

**RANCANG BANGUN WEBSITE PEMESANAN KENDARAAN
LISTRIK MENGGUNAKAN PAYMENT GATEWAY MIDTRANS
(STUDI KASUS PT MARLIP INDO MANDIRI)**

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan program Strata 1,
Program Studi Teknik Informatika,
Universitas Pasundan Bandung

oleh:

Daffaa Aprilino
nrp.22.304.0025



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
JULI 2026**

LEMBAR PENGESAHAN
LAPORAN TUGAS AKHIR

Telah diujikan dan disahkan Laporan Tugas Akhir, dari :

Nama : Daffaa Aprilino

NRP: 22.304.0025

Dengan judul :

“Rancang Bangun Website Pemesanan Kendaraan Listrik Menggunakan Payment Gateway
Midtrans (Studi Kasus: PT Marlip Indo Mandiri)”

Bandung, 22 Juli 2026

Menyetujui,
Pembimbing utama,

(Erik, ST., M.Kom.)

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa :

1. Tugas akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya
2. Tugas akhir ini merupakan gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing
3. Dalam tugas akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan dalam sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada tugas akhir ini
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan tugas akhir ini bukan hasil karya saya sendiri atau adanya plagiarisme dalam bagian-bagian tertentu, saya bersedia menerima sanksi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

Bandung, 22 Juli 2026
Yang membuat pernyataan,
Materai

(Daffaa Aprilino)
NRP. 22.304.0025

ABSTRAK

Industri kendaraan listrik di Indonesia mengalami perkembangan yang semakin pesat, namun PT Marlip Indo Mandiri sebagai produsen kendaraan listrik lokal masih menghadapi kendala dalam proses pemesanan, pengelolaan dokumen *Purchase Order* (PO), validasi pembayaran, dan penyampaian informasi tahapan produksi. Proses yang belum terintegrasi dalam satu sistem menyebabkan potensi inefisiensi administrasi, risiko kesalahan pencatatan data, serta keterbatasan informasi bagi Konsumen mengenai status pesanan. Penelitian ini bertujuan untuk merancang dan membangun aplikasi pemesanan kendaraan listrik berbasis web yang mendukung katalog produk, pengajuan pesanan, persetujuan *Digital Purchase Order*, integrasi *payment gateway* untuk pembayaran dua termin, serta pelacakan tahapan produksi berbentuk *checklist*.

Metode pengembangan perangkat lunak yang digunakan adalah *Waterfall*, yang meliputi tahapan analisis kebutuhan, perancangan, implementasi, pengujian, dan *deployment*. Aplikasi dibangun menggunakan Next.js *App Router*, React, Tailwind CSS, Prisma ORM, PostgreSQL, NextAuth, serta Midtrans Snap API pada lingkungan *sandbox*. Sistem menyediakan hak akses bagi Pengunjung, Konsumen, Admin, dan Kepala Produksi. Kepala Produksi dapat memulai dan menyelesaikan tahapan produksi secara berurutan, serta melengkapi catatan dan foto dokumentasi apabila diperlukan.

Hasil penelitian menunjukkan bahwa aplikasi berhasil mendukung alur pemesanan kendaraan listrik secara digital, mulai dari katalog produk, Formulir Pemesanan, persetujuan *Digital Purchase Order*, pembayaran *Down Payment* (DP) dan pelunasan melalui Midtrans Snap, hingga pelacakan tahapan produksi. Sistem menerapkan empat tahapan *checklist*, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai. *Invoice* pelunasan diterbitkan setelah seluruh tahapan selesai. Berdasarkan pengujian *Black Box*, fitur utama sistem berjalan sesuai dengan kebutuhan fungsional yang dirancang.

Kata kunci: *e-commerce*, kendaraan listrik, *Purchase Order*, *payment gateway*, Midtrans Snap, tahapan produksi, Next.js, *Waterfall*, *Black Box Testing*.

ABSTRACT

The electric vehicle industry in Indonesia is growing rapidly. However, PT Marlip Indo Mandiri, as a local electric vehicle manufacturer, still faces challenges in the ordering process, Purchase Order (PO) document management, payment validation, and the delivery of production-stage information. The lack of integration within a single system may cause administrative inefficiency, data recording errors, and limited order-status information for customers. This study aims to design and develop a web-based electric vehicle ordering application that supports product catalog access, order submission, Digital Purchase Order approval, payment gateway integration for two-stage payments, and checklist-based production-stage tracking.

The software development method used in this study is Waterfall, which consists of requirement analysis, design, implementation, testing, and deployment stages. The application was developed using Next.js App Router, React, Tailwind CSS, Prisma ORM, PostgreSQL, NextAuth, and Midtrans Snap API in a sandbox environment. The system provides access rights for Visitors, Customers, Administrators, and the Head of Production. The Head of Production can start and complete production stages sequentially, as well as attach notes and photo documentation when required.

The results show that the application successfully supports the digital electric vehicle ordering workflow, including product catalog access, Order Form submission, Digital Purchase Order approval, Down Payment (DP) and final payment through Midtrans Snap, and production-stage tracking. The system applies four checklist stages: Production Preparation, Vehicle Assembly, Final Inspection, and Production Completion. The final-payment invoice is issued only after all stages have been completed. Based on Black Box testing, the main system features operate according to the designed functional requirements.

Keywords: *e-commerce, electric vehicle, Purchase Order, payment gateway, Midtrans Snap, production stages, Next.js, Waterfall, Black Box Testing.*

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah Subhanahu Wa Ta'ala atas segala rahmat, karunia, dan hidayah-Nya, sehingga penulis dapat menyelesaikan penyusunan Laporan Tugas Akhir dengan judul “Rancang Bangun Website Pemesanan Kendaraan Listrik Menggunakan Payment Gateway Midtrans (Studi Kasus: PT Marlip Indo Mandiri)”.

Tugas akhir ini disusun sebagai salah satu syarat untuk menyelesaikan studi dan memperoleh gelar Sarjana Teknik pada Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pasundan Bandung. Penelitian ini dimaksudkan untuk memberikan solusi digitalisasi pada sistem pemesanan kendaraan listrik lokal agar lebih efisien dan terintegrasi.

Penulis menyadari bahwa penyusunan laporan ini tidak akan terwujud tanpa bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan kelancaran, kesehatan, dan kemudahan dalam setiap langkah penulis.
2. Bapak Anggoro Ari Nurcahyo, S.T., M.Kom, selaku Ketua Program Studi Teknik Informatika, atas arahan, motivasi, dan fasilitas yang diberikan selama masa studi hingga penyusunan tugas akhir ini.
3. Bapak Erik, S.T., M.Kom. selaku Dosen Pembimbing yang telah meluangkan waktu untuk memberikan arahan, masukan, dan bimbingan yang sangat berarti selama proses penyusunan Laporan Tugas Akhir ini.
4. Bapak Wanda Gusdya, S.T., M.T., selaku Koordinator Tugas Akhir yang telah banyak membantu memperlancar proses administrasi dan akademik.
5. Seluruh Bapak/Ibu Dosen Program Studi Teknik Informatika Universitas Pasundan yang telah membagikan ilmunya selama penulis duduk di bangku perkuliahan.
6. Pihak PT Marlip Indo Mandiri yang telah memberikan izin penelitian serta membantu dalam penyediaan data yang diperlukan.

Selain itu, dengan segenap rasa cinta dan kasih, karya ini penulis persembahkan untuk:

1. Kepada kedua orang tua penulis tercinta, Mamah Aisyah Tanjung dan Papah Ressa Natalio. Terima kasih atas doa, kasih sayang, serta pengorbanan moril maupun materil yang tiada pernah berhenti. Kalian adalah sumber kekuatan terbesar yang selalu mendorong penulis untuk bertahan dan tidak menyerah, bahkan di masa tersulit. Karya ini adalah wujud syukur dan upaya penulis untuk membalas kepercayaan serta menjadi anak yang membanggakan. Semoga Allah SWT membalas segala kebaikan kalian dengan kesehatan, kebahagiaan, serta pahala yang berlipat ganda di dunia dan akhirat.
2. Kepada adik kembar penulis, Naurah Aprilia. Terima kasih telah menjadi pelengkap dinamika dalam hidup penulis. Kita berjuang bersama mengejar gelar dan mimpi sejak kecil. Melihatmu berhasil menempuh sidang kemarin adalah kebanggaan tersendiri bagi penulis. Kehadiranmu adalah motivasi yang selalu menguatkan dan semoga langkahmu di masa depan senantiasa dipermudah serta segala impianmu tercapai.
3. Kepada Yusri, Aura, dan Fauzan, sahabat sejak masa sekolah. Terima kasih atas dukungan yang konsisten hingga saat ini. Senang rasanya bisa saling menyemangati dalam mengejar gelar dan mimpi bersama. Semoga langkah kita ke depan selalu dimudahkan dan sukses di jalan masing-masing.
4. Kepada Algi, Farid, dan Miftah, sahabat sejak awal perkuliahan. Terima kasih atas segala obrolan, canda tawa, serta tumpukan cerita yang kita lalui bersama. Senang rasanya bisa berjuang meraih gelar ini di samping kalian. Semoga langkah kita setelah ini senantiasa dimudahkan dan kita sukses di jalan masing-masing.

5. Kepada Prospere (Algi, Dena, Farid, Miftah, dan Syerli). Terima kasih telah menjadi bagian dari perjalanan ini, serta atas setiap tawa, diskusi, bantuan, dan kebersamaan yang membuat masa perkuliahan terasa lebih bermakna. Semoga langkah kita setelah ini selalu dimudahkan, diberikan kesuksesan di jalan masing-masing, dan semoga suatu saat nanti kita bisa kembali bertemu untuk mengenang semua cerita yang pernah kita lalui bersama.
6. Kepada kawan-kawan “manifesting 4 tahun aamiin”: Akbar, Daniel, Darryl, Farid, Fikry, Algi, Galuh, Joo, Esa, Miftah, Caca, Fadli, Dafsus, RB, Ebi, Rafi, dan Ranca. Terima kasih atas obrolan, diskusi, canda tawa, kumpul-kumpul, serta beragam informasi baru yang selalu mewarnai perjalanan kita. Semoga ilmu dan pelajaran yang kita bagi selama ini menjadi bekal berharga untuk masa depan kita semua.
7. Terakhir, kepada diri sendiri, Daffaa Aprilino. Terima kasih atas segala usaha, ketekunan, dan ketahanan dalam menyelesaikan proses pendidikan ini. Terima kasih telah mampu bertahan menghadapi berbagai tantangan hingga semua proses itu terbayar. Pencapaian ini bukan hanya sekadar tentang gelar akademik, melainkan bukti nyata tentang bagaimana kamu bertumbuh menjadi pribadi yang lebih kuat dan dewasa. Kelak, tulisan ini akan dibaca dengan rasa bangga sebagai pengingat bahwa semua perjuangan ini sangat layak diperjuangkan. *I am so proud of you.*

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan karena keterbatasan ilmu dan pengalaman. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan dan penyempurnaan di masa mendatang.

Akhir kata, penulis berharap Laporan Tugas Akhir ini dapat memberikan manfaat bagi perkembangan teknologi informasi, khususnya di lingkungan Universitas Pasundan dan industri kendaraan listrik nasional.

Bandung, 22 Juli 2026

Penulis

(Daffaa Aprilino)

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	v
DAFTAR ISTILAH.....	ix
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xii
DAFTAR LAMPIRAN	xv
DAFTAR SIMBOL	xvi
BAB 1 PENDAHULUAN	1-1
1.1 Latar Belakang	1-1
1.2 Identifikasi Masalah.....	1-2
1.3 Tujuan Tugas Akhir	1-2
1.4 Lingkup Tugas Akhir.....	1-3
1.5 Metodologi Tugas Akhir.....	1-4
1.6 Standar dan Kakas.....	1-6
1.7 Sistematika Penulisan	1-7
BAB 2 LANDASAN TEORI.....	2-1
2.1 Teori yang Digunakan.....	2-1
2.1.1 Metodologi Waterfall.....	2-1
2.1.2 Konsep Rancang Bangun.....	2-2
2.1.3 Kendaraan Listrik (<i>Electric Vehicle</i>)	2-2
2.1.4 Konsep Produk (Barang) dalam Sistem <i>Made by Order</i>	2-2
2.1.5 Sistem Pemesanan.....	2-3
2.1.6 Purchase Order dan Invoice Digital	2-3

2.1.7	Konsep E-Commerce (<i>Electronic Commerce</i>)	2-4
2.1.8	Payment Gateway (Midtrans)	2-4
2.1.9	Webhook	2-5
2.1.10	Aplikasi Berbasis Web	2-5
2.1.11	Autentikasi dan Otorisasi	2-6
2.1.12	Framework Next.js	2-6
2.1.13	Route Handler dan Server Action pada Next.js App Router	2-7
2.1.14	React	2-7
2.1.15	TypeScript	2-8
2.1.16	Vercel	2-9
2.1.17	Prisma ORM	2-9
2.1.18	PostgreSQL	2-10
2.1.19	Neon (Serverless PostgreSQL)	2-10
2.1.20	Tailwind CSS	2-11
2.1.21	Unified Modeling Language (UML)	2-11
2.1.22	Entity Relationship Diagram (ERD)	2-12
2.1.23	Pengujian Black-Box (<i>Black-Box Testing</i>)	2-12
2.2	Penelitian Terdahulu	2-13
BAB 3	SKEMA PENELITIAN	3-1
3.1	Alur Penyelesaian Tugas Akhir	3-1
3.2	Perumusan Masalah	3-4
3.2.1	Analisis Masalah	3-4
3.2.2	Solusi Masalah	3-5
3.2.3	Skema Analisis	3-6
3.3	Profil Tempat Penelitian	3-7
3.3.1	Objek Penelitian	3-7
3.3.2	Visi dan Misi	3-7

3.4	Teknik Pengumpulan Data.....	3-8
3.5	Instrumen Penelitian	3-9
3.5.1	Instrumen Pengumpulan Data.....	3-9
3.5.2	Alat Pengembangan Sistem	3-9
3.6	Metode Pengembangan Perangkat Lunak.....	3-10
BAB 4	ANALISIS DAN PERANCANGAN	4-1
4.1	Analisis Sistem.....	4-1
4.1.1	Analisis Alur Bisnis yang Berjalan.....	4-1
4.1.2	Analisis Alur Bisnis Usulan.....	4-3
4.1.3	Analisis Kebutuhan Berdasarkan Wawancara	4-6
4.1.4	Analisis Kebutuhan Pengguna (<i>User Requirement</i>)	4-8
4.1.5	Analisis Kebutuhan Perangkat Lunak (<i>Software Requirement</i>)	4-8
4.1.6	Analisis Kebutuhan Non-Fungsional.....	4-9
4.2	Perancangan Sistem	4-10
4.2.1	Use Case Diagram.....	4-10
4.2.2	Skenario Use Case	4-13
4.2.3	Activity Diagram	4-20
4.2.4	Sequence Diagram	4-23
4.2.5	Entity Relationship Diagram.....	4-29
4.2.6	Perancangan Antarmuka	4-31
BAB 5	IMPLEMENTASI DAN PENGUJIAN	5-1
5.1	Implementasi Sistem.....	5-1
5.1.1	Lingkungan Implementasi	5-1
5.1.2	Struktur Folder dan Berkas	5-2
5.1.3	Konfigurasi Aplikasi.....	5-4
5.1.4	Implementasi Basis Data.....	5-4
5.1.5	Implementasi Route Handler dan API	5-5

5.2	Implementasi Antarmuka Berdasarkan Role	5-7
5.2.1	Antarmuka Pengunjung dan Konsumen	5-7
5.2.2	Antarmuka Admin	5-15
5.2.3	Antarmuka Kepala Produksi	5-18
5.3	Implementasi Fitur Utama	5-19
5.3.1	Implementasi Katalog Produk dan Formulir Pemesanan.....	5-20
5.3.2	Implementasi Digital Purchase Order dan Tanda Tangan Elektronik	5-21
5.3.3	Implementasi Pembayaran Midtrans Snap.....	5-23
5.3.4	Implementasi Invoice dan Validasi Pembayaran Webhook	5-25
5.3.5	Implementasi Pelacakan Tahapan Produksi.....	5-26
5.4	Pengujian Fungsional Black Box.....	5-28
5.4.1	Pengujian Katalog dan Pemesanan	5-29
5.4.2	Pengujian Pengelolaan Produk Admin	5-29
5.4.3	Pengujian Peninjauan dan Persetujuan PO	5-30
5.4.4	Pengujian Pembayaran DP dan Pelunasan.....	5-30
5.4.5	Pengujian Pembaruan Tahapan Produksi.....	5-31
5.4.6	Pengujian Hak Akses Pengguna	5-32
5.4.7	Rekapitulasi Hasil Pengujian Black Box	5-33
5.5	Deployment.....	5-33
BAB 6	KESIMPULAN DAN SARAN.....	6-1
6.1	Kesimpulan	6-1
6.2	Saran	6-2
DAFTAR PUSTAKA.....		xvii
LAMPIRAN		L-1

DAFTAR ISTILAH

Istilah	Penjelasan / Keterangan
Black Box Testing	Metode pengujian sistem berdasarkan masukan dan keluaran tanpa melihat struktur kode program.
Digital Purchase Order	Dokumen Purchase Order berbentuk digital yang ditampilkan melalui sistem dan dapat dicetak atau disimpan sebagai PDF melalui fungsi cetak peramban.
Down Payment (DP)	Pembayaran uang muka sebesar 50% dari total tagihan pesanan.
E-Commerce	Aktivitas perdagangan produk atau jasa yang dilakukan melalui media internet.
Payment Gateway	Layanan perantara yang memproses dan memvalidasi pembayaran digital.
Route Handler	Fitur pada Next.js App Router untuk menangani permintaan HTTP pada sisi server.
Sandbox	Lingkungan pengujian yang digunakan untuk mencoba sistem tanpa transaksi nyata.
Server-Side Rendering	Proses menampilkan halaman web dari sisi server sebelum dikirimkan ke pengguna.
Pelacakan Tahapan Produksi	Fitur untuk memantau status dan riwayat tahapan produksi pesanan berdasarkan pembaruan Kepala Produksi.
User Interface	Tampilan antarmuka yang digunakan pengguna untuk berinteraksi dengan sistem.
Webhook	Notifikasi otomatis dari layanan eksternal ke sistem ketika terjadi perubahan status transaksi.
Deployment	Proses penerapan aplikasi ke lingkungan server agar dapat diakses secara daring.
Framework	Kerangka kerja yang digunakan untuk membantu proses pengembangan aplikasi.
Snap Token	Token pembayaran dari Midtrans yang digunakan untuk menampilkan halaman pembayaran Midtrans Snap.
Role	Hak akses pengguna dalam sistem berdasarkan fungsi dan kewenangannya.

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu.....	2-13
Tabel 3.1 Alur Penyelesaian Tugas Akhir.....	3-1
Tabel 3.2 Deskripsi Skema Analisis.....	3-6
Tabel 4.1 Hasil Wawancara.....	4-6
Tabel 4.2 Analisis Kebutuhan Pengguna.....	4-8
Tabel 4.3 Analisis Kebutuhan Perangkat Lunak	4-8
Tabel 4.4 Analisis Kebutuhan Non-Fungsional	4-9
Tabel 4.5 Deskripsi Aktor pada Use Case Diagram.....	4-12
Tabel 4.6 Deskripsi Use Case Diagram.....	4-12
Tabel 4.7 Skenario UC-01: Registrasi Akun.....	4-13
Tabel 4.8 Skenario UC-02: Login	4-14
Tabel 4.9 Skenario UC-03: Melihat Katalog Produk	4-14
Tabel 4.10 Skenario UC-04: Mengelola Katalog Produk	4-15
Tabel 4.11 Skenario UC-05: Mengajukan Pemesanan Standar atau Kustom	4-15
Tabel 4.12 Skenario UC-06: Mengelola Pesanan & Menerbitkan PO.....	4-16
Tabel 4.13 Skenario UC-07: Mencetak atau Menyimpan Dokumen PO & Invoice	4-16
Tabel 4.14 Skenario UC-08: Melakukan Pembayaran DP dan Pelunasan	4-17
Tabel 4.15 Skenario UC-09: Validasi Pembayaran Otomatis	4-18
Tabel 4.16 Skenario UC-10: Memperbarui Tahapan Produksi	4-19
Tabel 4.17 Skenario UC-11: Melacak Tahapan Produksi	4-20
Tabel 4.18 Halaman Beranda	4-31
Tabel 4.19 Halaman Katalog.....	4-32
Tabel 4.20 Halaman Detail Produk dan Formulir Pemesanan	4-33
Tabel 4.21 Halaman Registrasi.....	4-34
Tabel 4.22 Halaman Login	4-35
Tabel 4.23 Halaman Dashboard Konsumen	4-35
Tabel 4.24 Halaman Digital Purchase Order dan Persetujuan Draf PO	4-36
Tabel 4.25 Halaman Tanda Tangan Elektronik Konsumen	4-37
Tabel 4.26 Halaman Pembayaran Dua Termin	4-37
Tabel 4.27 Halaman Invoice Pembayaran.....	4-38
Tabel 4.28 Halaman Pembayaran Midtrans Snap	4-39
Tabel 4.29 Halaman Pelacakan Tahapan Produksi	4-40

Tabel 4.30 Halaman Dashboard Admin	4-41
Tabel 4.31 Halaman Manajemen Produk	4-42
Tabel 4.32 Halaman Pengelolaan Pesanan Admin.....	4-43
Tabel 4.33 Halaman Tanda Tangan Elektronik Admin.....	4-44
Tabel 4.34 Halaman Dashboard Kepala Produksi.....	4-45
Tabel 4.35 Halaman Pembaruan Tahapan Produksi.....	4-46
Tabel 5.1 Kebutuhan Perangkat Keras	5-1
Tabel 5.2 Kebutuhan Perangkat Lunak	5-1
Tabel 5.3 Struktur Folder Aplikasi	5-3
Tabel 5.4 Konfigurasi Environment Variable	5-4
Tabel 5.5 Implementasi Entitas Basis Data.....	5-5
Tabel 5.6 Implementasi Route Handler.....	5-6
Tabel 5.7 Pengujian Katalog dan Pemesanan.....	5-29
Tabel 5.8 Pengujian Pengelolaan Produk Admin.....	5-29
Tabel 5.9 Pengujian Peninjauan dan Persetujuan PO.....	5-30
Tabel 5.10 Pengujian Pembayaran DP dan Pelunasan	5-30
Tabel 5.11 Pengujian Pembaruan Tahapan Produksi	5-31
Tabel 5.12 Pengujian Hak Akses Pengguna	5-32
Tabel 5.13 Rekapitulasi Hasil Pengujian Black Box.....	5-33

DAFTAR GAMBAR

Gambar 1.1 Metodologi Tugas Akhir	1-4
Gambar 2.1 Metode Waterfall	2-1
Gambar 2.2 Logo Payment Gateway (Midtrans)	2-4
Gambar 2.3 Logo Framework Next.js	2-6
Gambar 2.4 Logo React	2-7
Gambar 2.5 Logo TypeScript	2-8
Gambar 2.6 Logo Vercel	2-9
Gambar 2.7 Logo PostgreSQL	2-10
Gambar 2.8 Logo Tailwind CSS	2-11
Gambar 2.9 Ilustrasi Pengujian Black-Box	2-12
Gambar 3.1 Fishbone Masalah	3-4
Gambar 3.2 Fishbone Solusi	3-5
Gambar 3.3 Skema Analisis	3-6
Gambar 3.4 Logo PT Marlip Indo Mandiri	3-7
Gambar 4.1 Flowmap Alur Bisnis yang Berjalan	4-2
Gambar 4.2 Flowmap Alur Bisnis Usulan	4-5
Gambar 4.3 Use Case Diagram	4-11
Gambar 4.4 Activity Diagram Pemesanan Standar dan Kustom	4-21
Gambar 4.5 Activity Diagram Pembayaran via Midtrans	4-22
Gambar 4.6 Diagram Sequence UC-01 - Registrasi Akun	4-23
Gambar 4.7 Diagram Sequence UC-02 - Login	4-23
Gambar 4.8 Diagram Sequence UC-03 - Melihat Katalog Produk	4-24
Gambar 4.9 Diagram Sequence UC-04 - Mengelola Katalog Produk	4-24
Gambar 4.10 Diagram Sequence UC-05 - Mengajukan Pemesanan Standar atau Kustom	4-25
Gambar 4.11 Diagram Sequence UC-06 - Mengelola Pesanan & Menerbitkan PO	4-26
Gambar 4.12 Diagram Sequence UC-07 - Mencetak atau Menyimpan Dokumen PO & Invoice	4-27
Gambar 4.13 Diagram Sequence UC-08 - Melakukan Pembayaran (DP & Lunas)	4-27
Gambar 4.14 Diagram Sequence UC-09 - Validasi Pembayaran Otomatis	4-28
Gambar 4.15 Diagram Sequence UC-10 – Memperbarui Tahapan Produksi	4-28
Gambar 4.16 Diagram Sequence UC-11 – Melacak Tahapan Produksi	4-29

Gambar 4.17 Entity Relationship Diagram (ERD).....	4-30
Gambar 4.18 Halaman Beranda	4-31
Gambar 4.19 Halaman Katalog	4-32
Gambar 4.20 Halaman Detail Produk dan Formulir Pemesanan	4-33
Gambar 4.21 Halaman Registrasi.....	4-34
Gambar 4.22 Halaman Login	4-35
Gambar 4.23 Halaman Dashboard Konsumen	4-35
Gambar 4.24 Halaman Digital Purchase Order dan Persetujuan Draf PO	4-36
Gambar 4.25 Halaman Tanda Tangan Elektronik Konsumen.....	4-37
Gambar 4.26 Halaman Pembayaran Dua Termin.....	4-37
Gambar 4.27 Halaman Invoice Pembayaran	4-38
Gambar 4.28 Halaman Pembayaran Midtrans Snap.....	4-39
Gambar 4.29 Halaman Pelacakan Tahapan Produksi.....	4-40
Gambar 4.30 Halaman Dashboard Admin	4-41
Gambar 4.31 Halaman Manajemen Produk	4-42
Gambar 4.32 Halaman Pengelolaan Pesanan Admin	4-43
Gambar 4.33 Halaman Tanda Tangan Elektronik Admin.....	4-44
Gambar 4.34 Halaman Dashboard Kepala Produksi	4-45
Gambar 4.35 Halaman Pembaruan Tahapan Produksi	4-46
Gambar 5.1 Struktur Folder App Router	5-2
Gambar 5.2 Struktur Folder Komponen dan Konfigurasi Aplikasi	5-3
Gambar 5.3 Antarmuka Halaman Beranda	5-8
Gambar 5.4 Antarmuka Halaman Katalog Produk.....	5-9
Gambar 5.5 Antarmuka Halaman Detail Produk dan Formulir Pemesanan	5-10
Gambar 5.6 Antarmuka Halaman Login	5-10
Gambar 5.7 Antarmuka Halaman Registrasi.....	5-11
Gambar 5.8 Antarmuka Dashboard Konsumen.....	5-11
Gambar 5.9 Antarmuka Persetujuan Draf Purchase Order Konsumen	5-12
Gambar 5.10 Antarmuka Pembayaran Midtrans Snap	5-13
Gambar 5.11 Antarmuka Invoice Pembayaran	5-14
Gambar 5.12 Antarmuka Pelacakan Tahapan Produksi	5-15
Gambar 5.13 Antarmuka Dashboard Admin.....	5-16
Gambar 5.14 Antarmuka Manajemen Produk.....	5-16
Gambar 5.15 Antarmuka Pengelolaan Pesanan Admin	5-16

Gambar 5.16 Antarmuka Peninjauan Pesanan dan Draf PO	5-17
Gambar 5.17 Antarmuka Tanda Tangan Elektronik Admin	5-17
Gambar 5.18 Antarmuka Dashboard Kepala Produksi	5-18
Gambar 5.19 Antarmuka Pembaruan Tahapan Produksi	5-19
Gambar 5.20 Tampilan Katalog Produk.....	5-20
Gambar 5.21 Tampilan Detail Produk dan Formulir Pemesanan.....	5-21
Gambar 5.22 Tampilan Digital Purchase Order	5-22
Gambar 5.23 Tampilan Tanda Tangan Elektronik Admin	5-22
Gambar 5.24 Tampilan Tanda Tangan Elektronik Konsumen.....	5-23
Gambar 5.25 Tampilan Halaman Pembayaran Dua Termin	5-24
Gambar 5.26 Tampilan Midtrans Snap	5-24
Gambar 5.27 Tampilan Invoice Pembayaran	5-25
Gambar 5.28 Tampilan Status Pembayaran Berhasil	5-26
Gambar 5.29 Tampilan Pelacakan Tahapan Produksi Konsumen	5-27
Gambar 5.30 Tampilan Pembaruan Tahapan Produksi Kepala Produksi	5-28

DAFTAR LAMPIRAN

Lampiran A Berita Acara Wawancara	2
Lampiran B Contoh Purchase Order	4
Lampiran C Contoh Invoice	5
Lampiran D Implementasi Model User dan Produk.....	6
Lampiran E Implementasi Model Transaksi	7
Lampiran F Implementasi Validasi Pengajuan Pesanan	8
Lampiran G Implementasi Pembuatan Purchase Order	9
Lampiran H Implementasi Validasi Webhook Midtrans.....	10
Lampiran I Implementasi Pembaruan Status Pembayaran dan Pesanan	11

DAFTAR SIMBOL

Simbol Use Case Diagram

No	Simbol	Nama	Deskripsi
1		Aktor	Mewakili pengguna atau sistem eksternal yang berinteraksi dengan sistem.
2		Use Case	Merepresentasikan fungsionalitas atau proses tertentu yang disediakan oleh sistem aplikasi kepada aktor.
3		Association	Menunjukkan hubungan interaksi antara aktor dan use case.

