mengungkapkan bahwa sistem helpdesk ticketing berbasis web dapat meminimalkan kesalahan komunikasi, mempercepat penanganan permasalahan, serta menyimpan seluruh riwayat pekerjaan secara terstruktur. Penelitian lain mengenai sistem pengaduan berbasis web di rumah sakit juga menunjukkan bahwa digitalisasi proses pengaduan mampu meningkatkan kualitas pelayanan karena setiap laporan dapat dipantau status penyelesaiannya secara daring (Wulansari et al., 2023).

Berdasarkan permasalahan tersebut, penelitian ini bertujuan merancang dan mengimplementasikan Sistem Informasi Ticketing Maintenance Berbasis Web di RST Wijayakusuma Purwokerto menggunakan metode Waterfall. Sistem yang dikembangkan memiliki tiga jenis pengguna, yaitu admin, teknisi, dan user. User dapat membuat laporan kerusakan secara daring, admin mengelola dan mendistribusikan tiket kepada teknisi, sedangkan teknisi memperbaiki status pekerjaan hingga perbaikan selesai. Dengan sistem ini, proses maintenance diharapkan menjadi lebih efektif, efisien, transparan, dan terdokumentasi dengan baik sehingga mendukung peningkatan kualitas pelayanan di RST Wijayakusuma Purwokerto.

B. METODE PENELITIAN

Penelitian ini menggunakan metode Research and Development (R&D) dengan model pengembangan perangkat lunak Waterfall. Model Waterfall dipilih karena memiliki tahapan terstruktur, meliputi analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan, sehingga sesuai untuk pengembangan sistem informasi berbasis web (Saputra et al., 2023). Pengumpulan data dilakukan melalui observasi, wawancara, dan studi pustaka. Observasi dan wawancara dilakukan di RST Wijayakusuma Purwokerto untuk mengidentifikasi kebutuhan pengguna dan permasalahan pada proses maintenance, sedangkan studi pustaka digunakan sebagai landasan teori dalam pengembangan sistem.

Perancangan sistem menggunakan Unified Modeling Language (UML) yang meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram. Implementasi sistem dilakukan menggunakan aplikasi berbasis web dengan basis data MySQL, kemudian diuji menggunakan metode Black Box Testing untuk memastikan seluruh fungsi sistem berjalan sesuai kebutuhan pengguna (Syabrullah et al., 2024).

2.1 Analisis Kebutuhan Sistem

Berdasarkan hasil observasi, sistem memiliki tiga aktor utama, yaitu Admin, Teknisi, dan User (Unit Pelapor).

a. Admin

Admin memiliki hak akses untuk:

- Login ke sistem.
- Mengelola data pengguna.
- Mengelola data divisi.
- Mengelola seluruh tiket maintenance.
- Menugaskan teknisi.
- Melakukan monitoring seluruh tiket.
- Menghasilkan laporan maintenance.

b. Teknisi

Teknisi memiliki hak akses untuk:

- Login ke sistem.
- Melihat daftar tiket.
- Mengambil tiket yang akan dikerjakan.
- Memberikan komentar perkembangan pekerjaan.
- Mengubah status pekerjaan hingga selesai.

c. User (Pelapor)

User memiliki hak akses untuk:

- Registrasi akun.
- Login ke sistem.
- Membuat tiket kerusakan.
- Mengunggah informasi kerusakan beserta lokasi.
- Melihat perkembangan status tiket secara real-time.
- Melihat riwayat pelaporan.

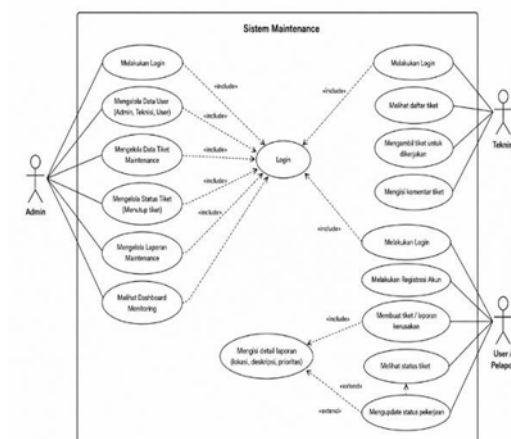
C. HASIL DAN PEMBAHASAN**Perancangan Sistem**

Perancangan sistem dilakukan menggunakan pendekatan Unified Modeling Language (UML) sebagai alat pemodelan. Diagram yang digunakan meliputi:

a. Use Case Diagram

Use Case Diagram menunjukkan interaksi antara Admin, Teknisi, dan User dalam sistem.

Masing-masing aktor memiliki hak akses sesuai tugas dan fungsinya.

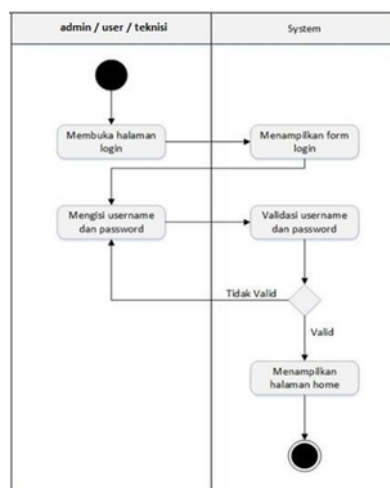


Sumber: Hasil Riset (2026)

Gambar 1. Rancangan Use Case Diagram

b. Activity Diagram

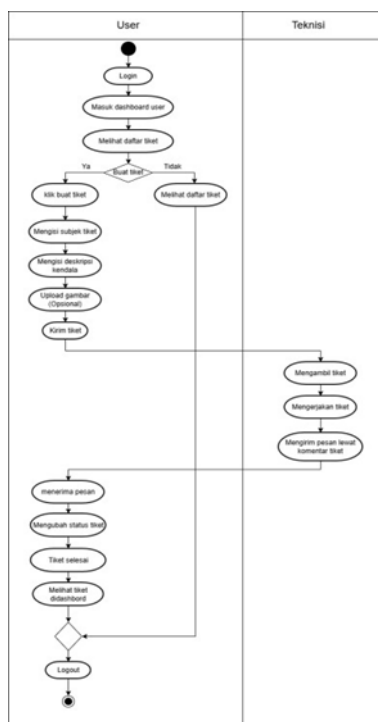
Diagram ini menggambarkan alur pengguna dalam membuat tiket kerusakan, mulai dari login hingga tiket berhasil dikirim ke sistem.



