



Pengembangan Aplikasi Website Helpdesk IT Dengan Implementasi Algoritma Round Robin Untuk Distribusi Tiket Studi Kasus: Corporate IT Lintasarta

Rio Fachrudin¹, R. Wisnu Prio Pamungkas², Fried Sinlae³

^{1,2,3} Fakultas Ilmu Komputer, Universitas Bhayangkara Jakarta Raya, Bekasi, Indonesia

Article Info

Article history:

Received Juli 1, 2025
Revised Juli 10, 2025
Accepted Juli 15, 2025

Kata Kunci:

IT Helpdesk,
Round Robin,
IT Corporate,
Web,
Agile SDLC

Keywords:

IT Helpdesk,
Round Robin,
IT Corporate,
Web,
Agile SDLC

ABSTRAK

Proses distribusi tiket helpdesk IT yang dilakukan secara manual oleh admin di lingkungan Corporate IT Lintasarta menimbulkan sejumlah kendala seperti beban kerja teknisi yang tidak merata, ketergantungan pada peran admin, serta tidak adanya sistem monitoring yang dapat menelusuri riwayat distribusi. Kondisi ini mengakibatkan pelayanan IT internal menjadi kurang efisien dan sulit dikendalikan. Penelitian ini bertujuan untuk mengembangkan aplikasi berbasis web yang mampu mengotomatisasi distribusi tiket menggunakan algoritma Round Robin. Algoritma ini dipilih karena mampu mendistribusikan beban kerja secara bergiliran dan adil kepada teknisi aktif yang terdaftar di sistem. Penelitian ini menggunakan metode pengembangan sistem SDLC Agile agar proses pengembangan dapat disesuaikan secara bertahap dengan kebutuhan pengguna. Hasil dari penelitian ini berupa sistem web helpdesk yang dapat menggantikan distribusi manual menjadi sistematis dan otomatis, serta memiliki fitur monitoring untuk melacak distribusi tiket berdasarkan teknisi dan waktu. Dengan adanya sistem ini, perusahaan diharapkan mampu meningkatkan efisiensi operasional, transparansi, serta pemerataan beban kerja di divisi IT. Penelitian selanjutnya diharapkan dapat memperluas fitur sistem, termasuk prioritas tiket dan integrasi dengan layanan eksternal.

ABSTRACT

The manual ticket distribution process in the Corporate IT division of Lintasarta presents several issues, such as unequal workload among technicians, heavy reliance on admin intervention, and the absence of a monitoring system to track ticket history. These issues lead to inefficiencies and a lack of control in IT service operations. This research aims to develop a web-based helpdesk application capable of automating ticket distribution using the Round Robin algorithm. This algorithm is chosen for its ability to assign workload evenly and in turn to all active technicians registered in the system. The development process follows the Agile SDLC methodology, allowing iterative development based on user needs. The result of this research is a web-based helpdesk system that replaces the manual distribution process with a systematic and automated approach, including a monitoring feature to track ticket assignments by technician and time. With the implementation of this system, the company is expected to achieve better operational efficiency, transparency, and equal workload distribution within the IT division. Future research is expected to expand the system's capabilities, such as ticket prioritization and integration with external services.

This is an open access article under the [CC BY](https://creativecommons.org/licenses/by/4.0/) license.

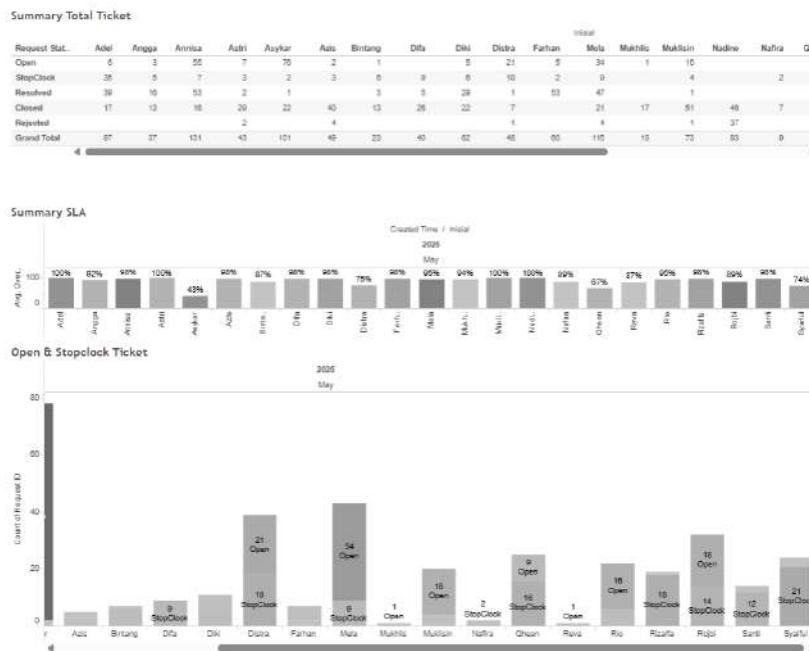


Corresponding Author:

R. Wisnu Prio Pamungkas
 Fakultas Ilmu Komputer, Universitas Bhayangkara Jakarta Raya,
 Bekasi, Indonesia
 Email: wisnu.prio@dsn.uhharajaya.ac.id

1. PENDAHULUAN

Di perusahaan penyedia layanan IT Lintasarta yang berlokasi di MH Thamrin, Jakarta Pusat, efektivitas penanganan masalah teknis menjadi salah satu fokus utama dalam mendukung produktivitas karyawan. Permasalahan yang muncul dari aplikasi internal maupun sistem yang digunakan sehari-hari perlu segera ditangani oleh divisi Corporate IT agar operasional perusahaan tidak terganggu. Untuk itu, Website Corporate IT dikembangkan sebagai sarana pelaporan masalah atau kendala teknis melalui sistem tiket (helpdesk ticketing system).



