

IMPLEMENTASI METODE *FIRST COME FIRST SERVED* PADA SISTEM *HELPDESK TICKETING* BERBASIS *WEB* GUNA MENGOPTIMALKAN *RESPONSE TIME* LAYANAN *IT SUPPORT*

Evan Eka Laksana¹, Aang Alim Murtopo², Syefudin³

¹⁻³ Teknik Informatika, Sains dan Teknologi, Universitas Harkat Negeri, Jl. Mataram No.9, Pesurungan Lor, Kec. Margadana, Kota Tegal, Jawa Tengah, Indonesia 52147.

*Penulis Korespondensi: evanlaksana5@gmail.com¹, aang.alim@gmail.com², syefudin5@gmail.com³

Abstract. *Company operational activities heavily depend on the quality of information technology services. However, the use of multiple communication media for reporting technical issues has resulted in unstandardized documentation, difficulties in monitoring ticket handling status, and an increased risk of overlooked reports. This study aims to design and develop a web-based Helpdesk Ticketing System to improve the effectiveness of issue reporting and handling by implementing the First Come First Served (FCFS) method. The research employed the Waterfall software development model, while data were collected through literature studies, interviews, and documentation. The system was designed using Unified Modeling Language (UML), implemented with the Laravel framework and MySQL database, and evaluated using Black Box Testing. The results demonstrate that the proposed system was successfully implemented with features including ticket creation, queue management based on the FCFS method, email notifications, Frequently Asked Questions (FAQ), and real-time communication between administrators and users through a chat feature. The implementation of the FCFS method effectively organized ticket handling in a systematic order, making it easier to monitor ticket status and improving the efficiency of issue resolution. Therefore, the developed system can enhance the effectiveness of IT Support services in managing technical issue reports, although further development is still required to accommodate more complex organizational needs.*

Keywords: *Helpdesk Ticketing System, First Come First Served (FCFS), Waterfall Model, Laravel Framework, Black Box Testing*

Abstrak. Aktivitas operasional perusahaan sangat bergantung pada kualitas layanan teknologi informasi. Namun, proses pelaporan gangguan yang masih menggunakan berbagai media menyebabkan dokumentasi tidak terstandar, sulitnya pemantauan status penanganan, serta meningkatnya risiko laporan terlewat. Penelitian ini bertujuan merancang dan membangun aplikasi *Helpdesk Ticketing System* berbasis web untuk meningkatkan efektivitas pelaporan dan penanganan gangguan melalui penerapan metode First Come First Served (FCFS). Metode penelitian yang digunakan adalah rekayasa perangkat lunak dengan model Waterfall, sedangkan pengumpulan data dilakukan melalui studi literatur, wawancara, dan dokumentasi. Sistem dirancang menggunakan Unified Modeling Language (UML), diimplementasikan dengan framework Laravel dan *Database MySQL*, serta diuji menggunakan metode Black Box Testing. Hasil penelitian menunjukkan bahwa sistem berhasil diimplementasikan dengan fitur pembuatan tiket, manajemen antrian berdasarkan metode FCFS, notifikasi email, FAQ, serta komunikasi antara *Admin* dan *User* melalui fitur chat. Penerapan metode FCFS mampu mengurutkan proses penanganan tiket secara sistematis sehingga memudahkan pemantauan status dan meningkatkan efisiensi penanganan laporan. Dengan demikian, sistem yang dikembangkan dapat membantu meningkatkan efektivitas layanan *IT Support* dalam mengelola laporan gangguan, meskipun masih diperlukan pengembangan lebih lanjut untuk mendukung kebutuhan yang lebih kompleks..

Kata kunci: *Helpdesk Ticketing System, First Come First Served (FCFS), Model Waterfall, Laravel, Black Box Testing*

1. LATAR BELAKANG

Aktivitas operasional di lingkungan organisasi modern sangat bergantung pada stabilitas layanan teknologi informasi. Gangguan pada perangkat keras, perangkat lunak, maupun jaringan sering kali berdampak langsung pada terhambatnya pekerjaan pengguna dan menurunnya produktivitas. Dalam praktiknya, banyak tim *IT Support* masih

menghadapi kesulitan dalam mengelola laporan gangguan karena proses pelaporan dilakukan melalui berbagai media yang terpisah, tidak terstandar, dan minim dokumentasi. Kondisi tersebut menyebabkan laporan keluhan mudah terlewat, serta menyulitkan proses pemantauan status penanganan. Selain itu, kurangnya riwayat penanganan yang terdokumentasi dengan baik juga menyulitkan tim *IT Support* dalam melakukan evaluasi kinerja dan perbaikan layanan secara berkelanjutan.

Berdasarkan permasalahan tersebut, diperlukan sebuah mekanisme yang mampu mengelola proses pelaporan dan penanganan gangguan layanan TI secara terintegrasi dan terdokumentasi. Mekanisme ini diharapkan dapat memfasilitasi pencatatan keluhan, pemantauan progres penanganan, serta komunikasi antara pengguna dan tim *IT Support* secara lebih terstruktur. Dengan penerapan mekanisme yang tepat, kualitas dan efisiensi layanan *IT Support* dapat ditingkatkan secara signifikan. Salah satu bentuk mekanisme tersebut dapat dilakukan melalui perancangan sebuah system tiket layanan bantuan berbasis *web* yang dikenal sebagai *Helpdesk Ticketing System*. Sistem ini memungkinkan setiap gangguan yang dilaporkan dapat dicatat dalam bentuk tiket, kemudian dikelola berdasarkan status dan tingkat prioritas, serta dipantau secara *real-time* oleh pihak terkait. Selain itu, system ini juga menyediakan media komunikasi terpusat antara pengguna dan tim *IT Support* sehingga seluruh proses penanganan terdokumentasi dengan baik dan dapat dijadikan bahan evaluasi layanan di masa mendatang.