Simbol Activity Diagram

No	Simbol	Nama	Deskripsi
1		Node Awal	Menandakan titik awal dimulainya suatu aliran pekerjaan atau aktivitas.
2		Node Akhir	Menandakan titik akhir atau berhentinya seluruh aliran aktivitas dalam diagram.
3		Aktivitas	Merepresentasikan suatu pekerjaan, proses, atau tindakan yang dilakukan oleh sistem atau aktor.
4		Keputusan	Menunjukkan percabangan alur aktivitas berdasarkan kondisi atau pilihan tertentu.
5		Aliran Kontrol	Menunjukkan arah urutan eksekusi dari satu tahapan aktivitas ke aktivitas selanjutnya.

Simbol Flowmap

No	Simbol	Nama	Deskripsi
1		Terminator	Menunjukkan awal atau akhir suatu alur proses.
2		Proses	Menunjukkan aktivitas atau langkah kerja yang dilakukan.
3		Dokumen	Menunjukkan dokumen atau keluaran informasi, seperti Purchase Order dan Invoice.
4		Keputusan	Menunjukkan percabangan alur berdasarkan kondisi atau pilihan tertentu.
5		Garis Alir	Menunjukkan arah dan urutan aliran proses antar simbol.

BAB 1

PENDAHULUAN

Bab ini memberikan pandangan umum mengenai pelaksanaan tugas akhir. Adapun penjelasan tersebut meliputi latar belakang, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, metodologi tugas akhir, standar dan kaskas, serta sistematika penulisan tugas akhir.

1.1 Latar Belakang

Perkembangan teknologi informasi di era Revolusi Industri 4.0 telah membawa perubahan signifikan pada berbagai sektor industri, termasuk industri otomotif. Di Indonesia, transformasi ini didukung oleh kebijakan pemerintah dalam percepatan program Kendaraan Bermotor Listrik Berbasis Baterai (KBLBB) melalui Peraturan Presiden No. 79 Tahun 2023. Kebijakan ini bertujuan mewujudkan ekosistem transportasi yang ramah lingkungan dan mendorong produsen lokal untuk berinovasi, tidak hanya pada sisi produk, tetapi juga pada sistem layanan dan pemasaran digital [IND23].

PT Marlip Indo Mandiri (MIM) memposisikan diri sebagai pionir kendaraan listrik produksi lokal yang memiliki sejarah panjang dalam pengembangan teknologi otomotif nasional. Berdasarkan profil resmi perusahaan, Marlip bermula dari hasil riset kendaraan listrik yang dilakukan oleh Lembaga Ilmu Pengetahuan Indonesia (LIPI) sejak tahun 1996. Riset tersebut kemudian dikomersialisasi menjadi entitas bisnis pada tahun 2004 dan diresmikan secara langsung oleh Presiden ke-6 Republik Indonesia, Dr. Susilo Bambang Yudhoyono. Saat ini, perusahaan berfokus memproduksi kendaraan listrik serbaguna (Special Purpose Vehicle) untuk berbagai kebutuhan, mulai dari mobil golf, patroli keamanan, hingga transportasi rumah sakit [MAR26].

Meskipun telah beroperasi lebih dari dua dekade, proses bisnis penjualan kendaraan di PT Marlip khususnya untuk pesanan kustom (Made by Order) masih menghadapi kendala operasional. Berdasarkan hasil observasi dan wawancara dengan manajemen perusahaan, ditemukan bahwa sistem transaksi saat ini masih dikelola secara konvensional. Konsumen harus melakukan pemesanan melalui komunikasi manual, menandatangani dokumen Purchase Order (PO) fisik, dan melakukan pembayaran melalui transfer bank yang memerlukan verifikasi manual oleh staf administrasi.

Kondisi ini memunculkan risiko human error dalam validasi pembayaran bertahap (Down Payment dan Pelunasan) serta inefisiensi waktu. Selain itu, konsumen sering kali kesulitan mendapatkan informasi terkini mengenai status perakitan kendaraan mereka karena ketiadaan fitur pelacakan (tracking) yang transparan. Hal ini dapat menurunkan kepercayaan konsumen terhadap layanan perusahaan.

Untuk mengatasi permasalahan tersebut, penelitian ini mengusulkan pengembangan sistem pemesanan berbasis web yang terintegrasi. Solusi yang ditawarkan meliputi pengembangan aplikasi menggunakan kerangka kerja Next.js yang memiliki performa tinggi [VER26], serta integrasi Payment Gateway Midtrans menggunakan Snap API untuk mengotomatisasi pembayaran bertahap. Selain itu, sistem akan dilengkapi fitur Digital Purchase Order dan pelacakan tahapan produksi berbentuk checklist untuk meningkatkan transparansi informasi produksi kepada konsumen berdasarkan pembaruan Kepala Produksi.

1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas, permasalahan yang dapat diidentifikasi dirumuskan dalam bentuk pertanyaan sebagai berikut:

1. Bagaimana mengotomatisasi validasi pembayaran yang saat ini dilakukan melalui pemeriksaan bukti transfer dan mutasi rekening secara manual agar lebih efisien dan akurat?
2. Bagaimana menampilkan dokumen Purchase Order (PO) digital pada sistem agar Konsumen dapat meninjau dan menyetujui pesanan tanpa bergantung pada dokumen fisik?
3. Bagaimana menyediakan informasi tahapan produksi kendaraan yang ditetapkan berdasarkan standar proses produksi PT Marlip Indo Mandiri serta diperbarui oleh Kepala Produksi yang berwenang?
4. Bagaimana mengembangkan website profil perusahaan yang bersifat informatif menjadi aplikasi pemesanan berbasis web yang mendukung katalog produk, pemesanan, pengelolaan PO, dan pembayaran?

1.3 Tujuan Tugas Akhir

Berdasarkan identifikasi masalah tersebut di atas, maka dapat diperoleh tujuan tugas akhir yaitu:

1. Merancang dan membangun sistem pemesanan kendaraan listrik yang terintegrasi dengan Payment Gateway Midtrans untuk mengotomatisasi validasi pembayaran dan meminimalkan kesalahan pemeriksaan manual.
2. Merancang dan mengimplementasikan fitur Digital Purchase Order agar sistem dapat menampilkan dokumen PO digital untuk ditinjau dan disetujui oleh Konsumen.
3. Merancang dan membangun fitur pelacakan tahapan produksi agar Konsumen dapat memperoleh informasi status produksi kendaraan berdasarkan standar proses produksi perusahaan dan pembaruan Kepala Produksi.
4. Merancang dan membangun aplikasi pemesanan kendaraan listrik berbasis web menggunakan Next.js sebagai pengembangan dari website profil perusahaan yang sebelumnya bersifat informatif.

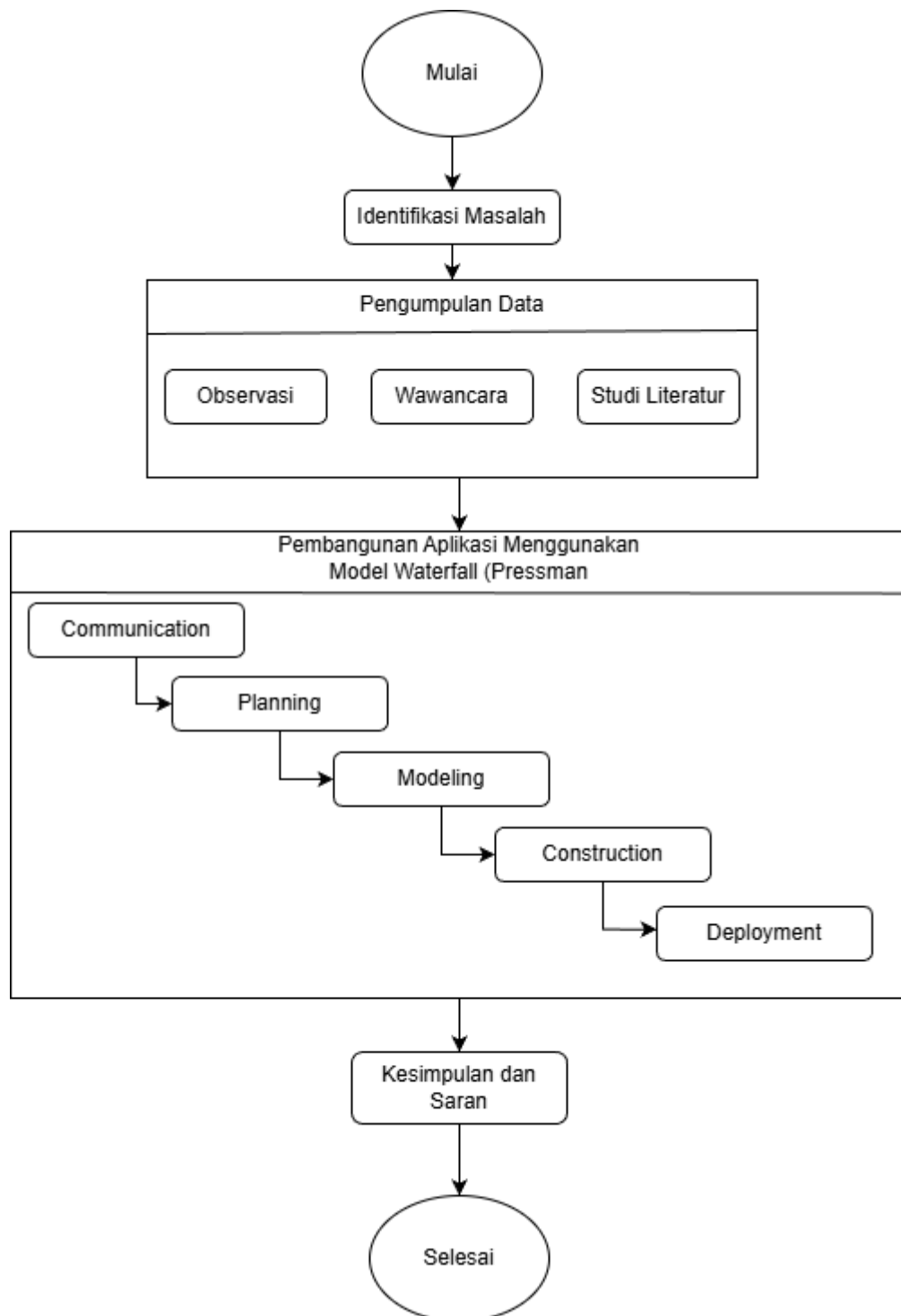
1.4 Lingkup Tugas Akhir

Berikut merupakan lingkup Tugas Akhir ini:

1. Aplikasi dikembangkan berbasis web menggunakan framework Next.js.
2. Sistem pembayaran menggunakan Midtrans Snap API pada lingkungan Sandbox untuk pengujian pembayaran dua termin, yaitu Down Payment (DP) sebesar 50% dan pelunasan sebesar sisa tagihan.
3. Pengunjung dapat melihat katalog dan detail produk tanpa login, sedangkan pengajuan pesanan hanya dapat dilakukan oleh pengguna dengan role Konsumen.
4. Formulir pemesanan menyediakan pilihan warna utama Hitam atau Putih, varian Standar atau Kustom, serta jumlah pesanan 1 sampai 20 unit. Catatan kebutuhan khusus hanya wajib diisi untuk varian Kustom.
5. Pembayaran DP dapat dilakukan setelah Konsumen menyetujui draf Purchase Order, sedangkan pembayaran pelunasan tersedia setelah seluruh tahapan produksi kendaraan dinyatakan selesai oleh Kepala Produksi.
6. Fitur pelacakan menampilkan tahapan produksi kendaraan berdasarkan standar proses produksi perusahaan dan pembaruan oleh Kepala Produksi. Pembaruan tahapan produksi dilakukan menggunakan checklist serta dokumentasi foto pada setiap tahapan yang diperlukan.
7. Aplikasi mencakup tiga hak akses pengguna utama, yaitu Konsumen, Admin, dan Kepala Produksi.
8. Penelitian tidak membahas perhitungan biaya pengiriman otomatis melalui Shipping API dari pihak ekspedisi.

1.5 Metodologi Tugas Akhir

Dalam penyusunan tugas akhir ini, penulis menerapkan tahapan metodologi yang sistematis untuk memastikan penelitian berjalan terarah. Alur penelitian digambarkan dalam diagram berikut:



Gambar 1.1 Metodologi Tugas Akhir

1. Tahap Identifikasi Masalah

Langkah awal penelitian dimulai dengan merumuskan masalah yang terjadi di lapangan. Pada tahap ini, penulis mengidentifikasi kendala utama pada sistem penjualan di PT Marlip Indo Mandiri, yaitu inefisiensi proses administrasi manual dan kurangnya transparansi informasi produksi bagi konsumen. Identifikasi ini menjadi dasar penentuan tujuan penelitian.

2. Tahap Pengumpulan Data

Untuk mendapatkan informasi yang akurat sebagai dasar perancangan sistem, penulis menggunakan metode pengumpulan data sebagai berikut:

- a. **Observasi (*Observation*):** Melakukan pengamatan langsung terhadap prosedur sistem yang sedang berjalan, termasuk alur administrasi dokumen *Purchase Order* (PO) fisik dan mekanisme verifikasi pembayaran manual yang diterapkan perusahaan saat ini.
- b. **Wawancara (*Interview*):** Melakukan tanya jawab langsung dengan pihak manajemen atau staf terkait di PT Marlip Indo Mandiri untuk menggali informasi lebih dalam mengenai kendala operasional serta kebutuhan fitur sistem yang diharapkan.
- c. **Studi Literatur (*Literature Review*):** Mempelajari teori-teori pendukung dari buku teks dan jurnal ilmiah terkait konsep *E-Commerce*, mekanisme *Payment Gateway* Midtrans, serta dokumentasi teknis teknologi *Framework* Next.js.

3. Tahap Pengembangan Sistem (Metode Waterfall)

Sesuai dengan diagram alur, metode pembangunan aplikasi yang digunakan adalah Model Waterfall menurut Roger S. Pressman (2020). Model ini dipilih karena kebutuhan sistem sudah terdefinisi dengan jelas dan alur pengerjaan dilakukan secara berurutan (*sekuensial*). Tahapan pengembangan meliputi:

- a. **Communication (Komunikasi & Analisis)** Tahap ini berfokus pada analisis kebutuhan perangkat lunak (*Software Requirement Analysis*). Penulis menerjemahkan hasil wawancara menjadi spesifikasi kebutuhan fungsional sistem, seperti fitur pembayaran otomatis dan pelacakan pesanan.
- b. **Planning (Perencanaan)** Tahap perencanaan teknis yang meliputi estimasi waktu pengerjaan, penjadwalan, dan pemilihan teknologi (*tech stack*). Pada tahap ini ditetapkan penggunaan *Framework* Next.js, Bahasa TypeScript, dan *Database* PostgreSQL sebagai fondasi aplikasi.
- c. **Modeling (Pemodelan)** Tahap perancangan logika dan desain sistem. Penulis membuat model visual menggunakan diagram UML (*Use Case, Activity Diagram, Dan Sequence Diagram*) untuk

menggambarakan alur sistem, merancang skema *database* (ERD), serta mendesain antarmuka pengguna (*User Interface*) agar sesuai dengan pengalaman pengguna yang baik.

- d. **Construction (Konstruksi)** Tahap penulisan kode program (*coding*). Penulis mengimplementasikan desain yang telah dibuat ke dalam baris kode menggunakan Next.js. Pada tahap ini juga dilakukan integrasi sistem dengan API pihak ketiga, yaitu Midtrans untuk pembayaran digital.
- e. **Deployment (Penyerahan & Evaluasi)** Tahap penyerahan perangkat lunak kepada pengguna (dosen pembimbing dan mitra). Karena sistem menggunakan lingkungan *Sandbox*, tahap ini berfokus pada pengujian fungsionalitas (*Black-Box Testing*) dan pengumpulan umpan balik (*feedback*) untuk memastikan sistem berjalan sesuai spesifikasi awal.

4. Tahap Kesimpulan dan Saran

Tahap akhir dari metodologi ini adalah menarik kesimpulan dari seluruh proses penelitian dan pengembangan yang telah dilakukan, serta memberikan saran-saran untuk pengembangan sistem lebih lanjut di masa depan.

1.6 Standar dan Kakas

Dalam pelaksanaan tugas akhir ini, proses pengembangan sistem didukung oleh lingkungan pengembangan yang mencakup perangkat keras (*hardware*) dan perangkat lunak (*software*) sebagai berikut:

1. Laptop Pribadi, digunakan sebagai perangkat keras utama dalam seluruh proses pengembangan aplikasi, manajemen basis data, hingga pengujian sistem.
2. Visual Studio Code, digunakan sebagai *code editor* utama yang mendukung ekosistem pengembangan modern.
3. Google Chrome / Web Browser, digunakan untuk menjalankan dan men-debug aplikasi web serta pengujian responsivitas antarmuka.
4. TypeScript, digunakan sebagai bahasa pemrograman utama untuk menjamin keamanan penulisan kode (*type safety*) dan meminimalisir kesalahan saat pengembangan.
5. Framework Next.js, digunakan sebagai kerangka kerja utama (*Fullstack Framework*) karena kemampuannya menangani *Server-Side Rendering* (SSR) untuk performa aplikasi yang optimal.
6. Library React, digunakan sebagai basis antarmuka pengguna untuk membangun komponen UI yang modular dan interaktif di dalam ekosistem Next.js.
7. Tailwind CSS, dimanfaatkan untuk proses *styling* antarmuka dengan pendekatan *utility-first* yang mempercepat pembuatan desain responsif.

8. Midtrans (Sandbox Environment), digunakan sebagai layanan gerbang pembayaran (*Payment Gateway*) untuk simulasi transaksi digital yang aman.
9. PostgreSQL dengan Neon, digunakan sebagai sistem manajemen basis data (DBMS) untuk penyimpanan data yang terstruktur.
10. Git dan GitHub, dimanfaatkan sebagai sistem kontrol versi untuk pengelolaan riwayat kode dan kolaborasi.
11. Figma / Draw.io, digunakan sebagai alat bantu perancangan desain antarmuka (*Mockup*) dan diagram alur sistem.

1.7 Sistematika Penulisan

Berikut di bawah ini merupakan sistematika penulisan yang akan digunakan dalam penyusunan tugas akhir ini:

BAB 1 PENDAHULUAN

Bab ini memberikan gambaran umum mengenai dasar pemikiran pelaksanaan tugas akhir. Pembahasan di dalamnya mencakup latar belakang masalah yang mendasari penelitian, identifikasi masalah dalam bentuk rumusan pertanyaan, tujuan yang ingin dicapai, metodologi pengembangan perangkat lunak, standar kakov pengembangan, serta sistematika penulisan laporan secara keseluruhan.

BAB 2 LANDASAN TEORI

Bab ini menguraikan berbagai teori pendukung dan konsep dasar yang digunakan sebagai fondasi dalam penelitian. Materi yang dibahas diawali dengan metodologi Waterfall, kemudian dilanjutkan dengan konsep *Electronic Commerce* (E-Commerce), mekanisme kerja *Payment Gateway* (Midtrans Snap API), teknologi *Framework* Next.js, teori basis data, serta tinjauan pustaka terhadap penelitian terdahulu.

BAB 3 SKEMA PENELITIAN

Bab ini menjelaskan kerangka kerja penelitian secara sistematis. Pembahasan dimulai dari profil objek penelitian yaitu PT Marlip Indo Mandiri, analisis masalah menggunakan diagram sebab-akibat (Fishbone Diagram), hingga kerangka pemecahan masalah yang diusulkan untuk mengatasi kendala operasional perusahaan.

BAB 4 ANALISIS DAN PERANCANGAN

Bab ini menyajikan analisis dan perancangan sistem solusi. Pembahasan mencakup analisis alur proses bisnis usulan (Business Process), analisis kebutuhan sistem berdasarkan hasil wawancara, pemodelan sistem menggunakan UML (Unified Modeling Language), perancangan basis data (ERD), serta rancangan visual antarmuka pengguna (Mockup).

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Bab ini memaparkan realisasi dari tahap perancangan ke dalam bentuk kode program (*coding*). Pembahasan meliputi konfigurasi lingkungan pengembangan, implementasi antarmuka (*frontend*) dan logika server (*backend*), integrasi API Midtrans, serta hasil pengujian sistem menggunakan metode *Black-Box Testing*.

BAB 6 KESIMPULAN DAN SARAN

Bab ini merupakan bagian penutup yang berisi kesimpulan menyeluruh dari hasil penelitian dan implementasi sistem yang telah dilakukan, serta saran-saran konstruktif untuk pengembangan sistem lebih lanjut di masa mendatang.

BAB 2

LANDASAN TEORI

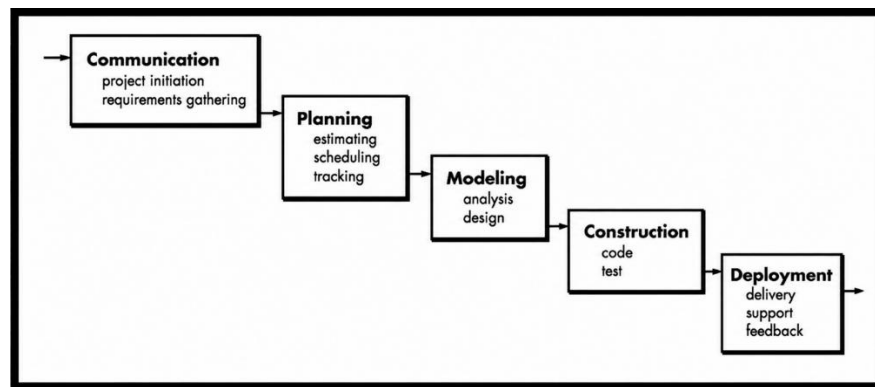
Bab ini menjelaskan kerangka teori yang digunakan sebagai landasan penelitian serta tinjauan pustaka dari penelitian terdahulu yang relevan.

2.1 Teori yang Digunakan

Bagian ini menguraikan konsep-konsep dasar, teknologi, dan metodologi yang menjadi landasan dalam pengembangan sistem pemesanan kendaraan listrik di PT Marlip Indo Mandiri.

2.1.1 Metodologi Waterfall

Metode *Waterfall* adalah model pengembangan perangkat lunak klasik yang bersifat sekuensial atau berurutan, di mana setiap fase harus diselesaikan sepenuhnya sebelum fase berikutnya dimulai. Menurut Pressman & Maxim (2020), model ini sering disebut sebagai *classic life cycle*.



Gambar 2.1 Metode Waterfall

Sumber: Pressman & Maxim (2020)

Seperti terlihat pada Gambar 2.1, tahapan metode Waterfall meliputi:

1. **Communication (Komunikasi):** Langkah inialisasi proyek yang mencakup analisis kebutuhan (*requirements gathering*) melalui diskusi mendalam dengan pengguna untuk memahami tujuan perangkat lunak.
2. **Planning (Perencanaan):** Tahap estimasi tugas teknis, penjadwalan sumber daya, dan perencanaan alur kerja proyek agar pengembangan berjalan sesuai waktu yang ditentukan.
3. **Modeling (Pemodelan):** Proses pembuatan model analisis dan desain sistem. Tujuannya adalah memahami struktur data, arsitektur perangkat lunak, dan antarmuka pengguna sebelum proses pengkodean dimulai.
4. **Construction (Konstruksi):** Proses penerjemahan desain ke dalam kode program (*code generation*) dan pengujian unit (*testing*) untuk mengungkap kesalahan pada kode.
5. **Deployment (Penyerahan):** Penyerahan perangkat lunak yang sudah jadi kepada konsumen untuk dievaluasi dan diberikan umpan balik guna penyempurnaan sistem [PRE20].

2.1.2 Konsep Rancang Bangun

Rancang bangun merupakan gabungan dari dua tahapan krusial dalam pengembangan perangkat lunak, yaitu "merancang" (desain) dan "membangun" (konstruksi). Menurut Pressman dan Maxim (2020), perancangan sistem adalah tahapan di mana spesifikasi kebutuhan, arsitektur perangkat lunak, struktur basis data, dan antarmuka didefinisikan secara konseptual [PRE20]. Tujuan utama dari perancangan ini adalah untuk memberikan gambaran yang jelas kepada pemrogram (*programmer*) sebelum proses penulisan kode dimulai.

Sedangkan tahap pembangunan atau konstruksi adalah proses menerjemahkan hasil rancangan tersebut ke dalam bentuk baris kode (*coding*) menggunakan bahasa pemrograman tertentu hingga menjadi sebuah perangkat lunak yang fungsional. Secara keseluruhan, rancang bangun adalah serangkaian proses rekayasa perangkat lunak yang sistematis untuk menghasilkan sebuah sistem informasi terkomputerisasi yang mampu memecahkan masalah operasional pada suatu organisasi.

2.1.3 Kendaraan Listrik (*Electric Vehicle*)

Kendaraan bermotor listrik atau *Electric Vehicle* (EV) adalah kendaraan yang digerakkan seluruhnya atau sebagian oleh motor listrik dan mendapatkan pasokan energi utamanya dari baterai yang dapat diisi ulang (*rechargeable*). Berbeda dengan kendaraan berbahan bakar fosil konvensional, kendaraan listrik menawarkan efisiensi energi yang jauh lebih tinggi dan bebas dari emisi gas buang karbon, sehingga menjadikannya solusi utama untuk mewujudkan ekosistem transportasi hijau yang ramah lingkungan.

Di Indonesia, transisi menuju ekosistem kendaraan listrik sangat didukung oleh pemerintah guna mengurangi ketergantungan pada energi fosil. Hal ini tertuang dalam Peraturan Presiden Nomor 79 Tahun 2023 tentang Percepatan Program Kendaraan Bermotor Listrik Berbasis Baterai [IND23]. Dalam industri perakitan lokal, kendaraan listrik tidak hanya diproduksi untuk angkutan penumpang umum, melainkan juga dirancang sebagai *Special Purpose Vehicle* (kendaraan serbaguna) yang spesifikasinya dapat disesuaikan (kustomisasi) untuk kebutuhan patroli, rumah sakit, maupun operasional khusus.

2.1.4 Konsep Produk (Barang) dalam Sistem *Made by Order*

Produk atau barang, menurut Albrecht, Green, dan Hoffman (2023), adalah segala sesuatu yang dapat ditawarkan ke pasar untuk diperhatikan, dimiliki, digunakan, atau dikonsumsi yang dapat memuaskan keinginan atau kebutuhan konsumen [ALB23]. Dalam ruang lingkup *e-commerce*, produk fisik harus dideskripsikan secara rinci melalui katalog digital yang memuat informasi mengenai model, spesifikasi teknis, dan harga dasar agar konsumen dapat membuat keputusan pembelian yang tepat.

Dalam industri manufaktur, pengelolaan penyediaan barang sering kali menggunakan pendekatan *Make to Order* atau *Made by Order*. Menurut Rafiei, Rabbani, dan Kokabi (2014), sistem *Made by Order* adalah strategi produksi di mana produk baru akan dibuat dan dirakit setelah perusahaan menerima pesanan resmi dan terkonfirmasi dari pelanggan [RAF14]. Model bisnis ini tidak memelihara status barang *ready stock* di gudang untuk produk jadi, melainkan berfokus pada kustomisasi pesanan, sehingga sangat memerlukan sistem pencatatan *Purchase Order* (PO) yang akurat.

2.1.5 Sistem Pemesanan

Pemesanan (*Order*) adalah suatu aktivitas transaksi manajerial di mana pihak pembeli mengajukan permintaan penyediaan barang atau jasa kepada pihak penjual sesuai dengan spesifikasi, harga, dan waktu pengiriman yang telah disepakati bersama. Dalam prosedur konvensional, pesanan umumnya didokumentasikan menggunakan formulir fisik berupa *Purchase Order* (PO) yang berfungsi sebagai dokumen administratif dan bukti kesepakatan antara kedua belah pihak sebelum proses administrasi lanjutan dilakukan.

Seiring dengan perkembangan teknologi internet, sistem pemesanan konvensional bertransformasi menjadi sistem pemesanan berbasis digital atau *E-Ordering*. Raisinghani (2025) menjelaskan bahwa sistem pemesanan digital memungkinkan otomatisasi alur pertukaran informasi bisnis [RAI25]. Melalui website, seluruh prosedur mulai dari eksplorasi katalog produk, pengisian formulir kustomisasi, hingga penerbitan dan persetujuan dokumen *Purchase Order* dapat difasilitasi dalam satu platform terpusat, yang secara efektif mampu meminimalisir *human error* dan manipulasi data.

2.1.6 Purchase Order dan Invoice Digital

Menurut Kalsid dan Septanto (2023), *Purchase Order* (PO) digital bertindak sebagai dokumen resmi yang diterbitkan oleh sistem atas pengajuan komitmen pembelian dari konsumen, yang memuat rincian pesanan dan kebutuhan khusus, kuantitas, harga, serta klausul syarat ketentuan kesepakatan. Dokumen ini berfungsi sebagai ikatan dokumen administratif dan bukti kesepakatan elektronik sebelum proses produksi barang dimulai di tingkat pabrik [KAL23].