Gambar 1. Grafik Persebaran Tiket Aduan

Berdasarkan data persebaran total tiket gangguan bulan Mei 2025, terlihat adanya ketimpangan distribusi beban kerja antar personel IT (PIC aplikasi). Contohnya, personel Mela tercatat menangani 115 tiket, dengan 34 tiket masih dalam status Open, yang merupakan angka tertinggi dibandingkan PIC lainnya. Sementara itu, ada juga PIC seperti Mukhis dan Syaiful yang hanya menangani masing-masing 18 dan 9 tiket secara total.

Distribusi beban kerja yang tidak merata ini turut tercermin pada indikator kinerja SLA (Service Level Agreement). Beberapa PIC memiliki tingkat penyelesaian 100%, seperti Adel, Annisa, dan Nadine, sedangkan PIC seperti Asykar hanya mencapai 43%, yang mengindikasikan adanya hambatan dalam penanganan tiket yang bisa jadi disebabkan oleh beban kerja yang terlalu tinggi atau alokasi tiket yang kurang tepat.

Selain itu, pada kategori Open & Stopclock Ticket, PIC seperti Mela dan Distran memiliki angka Open Ticket dan Stopclock Ticket tertinggi, masing-masing dengan 34 open tickets dan 18 stopclock tickets. Hal ini menunjukkan adanya potensi penumpukan tiket yang belum terselesaikan tepat waktu.

Kondisi ini menunjukkan bahwa proses distribusi tiket saat ini belum optimal dan masih bergantung pada keputusan manual. Ketidakseimbangan ini dapat berdampak pada kinerja layanan, efisiensi penyelesaian masalah, serta kepuasan pengguna internal. Oleh karena itu, diperlukan sistem distribusi tiket yang adil, efisien, dan terotomatisasi untuk memastikan beban kerja merata, SLA terpenuhi, dan produktivitas seluruh personel IT dapat dimaksimalkan [1] [2].

Namun, dalam implementasinya, proses distribusi tiket helpdesk kepada personel IT masih dilakukan secara manual atau berdasarkan penilaian subjektif dari admin sistem. Hal ini dapat menyebabkan ketidakseimbangan beban kerja, penundaan penyelesaian tiket, hingga berkurangnya kepuasan pengguna layanan [3], [4]. Kondisi ini menunjukkan perlunya sistem distribusi tiket yang adil, efisien, dan terotomatisasi [5]. [16]

Salah satu solusi yang dapat diterapkan untuk mengatasi permasalahan tersebut adalah dengan mengimplementasikan algoritma Round Robin. Algoritma Round Robin merupakan metode penjadwalan dengan pendekatan preemptive yang umum digunakan untuk mendistribusikan sumber daya secara merata dan sistematis. Dalam konteks sistem operasi, algoritma ini memberikan setiap proses jatah waktu eksekusi yang sama, disebut time quantum, dan mengeksekusinya secara bergiliran hingga seluruh proses terselesaikan [6] [17]

Automasi penugasan tiket dengan algoritma Round Robin terbukti dapat mengurangi waktu respons dan mempercepat penyelesaian masalah [7]. Sistem ini juga mencegah penumpukan beban pada satu personel IT dengan menyalurkan tiket secara bergilir [8]. Studi eksperimental menunjukkan bahwa round robin lebih efisien dibanding metode manual dalam konteks helpdesk yang memiliki volume tiket tinggi [9]. Bahkan dalam varian load-aware, sistem mempertimbangkan beban kerja sebelumnya untuk pengalokasian yang lebih adil [10].

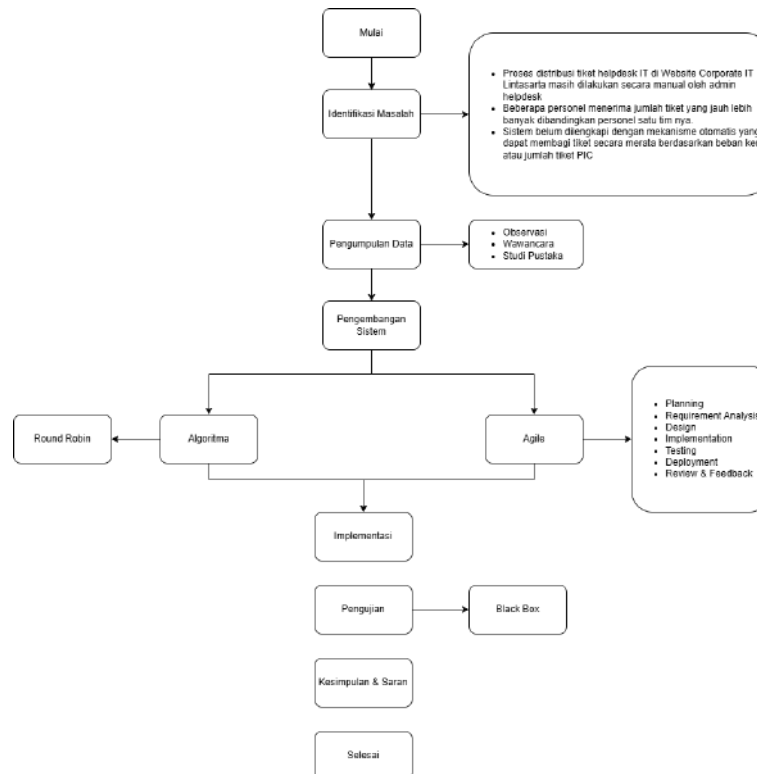
Implementasi sistem ini juga terbukti dapat meningkatkan kepuasan pengguna layanan IT dan mengurangi waktu penyelesaian rata-rata tiket hingga lebih dari 20% [11]. Dalam beberapa studi, penerapan algoritma distribusi tiket secara otomatis juga membantu menghindari praktik cherry-picking oleh agen helpdesk, yaitu memilih hanya tiket yang mudah diselesaikan [12]. Selain itu, automasi ini mendukung transparansi dan akuntabilitas dalam pelaporan kinerja agen IT [13].