2. KAJIAN TEORITIS

Rekayasa perangkat lunak merupakan disiplin ilmu yang mempelajari metode, teknik, dan pendekatan sistematis dalam pengembangan perangkat lunak. Ruang lingkup rekayasa perangkat lunak mencakup analisis kebutuhan, perancangan sistem, implementasi, pengujian, serta pemeliharaan perangkat lunak. Tujuan utama rekayasa perangkat lunak adalah menghasilkan perangkat lunak yang berkualitas, dapat memenuhi kebutuhan pengguna, serta mudah dikembangkan dan dipelihara di masa mendatang (Retnoningsih et al., n.d.).

Rekayasa perangkat lunak merupakan disiplin ilmu yang mempelajari metode, teknik, dan pendekatan sistematis dalam pengembangan perangkat lunak. Ruang lingkup rekayasa perangkat lunak mencakup analisis kebutuhan, perancangan sistem, implementasi, pengujian, serta pemeliharaan perangkat lunak. Tujuan utama rekayasa perangkat lunak adalah menghasilkan perangkat lunak yang berkualitas, dapat memenuhi kebutuhan pengguna, serta mudah dikembangkan dan dipelihara di masa mendatang (Nugroho & Ali, 2022).

Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk memvisualisasikan dan mendokumentasikan sistem perangkat lunak. UML menyediakan berbagai jenis diagram, seperti use case diagram, class diagram, sequence diagram, dan activity diagram, yang digunakan untuk menggambarkan struktur sistem serta interaksi antar komponen. Penggunaan UML membantu pengembang dalam memahami kebutuhan sistem secara menyeluruh sebelum proses implementasi dilakukan (Huwaiddy et al., n.d.).

Basis data merupakan kumpulan data yang disimpan secara terorganisasi dan saling berelasi sehingga dapat diakses, dikelola, dan diperbarui dengan mudah. Basis data digunakan dalam sistem informasi untuk menjaga konsistensi, integritas, dan keamanan data. Dengan basis data, sistem mampu menyimpan informasi dalam jumlah besar dan mendukung proses pencarian serta pengolahan data secara efisien (Maulidiah et al., 2025).

Helpdesk merupakan layanan yang berfungsi sebagai pusat kontak antara pengguna dan penyedia layanan teknologi informasi. *Helpdesk* bertugas menerima laporan gangguan, mencatat permasalahan, serta mengoordinasikan proses penanganan masalah oleh tim teknis. Keberadaan *Helpdesk* membantu organisasi dalam menjaga kualitas layanan TI melalui pengelolaan keluhan yang terstruktur dan terdokumentasi (Ramadhani et al., 2024).

Ticketing system merupakan mekanisme pengelolaan layanan yang digunakan untuk mencatat dan melacak permintaan layanan atau gangguan dalam bentuk tiket. Setiap tiket memuat informasi mengenai permasalahan, status penanganan, prioritas, serta riwayat penyelesaian. *Ticketing system* membantu tim teknis dalam mengatur alur kerja, memantau progres penyelesaian masalah, serta menyediakan dokumentasi layanan sebagai bahan evaluasi (Maliki & Sulistiyowati, 2022).

Metode First Come First Served (FCFS) merupakan teknik penjadwalan dalam sistem antrian yang mengutamakan proses berdasarkan urutan kedatangan, di mana proses pertama kali masuk akan dilayani terlebih dahulu hingga selesai. Algoritma ini memproses keadilan dan transparansi dalam pengelolaan antrian pengaduan atau job, seperti pada sistem *Helpdesk* di mana keluhan pertama diprioritaskan (Prayitno & Nurmalisa, 2024).

Metode Waterfall merupakan kerangka pengembangan sistem yang bersifat linier dan sekuensial, dimulai dari pengumpulan kebutuhan, desain implementasi, hingga pengujian, tanpa kembali ke tahap sebelumnya. Pendekatan ini digunakan untuk memandu proses pengembangan sistem informasi layanan pengaduan server secara terstruktur, memastikan setiap tahap selesai sebelum lanjut ke tahap berikutnya. Metode ini efektif untuk proyek dengan kebutuhan yang jelas dan stabil, seperti dalam pengembangan sistem monitoring siswa berbasis *web* (Azis & Voutama, 2025).

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman *scripting* sisi server yang banyak digunakan untuk pengembangan aplikasi *web* dinamis karena bersifat *open source*, fleksibel, dan memiliki dukungan komunitas yang luas (Endra et al., 2022).

Laravel merupakan *framework* PHP modern yang dirancang untuk mempermudah dan mempercepat pengembangan aplikasi *web* melalui struktur arsitektur yang teratur, seperti pola MVC, *routing*, *middleware*, dan ORM Eloquent yang memudahkan interaksi dengan basis data tanpa menulis perintah SQL secara manual (Fauzan Prasetyo Eka Putra, Reynal Widya Efendi, Alief Badrit Tamam, 2025). Penggunaan Laravel dalam pengembangan aplikasi PHP telah terbukti meningkatkan efisiensi penulisan kode, mempermudah pemeliharaan sistem, serta mendukung pengembangan aplikasi yang lebih terstruktur dan dapat diskalakan (Sinlae et al., 2024).

MySQL merupakan sistem manajemen basis data relasional yang bersifat *open-source* dan banyak digunakan dalam pengembangan aplikasi berbasis web karena mampu mengelola data secara terstruktur, efisien, aman, dan konsisten. Penerapan MySQL dalam sistem informasi mendukung proses penyimpanan, pengelolaan, dan akses data secara terintegrasi, sehingga data seperti keluhan, riwayat penanganan, maupun informasi transaksi dapat tersusun dengan baik dan mudah diolah sesuai dengan kebutuhan sistem yang ada (Dachi et al., 2025).