Di sisi lain, *Invoice* digital adalah dokumen penagihan elektronik yang diterbitkan oleh sistem sebagai representasi kewajiban pembayaran yang harus diselesaikan oleh pembeli. Berdasarkan studi kasus tata kelola administrasi oleh Rochman dkk. (2023), penerapan dokumen transaksi digital mampu menekan kebutuhan penggunaan dokumen fisik, mengurangi risiko kehilangan arsip, serta mempercepat proses pemeriksaan data transaksi [ROC23]. Pada implementasi penelitian ini, dokumen *Purchase Order* dan *Invoice* ditampilkan dalam format siap cetak, sehingga pengguna dapat mencetak atau menyimpannya sebagai PDF melalui fungsi cetak pada peramban.

2.1.7 Konsep E-Commerce (*Electronic Commerce*)

Electronic Commerce atau *E-Commerce* didefinisikan sebagai penggunaan internet, *World Wide Web* (Web), dan aplikasi seluler untuk melakukan transaksi bisnis. Berbeda dengan *E-Business* yang mencakup seluruh operasional digital perusahaan, *E-Commerce* secara spesifik berfokus pada transaksi komersial yang melibatkan pertukaran nilai (uang) melintasi batas-batas organisasi atau individu sebagai imbalan atas produk dan layanan. Menurut Laudon & Traver (2016), *E-Commerce* telah mengubah struktur pasar tradisional dengan menghilangkan asimetri informasi dan memungkinkan personalisasi produk secara massal [LAU16].

Dalam penerapannya, *E-Commerce* dibagi menjadi beberapa model bisnis, di mana salah satu yang paling umum adalah *Business-to-Consumer* (B2C). Model ini melibatkan interaksi langsung antara perusahaan penyedia barang dengan konsumen akhir melalui platform digital. Turban et al. (2018) menjelaskan bahwa keunggulan utama model B2C terletak pada jangkauan pasar yang global (*global reach*) dan ketersediaan layanan 24 jam (*ubiquity*), yang memungkinkan konsumen untuk mengakses katalog produk, melakukan pemesanan, dan menyelesaikan pembayaran tanpa batasan waktu dan lokasi fisik [TUR18].

2.1.8 Payment Gateway (Midtrans)



Gambar 2.2 Logo Payment Gateway (Midtrans)

Payment Gateway merupakan infrastruktur teknologi yang berperan sebagai perantara dalam memfasilitasi transaksi pembayaran digital antara aplikasi pedagang (*merchant*) dengan lembaga keuangan. Midtrans adalah salah satu penyedia layanan gerbang pembayaran terkemuka di Indonesia yang menyediakan API (*Application Programming Interface*) untuk integrasi pembayaran *online*. Midtrans bertindak sebagai jembatan yang aman, mengenkripsi data transaksi sensitif, dan meneruskannya ke bank atau penerbit *e-money* untuk divalidasi [MID26].

Layanan Midtrans Snap API yang digunakan dalam penelitian ini memungkinkan aplikasi *merchant* untuk menampilkan halaman pembayaran *pop-up* yang aman tanpa harus mengarahkan pengguna keluar dari situs web utama. Midtrans mendukung berbagai metode pembayaran, mulai dari transfer bank, kartu kredit, hingga dompet digital (*e-wallet*). Keunggulan lainnya adalah fitur notifikasi

otomatis (*webhook*) yang memberi tahu sistem *merchant* secara *real-time* mengenai status pembayaran (berhasil, gagal, atau tertunda), sehingga proses verifikasi dapat dilakukan secara otomatis tanpa campur tangan manusia [FAT23].

2.1.9 Webhook

Webhook merupakan mekanisme pengiriman notifikasi otomatis dari layanan eksternal ke sistem aplikasi ketika terjadi perubahan status transaksi atau peristiwa tertentu. Berbeda dengan metode *polling*, webhook tidak mengharuskan aplikasi melakukan pengecekan status secara berulang. Pada mekanisme webhook, server penyedia layanan akan mengirimkan data secara langsung ke *endpoint* sistem yang telah didaftarkan ketika suatu peristiwa terjadi.

Dalam implementasi sistem transaksi digital dan gerbang pembayaran, webhook memegang peranan penting dalam menjaga sinkronisasi status pembayaran. Berdasarkan penelitian teknis dari Fatman dkk. (2023), integrasi webhook mampu mengotomatisasi verifikasi pembayaran dengan cara menerima data konfirmasi transaksi dari server *payment gateway* secara langsung. Sistem penerima webhook kemudian memvalidasi *signature key* untuk memastikan integritas data, lalu memperbarui status pembayaran dan status pesanan di dalam basis data lokal tanpa memerlukan intervensi manual dari staf administrasi keuangan [FAT23].

2.1.10 Aplikasi Berbasis Web

Aplikasi berbasis web adalah perangkat lunak yang dapat diakses menggunakan peramban web (*web browser*) melalui jaringan internet atau intranet. Berbeda dengan aplikasi desktop yang perlu diinstal pada setiap perangkat komputer klien, aplikasi web disimpan pada sebuah *web server* terpusat dan dikirimkan ke perangkat pengguna saat diminta. Keunggulan utama aplikasi berbasis web adalah aksesibilitas lintas platform (*cross-platform*), yang memungkinkan pengguna untuk mengakses sistem dari berbagai jenis perangkat dan sistem operasi tanpa perlu melakukan instalasi tambahan atau pembaruan manual di sisi klien [PRE20].

Arsitektur aplikasi web modern umumnya mengadopsi pola *Client-Server*, di mana peramban bertindak sebagai klien yang mengirimkan permintaan HTTP, dan server memproses permintaan tersebut serta mengirimkan kembali respon berupa halaman HTML, CSS, dan JavaScript. Menurut Pressman (2020), evolusi teknologi web telah memungkinkan aplikasi web memiliki fungsionalitas dan interaktivitas yang setara dengan aplikasi desktop, didukung oleh teknologi seperti AJAX dan *Single Page Application* (SPA) yang meningkatkan responsivitas antarmuka pengguna [PRE20].

2.1.11 Autentikasi dan Otorisasi

Menurut Jones, Bradley, dan Sakimura (2015), autentikasi adalah proses pembuktian keaslian identitas dari seorang pengguna yang mencoba masuk ke dalam sistem, yang umumnya diwujudkan melalui pencocokan kombinasi *email* dan kata sandi yang telah di-hash. Setelah identitas pengguna dinyatakan valid, sistem akan menerbitkan sebuah token sesi, seperti *JSON Web Token* (JWT), sebagai bukti pengenalan aktif selama berselancar di dalam aplikasi [JON15].

Sementara itu, otorisasi adalah proses penentuan hak akses atau izin operasional yang diberikan kepada pengguna yang telah lolos tahap autentikasi tersebut. Berdasarkan teori kontrol akses dari Sandhu, Ferraiolo, dan Kuhn (2000), pendekatan yang paling umum digunakan dalam aplikasi transaksional adalah *Role-Based Access Control* (RBAC), di mana hak akses dikelompokkan berdasarkan peran spesifik pengguna di dalam organisasi. Implementasi pengetatan otorisasi ini memastikan bahwa aktor seperti Konsumen hanya dapat mengakses fitur pemesanan dan pelacakan, sedangkan fungsionalitas manajemen data produk dan pembaruan tahapan produksi hanya dapat dieksekusi oleh aktor Admin dan Kepala Produksi yang sah [SAN00].

2.1.12 Framework Next.js



Gambar 2.3 Logo Framework Next.js

Next.js adalah kerangka kerja (*framework*) pengembangan web berbasis React yang digunakan untuk membangun aplikasi web modern dengan dukungan *server-side rendering*, *static site generation*, *routing* berbasis sistem berkas, dan integrasi *server-side* melalui Route Handler. Berdasarkan dokumentasi resmi Vercel (2026), Next.js menyediakan fitur *Server-Side Rendering* (SSR) dan *Static Site Generation* (SSG) untuk meningkatkan performa halaman serta mendukung optimasi mesin pencari [VER26].

Selain keunggulan performa, Next.js juga menyediakan pengalaman pengembang (*Developer Experience*) yang lengkap melalui fitur *routing* berbasis sistem berkas (*file-system based routing*), optimasi gambar, dan dukungan App Router. Pada App Router, pengembang dapat menggunakan Route

Handler untuk menangani permintaan HTTP pada sisi server. Arsitektur Next.js memungkinkan pengembang membangun aplikasi *fullstack* dalam satu struktur proyek yang terintegrasi [VER26].

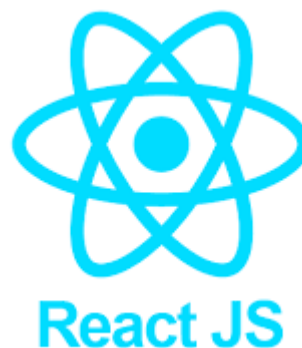
2.1.13 Route Handler dan Server Action pada Next.js App Router

Route Handler pada Next.js App Router merupakan fitur bawaan dalam kerangka kerja Next.js yang memungkinkan pengembang membangun *endpoint* API secara langsung di dalam satu proyek aplikasi web tanpa memerlukan server *backend* terpisah. Fitur ini memanfaatkan konsep *file-system based routing*, yaitu mekanisme pemetaan rute berdasarkan struktur folder dan berkas. Pada App Router, Route Handler diletakkan di dalam direktori `app/api` dengan menggunakan berkas `route.ts` untuk menangani permintaan HTTP seperti GET, POST, PATCH, dan DELETE [VER26].

Dalam konteks aplikasi *fullstack*, Route Handler berperan sebagai penghubung antara antarmuka pengguna, logika bisnis, basis data, serta layanan eksternal. Route Handler dapat digunakan untuk memproses data dari sisi *frontend*, menjalankan operasi basis data melalui ORM, memvalidasi autentikasi dan otorisasi pengguna, serta menerima kiriman data asinkron dari pihak ketiga seperti webhook pembayaran. Selain itu, penggunaan Route Handler juga membantu menjaga keamanan kunci akses API karena proses yang bersifat sensitif dijalankan pada sisi server dan tidak diekspos langsung ke peramban pengguna [VER26].

Server Action pada Next.js App Router merupakan mekanisme yang memungkinkan fungsi tertentu dijalankan langsung pada sisi server dari komponen atau formulir aplikasi. Fitur ini dapat digunakan untuk menangani proses yang membutuhkan keamanan lebih, seperti validasi data, autentikasi, atau perubahan data pada basis data. Dengan adanya Route Handler dan Server Action, pengembangan aplikasi *fullstack* pada Next.js dapat dilakukan secara lebih terintegrasi dalam satu struktur proyek.

2.1.14 React



Gambar 2.4 Logo React

React adalah pustaka (*library*) JavaScript yang bersifat *open-source* dan digunakan untuk membangun antarmuka pengguna (*User Interface*), khususnya untuk aplikasi satu halaman (*Single Page*

Application). Dikembangkan oleh Facebook (Meta), React memperkenalkan konsep *Virtual DOM* yang memungkinkan pembaruan tampilan dilakukan secara efisien hanya pada komponen yang berubah, tanpa perlu memuat ulang seluruh halaman. Pendekatan ini secara drastis meningkatkan performa aplikasi dibandingkan manipulasi DOM konvensional yang cenderung lambat [FRE19].

Fitur utama React adalah arsitektur berbasis komponen (*component-based*), di mana antarmuka pengguna dipecah menjadi bagian-bagian kecil yang mandiri dan dapat digunakan kembali (*reusable*). Komponen-komponen ini mengelola *state* (data internal) mereka sendiri, dan dapat digabungkan untuk membentuk antarmuka yang kompleks. React juga mendukung penulisan kode menggunakan JSX (JavaScript XML), sebuah ekstensi sintaks yang memungkinkan penulisan struktur HTML langsung di dalam kode JavaScript, sehingga memudahkan pengembang dalam memvisualisasikan struktur UI [FRE19].

2.1.15 TypeScript



Gambar 2.5 Logo TypeScript

TypeScript adalah bahasa pemrograman *open-source* yang dikembangkan oleh Microsoft sebagai *superset* dari JavaScript. TypeScript menambahkan fitur *static typing* (pengetikan statis) opsional ke dalam sintaks JavaScript, yang memungkinkan pengembang untuk mendefinisikan tipe data variabel secara eksplisit. Fitur ini sangat berguna dalam pengembangan aplikasi skala besar karena dapat mendeteksi kesalahan tipe (*type errors*) pada saat proses kompilasi (*compile-time*), bukan pada saat aplikasi dijalankan (*runtime*), sehingga meningkatkan keamanan dan stabilitas kode program secara signifikan [MIC24].

Selain pengetikan statis, TypeScript juga menyediakan fitur-fitur pemrograman berorientasi objek (*Object-Oriented Programming*) modern seperti *class*, *interface*, dan *module*. Kode TypeScript harus dikompilasi (*transpilasi*) menjadi JavaScript standar agar dapat dijalankan oleh peramban web atau lingkungan Node.js. Dengan dukungan alat bantu pengembangan yang kuat, seperti *autocompletion* dan

navigasi kode di editor modern, TypeScript telah menjadi standar industri untuk pengembangan aplikasi web yang kompleks dan membutuhkan pemeliharaan jangka panjang [MIC24].

2.1.16 Vercel



Gambar 2.6 Logo Vercel

Vercel adalah platform *cloud* yang dirancang khusus untuk men-*deploy* aplikasi *frontend*, khususnya yang dibangun menggunakan *framework* Next.js. Vercel menawarkan infrastruktur *serverless* yang memungkinkan aplikasi web untuk disebarluaskan secara global melalui jaringan pengiriman konten (*Edge Network*) dengan latensi rendah. Platform ini secara otomatis mengoptimalkan aset statis dan fungsi server, sehingga aplikasi dapat memiliki performa yang baik dan waktu muat yang cepat bagi pengguna [VER26].

Salah satu fitur unggulan Vercel adalah integrasi berkelanjutan (*Continuous Integration*) dengan penyedia repositori Git seperti GitHub. Setiap kali pengembang melakukan perubahan kode (*push*) ke repositori, Vercel dapat membangun (*build*) dan menyebarkan versi aplikasi secara otomatis. Hal ini memudahkan proses pengujian dan penerapan aplikasi web ke lingkungan *deployment* [VER26].

2.1.17 Prisma ORM

Berdasarkan dokumentasi Prisma (2026), *Prisma Object-Relational Mapping* (ORM) merupakan perkakas (*tooling*) basis data modern yang berfungsi sebagai jembatan antara kode aplikasi pemrograman dengan basis data relasional. Berbeda dengan ORM tradisional yang sering kali kompleks dalam konfigurasi, Prisma menyediakan pendekatan deskriptif melalui berkas skema (*Prisma Schema*) yang mendefinisikan model data secara deklaratif. Keunggulan utama dari Prisma terletak pada fitur *type-safety* otomatis yang terintegrasi penuh dengan bahasa TypeScript, sehingga kesalahan penulisan kueri dapat terdeteksi langsung sejak proses pengembangan aplikasi (*compile-time*) sebelum aplikasi dijalankan [PRI26].

Dalam arsitektur pengembangan web modern, Prisma bertindak sebagai penghubung antara aplikasi dan basis data melalui Prisma Client. Prisma Client menyediakan fungsi kueri yang terstruktur

untuk melakukan operasi *create*, *read*, *update*, dan *delete* (CRUD) terhadap data. Selain itu, Prisma juga mendukung pengelolaan perubahan struktur basis data melalui Prisma Migrate, sehingga perubahan skema dapat dikelola secara lebih terarah dalam proses pengembangan aplikasi [PRI26].

2.1.18 PostgreSQL



Gambar 2.7 Logo PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional (*Relational Database Management System* atau RDBMS) tingkat lanjut yang bersifat *open-source*. PostgreSQL dikenal karena keandalannya, integritas data yang kuat, dan kepatuhannya terhadap standar SQL. Basis data ini mendukung transaksi ACID (*Atomicity, Consistency, Isolation, Durability*) yang berperan dalam menjaga konsistensi data pada sistem transaksional [POS24].

PostgreSQL mengimplementasikan model basis data relasional yang diperkenalkan oleh Codd (1970), dengan struktur tabel dan relasi antartabel melalui kunci utama dan kunci tamu [COD70]. Selain itu, PostgreSQL mendukung tipe data JSONB untuk menyimpan data semi-terstruktur [POS24].

2.1.19 Neon (Serverless PostgreSQL)

Neon merupakan platform manajemen basis data PostgreSQL berbasis awan (*cloud-native*) yang mengusung arsitektur *serverless*. Inovasi utama dari Neon terletak pada pemisahan yang jelas antara fungsi komputasi (*compute*) dan fungsi penyimpanan data (*storage*). Arsitektur ini memungkinkan basis data untuk melakukan skala otomatis (*autoscaling*) secara dinamis sesuai dengan volume lalu lintas data yang masuk, bahkan memiliki kemampuan untuk menurunkan kapasitas komputasi hingga titik nol (*scale-to-zero*) saat sistem sedang tidak menerima permintaan transaksi untuk mengoptimalkan efisiensi sumber daya [NEO26].

Dalam ekosistem aplikasi web modern, Neon menyediakan solusi krusial terhadap kendala batasan koneksi yang sering terjadi pada lingkungan *serverless* konvensional. Berdasarkan analisis dari Berdasarkan dokumentasi Neon (2025), platform Neon dilengkapi dengan fitur *database branching* yang memungkinkan pengembang untuk membuat salinan basis data instan (*instant clones*) untuk keperluan pengujian fitur baru tanpa mengganggu basis data produksi utama. Integrasi koneksi yang cepat dan dukungan penuh terhadap protokol PostgreSQL standar menjadikan platform ini sebagai fondasi penyimpanan data transaksional yang andal bagi aplikasi berbasis *edge computing* [NEO26].

2.1.20 Tailwind CSS



Gambar 2.8 Logo Tailwind CSS

Tailwind CSS adalah kerangka kerja CSS yang mengutamakan utilitas (*utility-first framework*) untuk membangun desain antarmuka kustom dengan cepat. Berbeda dengan kerangka kerja tradisional seperti Bootstrap yang menyediakan komponen siap pakai, Tailwind menyediakan kumpulan kelas utilitas tingkat rendah (*low-level utility classes*) yang dapat dikombinasikan langsung di dalam elemen HTML. Pendekatan ini memungkinkan pengembang memiliki kontrol penuh terhadap desain tanpa harus menulis kode CSS kustom yang berlebihan, serta menghindari masalah spesifisitas gaya yang sering terjadi pada proyek besar.

Keunggulan lain dari Tailwind CSS adalah dukungannya terhadap desain responsif yang intuitif menggunakan *mobile-first approach*. Pengembang dapat dengan mudah menentukan gaya yang berbeda untuk berbagai ukuran layar hanya dengan menambahkan prefiks tertentu pada kelas utilitas. Selain itu, Tailwind memiliki fitur *purging* (pembersihan) otomatis pada saat produksi, yang menghapus semua kelas CSS yang tidak digunakan dalam proyek, sehingga menghasilkan ukuran berkas CSS yang sangat kecil dan meningkatkan kecepatan muat halaman web [TAI24].

2.1.21 Unified Modeling Language (UML)

Menurut Pressman dan Maxim (2020), *Unified Modeling Language* (UML) merupakan bahasa pemodelan standar yang menggunakan representasi grafis untuk melakukan spesifikasi, visualisasi, konstruksi, dan pendokumentasian artefak-artefak dari sebuah sistem perangkat lunak berbasis objek. UML menyediakan sekumpulan diagram standar yang memungkinkan tim analis sistem dan pengembang perangkat lunak untuk berkomunikasi menggunakan sudut pandang desain arsitektur yang seragam, sehingga kompleksitas alur kerja logika aplikasi dapat dipahami dengan mudah oleh seluruh pemangku kepentingan proyek [PRE20].

Dalam fase pemodelan siklus pengembangan perangkat lunak, diagram UML dibagi menjadi beberapa sudut pandang struktural dan perilaku sistem. Menurut Pressman dan Maxim (2020), diagram perilaku seperti *Use Case Diagram* berfungsi mendefinisikan batas sistem dan interaksi aktor, sedangkan *Activity Diagram* memetakan aliran aktivitas bisnis secara prosedural. Aliran komunikasi antar-objek yang terikat urutan waktu secara kronologis kemudian dimodelkan secara mendalam melalui *Sequence*

Diagram, sehingga cetak biru (*blueprint*) dari aplikasi web yang akan dibangun memiliki fondasi logika yang matang dan terstruktur sebelum masuk ke tahap penulisan kode program [PRE20].

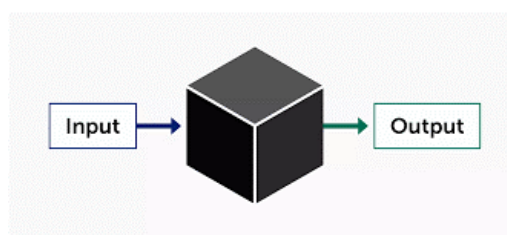
2.1.22 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan model konseptual yang digunakan untuk menggambarkan struktur logis, arsitektur data, dan hubungan keterkaitan antarentitas di dalam basis data relasional. ERD membantu perancang sistem dalam mengidentifikasi entitas, atribut, kunci utama, kunci tamu, serta hubungan antar data sebelum rancangan tersebut diimplementasikan ke dalam basis data fisik [WAT14].

Dalam menyusun struktur basis data yang optimal, ERD menetapkan hubungan antarentitas melalui representasi kardinalitas atau derajat relasi, seperti hubungan satu-ke-banyak (*one-to-many*) dan banyak-ke-banyak (*many-to-many*). Pemodelan ini menjadi dasar dalam pembentukan tabel-tabel pada *Relational Database Management System* (RDBMS), sehingga integritas dan konsistensi data dapat dijaga melalui penggunaan *Primary Key* dan *Foreign Key* [WAT14].

2.1.23 Pengujian Black-Box (*Black-Box Testing*)

Black-Box Testing adalah metode pengujian perangkat lunak yang berfokus pada fungsionalitas aplikasi tanpa melihat struktur kode internal atau logika pemrogramannya. Penguji memperlakukan sistem sebagai "kotak hitam" dan hanya memvalidasi apakah *input* yang diberikan menghasilkan *output* yang diharapkan. Menurut Nidhra & Dondeti (2012), teknik ini efektif untuk menemukan kesalahan pada fungsi yang hilang atau tidak benar, kesalahan antarmuka (*interface errors*), serta kesalahan kinerja pada saat inisialisasi maupun terminasi program [NID12].



Gambar 2.9 Ilustrasi Pengujian Black-Box

Dalam praktiknya, pengujian *Black-Box* sering menggunakan teknik *Equivalence Partitioning* dan *Boundary Value Analysis* untuk memastikan seluruh skenario penggunaan telah teruji. Penelitian oleh Mustaqbal et al. (2015) menunjukkan bahwa pengujian ini sangat relevan untuk menguji aplikasi berbasis web dari sudut pandang pengguna akhir (*end-user perspective*). Dengan metode ini, pengembang dapat memastikan bahwa alur bisnis aplikasi, seperti proses *checkout* pembayaran atau pelacakan pesanan, berjalan lancar dan mudah dipahami oleh pengguna awam [MUS15].

2.2 Penelitian Terdahulu

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Hasil Penelitian	Alasan Digunakan
1	Arribe, E., Jayema, A., & Putra, F. W. (2023)Jurnal Ilmiah Komputer Grafis (Pixel)	"Perancangan Sistem Informasi Penjualan Motor Listrik PT. Talabu"Penelitian ini membangun sistem penjualan motor listrik berbasis web untuk mengatasi masalah manajemen inventaris dan transaksi manual. Sistem menyediakan fitur katalog produk dan pencatatan transaksi yang terintegrasi database.	Menjadi referensi utama karena memiliki objek penelitian yang sejenis (Kendaraan Listrik), sehingga alur bisnis penjualannya sangat relevan untuk diadopsi. [ARR23]
2	Fatman, Y., Nafisah, N., & Pambudi, P. B. (2023)Jurnal KomtekInfo	"Implementasi Payment Gateway dengan Menggunakan Midtrans pada Website UMKM Geberco" Penelitian ini mengimplementasikan Midtrans API untuk mengotomatisasi verifikasi pembayaran. Hasilnya menunjukkan peningkatan efisiensi transaksi dan keamanan data pembayaran dibandingkan metode transfer manual.	Digunakan sebagai acuan teknis dalam integrasi Midtrans Snap API, khususnya untuk mekanisme verifikasi pembayaran otomatis yang menjadi fitur inti tugas akhir ini. [FAT23]
3	Rochman, F. N., dkk. (2023) Jurnal Minfo Polgan	"Perancangan Sistem Penjualan Mobil Berbasis Web Pada Dealer Dengan Menggunakan Metode Waterfall" Penelitian ini merancang sistem penjualan kendaraan berbasis web untuk memudahkan dealer dalam mengelola data master, purchase order (PO), dan transaksi secara online untuk memperluas jangkauan pembeli.	Memberikan referensi terkait alur perancangan sistem penjualan kendaraan dan manajemen pembuatan Purchase Order (PO) menggunakan pendekatan web dan metode Waterfall. [ROC23]
4	Kesuma, W., & Ramadhan, M. I. (2025) Jurnal Sistem Informasi Komputer (SIKOM)	"Perancangan Aplikasi Tracking Proses Produksi pada PT Makmur Bintang Plastindo Berbasis Web". Penelitian ini mengembangkan aplikasi pelacakan produksi berbasis web untuk meningkatkan efisiensi, mengurangi kesalahan pencatatan, dan memperbaiki koordinasi antarbagian.	Menjadi dasar pengembangan fitur Tracking Tahapan Produksi agar Konsumen dapat memantau tahapan perakitan kendaraan secara visual berdasarkan pembaruan Kepala Produksi, bukan secara real-time otomatis. [KES25]
5	Gibran, C., Dewi, A. R., & Hadinata, E. (2024)Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)	"Implementasi Framework Laravel untuk Pengembangan Website Penjualan Ayam Potong dengan Pemanfaatan Midtrans Menggunakan Metode FAST."	Sebagai pembanding penggunaan teknologi framework. Penelitian ini menggunakan Laravel, sedangkan tugas akhir ini menggunakan Next.js untuk performa yang lebih baik (Server-Side Rendering). [GIB24]

BAB 3

SKEMA PENELITIAN

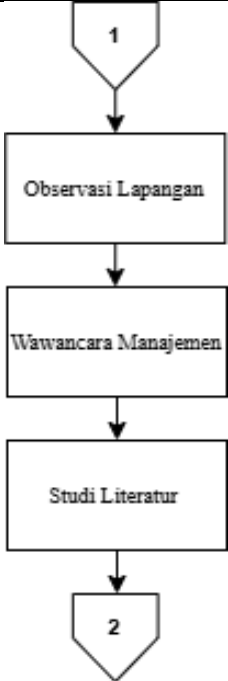
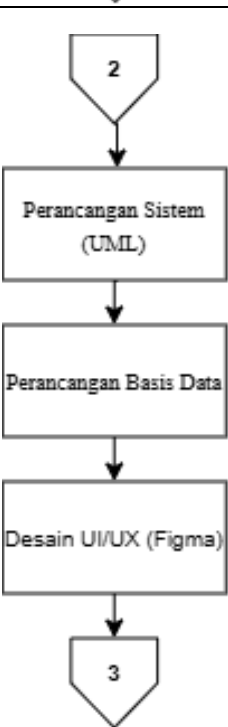
Bab ini berisi penjelasan alur penyelesaian tugas akhir, perumusan masalah, kerangka pemikiran teoretis, dan profil tempat penelitian. Skema penelitian berfungsi sebagai panduan awal bagi pembaca untuk memahami tujuan, ruang lingkup, dan landasan teoretis penelitian yang akan dilakukan. Dengan menyediakan informasi yang jelas mengenai alur penyelesaian tugas akhir, perumusan masalah, kerangka pemikiran teoretis, dan profil tempat penelitian, skema penelitian membantu pembaca untuk memahami konteks dan relevansi penelitian secara lebih baik.