Strategi adopsi Round Robin juga relatif mudah dilakukan karena telah tersedia pada berbagai platform ITSM (IT Service Management), seperti Freshdesk, Zoho, atau ManageEngine, yang memungkinkan aktivasi hanya melalui pengaturan admin [14]. Untuk memastikan efektivitas, dibutuhkan penyusunan aturan (rules) dan pemicu (triggers) yang tepat agar sistem benar-benar berjalan otomatis setelah tiket masuk [15].

2. METODE

2.1 Kerangka Penelitian

Untuk membantu dalam penyusunan penelitian ini, maka perlu adanya susunan kerangka penelitian yang jelas tahap – tahapannya. Kerangka penelitian ini merupakan langkah – langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas. Adapun kerangka penelitian yang digunakan sebagai berikut :



Gambar 2. Kerangka Penelitian

2.2 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan pada penelitian ini ada 3 yaitu:

2.2.1 Observasi

Penulis melakukan pengamatan secara langsung terhadap masalah yang diteliti. Dengan metode ini, peneliti dapat mengamati secara langsung kendala yang muncul, alur kerja administrasi, dan efektivitas sistem yang ada tanpa mempengaruhi jalannya proses tersebut

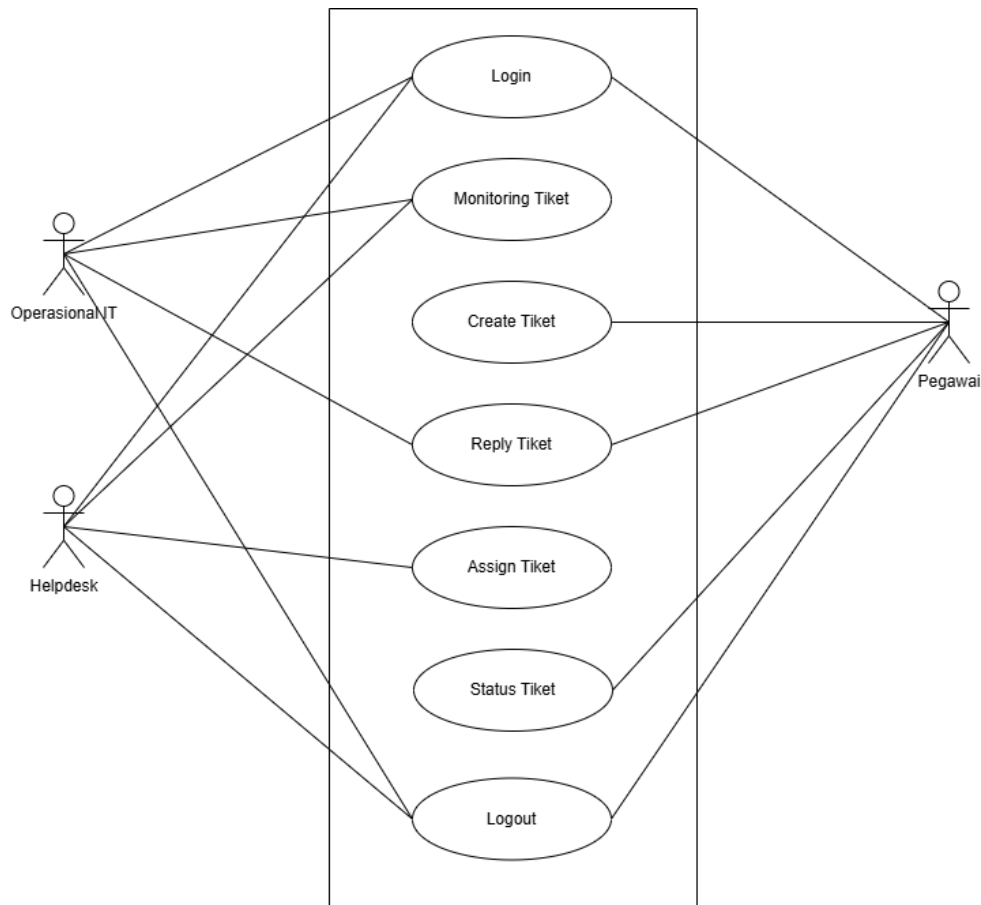
2.2.2 Wawancara

Pada data berisi pertanyaan dan jawaban hasil dari wawancara peneliti kepada narasumber yaitu Leader Operasional Div Corporate IT Perusahaan Lintasarta.