User interface (UI) dan *User experience (Ux)* merupakan dua aspek yang sangat penting dalam proses pengembangan suatu aplikasi karena berperan dalam membentuk kesan awal serta mempengaruhi kualitas interaksi pengguna dengan sistem. Oleh karena itu, perancangan UI dan UX perlu disesuaikan dengan kebutuhan pengguna agar aplikasi yang dikembangkan dapat digunakan secara efektif dan memberikan pengalaman yang

baik. Salah satu komponen utama dalam perancangan tersebut adalah *User interface (UI)*, yaitu seluruh komponen antarmuka yang ditampilkan pada layer sebagai media interaksi antara pengguna dengan sistem. Melalui antarmuka tersebut, sistem dapat menyampaikan informasi kepada pengguna sekaligus menerima masukan dari pengguna sehingga setiap fungsi pada aplikasi dapat dijalankan dengan baik (Tech, 2022).

Selain *User interface (UI)*, *User experience (Ux)* juga menjadi aspek yang tidak kalah penting dalam pengembangan suatu aplikasi. UX berkaitan dengan bagaimana pengguna dapat berinteraksi secara efektif dan nyaman Ketika menggunakan antarmuka yang telah dirancang. Penerapan UX bertujuan untuk meningkatkan kepuasan pengguna selama menggunakan aplikasi, baik pada platform website, perangkat bergerak (*mobile*), maupun desktop. Oleh karena itu, perancangan UX perlu memperhatikan kemudahan dan kenyamanan pengguna agar interaksi dengan sistem dapat berjalan secara lebih *optimal* (Putra et al., 2024).

Black-Box Testing merupakan metode pengujian perangkat lunak yang dilakukan dengan menguji fungsi-fungsi sistem berdasarkan dengan spesifikasi kebutuhan tanpa melihat struktur maupun kode program yang ada di dalamnya. Pengujian difokuskan pada kesesuaian antara data masukan dan data keluaran yang dihasilkan sistem, sehingga setiap fitur dapat dipastikan berfungsi sesuai dengan yang diharapkan. Metode ini bertujuan untuk mendeteksi kesalahan pada semua aspek mulai dari aspek fungsional, antarmuka, struktur data, performa, serta proses inialisasi dan terminasi sistem. Dengan demikian, Black-Box Testing menjadi salah satu metode yang efektif untuk mengevaluasi kualitas perangkat lunak dari sudut pandang pengguna sebelum sistem diimplementasikan (Albaqir et al., 2023).

3. METODE PENELITIAN

Penelitian ini menggunakan metode pengembangan perangkat lunak *Waterfall* dalam perancangan dan pengembangan *Helpdesk Ticketing System* berbasis web untuk mendukung layanan tim *IT Support*. Metode ini dipilih karena memiliki tahapan yang sistematis dan berurutan, meliputi perencanaan, analisis, perancangan, implementasi, pengujian, dan evaluasi, sehingga pengembangan sistem dapat dilakukan sesuai dengan kebutuhan pengguna.

Pengumpulan data dilakukan melalui metode wawancara, dokumentasi, dan studi literatur. Wawancara dilakukan dengan Kepala *IT Support* pada saat kegiatan magang untuk memperoleh informasi mengenai proses bisnis, kebutuhan sistem, serta permasalahan yang dihadapi. Dokumentasi digunakan untuk mengumpulkan referensi terkait alur kerja layanan *IT Support*, sistem sejenis, dan penelitian terdahulu, sedangkan studi literatur dilakukan melalui kajian buku, jurnal, dan sumber ilmiah yang berkaitan dengan *Helpdesk Ticketing System*, metode *First Come First Served (FCFS)*, *Unified Modeling Language (UML)*, serta teknologi pengembangan sistem berbasis web.

Setelah proses pengembangan selesai, sistem diuji menggunakan metode *Black Box Testing* untuk memverifikasi bahwa seluruh fungsionalitas sistem telah berjalan sesuai dengan spesifikasi fungsional yang telah ditetapkan.

4. HASIL DAN PEMBAHASAN

A. IMPLEMENTASI SISTEM

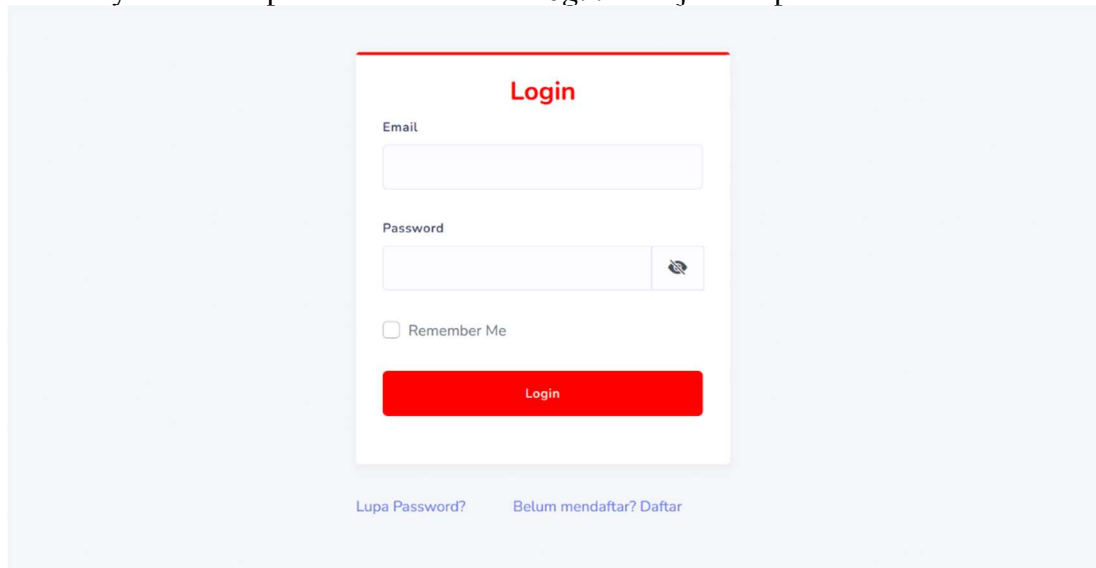
Pada tahap ini, hasil perancangan sistem diimplementasikan ke dalam bentuk aplikasi *Helpdesk Ticketing System* berbasis web untuk mendukung proses pelaporan dan penanganan gangguan layanan teknologi informasi secara lebih terstruktur. Sistem dikembangkan menggunakan bahasa pemrograman PHP dengan *framework* Laravel serta MySQL sebagai sistem manajemen basis data. Fitur utama yang diimplementasikan

