



**ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING*  
SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL  
BERBASIS *WEB* PADA PT PRIMA INDAH LESTARI**

**SKRIPSI**

**Oleh:**

Rhehuel Junaidi

20210700021

Program Studi Sistem Informasi

Fakultas Sains dan Teknologi

**UNIVERSITAS BUDDHI DHARMA**

**TANGERANG**

**2025**



**ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING*  
SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL  
BERBASIS *WEB* PADA PT PRIMA INDAH LESTARI**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk mendapatkan gelar Sarjana Komputer  
pada Program Studi Sistem Informasi Fakultas Sains dan Teknologi

Universitas Buddhi Dharma

Jenjang Pendidikan Strata 1

**Oleh:**

Rhehuel Junaidi

20210700021

Program Studi Sistem Informasi

Fakultas Sains dan Teknologi

**UNIVERSITAS BUDDHI DHARMA**

**TANGERANG**

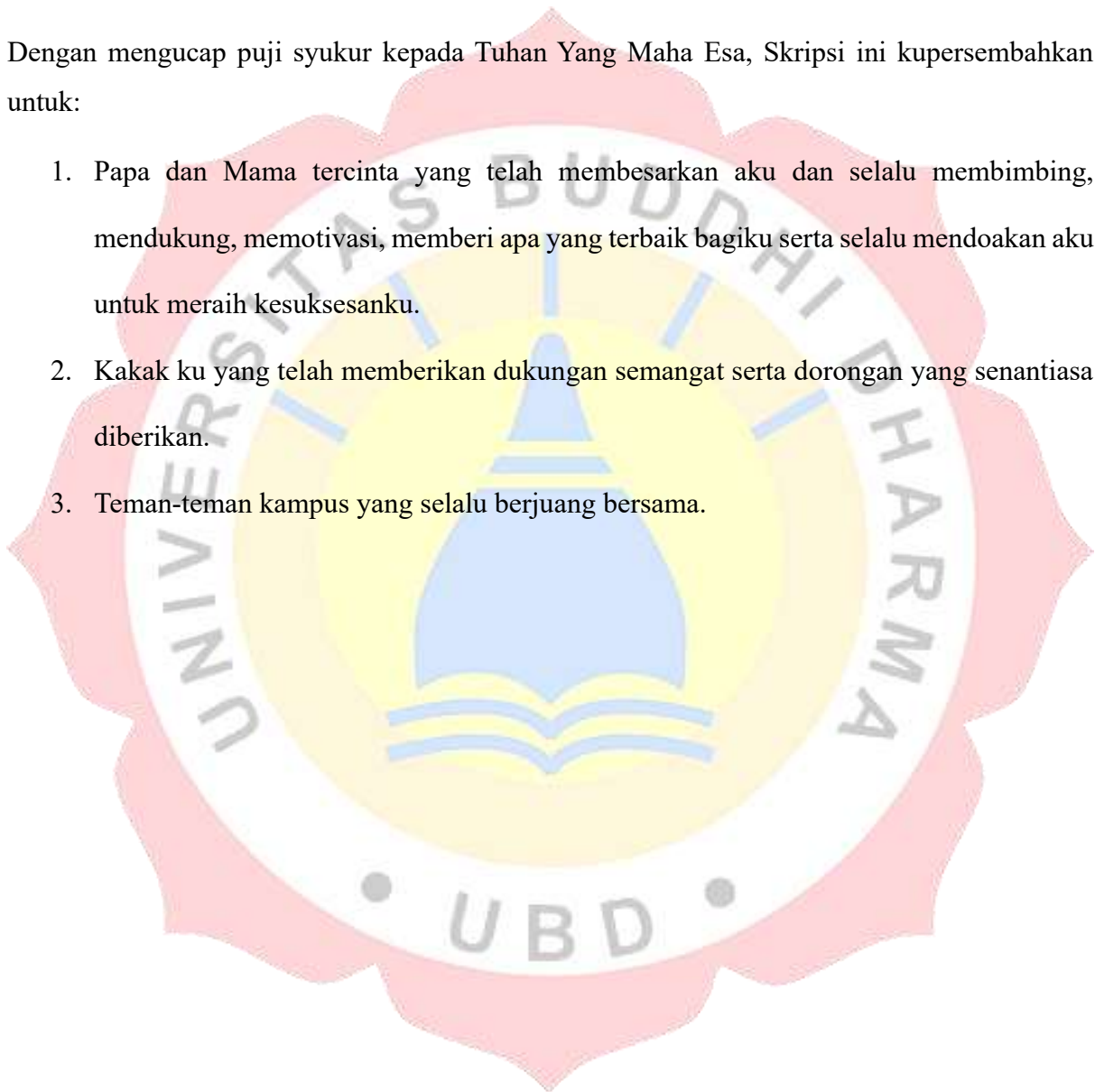
**2025**

## LEMBAR PERSEMBAHAN

**Skripsi ini bukan sekadar lembaran tugas akhir, tapi bukti bahwa mimpi dan usaha tak pernah saling mengkhianati**

Dengan mengucap puji syukur kepada Tuhan Yang Maha Esa, Skripsi ini kupersembahkan untuk:

1. Papa dan Mama tercinta yang telah membesarkan aku dan selalu membimbing, mendukung, memotivasi, memberi apa yang terbaik bagiku serta selalu mendoakan aku untuk meraih kesuksesanku.
2. Kakak ku yang telah memberikan dukungan semangat serta dorongan yang senantiasa diberikan.
3. Teman-teman kampus yang selalu berjuang bersama.



**UNIVERSITAS BUDDHI DHARMA**  
**LEMBAR PERNYATAAN KEASLIAN SKRIPSI**

Yang telah bertanda tangan dibawah ini.

NIM : 20210700021  
Nama : Rhehuel Junaidi  
Jenjang Studi : Strata 1  
Program Studi : Sistem Informasi  
Peminatan : E-Business

Dengan ini penulis menyatakan bahwa :

1. Skripsi ini benar-benar hasil karya saya sendiri dan belum pernah saya ajukan untuk mendapatkan gelar Sarjana atau syarat kelulusan di Universitas Buddhi Dharma maupun di institusi lain.
2. Saya membuat skripsi ini tanpa bantuan dari pihak lain, kecuali arahan yang saya terima dari dosen pembimbing.
3. Dalam skripsi ini, saya tidak menggunakan karya atau pendapat orang lain tanpa menyebutkan dengan jelas sebagai referensi dalam naskah, dengan mencantumkan nama pengarang dan masuk dalam daftar pustaka.
4. Skripsi ini tidak mengandung pemalsuan seperti buku, artikel, jurnal, data sekunder, pengolahan data, atau tanda tangan dosen atau Ketua Program Studi Universitas Buddhi Dharma, yang semuanya asli dan dapat di pertanggung jawabkan.
5. Saya membuat pernyataan ini dengan kesadaran penuh dan tanpa paksaan. Jika nanti terbukti ada ketidakbenaran dalam pernyataan ini, saya siap menerima sanksi akademik sesuai dengan peraturan yang berlaku di Universitas Buddhi Dharma.

Tangerang, 06-08-2025

Yang membuat pernyataan,



Rhehuel Junaidi  
NIM : 20210700021

# UNIVERSITAS BUDDHI DHARMA

## LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH

Yang beranda tangan dibawah ini.

NIM : 20210700021  
Nama : Rhehuel Junaidi  
Jenjang Studi : Strata 1  
Program Studi : Sistem Informasi  
Peminatan : E-Business

Dengan ini menyetujui untuk memberikan ijin kepada pihak Universitas Buddhi Dharma, Hak Bebas Royalti Non – Eksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah kami yang berjudul: "ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING* SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL BERBASIS *WEB* PADA PT PRIMA INDAH LESTARI", beserta alat yang diperlukan (apabila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini pihak Universitas Buddhi Dharma berhak menyimpan, mengalih-media atau format-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya, mendistribusikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Buddhi Dharma, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Tangerang, 06-08-2025

Yang membuat pernyataan,



Rhehuel Junaidi  
NIM : 20210700021

**UNIVERSITAS BUDDHI DHARMA**

**LEMBAR PENGESAHAN PEMBIMBING**

**ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING*  
SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL BERBASIS  
*WEB* PADA PT PRIMA INDAH LESTARI**

Dibuat Oleh:

NIM : 20210700021

Nama : Rhehuel Junaidi

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian  
Komprehensif

Program Studi Sistem Informasi  
Peminatan *Electronic-Business*

Tahun Akademik 2024/2025

Tangerang, 10 Juli 2025

Disahkan oleh,

**Pembimbing,**



(Andi Leo, S.Kom., M.Kom.)

NUPTK : 9337740641137083

**UNIVERSITAS BUDDHI DHARMA**

**LEMBAR PENGESAHAN SKRIPSI**

**ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING*  
SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL BERBASIS  
*WEB* PADA PT PRIMA INDAH LESTARI**

Dibuat Oleh:

NIM : 20210700021

Nama : Rhehuel Junaidi

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian Komprehensif

Program Studi Sistem Informasi Peminatan

*E-Business*

Tahun Akademik 2024/2025

Tangerang, 06-08-2025

Disahkan oleh,

Dekan,



Dr. Yakub, M.Kom., M.M.

NUPTK : 1836747648130172

Ketua Program Studi



Beny Daniawan, M.Kom

NUPTK : 8756768669130412

## LEMBAR PENGESAHAN TIM PENGUJI

Nama : Rhehucl Junaidi

Nim : 20210700021

Fakultas : Sains dan Teknologi

Judul Skripsi : ANALISIS DAN PERANCANGAN SISTEM APLIKASI BOOKING  
SERVIS DAN REMINDER KENDARAAN OPERASIONAL  
BERBASIS WEB PADA PT PRIMA INDAH LESTARI

Dinyatakan LULUS setelah mempertahankan di depan Tim Penguji pada hari RABU,06-08-2025

Nama Penguji  
Ketua Sidang : **Benny Daniawan, S.Kom.,M.Kom.**  
NUPTK : 8756768669130412

Tanda Tangan:



Penguji I : **Suwitno, S.Kom.,M.Kom.**  
NUPTK : 2845761662130212



Penguji II : **Andi Leo, S.Kom.,M.Kom.**  
NUPTK : 9337740641137083



Mengetahui

**Dekan Fakultas Sains dan Teknologi**



Dr. Yakub, M.Kom., M.M

NUPTK : 1836747648130172

## KATA PENGANTAR

Dengan penuh rasa syukur kepada Tuhan Yang Maha Esa, yang telah melimpahkan berkat dan karunia-Nya kepada saya, sehingga saya dapat menyusun dan menyelesaikan Skripsi ini dengan judul "ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING* SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL BERBASIS *WEB* PADA PT PRIMA INDAH LESTARI". Skripsi ini saya tulis sebagai syarat kelengkapan untuk mencapai kelulusan. Selama proses penyusunan Skripsi ini, saya mendapatkan banyak bantuan dan dorongan baik secara moral maupun materiil dari berbagai pihak. Oleh karena itu, pada kesempatan ini, saya ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Ibu Dr. Limajatini, S.E., M.M., B.K.P. sebagai Rektor Universitas Buddhi Dharma.
2. Bapak Dr. Yakub, M.Kom., M.M. Dekan Fakultas Sains dan Teknologi.
3. Bapak Benny Daniawan, M.Kom. sebagai Ketua Program Studi Sistem Informasi.
4. Bapak Andi Leo, S.Kom.,M.Kom. sebagai pembimbing yang telah membantu dan memberikan dukungan serta harapan untuk menyelesaikan penulisan Skripsi ini.
5. Orang tua dan keluarga yang selalu memberikan dukungan baik moril dan materiil.
6. Teman-teman yang selalu membantu dan memberikan semangat.

Dan juga kepada semua pihak yang telah memberikan kontribusi berarti untuk terwujudnya Skripsi ini. Saya menyadari bahwa tulisan ini belum sempurna, oleh karena itu saya memohon kritik dan saran yang membangun untuk penyempurnaan di masa mendatang. Terakhir, semoga Skripsi ini bermanfaat bagi saya dan bermanfaat pula bagi para pembaca.

Tangerang, 06-08-2025

Penulis

# ANALISIS DAN PERANCANGAN SISTEM APLIKASI *BOOKING* SERVIS DAN *REMINDER* KENDARAAN OPERASIONAL BERBASIS *WEB* PADA PT PRIMA INDAH LESTARI

208 Halaman + xxi / 45 Tabel/ 86 Gambar/ 10 Lampiran

## ABSTRAK

Penelitian ini membahas tentang analisis dan perancangan sistem manajemen kendaraan operasional berbasis *web* pada PT Prima Indah Lestari, perusahaan manufaktur kabel yang berlokasi di Jakarta Barat. Dalam era transformasi digital, pengelolaan aset bergerak khususnya kendaraan operasional menjadi faktor krusial dalam menjaga efisiensi operasional perusahaan. Permasalahan utama yang dihadapi meliputi ketiadaan dokumentasi riwayat pemeliharaan yang sistematis, sulitnya mengontrol penggunaan kendaraan, pengelolaan yang masih dilakukan secara manual dan terpisah-pisah, serta tidak adanya sistem pengingat otomatis untuk jadwal servis. Kondisi ini mengakibatkan seringnya terjadi keterlambatan pemeliharaan, peningkatan biaya perawatan, dan penurunan umur pakai kendaraan. Metodologi *waterfall* dipilih sebagai pendekatan pengembangan dengan lima fase: analisis kebutuhan, perancangan sistem, implementasi, integrasi dan pengujian, serta pemeliharaan berkelanjutan. Pengumpulan data dilakukan melalui observasi lapangan, wawancara terstruktur, dan kajian literatur. Arsitektur sistem dibangun menggunakan *framework Laravel, PHP dan JavaScript* untuk logika bisnis, *MySQL* sebagai basis data, serta *bootstrap* untuk antarmuka responsif. Pemodelan sistem mengadopsi *Unified Modeling Language* menghasilkan *use case diagram, activity diagram, sequence diagram, dan class diagram*. Solusi yang dikembangkan mengintegrasikan modul pemesanan servis digital, *dashboard* pengingat otomatis, repositori data terpusat, dan pencatatan riwayat servis. Sistem mengakomodasi tiga pengguna: Admin mengelola data master, Admin bengkel mengatur jadwal dan persetujuan, serta Pengemudi melakukan pengajuan dan *monitoring*. Pengujian *black box testing* memvalidasi seluruh modul berfungsi optimal. Implementasi sistem diproyeksikan meningkatkan efektivitas manajemen armada, meminimalkan *downtime* kendaraan, mengoptimalkan anggaran pemeliharaan, serta menyediakan basis data terintegrasi untuk pengambilan keputusan strategis. Sistem ini menjadi solusi transformasi digital mendukung efisiensi operasional perusahaan dalam pengelolaan aset kendaraan.

**Kata Kunci:** Manajemen Kendaraan, Sistem *Booking*, Sistem *Reminder* Servis, *User Acceptance Testing*

# *ANALYSIS AND DESIGN OF WEB-BASED BOOKING SERVICE AND OPERATIONAL VEHICLE REMINDER APPLICATION SYSTEM AT PT PRIMA INDAH LESTARI*

*208 Pages + xxi / 45 Tables/ 86 Pictures/ 10 References*

## **ABSTRACT**

*This research discusses the analysis and design of a web-based operational vehicle management system at PT Prima Indah Lestari, a cable manufacturing company located in West Jakarta. In the era of digital transformation, managing mobile assets, especially operational vehicles, has become a crucial factor in maintaining company operational efficiency. The main problems faced include the absence of systematic maintenance history documentation, difficulty in controlling vehicle usage, management that is still carried out manually and separately, as well as the lack of an automatic reminder system for service schedules. These conditions result in frequent maintenance delays, increased maintenance costs, and reduced vehicle lifespan. The waterfall methodology was chosen as the development approach with five phases: requirements analysis, system design, implementation, integration and testing, and continuous maintenance. Data collection was conducted through field observation, structured interviews, and literature review. The system architecture was built using Laravel framework, PHP and JavaScript for business logic, MySQL as the database, and Bootstrap for responsive interface. System modeling adopts Unified Modeling Language producing use case diagram, activity diagram, sequence diagram, and class diagram. The developed solution integrates digital service booking modules, automatic reminder dashboard, centralized data repository, and service history recording. The system accommodates three users: Admin manages master data, Workshop Admin manages schedules and approvals, and Drivers perform submissions and monitoring. Black box testing validates all modules function optimally. System implementation is projected to increase fleet management effectiveness, minimize vehicle downtime, optimize maintenance budgets, and provide an integrated database for strategic decision-making. This system becomes a digital transformation solution supporting the company's operational efficiency in vehicle asset management.*

**Keywords:** *Vehicle Management, Booking System, Service Reminder System, User Acceptance Testing*

## DAFTAR ISI

<b>LEMBAR PERSEMBAHAN</b> .....	<b>iii</b>
<b>LEMBAR PERNYATAAN KEASLIAN SKRIPSI</b> .....	<b>iv</b>
<b>LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH</b> .....	<b>iv</b>
<b>LEMBAR PENGESAHAN PEMBIMBING</b> .....	<b>v</b>
<b>LEMBAR PENGESAHAN SKRIPSI</b> .....	<b>vii</b>
<b>LEMBAR PENGESAHAN TIM PENGUJI</b> .....	<b>viii</b>
<b>KATA PENGANTAR</b> .....	<b>ix</b>
<b>ABSTRAK</b> .....	<b>x</b>
<i>ABSTRACT</i> .....	<b>xi</b>
<b>DAFTAR ISI</b> .....	<b>xii</b>
<b>DAFTAR TABEL</b> .....	<b>xvi</b>
<b>DAFTAR GAMBAR</b> .....	<b>xviii</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xxi</b>
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Identifikasi Masalah .....	4
1.3 Ruang Lingkup Masalah .....	5
1.4 Tujuan Dan Manfaat.....	5
1.5 Sistematika Penulisan.....	7
<b>BAB II LANDASAN TEORI</b> .....	<b>9</b>
2.1 Analisis Dan Perancangan.....	9
2.2 Sistem Informasi Manajemen (SIM).....	9
2.3 Sistem <i>Booking</i> Servis.....	10
2.4 <i>Automasi</i> .....	10
2.5 Sistem <i>Reminder</i> .....	11