2.3 Analisis Sistem Berjalan

Analisis system berjalan bertujuan untuk mengetahui bagaimana cara kerja system yang sedang berjalan dan masalah yang dihadapi oleh sistem. Dan untuk selanjutnya akan dijadikan usulan perancangan untuk sistem yang akan dikembangkan nantinya.

2.3.1 Use Case Diagram



Gambar 3. Use Case Diagram Sistem Berjalan

3. HASIL DAN PEMBAHASAN

3.1 Perhitungan Algoritma Round Robin

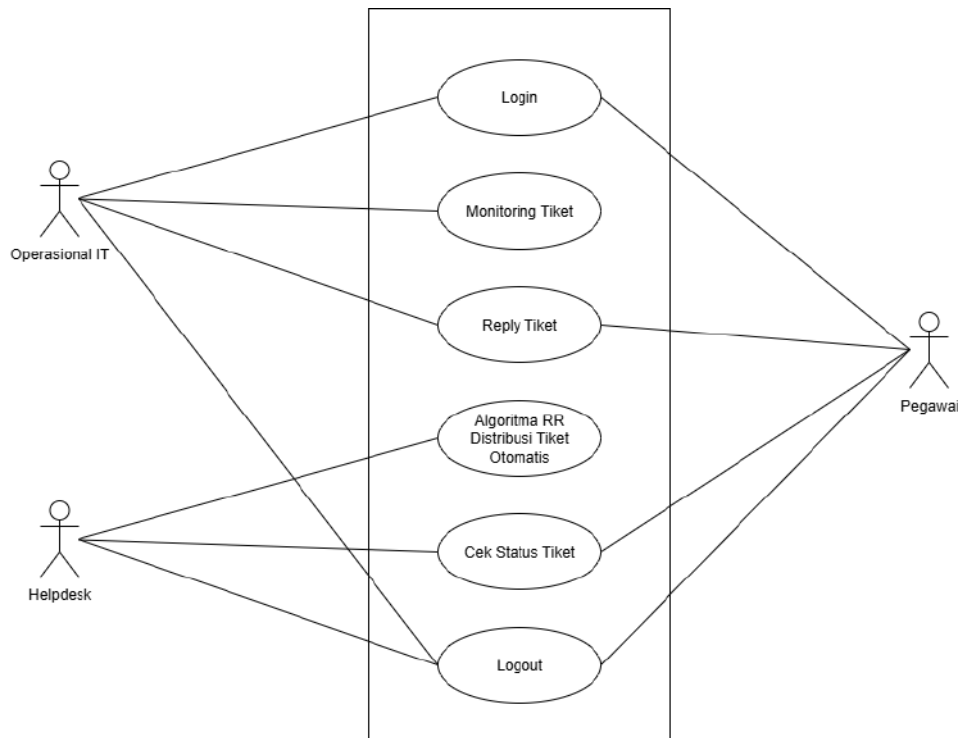
Algoritma yang diterapkan pada sistem informasi helpdesk IT ini difokuskan untuk mendistribusikan tiket secara merata kepada teknisi yang aktif dengan pendekatan adil dan terukur menggunakan algoritma Round Robin. Dalam penelitian ini, data yang digunakan adalah data simulasi tiket yang masuk ke sistem helpdesk Corporate IT Lintasarta. Data ini digunakan sebagai sampel untuk menguji efisiensi distribusi tiket menggunakan algoritma Round Robin. Pemilihan data tiket dilakukan dengan mempertimbangkan variasi waktu kedatangan dan estimasi durasi penanganan (burst time) untuk merepresentasikan kondisi riil pada sistem helpdesk.

3.2 Perancangan Sistem

Perancangan sistem bertujuan untuk memberikan gambaran menyeluruh mengenai arsitektur sistem yang akan dikembangkan. Adapun tahapan perancangan yang dilakukan meliputi perancangan sistem menggunakan diagram UML yaitu *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

3.2.1 Use Case Diagram

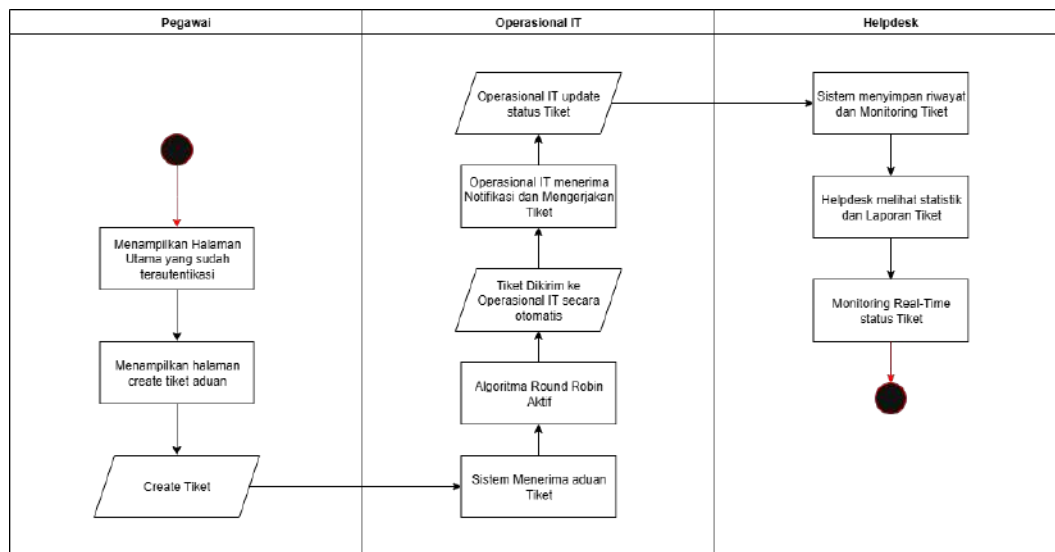
Berikut adalah Use Case Diagram dari sistem Helpdesk IT sebelum penerapan algoritma Round Robin:



Gambar 4. Sequence Diagram Asign Tiket

3.2.2 Activity Diagram

Berikut merupakan activity diagram dari sistem Helpdesk IT sesudah penerapan algoritma Round Robin:



Gambar 5. Activity Diagram Penerapan

3.2.3 Class Diagram

Gambar 5 menampilkan Class Diagram sistem Helpdesk IT setelah dilakukan pengembangan dan penerapan algoritma Round Robin sebagai metode pendistribusian tiket. Diagram ini menunjukkan struktur kelas-kelas utama dalam sistem serta hubungan antar objek yang mendukung proses otomatisasi penugasan tiket. Dalam sistem ini terdapat tujuh kelas utama: Pegawai, Helpdesk, Teknisi, Tiket, LogTiket, serta satu kelas tambahan yaitu RoundRobinManager yang menjadi kunci dalam pengambilan keputusan teknisi secara otomatis.

3.3 Proses Perencanaan *Agile*

Pada penelitian ini, proses pengembangan aplikasi web Helpdesk IT untuk Corporate IT Lintasarta menggunakan pendekatan Agile Software Development Life Cycle (Agile SDLC). Metode ini dipilih karena bersifat iteratif dan fleksibel, memungkinkan pengembang dan stakeholder untuk berkolaborasi secara aktif dalam mengembangkan sistem yang adaptif terhadap perubahan kebutuhan dan kondisi nyata di lapangan. Agile SDLC membagi proses pengembangan ke dalam beberapa sprint atau iterasi waktu yang singkat, di mana setiap sprint memiliki tujuan tertentu untuk menyelesaikan bagian dari sistem.

3.4 Sprint 0 : Perencanaan Awal dan Analisis Kebutuhan

3.4.1 Durasi Sprint

Tabel 1. Durasi Sprint 0

Tanggal	1 – 5 Juni 2025
Durasi	5 Hari
Tujuan	Menganalisis kebutuhan sistem, menyusun backlog awal, dan menyepakati lingkungan serta alur kerja pengembangan sistem Helpdesk IT berbasis Round Robin.

Identifikasi kebutuhan pengguna dilakukan pada tanggal 1 Juni 2025 melalui diskusi langsung antara peneliti dan stakeholder dari Divisi Corporate IT Lintasarta, yaitu Leader Operasional yang terlibat langsung dalam proses pengelolaan tiket gangguan internal. Tujuan dari diskusi ini adalah untuk menggali informasi mendalam terkait proses bisnis yang sedang berjalan, kendala yang dihadapi, serta harapan terhadap sistem yang akan dikembangkan.

Tabel 2. Hasil Diskusi dengan *stakeholder*

No	Pertanyaan	Jawaban
1	Bagaimana proses pembuatan tiket dilakukan saat ini?	Masih dilakukan secara manual melalui email atau chat ke admin helpdesk.
2	Apakah ada kendala dalam pembagian tiket ke teknisi?	Ya, pembagian tiket masih manual dan tidak merata.
3	Apakah admin membutuhkan fitur monitoring teknisi?	Iya, untuk mengetahui siapa yang sedang mengerjakan tiket dan bagaimana status tiketnya.
4	Apakah pengguna perlu melihat status tiket mereka sendiri?	Iya, supaya pengguna tidak perlu menanyakan progress ke admin atau teknisi.
5	Apa harapan terhadap sistem baru yang akan dibangun?	Harapannya sistem ini lebih cepat, otomatis, dan adil dalam mendistribusikan tiket.

3.4.2 Sprint Timeline

Tabel 2. Timeline *Sprint 0*

Aktivitas	1	2	3	4	5
Kick-off meeting dengan stakeholder Corporate IT, identifikasi kebutuhan					
Penyusunan product backlog					
Penetapan alur kerja tim dan tools pengembangan					
Diskusi teknis: arsitektur awal, database, dan alur sistem					
Finalisasi perencanaan dan validasi backlog awal dengan stakeholder					

3.4.3 Sprint Board

Tabel 3. *Sprint 0* Board

No	Backlog Fitur	Status	Catatan
1.	Halaman Create Tiket	Disiapkan	Draft flow user input tiket
2.	Otomatisasi distribusi tiket	Disiapkan	Akan dikembangkan dengan algoritma RR
3.	Riwayat dan Monitoring Tiket	Disiapkan	Monitoring berbasis status
4.	Halaman cek status tiket	Disiapkan	Akan dihubungkan dengan status otomatis
5.	Penentuan tools & stack teknologi	Selesai	PHP, MySQL, Oracle, AdminLTE, CI4, Git, XAMPP
6.	Penjadwalan sprint dan pembagian tugas	Selesai	Sprint 1 dimulai 6 Februari