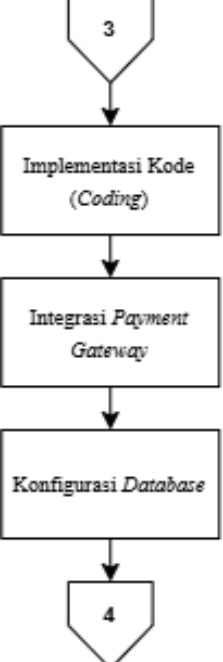
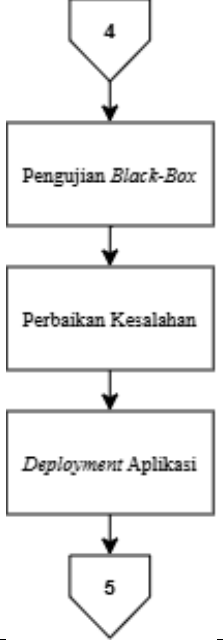
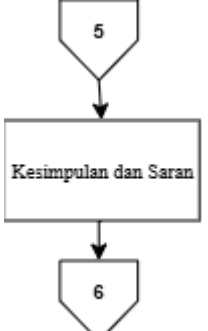
3.1 Alur Penyelesaian Tugas Akhir

Alur penyelesaian tugas akhir ini dilakukan berdasarkan diagram alur penyelesaian yang sistematis. Berikut ini merupakan tabel alur penyelesaian tugas akhir:

Tabel 3.1 Alur Penyelesaian Tugas Akhir

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 1: Menentukan topik tugas akhir, mengidentifikasi masalah tugas akhir, dan menentukan tujuan serta lingkup tugas akhir. Hasil: Dari proses penentuan topik tugas akhir didapatkan hasil mengenai topik sistem pemesanan kendaraan listrik. Selain itu, dari proses identifikasi masalah dapat mengetahui kendala penjualan manual (PO fisik) yang sedang terjadi dan menganalisis solusi integrasi <i>Payment Gateway</i> yang akan diterapkan. Setelah itu diketahui tujuan dan lingkup dari tugas akhir yang akan dilaksanakan. Kontribusi: Berguna untuk menentukan lingkup tugas akhir. Teori yang digunakan berguna untuk mengetahui konsep <i>E-Commerce</i> dan <i>Payment Gateway</i> yang dibutuhkan untuk menyelesaikan tugas akhir.	<pre> graph TD A[Menentukan Topik Tugas Akhir] --> B[Latar Belakang Masalah] B --> C[Identifikasi Masalah] C --> D[Tujuan dan Manfaat Tugas Akhir] D --> E[Lingkup Tugas Akhir] E --> F{1} </pre>	1. Profil PT Marlip Indo Mandiri 2. Teori Sistem Informasi 3. Metodologi Penelitian
Tahap 2: Pengumpulan data. Hasil: Memahami tentang teori yang akan digunakan dalam pembuatan website <i>e-commerce</i>, data spesifikasi kendaraan, serta dokumentasi teknis API Midtrans. Kontribusi:		1. Dokumentasi Next.js 2. Dokumentasi Midtrans 3. Jurnal Terkait

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Berguna untuk fondasi pembuatan database dan alur logika pembayaran website.	 <pre> graph TD 1{{1}} --> A[Observasi Lapangan] A --> B[Wawancara Manajemen] B --> C[Studi Literatur] C --> 2{{2}} </pre>	
Tahap 3: Pra-Production (Perancangan). Hasil: Gambaran atau <i>mockup</i> desain antarmuka website, diagram UML (<i>Use Case</i>, <i>Activity</i>), ERD, dan arsitektur sistem. Kontribusi: Berguna untuk panduan pembuatan tampilan <i>user interface</i> (UI) dan struktur <i>database</i> aplikasi.	 <pre> graph TD 2{{2}} --> D[Perancangan Sistem (UML)] D --> E[Perancangan Basis Data] E --> F[Desain UI/UX (Figma)] F --> 3{{3}} </pre>	1. Metode Waterfall (Pressman) 2. Teori UML & ERD 3. Konsep UI/UX Design
Tahap 4: Production (Pembangunan). Hasil: <i>Prototype</i> website pemesanan kendaraan listrik yang sudah selesai atau sudah jadi secara fungsional (dapat melakukan transaksi). Kontribusi: Menghasilkan produk perangkat lunak sebagai solusi utama tugas akhir.		1. Bahasa Pemrograman TypeScript 2. Library React & Tailwind 3. Database PostgreSQL

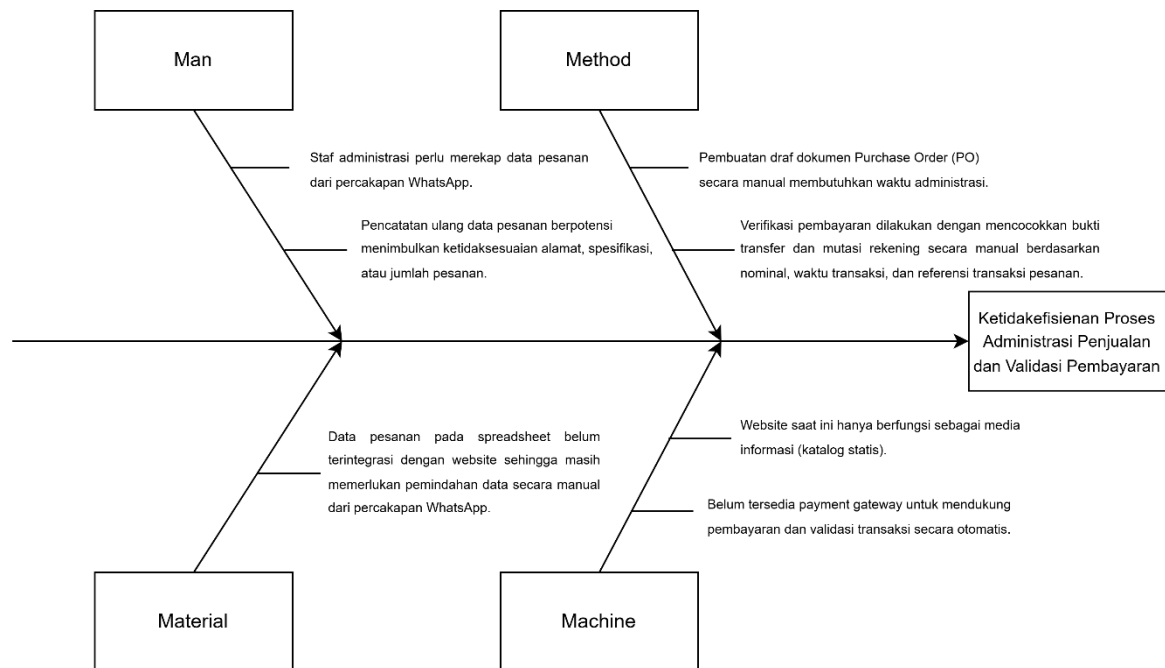
Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
	 <pre> graph TD 3{{3}} --> A[Implementasi Kode
(Coding)] A --> B[Integrasi Payment
Gateway] B --> C[Konfigurasi Database] C --> 4{{4}} </pre>	
Tahap 5: Deploy dan Pengujian. Hasil: <i>Prototype</i> website yang sudah teruji bebas <i>bug</i> (<i>Black-Box Testing</i>) dan berhasil di-deploy ke server publik (Vercel) agar dapat diakses. Kontribusi: Memastikan sistem berjalan lancar saat digunakan oleh pengguna (PT Marlip).	 <pre> graph TD 4{{4}} --> A[Pengujian Black-Box] A --> B[Perbaiki Kesalahan] B --> C[Deployment Aplikasi] C --> 5{{5}} </pre>	1. Teori <i>Black-Box Testing</i> 2. Dokumentasi Vercel
Tahap 6: Simpulan. Hasil: Kesimpulan dan saran. Kontribusi: Kesimpulan dan saran berdasarkan hasil yang telah dicapai dalam pembuatan website pemesanan kendaraan listrik.	 <pre> graph TD 5{{5}} --> A[Kesimpulan dan Saran] A --> 6{{6}} </pre>	1. Evaluasi Tujuan Awal

3.2 Perumusan Masalah

Perumusan masalah dilakukan untuk mengidentifikasi akar penyebab kendala yang terjadi pada sistem berjalan serta merumuskan solusi yang tepat. Penulis menggunakan diagram sebab-akibat (*Fishbone Diagram*) untuk memetakan masalah tersebut.

3.2.1 Analisis Masalah

Berdasarkan hasil observasi dan wawancara dengan pihak manajemen PT Marlip Indo Mandiri, ditemukan masalah utama yaitu "Ketidakefisienan Proses Administrasi Penjualan dan Validasi Pembayaran". Untuk mengetahui faktor penyebabnya, dilakukan analisis menggunakan diagram *Fishbone* sebagai berikut:



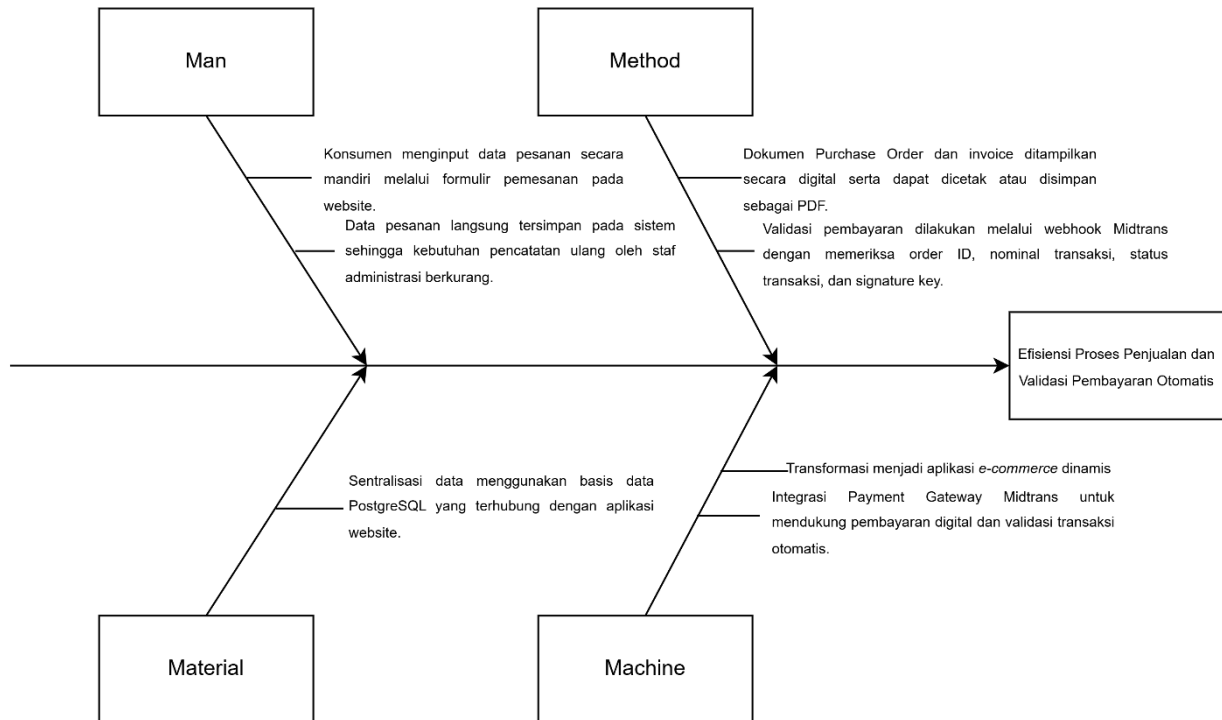
Gambar 3.1 Fishbone Masalah

Berdasarkan Gambar 3.1, berikut adalah penjelasan spesifik mengenai faktor-faktor penyebab masalah operasional di PT Marlip Indo Mandiri:

1. **Manusia (Man):** Staf administrasi perlu merekap data pesanan dari percakapan WhatsApp. Pencatatan ulang tersebut berpotensi menimbulkan ketidaksesuaian data pesanan, seperti alamat, spesifikasi, atau jumlah pesanan.
2. **Metode (Method):** Pembuatan draf dokumen Purchase Order (PO) masih dilakukan secara manual sehingga membutuhkan waktu administrasi. Selain itu, verifikasi pembayaran dilakukan dengan mencocokkan bukti transfer dan mutasi rekening secara manual berdasarkan nominal, waktu transaksi, dan referensi transaksi pesanan.
3. **Material (Data):** Data pesanan pada spreadsheet belum terintegrasi dengan website. Kondisi tersebut menyebabkan staf administrasi masih perlu memindahkan data pesanan secara manual dari percakapan WhatsApp ke media pencatatan.
4. **Teknologi (Machine):** Website perusahaan saat ini masih berfungsi sebagai media informasi atau katalog statis dan belum mendukung proses pemesanan secara terintegrasi. Selain itu, belum tersedia payment gateway untuk mendukung pembayaran dan validasi transaksi secara otomatis.

3.2.2 Solusi Masalah

Untuk mengatasi permasalahan yang telah diuraikan pada diagram sebelumnya, penulis merumuskan skema solusi menggunakan Diagram Fishbone Solusi. Diagram ini memetakan strategi perbaikan dari aspek manusia, metode, data, dan teknologi untuk mencapai tujuan efisiensi sistem.



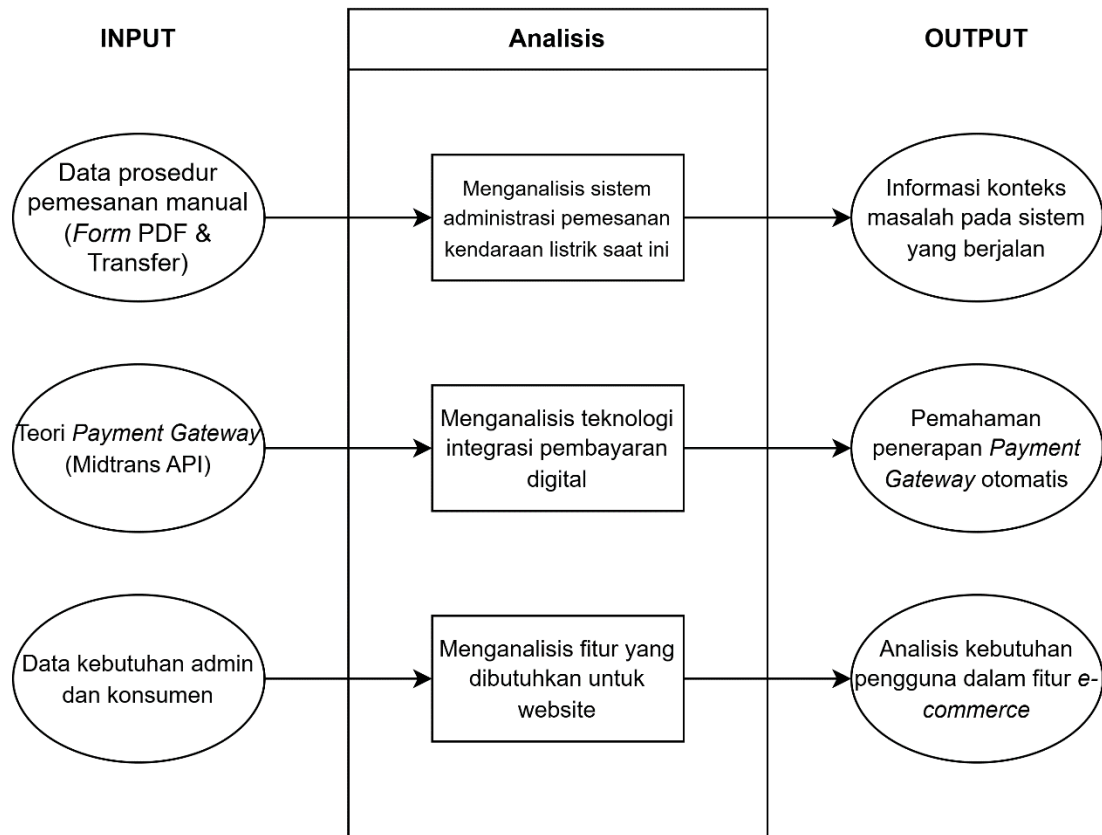
Gambar 3.2 Fishbone Solusi

Berdasarkan Gambar 3.2, berikut adalah penjelasan mengenai usulan solusi secara spesifik:

1. **Manusia (Man):** Konsumen menginput data pesanan secara mandiri melalui formulir pemesanan pada website. Data pesanan kemudian langsung tersimpan pada sistem sehingga kebutuhan pencatatan ulang oleh staf administrasi dapat dikurangi.
2. **Metode (Method):** Dokumen Purchase Order dan invoice ditampilkan secara digital serta dapat dicetak atau disimpan sebagai PDF melalui fungsi cetak peramban. Validasi pembayaran dilakukan melalui webhook Midtrans dengan memeriksa order ID, nominal transaksi, status transaksi, dan signature key. Dengan mekanisme tersebut, pemeriksaan mutasi rekening secara manual tidak lagi diperlukan.
3. **Material (Data):** Sistem menerapkan sentralisasi data menggunakan basis data PostgreSQL yang terhubung dengan aplikasi website. Data produk, pesanan, dokumen PO, invoice, pembayaran, dan dokumentasi produksi disimpan secara terpusat serta dapat diakses sesuai dengan hak akses pengguna.
4. **Teknologi (Machine):** Website dikembangkan menjadi aplikasi e-commerce dinamis menggunakan framework Next.js. Sistem juga diintegrasikan dengan Midtrans Snap API untuk mendukung pembayaran digital serta validasi transaksi otomatis.

3.2.3 Skema Analisis

Skema analisis merupakan gambaran dari tahapan analisis yang dilakukan. Dalam pengembangan sistem ini, skema analisis terdiri dari input, proses, dan output. Input merupakan data yang akan dimanfaatkan pada proses analisis yang menghasilkan output. Skema analisis dapat dilihat pada Gambar 3.3.



Gambar 3.3 Skema Analisis

Setelah digambarkan skema analisis, kemudian akan dijelaskan tahapan yang dilakukan dan maksud dari setiap tahapan yang akan dilakukan. Pada Tabel 3.2 merupakan deskripsi tahapan skema analisis.

Tabel 3.2 Deskripsi Skema Analisis

Langkah Analisis	Objek Analisis	Hasil Analisis	Maksud Analisis
Mengumpulkan data mengenai sistem pemesanan kendaraan listrik saat ini.	Data prosedur pemesanan manual (Form PDF dan transfer bank).	Informasi tentang alur pencatatan transaksi dan validasi pembayaran manual.	Untuk memahami konteks masalah pada sistem berjalan yang harus diotomatisasi.
Menganalisis teknologi integrasi pembayaran digital.	Teori tentang Payment Gateway (Midtrans API).	Pemahaman tentang penerapan Payment Gateway.	Untuk mengetahui pemahaman tentang cara integrasi pembayaran otomatis.
Mengumpulkan data mengenai fitur yang dibutuhkan admin dan konsumen.	Kebutuhan admin dan konsumen.	Analisis kebutuhan pengguna dalam fitur e-commerce kendaraan listrik.	Untuk mendapatkan gambaran informasi tentang fitur yang dibutuhkan oleh pengguna.

3.3 Profil Tempat Penelitian



Gambar 3.4 Logo PT Marlip Indo Mandiri

Penelitian ini dilaksanakan di PT Marlip Indo Mandiri, sebuah perusahaan manufaktur yang bergerak di bidang perakitan dan penjualan kendaraan listrik khusus (*Special Purpose Vehicle*). Pemahaman terhadap profil instansi diperlukan untuk memastikan solusi sistem *e-commerce* yang dirancang sesuai dengan karakteristik operasional dan kebutuhan transaksi perusahaan.

3.3.1 Objek Penelitian

Objek penelitian dalam penyusunan tugas akhir ini adalah sistem administrasi penjualan dan validasi pembayaran di PT Marlip Indo Mandiri. Pemilihan objek penelitian ini didasarkan pada urgensi kebutuhan digitalisasi proses pemesanan kendaraan. Saat ini, proses pelayanan penjualan masih terkendala oleh sistem administrasi konvensional, seperti penggunaan *Purchase Order* (PO) fisik dan pengecekan mutasi rekening secara manual, yang memunculkan inefisiensi waktu dan kerentanan data.

Adapun identitas objek penelitian adalah sebagai berikut:

- **Nama Instansi:** PT Marlip Indo Mandiri.
- **Fokus Penelitian:** Pengembangan platform *E-Commerce* kendaraan listrik berbasis web (*Next.js*) yang terintegrasi dengan *Payment Gateway* (Midtrans) untuk otomatisasi proses pemesanan dan validasi pembayaran.

3.3.2 Visi dan Misi

Visi dan misi PT Marlip Indo Mandiri dirangkum sebagai berikut:

- **Visi:** Menjadi Industri Kendaraan Listrik Nomor 1 di Indonesia.
- **Misi:**
 1. **Transisi:** Mempercepat transisi Indonesia penggunaan energi berkelanjutan yang ramah lingkungan.

2. **Dominasi & Kualitas:** Menguasai pasar kendaraan listrik di Indonesia, serta memproduksi produk kendaraan yang unggul dan berkualitas.
3. **Inovasi & Produktivitas:** Efisien, produktif, inovatif, dan selalu terdepan melalui penguasaan teknologi dan pembangunan Sumber Daya Manusia (SDM).
4. **Integritas & Kesejahteraan:** Mewujudkan kesejahteraan karyawan dan menguntungkan pemangku kepentingan (*stakeholder*) serta pemegang saham (*shareholder*), serta menjalankan manajemen anti korupsi, jujur, dan berintegritas tinggi.

3.4 Teknik Pengumpulan Data

Untuk memperoleh data dan informasi yang akurat serta relevan dalam pembangunan sistem *e-commerce*, penulis menggunakan beberapa teknik pengumpulan data sebagai berikut:

1. Studi Literatur

Penulis mengumpulkan dan mempelajari referensi teoritis dari buku, jurnal ilmiah, dan dokumentasi teknis yang berkaitan dengan topik penelitian. Fokus studi literatur meliputi konsep *E-Commerce*, keamanan transaksi digital, serta dokumentasi teknis pengembangan antarmuka menggunakan *Framework Next.js* dan integrasi *API Payment Gateway* (Midtrans). Hasil studi literatur ini menjadi landasan teoretis untuk merumuskan solusi.

2. Wawancara (*Interview*)

Teknik ini dilakukan melalui tanya jawab secara langsung dengan narasumber dari pihak manajemen PT Marlip Indo Mandiri. Penulis melakukan wawancara untuk menggali informasi mengenai:

- Alur kerja pelayanan penjualan yang sedang berjalan (*current workflow*).
- Kendala operasional terkait validasi bukti transfer manual.
- Kebutuhan fitur yang diharapkan pada sistem baru (seperti pelacakan tahapan produksi).

3. Observasi (*Observation*)

Penulis melakukan pengamatan langsung ke lokasi penelitian untuk memahami proses bisnis secara visual. Objek observasi meliputi prosedur penerimaan pesanan via WhatsApp dan pembuatan formulir fisik *Purchase Order*. Hasil observasi ini memvalidasi adanya inefisiensi pada prosedur manual yang berjalan.

4. Dokumentasi

Dokumentasi dilakukan dengan mengumpulkan artefak pendukung seperti contoh formulir *Purchase Order* fisik, katalog produk kendaraan, serta spesifikasi teknis kendaraan. Data dokumentasi ini digunakan sebagai bahan analisis lebih lanjut dalam perancangan basis data dan antarmuka sistem.

3.5 Instrumen Penelitian

Instrumen penelitian adalah alat bantu yang digunakan penulis untuk mengumpulkan data dan mengembangkan sistem agar pekerjaan menjadi lebih cermat, lengkap, dan sistematis. Instrumen yang digunakan dalam penelitian ini dibagi menjadi dua kategori:

3.5.1 Instrumen Pengumpulan Data

Dalam tahap pengumpulan data, penulis bertindak sebagai instrumen utama dengan bantuan pedoman sebagai berikut:

1. **Pedoman Wawancara (*Interview Guide*):** Daftar pertanyaan terstruktur yang diajukan kepada narasumber terkait alur pemesanan, kendala validasi pembayaran, dan kebutuhan fitur sistem.
2. **Lembar Observasi (*Observation Sheet*):** Daftar periksa untuk mengamati prosedur manual, seperti pengelolaan dokumen PO fisik.

3.5.2 Alat Pengembangan Sistem

Untuk merancang dan membangun platform *E-Commerce*, penulis menggunakan perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut:

a. Perangkat Keras (*Hardware*):

- Laptop Dell Latitude 7300.
- Prosesor: Intel(R) Core(TM) i7-8665U CPU @ 1.90GHz
- RAM: 16 GB.
- Penyimpanan: SSD NVMe 239 GB.

b. Perangkat Lunak (*Software*):

- Sistem Operasi: Windows 11 Pro 64-bit.
- Code Editor: Visual Studio Code.
- Desain UI/UX: Figma.
- Web Browser: Google Chrome.

- Runtime Environment: Node.js.
- Bahasa Pemrograman: TypeScript.
- Framework & Library: Next.js (App Router), React, dan Tailwind CSS.
- Database & ORM: PostgreSQL (Neon) dan Prisma ORM.
- Payment Gateway: Midtrans Snap API (Sandbox).

3.6 Metode Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan dalam membangun website pemesanan kendaraan listrik pada PT Marlip Indo Mandiri adalah model sekuensial linier atau yang lebih dikenal dengan model Waterfall (Air Terjun) menurut Pressman dan Maxim (2020). Model ini mengusulkan sebuah pendekatan perangkat lunak yang sistematis dan sekuensial yang dimulai pada tingkat sistem dan kemajuan melalui analisis, desain, kode, pengujian, dan pemeliharaan.

Adapun tahapan-tahapan pengembangan yang dilakukan berdasarkan model Waterfall Pressman adalah sebagai berikut:

1. Communication (Komunikasi)

Tahapan ini merupakan langkah awal yang krusial, di mana pengembang melakukan inisiasi proyek dan pengumpulan kebutuhan sistem. Pada tahap ini, dilakukan observasi langsung dan wawancara dengan pihak PT Marlip Indo Mandiri (Bapak Ir. Masrah Marang, M.M.) untuk mengidentifikasi alur bisnis *made by order*, mekanisme pembayaran bertahap dua termin, serta kendala pada sistem pemesanan yang berjalan saat ini.

2. Planning (Perencanaan)

Setelah kebutuhan sistem teridentifikasi, tahapan selanjutnya adalah melakukan perencanaan strategis. Tahap ini berfokus pada estimasi teknis pembuatan perangkat lunak, penjadwalan pengerjaan tugas akhir, serta alokasi sumber daya. Perencanaan dilakukan agar pembangunan website pemesanan dan integrasi *Payment Gateway* Midtrans dapat selesai sesuai dengan tenggat waktu yang telah ditetapkan.

3. Modeling (Pemodelan)

Tahapan pemodelan merupakan proses penerjemahan syarat kebutuhan ke dalam bentuk rancangan perangkat lunak yang dapat diperkirakan sebelum proses pengkodean (*coding*) dimulai. Pada penelitian ini, pemodelan sistem menggunakan *Unified Modeling Language* (UML) yang terdiri dari *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram*. Selain

itu, digunakan *Entity Relationship Diagram* (ERD) untuk menggambarkan rancangan basis data serta dilakukan perancangan antarmuka pengguna (*User Interface*) untuk Konsumen, Admin, dan Kepala Produksi.

4. Construction (Konstruksi)

Tahapan ini merupakan proses mengubah rancangan pemodelan menjadi perangkat lunak fungsional melalui penulisan kode program (*coding*). Pembangunan sistem (*front-end* dan *back-end*) dilakukan menggunakan kerangka kerja Next.js, antarmuka Tailwind CSS, dan basis data PostgreSQL. Setelah kode selesai ditulis, dilakukan tahapan pengujian (*testing*) berupa *Black Box Testing* untuk memastikan setiap fungsionalitas, seperti fitur *checkout* dan *Webhook* Midtrans, berjalan dengan baik dan bebas dari *error*.

5. Deployment (Penerapan)

Tahapan akhir ini melibatkan peluncuran atau penerapan sistem agar dapat digunakan oleh pengguna secara nyata. Website pemesanan yang telah lulus uji coba akan diunggah ke *server cloud* (seperti Vercel) agar dapat diakses secara daring oleh pihak manajemen PT Marlip Indo Mandiri dan calon konsumen. Selanjutnya, dilakukan evaluasi kinerja sistem dan dukungan pemeliharaan apabila terdapat penyesuaian di masa mendatang.

BAB 4

ANALISIS DAN PERANCANGAN

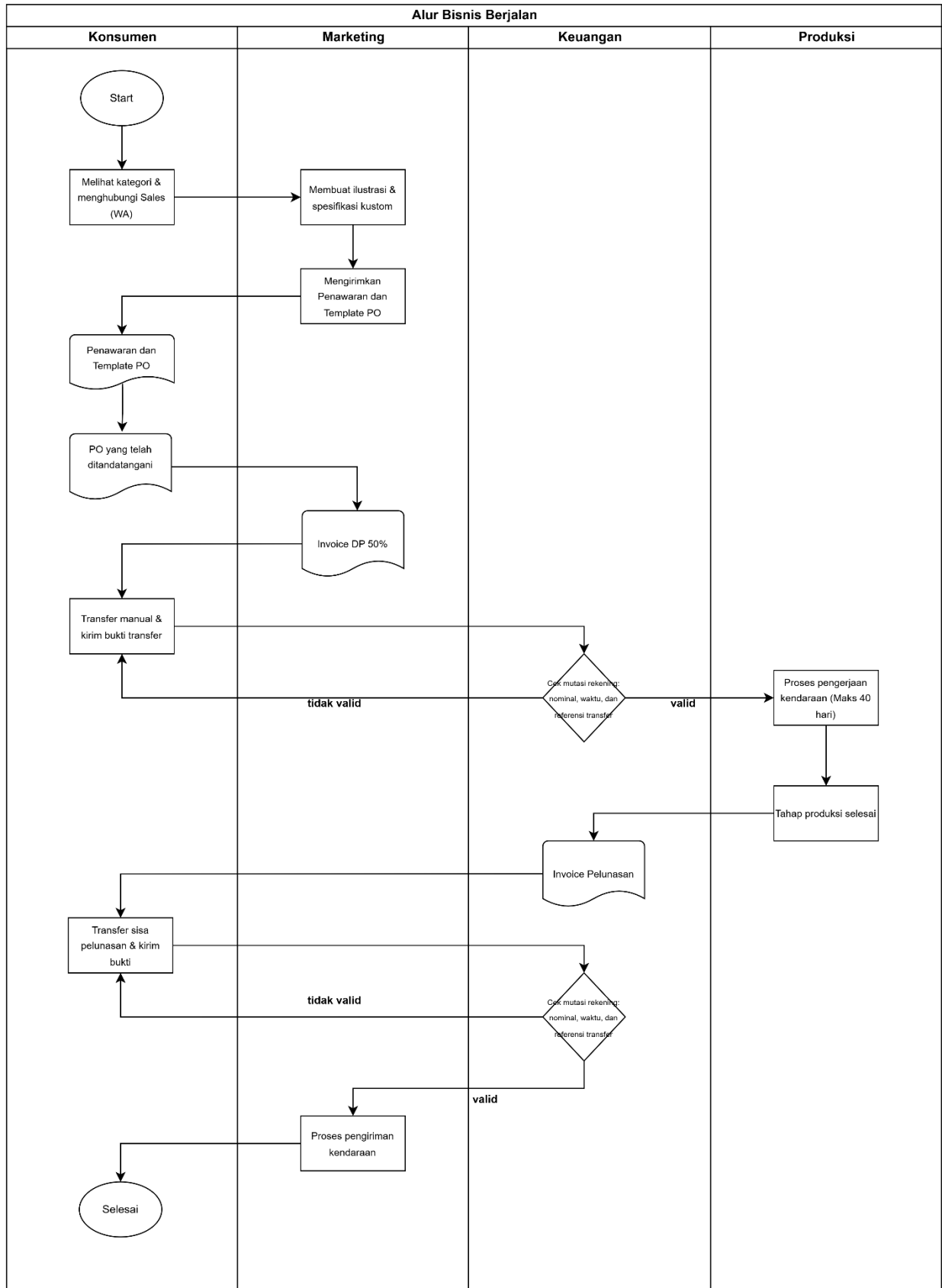
4.1 Analisis Sistem

Analisis sistem merupakan tahapan penting untuk memahami bagaimana proses bisnis di PT Marlip Indo Mandiri beroperasi saat ini, mengidentifikasi kelemahan yang ada, serta merumuskan kebutuhan untuk sistem yang baru. Analisis ini menjadi landasan dasar dalam merancang arsitektur dan fungsionalitas aplikasi *e-commerce* yang akan dibangun.