meliputi *Login*, *Dashboard*, *Helpdesk*, pembuatan dan penanganan laporan tiket, fitur *Chat* antara *User* dan *IT Support*, serta fitur *FAQ troubleshooting* yang membantu pengguna menyelesaikan permasalahan teknis ringan secara mandiri.

Implementasi sistem dilakukan berdasarkan hasil analisis kebutuhan, perancangan *Unified Modeling Language (UML)*, serta desain antarmuka yang telah disusun pada tahap sebelumnya. Melalui implementasi tersebut, sistem mampu mendukung proses pelaporan, penanganan, pemantauan status tiket, dan komunikasi antara *User*, *IT Support*, dan *Admin* secara lebih efektif serta terdokumentasi dengan baik.

1. Implementasi Login

Pada tahap implementasi, rancangan *wireframe* halaman *Login* direalisasikan menjadi antarmuka fungsional sesuai dengan hasil perancangan yang telah disusun sebelumnya. Hasil implementasi halaman *Login* ditunjukkan pada Gambar berikut.

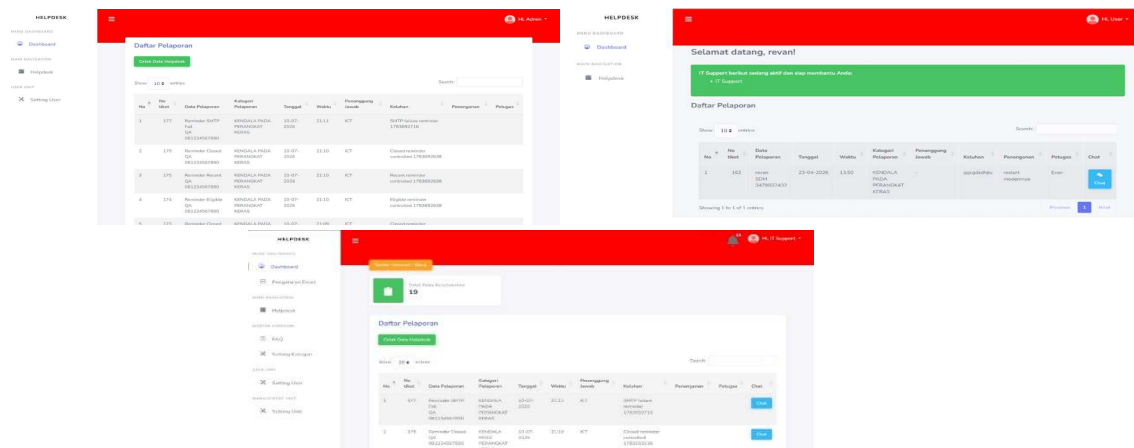


Gambar di atas menunjukkan implementasi halaman *Login* pada *Helpdesk Ticketing System*. Halaman ini berfungsi sebagai media autentikasi bagi *User*, *IT Support*, dan *Admin* sebelum mengakses sistem melalui kolom *Email* dan *Password* yang divalidasi dengan data pada basis data. Selain itu, tersedia fitur *Lupa Password* untuk membantu *User* melakukan pemulihan kata sandi serta tautan *Daftar* yang digunakan untuk registrasi akun baru.

2. Implementasi Tampilan Dashboard

Pada tahap implementasi, hasil perancangan antarmuka *Dashboard* telah direalisasikan ke dalam aplikasi *Helpdesk Ticketing System* berbasis web. Halaman *Dashboard* merupakan halaman utama yang ditampilkan setelah pengguna berhasil melakukan *Login*. Tampilan *Dashboard* disesuaikan dengan hak akses masing-masing aktor, yaitu *User*, *IT Support*, dan *Admin*, sehingga informasi yang ditampilkan sesuai dengan kebutuhan setiap pengguna. Hasil implementasi tampilan *Dashboard* dapat dilihat pada Gambar berikut.

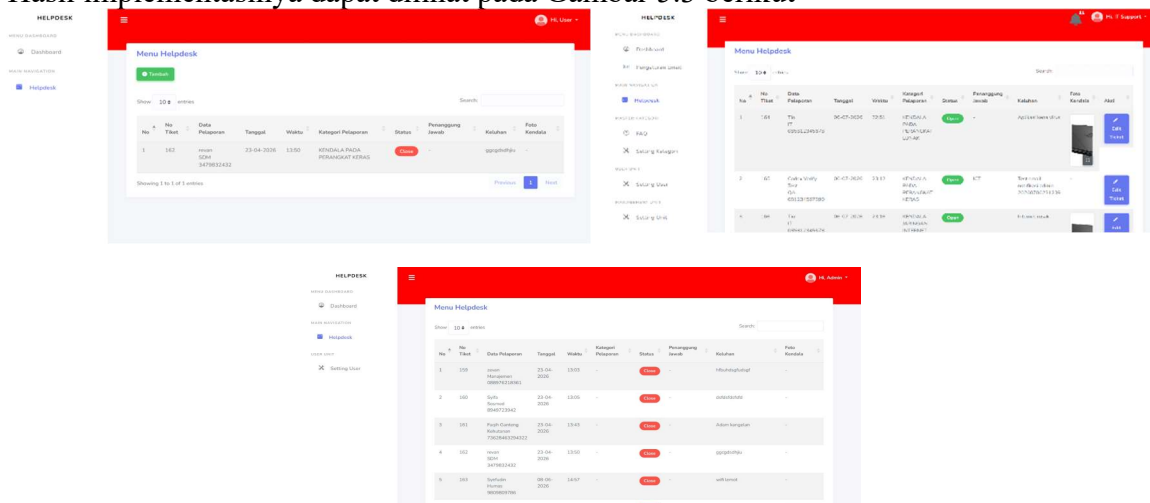
IMPLEMENTASI METODE FIRST COME FIRST SERVED PADA SISTEM HELPDESK TICKETING BERBASIS WEB GUNA MENGOPTIMALKAN RESPONSE TIME LAYANAN IT SUPPORT