2.6 Manajemen Armada .....	11
2.7 Pemeliharaan <i>Preventif</i> .....	11
2.8 Sistem Manajemen Berbasis <i>Web</i> .....	12
2.9 Basis Data.....	12
2.10 <i>Laravel</i> .....	13
2.11 <i>PHP</i> .....	14
2.12 <i>Javascript</i> .....	14
2.13 Metodologi <i>Waterfall</i> .....	15
2.14 Perancangan <i>UML</i> .....	17
2.15 Tinjauan Studi .....	25
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>31</b>
3.1 Gambaran Umum Perusahaan.....	31
3.1.1 Sejarah Perusahaan.....	31
3.1.2 Visi dan Misi.....	31
3.1.3 Struktur Organisasi.....	32
3.1.4 Tugas & Wewenang.....	33
3.2 Teknik Pengumpulan Data .....	37
3.3 Prosedur Sistem Berjalan .....	40
3.4 <i>Activity Diagram</i> Sistem <i>Booking</i> Servis Berjalan .....	41
3.5 <i>Use Case Diagram</i> Sistem Berjalan.....	42
3.6 <i>Activity Diagram</i> Sistem Usulan.....	43
3.7 Dokumentasi <i>Input</i> Dan <i>Output</i> .....	44
3.7.1 Dokumentasi <i>Input</i> .....	44
3.7.2 Dokumentasi <i>Output</i> .....	45
3.8 Analisis Masalah .....	46
3.9 Identifikasi Kebutuhan Sistem .....	48
3.10 Sistem Yang Diusulkan .....	50

3.11 Metodologi Penelitian .....	52
3.12 <i>Requirement Elicitation</i> .....	54
3.12.1 Tahap 1.....	54
3.12.2 Tahap 2.....	55
3.12.3 Tahap 3.....	56
3.12.4 Tahap 4.....	57
3.13 Jadwal Penelitian.....	58
3.14 Kerangka Pemikiran.....	59
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>62</b>
4.1 Prosedur Sistem Usulan .....	62
4.2 Rancangan Sistem Usulan.....	63
4.2.1 <i>Use Case Diagram</i> .....	63
4.2.2 <i>Use Case Diagram Scenario Admin, Admin Bengkel, Driver</i> .....	64
4.2.3 <i>Activity Diagram</i> .....	69
4.2.4 <i>Sequence Diagram</i> .....	82
4.3 Rancangan <i>Database</i> .....	104
4.3.1 <i>Class Diagram</i> .....	104
4.3.2 Struktur <i>File</i> (Spesifikasi Basis Data) .....	104
4.4 Rancangan Tampilan Program .....	112
4.5 Implementasi Sistem .....	139
4.5.1 Tampilan Program .....	139
4.5.2 Spesifikasi <i>Software &amp; Hardware</i> .....	166
4.6 Pengujian <i>Blackbox</i> .....	169
4.6.1 Pengujian <i>Login &amp; Dashboard</i> .....	169
4.6.2 Pengujian <i>Dashboard Driver</i> .....	170
4.6.3 Pengujian Pengajuan Servis.....	171
4.6.4 Pengujian <i>Approval Dashboard</i> .....	172

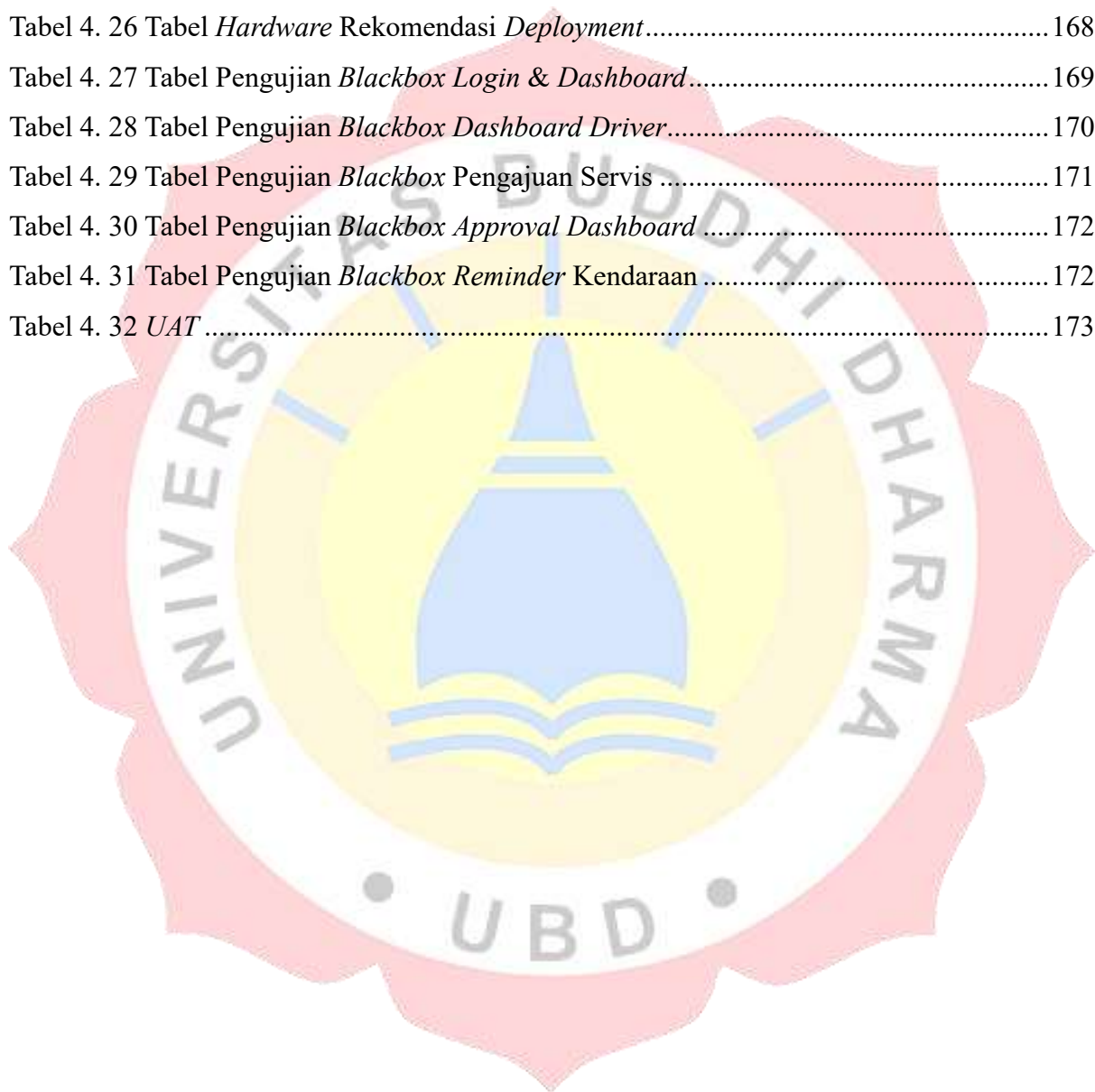
4.6.5 Pengujian <i>Reminder</i> Kendaraan .....	172
4.6 <i>UAT</i> .....	173
BAB V Simpulan Dan Saran .....	176
5.1 Simpulan.....	176
5.2 Saran.....	176
DAFTAR PUSTAKA.....	178
DAFTAR RIWAYAT HIDUP .....	182
LAMPIRAN.....	183



## DAFTAR TABEL

Tabel 2. 1 Simbol-Simbol <i>Activity Diagram</i> .....	18
Tabel 2. 2 Simbol-Simbol <i>Use Case Diagram</i> .....	20
Tabel 2. 3 Simbol-Simbol <i>Class Diagram</i> .....	22
Tabel 2. 4 Simbol-Simbol <i>Sequence Diagram</i> .....	23
Tabel 3. 1 Proses Manual Yang Berlangsung.....	47
Tabel 3. 2 Dampak Ketiadaan Pengingat Otomatis.....	47
Tabel 3. 3 Dampak Sulitnya Melacak Riwayat Pemeliharaan.....	48
Tabel 3. 4 Tantangan Integrasi Data Lama.....	48
Tabel 3. 5 <i>Requirement Elicitation</i> Tahap 1.....	55
Tabel 3. 6 <i>Requirement Elicitation</i> Tahap 1.....	55
Tabel 3. 7 <i>Requirement Elicitation</i> Tahap 3.....	56
Tabel 3. 8 <i>Requirement Elicitation</i> Tahap 4.....	57
Tabel 3. 9 <i>Gantt Chart</i> .....	58
Tabel 4. 1 Prosedur Sistem Usulan.....	62
Tabel 4. 2 <i>Use Case Diagram Scenario Login</i> .....	64
Tabel 4. 3 <i>Use Case Diagram Scenario Logout</i> .....	64
Tabel 4. 4 <i>Use Case Diagram Scenario</i> Kelola Kendaraan.....	64
Tabel 4. 5 <i>Use Case Diagram Scenario</i> Kelola Driver.....	65
Tabel 4. 6 <i>Use Case Diagram Scenario</i> Membuat Laporan.....	65
Tabel 4. 7 <i>Use Case Diagram Scenario</i> Kelola Jadwal Servis.....	66
Tabel 4. 8 <i>Use Case Diagram Scenario</i> Menyetujui Servis.....	66
Tabel 4. 9 <i>Use Case Diagram Scenario</i> Membuat Laporan Servis.....	67
Tabel 4. 10 <i>Use Case Diagram Scenario</i> Melihat Jadwal Servis.....	67
Tabel 4. 11 <i>Use Case Diagram Scenario</i> Booking Servis.....	67
Tabel 4. 12 <i>Use Case Diagram Scenario</i> Menerima Notifikasi <i>Reminder</i> .....	68
Tabel 4. 13 Struktur <i>Data Users</i> .....	104
Tabel 4. 14 Struktur <i>Data Driver</i> .....	105
Tabel 4. 15 Struktur <i>Data Vehicle_List</i> .....	106
Tabel 4. 16 Struktur <i>Data Submission_lists</i> .....	107
Tabel 4. 17 Struktur <i>Vehicle_Reminders</i> .....	109
Tabel 4. 18 Struktur <i>Data Histories</i> .....	110
Tabel 4. 19 Tabel <i>Software Server-Side</i> .....	166

Tabel 4. 20 Tabel <i>Software Client-Side</i> .....	166
Tabel 4. 21 Tabel <i>Software Keamanan</i> .....	166
Tabel 4. 22 Tabel <i>Hardware Server</i> .....	167
Tabel 4. 23 Tabel <i>Hardware Client (Admin&amp;Maintenance)</i> .....	167
Tabel 4. 24 <i>Hardware Client (Driver)</i> .....	167
Tabel 4. 25 Tabel <i>Hardware Infrastruktur Jaringan</i> .....	168
Tabel 4. 26 Tabel <i>Hardware Rekomendasi Deployment</i> .....	168
Tabel 4. 27 Tabel Pengujian <i>Blackbox Login &amp; Dashboard</i> .....	169
Tabel 4. 28 Tabel Pengujian <i>Blackbox Dashboard Driver</i> .....	170
Tabel 4. 29 Tabel Pengujian <i>Blackbox Pengajuan Servis</i> .....	171
Tabel 4. 30 Tabel Pengujian <i>Blackbox Approval Dashboard</i> .....	172
Tabel 4. 31 Tabel Pengujian <i>Blackbox Reminder Kendaraan</i> .....	172
Tabel 4. 32 <i>UAT</i> .....	173



## DAFTAR GAMBAR

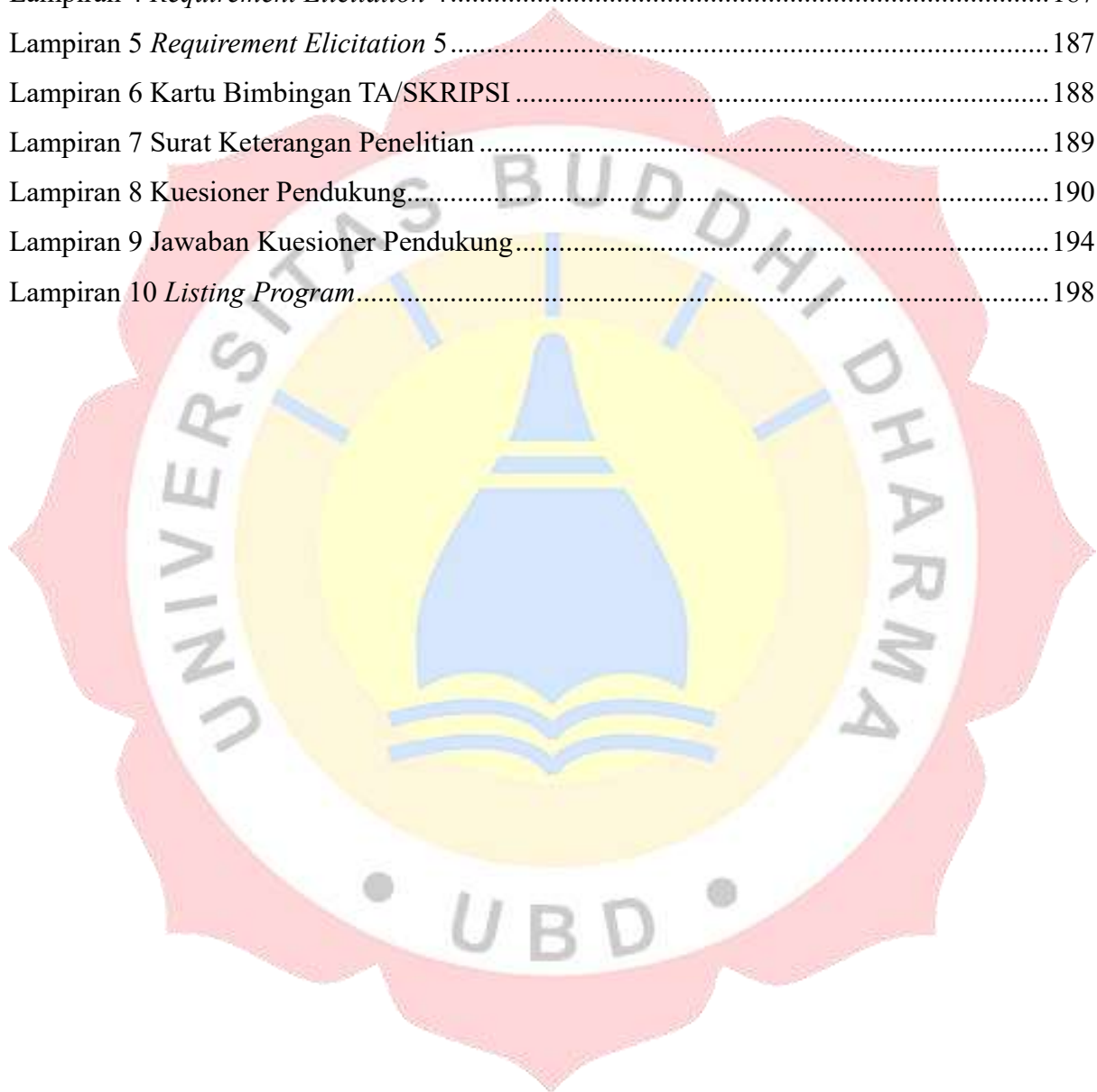
Gambar 2. 1 <i>Waterfall Model</i> .....	17
Gambar 3. 1 Struktur Organisasi .....	32
Gambar 3. 2 <i>Activity Diagram</i> Sistem <i>Booking</i> Servis Berjalan .....	41
Gambar 3. 3 <i>Use Case Diagram</i> Sistem Berjalan.....	42
Gambar 3. 4 <i>Activity Diagram</i> Usulan Proses <i>Booking</i> Servis.....	43
Gambar 3. 5 Kerangka Pemikiran.....	59
Gambar 4. 1 <i>Use Case Diagram</i> .....	63
Gambar 4. 2 <i>Activity Diagram</i> Login <i>Driver</i> .....	69
Gambar 4. 3 <i>Activity Diagram</i> Login Admin Bengkel .....	70
Gambar 4. 4 <i>Activity Diagram</i> Login Admin.....	71
Gambar 4. 5 <i>Activity Diagram</i> Logout.....	72
Gambar 4. 6 <i>Activity Diagram</i> Membuat Laporan Pengajuan.....	73
Gambar 4. 7 <i>Activity Diagram</i> Kelola Kendaraan .....	74
Gambar 4. 8 <i>Activity Diagram</i> Kelola <i>Driver</i> .....	75
Gambar 4. 9 <i>Activity Diagram</i> Kelola Jadwal Servis .....	76
Gambar 4. 10 <i>Activity Diagram</i> Membuat Laporan Servis .....	77
Gambar 4. 11 <i>Activity Diagram</i> Menyetujui <i>Booking</i> Servis .....	78
Gambar 4. 12 <i>Activity Diagram</i> Melihat Jadwal Servis .....	79
Gambar 4. 13 <i>Activity Diagram</i> <i>Booking</i> Servis.....	80
Gambar 4. 14 <i>Activity Diagram</i> Melihat Notifikasi <i>Reminder</i> .....	81
Gambar 4. 15 <i>Sequence Diagram</i> Login Admin.....	82
Gambar 4. 16 <i>Sequence Diagram</i> Login Admin Bengkel.....	83
Gambar 4. 17 <i>Sequence Diagram</i> Login <i>Driver</i> .....	85
Gambar 4. 18 <i>Sequence Diagram</i> Logout <i>User</i> .....	86
Gambar 4. 19 <i>Sequence Diagram</i> <i>Driver</i> Melihat Jadwal Servis.....	87
Gambar 4. 20 <i>Sequence Diagram</i> <i>Driver</i> <i>booking</i> Servis.....	89
Gambar 4. 21 <i>Sequence Diagram</i> Admin Bengkel Acc Servis.....	90
Gambar 4. 22 <i>Sequence Diagram</i> Admin Bengkel Membuat Laporan Servis .....	92
Gambar 4. 23 <i>Sequence Diagram</i> Admin Membuat Laporan Pengajuan .....	94
Gambar 4. 24 <i>Sequence Diagram</i> <i>Reminder</i> <i>Driver</i> .....	96
Gambar 4. 25 <i>Sequence Diagram</i> Admin Bengkel Kelola Jadwal Servis .....	97
Gambar 4. 26 <i>Sequence Diagram</i> Admin Kelola Daftar Kendaraan .....	99