Sumber: Hasil Riset (2026)

Gambar 2. Activity Diagram Melakukan Login

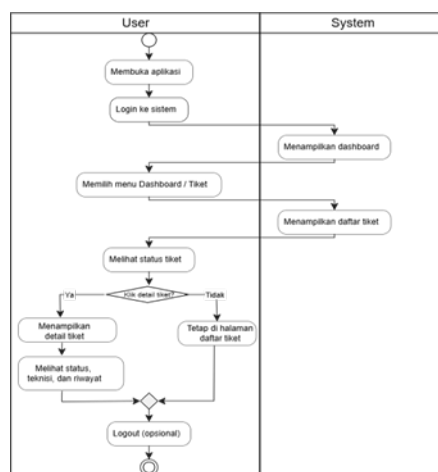
Admin, user, dan teknisi membuka halaman login untuk mengisi username dan password agar divalidasi oleh sistem. Jika data tidak valid, pengguna diminta mengulang pengisian; jika valid, sistem menampilkan halaman home atau dashboard.



Sumber: Hasil Riset (2026)

Gambar 3. Activity Diagram User Membuat Tiket

Setelah user berhasil login dan masuk ke dashboard, proses berlanjut dengan memilih menu melihat daftar tiket. Pada tahap ini, user dapat melihat seluruh tiket yang pernah dibuat beserta statusnya. Jika ingin membuat laporan baru, user memilih opsi buat tiket, lalu sistem menampilkan form pembuatan tiket yang harus diisi, yaitu subjek tiket dan deskripsi kendala. User juga dapat mengunggah gambar sebagai pendukung.

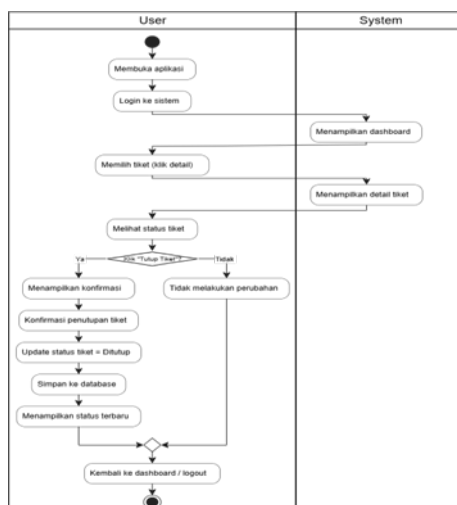


Sumber: Hasil Riset (2026)

Gambar 4. Activity Diagram User Melihat Status Tiket

User membuka aplikasi dan login ke sistem untuk menampilkan dashboard, kemudian memilih menu dashboard atau tiket agar sistem menampilkan daftar tiket, melihat status tiket,

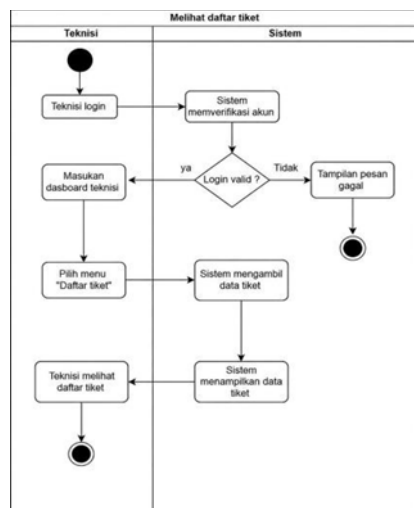
serta memilih untuk melihat detail tiket atau tetap di halaman daftar tiket sebelum akhirnya logout.



Sumber: Hasil Riset (2026)

Gambar 5. Activity Diagram User Mengupdate Status Tiket

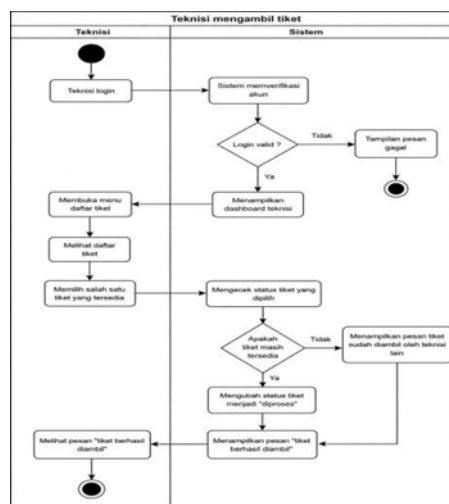
User membuka aplikasi dan login ke sistem untuk menampilkan dashboard, lalu memilih tiket dan melihat detail tiket, serta dapat mengonfirmasi penutupan tiket agar statusnya diperbarui di database dan ditampilkan sebagai status terbaru sebelum kembali ke dashboard atau logout.