Gambar di atas menunjukkan implementasi tampilan *Dashboard* berdasarkan hak akses *User*, *IT Support*, dan *Admin*. Halaman *Dashboard* berfungsi menampilkan daftar laporan tiket sesuai dengan peran masing-masing pengguna. *Dashboard User* menampilkan laporan tiket yang dibuat oleh pengguna beserta status penanganannya, sedangkan *Dashboard IT Support* dan *Dashboard Admin* menampilkan seluruh laporan tiket yang telah dikirim oleh *User*.

3. Implementasi Tampilan Helpdesk

Pada bagian implementasi ini, disajikan tampilan antarmuka formulir peminjaman aset setelah melewati tahap pengembangan kode program. Halaman ini berfungsi bagi pengguna untuk mengajukan permohonan penggunaan barang inventaris secara daring. Hasil implementasinya dapat dilihat pada Gambar 5.3 berikut



Gambar di atas menunjukkan implementasi menu *Helpdesk* berdasarkan hak akses *User*, *IT Support*, dan *Admin*. Pada sisi *User*, menu ini menampilkan daftar laporan tiket serta menyediakan tombol “Tambah” untuk membuat laporan baru. Pada sisi *IT Support*, menu *Helpdesk* menampilkan seluruh laporan tiket dan menyediakan tombol “Edit Tiket” untuk memberikan penanganan terhadap setiap laporan berdasarkan metode *First Come First Served (FCFS)*. Sementara itu, pada sisi *Admin*, menu *Helpdesk* digunakan untuk menampilkan seluruh laporan tiket beserta status penanganannya sebagai media pemantauan.

4. Implementasi Tampilan Form Pembuatan laporan dan Penanganan laporan

Pada tahap implementasi ini, hasil perancangan formulir pembuatan dan penanganan laporan telah direalisasikan ke dalam antarmuka aplikasi *Helpdesk Ticketing*

IMPLEMENTASI METODE *FIRST COME FIRST SERVED* PADA SISTEM *HELPPDESK TICKETING* BERBASIS WEB GUNA MENGOPTIMALKAN *RESPONSE TIME* LAYANAN *IT SUPPORT*

System berbasis web. Formulir pembuatan laporan dirancang untuk memudahkan *User* dalam menyampaikan keluhan secara terstruktur, sedangkan formulir penanganan laporan digunakan oleh *IT Support* untuk memberikan solusi serta memperbarui status penyelesaian tiket. Hasil implementasi kedua formulir tersebut dapat dilihat pada Gambar berikut:

The image displays two screenshots of a web-based helpdesk system. The left screenshot shows the 'Tambah Helpdesk' (Add Helpdesk) form, which includes a sidebar with navigation links like 'Dashboard', 'Helpdesk', 'FAQ', 'Setting Kategori', 'Setting User', and 'Setting Unit'. The main content area contains a form with fields for 'Status' (a dropdown menu), 'Penanganan' (a text area), 'Petugas' (a text area), and 'Close Date' (a date picker). The right screenshot shows the 'Edit Tiket' (Edit Ticket) form, which is similar to the 'Tambah Helpdesk' form but includes a 'Status' dropdown menu and a 'Close Date' field.

Gambar di atas menunjukkan implementasi formulir pembuatan laporan oleh *User* dan formulir penanganan laporan oleh *IT Support*. Pada sisi *User*, sistem menampilkan panel *Frequently Asked Questions (FAQ)* sebagai panduan sebelum pengguna membuat laporan. Apabila kendala belum dapat diselesaikan, *User* dapat mengisi formulir laporan yang terdiri atas kategori keluhan, tujuan unit penanganan, deskripsi permasalahan, serta lampiran foto sebagai bukti pendukung. Sementara itu, formulir penanganan digunakan oleh *IT Support* untuk memperbarui status tiket, memberikan solusi, mencatat petugas yang menangani, dan tanggal penyelesaian (*close date*). Proses penanganan menerapkan metode *First Come First Served (FCFS)* sehingga laporan yang pertama kali masuk akan diprioritaskan untuk ditangani.

5. Implementasi Tampilan Fitur *FAQ User* dan *IT Support*

Pada tahap implementasi, hasil perancangan fitur *Frequently Asked Questions (FAQ)* telah direalisasikan ke dalam aplikasi *Helpdesk Ticketing System* berbasis web. Fitur ini dikembangkan sebagai sarana penyedia informasi dan solusi awal terhadap permasalahan teknis yang sering dialami oleh pengguna. Selain dimanfaatkan oleh *User* sebagai media *self-service troubleshooting*, fitur *FAQ* juga menyediakan fasilitas bagi *IT Support* untuk mengelola daftar pertanyaan dan jawaban sehingga informasi yang disampaikan kepada pengguna dapat diperbarui sesuai kebutuhan. Hasil implementasi fitur *FAQ* dapat dilihat pada Gambar berikut:

The image displays a screenshot of the 'Daftar FAQ' (FAQ List) table. The table has three columns: 'Pertanyaan' (Question), 'Jawaban' (Answer), and 'Aksi' (Action). The 'Aksi' column contains buttons for 'Tambah FAQ' (Add FAQ), 'Edit', 'Hapus' (Delete), and 'Detail'. The table lists several common technical issues and their solutions, such as 'Apa penyebab umum mouse atau keyboard tidak berfungsi?' (What are the common causes of mouse or keyboard not working?) and 'Mengapa aplikasi tiba-tiba tertutup sendiri?' (Why does the application suddenly close itself?).