3.4.4 Definition of Done (DoD)

Tabel 4. DOD *Sprint 0*

No	Deliverable	Kriteria	Checklist
1.	Identifikasi kebutuhan pengguna	Hasil wawancara atau diskusi dengan stakeholder terdokumentasi dalam bentuk tabel dan dianalisis secara deskriptif.	√
2.	Penyusunan product backlog	Minimal 4 fitur utama (Create Tiket, Distribusi Tiket, Monitoring, Cek Status Tiket) tercantum dan divalidasi.	√
3.	Penetapan tools & platform pengembang	Tools utama seperti CodeIgniter 4, MySQL, AdminLTE, Git, XAMPP ditentukan dan disetujui oleh tim.	√
4.	Penjadwalan sprint	Timeline pengembangan disusun (Sprint 0-4) dan durasi tiap sprint ditetapkan secara realistis.	√
5.	Pembagian tugas & peran tim	Tugas untuk pengembang backend, frontend, scrum lead ditentukan dan didokumentasikan.	√
6.	Validasi stakeholder	Semua hasil Sprint 0 dikonfirmasi kepada stakeholder Corporate IT dan dinyatakan disetujui.	√

3.5 Sprint 1 – Perancangan Sistem (System Design)

Sprint 1 difokuskan pada tahap perancangan sistem Helpdesk IT berbasis web untuk Corporate IT Lintasarta. Sprint ini merupakan tahap penting dalam proses pengembangan karena menghasilkan blueprint awal dari sistem yang akan dibangun. Pada sprint ini, dilakukan perancangan alur interaksi, struktur data, serta antarmuka pengguna untuk memastikan sistem memenuhi kebutuhan pengguna yang telah diidentifikasi pada Sprint 0.

3.5.1 Durasi Sprint

Tabel 5. Durasi *Sprint 1*

Tanggal	6 – 12 Juni 2025
Durasi	7 Hari
Tujuan	Merancang struktur sistem, antarmuka, dan proses distribusi tiket yang akan dikembangkan pada sprint berikutnya.

3.5.2 Sprint Timeline

Tabel 6. Timeline *Sprint 1*

Aktivitas	6	7	8	9	10	11	12
Review product backlog dan pembagian tugas desain							
Pembuatan Use Case Diagram dan Activity Diagram							
Pembuatan Sequence Diagram dan struktur alur Round Robin							
Perancangan Struktur Database (ERD) dan skema relasi tabel							
Perancangan Antarmuka (Wireframe) halaman utama dan form tiket							
Dokumentasi rancangan sistem dan integrasi semua desain							
Sprint review dan validasi hasil perancangan bersama stakeholder							

3.6 Sprint 2 – Implementasi Fitur Distribusi Tiket

3.6.2 Durasi *Sprint*

Tabel 7. Durasi *Sprint 2*

Tanggal	13 – 18 Juni 2025
Durasi	6 Hari
Tujuan	Menerapkan algoritma Round Robin ke dalam sistem distribusi tiket helpdesk IT secara otomatis.

3.6.3 *Sprint Timeline*

Tabel 8. Timeline *Sprint 2*

Aktivitas	13	14	15	16	17	18
Pemahaman detail alur distribusi manual dan penyusunan pseudocode algoritma						
Implementasi algoritma Round Robin dalam modul distribusi tiket						
Integrasi modul RR ke sistem backend & testing awal						
Pengujian dan debugging logika algoritma						
Pengujian end-to-end alur distribusi otomatis						
Validasi stakeholder dan dokumentasi implementasi						

3.6.4 Hasil *Sprint*

1. Analisis distribusi tiket manual: Tim menganalisis alur saat ini di mana admin memilih teknisi secara subjektif, mencatat waktu rata-rata alokasi teknisi ± 2 menit per tiket.
2. Penyusunan algoritma RR: Menggunakan array queue dengan pointer index berputar. Simulasi dilakukan dengan 5 teknisi aktif menunjukkan pemerataan alokasi dalam 10 tiket.
3. Implementasi kode PHP

Menggunakan tabel teknisi aktif dan indeks distribusi, dipadukan SQL untuk menambahkan tiket baru ke teknisi. Berikut ini hasil implementasi algoritma ke dalam sistem yang berjalan saat ini :

```
// Assign ticket using Round Robin
function assignTicketRoundRobin(ticket) {
  let agentType = ticket.requestType === 'application' ? 'application' : 'infrastructure';
  let agentPool = agents[agentType].filter(agent =>
    agent.handledApps.includes(ticket.subcategory.toLowerCase())
  );

  if (agentPool.length === 0) {
    console.error('No agents available for this subcategory');
    return;
  }

  // Sort agents by the number of assigned tickets (least assigned first)
  agentPool.sort((a, b) => a.assignedTickets.length - b.assignedTickets.length);

  // Assign the ticket to the agent with the least assigned tickets
  const agent = agentPool[0];
  ticket.assignedAgent = agent.name;
  agent.assignedTickets.push(ticket.id);

  // Update the index for the next assignment
  if (agentType === 'application') {
    currentAppAgentIndex = (agentPool.indexOf(agent) + 1) % agentPool.length;
  } else {
    currentInfraAgentIndex = (agentPool.indexOf(agent) + 1) % agentPool.length;
  }

  updateMonitoring();
}
```

Gambar 6. Hasil Implementasi Algoritma

4. Integrasi ke sistem existing: Tiket baru akan langsung ditandai dengan teknisi hasil algoritma.
5. Debug dan uji: Simulasi 50 tiket ke 3 teknisi memperlihatkan distribusi 17-17-16. Waktu proses rata-rata turun menjadi 0.3 detik per tiket.
6. Validasi stakeholder: Stakeholder menyatakan sistem kini lebih adil, mengurangi beban manual.