4.1.1 Analisis Alur Bisnis yang Berjalan

Berdasarkan hasil wawancara dengan manajemen, sistem penjualan kendaraan listrik di PT Marlip Indo Mandiri menggunakan mekanisme *pre-order* dengan pendekatan pesanan sesuai kebutuhan konsumen. Perusahaan tidak menyediakan penjualan unit *ready stock*. Alur bisnis manual yang berjalan saat ini adalah sebagai berikut:

1. **Eksplorasi Produk dan Permintaan Kustomisasi:** Konsumen melihat kategori produk, kemudian menghubungi pihak Marketing melalui WhatsApp atau telepon untuk mendiskusikan kebutuhan tambahan kendaraan, seperti penambahan kursi menghadap belakang, pelindung hujan, atau *coolbox*.
2. **Penawaran dan Pembuatan PO:** Pihak Marketing membuat ilustrasi serta spesifikasi kendaraan sesuai kebutuhan Konsumen, kemudian mengirimkan penawaran dan template *Purchase Order* (PO). Konsumen mengunduh, mengisi, menandatangani, dan mengirimkan kembali dokumen PO kepada pihak Marketing.
3. **Pembayaran DP Manual:** Perusahaan menerbitkan Invoice DP minimal sebesar 50% dari total harga. Konsumen melakukan transfer bank dan mengirimkan bukti transfer. Bagian Keuangan kemudian memeriksa mutasi rekening secara manual dengan mencocokkan nominal pembayaran, waktu transaksi, dan referensi transfer dengan data pesanan.
4. **Proses Produksi:** Setelah pembayaran DP divalidasi, pesanan masuk ke antrean produksi. Waktu pengerjaan untuk pesanan standar diperkirakan maksimal 40 hari, sedangkan pesanan kustom disesuaikan dengan spesifikasi yang disetujui. Informasi tahapan produksi belum tersedia dalam sistem dan hanya disampaikan apabila Konsumen menghubungi pihak Marketing.
5. **Pelunasan dan Pengiriman:** Setelah seluruh proses produksi selesai, Bagian Keuangan mengirimkan Invoice pelunasan kepada Konsumen. Konsumen melakukan transfer sisa pembayaran, kemudian Bagian Keuangan kembali mencocokkan mutasi rekening secara manual. Setelah pembayaran dinyatakan lunas, kendaraan diproses untuk dikirim ke alamat Konsumen.



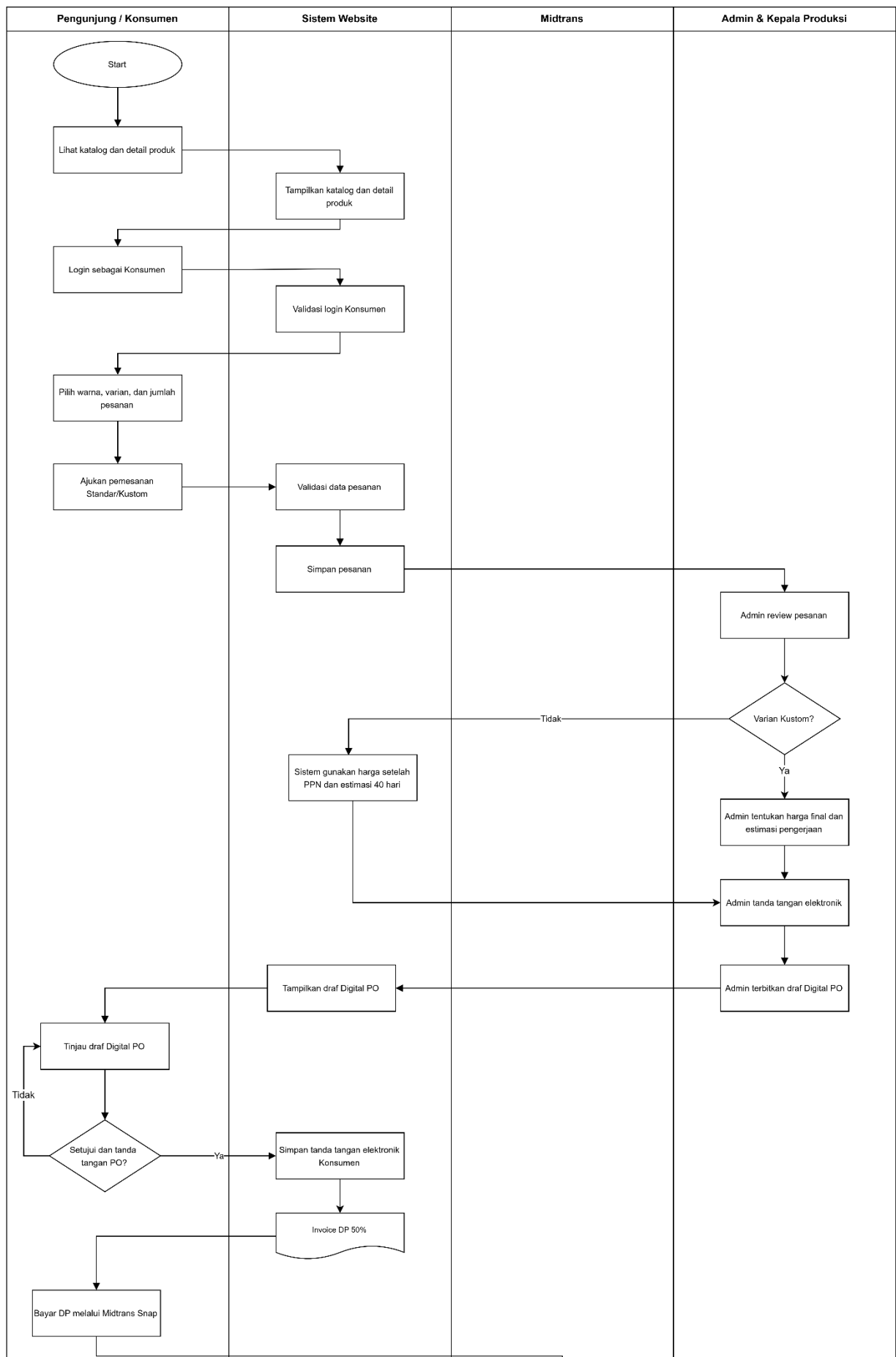
Gambar 4.1 Flowmap Alur Bisnis yang Berjalan

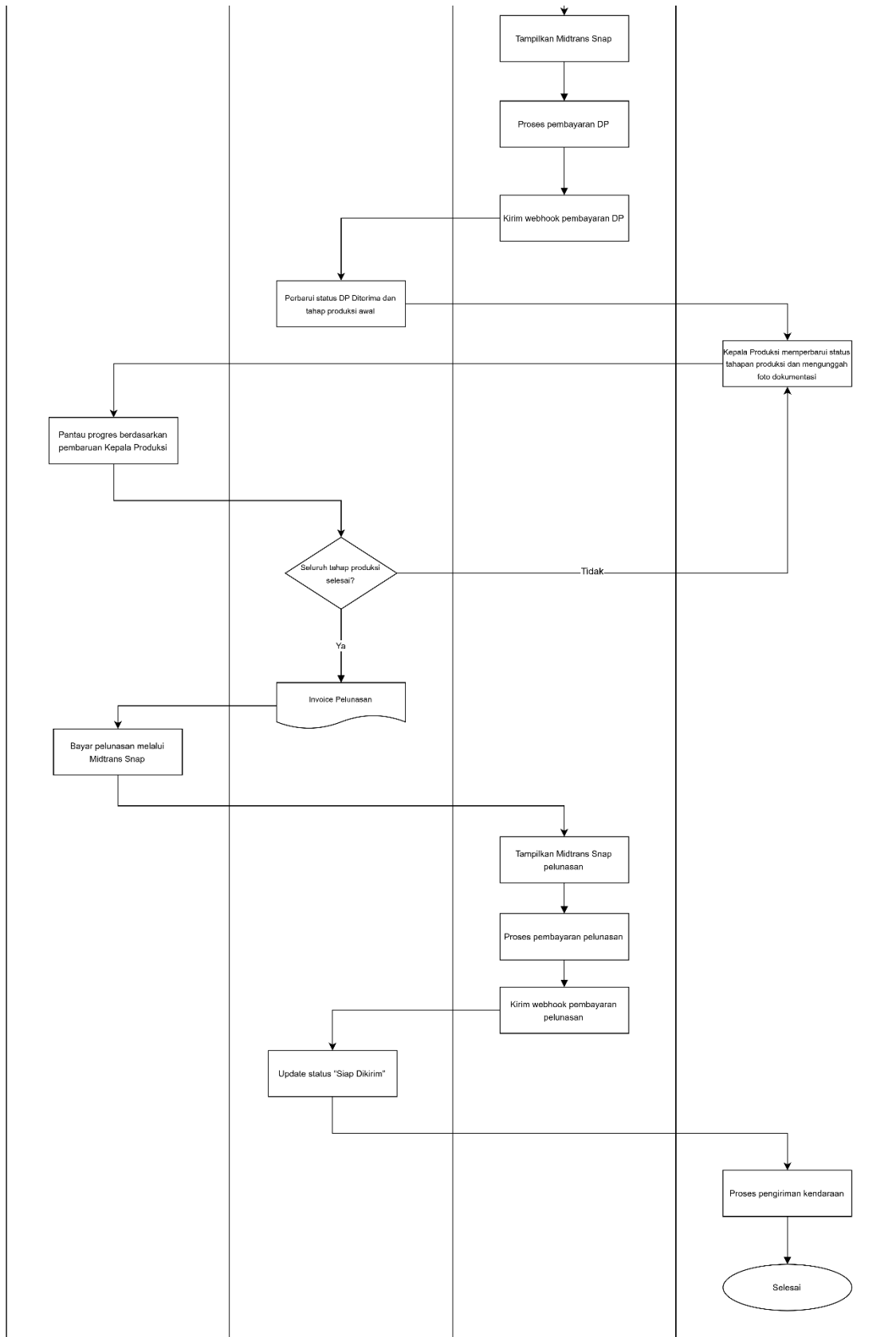
4.1.2 Analisis Alur Bisnis Usulan

Untuk mengatasi komunikasi manual serta ketidakefisienan validasi pembayaran, diusulkan aplikasi pemesanan kendaraan listrik berbasis web yang terintegrasi dengan *payment gateway* Midtrans. Alur bisnis usulan adalah sebagai berikut:

1. **Eksplorasi Produk dan Pengajuan Pemesanan:** Pengunjung dapat melihat katalog dan detail produk tanpa login. Pengguna yang akan mengajukan pesanan wajib login sebagai Konsumen. Konsumen memilih warna utama Hitam atau Putih, varian Standar atau Kustom, serta jumlah pesanan sebanyak 1 sampai 20 unit. Catatan kebutuhan khusus hanya ditampilkan dan wajib diisi apabila Konsumen memilih varian Kustom.
2. **Peninjauan dan Persetujuan Digital PO:** Admin meninjau pesanan Konsumen. Untuk varian Kustom, Admin menentukan harga final dan estimasi pengerjaan berdasarkan hasil peninjauan. Untuk varian Standar, sistem menggunakan harga setelah PPN dan estimasi pengerjaan 40 hari. Admin memberikan tanda tangan elektronik dan menerbitkan draf Digital PO. Konsumen meninjau serta memberikan tanda tangan elektronik sebagai persetujuan. Setelah PO disetujui, sistem menerbitkan Invoice DP sebesar 50%. Dokumen PO dapat dicetak atau disimpan sebagai PDF melalui fungsi cetak pada peramban.
3. **Pembayaran DP melalui Midtrans:** Konsumen melakukan pembayaran DP melalui antarmuka Midtrans Snap dengan memilih metode pembayaran yang tersedia.
4. **Validasi Pembayaran Otomatis:** Midtrans mengirimkan *webhook* kepada sistem setelah transaksi diproses. Sistem memvalidasi *order ID*, nominal transaksi, status transaksi, dan *signature key*. Apabila valid, status pesanan berubah menjadi DP Diterima tanpa pemeriksaan mutasi rekening secara manual. Pesanan kemudian memasuki tahap produksi awal.
5. **Pembaruan Tahapan Produksi:** Kepala Produksi yang berwenang memperbarui status setiap tahapan produksi sesuai urutan tahapan yang ditetapkan pada sistem berdasarkan kebutuhan proses produksi PT Marlip Indo Mandiri. Tahapan ditampilkan dalam bentuk daftar periksa, bukan persentase progres. Konsumen dapat memantau tahapan yang telah diselesaikan melalui dashboard berdasarkan pembaruan dari Kepala Produksi.
6. **Penerbitan Invoice Pelunasan:** Setelah seluruh tahapan produksi dinyatakan selesai oleh Kepala Produksi, sistem menerbitkan Invoice pelunasan yang dapat diakses Konsumen melalui dashboard. Status pesanan tetap berada pada tahap Proses Perakitan sampai pembayaran pelunasan berhasil divalidasi.
7. **Pelunasan dan Pengiriman:** Konsumen melakukan pembayaran pelunasan melalui Midtrans. Setelah pembayaran berhasil divalidasi melalui *webhook*, status pesanan berubah menjadi Siap Dikirim dan kendaraan diproses untuk pengiriman.

Alur Bisnis Usulan





Gambar 4.2 Flowmap Alur Bisnis Usulan

4.1.3 Analisis Kebutuhan Berdasarkan Wawancara

Untuk merumuskan spesifikasi sistem yang tepat guna, dilakukan analisis terhadap hasil wawancara bersama pihak manajemen PT Marlip Indo Mandiri.

Tabel 4.1 Hasil Wawancara

Tanggal	Pembahasan
2 Februari 2026	A. Sistem Pembelian & Pembayaran (SOP) 1. Alur & SOP Pemesanan: Bagaimana SOP lengkap pemesanan unit? Apakah konsumen pilih unit -> bayar -> kirim, atau ada proses verifikasi survei terlebih dahulu? Jawaban: Proses verifikasi tetap diperlukan karena produk kami bersifat customizable (dapat disesuaikan) dan belum diproduksi secara massal (fix product). Berikut adalah alurnya: 1. Eksplorasi: Konsumen masuk ke website, memilih kategori, dan melihat daftar produk. 2. Diskusi Kebutuhan: Pada halaman produk, konsumen menekan tombol "Order/Diskusi" untuk menyesuaikan spesifikasi. Tim Sales akan mengonfirmasi apakah unit standar sudah sesuai atau ada kebutuhan tambahan (misalnya: penambahan kursi menghadap belakang, pelindung hujan, coolbox, dll). 3. Finalisasi Spesifikasi: Konsumen memberikan permintaan khusus (requirement). Tim Marketing akan mengirimkan ilustrasi visual dan spesifikasi teknis sesuai permintaan tersebut. 4. Pemesanan: Setelah spesifikasi disepakati, proses berlanjut ke tahap administrasi pemesanan (Purchase Order).
	2. Mekanisme Pembayaran: Apakah wajib Full Payment (Lunas di muka), atau boleh membayar Booking Fee (Tanda Jadi) terlebih dahulu? Jawaban: Sistem pembayaran menggunakan metode Down Payment (DP) dengan alur sebagai berikut: 1. Penerbitan PO: Setelah spesifikasi disepakati (Fix), perusahaan memberikan penawaran resmi. Konsumen memilih/membuat Purchase Order (PO) secara online. 2. Invoicing & DP: Sistem mengirimkan Invoice. Konsumen wajib membayar Uang Muka (Down Payment) minimal 50% dari total nilai pesanan. Pembayaran dapat dilakukan via transfer bank manual atau Virtual Account. 3. Produksi & Tracking: Masa produksi berlangsung hingga 40 hari. Selama masa ini, sistem harus menampilkan Laporan Progres (Progress Report) mingguan kepada konsumen (misal: Minggu 1 progres sekian persen, dilengkapi foto dari pabrik). 4. Pelunasan: Satu minggu sebelum estimasi selesai (atau saat progres mendekati 100%), sistem mengirimkan tagihan pelunasan. 5. Pengiriman: Unit dikirim (delivery) setelah pelunasan diterima.
	3. Nominal Booking Fee: Jika menggunakan sistem Booking Fee: o Apakah nominalnya Fix/Tetap (Misal: Rp 5 Juta rata untuk semua mobil) Atau menggunakan Persentase (Misal: 10% dari harga unit)? Jawaban: Tidak ada nominal tetap (fix). Biaya DP ditetapkan sebesar 50% dari total harga unit yang disepakati.
	4. Fasilitas & Tenor Cicilan: • Apakah tersedia fasilitas cicilan? • Jika ya, apakah cicilan Internal (ke Perusahaan) atau via Pihak Ketiga (Leasing)? • Berapa kali maksimal termin cicilannya? (Misal: 3x, 6x, atau 12 bulan). Jawaban: Saat ini fitur cicilan belum tersedia
	5. Syarat Administrasi: Dokumen apa saja yang wajib diunggah oleh konsumen saat melakukan pemesanan? (Misal: KTP, KK, atau NPWP). Jawaban: Saat melakukan pemesanan proses order jadi disitu tersedia Dokumen Purchase Order (PO) yang konsumen mesti tanda tangan dan itu harus mengunggah kembali untuk jadi bukti bahwa konsumen sudah memesan
	B. Manajemen Stok & Verifikasi Pesanan 1. Logika Ketersediaan Unit (Frontend):

Tanggal	Pembahasan
	• Jika stok di database terbaca kosong (0), bagaimana tampilan yang diinginkan di website? • Apakah tombol beli dimatikan (Status: Tidak Tersedia)? • Atau berubah menjadi Pre-Order (PO) dan konsumen tetap bisa memesan? (Jika PO, berapa lama estimasi tunggunya?). Jawaban: Karena sistem kami adalah Made by Order (dibuat berdasarkan pesanan), maka tidak ada stok "Ready" yang habis. Tombol "Beli" fungsinya adalah untuk melakukan Pre-Order (PO). 2. Verifikasi & Update Status (Backend): • Setelah pesanan masuk, apakah Admin perlu melakukan cek stok fisik di gudang sebelum memproses pesanan? • Jika ternyata unitnya unavailable (rusak/habis mendadak), apakah Admin akan mengubah status pesanan secara manual menjadi "Cancel/Dibatalkan" dengan alasan Out of Stock? Jawaban: Tidak ada proses pengecekan stok fisik atau pembatalan karena alasan "stok habis". Karena sistemnya Pre-Order, setiap pesanan yang masuk otomatis dianggap sebagai permintaan produksi baru sesuai spesifikasi yang disepakati.
	3. Kebijakan Pembatalan & Refund: • Skenario 1 (Dari Admin): Jika pesanan dibatalkan oleh Admin karena stok habis, bagaimana prosedur pengembalian dananya (Refund)? • Skenario 2 (Dari Konsumen): Jika konsumen yang membatalkan pesanan secara sepihak (berubah pikiran) setelah bayar Booking Fee, apakah uang hangus atau bisa dikembalikan? Jawaban: Tidak ada pembatalan karena stok habis (karena sistem PO) Pesanan yang sudah masuk tahap PO dan dibayar bersifat mutlak. Jika konsumen membatalkan secara sepihak, maka dana DP dinyatakan hangus (tidak ada refund) C. Teknis & Integrasi Data 1. Data Produk: Untuk keperluan database sistem baru ini, apakah saya diizinkan mengambil data spesifikasi dan foto produk dari arsip website (marlip.tech)? Jawaban: Diizinkan. Data bisa diambil dari website yang sudah ada, atau berkoordinasi langsung dengan tim developer sebelumnya (kontak akan diberikan). 2. Pengelola Website: Siapakah yang saat ini mengelola website? (Jika ada tim IT, apakah saya boleh meminta kontakannya untuk koordinasi teknis?). Jawaban: Dikelola oleh pihak ketiga (Vendor/Developer luar). Kontak akan diberikan untuk koordinasi teknis. D. Kebutuhan Admin (Dashboard) 1. Laporan Penjualan: Laporan atau fitur apa yang paling Bapak butuhkan untuk memantau penjualan? (Misal: Grafik penjualan bulanan, daftar konsumen yang belum lunas, atau stok terlaris). Jawaban: Dashboard harus menyajikan laporan komprehensif, meliputi: 1. Interest/Leads: Data pengunjung yang baru sekadar melihat-lihat atau mengklik produk. 2. Order Masuk: Jumlah PO yang terbentuk dalam periode tertentu (bulanan). 3. Status Pesanan: Laporan jumlah pesanan yang Batal, sedang Produksi, hingga Rilis/Terikirim (Delivery).
11 Juni 2026	Klarifikasi Hasil Wawancara Lanjutan Berdasarkan klarifikasi lanjutan dengan pihak PT Marlip Indo Mandiri, formulir pemesanan kendaraan dibedakan menjadi dua varian, yaitu varian Standar dan varian Kustom. Kedua varian menyediakan pilihan warna utama Hitam atau Putih serta jumlah pesanan sebanyak 1–20 unit. Pada varian Standar, Konsumen tidak perlu mengisi catatan tambahan karena spesifikasi mengikuti standar produk yang tersedia. Pada varian Kustom, Konsumen wajib mengisi catatan kebutuhan khusus sebagai dasar peninjauan oleh Admin.

Tanggal	Pembahasan
	Harga final dan estimasi pengerjaan untuk varian Kustom ditentukan oleh Admin setelah dilakukan peninjauan terhadap kebutuhan Konsumen. Sementara itu, varian Standar menggunakan harga setelah PPN dan estimasi pengerjaan selama 40 hari. Setelah Konsumen menyetujui draf Digital PO dan pembayaran DP berhasil diverifikasi, sistem menetapkan progres awal pesanan sebesar 15% sebagai tahap persiapan produksi. Dalam implementasi sistem, Invoice pelunasan diterbitkan ketika progres produksi telah mencapai 100%. Setelah pelunasan berhasil diverifikasi, status pesanan berubah menjadi Siap Dikirim sehingga pihak perusahaan dapat memproses pengiriman kendaraan. Selain itu, berdasarkan informasi dari pihak perusahaan, ketentuan garansi kendaraan meliputi garansi baterai selama 3 tahun, garansi rangka selama 5 tahun, dan garansi komponen kelistrikan selama 1 tahun sesuai syarat dan ketentuan PT Marlip Indo Mandiri.

4.1.4 Analisis Kebutuhan Pengguna (*User Requirement*)

Analisis kebutuhan pengguna merupakan proses identifikasi kebutuhan setiap aktor yang berinteraksi dengan sistem. Pada perancangan sistem pemesanan kendaraan listrik PT Marlip Indo Mandiri, kebutuhan pengguna dijelaskan pada Tabel 4.2 berikut:

Tabel 4.2 Analisis Kebutuhan Pengguna

Kode	Deskripsi Kebutuhan Pengguna
UR-01	Konsumen ingin dapat membuat akun dan melakukan login agar identitas pemesan, riwayat pesanan, persetujuan PO, dan data tagihan terhubung dengan akun yang sah.
UR-02	Konsumen ingin dapat melihat katalog dan detail produk kendaraan sebelum mengajukan pesanan.
UR-03	Konsumen ingin dapat mengajukan pemesanan kendaraan dengan memilih varian Standar atau Kustom sesuai kebutuhan.
UR-04	Konsumen ingin sistem dapat menerbitkan draf Digital Purchase Order untuk ditinjau dan disetujui serta menyediakan pembayaran dua termin melalui Midtrans Snap.
UR-05	Konsumen ingin dapat melacak tahapan produksi kendaraan dan mencetak atau menyimpan dokumen transaksi, yaitu PO dan Invoice, sebagai PDF melalui fungsi cetak peramban.
UR-06	Admin sebagai penanggung jawab administrasi penjualan ingin dapat menambah, memperbarui, dan menonaktifkan data produk; meninjau pesanan; menentukan harga final serta estimasi pengerjaan pesanan Kustom; dan menerbitkan draf PO.
UR-07	Kepala Produksi sebagai penanggung jawab proses produksi ingin dapat memperbarui status tahapan produksi berdasarkan ketentuan proses produksi perusahaan serta mengunggah foto dokumentasi produksi.

4.1.5 Analisis Kebutuhan Perangkat Lunak (*Software Requirement*)

Software requirement merupakan deskripsi fungsi yang harus disediakan perangkat lunak untuk memenuhi kebutuhan pengguna. Identifikasi kebutuhan fungsional perangkat lunak ditunjukkan pada Tabel 4.3 berikut:

Tabel 4.3 Analisis Kebutuhan Perangkat Lunak

Kode UR	User Requirement	Kode SR	Software Requirement
UR-01	Konsumen ingin dapat membuat akun dan login.	SR-01	Sistem harus menyediakan fitur registrasi bagi Konsumen baru dengan menyimpan data identitas pengguna ke dalam basis data.
		SR-02	Sistem harus menyediakan fitur autentikasi bagi Konsumen, Admin, dan Kepala Produksi berdasarkan hak akses masing-masing.
UR-02	Konsumen ingin melihat katalog dan detail produk.	SR-03	Sistem harus menampilkan katalog dan detail produk kendaraan listrik beserta kategorinya.
UR-03	Konsumen ingin mengajukan pemesanan kendaraan.	SR-04	Sistem harus menyediakan formulir pemesanan yang memuat warna utama, varian Standar atau Kustom, jumlah pesanan 1 sampai 20 unit, serta catatan kebutuhan khusus yang wajib diisi untuk varian Kustom.
UR-04	Konsumen ingin meninjau Digital PO dan melakukan pembayaran melalui Midtrans.	SR-05	Sistem harus menampilkan draf Digital Purchase Order, menyediakan tanda tangan elektronik, dan menyediakan persetujuan PO oleh Konsumen.
		SR-06	Sistem harus terintegrasi dengan <i>payment gateway</i> Midtrans untuk memproses pembayaran dua termin, yaitu DP sebesar 50% dan pelunasan.

Kode UR	User Requirement	Kode SR	Software Requirement
		SR-07	Sistem harus memperbarui status pembayaran secara otomatis melalui penerimaan dan validasi <i>webhook</i> dari Midtrans.
UR-05	Konsumen ingin melacak tahapan produksi serta mencetak atau menyimpan dokumen transaksi.	SR-08	Sistem harus menyediakan dashboard pelacakan yang menampilkan status tahapan produksi berdasarkan pembaruan Kepala Produksi.
		SR-09	Sistem harus menyediakan tampilan Digital PO dan Invoice yang dapat dicetak atau disimpan sebagai PDF melalui fungsi cetak pada peramban.
UR-06	Admin ingin mengelola produk dan pesanan.	SR-10	Sistem harus menyediakan fitur untuk menambahkan, melihat, memperbarui, dan menonaktifkan produk. Produk aktif harus ditampilkan pada katalog Konsumen.
		SR-11	Sistem harus memfasilitasi Admin untuk meninjau pesanan, menentukan harga final dan estimasi pengerjaan pesanan Kustom, memberikan tanda tangan elektronik, serta menerbitkan draf PO.
UR-07	Kepala Produksi ingin memperbarui tahapan produksi dan mengunggah dokumentasi.	SR-12	Sistem harus memfasilitasi Kepala Produksi untuk memperbarui status tahapan produksi berdasarkan ketentuan proses produksi perusahaan.
		SR-13	Sistem harus memungkinkan Kepala Produksi mengunggah foto dokumentasi pada tahapan produksi yang diperbarui.

4.1.6 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional mendefinisikan batasan, kriteria, dan standar kualitas yang harus dipenuhi oleh sistem agar dapat beroperasi dengan baik.