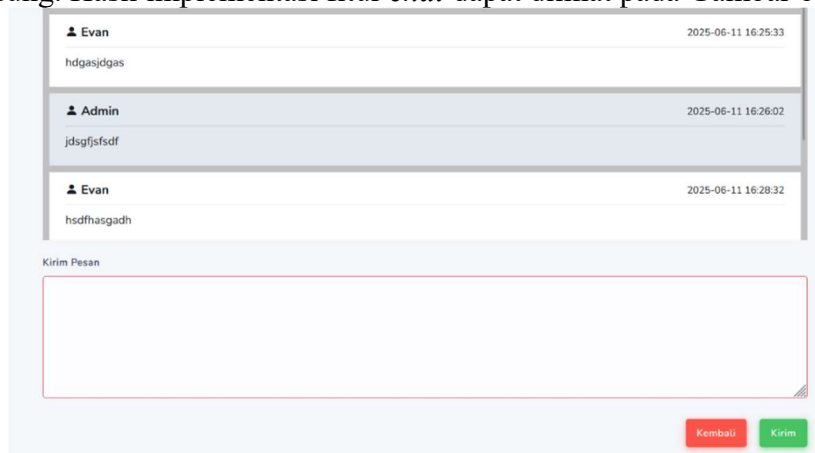
Pertanyaan	Jawaban	Aksi
Apa penyebab umum mouse atau keyboard tidak berfungsi?	Baterei habis (wireless) atau kabel tergangguk.	Tambah FAQ, Edit, Hapus, Detail
Mengapa aplikasi tiba-tiba tertutup sendiri?	Bug aplikasi atau kurangnya sumber daya sistem.	Tambah FAQ, Edit, Hapus, Detail
Apa yang harus dilakukan jika komputer tidak mau menyala sama sekali?	Periksa kabel power, pastikan terhubung dengan benar ke stopkontak dan unit komputer.	Tambah FAQ, Edit, Hapus, Detail
Apa yang menyebabkan performa jaringan Wi-Fi menjadi lambat?	Terdapat banyak perangkat terhubung, sinyal lemah, atau masalah pada router.	Tambah FAQ, Edit, Hapus, Detail

Gambar di atas menunjukkan implementasi fitur *Frequently Asked Questions (FAQ)* berdasarkan hak akses *User* dan *IT Support*. Pada sisi *User*, halaman *FAQ* menampilkan daftar pertanyaan beserta solusi terhadap permasalahan teknis ringan yang umum terjadi sebagai media *self-service troubleshooting*. Melalui fitur ini, *User* dapat

mencoba menyelesaikan kendala secara mandiri sebelum membuat laporan tiket, sehingga permasalahan yang masih dapat ditangani sendiri tidak perlu dilaporkan kepada *IT Support*. Sementara itu, pada sisi *IT Support*, halaman *FAQ* digunakan untuk menambahkan, mengubah, dan menghapus data pertanyaan beserta jawabannya. Setiap perubahan akan langsung ditampilkan pada halaman *FAQ* milik *User*, sehingga informasi yang tersedia tetap diperbarui dan dapat dimanfaatkan sebagai solusi awal untuk mengurangi jumlah laporan terkait permasalahan teknis sederhana.

6. Implementasi Tampilan Fitur Chat *IT Support* dan *User*

Pada tahap implementasi, hasil perancangan fitur *chat* telah direalisasikan ke dalam aplikasi *Helpdesk Ticketing System* berbasis web sebagai media komunikasi antara *User* dan *IT Support*. Fitur ini dirancang untuk mendukung proses koordinasi selama penanganan laporan tiket sehingga pengguna dan *IT Support* dapat bertukar informasi secara langsung. Hasil implementasi fitur *chat* dapat dilihat pada Gambar berikut:



Gambar di atas menunjukkan implementasi fitur *chat* yang dapat diakses oleh *User* dan *IT Support* pada aplikasi *Helpdesk Ticketing System*. Fitur ini dapat diakses melalui halaman *Dashboard* pada masing-masing pengguna, yaitu melalui tombol *Chat* yang tersedia pada setiap data laporan tiket. Ketika tombol tersebut dipilih, sistem akan menampilkan ruang percakapan yang terhubung dengan laporan tiket terkait sehingga komunikasi dapat dilakukan secara terfokus sesuai dengan permasalahan yang sedang ditangani.

Melalui fitur *chat* ini, *User* dan *IT Support* dapat bertukar informasi, dan menyampaikan perkembangan proses penanganan laporan secara langsung (*real-time*). Implementasi fitur ini diharapkan dapat meningkatkan efektivitas komunikasi selama proses penyelesaian gangguan layanan teknologi informasi serta mempermudah koordinasi antara *User* dan *IT Support* dalam menyelesaikan setiap laporan tiket.

B. HASIL PENGUJIAN

Setelah proses implementasi selesai, dilakukan pengujian sistem menggunakan metode *Black Box Testing* untuk memverifikasi bahwa setiap fungsi pada *Helpdesk Ticketing System* telah berjalan sesuai dengan kebutuhan fungsional yang telah ditetapkan. Metode ini menguji kesesuaian antara masukan (*input*) yang diberikan dengan keluaran (*output*) yang dihasilkan oleh sistem tanpa berfokus pada implementasi kode program.