Gambar 4. 27 <i>Sequence Diagram</i> Admin Kelola <i>Driver</i> .....	101
Gambar 4. 28 Tampilan <i>Login</i> Admin .....	112
Gambar 4. 29 Tampilan <i>Dashboard</i> Admin .....	113
Gambar 4. 30 Tampilan Tambah Kendaraan Admin.....	116
Gambar 4. 31 Tampilan <i>List</i> Pengguna (Admin) .....	117
Gambar 4. 32 Tampilan <i>Create User</i> (Admin) .....	118
Gambar 4. 33 Tampilan <i>Monitoring</i> Dan <i>Reminder</i> .....	119
Gambar 4. 34 Tampilan Riwayat Kegiatan Admin .....	120
Gambar 4. 35 Tampilan <i>Master Data</i> <i>Driver</i> (Admin).....	121
Gambar 4. 36 Tampilan Tambah <i>Driver</i> Baru (Admin).....	122
Gambar 4. 37 Tampilan <i>Master Data</i> Kendaraan .....	123
Gambar 4. 38 Tampilan Tambah Kendaraan (Admin).....	124
Gambar 4. 39 Tampilan <i>Master Data Approval</i> (Admin).....	125
Gambar 4. 40 Tampilan Tambah Pic <i>Approval</i> (Admin) .....	126
Gambar 4. 41 Tampilan <i>Master Data</i> Admin (Admin).....	127
Gambar 4. 42 Tampilan Tambah Admin (Admin) .....	128
Gambar 4. 43 Tampilan <i>Login Approval</i> ( <i>Approval</i> ) .....	129
Gambar 4. 44 Tampilan <i>Dashboard</i> ( <i>Approval</i> ) .....	130
Gambar 4. 45 Tampilan Tabel Pengajuan ( <i>Approval</i> ).....	131
Gambar 4. 46 Tampilan Riwayat Kegiatan ( <i>Approval</i> ) .....	132
Gambar 4. 47 Tampilan <i>Login Driver</i> .....	133
Gambar 4. 48 Tampilan <i>Dashboard Driver</i> .....	134
Gambar 4. 49 Tampilan Daftar Pengajuan.....	135
Gambar 4. 50 Tampilan Buat Pengajuan <i>Service</i> Baru .....	136
Gambar 4. 51 Tampilan <i>Request</i> Perubahan Tanggal .....	137
Gambar 4. 52 Tampilan <i>Reminder Driver</i> .....	138
Gambar 4. 53 Tampilan <i>Login</i> (Admin).....	139
Gambar 4. 54 Tampilan <i>Dashboard</i> (Admin).....	140
Gambar 4. 55 Tampilan Tabel Pengajuan (Admin) .....	141
Gambar 4. 56 Tampilan List Kendaraan (Admin) .....	142
Gambar 4. 57 Tampilan Tambah Kendaraan (Admin).....	143
Gambar 4. 58 Tampilan <i>List</i> Pengguna Admin & <i>Approval</i> (Admin) .....	144
Gambar 4. 59 Tampilan <i>Create User</i> (Admin) .....	145
Gambar 4. 60 Tampilan <i>Monitoring</i> & <i>Reminder</i> (Admin).....	146

Gambar 4. 61 Tampilan Riwayat Kegiatan (Admin) .....	147
Gambar 4. 62 Tampilan <i>Master Data Driver</i> (Admin).....	148
Gambar 4. 63 Tampilan Tambah <i>Driver</i> Baru (Admin).....	149
Gambar 4. 64 Tampilan <i>Master Data Kendaraan</i> (Admin) .....	150
Gambar 4. 65 Tampilan Tambah Kendaraan (Admin).....	151
Gambar 4. 66 Tampilan <i>Master Data Pic Approval</i> (Admin).....	152
Gambar 4. 67 Tampilan Tambah <i>Pic Approval</i> Baru (Admin) .....	153
Gambar 4. 68 Tampilan <i>Master Data Admin</i> (Admin).....	154
Gambar 4. 69 Tampilan Tambah Admin Baru (Admin) .....	155
Gambar 4. 70 Tampilan <i>Login (Approval)</i> .....	156
Gambar 4. 71 Tampilan <i>Dashboard (Approval)</i> .....	157
Gambar 4. 72 Tampilan Tabel Pengajuan ( <i>Approval</i> ).....	158
Gambar 4. 73 Tampilan Riwayat Kegiatan ( <i>Approval</i> ) .....	159
Gambar 4. 74 Tampilan <i>Login (Driver)</i> .....	160
Gambar 4. 75 Tampilan <i>Dashboard (Driver)</i> .....	161
Gambar 4. 76 Tampilan Daftar pengajuan servis ( <i>driver</i> ) .....	162
Gambar 4. 77 Tampilan Buat Pengajuan Servis Baru ( <i>Driver</i> ) .....	163
Gambar 4. 78 Tampilan Request Perubahan Tanggal Servis ( <i>Driver</i> ).....	164
Gambar 4. 79 Tampilan <i>Reminder Servis (Driver)</i> .....	165

## DAFTAR LAMPIRAN

Lampiran 1 <i>Requirement Elicitation 1</i> .....	184
Lampiran 2 <i>Requirement Elicitation 2</i> .....	185
Lampiran 3 <i>Requirement Elicitation 3</i> .....	185
Lampiran 4 <i>Requirement Elicitation 4</i> .....	187
Lampiran 5 <i>Requirement Elicitation 5</i> .....	187
Lampiran 6 Kartu Bimbingan TA/SKRIPSI .....	188
Lampiran 7 Surat Keterangan Penelitian .....	189
Lampiran 8 Kuesioner Pendukung.....	190
Lampiran 9 Jawaban Kuesioner Pendukung.....	194
Lampiran 10 <i>Listing Program</i> .....	198



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Di tengah persaingan bisnis yang semakin rumit dan dinamis peningkatan efisiensi operasional menjadi faktor kunci dalam memastikan kelangsungan dan keberhasilan sebuah organisasi. Transformasi digital tidak lagi menjadi pilihan, melainkan keharusan bagi perusahaan yang ingin tetap relevan dan kompetitif di pasar global. Setiap aspek dalam bisnis kini dituntut untuk beradaptasi dengan perkembangan teknologi yang pesat, menghadirkan solusi yang inovatif, mampu meningkatkan efisiensi, menekan biaya, serta menghasilkan data yang akurat untuk pengambilan keputusan strategis. Perusahaan yang berhasil mengintegrasikan teknologi informasi ke dalam proses bisnis mereka, memiliki keunggulan signifikan dalam hal produktivitas, responsivitas terhadap perubahan pasar, serta kemampuan memenuhi ekspektasi pelanggan yang terus meningkat. Salah satu area yang sering terabaikan namun memiliki dampak fundamental terhadap keseluruhan performa bisnis adalah pengelolaan aset fisik yang mendukung mobilitas dan distribusi dalam rantai pasok perusahaan.

Dalam konteks tersebut, manajemen aset bergerak telah menjadi fokus strategis bagi berbagai industri, mulai dari manufaktur hingga jasa. Kebutuhan akan sistem terintegrasi yang dapat mengelola seluruh siklus hidup aset, khususnya yang berkaitan dengan transportasi dan logistik, semakin mendesak seiring dengan meningkatnya kompleksitas bisnis. Dengan adanya sistem manajemen yang komprehensif, perusahaan dapat *memonitor* kondisi aset, memprediksi kebutuhan pemeliharaan, serta mengoptimalkan penggunaan setiap unit untuk memaksimalkan nilai investasi. Tantangan utama yang dihadapi banyak perusahaan adalah bagaimana menerapkan sistem yang tidak hanya efektif dalam

mengelola jadwal pemeliharaan tetapi juga mampu mengintegrasikan data dari berbagai sumber untuk menghasilkan analisis prediktif yang akurat. Implementasi teknologi informasi dalam bidang ini menawarkan solusi yang menjanjikan, dengan kemampuan untuk mengotomatisasi proses, mengurangi kesalahan manusia, serta menyediakan data *real-time* yang esensial untuk pengambilan keputusan cepat dan tepat.

Merawat kendaraan operasional merupakan hal yang penting karena kendaraan tersebut merupakan aset vital yang mendukung kelancaran operasional sehari-hari sebuah perusahaan. Fungsi utama kendaraan operasional mencakup pengiriman barang, mobilitas karyawan, serta mendukung berbagai aktivitas logistik yang krusial bagi kelancaran bisnis. Tanpa perawatan yang baik, kendaraan akan rentan mengalami kerusakan yang dapat mengganggu kelancaran proses bisnis, seperti keterlambatan pengiriman barang. Perawatan rutin juga memastikan kendaraan tetap berfungsi optimal, memperpanjang umur kendaraan, serta mengurangi kemungkinan kerusakan mendadak yang dapat memengaruhi produktivitas perusahaan.

Selain itu, perawatan kendaraan operasional yang tepat dapat mengurangi biaya operasional dalam jangka panjang. Biaya perawatan yang teratur dan pencegahan kerusakan lebih awal jauh lebih murah dibandingkan dengan biaya perbaikan besar atau penggantian kendaraan yang rusak parah. Kendaraan yang tidak terawat dengan baik juga berisiko mengeluarkan biaya tak terduga, seperti peningkatan konsumsi bahan bakar, kerusakan pada komponen mesin, hingga denda akibat kelalaian dalam memenuhi kewajiban administrasi seperti pajak kendaraan. Dengan demikian, perawatan kendaraan operasional yang tepat dan terencana akan membantu perusahaan menghindari kerugian yang tidak perlu serta mendukung keberlanjutan operasional yang lebih efisien dan produktif.

Perkembangan teknologi informasi dan komunikasi telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk dalam manajemen operasional perusahaan. Di era digital ini, banyak perusahaan yang mulai mengadopsi sistem berbasis *web* untuk mengoptimalkan kinerja dan efisiensi operasional mereka, seperti yang sudah diterapkan di PT United Tractors Tbk, Perusahaan tersebut sudah dapat mengakses informasi perawatan kendaraan, pembayaran kendaraan, sehingga aplikasi dapat mempercepat proses bisnis yang sebelumnya dilakukan menggunakan *excel* (Aji Pratama, 2018). PT Prima Indah Lestari, sebagai perusahaan yang bergerak di bidang manufaktur kabel, tidak terkecuali dalam menghadapi tantangan ini. Efektivitas pemeliharaan kendaraan operasional menjadi salah satu fokus utama dalam upaya meningkatkan produktivitas dan efisiensi perusahaan.

Kendaraan operasional memiliki peran penting dalam mendukung aktivitas sehari-hari PT Prima Indah Lestari. Kendaraan ini digunakan untuk berbagai keperluan, mulai dari pengiriman produk hingga mobilitas staff. Ketidakmampuan mengelola kendaraan bagaikan menjerumuskan diri ke jurang kerugian, seperti peningkatan biaya perawatan, penundaan pengiriman, dan penurunan umur kendaraan. Oleh karena itu, diperlukan sistem manajemen kendaraan yang efektif untuk memastikan operasional berjalan lancar.

Namun, pengelolaan kendaraan yang tidak jelas dapat menyebabkan berbagai masalah, seperti terhambatnya pengiriman, melonjaknya biaya perawatan, usia kendaraan yang pendek, dan lupa membayar pajak kendaraan. Oleh karena itu, dibutuhkan sistem manajemen kendaraan yang efektif untuk memaksimalkan operasional dan meminimalisir kerugian.

PT Prima Indah Lestari terletak di Jakarta Barat, Perusahaan ini bergerak di bidang produksi kabel tegangan rendah dan tinggi, dengan nama produk "*Extrana Cable*". *Brand* ini sudah sangat terkenal dan biasanya berpartisipasi dalam proyek pemerintah misalnya

pada proyek stadion JIS. Menurut Andree Lordianto selaku President Direktur PT Prima Indah Lestari “menuturkan pihaknya turut serta berpartisipasi mendukung pembangunan JIS dengan menyediakan kabel berkualitas tinggi dan berteknologi canggih” (Law Malau, 2022).

Permasalahan pada PT Prima Indah Lestari, khususnya pada masalah pengawasan kontrol kendaraan operasional seperti, Ketiadaan dokumentasi riwayat pemeliharaan, Minim nya sistem pengingat otomatis untuk melakukan servis kendaraan, Pengelolaan kendaraan operasional yang masih manual dan terpisah-pisah, Kesulitan dalam *monitoring* dan pelacakan riwayat perawatan kendaraan.

Untuk mengatasi permasalahan tersebut, PT Prima Indah Lestari membutuhkan sebuah sistem servis dan *reminder* kendaraan operasional yang terintegrasi dan terkomputerisasi. Sistem ini akan dirancang untuk mempermudah proses pemesanan, penjadwalan, pemantauan, serta pelaporan terkait penggunaan kendaraan operasional secara lebih akurat, efisien, dan transparan.

## 1.2 Identifikasi Masalah

Berdasarkan permasalahan yang terdapat pada latar belakang yang telah diuraikan, maka dapat diidentifikasi permasalahan yang ditemui adalah sebagai berikut :

1. Proses servis kendaraan masih manual : membutuhkan koordinasi langsung antara *driver* dan divisi *maintenance* yang memakan waktu dan berisiko tinggi terhadap kesalahan penjadwalan.
2. Tidak ada sistem pengingat yang terorganisir : menyebabkan jadwal perawatan kendaraan sering terlewat karena mengandalkan catatan manual yang mudah hilang.
3. Kinerja kendaraan tidak optimal : akibat perawatan yang tidak tepat waktu sehingga menurunkan efisiensi bahan bakar dan memperpendek umur kendaraan.

### 1.3 Ruang Lingkup Masalah

Ruang lingkup masalah dalam penelitian ini meliputi :

1. Sistem ini akan fokus pada pengaturan jadwal servis kendaraan operasional pada PT Prima Indah Lestari, Seperti pembuatan jadwal servis berkala hingga alokasi waktu servis untuk setiap kendaraan
2. Sistem ini akan menyediakan fitur pengingat untuk melakukan perawatan rutin kendaraan melalui *dashboard* yang menampilkan jadwal servis mendatang, seperti penggantian oli rutin, servis berkala, serta pemeriksaan mesin secara berkala.
3. Sistem ini akan menyimpan dan menampilkan histori perawatan setiap kendaraan, termasuk jenis servis yang telah dilakukan, tanggal, dan suku cadang apa saja yang diganti. Data ini akan membantu dalam analisis dan pengambilan keputusan terkait pemeliharaan kendaraan.
4. Sistem ini akan menyediakan fitur akses bagi pengguna yang berbeda, seperti pengelola armada, dan *driver*. Setiap pengguna akan memiliki akses ke fungsi yang sesuai dengan perannya, seperti membuat jadwal, menerima pengingat, atau melihat histori perawatan.

### 1.4 Tujuan Dan Manfaat

#### 1.4.1 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Mengotomatisasi proses servis kendaraan: Aplikasi ini bertujuan untuk menyediakan *platform* berbasis *web* yang memungkinkan karyawan atau pengguna kendaraan operasional untuk melakukan pemesanan jadwal servis secara *online* dengan mudah dan efektif.

2. Implementasi sistem *reminder*: Membuat sistem pengingat untuk pemeliharaan kendaraan yang dapat diakses melalui *dashboard*, sehingga pengguna dapat melihat jadwal servis yang akan datang untuk memastikan tidak ada jadwal servis yang terlewat.
3. Meningkatkan kondisi kendaraan : dengan melakukan pemeliharaan rutin yang tepat waktu, dapat meningkatkan umur pakai kendaraan operasional perusahaan

#### 1.4.2 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah :

1. Aplikasi ini dapat memberikan kemudahan bagi karyawan dalam melakukan pengajuan servis secara *online*, Serta membantu mengingatkan pengguna untuk melakukan servis.
2. Peningkatan Kedisiplinan Pemeliharaan, Sistem *reminder* yang terintegrasi dalam *dashboard* membantu memastikan tidak ada jadwal servis yang terlewat, sehingga pemeliharaan kendaraan dapat dilakukan secara konsisten dan tepat waktu.
3. Implementasi sistem ini dapat memberikan manfaat bagi perusahaan seperti, memperpanjang umur kendaraan sehingga bisa menghemat budget Perusahaan untuk peremajaan armada, membantu perusahaan dalam memenuhi regulasi terkait pemeliharaan keamanan kendaraan operasional.

## 1.5 Sistematika Penulisan

Penulisan laporan penelitian ini dibagi kedalam beberapa bab agar mempermudah

Pencarian isi dari skripsi. Pembagian bab disini terbagi sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini menjelaskan latar belakang dari penulisan penelitian ini, yaitu tentang pentingnya sistem aplikasi servis dan *reminder* kendaraan operasional berbasis *web* di PT Prima Indah Lestari. Identifikasi masalah, ruang lingkup penelitian, tujuan dan manfaat Penelitian, teknik pengumpulan data dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab ini menjelaskan teori umum, teori khusus, teori analisis dan perancangan, teori perancangan *UML*, *Database*, Tinjauan studi serta kerangka pemikiran dalam pembuatan sistem aplikasi servis dan *reminder* kendaraan operasional ini.

### **BAB III METODOLOGI PENELITIAN**

Bab ini menjelaskan tentang gambaran umum perusahaan, prosedur sistem yang telah berjalan, prosedur sistem usulan, *activity diagram* sistem berjalan, dokumentasi *input* dan *output*, analisis masalah, identifikasi kebutuhan sistem, serta jadwal penelitian.

### **BAB IV HASIL DAN PEMBAHASAN**

Pada bab ini akan menjelaskan hasil dari sistem yang dibuat, seperti tampilan *web*, *database*, serta pengujian sistem aplikasi ini.

## **BAB V SIMPULAN DAN SARAN**

Bab ini menjelaskan tentang kesimpulan dari penelitian dan rekomendasi untuk optimalisasi dan pengembangan sistem ke depan.



## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Analisis Dan Perancangan**

Analisis dan perancangan sistem informasi berfokus pada pemahaman kebutuhan pengguna dan bagaimana merancang sistem yang efektif untuk memenuhi kebutuhan tersebut (Hafsari, Arribe Edo, et al., 2024). Analisis sistem mencakup identifikasi dan pemahaman masalah yang ada, serta pengumpulan informasi untuk menentukan spesifikasi fungsionalitas sistem. Proses ini melibatkan pemodelan data, proses bisnis, dan interaksi pengguna dengan sistem. Setelah analisis, perancangan sistem dilakukan dengan menyusun solusi teknis yang mencakup desain antarmuka pengguna (UI), struktur basis data, dan arsitektur perangkat lunak yang sesuai. Dalam perancangan, penting untuk mempertimbangkan efisiensi, skalabilitas, dan keamanan sistem, serta mengoptimalkan pengalaman pengguna. Pendekatan perancangan dapat menggunakan metodologi pengembangan perangkat lunak seperti *Waterfall* atau *Agile*, yang memungkinkan implementasi sistem yang lebih terstruktur dan responsif terhadap perubahan kebutuhan.