Tabel 4.4 Analisis Kebutuhan Non-Fungsional

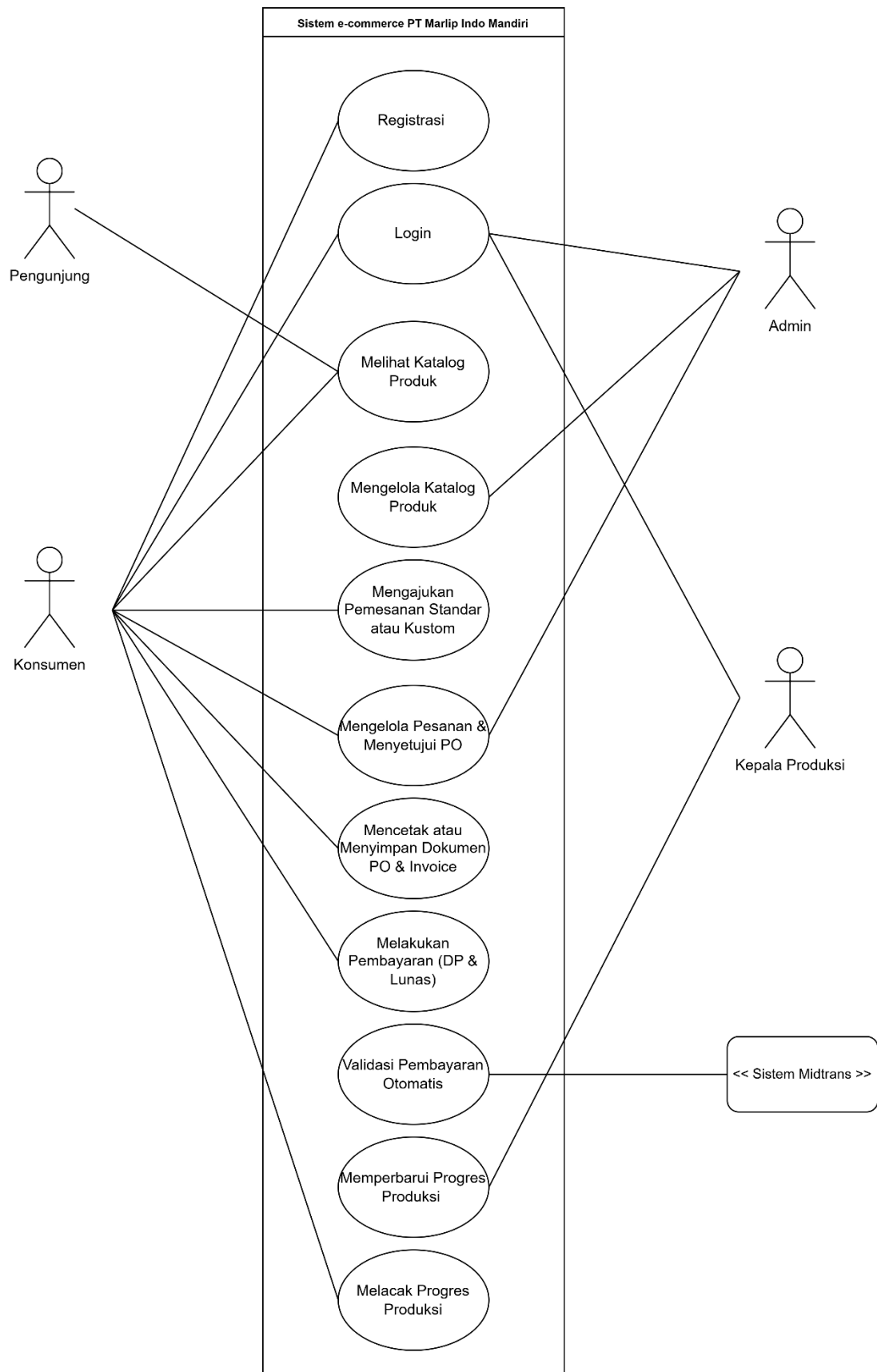
Kode	Kategori	Deskripsi Kebutuhan
NFR-01	Technology Constraint	Sistem dibangun menggunakan kerangka kerja (framework) Next.js dengan bahasa pemrograman TypeScript, antarmuka Tailwind CSS, dan pengolahan basis data menggunakan PostgreSQL.
NFR-02	Environment & Usability	Antarmuka sistem harus bersifat responsif (dapat menyesuaikan tampilan secara optimal pada perangkat desktop, tablet, maupun mobile) dan dapat diakses melalui peramban web modern tanpa instalasi perangkat lunak tambahan.
NFR-03	Security	Sistem menerapkan pembatasan akses halaman berdasarkan peran Konsumen, Admin, dan Kepala Produksi serta menggunakan hashing untuk melindungi kata sandi pengguna.
NFR-04	Security & Integrity	Sistem harus memvalidasi setiap transaksi pembayaran melalui integrasi Webhook Midtrans yang diverifikasi menggunakan signature key untuk mencegah manipulasi status pembayaran oleh pihak yang tidak berwenang.
NFR-05	Performance	Sistem harus mampu merender halaman dan memuat data katalog produk kendaraan listrik dengan waktu respons kurang dari 3 detik pada kondisi koneksi internet standar.
NFR-06	Reliability	Sistem harus menjaga konsistensi status pembayaran antara sistem dan Midtrans melalui <i>webhook</i> dan pemeriksaan status transaksi.
NFR-07	Portability	Sistem dapat dijalankan pada browser modern (Chrome, Edge, Safari, Firefox).
NFR-08	Recoverability	Sistem memastikan keamanan data pesanan kustom jika terjadi interupsi koneksi.

4.2 Perancangan Sistem

Pada tahap perancangan sistem, dilakukan pemodelan menggunakan *Unified Modeling Language* (UML) untuk memvisualisasikan arsitektur dan fungsionalitas sistem yang akan dibangun.

4.2.1 Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna (*actor*) dengan sistem, serta mendefinisikan apa saja hak akses atau fungsionalitas yang dapat dilakukan oleh masing-masing aktor di dalam sistem *e-commerce* PT Marlip Indo Mandiri.



Gambar 4.3 Use Case Diagram

Berikut adalah deskripsi dari masing-masing aktor yang berinteraksi dengan sistem pada diagram di atas:

Tabel 4.5 Deskripsi Aktor pada Use Case Diagram

No	Nama Aktor	Deskripsi
1	Pengunjung	Pengguna yang belum login dan dapat melihat katalog serta detail produk. Pengunjung harus melakukan login atau registrasi apabila akan mengajukan pesanan.
2	Konsumen	Pengguna yang telah memiliki akun dan dapat mengajukan pesanan, menyetujui draf PO, melakukan pembayaran, mencetak atau menyimpan dokumen transaksi, serta melacak tahapan produksi kendaraan.
3	Admin	Pengguna internal yang bertanggung jawab mengelola katalog produk, meninjau pesanan, menentukan harga dan estimasi pesanan Kustom, serta menerbitkan draf Digital Purchase Order. Akun Admin tidak dibuat melalui registrasi publik.
4	Kepala Produksi	Pengguna internal dari divisi produksi yang bertugas login untuk memperbarui status tahapan produksi berdasarkan ketentuan proses produksi perusahaan serta mengunggah dokumentasi produksi.
5	Midtrans API	Aktor sistem eksternal berupa <i>payment gateway</i> yang memproses transaksi pembayaran Konsumen dan mengirimkan notifikasi pembayaran melalui <i>webhook</i> kepada sistem.

Tabel 4.6 Deskripsi Use Case Diagram

No	Kode	Nama Use Case	Deskripsi
1	UC-01	Registrasi Akun	Use case yang digunakan Konsumen baru untuk mendaftarkan data diri dan membuat akun di dalam sistem.
2	UC-02	Login	Use case yang digunakan Konsumen, Admin, dan Kepala Produksi untuk memverifikasi kredensial dan masuk ke sistem sesuai hak aksesnya.
3	UC-03	Melihat Katalog Produk	Use case yang digunakan Pengunjung dan Konsumen untuk melihat daftar produk kendaraan listrik beserta detail dan spesifikasinya.
4	UC-04	Mengelola Katalog Produk	Use case yang digunakan Admin untuk menambah, mengubah, dan menonaktifkan data produk kendaraan pada sistem.
5	UC-05	Mengajukan Pemesanan Standar atau Kustom	Use case yang memfasilitasi Konsumen mengisi formulir pemesanan dengan pilihan warna, varian Standar atau Kustom, jumlah pesanan, serta catatan kebutuhan khusus untuk varian Kustom.
6	UC-06	Mengelola Pesanan dan Menerbitkan PO	Use case yang digunakan Admin untuk meninjau pesanan masuk, menentukan harga final dan estimasi pesanan Kustom, memberikan tanda tangan elektronik, serta menerbitkan draf Digital Purchase Order.
7	UC-07	Mencetak atau Menyimpan Dokumen PO dan Invoice	Use case yang memungkinkan Konsumen mencetak atau menyimpan dokumen PO dan Invoice sebagai PDF melalui fungsi cetak peramban.
8	UC-08	Melakukan Pembayaran DP dan Pelunasan	Use case yang digunakan Konsumen untuk melakukan pembayaran DP dan pelunasan melalui integrasi Midtrans Snap.
9	UC-09	Validasi Pembayaran Otomatis	Use case yang dijalankan ketika Midtrans mengirimkan <i>webhook</i> pembayaran. Sistem memvalidasi data transaksi sebelum memperbarui status Invoice dan Purchase Order.
10	UC-10	Memperbarui Tahapan Produksi	Use case yang digunakan Kepala Produksi untuk memperbarui status tahapan produksi berdasarkan ketentuan proses produksi perusahaan serta mengunggah foto dokumentasi.
11	UC-11	Melacak Tahapan Produksi	Use case yang digunakan Konsumen untuk melihat status tahapan produksi dan dokumentasi kendaraan yang diperbarui oleh Kepala Produksi.

4.2.2 Skenario Use Case

Skenario *use case* menjelaskan secara rinci alur interaksi antara aktor dan sistem. Berikut adalah penjabaran skenario dari masing-masing fungsionalitas utama pada sistem pemesanan e-commerce PT Marlip Indo Mandiri:

Tabel 4.7 Skenario UC-01: Registrasi Akun

Identifikasi		
Use Case ID	UC-01	
Nama Use Case	Registrasi Akun	
Aktor	Konsumen	
Prekondisi	Konsumen belum memiliki akun dan berada di halaman Pendaftaran.	
Postkondisi	Akun berhasil dibuat, konsumen diarahkan ke halaman login.	
Tujuan	Memfasilitasi konsumen baru untuk mendaftarkan data diri ke dalam sistem.	
Skenario Utama		
Kondisi Awal: Halaman Registrasi		
No	Aktor	Sistem
1	Konsumen mengakses halaman pendaftaran	
2		Sistem menampilkan formulir registrasi yang memuat Nama Lengkap, Instansi/Perusahaan, Nomor WhatsApp, Email, Alamat Lengkap, dan Kata Sandi
3	Konsumen mengisi formulir dan menekan tombol "Daftar".	
4		Sistem memvalidasi kelengkapan data, format dan keunikan email, serta panjang kata sandi minimal delapan karakter.
5		Sistem melakukan <i>hashing</i> kata sandi, kemudian menyimpan data akun ke dalam basis data.
6		Sistem menampilkan notifikasi bahwa registrasi berhasil dan mengarahkan Konsumen ke halaman Login.
Kondisi Akhir : Halaman Login		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
4a	Data wajib belum lengkap atau format data tidak valid.	Sistem menolak registrasi dan menampilkan pesan kesalahan validasi.
4b	Email sudah terdaftar dalam basis data.	Sistem menolak registrasi dan menampilkan pesan "Email sudah digunakan".
4c	Kata sandi kurang dari delapan karakter.	Sistem menolak registrasi dan menampilkan pesan bahwa kata sandi minimal delapan karakter.

Tabel 4.8 Skenario UC-02: Login

Identifikasi		
Use Case ID	UC-02	
Nama Use Case	Login	
Aktor	Konsumen, Admin, Kepala Produksi	
Prekondisi	Pengguna telah memiliki akun terdaftar dan berada di halaman Login.	
Postkondisi	Pengguna berhasil masuk dan diarahkan ke dashboard masing-masing.	
Tujuan	Melakukan autentikasi dan otorisasi pengguna dengan mekanisme Role-Based Access Control.	
Skenario Utama		
Kondisi Awal: Halaman Login		
No	Aktor	Sistem
1	Pengguna mengakses halaman login	
2		Sistem menampilkan formulir login
3	Pengguna mengisi Email dan Password	
4	Pengguna klik tombol 'Masuk'	
5		Sistem memvalidasi kredensial dengan mencocokkan database
6		Sistem memeriksa role pengguna (Konsumen/Admin/Produksi)
7		Sistem membuat sesi login dan mengarahkan ke dashboard terkait
Kondisi Akhir: Halaman Dashboard Utama		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
5a	Email tidak terdaftar / Password salah	Sistem menampilkan pesan error: 'Email atau kata sandi salah'

Tabel 4.9 Skenario UC-03: Melihat Katalog Produk

Identifikasi		
Use Case ID	UC-03	
Nama Use Case	Melihat Katalog Produk	
Aktor	Pengunjung dan Konsumen	
Prekondisi	Pengunjung atau Konsumen berada di halaman utama atau halaman katalog.	
Postkondisi	Pengunjung atau Konsumen dapat melihat daftar katalog dan detail produk.	
Tujuan	Menampilkan daftar produk aktif beserta spesifikasi dan harga dasarnya.	
Skenario Utama		
Kondisi Awal: Halaman Beranda atau Katalog		
No	Aktor	Sistem
1	Pengunjung atau Konsumen membuka halaman katalog produk.	
2		Sistem mengambil data produk aktif dari basis data
3		Sistem mengurutkan produk berdasarkan kategori Kendaraan, EV Bike, Alat Pembelajaran, dan Suku Cadang
4		Sistem menampilkan daftar produk pada halaman katalog
5	Pengunjung atau Konsumen memilih salah satu produk	
6		Sistem menampilkan detail produk yang dipilih.
Kondisi Akhir: Daftar katalog atau halaman detail produk ditampilkan.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	Data produk tidak tersedia di basis data.	Sistem menampilkan pesan bahwa produk belum dapat dimuat atau belum tersedia.

Tabel 4.10 Skenario UC-04: Mengelola Katalog Produk

Identifikasi		
Use Case ID	UC-04	
Nama Use Case	Mengelola Katalog Produk	
Aktor	Admin	
Prekondisi	Admin sudah login dan berada di Dashboard Admin.	
Postkondisi	Data produk berhasil ditambahkan, diperbarui, atau dinonaktifkan pada basis data. Perubahan produk aktif ditampilkan pada katalog.	
Tujuan	Memungkinkan Admin untuk menambahkan, memperbarui, dan menonaktifkan data produk yang ditampilkan pada katalog.	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Produk		
No	Aktor	Sistem
1	Admin menekan tombol "Tambah Produk Baru"	
2		Sistem menampilkan formulir data produk
3	Admin mengisi atau memperbarui nama produk, kategori, deskripsi, spesifikasi, harga dasar, dan foto produk	
4	Admin menekan tombol "Simpan"	
5		Sistem memvalidasi kelengkapan input data dan gambar
6		Sistem menyimpan perubahan data produk ke basis data dan menampilkan status keberhasilan
Kondisi Akhir: Halaman Daftar Produk Terbaru		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
5a	Format foto bukan JPG, PNG, atau WebP, atau ukuran file melebihi 20 MB	Sistem menolak unggahan foto dan menampilkan pesan kesalahan

Tabel 4.11 Skenario UC-05: Mengajukan Pemesanan Standar atau Kustom

Identifikasi		
Use Case ID	UC-05	
Nama Use Case	Mengajukan Pemesanan Standar atau Kustom	
Aktor	Konsumen	
Prekondisi	Konsumen sudah login dan berada di halaman detail produk kendaraan.	
Postkondisi	Data pemesanan tersimpan sebagai pesanan berstatus "Menunggu Review"	
Tujuan	Memfasilitasi Konsumen untuk mengajukan pemesanan kendaraan varian Standar atau Kustom.	
Skenario Utama		
Kondisi Awal: Halaman Detail Produk		
No	Aktor	Sistem
1	Konsumen menekan tombol "Ajukan Pemesanan"	Sistem menampilkan formulir pemesanan kendaraan.
2		
3	Konsumen memilih warna utama Hitam atau Putih, varian Standar atau Kustom, serta jumlah 1–20 unit. Jika memilih varian Kustom, Konsumen wajib mengisi catatan kebutuhan khusus.	
4	Konsumen menekan tombol "Ajukan"	
5		Sistem menyimpan data pesanan ke dalam basis data
6		Sistem meneruskan pengajuan ke Dashboard Admin
Kondisi Akhir: Halaman Riwayat Pesanan Konsumen		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Pengguna belum login	Sistem mengarahkan pengguna ke halaman login.
3b	Catatan Kustom kosong	Sistem menolak pengajuan dan meminta Konsumen mengisi catatan kebutuhan khusus.
3c	Jumlah pesanan kurang dari 1 atau lebih dari 20 unit	Sistem menampilkan pesan kesalahan validasi.

Tabel 4.12 Skenario UC-06: Mengelola Pesanan & Menerbitkan PO

Identifikasi		
Use Case ID	UC-06	
Nama Use Case	Mengelola Pesanan & Menerbitkan PO	
Aktor	Admin	
Prekondisi	Terdapat pengajuan pesanan Standar atau Kustom baru dari Konsumen.	
Postkondisi	Admin memberikan tanda tangan elektronik dan menerbitkan draf Purchase Order untuk ditinjau oleh Konsumen.	
Tujuan	Memfasilitasi Admin untuk meninjau pesanan, menentukan harga final dan estimasi varian Kustom, serta menerbitkan draf Purchase Order.	
Skenario Utama		
Kondisi Awal: Halaman Pesanan Masuk Admin		
No	Aktor	Sistem
1	Admin membuka menu daftar pesanan masuk	
2		Sistem menampilkan daftar pesanan yang menunggu peninjauan Admin.
3	Admin meninjau rincian pesanan dan kebutuhan Konsumen.	
4	Untuk pesanan Kustom, Admin memasukkan harga final dan estimasi pengerjaan. Untuk pesanan Standar, sistem menggunakan harga setelah PPN dan estimasi 40 hari secara otomatis.	
5	Admin memberikan tanda tangan elektronik dan menekan tombol "Terbitkan PO".	
6		Sistem mengubah status pesanan menjadi "Draf PO" dan menampilkan draf PO kepada Konsumen.
Kondisi Akhir: Halaman Daftar Pesanan (Status Berubah)		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Pesanan tidak dapat dipenuhi oleh perusahaan.	Admin menekan tombol "Tolak Pesanan" dan sistem membatalkan pengajuan.

Tabel 4.13 Skenario UC-07: Mencetak atau Menyimpan Dokumen PO & Invoice

Identifikasi		
Use Case ID	UC-07	
Nama Use Case	Mencetak atau Menyimpan Dokumen PO dan Invoice	
Aktor	Konsumen	
Prekondisi	Konsumen memiliki dokumen PO atau Invoice pada pesanan miliknya.	
Postkondisi	Dokumen PO atau Invoice berhasil dicetak atau disimpan sebagai PDF melalui fungsi cetak peramban.	
Tujuan	Menyediakan dokumen PO dan Invoice dalam format siap cetak untuk Konsumen.	
Skenario Utama		
Kondisi Awal: Halaman Detail Transaksi		
No	Aktor	Sistem
1	Konsumen membuka rincian pesanan miliknya	
2	Konsumen menekan tombol "Cetak/Simpan PDF".	
3		Sistem menampilkan dialog cetak dari peramban.
4	Konsumen memilih untuk mencetak dokumen atau menyimpannya dalam format PDF.	
Kondisi Akhir: Dokumen berhasil dicetak atau disimpan sebagai PDF melalui fasilitas cetak peramban.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	Dokumen PO atau Invoice belum tersedia.	Sistem belum menampilkan tombol Cetak/Simpan PDF pada antarmuka.

Tabel 4.14 Skenario UC-08: Melakukan Pembayaran DP dan Pelunasan

Identifikasi		
Use Case ID	UC-08	
Nama Use Case	Melakukan Pembayaran DP dan Pelunasan	
Aktor	Konsumen	
Prekondisi	Untuk pembayaran DP, Konsumen telah menyetujui draf PO dan pesanan berstatus Menunggu DP. Untuk pelunasan, pembayaran DP telah berstatus <i>paid</i> , seluruh tahapan produksi berstatus Selesai, dan <i>Invoice</i> pelunasan telah diterbitkan.	
Postkondisi	Transaksi pembayaran tercatat pada Midtrans dan sistem menunggu notifikasi pembayaran melalui <i>webhook</i> .	
Tujuan	Memfasilitasi pembayaran bertahap dua termin secara aman melalui Midtrans Snap.	
Skenario Utama		
Kondisi Awal: Halaman Tagihan/Invoice		
No	Aktor	Sistem
1	Konsumen menekan tombol “Bayar Sekarang” pada tagihan DP atau pelunasan.	
2		Sistem memvalidasi status pesanan, termin pembayaran, dan syarat pembayaran.
3		Sistem meminta atau mengambil <i>Snap Token</i> dari Midtrans.
4		Sistem menampilkan antarmuka pembayaran Midtrans Snap.
5	Konsumen memilih metode pembayaran yang tersedia.	
6	Konsumen menyelesaikan pembayaran sesuai metode yang dipilih.	
7		Sistem menunggu notifikasi pembayaran melalui <i>webhook</i> Midtrans.
Kondisi Akhir: Transaksi pembayaran menunggu validasi <i>webhook</i> .		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
4a	Konsumen menutup antarmuka Midtrans Snap tanpa membayar.	Sistem mempertahankan status pembayaran sebagai <i>pending</i> .
2a	Konsumen mencoba membayar pelunasan sebelum seluruh tahapan produksi selesai.	Sistem menolak pembuatan transaksi pelunasan dan menampilkan pesan bahwa seluruh tahapan produksi harus selesai.

Tabel 4.15 Skenario UC-09: Validasi Pembayaran Otomatis

Identifikasi		
Use Case ID	UC-09	
Nama Use Case	Validasi Pembayaran Otomatis	
Aktor	Sistem Midtrans	
Prekondisi	Midtrans mengirimkan <i>webhook</i> setelah Konsumen melakukan pembayaran.	
Postkondisi	Status <i>Invoice</i> diperbarui sesuai status transaksi. Jika DP berstatus <i>paid</i> , status PO menjadi DP Diterima dan sistem menyiapkan empat tahapan produksi berstatus Belum Dimulai. Jika pelunasan berstatus <i>paid</i> dan seluruh tahapan telah selesai, status PO menjadi Siap Dikirim.	
Tujuan	Memvalidasi transaksi pembayaran secara otomatis dan mencegah manipulasi status pembayaran.	
Skenario Utama		
Kondisi Awal: Transaksi Berhasil di Sisi Bank		
No	Aktor	Sistem
1	Midtrans mengirimkan <i>HTTP POST payload</i> notifikasi pembayaran.	
2		Sistem menerima data pada <i>Route Handler webhook</i> .
3		Sistem memvalidasi <i>signature key</i> , <i>order ID</i> , nominal transaksi (<i>gross amount</i>), dan status transaksi.
4		Sistem memperbarui status <i>Invoice</i> sesuai data transaksi yang valid.
5		Jika pembayaran merupakan DP berstatus <i>paid</i> , sistem mengubah status PO menjadi DP Diterima dan menyiapkan empat tahapan produksi.
6		Jika pembayaran merupakan pelunasan berstatus <i>paid</i> serta seluruh tahapan produksi telah selesai, sistem mengubah status PO menjadi Siap Dikirim.
Kondisi Akhir: Status pembayaran dan status pesanan diperbarui pada basis data.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	<i>Signature key</i> tidak valid.	Sistem menolak <i>webhook</i> dan mengembalikan respons HTTP 400.
3b	<i>Order ID</i> atau <i>gross amount</i> tidak sesuai dengan data <i>Invoice</i> .	Sistem menolak <i>webhook</i> dan tidak memperbarui data pembayaran.

Tabel 4.16 Skenario UC-10: Memperbarui Tahapan Produksi

Identifikasi		
Use Case ID	UC-10	
Nama Use Case	Memperbarui Tahapan Produksi	
Aktor	Kepala Produksi	
Prekondisi	Pembayaran DP telah terverifikasi dan status pesanan berada pada tahap DP Diterima atau Proses Perakitan.	
Postkondisi	Status tahapan produksi, waktu pembaruan, pengguna yang melakukan pembaruan, catatan, dan foto dokumentasi tersimpan pada basis data.	
Tujuan	Memungkinkan Kepala Produksi memperbarui tahapan produksi secara terurut sesuai ketentuan sistem.	
Skenario Utama		
Kondisi Awal: Halaman Dashboard Produksi.		
No	Aktor	Sistem
1	Kepala Produksi memilih pesanan yang telah memasuki tahap produksi.	
2		Sistem menampilkan empat tahapan produksi, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai.
3	Kepala Produksi memilih tahapan berstatus Belum Dimulai dan menekan tombol "Mulai Tahapan".	
4		Sistem memvalidasi urutan tahapan. Tahapan berikutnya hanya dapat dimulai jika tahapan sebelumnya telah selesai.
5		Sistem mengubah status tahapan menjadi Sedang Diproses serta menyimpan waktu dan identitas Kepala Produksi yang memulai tahapan.
6	Kepala Produksi melengkapi catatan dan foto dokumentasi apabila diperlukan, lalu menekan tombol "Selesaikan Tahapan".	
7		Sistem mengubah status tahapan menjadi Selesai serta menyimpan waktu, identitas pengguna, catatan, dan foto dokumentasi.
8		Jika seluruh tahapan berstatus Selesai, sistem menerbitkan satu Invoice pelunasan. Status pesanan tetap Proses Perakitan sampai pembayaran pelunasan berhasil diverifikasi.
Kondisi Akhir: Data tahapan produksi diperbarui pada halaman Dashboard Produksi.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Kepala Produksi mencoba memulai tahapan berikutnya sebelum tahapan sebelumnya selesai.	Sistem menolak pembaruan dan menampilkan pesan bahwa tahapan sebelumnya harus diselesaikan terlebih dahulu.
6a	Kepala Produksi mencoba menyelesaikan tahapan yang belum dimulai.	Sistem menolak pembaruan dan menampilkan pesan bahwa tahapan harus dimulai terlebih dahulu.
6b	Kepala Produksi mencoba mengubah tahapan yang telah berstatus Selesai.	Sistem menolak pembaruan karena tahapan yang selesai tidak dapat diubah kembali.

Tabel 4.17 Skenario UC-11: Melacak Tahapan Produksi

Identifikasi		
Use Case ID	UC-11	
Nama Use Case	Melacak Tahapan Produksi	
Aktor	Konsumen	
Prekondisi	Pembayaran DP telah terverifikasi dan sistem telah menyiapkan tahapan produksi untuk pesanan Konsumen.	
Postkondisi	Konsumen dapat melihat status tahapan produksi dan dokumentasi berdasarkan pembaruan Kepala Produksi.	
Tujuan	Menyediakan transparansi informasi tahapan produksi kepada Konsumen.	
Skenario Utama		
Kondisi Awal: Halaman Riwayat Pesanan Konsumen		
No	Aktor	Sistem
1	Konsumen memilih pesanan aktif miliknya.	
2	Konsumen menekan tombol "Lacak Tahapan Produksi".	
3		Sistem memvalidasi kepemilikan pesanan Konsumen.
4		Sistem mengambil data tahapan produksi, status tahapan, catatan, foto dokumentasi, serta riwayat waktu dan pengguna yang melakukan pembaruan.
5		Sistem menampilkan tahapan produksi secara berurutan dalam bentuk <i>checklist</i> .
Kondisi Akhir: Halaman pelacakan tahapan produksi ditampilkan.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Konsumen mencoba membuka pesanan milik pengguna lain.	Sistem menolak akses terhadap data pesanan.
4a	Tahapan produksi belum dimulai setelah pembayaran DP diverifikasi.	Sistem menampilkan empat tahapan dengan status Belum Dimulai dan keterangan bahwa tahapan akan diperbarui oleh Kepala Produksi.