Sumber: Hasil Riset (2026)

Gambar 6. Activity Diagram Melihat Daftar Tiket

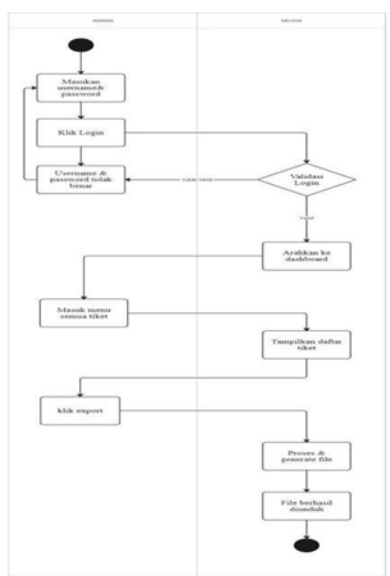
Teknisi melakukan login untuk diverifikasi oleh sistem. Jika login tidak valid, sistem menampilkan pesan gagal; jika valid, teknisi masuk ke dashboard, memilih menu daftar tiket agar sistem mengambil serta menampilkan data tiket untuk dilihat oleh teknisi.



Sumber: Hasil Riset (2026)

Gambar 7. Activity Diagram Teknisi Mengambil Tiket

Teknisi melakukan login untuk diverifikasi oleh sistem. Jika login tidak valid, sistem menampilkan pesan gagal; jika valid, sistem menampilkan dashboard teknisi sehingga teknisi dapat membuka menu daftar tiket untuk melihat seluruh tiket yang tersedia, kemudian memilih salah satu tiket yang akan dikerjakan. Sistem akan memeriksa ketersediaan tiket tersebut. Jika tiket masih tersedia dan belum diambil teknisi lain, sistem mengubah status tiket menjadi "diproses" serta menampilkan notifikasi "tiket berhasil diambil" sebagai konfirmasi bahwa tiket siap ditangani.

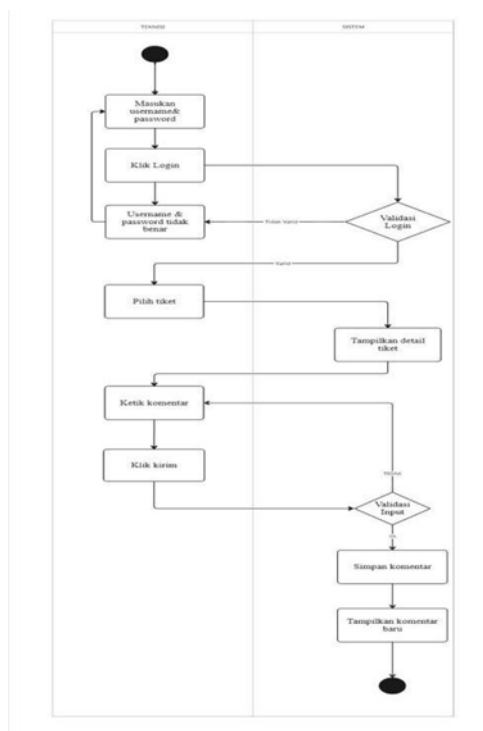


Sumber: Hasil Riset (2026)

Gambar 8. Activity Diagram Teknisi Mengisi Komentar Tiket

Teknisi memasukkan username dan password, kemudian mengklik tombol Login untuk diproses validasi oleh sistem. Apabila data login tidak valid, sistem menampilkan pesan

kesalahan sehingga teknisi harus mengulangi pengisian username dan password dengan benar. Sebaliknya, jika login berhasil, sistem menampilkan dashboard teknisi sehingga teknisi dapat memilih salah satu tiket untuk melihat informasi dan detail permasalahan yang dilaporkan. Setelah itu, teknisi mengetik komentar atau catatan terkait penanganan tiket, kemudian mengirimkannya ke sistem. Sistem akan memvalidasi isi komentar, dan apabila valid, komentar tersebut disimpan ke dalam basis data serta ditampilkan sebagai komentar baru pada halaman detail tiket sehingga dapat dilihat oleh pengguna yang berkaitan.



Sumber: Hasil Riset (2026)

Gambar 9. Activity Diagram Admin Mendownload Laporan

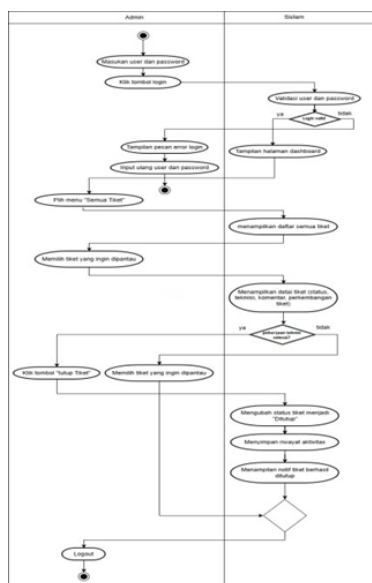
Admin memasukkan username dan password, lalu mengklik login untuk divalidasi oleh sistem. Jika login tidak valid, admin mengulang pengisian; jika valid, sistem mengarahkan ke dashboard sehingga admin dapat masuk ke menu semua tiket untuk melihat daftar tiket, lalu mengklik ekspor agar sistem memproses dan menghasilkan file yang berhasil diunduh.



Sumber: Hasil Riset (2026)

Gambar 10. Activity Diagram Monitoring Teknisi

Admin memasukkan username dan password, kemudian sistem melakukan validasi terhadap data login yang telah dimasukkan. Apabila data login valid, sistem akan menampilkan halaman dashboard sebagai halaman utama admin. Selanjutnya, admin memilih menu "Monitoring Teknisi", kemudian sistem memproses permintaan tersebut dengan mengambil data monitoring teknisi dari basis data dan menampilkannya dalam bentuk informasi yang berisi aktivitas, status penanganan tiket, serta progres pekerjaan masing-masing teknisi. Data tersebut dapat digunakan oleh admin untuk memantau kinerja dan perkembangan penanganan tiket secara real-time.