#### **2.2 Sistem Informasi Manajemen (SIM)**

Sistem Informasi Manajemen adalah kumpulan komponen yang saling berinteraksi untuk mengumpulkan, menyimpan, dan memproses data yang berguna untuk mendukung pengambilan keputusan dalam suatu perusahaan. Dalam konteks Teknologi Informasi, SIM juga berfungsi untuk mengelola informasi yang diperlukan

untuk menjalankan operasional dan mencapai tujuan strategis perusahaan (Tores et al., 2018).

Penelitian ini berkaitan dengan pengembangan sistem informasi untuk mengelola kendaraan operasional perusahaan. Teori sistem informasi manajemen relevan dalam konteks bagaimana informasi dikumpulkan, diproses, dan digunakan untuk pengambilan keputusan dalam perusahaan.

### **2.3 Sistem *Booking* Servis**

Sistem *booking* servis adalah sistem yang memungkinkan *driver* untuk melakukan pemesanan jadwal servis secara *online*. Dalam konteks penelitian ini, sistem ini bertujuan untuk mengotomatisasi proses pemesanan jadwal servis kendaraan operasional di PT Prima Indah Lestari. Sistem ini menggantikan proses manual yang sebelumnya membutuhkan koordinasi langsung antara *driver* dan divisi *maintenance* (Aditiya Maulana et al., 2024).

### **2.4 *Automasi***

*Automasi* adalah penerapan teknologi untuk menjalankan proses yang sebelumnya dilakukan secara manual, dengan tujuan meningkatkan efisiensi dan mengurangi kelalaian manusia. Dalam konteks manajemen armada, *automasi* dapat diterapkan pada proses servis dan pengingat perawatan kendaraan. Dengan adanya sistem pengingat otomatis, organisasi dapat memastikan bahwa semua jadwal perawatan dilakukan tanpa adanya kelalaian (Zariatina et al., 2016).

## 2.5 Sistem *Reminder*

Sistem *reminder* atau pengingat adalah sistem yang secara otomatis mengirimkan pemberitahuan kepada pengguna mengenai jadwal atau kegiatan yang akan datang. Dalam penelitian ini, sistem *reminder* berbasis *dashboard* digunakan untuk menampilkan jadwal pemeliharaan rutin kendaraan yang dapat diakses oleh *driver* melalui *interface web*, seperti penggantian oli, servis berkala, dan pemeriksaan mesin (Aditiya Maulana et al., 2024).

## 2.6 Manajemen Armada

Manajemen armada adalah proses pengelolaan kendaraan operasional dalam suatu perusahaan untuk memastikan penggunaan yang efisien dan efektif. Proses ini mencakup perencanaan, pengawasan, dan pengendalian kendaraan serta kegiatan terkait seperti pemeliharaan, penjadwalan, dan penganggaran. Tujuan utama dari manajemen armada adalah untuk meningkatkan produktivitas, mengurangi biaya operasional, dan memastikan keselamatan serta kepatuhan terhadap regulasi yang berlaku (Mulyani et al., 2022).

## 2.7 Pemeliharaan *Preventif*

Pemeliharaan *preventif* adalah upaya perawatan yang dilakukan secara berkala untuk menghindari kerusakan mendadak pada peralatan (Deradjad Pranowo Ignatius, 2019). Teori ini menekankan pentingnya pemeliharaan berkala, seperti penggantian oli, pengecekan mesin, atau penggantian suku cadang, untuk menghindari kerusakan yang lebih besar. Dalam penelitian ini, pemeliharaan rutin digunakan pada kendaraan

operasional untuk memperpanjang usia pakai kendaraan dan memastikan kinerjanya tetap optimal. mengurangi frekuensi kerusakan mendadak, serta menjaga kinerja kendaraan tetap optimal. Hal ini berdampak positif pada penurunan biaya operasional, peningkatan ketersediaan kendaraan, dan minimisasi risiko kecelakaan akibat kerusakan komponen kendaraan.

## 2.8 Sistem Manajemen Berbasis *Web*

Penelitian ini menggunakan pendekatan sistem informasi berbasis *web* untuk mengimplementasikan solusi. Karena Sistem manajemen berbasis *web* menawarkan banyak keuntungan, termasuk kemudahan akses karena ketika ingin mengaksesnya tidak perlu mengunduh aplikasi, peningkatan efisiensi kerja, dan integrasi berbagai fitur dalam satu sistem. Teknologi seperti *PHP*, *MySQL*, dan *framework CodeIgniter* berperan penting dalam membangun sistem yang aman dan berkinerja tinggi (Amirullah & Al Islami, 2022).

## 2.9 Basis Data

Basis data adalah sekumpulan data yang terorganisir, tersusun, dan tersimpan secara sistematis di dalam komputer, yang memungkinkan pengguna untuk mencari, mengambil, dan memanipulasi data yang ada didalamnya. Basis data terdiri dari tabel-tabel yang menyimpan data dalam format baris dan kolom, di mana setiap tabel berisi informasi yang berbeda-beda seperti data kendaraan, Jadwal servis, serta riwayat servis. Dengan basis data, perusahaan bisa menyimpan banyak sekali data dengan rapi dan akurat. Data-data ini bisa diakses dengan mudah kapanpun dibutuhkan (Wijaya et al., 2021).

Sistem basis data dikelola melalui perangkat lunak yang disebut Sistem Manajemen Basis Data (*DBMS*), yang berfungsi untuk membantu kita mengelola dan mengoperasikan basis data. *DBMS* membantu mengatur operasi seperti penyimpanan, pengambilan, dan pembaruan data, serta memastikan bahwa data aman dan terlindungi dari akses yang tidak sah. Keberadaan basis data sangat penting dalam berbagai bidang, mulai dari bisnis, pendidikan, kesehatan, hingga pemerintahan, karena basis data dapat membantu mengelola informasi dengan baik. Dengan basis data, keputusan dapat diambil berdasarkan informasi yang akurat dan terintegrasi, sehingga meningkatkan efisiensi dan produktivitas suatu organisasi.

#### **2.10 Laravel**

*Laravel* adalah sebuah alat atau *framework* yang digunakan untuk membuat *website* atau aplikasi berbasis *web* dengan menggunakan bahasa pemrograman *PHP*. *Framework* ini membuat proses pembuatan aplikasi jadi lebih mudah dan cepat karena sudah menyediakan banyak fitur yang berguna. Misalnya, untuk mengelola *database*, mengatur halaman-halaman *website*, hingga membuat pengaturan keamanan (Nurmiati & Al Hafidz, 2021). *Laravel* juga sangat membantu pengembang dengan memberikan sistem yang jelas dan rapi, jadi kode program yang dibuat lebih terstruktur dan mudah dipahami. Salah satu fitur penting dari *Laravel* adalah dapat memudahkan pengembang mengelola dan menyimpan data, serta melindungi aplikasi dari ancaman yang sering terjadi di *internet*, seperti serangan dari *hacker*.

## 2.11 PHP

*PHP* adalah bahasa pemrograman yang digunakan untuk membuat halaman *web* dinamis, yaitu halaman *web* yang bisa berubah-ubah sesuai dengan interaksi pengguna atau data yang ada (Arafat et al., 2022). *PHP* bekerja di *server*, yang artinya dia menangani data dan memproses permintaan dari pengguna. Dengan *PHP*, kita bisa membuat *website* yang bisa mengolah data dari *database*, misalnya untuk *login*, menyimpan data, atau menampilkan informasi yang berubah-ubah. *PHP* sangat sering digunakan bersama dengan *database* seperti *MySQL*, yang berguna untuk menyimpan data yang dibutuhkan oleh *website*. *PHP* juga sangat fleksibel dan banyak digunakan karena kemudahannya dalam mengelola *website* yang lebih kompleks.

## 2.12 Javascript

*JavaScript* adalah bahasa pemrograman yang digunakan untuk membuat situs *web* lebih interaktif (Juliarto Rendi, 2020). Dengan *JavaScript*, halaman *web* bisa merespons tindakan pengguna seperti klik, mengetik, atau menggeser layar, sehingga membuat situs terasa lebih hidup dan dinamis. *JavaScript* bekerja langsung di dalam *browser*, memungkinkan perubahan elemen di halaman tanpa perlu memuat ulang seluruh halaman. Selain itu, *JavaScript* memungkinkan aplikasi *web* untuk mengambil data dari *server* secara cepat tanpa mengganggu pengalaman pengguna. Dengan menggunakan berbagai alat seperti *framework React* atau *Vue*, pengembang dapat membuat aplikasi *web* yang lebih mudah dibangun dan dikelola. Keuntungan utama *JavaScript* adalah kecepatannya dan kemampuannya untuk digunakan baik di sisi pengguna (*frontend*) maupun di sisi *server* (*backend*). Karena didukung oleh hampir

semua *browser*, *JavaScript* menjadi bahasa yang sangat penting dalam pengembangan *web*

### 2.13 Metodologi *Waterfall*

Dalam penelitian ini, metodologi *Waterfall* dipilih sebagai pendekatan pengembangan sistem manajemen kendaraan operasional berbasis *web*. *Waterfall* merupakan salah satu metodologi pengembangan perangkat lunak yang paling umum digunakan, terutama untuk proyek yang memiliki kebutuhan yang sudah jelas dan tahapan yang jelas. Metodologi ini memiliki pendekatan yang linear dan berurutan, di mana setiap fase harus diselesaikan sepenuhnya sebelum fase berikutnya dimulai. Metodologi ini cocok untuk penelitian dengan jangka waktu yang relatif terstruktur dan tidak memerlukan banyak perubahan selama proses pengembangan sistem. (Anis et al., 2023) Tahapan-Tahapan dalam menggunakan *Waterfall Model* menurut (Meilinaeka, 2023) yaitu sebagai berikut :

#### 1. Analisis :

Analisis merupakan tahap pertama dalam *Waterfall Model*, Tahap ini dimulai dengan memahami tujuan dan kebutuhan dari perangkat lunak yang akan dikembangkan. Tim pengembang akan meneliti kebutuhan pengguna serta menentukan fitur dan fungsi yang dibutuhkan

#### 2. Desain :

Setelah memahami kebutuhan, selanjutnya tim yang menggunakan Metode *Waterfall* merancang arsitektur, desain, dan spesifikasi teknis *software*. Perancangan juga melibatkan pembuatan *diagram* alir dan desain antarmuka pengguna.

### 3. Implementasi :

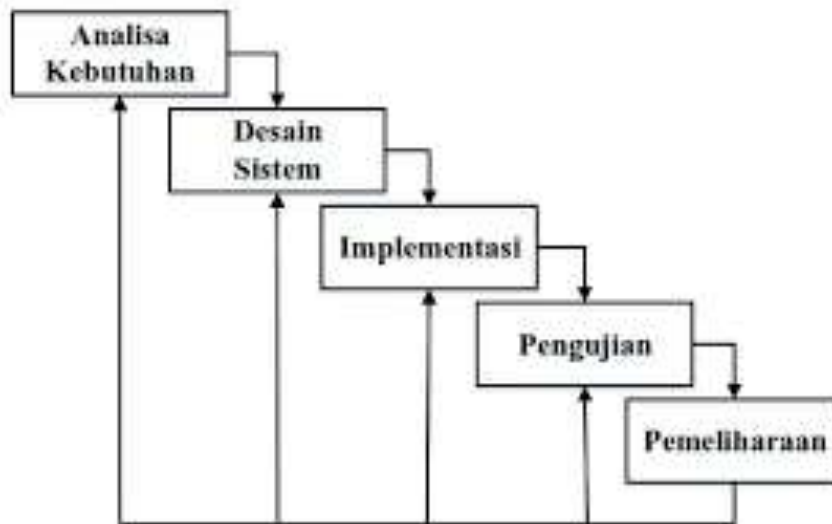
Pada tahap ini sistem yang sudah di desain dan dirancang akan dikembangkan secara lebih nyata, Proses pengembangan ini biasanya melibatkan penulisan kode program untuk menghasilkan perangkat lunak.

### 4. Integrasi dan Pengujian :

Setelah semua bagian sistem selesai dibuat kita akan menggabungkan semua bagian tersebut menjadi satu sistem utuh. Kemudian, kita akan menguji sistem yang sudah utuh ini secara menyeluruh untuk memastikan bahwa sistem tersebut bekerja dengan benar dan sesuai dengan yang di inginkan oleh pengguna.

### 5. Pemeliharaan :

Tahap akhir dalam metodologi *Waterfall* adalah fase pemeliharaan. Setelah sistem berhasil dikembangkan, diuji, dan diterapkan dalam lingkungan perusahaan, sistem akan dioperasikan dan *dimonitor* secara berkelanjutan. Apabila terdapat kesalahan atau bug yang teridentifikasi, tindakan pemeliharaan akan dilakukan untuk memperbaiki masalah tersebut dan memastikan kelancaran operasional sistem.



Gambar 2. 1 *Waterfall Model*

Sumber : (Gusti Made Adi Putra et al., 2020)

Garis panah balik dalam gambar menunjukkan bahwa meskipun alur kerja bersifat linier, setiap fase dapat kembali ke fase sebelumnya jika ada masalah atau perubahan. Ini memberikan fleksibilitas untuk memperbaiki kesalahan atau menyesuaikan desain selama proses pengembangan.



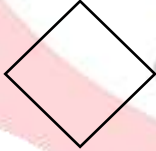

## 2.14 Perancangan *UML*

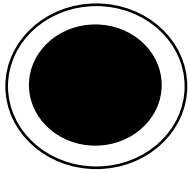
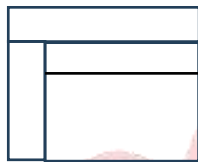
Teori *Unified Modeling Language (UML)* adalah bahasa pemodelan visual standar yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan sistem perangkat lunak berorientasi objek (Yakub et al., 2025). *UML* menyediakan berbagai jenis *diagram* untuk merepresentasikan berbagai aspek sistem, dengan menggunakan *UML* dapat merancang sistem servis dan *reminder* kendaraan operasional dengan lebih terstruktur dan mudah dipahami oleh berbagai pihak yang terlibat dalam pengembangan sistem

### A. Activity Diagram

*Activity Diagram* adalah salah satu jenis *diagram* dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan alur kerja atau aktivitas dalam sebuah sistem. Dalam konteks sistem servis dan *reminder* kendaraan operasional yang sedang dikembangkan, *Activity Diagram* dapat digunakan untuk merepresentasikan proses-proses kunci seperti servis, serta *reminder* pemeliharaan kendaraan. *Diagram* ini menggambarkan langkah-langkah dari awal hingga akhir suatu proses, termasuk titik-titik keputusan, percabangan. Komponen utama dalam *Activity Diagram* meliputi start node, action node, decision node, join node, serta end node.

Tabel 2. 1 Simbol-Simbol *Activity Diagram*

No	Simbol	Deskripsi
1	Start (Status Awal) 	Simbol ini menunjukkan titik awal dari suatu proses atau aktivitas, setiap <i>Activity Diagram</i> pasti dimulai dari status awal ini
2	Aktivitas 	Pada simbol ini biasanya mewakili suatu Tindakan atau langkah yang dilakukan oleh sistem atau pengguna sistem, Biasanya diawali dengan kata kerja
3	Percabangan / Decision 	Simbol ini melambangkan sistem atau pengguna harus mengambil keputusan sehingga menimbulkan percabangan aktivitas dari keputusan tersebut
4	Penggabungan / Join 	Simbol ini melambangkan penggabungan dari beberapa aktivitas yang sebelumnya terpisah. Setelah aktivitas-aktivitas tersebut selesai, alur kembali menjadi satu.

5	End (Status Akhir) 	Simbol ini Menunjukkan akhir dari alur atau proses dalam <i>diagram</i> aktivitas. Ini adalah titik di mana semua aktivitas berakhir.
6	Swimlane 	Simbol ini digunakan untuk memisahkan aktivitas-aktivitas berdasarkan aktor atau unit organisasi yang bertanggung jawab.

Sumber : (Yakub et al., 2025)

### B. Use Case Diagram

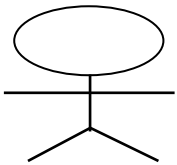
*Use Case Diagram* adalah salah satu jenis *diagram* dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem (Arianti et al., 2022). Dalam konteks sistem servis dan *reminder* kendaraan operasional yang sedang dikembangkan, *Use Case Diagram* akan memvisualisasikan berbagai fungsi yang dapat dilakukan oleh pengguna sistem, seperti *driver*, staf *maintenance*, dan manajemen. *Diagram* ini terdiri dari beberapa elemen utama: aktor (yang merepresentasikan pengguna atau sistem eksternal), *use case* (yang menggambarkan fungsi atau layanan yang disediakan oleh sistem), dan hubungan antara aktor dan *use case*. *Use Case Diagram* memberikan gambaran tingkat tinggi tentang fungsionalitas sistem dan bagaimana berbagai aktor berinteraksi dengannya.

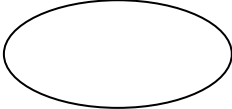


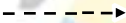
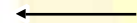
*Use case diagram* memiliki beberapa komponen utama, yaitu aktor, *use case*, dan hubungan antara keduanya. Aktor digambarkan dengan

simbol orang yang mewakili pengguna eksternal atau sistem lain yang berinteraksi dengan sistem utama. Sedangkan, *use case* digambarkan dengan elips dan berisi deskripsi dari fungsionalitas atau layanan yang disediakan oleh sistem. Hubungan antara aktor dan *use case* biasanya direpresentasikan dengan garis sederhana yang menunjukkan interaksi atau keterlibatan aktor dalam sebuah *use case*. Selain itu, *use case* juga dapat memiliki hubungan extend atau include untuk menggambarkan variasi atau ketergantungan antar fungsionalitas.

*Use case diagram* sangat bermanfaat dalam proses pengembangan sistem karena dapat membantu tim pengembang untuk memahami apa yang diinginkan oleh pengguna dan bagaimana sistem harus merespons setiap skenario. *Diagram* ini tidak hanya digunakan pada tahap analisis dan desain, tetapi juga dapat berfungsi sebagai alat komunikasi antara pengembang, manajer proyek, dan pemangku kepentingan. Dengan visualisasi yang sederhana namun informatif, *use case diagram* dapat memastikan bahwa semua pihak memiliki pemahaman yang sama mengenai fungsionalitas sistem yang diharapkan

Tabel 2. 2 Simbol-Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1	Actor 	Aktor dalam <i>Use Case diagram</i> merupakan representasi dari seseorang, sistem lain, atau perangkat yang berinteraksi dengan sistem. Aktor bisa berupa pengguna, perangkat keras, atau sistem eksternal yang menggunakan <i>use case</i> .