3.6.5 Sprint Board

Tabel 9. Sprint 2 Board

No	Backlog Fitur	Status	Catatan
1.	Implementasi algoritma RR	Selesai	Algoritma berjalan berdasarkan urutan teknis
2.	Integrasi dengan create tiket	Selesai	Tiket baru otomatis didistribusikan
3.	Debugging & Validasi RR	Selesai	Sudah dites berbagai skenario teknisi

3.6.6 Definition of Done (DOD)

Tabel 10. DOD Sprint 2

No	Deliverable	Kriteria	Checklist
1.	Algoritma Round Robin	Mampu mendistribusi tiket ke teknisi secara merata.	√
2.	Integrasi dengan sistem helpdesk	Modul terhubung ke proses create tiket dan database teknisi.	√
3.	Pengujian dan validasi fitur	Lolos uji dengan 95% skenario serta validasi stakeholder.	√

3.7 Sprint 3 – Pengembangan Fitur Monitoring

3.7.1 Durasi Sprint

Tabel 11. Durasi Sprint 3

Tanggal	20 – 25 Juni 2025
Durasi	6 Hari
Tujuan	Mengembangkan fitur monitoring distribusi tiket dan riwayat teknisi.

3.7.2 Sprint Timeline

Tabel 12. Timeline Sprint 3

Aktivitas	20	21	22	23	24	25
Perancangan tampilan monitoring & perumusan kebutuhan data						
Pengembangan modul pemantauan berdasarkan waktu & teknisi						
Integrasi modul dengan database distribusi tiket						
Uji coba tampilan & perbaikan visualisasi data						
Penyesuaian filter & fitur sorting						
Validasi stakeholder						

3.7.3 Hasil Sprint

1. Perancangan UI monitoring: Menggunakan wireframe dengan layout tabel teknisi-tanggal-status.
2. Pembuatan query distribusi: SQL dikembangkan untuk menampilkan data 30 hari terakhir.
3. Implementasi visualisasi data: Menggunakan DataTables dan filter drop-down.
4. Testing: Uji coba pada 150 data distribusi, respons < 1 detik.
5. Stakeholder review: Validasi melalui demo fitur dan disetujui.

3.7.4 Sprint Board

Tabel 13. Sprint 3 Board

No	Backlog Fitur	Status	Catatan
1.	Halaman monitoring tiket	Selesai	Data teknisi, waktu, dan status ditampilkan rapi
2.	Modul filter & sort	Selesai	Sorting per status & tanggal
3.	Validasi stakeholder	Selesai	Hasil valid sesuai ekspektasi

3.7.5 Definition of Done (DOD)

Tabel 14. DOD Sprint 3

No	Deliverable	Kriteria	Checklist
1.	Monitoring distribusi	Tampil nama teknisi, status, tanggal	√
2.	Filter & sort	Fitur sorting per status dan tanggal aktif.	√
3.	Validasi stakeholder	Telah diuji dan disetujui.	√

3.8 Sprint 4 – Pengujian, Evaluasi, dan Perbaikan

3.8.1 Durasi Sprint

Tabel 15. Durasi Sprint 4

Tanggal	26 – 30 Juni 2025
Durasi	5 Hari
Tujuan	Melakukan pengujian sistem secara keseluruhan dan validasi akhir.

3.8.2 Sprint Timeline

Tabel 15. Timeline Sprint 4

Aktivitas	20	21	22	23	24
Penyusunan skenario pengujian (black box testing)					
Pengujian fungsional tiap fitur					
Perbaiki bug minor & dokumentasi testing					
Pengujian UAT (User Acceptance Testing) bersama stakeholder					
Finalisasi evaluasi dan persiapan deployment					

3.8.3 Hasil Sprint

1. Penyusunan skenario uji: 10 skenario uji, mencakup fitur inti sistem.
2. Uji fungsional oleh QA: Dilakukan selama 2 hari, bug ditemukan pada validasi input.
3. Perbaiki bug dan uji ulang: Masalah diselesaikan dengan validasi regex dan logika tambahan.
4. UAT dengan stakeholder: Disimulasikan alur 10 tiket, sistem lolos uji.
5. Persiapan deployment: Disiapkan file konfigurasi, backup, dan SOP implementasi.

3.8.4 Sprint Board

Tabel 16. Sprint Board 4

No	Backlog Fitur	Status	Catatan
1.	Pengujian black box	Selesai	Semua fitur lolos skenario uji
2.	UAT	Selesai	Validasi akhir dengan stakeholder
3.	Perbaiki bug minor	Selesai	Masalah validasi & tampilan disesuaikan

3.8.5 Definition of Done (DOD)

Tabel 17. DOD Sprint 4

No	Deliverable	Kriteria	Checklist
1.	Laporan pengujian	Minimal 10 skenario, disertai hasil uji dan bukti screenshot	√
2.	Sistem final	Semua fitur berjalan tanpa error kritis.	√
3.	Validasi akhir	Disetujui oleh stakeholder.	√