Sumber: Hasil Riset (2026)

Gambar 11. Activity Diagram Monitoring Semua Tiket

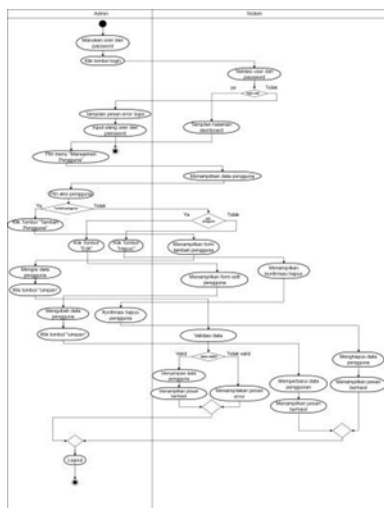
Admin login ke sistem, kemudian sistem memvalidasi dan menampilkan dashboard. Admin memilih menu "Semua Tiket" dan sistem menampilkan daftar tiket. Admin memilih tiket untuk dipantau, lalu sistem menampilkan detail dan status tiket. Jika pekerjaan teknisi sudah selesai, admin menutup tiket dan sistem mengubah status menjadi "Ditutup" serta menampilkan notifikasi. Terakhir, admin logout dan proses selesai.



Sumber: Hasil Riset (2026)

Gambar 12. Activity Diagram Manajemen Pengguna

Admin melakukan login ke dalam sistem dengan memasukkan username dan password, kemudian sistem melakukan proses validasi terhadap data login tersebut. Apabila login berhasil, sistem akan menampilkan halaman dashboard sebagai tampilan utama. Selanjutnya, admin memilih menu "Manajemen Pengguna", kemudian sistem mengambil dan menampilkan seluruh data pengguna yang tersimpan pada basis data. Pada menu ini, admin dapat melakukan berbagai pengelolaan data pengguna, seperti menambah, mengedit, maupun menghapus data pengguna sesuai kebutuhan. Setiap tindakan yang dilakukan admin akan diproses oleh sistem, kemudian sistem menyimpan perubahan ke dalam basis data dan menampilkan hasil pembaruan data sebagai konfirmasi bahwa proses telah berhasil dilakukan. Setelah seluruh aktivitas selesai, admin melakukan logout, dan sistem mengakhiri sesi pengguna serta mengembalikan tampilan ke halaman login.



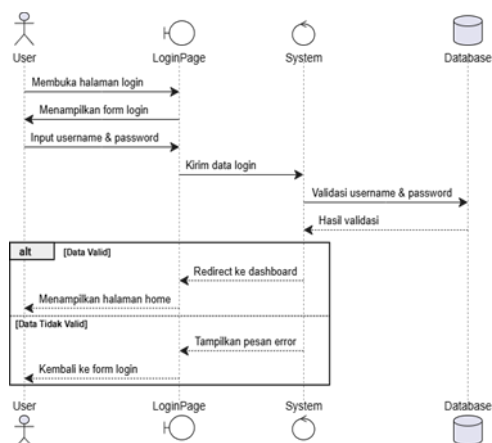
Sumber: Hasil Riset (2026)

Gambar 13. Activity Diagram Manajemen Divisi

Admin login ke sistem, lalu sistem memvalidasi dan menampilkan dashboard. Admin memilih menu "Manajemen Divisi" dan sistem menampilkan data divisi. Admin dapat menambah, mengedit, atau menghapus divisi, kemudian sistem memproses dan menampilkan hasilnya.

c. Sequence Diagram

Sequence Diagram memperlihatkan urutan interaksi antara aktor (pengguna), sistem, dan basis data selama suatu proses berlangsung. Diagram ini menggambarkan alur komunikasi dalam bentuk pertukaran pesan secara berurutan berdasarkan waktu, sehingga setiap langkah proses dapat dipahami dengan jelas.

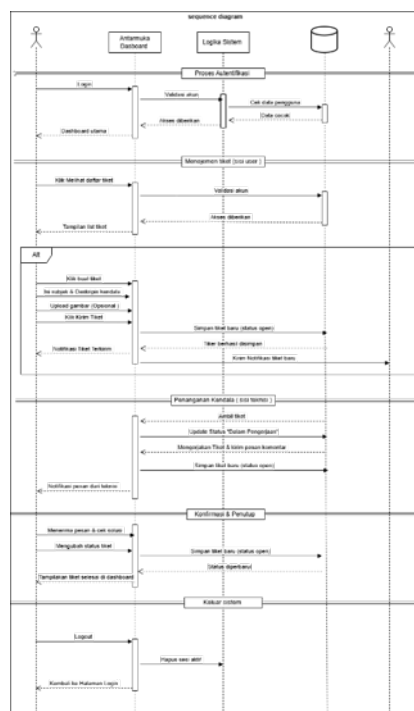


Sumber: Hasil Riset (2026)

Gambar 14. Sequence Diagram Melakukan Login

User membuka halaman login yang menampilkan form login untuk memasukkan username dan password, lalu LoginPage mengirimkan data login ke sistem untuk divalidasi ke database. Jika data valid, sistem me-redirect ke dashboard dan menampilkan halaman home;

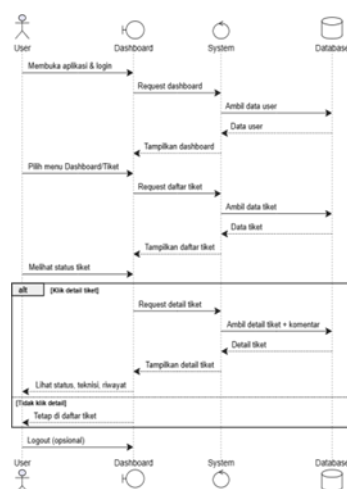
sedangkan jika tidak valid, sistem menampilkan pesan error dan kembali ke form login.