2	<i>Use Case</i> 	<i>Use Case</i> merupakan interaksi antara actor dengan sistem
3	<i>Association</i> 	Menggambarkan hubungan antara actor dan <i>use case</i> yang terlibat
4	Generalisasi/ <i>Generalization</i> 	Melambangkan peran khusus dari actor yang terlibat dalam <i>use case</i>
5	Include 	Simbol ini menjelaskan bahwa satu <i>use case</i> mengandung keseluruhan fungsi dari <i>use case</i> lain. Dengan kata lain, <i>use case</i> yang mengarah ke panah akan selalu menjalankan <i>use case</i> yang dilabeli dengan include.
6	Extend 	Melambangkan bahwa satu <i>use case</i> bisa memiliki tambahan fungsionalitas jika kondisi tertentu terpenuhi. Ini adalah bentuk opsional yang menambahkan perilaku pada <i>use case</i> utama.

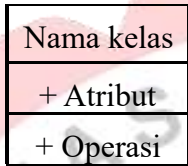




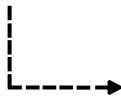
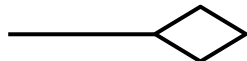
Sumber : (Yakub et al., 2025)

### C. Class Diagram

Menurut (Fachriza Pulungan & Purwanto, 2023). *Class Diagram* adalah sebuah *diagram* struktur statis dalam *UML* yang menggambarkan susunan sistem dengan menampilkan kelas-kelas dalam sistem, atribut, metode, serta hubungan antar objek. *Diagram* ini disebut *diagram* struktur karena menunjukkan elemen-elemen yang harus ada dalam sistem yang sedang dimodelkan melalui berbagai komponen. Komponen-komponen tersebut dapat menggambarkan kelas yang akan diprogram, objek penting, atau

interaksi antara kelas dan objek. Kelas sendiri merupakan istilah yang digunakan untuk menggambarkan sekelompok objek yang memiliki fungsi serupa dalam sistem.

Tabel 2. 3 Simbol-Simbol *Class Diagram*

NO	Simbol	Deskripsi
1	Kelas 	Simbol ini merepresentasikan sebuah kelas dalam sistem, lengkap dengan atribut dan operasinya.
2	Antarmuka / <i>Interface</i> 	Mirip dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	<i>Association</i> 	Menunjukkan hubungan antar kelas. Hubungan ini terjadi Ketika ada lebih dari satu pilihan aktivitas
4	Asosiasi Berarah / <i>Directed Association</i> 	Menunjukkan hubungan antar kelas yang satu arah, biasanya berarti satu kelas menggunakan atau bergantung pada kelas lain. Biasanya disertai dengan <i>multiplicity</i>
5	Generalisasi 	Menggambarkan relasi pewarisan antar kelas umum dan kelas khusus yang lebih spesifik
6	Ketergantungan 	Simbol ini menggambarkan adanya ketergantungan antar kelas, di mana Ketika terjadi perubahan pada satu kelas dapat mempengaruhi kelas lain.
7	Agregasi/ <i>aggregation</i> 	Menggambarkan relasi antar kelas dengan konsep <i>whole-part</i>


Sumber : (Yakub et al., 2025)


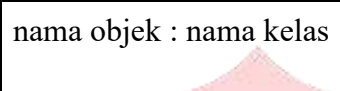

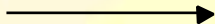


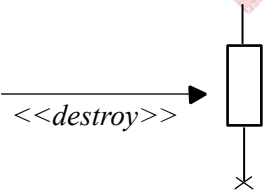
#### D. Sequence Diagram (Diagram Urutan)

Menurut (Setiawan, 2021). *Sequence diagram* atau *diagram* urutan adalah sebuah *diagram* yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu *sequence* diagram juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek-objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan.

Secara umum, *sequence diagram* berguna dalam menggambarkan skenario spesifik atau *use case* yang terjadi dalam sebuah sistem. Dengan menunjukkan bagaimana objek berinteraksi dan bagaimana pesan-pesan dikirimkan secara berurutan, *diagram* ini memberikan gambaran yang jelas tentang aliran data dan peran setiap objek dalam sistem. Hal ini membantu dalam mendesain sistem yang kompleks, memastikan bahwa interaksi antarobjek terdefinisi dengan baik dan berjalan sesuai urutan yang diinginkan

Tabel 2. 4 Simbol-Simbol *Sequence Diagram*

No	Simbol	Deskripsi
1	<p>Aktor </p> <p>Atau Tanpa waktu aktif</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>Nama Aktor</p> <hr style="width: 50%; margin: 5px auto;"/> </div>	<p>Representasi dari orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi dan memperoleh manfaat dari sistem tersebut. Biasanya aktor ditempatkan di bagian atas <i>diagram</i>.</p>

2	Garis Hidup / <i>lifeline</i> 	Menyatakan aktivitas dan siklus hidup suatu objek
3	Objek 	Menggambarkan objek yang sedang berinteraksi atau melakukan kegiatan dengan sistem
4	Waktu aktif 	Simbol ini menunjukkan periode waktu ketika objek sedang beroperasi atau melakukan sesuatu dalam sistem. Ini adalah fase ketika objek aktif dan berinteraksi dengan objek lain.
5	Pesan tipe <i>create</i>  Pesan tipe <i>call</i>	Simbol ini melambangkan untuk membuat atau menghasilkan objek baru. Pesan ini seperti perintah untuk menciptakan sesuatu yang belum ada sebelumnya.
6	Pesan tipe <i>send</i>  1:masukan	Simbol ini menggambarkan bahwa pemanggilan dan pengiriman data kepada objek lain
7	Pesan tipe <i>return</i>  1 : keluaran	Menggambarkan bahwa suatu objek yang telah melakukan operasi atau metode menghasilkan <i>output</i> yang diterima oleh objek tertentu, dengan panah yang mengarah ke objek penerima
8	 << <i>destroy</i> >>	Simbol ini menggambarkan bahwa suatu objek mengakhiri siklus hidup objek lainnya, dengan panah yang mengarah ke objek yang diakhiri; idealnya, jika terdapat pembuatan objek ( <i>create</i> ), harus ada penghancuran objek ( <i>destroy</i> )

Sumber: (Yakub et al., 2025)

## 2.15 Tinjauan Studi

Perkembangan teknologi informasi telah mendorong transformasi digital dalam berbagai sektor industri, termasuk dalam pengelolaan kendaraan operasional dan sistem *booking* servis. *Evolusi* ini dimulai dari sistem manual yang rentan kesalahan menuju sistem otomatis yang terintegrasi, memberikan dampak signifikan terhadap efisiensi dan efektivitas operasional perusahaan.

Dalam konteks pengembangan sistem *booking* servis, (Anis & Sendi Rifa, 2023). menjadi pionir dengan mengembangkan sistem informasi *e-booking* untuk jasa steam mobil dan motor berbasis *web* menggunakan metodologi *Waterfall*. Penelitian ini membuktikan bahwa implementasi sistem *booking online* tidak hanya meningkatkan efektifitas proses pemesanan layanan, tetapi juga mengoptimalkan pengelolaan data pelanggan di Cheers Autocare, Solo. Hasil penelitian menunjukkan peningkatan signifikan dalam efektivitas operasional, terutama pada jam sibuk ketika volume transaksi tinggi.

Berkembang dari temuan tersebut, (Diantara et al., 2022) memperluas konsep sistem *booking* dengan menggunakan pendekatan *Software Development Life Cycle* (SDLC) dalam perancangan aplikasi *booking service online* berbasis *web* untuk bengkel. Penelitian ini tidak hanya memvalidasi efektivitas sistem *booking*, tetapi juga mendemonstrasikan bagaimana teknologi *web* dapat mengurangi ketergantungan pada reservasi telepon dan mempermudah pengelolaan antrian pelanggan. Kontribusi penting dari penelitian ini adalah penggunaan kuesioner dalam pengumpulan data untuk memastikan kebutuhan pengguna terpenuhi secara optimal.

Momentum pengembangan sistem *booking* kemudian diperkuat oleh (Akhruddin M .Annafi, 2024) yang mengimplementasikan aplikasi *booking service*

berbasis *web* pada Bengkel Caraka Auto Service Kediri menggunakan metode SDLC. Penelitian ini memberikan *validasi* tambahan melalui pengujian *Black Box Testing*, membuktikan bahwa sistem *booking* berbasis *web* dapat meningkatkan efisiensi layanan pemesanan dan kepuasan pelanggan secara terukur. Meskipun demikian, penelitian ini juga mengidentifikasi kebutuhan pengembangan lebih lanjut, khususnya dalam hal integrasi transaksi dan peningkatan antarmuka pengguna.

Seiring dengan perkembangan teknologi *mobile*, paradigma sistem *booking* mulai bergeser dari *platform web* tradisional menuju aplikasi *mobile* yang lebih *accessible*. (Reddy & Savant, 2022) mengembangkan Car Service Slot Booking System berbasis *Android* yang memungkinkan pengguna memesan *slot* layanan perawatan kendaraan dengan notifikasi terintegrasi. Inovasi ini menunjukkan *evolusi* natural dari sistem *booking* statis menuju *platform* yang lebih interaktif dan *real-time*, dengan kemampuan menyimpan *detail* pemesanan dalam *database* untuk akses administrator.

Tren *mobile application* kemudian diperkuat oleh (Farizqi et al., 2024) yang merancang aplikasi pemesanan layanan servis sepeda motor berbasis *Android* dengan fokus utama pada peningkatan efisiensi waktu dan pengurangan waktu antri pelanggan. Penelitian ini berhasil mencapai otomatisasi penuh proses *booking* servis motor, meningkatkan kenyamanan pengguna, dan memungkinkan sinkronisasi data secara *real-time*. Temuan ini mengkonfirmasi bahwa *platform mobile* bukan hanya melengkapi sistem *web*, tetapi memberikan keunggulan kompetitif dalam hal aksesibilitas dan *user experience*.

*Evolusi* sistem *booking* mencapai tingkat kematangan yang lebih tinggi dalam penelitian (Selvia Lauryn et al., 2024) yang mengembangkan e-reservasi servis perawatan mobil berbasis *web* pada PT. Indomobil Trada Nasional Cabang Gading

Serpong. Penelitian ini menunjukkan integrasi keseluruhan antara *booking*, konsultasi *sparepart*, dan peningkatan interaksi *customer-service* secara *online*. Pendekatan komprehensif ini menandai transisi dari sistem *booking* sederhana menuju *platform* layanan terintegrasi yang lebih *sophisticated*.

Pergeseran paradigma selanjutnya terjadi ketika fokus penelitian mulai beralih dari *booking* layanan eksternal menuju manajemen internal kendaraan operasional perusahaan. (Pratiwi & Iqbal, 2023) memulai transisi ini dengan mengembangkan aplikasi pendataan kendaraan operasional menggunakan metode *prototype* di PT. HM Sampoerna Tbk, Medan. Penelitian ini menandai *evolusi* penting dari sistem *booking* reaktif menuju manajemen proaktif, dimana sistem tidak hanya memfasilitasi pemesanan layanan, tetapi juga menyediakan informasi riwayat perbaikan komprehensif dan *reminder* perawatan berkala untuk optimalisasi kinerja kendaraan operasional.

Konsep manajemen kendaraan operasional kemudian diperluas oleh (Wahyu Kurniawan & Yasin Irsyadi, 2021). yang mengembangkan aplikasi manajemen peminjaman kendaraan berbasis *web* dengan *framework CodeIgniter* di PT Surveyor Indonesia. Menggunakan metodologi *Research and Development (R&D)*, penelitian ini menunjukkan bagaimana sistem manajemen kendaraan dapat meningkatkan efisiensi operasional melalui pencatatan yang lebih akurat dan proses peminjaman yang terstruktur. Kontribusi signifikan penelitian ini adalah demonstrasi integrasi antara fungsi *booking* dan manajemen aset dalam satu *platform* terpadu.

Pada skala *enterprise* yang lebih luas, (Nsama et al., 2024) mengembangkan sistem manajemen kendaraan berbasis *web* (VMS) yang dirancang untuk mengoptimalkan distribusi armada pengiriman dan mengurangi pengeluaran sumber

daya transportasi. Penelitian ini menunjukkan maturitas konsep manajemen kendaraan dari level individual menuju manajemen *fleet comprehensive*, dengan fokus khusus pada efisiensi bahan bakar dan optimalisasi resource allocation. Hasil penelitian membuktikan bahwa sistem berbasis *web* dapat secara signifikan meningkatkan efisiensi pengelolaan kendaraan dan mengurangi masalah operasional terkait penggunaan bahan bakar.

Paralel dengan pengembangan sistem manajemen kendaraan, aspek *reminder* dan notifikasi otomatis muncul sebagai komponen krusial dalam ekosistem manajemen kendaraan modern. (Prasetyo Yudi & Desyani Teti, 2023) mengintegrasikan teknologi *WhatsApp Gateway* dalam aplikasi *reminder service* kendaraan berbasis *website* untuk PT. Telkom Akses Jakarta Utara. Penelitian ini mendemonstrasikan bagaimana integrasi teknologi komunikasi dapat meningkatkan efisiensi penggunaan kendaraan dan memastikan kondisi kendaraan selalu optimal, sambil mempermudah administrator dalam mengelola data kendaraan dan scheduling pemeliharaan.

Pengembangan sistem *reminder* kemudian dieksplorasi dari perspektif personal *user* oleh (Kehat Driesa & Somya, 2023) yang merancang aplikasi *service reminder* sepeda motor berbasis *Android mobile*. Penelitian ini berhasil mempermudah pengendara dalam mengingat jadwal perawatan kendaraan pribadi dan memberikan informasi *real-time* mengenai kondisi *sparepart* yang memerlukan penggantian atau perawatan. Kontribusi penelitian ini penting karena menunjukkan aplikabilitas sistem *reminder* tidak hanya untuk fleet management perusahaan, tetapi juga untuk kebutuhan individual, memperluas spektrum implementasi teknologi *reminder*.

Seiring dengan semakin kompleksnya sistem yang dikembangkan, aspek *user experience* dan metodologi desain menjadi fokus penelitian yang krusial. (Thaariq et

al., 2022) mengimplementasikan pendekatan Design Thinking dalam perancangan *user experience* aplikasi *mobile* MECHANIC di CV. Cirebon Niaga Sejahtera. Metodologi *user-centered design* ini menghasilkan aplikasi komprehensif dengan fitur canggih termasuk pencarian bengkel berdasarkan lokasi, jenis kerusakan, dan rating, sistem pemesanan dan reservasi terintegrasi, serta sistem pembayaran yang *seamless*. Penelitian ini menunjukkan pentingnya pendekatan *holistik* yang tidak hanya mempertimbangkan fungsionalitas sistem, tetapi juga aspek psikologis dan behavioral pengguna.

Dalam konteks metodologi pengembangan, (Sita Eriana et al., 2022) mengeksplorasi alternatif pendekatan melalui implementasi metode *Rapid Application Development* (RAD) dalam perancangan sistem informasi *booking service* kendaraan di Honda Megatama Kapuk. Penelitian ini memberikan perspektif berbeda dalam hal kecepatan *development*, meskipun hasil menunjukkan bahwa pendekatan *rapid development* masih memerlukan iterasi tambahan untuk memenuhi keseluruhan kebutuhan pengguna. Temuan ini memberikan insight berharga mengenai trade-off antara kecepatan pengembangan dan completeness fungsionalitas sistem.

Pada frontier teknologi terdepan, (Bhargava et al., 2022) mengeksplorasi integrasi *Industrial IoT* dan *Artificial Intelligence* dalam logistik kendaraan dan manajemen supply chain untuk sistem transportasi. Penelitian ini mengembangkan model IoT terintegrasi dengan *smart logistics* untuk mengoptimalkan operasi, meningkatkan *customer experience*, dan meminimalkan biaya transportasi. Hasil penelitian menunjukkan peningkatan *performance* dari 77% menjadi 98%, mengindikasikan bahwa integrasi teknologi advanced dapat memberikan quantum leap dalam efisiensi operasional. Penelitian ini memberikan vision mengenai masa depan sistem manajemen kendaraan yang *fully automated* dan *AI-driven*.

*Validasi* terhadap dampak strategis sistem manajemen kendaraan diberikan oleh (Aflabo & Kraa, 2020), yang menganalisis pengaruh praktik *fleet management* terhadap *competitive advantage* di industri transportasi Ghana. Melalui penelitian kuantitatif dengan 200 responden, studi ini membuktikan bahwa manajemen perbaikan dan pemeliharaan, *fuel management*, *vehicle tracking*, serta *driver management* dan training secara signifikan mempengaruhi keunggulan kompetitif perusahaan. Temuan bahwa *driver management* dan training memberikan impact terbesar terhadap *competitive advantage* memberikan insight penting bahwa aspek human factor tetap krusial meskipun teknologi terus berkembang.

Berdasarkan analisis menyeluruh terhadap lima belas penelitian sebelumnya, ditemukan perkembangan yang jelas dalam sistem manajemen kendaraan dari *booking* servis sederhana menjadi sistem yang lebih terpadu dan canggih. Namun, sebagian besar penelitian masih fokus pada bagian terpisah seperti *booking service*, manajemen kendaraan operasional, atau sistem *reminder* saja. Terdapat kesenjangan dalam kebutuhan sistem terpadu yang menggabungkan ketiga aspek penting tersebut dalam satu *platform*, khususnya untuk kendaraan operasional perusahaan manufaktur yang memiliki kebutuhan berbeda dengan sektor *automotive service* atau transportasi umum. Penelitian ini berkontribusi mengisi kesenjangan tersebut dengan mengembangkan sistem yang menggabungkan *booking* servis, pengingat otomatis, dan pengelolaan kendaraan yang lengkap dalam satu *platform* berbasis *web* menggunakan metode *Waterfall* yang telah terbukti efektif. Sistem ini memberikan solusi khusus untuk mengoptimalkan manajemen kendaraan operasional di lingkungan manufaktur yang membutuhkan keandalan, efisiensi, dan pemeliharaan pencegahan yang sistematis untuk mendukung keunggulan operasional.

## **BAB III METODOLOGI PENELITIAN**

### **3.1 Gambaran Umum Perusahaan**

#### **3.1.1 Sejarah Perusahaan**

PT Prima Indah Lestari beroperasi sejak tahun 1995 , yang juga dikenal dengan nama *brand* produknya "Extrana Cable," adalah perusahaan yang berlokasi di Jakarta Barat, Indonesia. Perusahaan ini bergerak di bidang manufaktur kabel, khususnya kabel tegangan rendah dan tinggi, dan dikenal karena kualitas tinggi serta teknologi canggih dalam produk-produknya. Extrana Cable berpartisipasi dalam berbagai proyek besar, termasuk proyek-proyek pemerintah seperti pembangunan Stadion JIS. PT Prima Indah Lestari berfokus pada penyediaan solusi kabel yang handal dan inovatif untuk memenuhi kebutuhan industri dan konstruksi. Dengan reputasi yang solid di pasar, perusahaan ini berkomitmen untuk meningkatkan efisiensi operasional dan produktivitas melalui berbagai inisiatif teknologi dan manajerial, termasuk pengelolaan kendaraan operasional yang optimal.