4. KESIMPULAN

Penelitian Ini Berhasil Mengembangkan Sistem Aplikasi Web Helpdesk IT Yang Mengotomatisasi Proses Distribusi Tiket Pada Lingkungan Corporate IT Lintasarta. Sistem Ini Menggantikan Proses Manual Yang Sebelumnya Dilakukan Oleh Admin Secara Langsung. Penerapan Algoritma Round Robin Terbukti Efektif Dalam Membagi Tiket Secara Bergiliran Dan Adil Kepada Teknisi Yang Aktif, Sehingga Membantu Pemerataan Beban Kerja Dan Menghindari Penumpukan Tugas Pada Teknisi Tertentu. Penggunaan Metode Pengembangan Agile SDLC Memungkinkan Sistem Dibangun Secara Bertahap Dan Fleksibel, Dengan Memperhatikan Masukan Pengguna Di Setiap Tahap Sprint. Hal Ini Meningkatkan Efektivitas Pengembangan Serta Kesesuaian Sistem Terhadap Kebutuhan Nyata Di Lapangan. Sistem Yang Dikembangkan Juga Dilengkapi Dengan Fitur Monitoring Yang Memungkinkan Admin Melihat Riwayat Dan Status Distribusi Tiket, Sehingga Proses Menjadi Lebih Transparan, Terukur, Dan Mudah Untuk Dievaluasi.

REFERENSI

- [1] Desk365, "Round Robin Ticket Assignment in Helpdesk," *Desk365*, 2024. [Online]. Available: <https://www.desk365.io/round-robin-ticket-assignment/>
- [2] Sobot, "Automated Ticket Assignment: A Complete Guide," *Sobot Help Center*, 2025. [Online]. Available: <https://www.sobot.io/article/automated-ticket-assignment-guide/>
- [3] ProProfs Help Desk, "What Is Automated Ticket Routing?," *ProProfsDesk*, 2025. [Online]. Available: <https://www.proprofsdesk.com/blog/automated-ticket-routing/>
- [4] Freshdesk, "Setting Up Round Robin Ticket Assignment," *Freshdesk Support*, 2024. [Online]. Available: <https://support.freshdesk.com/support/solutions/articles/221904-setting-up-round-robin-ticket-assignment>
- [5] Gmelius, "What is Round Robin Assignment?," *Gmelius Blog*, 2025. [Online]. Available: <https://gmelius.com/blog/what-is-round-robin-assignment/>
- [6] R. Jain and A. Tripathi, "Efficient Load Balancing using Round Robin Algorithm in Cloud Environment," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 6, no. 3, pp. 35–39, 2020. [Online]. Available: <https://doi.org/10.32628/CSEIT206321>
- [7] A. Akhter, M. M. Rahman, and S. Khan, "Comparative Analysis of Task Scheduling Algorithms in Cloud Computing," *International Journal of Computer Applications*, vol. 178, no. 10, pp. 1–5, 2021. [Online]. Available: <https://doi.org/10.5120/ijca2021921423>
- [8] H. J. Singh and N. Kaur, "An Optimized Load Distribution Algorithm Using Modified Round Robin in Cloud Computing," *International Journal of Innovative Science and Research Technology*, vol. 5, no. 4, pp. 406–410, 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.3745456>
- [9] S. Patel and M. Dave, "A Review on Round Robin Algorithm and Its Variants in Cloud Environment," *Journal of Emerging Technologies and Innovative Research*, vol. 7, no. 12, pp. 221–226, 2021. [Online]. Available: <https://doi.org/10.6084/m9.figshare.14758661.v1>
- [10] ManageEngine, "Load Balancing and Ticket Assignment Automation," *ManageEngine ServiceDesk Plus*, 2023. [Online]. Available: <https://www.manageengine.com/products/service-desk/automation/assign-tickets-load-balancing.html>
- [11] M. K. Gupta and P. R. Kumar, "Performance Evaluation of SLA Based Task Scheduling in Cloud Using Round Robin," *International Journal of Cloud Applications and Computing (IJCAC)*, vol. 11, no. 3, pp. 45–60, 2021. [Online]. Available: <https://doi.org/10.4018/IJCAC.2021070103>
- [12] InvGate, "Automatic Ticket Assignment at a Service Desk: Benefits and Best Practices," *InvGate Blog*, 2023. [Online]. Available: <https://blog.invgate.com/automatic-ticket-assignment-at-a-service-desk>
- [13] DeskPro, "Round Robin Not Working Properly?," *DeskPro Support*, 2025. [Online]. Available: <https://support.deskpro.com/bg/kb/articles/im-having-trouble-with-the-round-robin-working-properly>
- [14] Freshdesk, "Auto-Assignment Methods: Explained," *Freshdesk Support*, 2024. [Online]. Available: <https://support.freshdesk.com/support/solutions/articles/215267-automatic-ticket-assignment>
- [15] A. Bhattacharya et al., "An AI-driven Auto Ticket Assignment System," *arXiv preprint arXiv:2103.05970*, 2021. [Online]. Available: <https://arxiv.org/abs/2103.05970>
- [16] R. W. P. Pamnungkas, A. D. Alexander, and A. Reza, "Perancangan Sistem Informasi Helpdesk Menggunakan Website Design Metode Dalam Mendukung Tata Kelola Teknologi Informasi," <http://ejurnal.tunasbangsa.ac.id/index.php/jsakti/article/view/141/123>.
- [17] M. Darip, S. Auliana, and B. Rakhim Setya Permana, "Pengelolaan Waktu Kerja Karyawan Menggunakan Algoritma Round_Robin Dalam Sistem Informasi Manajemen Sdm Di Perusahaan

Outsourcing,” *INFOTECH journal*, vol. 11, pp. 28–36, Feb. 2025, doi:
10.31949/infotech.v11i1.12946.