Sumber: Hasil Riset (2026)

Gambar 15. Sequence Diagram User Membuat Tiket

Diagram Sequence Teknisi menjelaskan alur interaksi antara teknisi, sistem, dan basis data dalam proses pengelolaan tiket hingga penyelesaian pekerjaan. Proses diawali dengan teknisi melakukan login, kemudian sistem melakukan autentikasi menggunakan data yang tersimpan pada basis data. Setelah login berhasil, sistem menampilkan dashboard utama sehingga teknisi dapat melihat daftar tiket yang tersedia. Selanjutnya, teknisi memilih salah satu tiket untuk melihat detail permasalahan, kemudian sistem mengambil dan menampilkan informasi tiket secara lengkap.



Sumber: Hasil Riset (2026)

Gambar 16. Sequence Diagram User Melihat Status Tiket

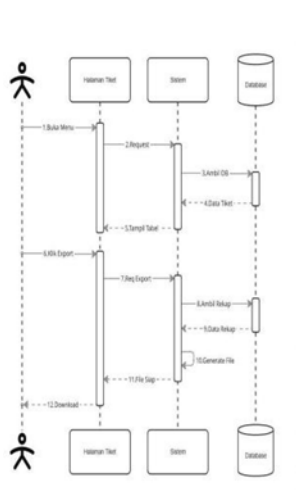
User membuka aplikasi kemudian melakukan login dengan memasukkan username dan password. Selanjutnya, sistem melakukan proses autentikasi dengan mengambil data pengguna dari database. Apabila data login valid, sistem akan menampilkan halaman dashboard sebagai halaman utama pengguna. Dari dashboard, user dapat memilih menu Dashboard atau Tiket, kemudian sistem mengambil data tiket dari basis data dan menampilkan daftar tiket yang dimiliki oleh pengguna. Selanjutnya, user dapat melihat status setiap tiket yang telah dibuat serta memilih salah satu tiket dengan mengklik menu Detail untuk melihat informasi yang lebih lengkap, seperti status penanganan tiket, teknisi yang menangani, dan riwayat aktivitas atau komentar pada tiket tersebut. Apabila tidak memilih menu detail, user tetap berada pada halaman daftar tiket untuk melihat informasi tiket lainnya. Setelah seluruh aktivitas selesai, user dapat melakukan logout (opsional) sehingga sistem mengakhiri sesi pengguna dan mengembalikan tampilan ke halaman login.



Sumber: Hasil Riset (2026)

Gambar 17. Sequence Diagram User Mengupdate Status Tiket

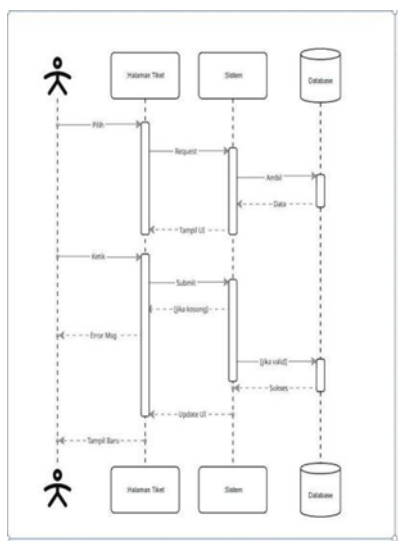
User membuka aplikasi dan login melalui DetailPage untuk meminta dashboard dan mengambil data user dari database, lalu memilih tiket untuk melihat detail tiket, serta melalui kondisi alternatif (alt) dapat mengklik "Tutup Tiket" untuk mengonfirmasi penutupan, memperbarui status di database, dan menampilkan status terbaru, atau memilih "Tidak klik" sehingga tidak ada perubahan, sebelum akhirnya kembali ke dashboard atau melakukan logout.



Sumber: Hasil Riset (2026)

Gambar 18. Sequence Diagram Teknisi Mengisi Komentar Tiket

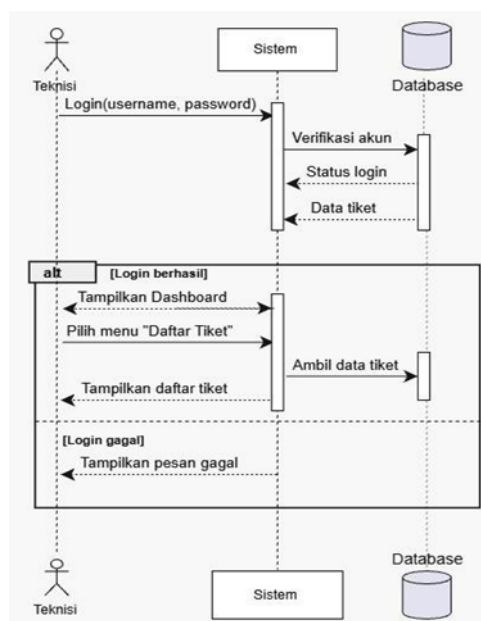
Teknisi memilih halaman tiket untuk mengirim request ke sistem yang mengambil data dari database dan menampilkannya pada UI, lalu teknisi mengetik komentar dan melakukan submit, di mana jika input kosong sistem memunculkan pesan error, dan jika valid sistem menyimpannya ke database dengan status sukses serta meng-update UI untuk menampilkan komentar baru kepada teknisi.