#### **3.1.2 Visi dan Misi**

Visi dan misi PT Prima Indah Lestari berfokus pada pengembangan produk berkualitas tinggi dalam industri kabel listrik. Produk perusahaan ini, yang dikenal dengan merek *Extrana Cable*, bertujuan menjadi patokan dalam kualitas dan inovasi produk kabel di Indonesia. Mereka berkomitmen untuk meningkatkan standar produk dan layanan secara berkelanjutan.

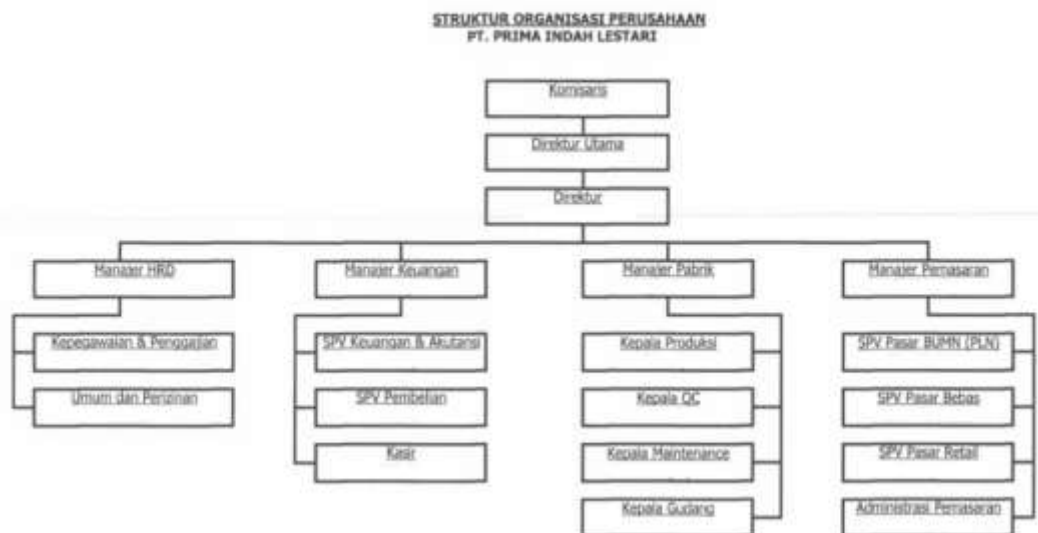
## 1. VISI

Menjadi tolok ukur dalam hal kualitas dan inovasi produk kabel listrik, serta menjaga keberlanjutan dengan memenuhi standar nasional dan kebutuhan pelanggan.

## 2. Misi

1. Menyediakan produk yang berkualitas tinggi yang memenuhi standar nasional dan internasional, termasuk Standar Nasional Indonesia (SNI) dan Standar Perusahaan Listrik Negara (PLN).
2. Menyediakan layanan pelanggan yang unggul dengan solusi yang tepat sesuai kebutuhan pelanggan

### 3.1.3 Struktur Organisasi



Gambar 3. 1 Struktur Organisasi

(Sumber: PT Prima Indah Lestari)

### 3.1.4 Tugas & Wewenang

Berikut merupakan penjelasan Tugas dan Wewenang dari masing masing posisi yang ada di struktur organisasi diatas :

1. Komisaris:

- a. Tugas : Mengawasi kinerja perusahaan secara keseluruhan, memberikan arahan strategis, serta meninjau keputusan penting.
- b. Wewenang : Mengambil keputusan strategis, mengevaluasi kinerja direksi, serta memberikan persetujuan pada kebijakan besar.

2. Direktur Utama

- a. Tugas : Mengatur dan mengawasi seluruh operasional perusahaan, menetapkan visi dan misi, serta menjaga hubungan dengan pihak eksternal.
- b. Wewenang: Membuat keputusan eksekutif, menyetujui rencana strategis perusahaan, dan mewakili perusahaan dalam negosiasi penting.

3. Direktur

- a. Tugas: Mendukung direktur utama dalam menjalankan operasional harian dan memastikan kebijakan perusahaan diterapkan di setiap divisi.
- b. Wewenang: Mengawasi para manajer, mengambil keputusan terkait operasional, dan melapor kepada direktur utama.

4. Manajer HRD

- a. Tugas: Mengelola sumber daya manusia, rekrutmen, pelatihan, pengembangan karyawan, serta manajemen kinerja.

- b. Wewenang: Menyetujui keputusan terkait karyawan, menyusun kebijakan HR, dan memimpin tim HR.

#### 5. Kepegawaian & Penggajian

- a. Tugas: Mengelola administrasi kepegawaian, penggajian, dan tunjangan karyawan.
- b. Wewenang: Menentukan dan memverifikasi pembayaran gaji serta memberikan masukan terkait kebijakan penggajian.

#### 6. Umum & Perizinan

- a. Tugas: Mengurus kebutuhan umum dan administrasi perizinan perusahaan.
- b. Wewenang: Mengatur kebutuhan operasional umum dan memastikan semua perizinan sesuai dengan peraturan yang berlaku.

#### 7. Manajer Keuangan

- a. Tugas: Mengatur keuangan perusahaan, membuat laporan keuangan, dan menyusun anggaran.
- b. Wewenang: Menyetujui anggaran, mengawasi pengeluaran, dan memberikan rekomendasi finansial kepada direksi.

#### 8. SPV Keuangan & Akuntansi

- a. Tugas: Memastikan laporan keuangan dibuat dengan akurat, dan mengelola pembukuan.
- b. Wewenang: Mengawasi tim akuntansi, memverifikasi laporan keuangan, dan bertanggung jawab atas kepatuhan akuntansi.

9. SPV Pembelian

- a. Tugas: Mengatur pembelian barang dan jasa yang dibutuhkan perusahaan.
- b. Wewenang: Menyetujui pesanan pembelian, memilih vendor, dan mengawasi kinerja pemasok.

10. Kasir

- a. Tugas: Mengelola kas perusahaan, menerima pembayaran, dan mencatat transaksi keuangan.
- b. Wewenang: Mengelola uang tunai dan mencatat pengeluaran serta penerimaan kas.

11. Manajer Pabrik

- a. Tugas: Mengawasi seluruh operasional produksi, termasuk proses produksi, kualitas, dan pemeliharaan.
- b. Wewenang: Menyetujui prosedur produksi, mengawasi tenaga kerja di pabrik, dan memastikan target produksi terpenuhi.

12. Kepala Produksi

- a. Tugas: Mengatur jalannya proses produksi agar sesuai dengan standar kualitas dan efisiensi.
- b. Wewenang: Mengelola tenaga kerja di lini produksi dan memantau jadwal produksi.

13. Kepala QC (*Quality Control*)

- a. Tugas: Memastikan produk yang dihasilkan memenuhi standar kualitas perusahaan.
- b. Wewenang: Mengambil keputusan mengenai produk sebelum distribusi dan melakukan pengawasan kualitas.

14. Kepala *Maintenance*

- a. Tugas: Mengelola pemeliharaan asset perusahaan dan memastikan peralatan bekerja dengan baik.
- b. Wewenang: Mengatur jadwal pemeliharaan dan memperbaiki aset yang rusak.

15. Kepala Gudang

- a. Tugas: Mengatur penyimpanan bahan baku dan barang jadi di gudang.
- b. Wewenang: Mengontrol stok dan mengawasi kegiatan logistik di gudang.

16. Manajer Pemasaran

- a. Tugas: Mengembangkan strategi pemasaran, promosi, dan penjualan produk.
- b. Wewenang: Menyusun rencana pemasaran, mengelola anggaran pemasaran, dan mengevaluasi efektivitas strategi pemasaran.

17. SPV Pasar BUMN (PLN)

- a. Tugas: Mengelola penjualan dan pemasaran ke pasar BUMN, khususnya PLN.
- b. Wewenang: Mencari peluang bisnis di sektor BUMN dan menjaga hubungan dengan klien.

18. SPV Pasar Bebas

- a. Tugas: Mengelola penjualan ke pasar umum
- b. Wewenang: Menentukan strategi untuk pasar umum dan *memonitor* hasil penjualannya

### 19. SPV Pasar Retail

- a. Tugas: Mengelola penjualan produk di pasar retail.
- b. Wewenang: Mengatur strategi pemasaran di pasar retail dan menjaga hubungan dengan mitra retail.

### 20. Administrasi Penjualan

- a. Tugas: Mengurus administrasi yang berhubungan dengan pemasaran, seperti dokumen penjualan dan laporan.
- b. Wewenang: Mengelola administrasi terkait pemasaran dan memastikan kelancaran dokumentasi.

## 3.2 Teknik Pengumpulan Data

Dalam mendapatkan sebuah data yang diperlukan untuk melakukan penelitian ini ,  
Maka dilakukan pengumpulan data seperti :

### 1. Observasi :

#### A. Tantangan dalam sistem manual

1. Pengelolaan jadwal servis: Jadwal sering terlewat karena tidak adanya pengingat untuk jadwal servis selanjutnya. Hal ini menyebabkan kendaraan sering mengalami kerusakan yang seharusnya dapat dicegah.
2. Pencatatan manual: Data pencatatan manual sulit dilacak, terutama saat mencari riwayat perawatan kendaraan tertentu.
3. Efektivitas rendah: Kalender fisik atau pengingat pribadi yang digunakan sering tidak efektif, sehingga menciptakan risiko kelalaian.

## B. Metode manual yang sedang digunakan

1. Catatan dibuat di buku atau menggunakan *file Excel*.
2. *Driver* melapor secara langsung kepada tim *maintenance* untuk menjadwalkan servis kendaraan.
3. Peningkat dilakukan menggunakan kalender fisik atau metode pribadi tanpa adanya sistem otomatis.

## C. Manfaat yang Diharapkan dari Sistem Berbasis *Web*

1. Pemberitahuan jadwal : Peningkat jadwal servis kendaraan akan mengurangi risiko kelalaian.
2. Kemudahan pencatatan: Sistem akan mempermudah pencatatan dan akses data riwayat perawatan kendaraan.
3. Pengurangan kesalahan: Dengan data yang terpusat, kesalahan akibat proses manual akan diminimalkan.
4. Analisis kebutuhan perawatan: Data yang terpusat akan memungkinkan analisis yang lebih baik untuk kebutuhan perawatan.

## D. Fitur Peningkat Otomatis

1. Fitur ini dianggap sangat membantu dalam memastikan kendaraan diservis tepat waktu.
2. Akan mengurangi risiko kerusakan mendadak.
3. Menjaga kendaraan dalam kondisi optimal, yang akan mendukung kelancaran operasional perusahaan.

## E. Efisiensi & Efektifitas Operasional

1. Sistem akan mengurangi waktu yang dihabiskan untuk pencatatan dan pencarian data manual.
2. Memungkinkan *monitoring* kendaraan secara *real-time*.

3. Perawatan bisa dilakukan lebih proaktif, yang akan menurunkan biaya operasional secara keseluruhan.

## 2. Studi Pustaka :

### A. Sistem Aplikasi *Booking* Servis

Menurut buku *Management Information Systems: Managing the Digital Firm* Hasil karya (Laudon Kenneth & Laudon Jane.p, 2012) menjelaskan bahwa sistem pemesanan berbasis teknologi informasi dapat meningkatkan efisiensi operasional dengan menyediakan antarmuka yang mudah diakses oleh pengguna. Sistem ini memungkinkan pelanggan untuk memesan layanan, mengatur jadwal, serta mendapatkan konfirmasi secara langsung melalui *platform* digital, yang mengurangi kesalahan manusia dan mempercepat proses administrasi.

### B. Sistem Pengingat Dalam Pemeliharaan Kendaraan

Sistem pengingat atau *reminder* sistem telah banyak diterapkan di berbagai bidang untuk memastikan bahwa tugas atau aktivitas tertentu dilakukan pada waktu yang tepat (Elyza, 2023). Dalam konteks pemeliharaan kendaraan, sistem pengingat otomatis untuk jadwal servis adalah alat yang sangat berguna untuk menghindari keterlambatan dalam pemeliharaan kendaraan .

### C. Sistem *Booking* Servis Kendaraan Berbasis *Web*

Menurut (Nurmiati & Al Hafidz, 2021). Sistem pemesanan servis kendaraan berbasis *web* menjadi solusi yang efektif untuk perusahaan yang memiliki armada kendaraan operasional, terutama dalam konteks pengelolaan armada besar. Sistem ini memungkinkan karyawan atau pengguna kendaraan untuk memesan jadwal servis kendaraan secara *online* dengan mudah dan cepat. Implementasi sistem berbasis *web* memiliki berbagai keuntungan, seperti kemudahan akses,

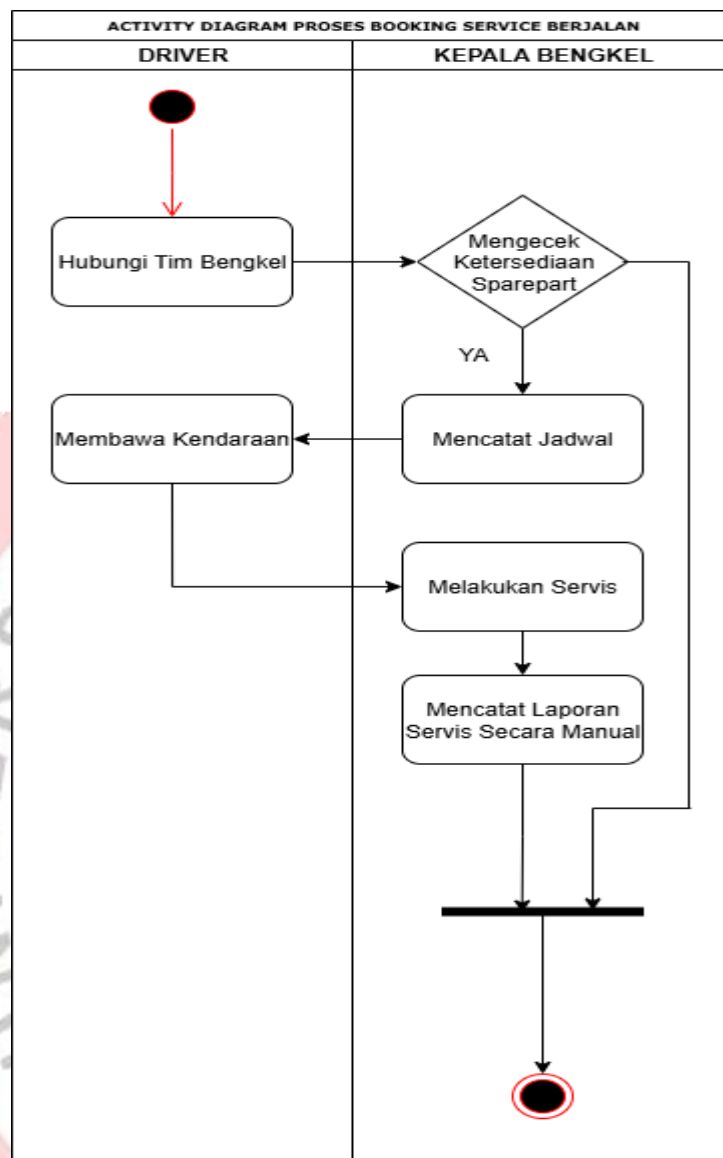
keterjangkauan, dan fleksibilitas, yang sangat diperlukan oleh perusahaan yang mengelola kendaraan operasional secara rutin.

### 3.3 Prosedur Sistem Berjalan

Berikut merupakan prosedur sistem berjalan yang dilakukan secara manual :

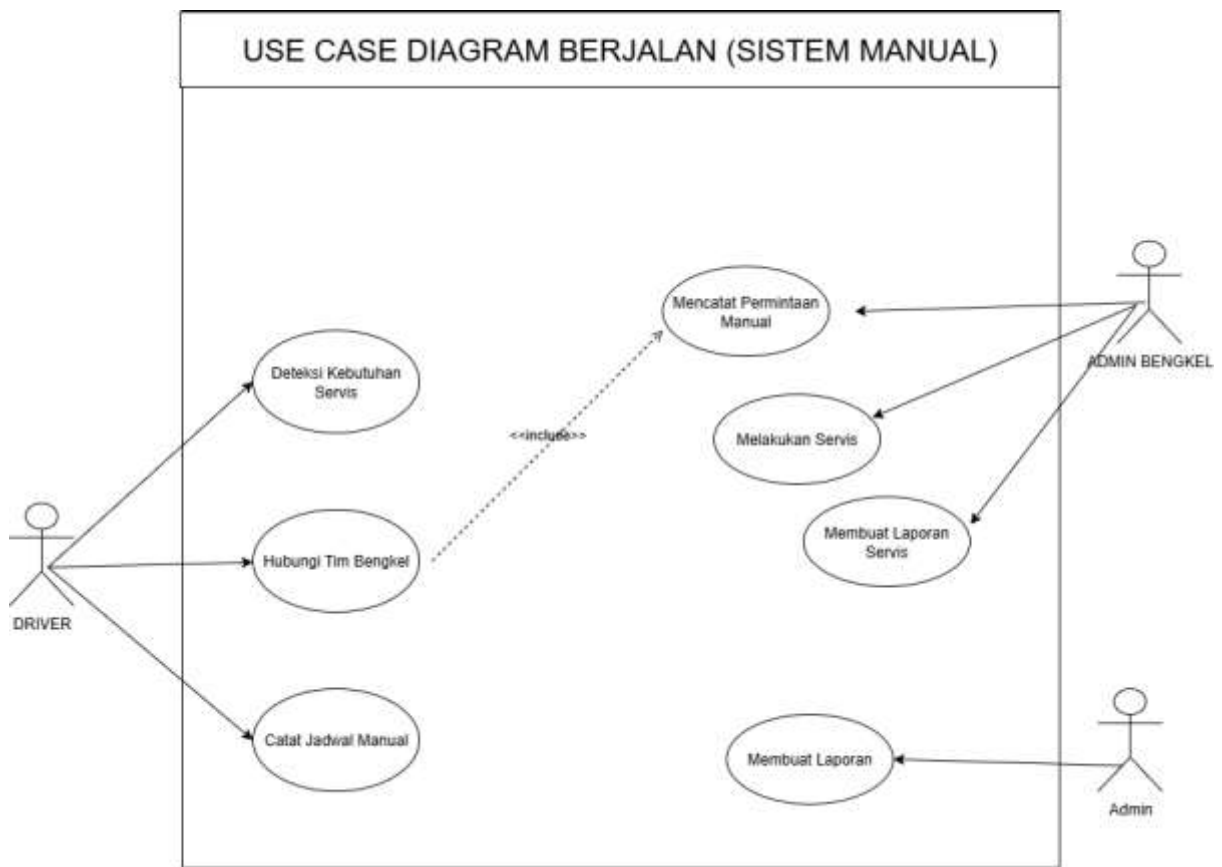
1. *Driver* kendaraan operasional menghubungi kepala bengkel untuk mengajukan permintaan servis kendaraan.
2. Pengecekan ketersediaan *Sparepart*: Kepala bengkel melakukan pengecekan manual terhadap ketersediaan *sparepart* yang dibutuhkan untuk servis kendaraan tersebut.
3. Pengambilan keputusan: Jika *sparepart* tersedia, maka proses dilanjutkan ke tahap berikutnya. Tetapi Jika *sparepart* tidak tersedia, proses terhenti dan perlu menunggu pengadaan *sparepart*
4. Penjadwalan servis : Apabila *sparepart* tersedia, Kepala Bengkel mencatat jadwal servis secara manual dalam buku catatan atau formulir fisik, termasuk mencatat *detail* kendaraan, jenis servis, dan waktu pelaksanaan.
5. Pelaksanaan Servis: *Driver* membawa kendaraan sesuai jadwal yang telah ditentukan, kemudian Kepala Bengkel atau tim bengkel yang ditugaskan melakukan proses servis kendaraan sesuai dengan kebutuhan.
6. Dokumentasi Manual: Setelah servis selesai dilakukan, Kepala Bengkel mencatat laporan servis secara manual dalam buku catatan, meliputi jenis servis yang dilakukan, *sparepart* yang digunakan.
7. Penyelesaian Proses: Proses *booking* servis dinyatakan selesai dan kendaraan dapat digunakan kembali untuk operasional.