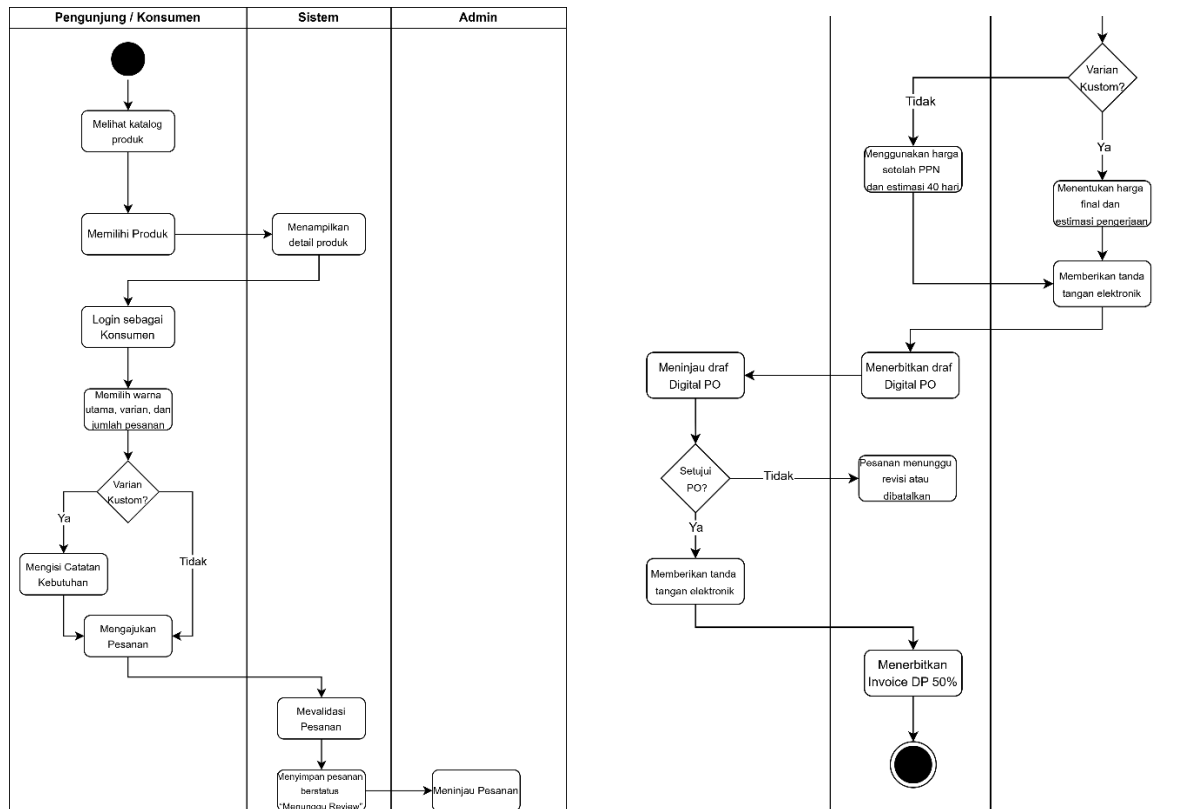
4.2.3 Activity Diagram

Activity Diagram merupakan pemodelan yang menggambarkan alur kerja (workflow) atau aktivitas dari sebuah sistem atau proses bisnis. Diagram ini menunjukkan bagaimana sebuah aktivitas dimulai, keputusan apa yang mungkin terjadi di tengah alur, dan bagaimana aktivitas tersebut berakhir.

Pada perancangan sistem e-commerce PT Marlip Indo Mandiri, Activity Diagram dibuat untuk memvisualisasikan dua alur proses utama yang menjadi fokus pengembangan sistem.

A. Activity Diagram Pemesanan Standar dan Kustom

Alur ini menggambarkan proses ketika Pengunjung melihat katalog produk, kemudian login sebagai Konsumen untuk mengajukan pemesanan kendaraan varian Standar atau Kustom hingga sistem menerbitkan Invoice DP.

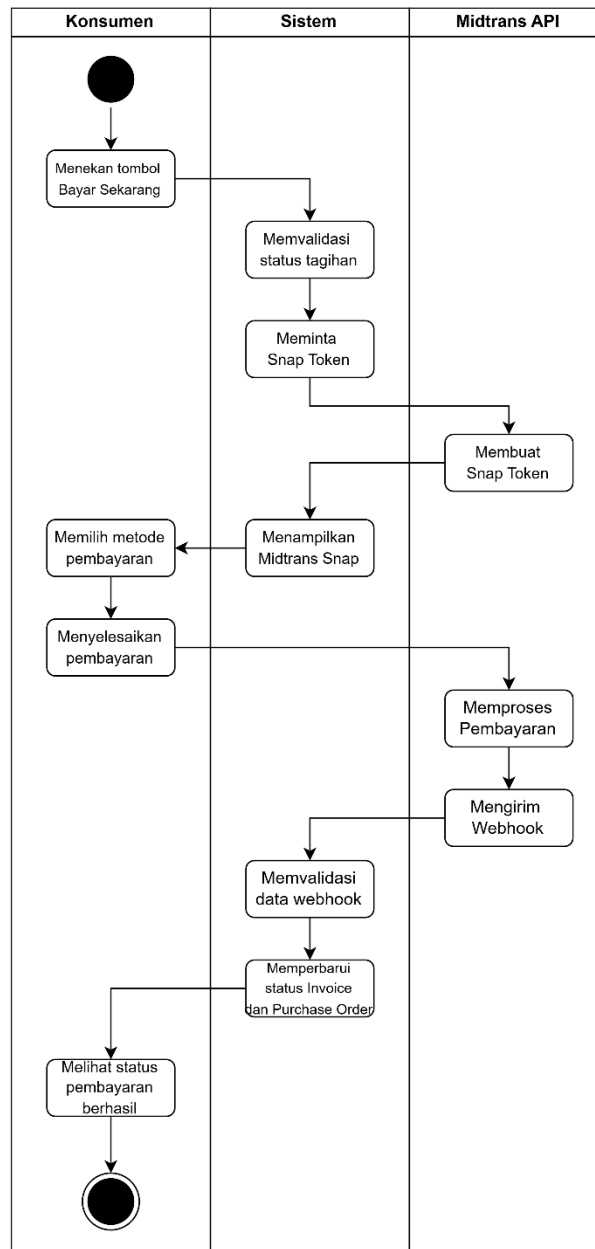


Gambar 4.4 Activity Diagram Pemesanan Standar dan Kustom

Berdasarkan Gambar 4.4, alur pemesanan dimulai ketika Pengunjung atau Konsumen melihat katalog dan memilih produk kendaraan listrik. Setelah detail produk ditampilkan, pengguna harus login sebagai Konsumen untuk mengajukan pesanan. Konsumen kemudian memilih warna utama, varian Standar atau Kustom, serta jumlah pesanan. Apabila memilih varian Kustom, Konsumen wajib mengisi catatan kebutuhan khusus. Sistem memvalidasi dan menyimpan pesanan dengan status “Menunggu Review”. Admin kemudian meninjau pesanan tersebut. Pada varian Kustom, Admin menentukan harga final dan estimasi pengerjaan, sedangkan pada varian Standar sistem menggunakan harga setelah PPN dan estimasi 40 hari secara otomatis. Setelah Admin memberikan tanda tangan elektronik, sistem menerbitkan draf Digital PO untuk ditinjau Konsumen. Jika Konsumen menyetujui PO dan memberikan tanda tangan elektronik, sistem menerbitkan Invoice DP sebesar 50%.

B. Activity Diagram Pembayaran Menggunakan Midtrans

Alur ini memodelkan tahapan pembayaran bertahap, yaitu Down Payment (DP) sebesar 50% dan pelunasan, yang dilakukan oleh Konsumen melalui integrasi gerbang pembayaran Midtrans.



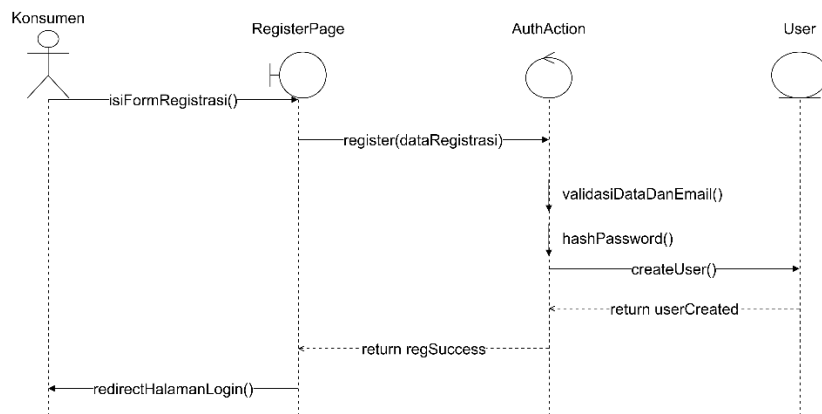
Gambar 4.5 Activity Diagram Pembayaran via Midtrans

Berdasarkan Gambar 4.5, alur pembayaran dimulai ketika Konsumen menekan tombol Bayar Sekarang pada tagihan yang tersedia. Sistem memvalidasi status tagihan, kemudian meminta *Snap Token* kepada Midtrans. Setelah Midtrans membuat *Snap Token*, sistem menampilkan antarmuka Midtrans Snap agar Konsumen dapat memilih metode pembayaran dan menyelesaikan transaksi. Setelah transaksi diproses, Midtrans mengirimkan *webhook* pembayaran kepada sistem. Sistem memvalidasi *order ID*, nominal transaksi, status transaksi, dan *signature key*, kemudian memperbarui status Invoice serta Purchase Order sesuai termin pembayaran. Pada pembayaran DP, status pesanan berubah menjadi “DP Diterima” dan pesanan memasuki tahap produksi awal. Pada pembayaran pelunasan, status pesanan berubah menjadi “Siap Dikirim” apabila seluruh tahapan produksi telah dinyatakan selesai oleh Kepala Produksi.

4.2.4 Sequence Diagram

Sequence Diagram digunakan untuk memodelkan aliran logika antar objek dalam sistem berdasarkan urutan waktu. Pemodelan ini menggunakan pendekatan Boundary-Control-Entity untuk menggambarkan interaksi antara halaman antarmuka, route handler, dan entitas data pada sistem berbasis Next.js App Router. Berikut adalah urutan aktivitas untuk setiap *use case* yang diusulkan:

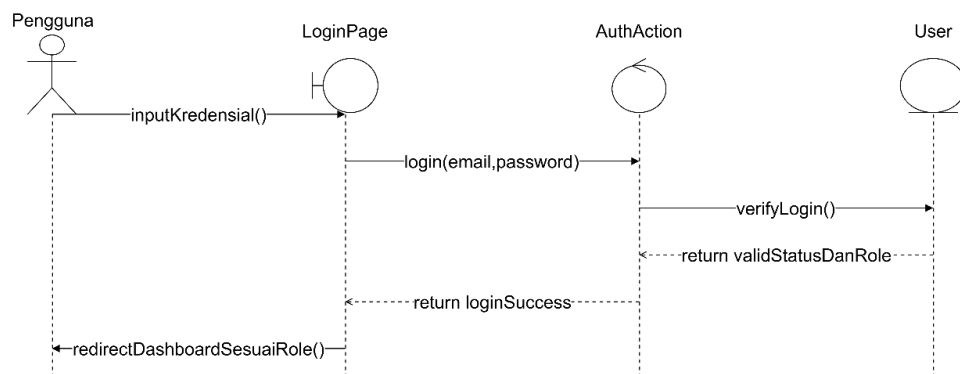
4.2.4.1 Diagram Sequence UC-01 - Registrasi Akun



Gambar 4.6 Diagram Sequence UC-01 - Registrasi Akun

Penjelasan Alur: Proses registrasi dimulai ketika Konsumen mengisi formulir registrasi pada halaman autentikasi. Sistem memvalidasi kelengkapan data, memeriksa ketersediaan email, melakukan hashing password, lalu menyimpan data pengguna baru pada entitas User. Setelah registrasi berhasil, sistem mengarahkan Konsumen ke halaman login.

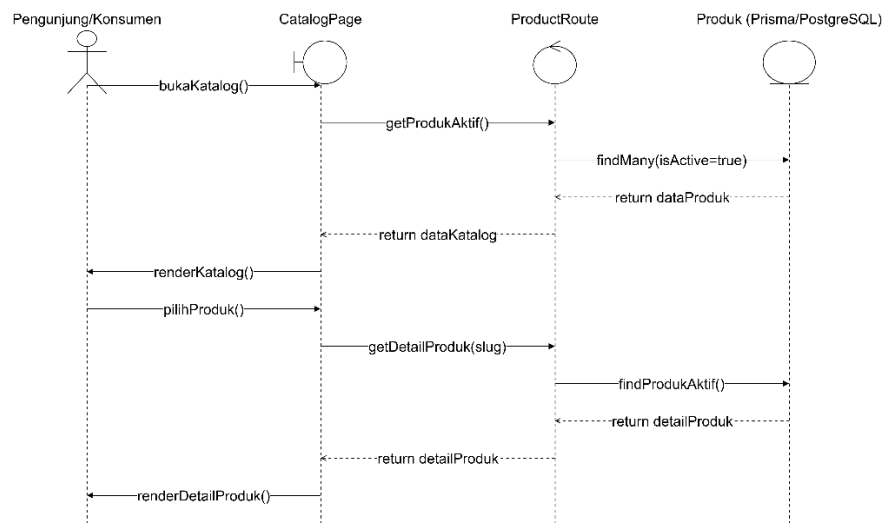
4.2.4.2 Diagram Sequence UC-02 – Login



Gambar 4.7 Diagram Sequence UC-02 - Login

Penjelasan Alur: Proses autentikasi dimulai ketika pengguna mengirimkan inputKredensial() pada :LoginPage. :AuthController menerima signInCredentials() dan memanggil findUserByEmail() pada entitas :User. Setelah data ditemukan, sistem menjalankan comparePasswordHash() untuk verifikasi keamanan. Jika valid, sistem membuat sesi autentikasi dan mengarahkan pengguna ke dashboard sesuai perannya, yaitu Konsumen, Admin, atau Kepala Produksi.

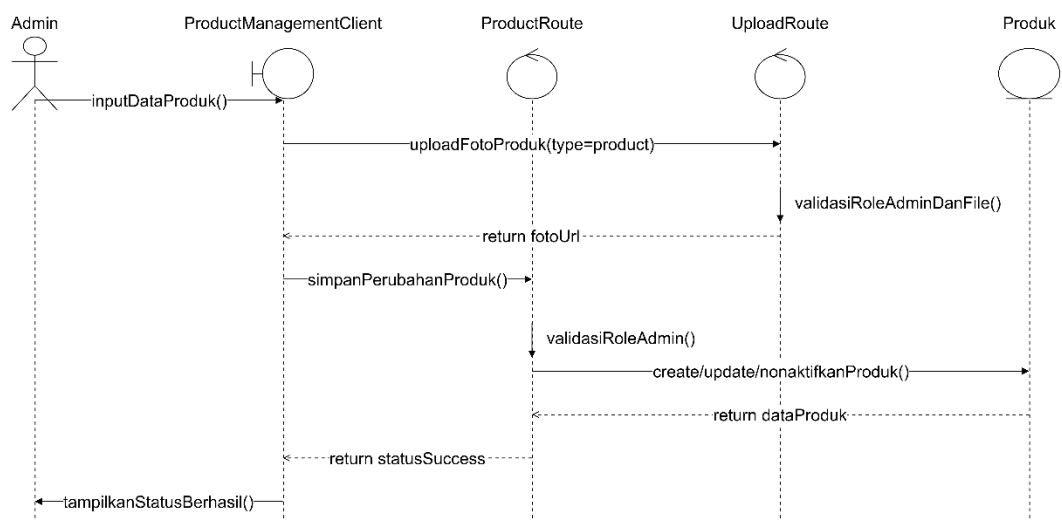
4.2.4.3 Diagram Sequence UC-03 - Melihat Katalog Produk



Gambar 4.8 Diagram Sequence UC-03 - Melihat Katalog Produk

Penjelasan Alur: Pengunjung atau Konsumen membuka halaman katalog produk. Sistem mengambil data produk aktif dari entitas Produk melalui Prisma/PostgreSQL, kemudian menampilkan daftar produk berdasarkan kategori Kendaraan, EV Bike, Alat Pembelajaran, dan Suku Cadang. Pengunjung atau Konsumen dapat memilih salah satu produk untuk melihat detail produk.

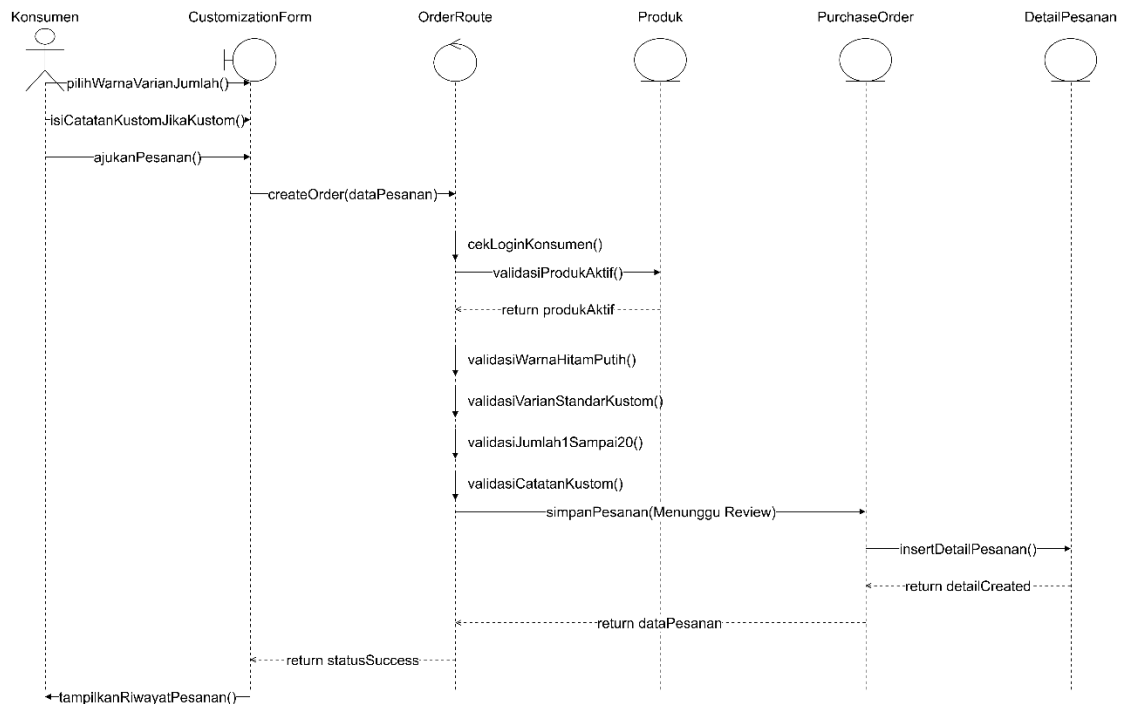
4.2.4.4 Diagram Sequence UC-04 - Mengelola Katalog Produk



Gambar 4.9 Diagram Sequence UC-04 - Mengelola Katalog Produk

Penjelasan Alur: Admin memulai dengan pilihTambahAtauSunting() dan unggahFotoProduk() pada :DashboardAdminPage. Sistem memanggil :UploadRoute untuk memproses uploadFileFoto() dan mengembalikan fotoUrl. Admin kemudian menjalankan simpanDataProduk(), yang diproses oleh :ProductRoute melalui createAtauUpdateProduk() dan disimpan ke entitas :Produk melalui saveProdukDanSpesifikasi(). Fitur softDelete(isActive=false) digunakan untuk menonaktifkan produk tanpa menghapus data secara permanen.

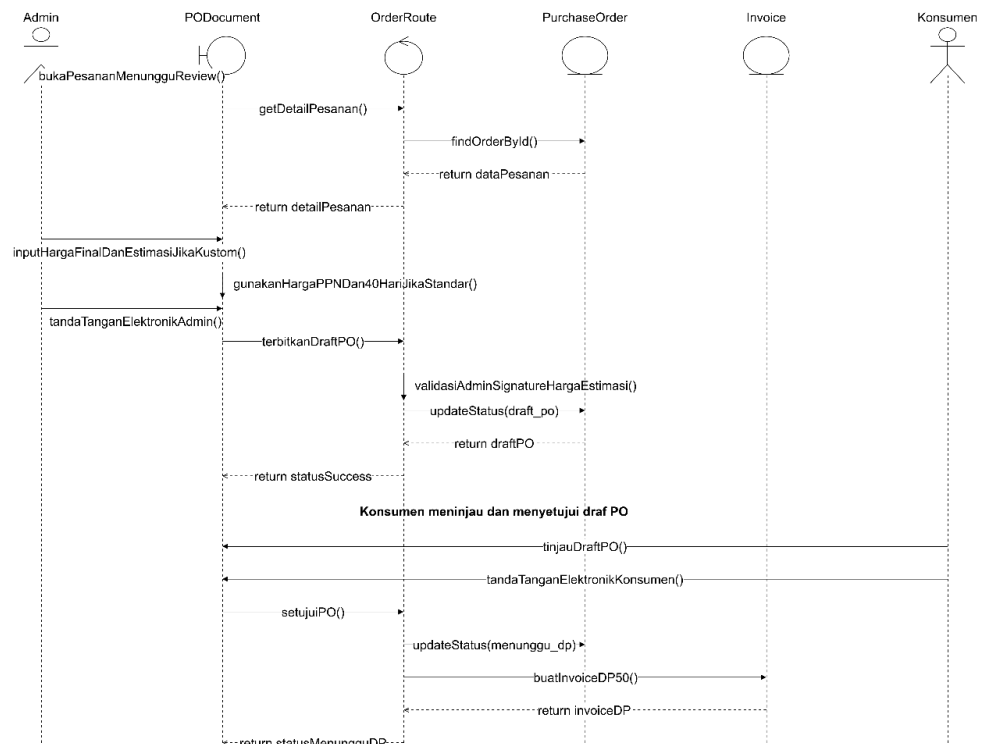
4.2.4.5 Diagram Sequence UC-05 - Mengajukan Pemesanan Standar atau Kustom



Gambar 4.10 Diagram Sequence UC-05 - Mengajukan Pemesanan Standar atau Kustom

Penjelasan Alur: Konsumen mengisi formulir pemesanan pada halaman detail produk dengan memilih warna utama Hitam atau Putih, varian Standar atau Kustom, serta jumlah pesanan 1-20 unit. Jika memilih varian Kustom, Konsumen wajib mengisi catatan kebutuhan khusus. Sistem memvalidasi status login, produk aktif, data pilihan, dan jumlah pesanan. Setelah valid, sistem menyimpan data Purchase Order dan Detail Pesanan dengan status Menunggu Review.

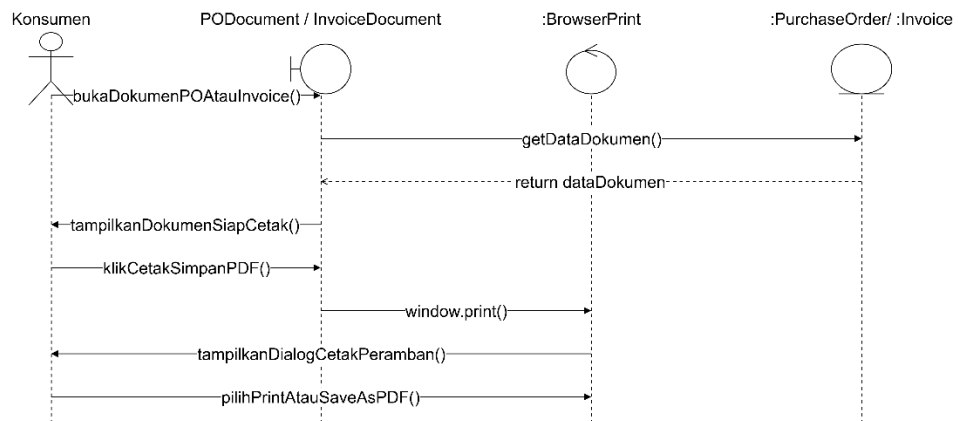
4.2.4.6 Diagram Sequence UC-06 - Mengelola Pesanan & Menerbitkan PO



Gambar 4.11 Diagram Sequence UC-06 - Mengelola Pesanan & Menerbitkan PO

Penjelasan Alur: Admin membuka pesanan berstatus Menunggu Review dan meninjau rincian pesanan. Untuk varian Kustom, Admin mengisi harga final dan estimasi pengerjaan. Untuk varian Standar, sistem menggunakan harga setelah PPN dan estimasi 40 hari secara otomatis. Admin memberikan tanda tangan elektronik dan menerbitkan draf PO. Konsumen kemudian meninjau draf PO, memberikan tanda tangan elektronik, dan menyetujui PO sehingga status pesanan berubah menjadi Menunggu DP dan sistem menerbitkan Invoice DP sebesar 50%.

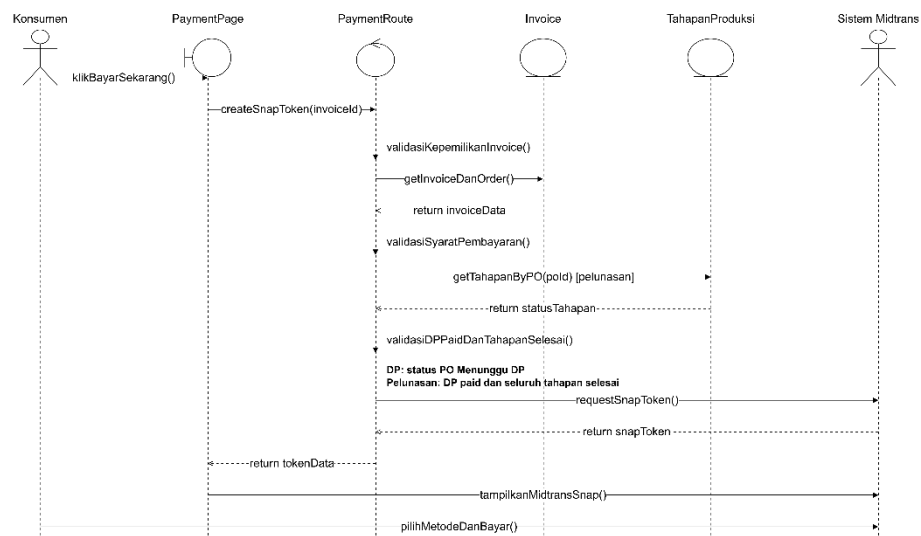
4.2.4.7 Diagram Sequence UC-07 - Mencetak atau Menyimpan Dokumen PO & Invoice



Gambar 4.12 Diagram Sequence UC-07 - Mencetak atau Menyimpan Dokumen PO & Invoice

Penjelasan Alur: Konsumen membuka dokumen PO atau Invoice pada halaman dokumen transaksi. Sistem mengambil data Purchase Order dan Invoice yang terkait, lalu menampilkan dokumen dalam format siap cetak. Ketika Konsumen memilih fitur Cetak atau Simpan PDF, sistem menjalankan fungsi cetak peramban sehingga dokumen dapat dicetak atau disimpan sebagai PDF.

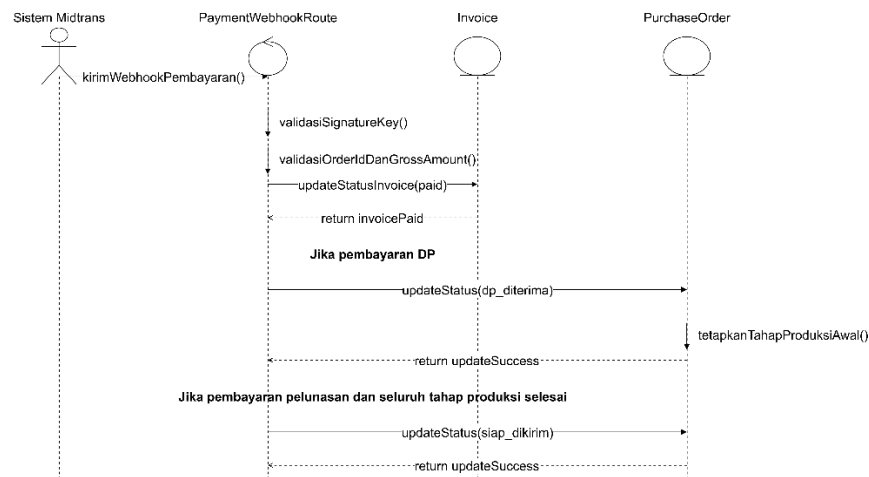
4.2.4.8 Diagram Sequence UC-08 - Melakukan Pembayaran (DP & Lunas)



Gambar 4.13 Diagram Sequence UC-08 - Melakukan Pembayaran (DP & Lunas)

Penjelasan Alur: Konsumen memilih termin DP atau pelunasan dan menekan tombol Bayar Sekarang pada :PaymentPage. :PaymentRoute memvalidasi kepemilikan *Invoice* dan syarat pembayaran, kemudian meminta *Snap Token* kepada <<Sistem Midtrans>>. Pembayaran DP hanya dapat dilakukan ketika status pesanan Menunggu DP. Pembayaran pelunasan hanya dapat dilakukan setelah pembayaran DP berstatus *paid* dan seluruh tahapan produksi telah selesai. Setelah *Snap Token* diterima, sistem menampilkan antarmuka Midtrans Snap agar Konsumen dapat memilih metode pembayaran dan menyelesaikan transaksi. Status akhir pembayaran diperbarui melalui *webhook* Midtrans pada proses UC-09.

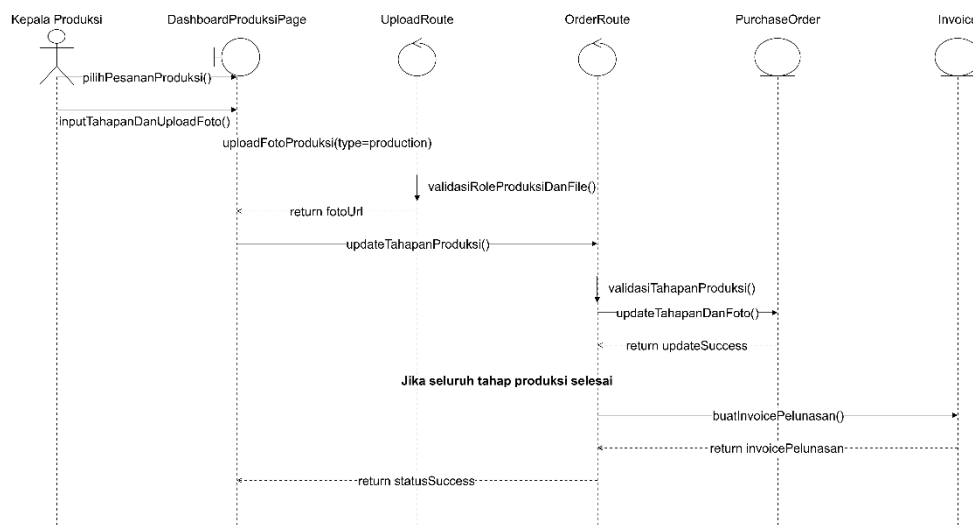
4.2.4.9 Diagram Sequence UC-09 - Validasi Pembayaran Otomatis



Gambar 4.14 Diagram Sequence UC-09 - Validasi Pembayaran Otomatis

Penjelasan Alur: Sistem Midtrans mengirimkan *webhook* pembayaran ke PaymentWebhookRoute. Sistem memvalidasi *signature key*, *order ID*, dan *gross amount* sebelum memperbarui status Invoice. Jika pembayaran merupakan DP, sistem mengubah status Purchase Order menjadi DP Diterima dan menetapkan tahap produksi awal. Jika pembayaran merupakan pelunasan dan seluruh tahap produksi telah selesai, sistem mengubah status Purchase Order menjadi Siap Dikirim.