Sumber: Hasil Riset (2026)

Gambar 19. Sequence Diagram Admin Mendownload Laporan

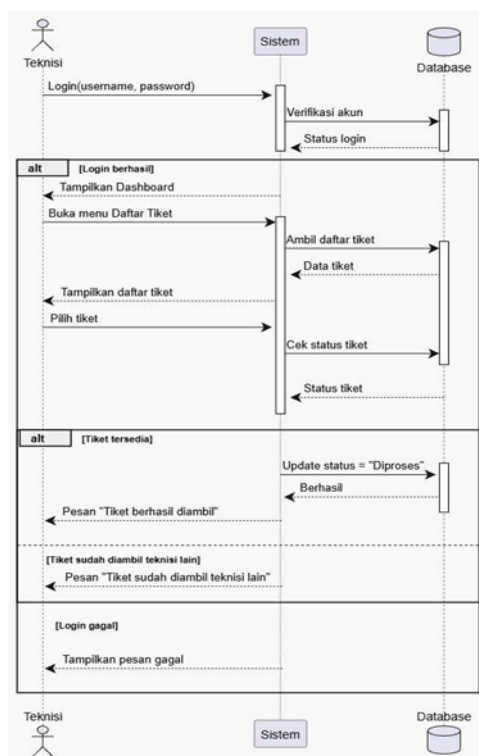
Admin membuka menu halaman tiket untuk melakukan request ke sistem, yang kemudian mengambil data dari database dan menampilkannya dalam bentuk tabel, lalu pengguna mengklik export agar sistem mengambil data rekap dari database, menghasilkan file, dan menyediakannya untuk di download oleh pengguna.



Sumber: Hasil Riset (2026)

Gambar 20. Sequence Diagram Melihat Daftar Tiket

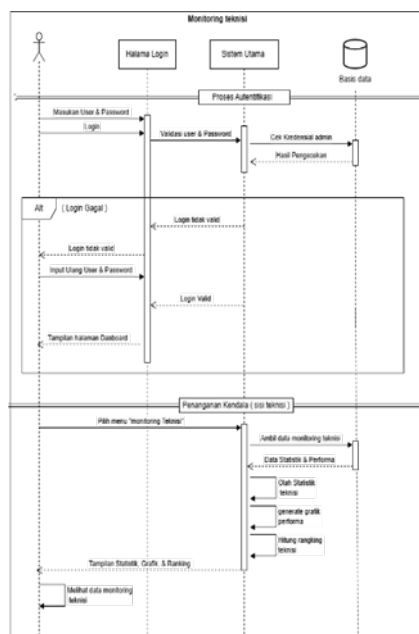
Teknisi melakukan login dengan memasukkan username dan password ke dalam sistem. Selanjutnya, sistem melakukan proses autentikasi dengan memverifikasi data login ke basis data (database). Apabila data yang dimasukkan valid, sistem akan menampilkan dashboard sebagai halaman utama teknisi. Dari dashboard, teknisi dapat memilih menu Daftar Tiket, kemudian sistem mengambil data tiket yang tersimpan pada basis data dan menampilkan seluruh daftar tiket yang tersedia untuk ditangani. Selanjutnya, teknisi dapat memilih salah satu tiket untuk melihat informasi dan detail permasalahan yang dilaporkan. Sebaliknya, apabila data login yang dimasukkan tidak valid, sistem akan menampilkan pesan login gagal, sehingga teknisi harus memasukkan kembali username dan password yang benar sebelum dapat mengakses sistem.



Sumber: Hasil Riset (2026)

Gambar 21. Sequence Diagram Teknisi Mengambil Tiket

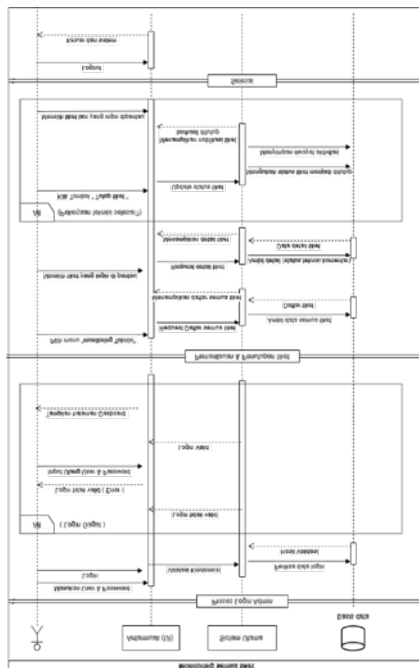
Teknisi melakukan login dengan mengirimkan username dan password ke sistem untuk verifikasi akun ke database. Jika login berhasil, sistem menampilkan dashboard, mengambil serta menampilkan daftar tiket. Ketika teknisi memilih tiket, sistem mengecek statusnya; jika tiket tersedia, sistem memperbarui status menjadi "Diproses" dan menampilkan pesan "Tiket berhasil diambil". Jika tiket sudah diambil teknisi lain, sistem menampilkan pesan tersebut, dan jika login gagal, sistem menampilkan pesan gagal.



Sumber: Hasil Riset (2026)

Gambar 22. Sequence Diagram Monitoring Teknisi

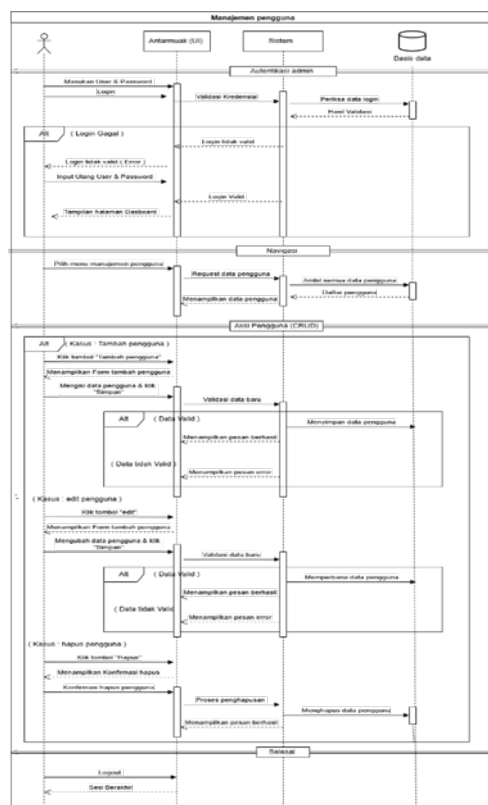
Diagram menunjukkan proses admin dalam memantau dan mengelola seluruh tiket maintenance.