### 3.4 Activity Diagram Sistem Booking Servis Berjalan



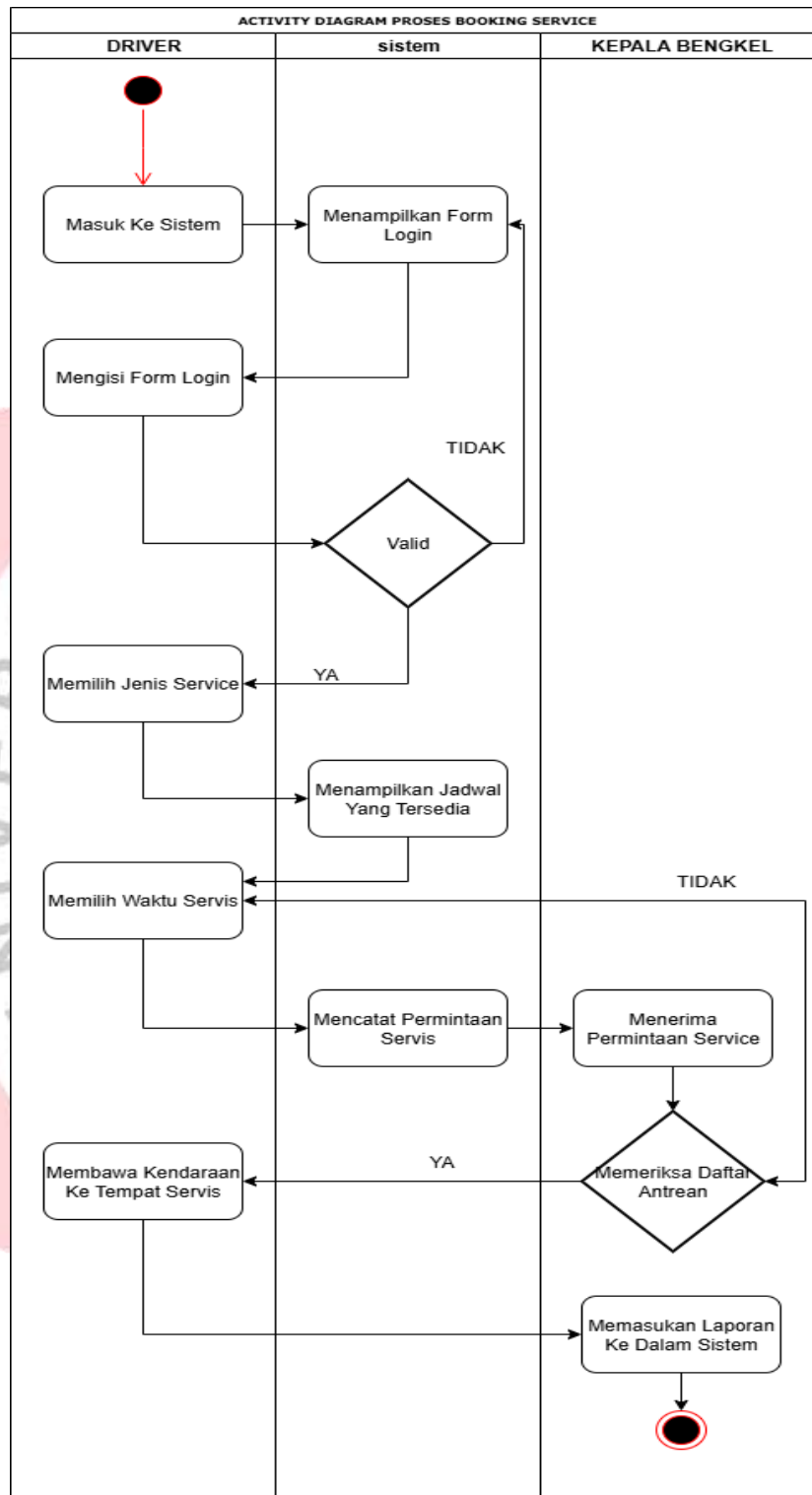
Gambar 3. 2 Activity Diagram Sistem Booking Servis Berjalan

### 3.5 Use Case Diagram Sistem Berjalan



Gambar 3. 3 Use Case Diagram Sistem Berjalan

### 3.6 Activity Diagram Sistem Usulan



Gambar 3. 4 Activity Diagram Usulan Proses Booking Servis

### 3.7 Dokumentasi *Input* Dan *Output*

#### 3.7.1 Dokumentasi *Input*

Dokumentasi *input* mencatat semua data yang diperlukan sistem untuk menjalankan fungsinya sesuai dengan yang dirancang. Oleh karena itu berikut merupakan dokumentasi *input* dan *output* dari perancangan sistem ini :

- a. *Form input* kendaraan : Dokumentasi *input* ini merupakan data mengenai kendaraan operasional yang perlu dimasukkan ke dalam sistem untuk keperluan manajemen.

*Input Fields* :

1. Nomor Polisi Kendaraan: Misalnya, B 9999 ABC.
  2. Nama Mobil :Misalnya avanza, agya, fuso, canter box.
  3. Nomor Mesin : Nomor mesin kendaraan yang pastinya setiap kendaraan memiliki nomor mesin yang berbeda.
  4. Nama *Driver* : Nama *driver* yang mengemudikan mobil tersebut.
- b. *Form input* jadwal servis : *Input* data terkait jadwal servis kendaraan oleh bagian admin.

*Input Fields* :

1. Nomor polisi kendaraan : Untuk identifikasi kendaraan yang dijadwalkan untuk servis..
2. Jenis Servis: Servis berkala, penggantian oli, pengecekan rem.
3. Tanggal Pengajuan Servis: Tanggal pengajuan jadwal servis.
4. Tanggal Servis yang Dijadwalkan: Kapan kendaraan dijadwalkan untuk diservis.

c. *Form input* laporan catatan servis : Untuk pencatatan hasil pelaksanaan servis kendaraan.

*Input Fields* :

1. Nomor polisi kendaraan : Untuk identifikasi kendaraan yang diservis.
2. Tanggal Pelaksanaan Servis: Tanggal ketika kendaraan diservis.
3. Jenis servis yang dilakukan : misal penggantian oli, perbaikan rem.
4. Suku cadang yang diganti: Informasi suku cadang yang diganti.
5. Catatan tambahan : catatan tambahan mengenai kondisi kendaraan.

### **3.7.2 Dokumentasi *Output***

Dokumentasi *output* merupakan sekumpulan informasi yang dihasilkan sistem sebagai respons terhadap *input data*.

- a. Laporan status kendaraan : Laporan yang dihasilkan sistem mengenai kondisi terkini dan kinerja armada kendaraan.

*Output Fields* :

1. Nomor polisi kendaraan : Untuk identifikasi kendaraan yang terdapat dalam laporan.
2. Jarak tempuh (Kilometer) kendaraan saat ini.
3. Laporan perawatan terakhir: Dokumentasi resmi mengenai jenis, tanggal pelaksanaan servis kendaraan, dan komponen kendaraan yang diganti

- b. Laporan pengingat servis : Pengingat otomatis terkait jadwal servis yang akan datang.
  1. Nomor Polisi Kendaraan.
  2. Jenis servis yang dijadwalkan pada servis berikutnya
  3. Tanggal servis yang dijadwalkan pada servis berikutnya.

### 3.8 Analisis Masalah

Analisis masalah dalam sistem manajemen kendaraan operasional menjadi hal yang penting untuk mengidentifikasi hambatan dan upaya-upaya yang dapat dilakukan untuk mengurangi hambatan tersebut. Berikut adalah rincian dari analisis masalahnya:

1. Proses pengelolaan kendaraan yang masih manual : Sistem manajemen kendaraan yang saat ini digunakan masih berbasis manual, yang melibatkan pencatatan manual dan koordinasi langsung antara pihak terkait, seperti *driver*, tim bengkel, dan admin. Hal ini memakan waktu dan rentan terhadap kesalahan manusia. Selain itu, ketiadaan sistem terpusat membuat kesulitan dalam *memonitor* dan mengelola jadwal servis kendaraan. Hal ini memperlambat pengambilan keputusan dan meningkatkan potensi risiko kesalahan dalam perawatan kendaraan.

Tabel 3. 1 Proses Manual Yang Berlangsung

<b>Aktivitas Pengelolaan Kendaraan</b>	<b>Metode yang Digunakan Saat Ini</b>	<b>Dampak yang Ditimbulkan</b>
1. Pencatatan Jadwal Servis	Buku <i>log/Spreadsheet</i>	Kesalahan pencatatan, keterlambatan servis
2. Pencatatan Penggantian Suku Cadang	Buku Manual / <i>Excel</i>	Ketidakkuratan riwayat suku cadang
3. Koordinasi Antar Tim	Secara manual, tidak terpusat	Koordinasi yang lambat, potensi miskomunikasi

2. Tidak ada pengingat otomatis : sistem yang ada pada saat ini tidak memiliki fitur pengingat otomatis untuk jadwal servis kendaraan, pengingat yang bergantung pada metode manual, seperti kalender fisik atau pengingat pribadi, seringkali tidak tepat dan sering terlewatkan. sehingga sering terjadi kelalaian dalam melakukan perawatan yang tepat waktu, yang dapat menimbulkan resiko kerusakan kendaraan.

Tabel 3. 2 Dampak Ketiadaan Pengingat Otomatis

<b>Ketiadaan Fitur Pengingat Otomatis</b>	<b>Dampak yang Terjadi</b>
1. Pengingat manual (kalender fisik)	Sering terlewatkan, mengarah pada keterlambatan servis
2. Ketergantungan pada pengingat pribadi	Tidak sistematis, sering tidak terkoordinasi
3. Tidak ada pengingat terpusat	Tidak ada kontrol terpusat terhadap status pemeliharaan kendaraan

3. Sulitnya melacak riwayat pemeliharaan kendaraan : Riwayat pemeliharaan kendaraan dicatat secara manual, sehingga dapat membuat sulit untuk diakses oleh manajemen maupun tim bengkel. Ini mengakibatkan kesulitan dalam menganalisis data perawatan kendaraan dan menentukan tindakan yang perlu diambil untuk menjaga kendaraan tetap dalam kondisi optimal.

Tabel 3. 3 Dampak Sulitnya Melacak Riwayat Pemeliharaan

Masalah yang Dihadapi	Dampak pada Manajemen Kendaraan
1. Riwayat pemeliharaan tercatat manual	Analisis data yang lambat dan tidak efisien
2. Riwayat pemeliharaan tidak terpusat	Tidak dapat melihat status kendaraan secara menyeluruh
3. Riwayat servis tidak tersusun rapih	Memperburuk pengambilan keputusan dan perencanaan perawatan

4. Kesulitan dalam integrasi data lama : Hal ini merupakan salah satu tantangan utama adalah mengintegrasikan data kendaraan dan pemeliharaan yang sebelumnya dicatat secara manual atau menggunakan *format* yang berbeda, seperti *excel* atau catatan fisik. Proses migrasi data dari sistem lama ke sistem baru dapat memakan waktu dan beresiko menimbulkan kesalahan jika tidak dilakukan dengan hati-hati.

Tabel 3. 4 Tantangan Integrasi Data Lama

Tantangan Integrasi Data Lama	Dampak yang Ditimbulkan
1. Data kendaraan tersebar di berbagai sumber	Sulit menggabungkan dan mengelola data secara terpusat
2. Penggunaan <i>format</i> data yang tidak standar	Risiko kesalahan dalam migrasi dan analisis data
3. Data lama yang tidak terstruktur	Keterlambatan dalam mengambil keputusan berbasis data

### 3.9 Identifikasi Kebutuhan Sistem

Identifikasi kebutuhan sistem adalah proses yang sangat penting, karena untuk mengklarifikasi dan memahami secara mendalam kebutuhan sistem yang diinginkan oleh pengguna. Identifikasi ini bertujuan untuk merancang sistem manajemen kendaraan operasional di PT Prima Indah Lestari yang dapat berfungsi sesuai dengan ekspektasi pengguna. Berikut merupakan identifikasi kebutuhan dari penelitian ini :

1. Kebutuhan Fungsional :

- A. Pencatatan Kendaraan : sistem harus memungkinkan pengguna untuk menyimpan data kendaraan operasional, termasuk informasi seperti tipe kendaraan, nomor plat, tanggal servis, dan status kendaraan.
- B. Jadwal Servis Otomatis: Sistem perlu mengatur dan menampilkan jadwal servis kendaraan secara otomatis berdasarkan periode waktu yang telah ditentukan. Peningkat ini harus dapat disesuaikan dan diatur oleh admin.
- C. Manajemen Pengguna : Setiap pengguna (admin, tim bengkel, *driver*) memiliki akses yang disesuaikan dengan tugasnya (kelola data, catat servis, catat penggunaan kendaraan).

2. Kebutuhan Non-Fungsional :

- A. Keamanan : Untuk menjaga kerahasiaan data kendaraan, riwayat servis, dan informasi pribadi pengguna, sistem harus dilengkapi dengan perlindungan yang kuat. Seperti autentikasi pengguna dan enkripsi data.
- B. Mudah Digunakan : Tampilan sistem yang dibuat harus sederhana dan mudah dipahami oleh semua pengguna, baik itu admin, teknisi, maupun *driver* yang kurang mengerti teknologi.
- C. Skalabilitas : Sistem harus mampu berkembang seiring dengan pertumbuhan bisnis, dengan kapasitas untuk menampung peningkatan jumlah kendaraan dan pengguna tanpa menghambat kinerja.

3. Kebutuhan Teknologi :

- A. *Platform Web* : Sistem akan dibuat dengan berbasis *web*, sehingga bisa diakses menggunakan perangkat apa saja dan dimana saja, hanya memerlukan *internet* dan *browser*
- B. *Database* : Untuk menyimpan dan mengelola data secara baik dan terstruktur, sistem ini akan menggunakan *database. MySQL* adalah salah satu pilihan yang memungkinkan.

4. Kebutuhan Pelatihan :

- A. Pelatihan Pengguna : Penyediaan pelatihan untuk pengguna, terutama teknisi dan pengemudi, agar dapat menggunakan sistem secara maksimal.
- B. Dokumentasi Sistem: Pengguna membutuhkan panduan penggunaan sistem yang jelas, baik dalam bentuk dokumen atau video tutorial, untuk memahami cara menggunakan sistem dengan benar.

### 3.10 Sistem Yang Diusulkan

Dari sistem yang ingin dirancang berikut merupakan prosedur sistem berjalan yang diusulkan :

1. Akses Sistem dan *Login* Sistem :

- a. Pengguna sistem : Terdapat berbagai posisi yang menggunakan sistem ini, seperti manajemen, tim bengkel, dan *driver*
- b. Setiap pengguna yang ingin masuk ke sistem harus menggunakan *username* dan *password* yang telah dibuat
- c. Tim bengkel : Bertanggung jawab dalam mengatur jadwal servis dan memastikan kendaraan selalu dalam kondisi prima.

d. *Driver* : Dapat melihat jadwal servis kendaraan yang mereka gunakan dan menerima pemberitahuan jadwal servis selanjutnya

2. *Booking* Servis Oleh *Driver* :

- a. *Driver* mengakses sistem dan memilih jenis servis yang dibutuhkan, seperti servis berkala, penggantian oli, dan pengecekan mesin
- b. *Driver* memasukkan waktu yang diinginkan untuk servis, yang kemudian akan disesuaikan dengan jadwal yang tersedia di sistem.
- c. Sistem akan menampilkan jadwal servis yang tersedia, dan *driver* melakukan mengkonfirmasi *booking*.

3. Penjadwalan Servis oleh Bagian Bengkel :

- a. Tim bengkel menerima *notifikasi* mengenai permintaan servis yang telah dilakukan oleh *driver*.
- b. Tim bengkel memeriksa ketersediaan waktu dan sumber daya (teknisi, suku cadang) yang diperlukan untuk servis tersebut.
- c. Jika tersedia, mereka akan mengkonfirmasi jadwal yang diusulkan oleh *driver*, jika tidak maka akan memberikan alternatif jadwal servis

4. Peningat servis:

- a. Sistem akan menyediakan fitur peningat melalui *dashboard* yang menampilkan jadwal servis yang akan datang. *Driver* dan tim bengkel dapat melihat *reminder* servis kendaraan melalui halaman khusus *reminder* yang tersedia dalam sistem.
- b. *Dashboard reminder* akan menampilkan daftar kendaraan beserta tanggal servis berikutnya, jenis servis yang diperlukan, dan status *reminder* (pending/selesai).