Pengujian dilakukan terhadap seluruh fitur utama sistem, meliputi *Login*, *Dashboard*, pembuatan laporan tiket, penanganan laporan tiket, menu *Helpdesk*, pengelolaan *FAQ*, fitur *Chat*, verifikasi akun *User*, pengelolaan akun *User*, hingga proses

**IMPLEMENTASI METODE FIRST COME FIRST SERVED PADA SISTEM HELPDESK
TICKETING BERBASIS WEB GUNA MENGOPTIMALKAN RESPONSE TIME LAYANAN
IT SUPPORT**

Logout. Hasil pengujian menunjukkan bahwa seluruh fitur dapat berfungsi sesuai dengan skenario yang telah dirancang, sehingga sistem mampu memenuhi kebutuhan fungsional pengguna. Ringkasan hasil pengujian menggunakan metode *Black Box Testing* ditunjukkan pada Tabel dibawah ini.

No	Fitur Sistem	Skenario Pengujian	Hasil Yang Diharapkan	Status Pengujian
1.	Form <i>Login</i>	<i>User, IT Support, dan Admin</i> menginputkan <i>username</i> dan <i>password</i> yang valid.	Pengguna berhasil masuk ke halaman <i>Dashboard</i> sesuai dengan hak akses masing-masing.	Berhasil
2.	Dashboard	<i>User, IT Support, dan Admin</i> membuka halaman <i>Dashboard</i> setelah berhasil <i>login</i> .	Halaman <i>Dashboard</i> tampil sesuai dengan hak akses masing-masing serta menampilkan daftar laporan tiket.	Berhasil
3.	Pembuatan Laporan Tiket	<i>User</i> mengklik tombol Tambah Laporan, kemudian mengisi formulir laporan yang terdiri dari jenis keluhan, deskripsi laporan, dan unggah gambar kendala, selanjutnya menekan tombol Tambah.	Data laporan tiket berhasil disimpan dan ditampilkan pada halaman <i>Dashboard</i> dan <i>Helpdesk</i> .	Berhasil
4.	Penanganan Laporan Tiket	<i>IT Support</i> memilih tiket, mengisi penanganan, kemudian menekan tombol Simpan.	Data penanganan berhasil disimpan, status tiket berubah menjadi Close, dan solusi penanganan ditampilkan kepada <i>User</i> .	Berhasil
5.	Kelola <i>FAQ</i>	<i>IT Support</i> menambah, mengubah, atau menghapus data <i>FAQ</i> .	Perubahan data <i>FAQ</i> berhasil disimpan dan ditampilkan pada sistem.	Berhasil
6.	Menu <i>Helpdesk</i>	<i>User, IT Support, dan Admin</i> membuka menu <i>Helpdesk</i> .	Sistem menampilkan tabel detail laporan tiket sesuai dengan hak akses masing-masing.	Berhasil
7.	Fitur Chat	<i>User</i> dan <i>IT Support</i> saling mengirim dan menerima pesan.	Pesan berhasil dikirim, diterima, dan ditampilkan pada kedua pengguna secara <i>real-time</i> .	Berhasil

8.	Tampilan FAQ	User mengklik tombol Tambah Laporan untuk membuat laporan tiket baru.	Sistem menampilkan daftar FAQ yang telah dibuat oleh <i>IT Support</i> sebagai panduan sebelum <i>User</i> membuat laporan tiket.	Berhasil
----	--------------	---	---	----------

Berdasarkan hasil pengujian pada Tabel diatas, seluruh fitur utama pada *Helpdesk Ticketing System* memperoleh status berhasil. Hasil tersebut menunjukkan bahwa sistem telah mampu menjalankan setiap fungsi sesuai dengan kebutuhan fungsional yang dirancang, sehingga dapat mendukung proses pelaporan dan penanganan gangguan layanan teknologi informasi dengan baik.

PEMBAHASAN HASIL

1. Analisis Ketercapaian Tujuan Penelitian

Hasil implementasi menunjukkan bahwa *Helpdesk Ticketing System* berbasis web berhasil memenuhi tujuan penelitian sebagai media pelaporan dan penanganan gangguan layanan teknologi informasi. Sistem mampu menyediakan proses pelaporan yang lebih terstruktur, memudahkan *IT Support* dalam menangani laporan menggunakan metode *First Come First Served (FCFS)*, serta meningkatkan kemudahan pengguna melalui fitur *FAQ* dan *Chat*.

2. Analisis Hasil Pengujian dan Efektivitas Sistem

Berdasarkan hasil pengujian menggunakan metode *Black Box Testing*, seluruh fitur utama sistem berhasil berjalan sesuai dengan kebutuhan fungsional yang telah dirancang. Implementasi sistem membantu meningkatkan efektivitas pengelolaan laporan melalui proses pencatatan yang terintegrasi, penyampaian informasi secara *real-time*, serta kemudahan pemantauan status penanganan laporan oleh *User* maupun *IT Support*.

3. Analisis Penerapan Metode Waterfall

Penerapan metode *Waterfall* memberikan alur pengembangan sistem yang sistematis, mulai dari tahap perencanaan, analisis kebutuhan, perancangan, implementasi, hingga pengujian. Pendekatan ini memudahkan proses pengembangan karena setiap tahapan diselesaikan secara berurutan, sehingga sistem yang dihasilkan dapat memenuhi kebutuhan pengguna sesuai dengan tujuan penelitian.

5. KESIMPULAN DAN SARAN

KESIMPULAN

Berdasarkan hasil penelitian, perancangan, implementasi, dan pengujian sistem yang telah dilakukan, maka dapat disimpulkan dalam beberapa point utama sebagai berikut :

1. Sistem layanan bantuan (*Helpdesk Ticketing System*) berbasis *web* telah berhasil dirancang dan dikembangkan dengan memanfaatkan *framework* yang tepat sehingga mampu memfasilitasi alur pelaporan hingga penanganan keluhan teknis secara sistematis, terstruktur, dan efisien dibandingkan dengan metode konvensional.
2. Sistem membantu pengguna menyelesaikan permasalahan teknis ringan secara mandiri melalui penyediaan basis data solusi pada fitur panduan *troubleshooting (FAQ)* yang dapat memungkinkan pengguna untuk mencari, membaca, dan mengikuti langkah-langkah penanganan secara mandiri sebelum memutuskan untuk membuat dan mengirimkan tiket laporan ke tim *IT Support*.