Sumber: Hasil Riset (2026)

Gambar 23. Sequence Diagram Monitoring Semua Tiket

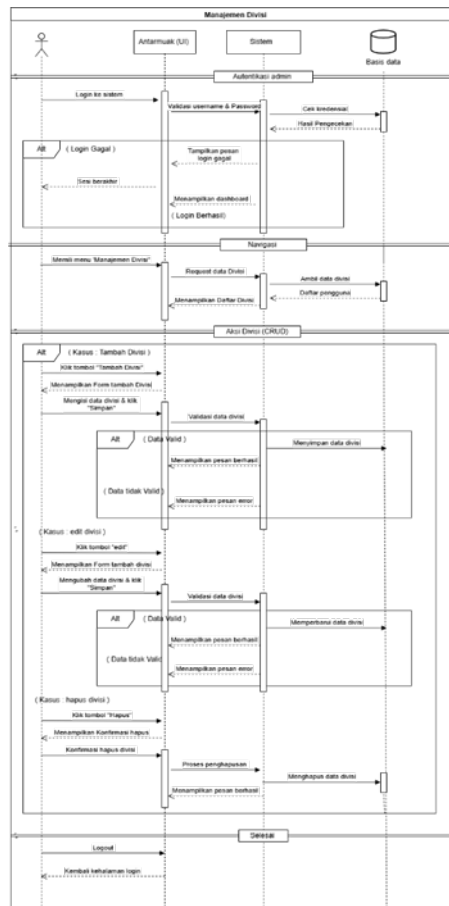
Diagram ini menggambarkan interaksi antara admin, sistem, dan basis data dalam proses memantau seluruh tiket yang ada pada sistem. Proses diawali dengan admin melakukan login, kemudian sistem memverifikasi data login ke basis data. Setelah autentikasi berhasil, sistem menampilkan dashboard admin. Selanjutnya, admin memilih menu Monitoring Semua Tiket, kemudian sistem mengambil seluruh data tiket dari basis data dan menampilkannya dalam bentuk daftar tiket.



Sumber: Hasil Riset (2026)

Gambar 24. Sequence Diagram Manajemen Pengguna

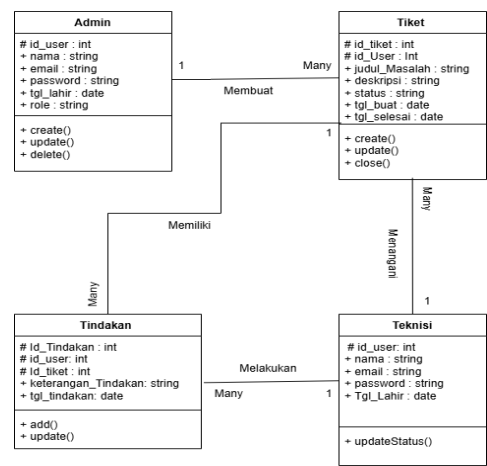
Diagram Sequence Manajemen Pengguna menunjukkan proses interaksi antara admin, sistem, dan basis data dalam mengelola data pengguna yang tersimpan pada sistem. Proses diawali dengan admin melakukan login menggunakan username dan password, kemudian sistem melakukan autentikasi dengan memverifikasi data login ke basis data. Setelah login berhasil, sistem menampilkan dashboard admin. Selanjutnya, admin memilih menu Manajemen Pengguna, kemudian sistem mengambil data pengguna dari basis data dan menampilkannya dalam bentuk daftar pengguna. Pada menu ini, admin dapat melakukan operasi Create, Read, Update, Delete (CRUD), yaitu menambahkan pengguna baru, melihat data pengguna, mengubah informasi pengguna, maupun menghapus data pengguna yang tidak diperlukan.



Gambar 25. Sequence Diagram Manajemen Divisi

c. Class Diagram

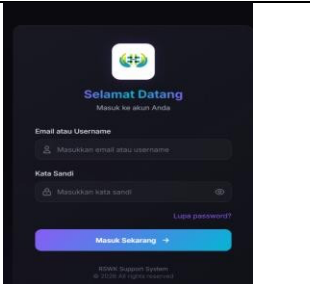
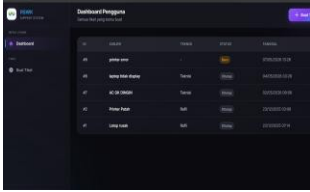
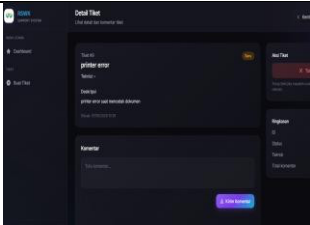
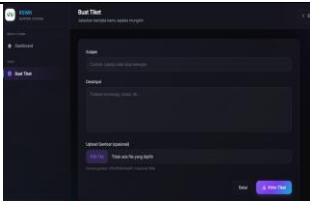
<https://journal.hasbaedukasi.co.id/index.php/jurmie>