5. Pelaksanaan servis
  - a. Pada hari yang telah dijadwalkan kendaraan dibawa ke lokasi servis.
  - b. Tim bengkel akan melakukan pemeriksaan dan servis kendaraan.
  - c. Sistem akan menyimpan proses servis yang dilakukan, termasuk suku cadang yang diganti, jenis perawatan, dan tanggal servis.
6. Pelaporan dan Riwayat Servis
  - a. Setelah servis selesai, tim bengkel memasukkan laporan ke dalam sistem mengenai jenis servis yang dilakukan, suku cadang yang digunakan, dan kondisi kendaraan.
  - b. *Driver* dan manajemen dapat melihat riwayat servis kendaraan secara lengkap, termasuk tanggal-tanggal servis sebelumnya dan jenis perawatan yang telah dilakukan.
  - c. Manajemen dapat menggunakan data ini untuk menganalisis efisiensi perawatan kendaraan dan membuat keputusan terkait perawatan atau penggantian kendaraan.

### 3.11 Metodologi Penelitian

Metodologi penelitian yang digunakan pada penelitian ini adalah metode *waterfall*, Berikut merupakan proses tahapan metode *waterfall* dalam penelitian ini :

1. Analisis : Tahap pertama dalam metode *waterfall* adalah menganalisis kebutuhan dengan tujuan memahami persoalan yang dihadapi oleh sistem manajemen kendaraan operasional di PT Prima Indah Lestari. Pada tahap ini peneliti mengidentifikasi masalah, seperti pengelolaan jadwal servis kendaraan yang masih dilakukan secara manual, kesulitan mengakses catatan riwayat pemeliharaan, dan kekurangan sistem pengingat otomatis.

Data-data ini diperoleh melalui wawancara mendalam dengan pihak perusahaan.

2. Desain : Tahap desain dilakukan setelah memahami kebutuhan sistem, pada tahap ini rancangan arsitektur sistem dan spesifikasi teknis mulai dibentuk untuk menggambarkan bagaimana sistem akan beroperasi. Selain itu, dibuat *diagram* seperti *use case diagram* untuk menggambarkan interaksi pengguna dengan sistem dan *activity diagram* untuk memvisualisasikan alur kerja utama, seperti proses pemesanan servis dan pengingat otomatis. Desain antarmuka pengguna juga dirancang dengan fokus pada kemudahan penggunaan bagi administrator, tim bengkel, dan pengemudi.
3. Implementasi: Pada tahap implementasi, sistem mulai dikembangkan berdasarkan desain yang telah dibuat sebelumnya. Proses ini dimulai dengan penulisan kode untuk membangun *backend* menggunakan *framework* seperti *Laravel*. *Backend* ini mencakup pengembangan fitur utama seperti modul *booking* servis, pengingat otomatis, dan pelaporan riwayat pemeliharaan kendaraan. Selanjutnya, *frontend* dikembangkan menggunakan teknologi seperti *HTML*, *CSS*, dan *JavaScript* untuk menciptakan antarmuka yang ramah pengguna. Data kendaraan, jadwal servis, dan laporan dimasukkan ke dalam *database* yang dirancang menggunakan *MySQL*.
4. Pengujian: Setelah semua bagian sistem selesai dikembangkan, langkah selanjutnya adalah mengintegrasikan *frontend*, *backend*, dan *database* menjadi satu sistem yang utuh. Setelah sistem terintegrasi, dilakukan

serangkaian pengujian untuk memastikan bahwa sistem bekerja sesuai dengan yang diharapkan.

5. Pemeliharaan : Tahap pemeliharaan merupakan langkah akhir dalam metode *Waterfall*, yang dimulai setelah sistem secara resmi digunakan di perusahaan. Pada tahap ini, sistem akan terus dipantau untuk mendeteksi potensi masalah dan memperbaiki *bug* yang mungkin muncul selama penggunaan. Selain itu, sistem juga akan terus dikembangkan berdasarkan masukan dari pengguna, seperti penambahan fitur baru atau peningkatan kinerja. Tim bengkel secara rutin mengevaluasi data operasional dan kinerja sistem untuk memastikan aplikasi berjalan dengan stabil dan dapat terus mendukung operasional PT Prima Indah Lestari. Proses pemeliharaan ini bersifat berkelanjutan untuk memastikan sistem selalu relevan dengan kebutuhan bisnis yang terus berubah.

### **3.12 Requirement Elicitation**

*Requirement Elicitation* adalah proses yang sangat penting, karena untuk mengklarifikasi dan memahami secara mendalam kebutuhan sistem yang diinginkan oleh pengguna. hal ini bertujuan untuk merancang sistem manajemen kendaraan operasional di PT Prima Indah Lestari yang dapat berfungsi sesuai dengan ekspektasi pengguna.

#### **3.12.1 Tahap 1**

Pada tahap ini elisitasi disusun menurut hasil dari 5 pengguna, yaitu 2 admin HR, 1 supervisor admin, 1 staff administrasi , 1 *driver* dan berikut adalah hasil tabel nya:

Tabel 3. 5 Requirement Elicitation Tahap 1

No	User ingin sistem dapat :
1.	Memberikan akses <i>login</i> yang aman bagi pengguna
2.	Melakukan <i>booking</i> servis kendaraan operasional secara <i>online</i>
3.	Menampilkan jadwal servis selanjutnya
4.	Memastikan keamanan dan privasi data pengguna.
5.	Menyediakan informasi riwayat servis yang relevan dan memudahkan dalam pemeliharaan kendaraan.
6.	Bisa mengakses <i>master data</i> kendaraan
7.	Menyimpan dan menampilkan riwayat servis kendaraan.
8.	Menyediakan akses untuk berbagai peran pengguna seperti <i>driver</i> , <i>admin</i> , dll.
9.	Menampilkan <i>interface</i> yang mudah digunakan oleh semua pengguna
10	Mengelola jadwal servis berkala untuk seluruh kendaraan

### 3.12.2 Tahap 2

Pada tahap ini mengumpulkan permintaan dari pengguna, yang selanjutnya akan dibagi menjadi 3 kategori yaitu :

1. *Mandatory* (penting) : Kebutuhan yang harus ada didalam sistem
2. *Desirable* (perlu dipertimbangkan) : Kebutuhan yang diinginkan tetapi tidak mutlak harus ada
3. *Inesstial* (tidak relevan) : Kebutuhan yang tidak diperlukan dalam rancangan sistem.

Tabel 3. 6 Requirement Elicitation Tahap 1

No	User ingin sistem dapat :	<i>Mandatory</i>	<i>Desirable</i>	<i>Inessential</i>
1.	Memberikan akses <i>login</i> yang aman bagi pengguna	✓		
2.	Melakukan <i>booking</i> servis kendaraan operasional secara <i>online</i>	✓		
3.	Menampilkan jadwal servis selanjutnya	✓		
4.	Memastikan keamanan dan privasi data pengguna.	✓		

5.	Menyediakan informasi riwayat servis yang relevan dan memudahkan dalam pemeliharaan kendaraan.	✓		
6.	Bisa mengakses <i>master data</i> kendaraan	✓		
7.	Menyimpan dan menampilkan riwayat servis kendaraan.	✓		
8.	Menyediakan akses untuk berbagai peran pengguna seperti <i>driver</i> , admin, dll.		✓	
9.	Menampilkan <i>interface</i> yang mudah digunakan oleh semua pengguna.	✓		
10.	Mengelola jadwal servis berkala untuk seluruh kendaraan.	✓		

### 3.12.3 Tahap 3

Pada tahap ini, kebutuhan dari kategori *Mandatory* dan *Desirable* dikategorikan berdasarkan aspek Teknis (T), Operasional (O), dan Ekonomi (E) dengan tingkatan *Low* (L), *Middle* (M), dan *High* (H).

Tabel 3. 7 Requirement Elicitation Tahap 3

NO	Keinginan user pada sistem	T			O			E		
		H	M	L	H	M	L	H	M	L
1	Memberikan akses <i>login</i> yang aman bagi pengguna	✓				✓		✓		
2	Melakukan <i>booking</i> servis kendaraan operasional secara <i>online</i>		✓		✓			✓		
3	Menampilkan jadwal servis selanjutnya		✓		✓				✓	
4	Memastikan keamanan dan privasi data pengguna.	✓				✓				✓
5	Menyediakan informasi riwayat servis yang relevan dan memudahkan dalam pemeliharaan kendaraan.			✓		✓				✓
6	Bisa mengakses <i>master data</i> kendaraan		✓		✓				✓	

7	Menyimpan dan menampilkan riwayat servis kendaraan		✓			✓			✓	
8	Menyediakan akses untuk berbagai peran pengguna seperti <i>driver</i> , admin, dll.	✓			✓				✓	
9	Menampilkan <i>interface</i> yang mudah digunakan oleh semua pengguna			✓			✓			✓
10	Mengelola jadwal servis berkala untuk seluruh kendaraan.		✓		✓			✓		

### 3.12.4 Tahap 4

Pada tahap akhir ini, keinginan-keinginan *user* mulai disaring dan dapat digunakan sebagai dasar untuk pembuatan sistem

Tabel 3. 8 Requirement Elicitation Tahap 4

No.	<i>User</i> ingin sistem dapat :
1.	Memberikan akses <i>login</i> yang aman bagi pengguna
2.	Melakukan <i>booking</i> servis kendaraan operasional secara <i>online</i>
3.	Menampilkan jadwal servis selanjutnya
4.	Memastikan keamanan dan privasi data pengguna.
5.	Menyediakan informasi riwayat servis yang relevan dan memudahkan dalam pemeliharaan kendaraan.
6.	Bisa mengakses <i>master data</i> kendaraan
7.	Menyimpan dan menampilkan riwayat servis kendaraan.
8.	Menyediakan akses untuk berbagai peran pengguna seperti <i>driver</i> , admin, dll.
9.	Menampilkan <i>interface</i> yang mudah digunakan oleh semua Pengguna.
10.	Mengelola jadwal servis berkala untuk seluruh kendaraan.

### 3.13 Jadwal Penelitian

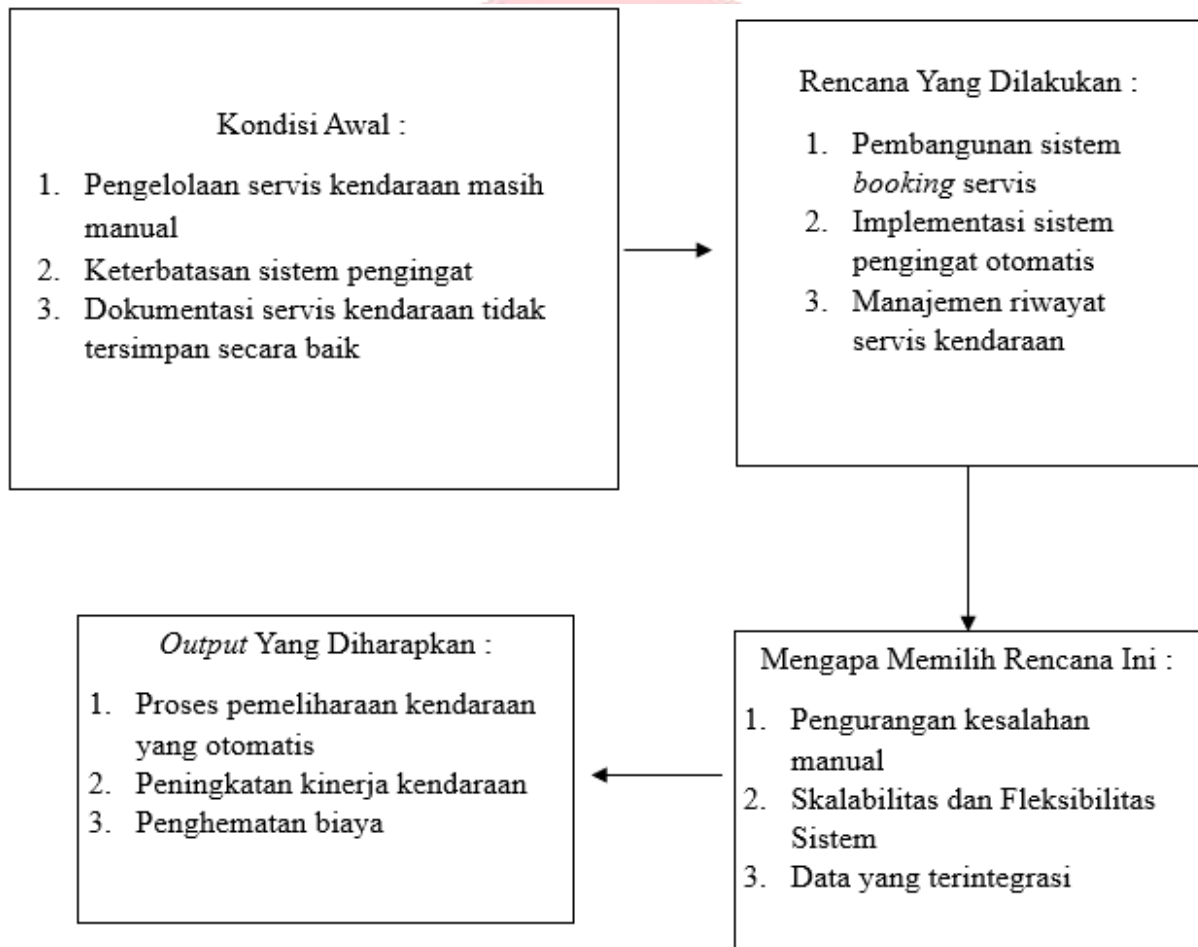
Jadwal penelitian yang direncanakan ini menguraikan kegiatan-kegiatan inti beserta perkiraan durasi pelaksanaannya, yang terstruktur dalam rentang waktu antara Maret hingga Juli. Berikut adalah rincian kegiatan berdasarkan fase dan bulan :

Tabel 3. 9 *Gantt Chart*

Kegiatan	2025																			
	Maret				April				Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
<b>Analisis</b>	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Pengamatan Terhadap Sistem Berjalan	█																			
Perumusan masalah & Identifikasi masalah	█	█																		
Menentukan Kebutuhan Sistem			█	█																
<b>Desain Sistem</b>																				
Melakukan Perancangan Arsitektur				█	█															
Melakukan Perancangan Desain					█	█														
Menentukan Spesifikasi Software						█	█													
Membuat Prototype							█	█												
<b>Pembuatan Sistem</b>																				
Koding Sistem									█	█	█	█	█	█	█	█	█	█	█	█
<b>Integrasi Dan Pengujian</b>																				
Unit Testing Setiap Modul																	█			
Integrasi Testing Setiap Modul																		█		
Black Box Testing																			█	
User Acceptance Testing																				█
<b>Pemeliharaan</b>																				
Pelatihan & Implementasi Sistem																				█
Monitoring & Maintenance																				█

### 3.14 Kerangka Pemikiran

Kerangka pemikiran ini menunjukkan bagaimana masalah yang diidentifikasi akan diatasi melalui implementasi sistem yang diusulkan, dan hasil yang diharapkan dari penerapan sistem tersebut. Berikut merupakan kerangka pemikiran dari penelitian ini:



Gambar 3. 5 Kerangka Pemikiran

Berikut merupakan penjelasan dari kerangka pemikiran diatas :

1. Kondisi saat ini :
  - a. Pengelolaan servis kendaraan masih manual : Pengelolaan servis dan perawatan kendaraan operasional di PT Prima Indah Lestari saat ini masih dilakukan secara manual. Hal ini menyebabkan berbagai kendala, seperti jadwal perawatan yang

tidak teratur, penurunan performa kendaraan, hingga meningkatnya biaya perawatan akibat kelalaian dalam menjalankan servis rutin.

- b. Keterbatasan sistem pengingat: Tidak adanya sistem yang secara otomatis memberikan pengingat terkait servis atau perawatan kendaraan membuat proses ini bergantung pada ingatan manual dan catatan yang tidak terorganisir dengan baik
- c. Dokumentasi servis tidak tersimpan dengan baik : Catatan perawatan kendaraan tidak dikelola secara rapi. Hal ini menyebabkan kesulitan dalam melacak riwayat perawatan, meningkatkan risiko kehilangan data, serta mempersulit pengambilan keputusan terkait pemeliharaan. Akibatnya, kendaraan bisa mengalami perawatan yang tidak tepat waktu, yang dapat mengurangi efisiensi operasional.

2. Rencana yang dilakukan :

- a. Pembangunan Sistem *Booking* Servis: Sistem aplikasi ini akan memungkinkan *driver* untuk memesan servis kendaraan secara *online* dengan sistem yang terpusat.
- b. Implementasi Sistem Pengingat Otomatis: Sistem pengingat otomatis berbasis *dashboard* akan dibangun untuk menampilkan jadwal perawatan yang akan datang secara *real-time*, memastikan tidak ada jadwal perawatan yang terlewat.
- c. Manajemen Riwayat Servis Kendaraan: Sistem aplikasi akan menyimpan seluruh riwayat servis kendaraan, sehingga informasi ini mudah diakses oleh tim bengkel dan manajemen untuk analisis dan pengambilan keputusan.

3. Mengapa memilih rencana ini

- a. Pengurangan kesalahan manual : Sistem otomatis akan mengurangi risiko kesalahan manusia yang sering terjadi pada sistem manual, seperti kelalaian mencatat jadwal servis, hilangnya data, atau miskomunikasi antara *driver* dan bagian pemeliharaan.
- b. Skalabilitas dan fleksibilitas sistem : Aplikasi berbasis *web* lebih mudah untuk dikembangkan dan ditingkatkan di masa mendatang, ketika perusahaan ingin menambah fungsi sistem
- c. Data yang terintegrasi : Sistem manajemen riwayat perawatan kendaraan mempermudah analisis data, sehingga keputusan terkait perbaikan atau penggantian kendaraan dapat dibuat dengan lebih akurat, yang pada akhirnya menghemat biaya dalam jangka panjang.

4. *Output* yang diharapkan :

- a. Proses Pemeliharaan Kendaraan yang Otomatis: Pengguna dapat memesan servis dengan mudah dan memantau jadwal perawatan kendaraan melalui *dashboard* pengingat otomatis
- b. Peningkatan Kinerja Kendaraan: Dengan perawatan yang dilakukan tepat waktu, kinerja kendaraan akan lebih optimal, mengurangi frekuensi kerusakan.
- c. Penghematan Biaya: Dengan pemeliharaan yang teratur dan sistematis, biaya yang dikeluarkan untuk perawatan yang tidak terencana dapat ditekan.