3. Sistem telah menerapkan fitur manajemen status tiket dan pemantauan prioritas yang memungkinkan tim *IT Support* memantau progress penyelesaian masalah secara *real-time*, sehingga secara langsung dapat meningkatkan kinerja tim IT dalam mengalokasikan sumber daya secara lebih efektif. Peningkatan efektivitas komunikasi berhasil dicapai melalui integrasi fitur *chat* dua arah dan sistem notifikasi pada *email* yang mampu menjembatani interaksi antara pengguna dengan tim teknis secara lebih cepat, transparan, dan terintegrasi dalam satu platform.
4. Peningkatan efektivitas komunikasi berhasil dicapai melalui integrasi fitur *chat* dua arah dan sistem notifikasi pada *email* yang mampu menjembatani interaksi antara pengguna dengan tim teknis secara lebih cepat, transparan, dan terintegrasi dalam satu platform.

SARAN

Berdasarkan hasil penelitian dan pengembangan sistem yang telah dilakukan, terdapat beberapa saran untuk pengembangan sistem di masa mendatang, yaitu sebagai berikut:

1. Sistem perlu dilengkapi dengan integrasi *API WhatsApp* secara otomatis sebagai saluran notifikasi tambahan untuk meningkatkan fleksibilitas dan kecepatan respons bagi *administrator* dalam melakukan penanganan keluhan pengguna, sehingga dapat menggantikan penggunaan sistem notifikasi *email* saat ini demi meningkatkan efisiensi operasional.
2. Penelitian selanjutnya dapat mengembangkan fitur *dashboard* analitik visual berupa grafik tren kendala bulanan yang akan mempermudah tim *IT Support* dalam memetakan potensi masalah secara strategis guna merumuskan langkah-langkah pencegahan yang bersifat preventif.
3. Pengembangan antarmuka sistem ke dalam bentuk aplikasi *mobile* perlu dilakukan untuk memberikan kemudahan bagi pengguna dalam melaporkan kendala teknis secara *real-time* di mana pun dan kapan pun, sehingga integrasi layanan dapat berjalan dengan lebih komprehensif dan *responsive* sesuai dengan kebutuhan aksesibilitas yang dinamis di lingkungan kerja.

DAFTAR REFERENSI

- Albaqir, A. H., Putra, A. R., & Saifudin, A. (2023). *Pengujian Aplikasi Pembelian Tiket Online menggunakan Metode Pengujian Black Box Testing Berbasis Equivalence Partitions*. 1(4), 1–9.
- Azis, A. A., & Voutama, A. (2025). *RANCANGAN SISTEM MONITORING SISWA BERBASIS WEB DENGAN METODE WATERFALL*. 13(3).
- Dachi, J. V., Suhada, J., Informatika, T., Komputer, F. I., & Pamulang, U. (2025). *Implementasi Basis Data Sederhana Menggunakan MySQL / PostgreSQL sistematis untuk merancang , mengimplementasikan , dan menguji basis data sederhana pada*. 3(1), 49–59.
- Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2022). *Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website*. 8(200), 48–55.
- Fauzan Prasetyo Eka Putra, Reynal Widya Efendi, Alief Badrit Tamam, W. A. P. (2025). *TREN DAN PRAKTIK TERBAIK DALAM PENGEMBANGAN WEB BERBASIS API: KAJIAN LITERATUR TERHADAP FRAMEWORK*. 27, 165–178. <https://doi.org/10.23969/infomatek.v27i1.25122>

- Huwaidy, M. F., Voutama, A., & Timur, T. (n.d.). *WEBSITE SISTEM INFORMASI ALUMNI PESANTREN*. 13(3), 296–304.
- Maliki, I., & Sulistiyowati, R. (2022). *Perancangan Dan Implementasi Pemodelan Sistem Informasi Pemesanan Tiket Online Di Galeri Nasional Indonesia*. 16, 74–79.
- Maulidiah, S., Encep, M., Susanti, A. R., Satria, R. Y., & Elgar, M. (2025). *Rancang Bangun Basis Data Untuk Sistem Informasi Terintegrasi Tata Usaha dan Laboratorium*. 5, 1132–1140.
- Nugroho, F., & Ali, H. (2022). *DETERMINASI SIMRS : HARDWARE , SOFTWARE DAN BRAINWARE*. 3(1), 254–265.
- Prayitno, M. H., & Nurmalisa, A. (2024). *PENERAPAN ALGORITMA FIRST COME FIRST SERVE*. 2(1), 32–43.
- Putra, T. S., Ma, H., Bandung, P. L. P. I., No, J. P., & Barat, J. (2024). *Perancangan UI / UX Aplikasi Jemput Sampah Berbasis Mobile UI / UX Design of Mobile-Based Trash Pickup Application*.
- Ramadhani, Y., Safikri, N. S., Kusuma, B. A., Elektro, T., Soedirman, U. J., Informasi, T., Purwokerto, U. A., & Process, A. H. (2024). *OPTIMASI OPERASIONAL HELPDESK : SELEKSI SISTEM TIKET*. 1(1), 106–117.
<https://doi.org/10.61124/sinta.v1i2.17>
- Retnoningsih, D., Vandika, A. Y., & Fuadi, A. (n.d.). *Inovasi Terbaru Dalam Rekayasa Perangkat Lunak Ilmu Komputer Penulis Editor :*
- Sinlae, F., Irwanda, E., Maulana, Z., & Syahputra, V. E. (2024). *Penggunaan Framework Laravel dalam Membangun Aplikasi Website Berbasis PHP*. 2(2), 119–132.
- Tech, J. K. (2022). *Jurnal KHARISMA Tech* |. 02, 184–198.