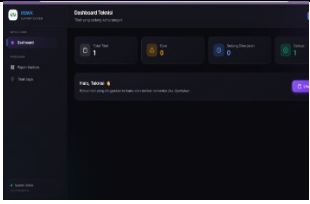
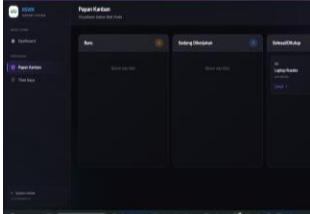
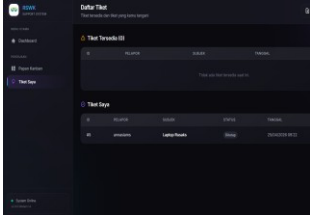
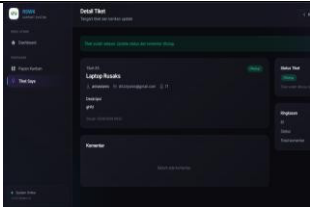
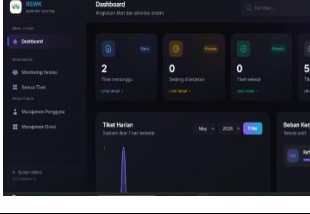
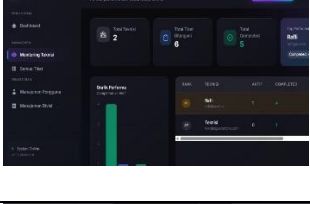
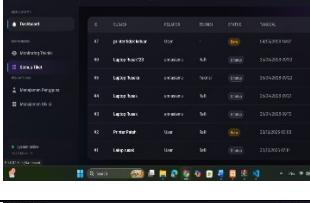
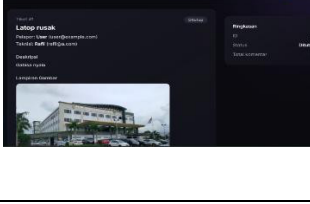


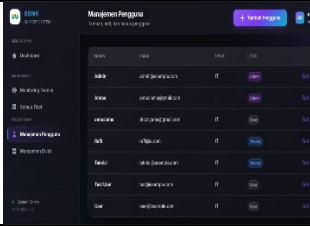
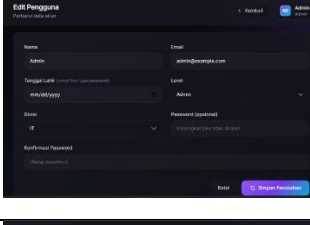
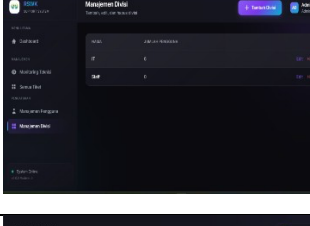
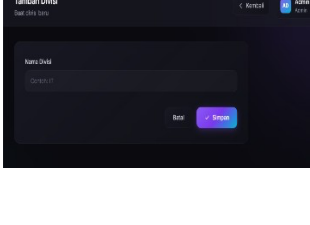
Sumber: Hasil Riset (2026)

Gambar 26. Class Diagram

Implementasi Sistem

Nama Tampilan	Keterangan
Tampilan login User	
Tampilan Dashboard user	
Tampilan detail tiket	
Tampilan Buat Tiket	

Tampilan halaman dashboard Teknisi	
Halaman papan kanban	
Halaman daftar tiket	
Halaman detail tiket	
Tampilan dashboard admin	
Tampilan monitoring teknisi	
Halaman Semua Tiket	
Halaman Detail Tiket	

Halaman Manajem en Pengguna	
Halaman Edit Pengguna	
Halaman ini mengelola data divisi	
Halaman ini menamba hkan divisi baru	

D. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan Sistem Informasi Ticketing Maintenance Berbasis Web di RST Wijayakusuma Purwokerto dengan menggunakan metode pengembangan perangkat lunak Waterfall. Sistem yang dikembangkan memiliki tiga jenis pengguna, yaitu admin, teknisi, dan user (pelapor), dengan fitur utama meliputi pelaporan kerusakan, pengelolaan tiket, penugasan teknisi, pembaruan status pekerjaan, monitoring maintenance, manajemen pengguna, manajemen divisi, serta pembuatan laporan maintenance.

Implementasi sistem ini mampu mengatasi berbagai kendala pada proses maintenance yang sebelumnya dilakukan secara manual, seperti kurangnya dokumentasi pelaporan, kesulitan dalam memonitor status pekerjaan, serta lambatnya penyusunan laporan. Dengan adanya sistem ini, proses pelaporan kerusakan menjadi lebih terstruktur, penugasan teknisi dapat dilakukan secara terkoordinasi, dan perkembangan penyelesaian pekerjaan dapat dipantau secara real-time. Selain itu, sistem menghasilkan dokumentasi riwayat maintenance

yang lebih baik sehingga mampu mendukung pengambilan keputusan serta meningkatkan efektivitas dan efisiensi pengelolaan maintenance di RST Wijayakusuma Purwokerto.

E. DAFTAR PUSTAKA

- Deffan, dkk. (2022). Rancang Bangun Sistem Informasi Ticketing Berbasis Website pada STF Muhammadiyah Cirebon. 6, 674–685.
- Jurnal, J., Informatika, R., Setiawan, F., Putra, F. K., Raafi, H. A., & Supriatna, S. (2026). Sistem Informasi Maintenance Alat Medis Berbasis Web Menggunakan Metode Waterfall pada PT Abadi Perkasa Bersama Alkesindo. 3(9), 2411–2419.
- Wulansari, N. D., Marthasari, G. I., & Wiyono, B. S. (2023). Perancangan Sistem Informasi Pengaduan RSUD Dr . Soedarsono Pasuruan Berbasis Website Dengan Menggunakan Metode Waterfall. 5(3), 747–758.