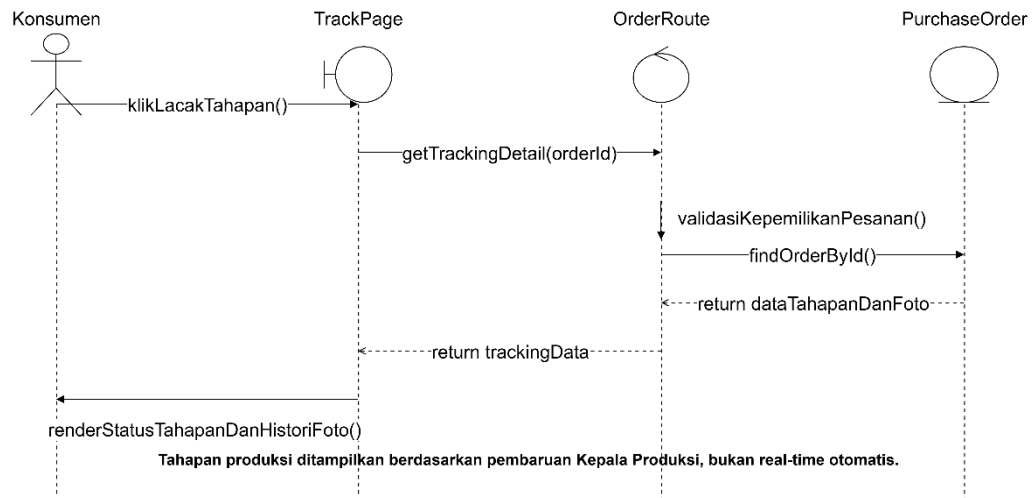
4.2.4.10 Diagram Sequence UC-10 – Memperbarui Tahapan Produksi



Gambar 4.15 Diagram Sequence UC-10 – Memperbarui Tahapan Produksi

Penjelasan Alur: Kepala Produksi memilih pesanan yang telah memasuki proses produksi, memperbarui status tahapan produksi, dan mengunggah foto dokumentasi. Sistem memvalidasi role Kepala Produksi, format file, serta tahapan produksi yang dipilih berdasarkan ketentuan proses produksi perusahaan. Setelah valid, sistem menyimpan status tahapan dan foto dokumentasi pada Purchase Order. Jika seluruh tahapan produksi telah selesai, sistem menerbitkan Invoice pelunasan, tetapi status pesanan tetap Proses Perakitan sampai pelunasan berhasil dibayar.

4.2.4.11 Diagram Sequence UC-11 - Melacak Tahapan Produksi

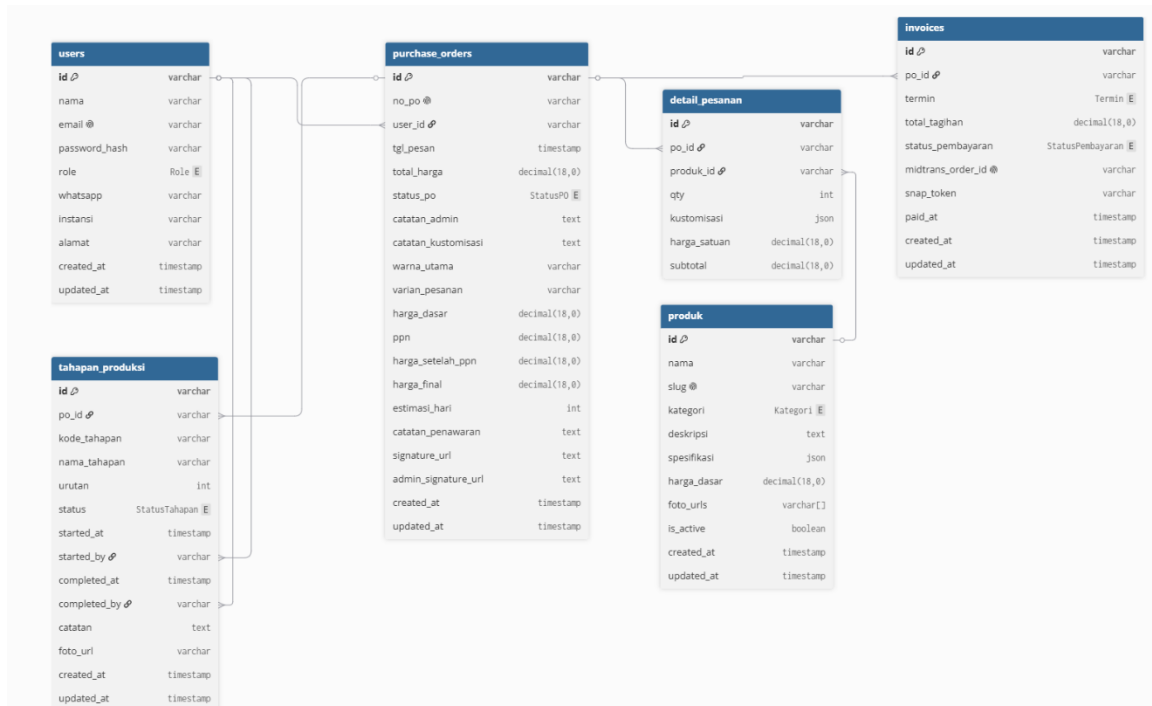


Gambar 4.16 Diagram Sequence UC-11 – Melacak Tahapan Produksi

Penjelasan Alur: Konsumen membuka halaman pelacakan pesanan. Sistem memvalidasi kepemilikan pesanan, mengambil data Purchase Order, status tahapan produksi, dan foto dokumentasi produksi. Data tersebut kemudian ditampilkan dalam bentuk status tahapan dan riwayat dokumentasi. Informasi tahapan produksi berasal dari pembaruan Kepala Produksi, bukan dari pelacakan *real-time* otomatis.

4.2.5 Entity Relationship Diagram

Entity Relationship Diagram (ERD) digunakan untuk memodelkan struktur logis dan arsitektur penyimpanan data pada sistem. Perancangan ERD bertujuan menjaga keterkaitan dan integritas data dalam proses pemesanan kendaraan listrik, pengelolaan varian Standar atau Kustom, penerbitan dokumen *Purchase Order*, pembayaran melalui Midtrans, serta pelacakan tahapan produksi.



Gambar 4.17 Entity Relationship Diagram (ERD)

Gambar 4.17 menyajikan rancangan basis data yang terdiri dari enam entitas utama, yaitu users, produk, purchase_orders, detail_pesanan, invoices, dan tahapan_produksi. Relasi antarentitas dijelaskan sebagai berikut:

1. **Relasi users dan purchase_orders (1:N / One-to-Many)** Entitas users menyimpan data autentikasi dan profil pengguna, yaitu Konsumen, Admin, dan Kepala Produksi. Satu Konsumen dapat memiliki banyak data pemesanan kendaraan listrik. Relasi tersebut dibentuk melalui foreign key user_id pada entitas purchase_orders.
2. **Relasi purchase_orders dan detail_pesanan (1:N/One-to-Many)** Entitas purchase_orders merupakan tabel transaksi utama yang menyimpan informasi pesanan, status PO, varian pesanan, warna utama, harga, estimasi pengerjaan, catatan, dan tanda tangan elektronik. Satu Purchase Order dapat memiliki satu atau lebih rincian pesanan yang disimpan pada entitas detail_pesanan melalui foreign key po_id.
3. **Relasi produk dan detail_pesanan (1:N / One-to-Many)** Entitas produk menyimpan data katalog produk, kategori, spesifikasi, harga dasar, foto produk, dan status aktif produk. Satu produk dapat tercatat pada banyak data detail_pesanan melalui foreign key produk_id.
4. **Relasi purchase_orders dan invoices (1:N / One-to-Many)** Entitas invoices menyimpan data tagihan pembayaran berdasarkan termin. Satu Purchase Order dapat memiliki maksimal dua invoice, yaitu invoice Down Payment (DP) dan invoice pelunasan.
5. **Relasi purchase_orders dan tahapan_produksi (1:N/One-to-Many)** Entitas tahapan_produksi menyimpan pelacakan produksi berbentuk checklist. Pesanan yang telah memasuki produksi memiliki empat tahapan, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai.

6. **Relasi users dan tahapan_produksi (1:N / One-to-Many)** Entitas tahapan_produksi berelasi dengan users melalui atribut started_by dan completed_by untuk mencatat Kepala Produksi serta waktu pembaruan setiap tahapan.

Pada rancangan ini, atribut nominal seperti harga_dasar, total_harga, ppn, harga_setelah_ppn, harga_final, harga_satuan, subtotal, dan total_tagihan menggunakan tipe data Decimal(18,0). Penggunaan tipe data tersebut menjaga nilai mata uang Rupiah tetap konsisten tanpa nilai pecahan pada proses transaksi pembayaran.

4.2.6 Perancangan Antarmuka

Tabel 4.18 Halaman Beranda

Halaman Beranda

[Beranda](#)
[Katalog](#)
[Tentang](#)

[Masuk](#)
[Daftar](#)

PT Marlip Indo Mandiri — Est. 2004

Bandung, Indonesia

Kendaraan Listrik Dirancang Khusus untuk Anda.

500+

Klien Puas

220+

Unit Terjual

60+

Teknisi Berpengalaman

20+

Kota Dijangkau

Solusi Mobilitas Listrik yang Lebih Cerdas dan Berkelanjutan

☐ Ramah Lingkungan

☐ Hemat Biaya Operasional

☐ Kustomisasi Penuh

☐ Produksi 100% Lokal

Proses Pemesanan Empat Langkah

1. Kustomisasi Spesifikasi
2. Purchase Order Digital
3. Pembayaran Bertahap
4. Pantau Progres Langsung

Produk Terlaris Kami

MESA - EV City Car

Rp 185.000.000

Marlip Gully

Rp 125.000.000

Marlip e-Motor Roda 3

Rp 35.000.000

Dipercaya oleh Berbagai Instansi

BANK ABRI

Bank Militer Indonesia

RIKTES

Industri Kereta Api Indonesia

DAKPA

Daerah Khusus Ibukota Jakarta

Pertanyaan yang Sering Ditanyakan

Apakah kendaraan listrik layak?

Bagaimana cara pengisian listrik?

Bagaimana cara perawatan?

Mengapa kendaraan listrik lebih murah dari mobil konvensional?

Apakah kendaraan listrik aman untuk anak-anak?

Apakah kendaraan listrik ramah lingkungan?

Konsultasikan Kebutuhan Armada Anda

[Konsultasi Gratis](#) [Lihat Katalog Lengkap](#)

Marlip

PRODUK KAMI

Kendaraan Listrik EV Roda 3

EV Roda 3

Kendaraan Listrik EV Roda 3

Kendaraan Listrik EV Roda 3

PERUSAHAAN

PT Marlip Indo Mandiri

Jl. Raya Bandung No. 123

Bandung, Jawa Barat 40132

Telp: +62 22 12345678

Email: info@marlip.co.id

INFORMASI KONTAK

Jl. Raya Bandung No. 123

Bandung, Jawa Barat 40132

Telp: +62 22 12345678

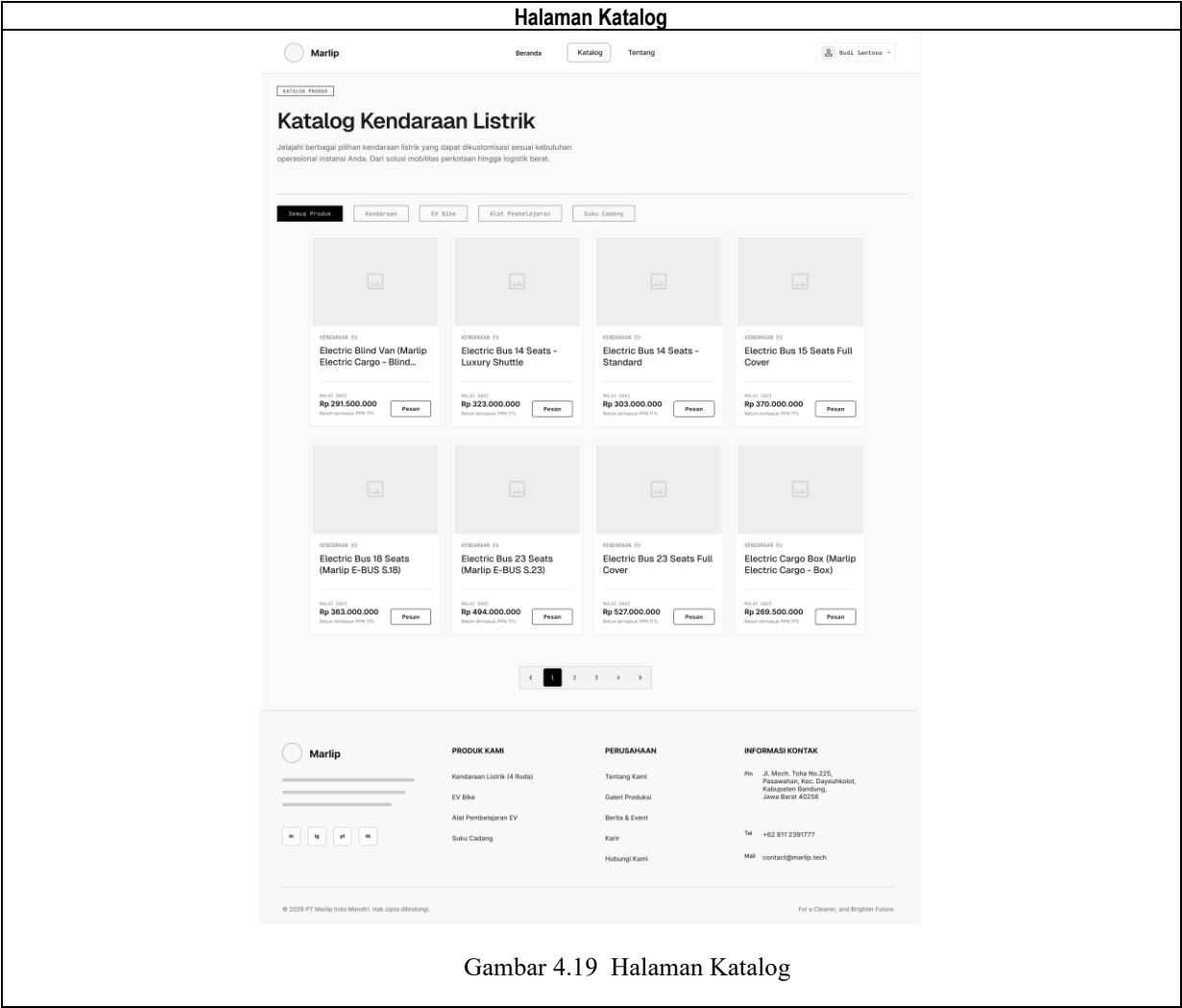
Email: info@marlip.co.id

Gambar 4.18 Halaman Beranda

Deskripsi

Halaman ini merupakan tampilan awal aplikasi yang dapat diakses oleh Pengunjung. Halaman menampilkan informasi perusahaan, produk unggulan, layanan yang tersedia, serta tombol untuk menuju katalog, registrasi, dan login.

Tabel 4.19 Halaman Katalog



Gambar 4.19 Halaman Katalog

Deskripsi

Halaman ini menampilkan daftar produk aktif yang dapat dilihat oleh Pengunjung maupun Konsumen. Informasi yang ditampilkan meliputi gambar produk, nama produk, kategori, harga dasar, dan informasi singkat. Pengguna dapat memilih salah satu produk untuk melihat detail produk.

Deskripsi

Halaman ini menampilkan galeri produk, nama produk, harga dasar, informasi PPN, dan spesifikasi teknis. Konsumen dapat memilih warna utama Hitam atau Putih, varian Standar atau Kustom, serta jumlah pesanan 1–20 unit. Catatan kebutuhan hanya ditampilkan dan wajib diisi apabila Konsumen memilih varian Kustom.

Tabel 4.21 Halaman Registrasi

Tabel 4.22 Halaman Login

Halaman Login
Gambar 4.22 Halaman Login
Deskripsi Halaman ini digunakan pengguna untuk memasukkan alamat email dan kata sandi. Sistem memverifikasi kredensial dan mengarahkan pengguna menuju dashboard sesuai dengan perannya, yaitu Konsumen, Admin, atau Kepala Produksi.

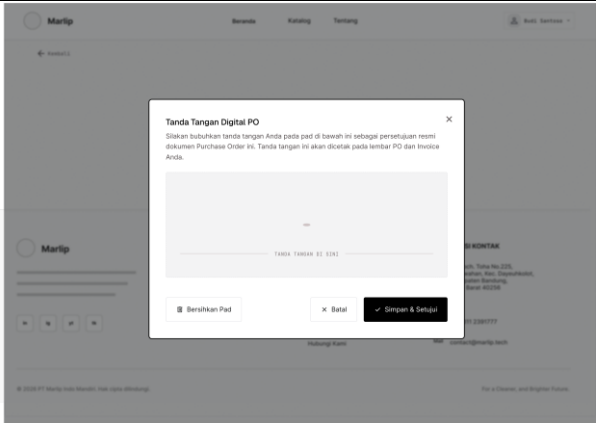
Tabel 4.23 Halaman Dashboard Konsumen

Halaman Dashboard Konsumen
Gambar 4.23 Halaman Dashboard Konsumen
Deskripsi Halaman ini menampilkan ringkasan dan daftar pesanan milik Konsumen berdasarkan statusnya. Konsumen dapat melihat nomor PO, status pesanan, ringkasan tahapan produksi, dokumen PO, <i>invoice</i>, pembayaran DP atau pelunasan, serta halaman pelacakan tahapan produksi.

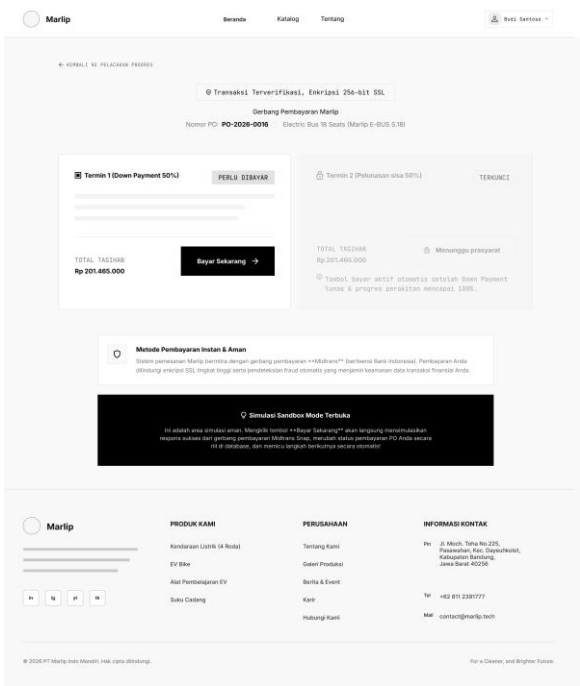
Tabel 4.24 Halaman Digital Purchase Order dan Persetujuan Draf PO

The screenshot displays a digital purchase order interface for Marlip. At the top, there's a navigation bar with 'Beranda', 'Katalog', and 'Tentang'. The main header shows the Marlip logo and a 'Buat Setorance' button. The page title is 'Halaman Digital Purchase Order dan Persetujuan Draft PO'. The main content area is titled 'Tinjau Draft Digital PO Anda' and contains a message from the Admin. Below this, there's a form for the purchase order. The form includes a table of items with columns for 'No', 'Model', 'Unit', 'Qty', 'Price', and 'Total Price'. The items listed are 'MESIN - EV CITY 500' with a unit price of 166,666.667 and a total price of 833,333.333. The form also includes a summary section with 'Subtotal: 833,333.333', 'PPN 10%: 91,666.667', and 'Total (Cum.): Rp 925,000.000'. There are sections for 'To (Buyer):' and 'To (Seller):' with fields for company name, address, and phone number. The 'To (Buyer):' section is filled with 'Budi Santoso' and 'Jl. Raya Suroboyo No. 100, Surabaya 60119, Indonesia'. The 'To (Seller):' section is filled with 'PT Marlip Indo Mandiri', 'Jl. Raya Suroboyo No. 100, Surabaya 60119, Indonesia', and '031-1234-5678'. There are also sections for 'NOTE' and 'APPROVED' with checkboxes for 'Status Payment', 'Status PO', 'Payment Term', 'Estimate Pengerjaan', and 'Delivery Location'. The 'APPROVED' section has a 'Persetujuan' button. At the bottom, there's a footer with '© 2023 PT Marlip Indo Mandiri. Hak-hak dilindungi.' and 'For a Cleaner, and Brighter Future.'.

Tabel 4.25 Halaman Tanda Tangan Elektronik Konsumen

Halaman Tanda Tangan Elektronik Konsumen	
	
Deskripsi Halaman atau modal ini digunakan Konsumen untuk memberikan tanda tangan elektronik sebagai bentuk persetujuan terhadap draf Purchase Order. Tanda tangan yang diberikan akan disimpan dan ditampilkan pada dokumen PO.	

Tabel 4.26 Halaman Pembayaran Dua Termin

Halaman Pembayaran Dua Termin	
	
Deskripsi Halaman ini menampilkan pilihan pembayaran dua termin, yaitu DP sebesar 50% dan pelunasan sebesar sisa tagihan. Pembayaran DP dapat dilakukan setelah Konsumen menyetujui draf PO, sedangkan pelunasan hanya tersedia setelah pembayaran DP berstatus <i>paid</i> dan seluruh tahapan produksi selesai.	

Tabel 4.27 Halaman Invoice Pembayaran

Marlip

Beranda

Katalog

Tentang

2

Budi Santoso

Kembali

Cetak / Simpan PDF

LOGO

MARLIP

PT MARLIP INDO MANDIRI

MPIBP : 0412 8763 8762

8000

MARLIP INDO MANDIRI

INVOICE

No : INV-cmptlsky5005uud8tjydeksa-17524...

Pembeli/ Buyer :

Budi Santoso

Dinas Lingkungan Hidup Bandung

-

konsumen@budi.com | +62813456789

Surat Pesanan/PO/SPK :

No :

PO-2026-0008

Tgl/ date :

26-Jun-26

Dikirim/ Ship to :

Alamat Instalasi konsumen sesuai pendaftaran.

No	Type	Quantity	Price	Subtotal
1	ELECTRIC BUS 16 SEATS (MARLIP 6-BUS 6-16) Lisensi (DP 0260) Warna Utama Hitam Tipean Premium Standar Estimasi Pengiriman: 45 hari	1	181.500.000	181.500.000

Harga Dasar181.500.000

PPN 1%10.965.000

Total TagihanRp 201.465.000

Terbilang :

DUA RATUS SATU JUTA ENMPAT RATUS ENAM PULUH LIMA RIBU RUPIAH

Jatuh Tempo : 30-Jul-26

Pagihari dua juta

INFORMASI PEMBELAYAN :

STATUS LUNAS (PAID)

Metode Pembayaran : 26-Jun-2026, 14:30

Tgl Lunas

KONFIRMASI BUDEN CUSTOMER

(Budi Santoso)

Bandung, 26-Jun-26

PT MARLIP INDO MANDIRI

Edin Juarna

Marlip

PRODUK KAMI

Kendaraan Listrik (4 Roda)

EV Bike

Alat Pembelajaran EV

Suku Cadang

PERUSAHAAN

Tentang Kami

Galeri Produk

Berita & Event

Hubungi Kami

INFORMASI KONTAK

Pe Jl. Moch. Toha No.225,
Pasawahan, Kec. Dayeuhkolot,
Kabupaten Bandung,
Jawa Barat 40256

Tel +62 811 2391777

Mail contact@marlip.tech

© 2026 PT Marlip Indo Mandiri. Hak cipta dilindungi.

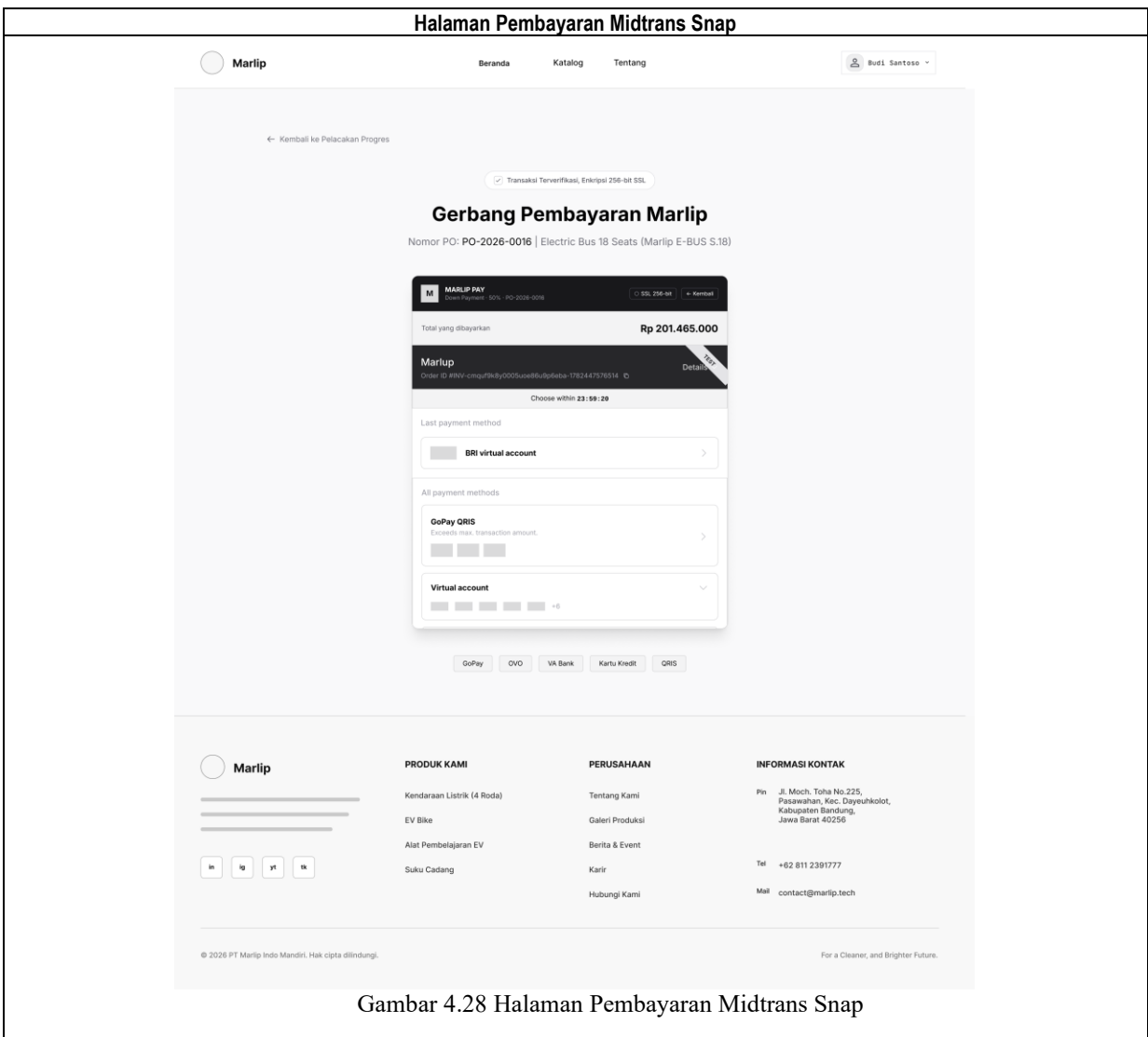
For a Cleaner, and Brighter Future.

Gambar 4.27 Halaman Invoice Pembayaran

Deskripsi

Halaman ini menampilkan informasi tagihan pembayaran yang meliputi nomor invoice, termin pembayaran, total tagihan, status pembayaran, dan waktu pembayaran. Konsumen dapat melanjutkan pembayaran apabila tagihan belum dibayar, serta mencetak atau menyimpan invoice sebagai PDF melalui fungsi cetak peramban.

Tabel 4.28 Halaman Pembayaran Midtrans Snap



Deskripsi	
-----------	--

Halaman ini menampilkan antarmuka pembayaran Midtrans Snap setelah Konsumen menekan tombol Bayar Sekarang. Melalui antarmuka ini, Konsumen dapat memilih metode pembayaran yang tersedia dan menyelesaikan transaksi sesuai termin tagihan.

Tabel 4.29 Halaman Pelacakan Tahapan Produksi

The image is a screenshot of a web application for 'Marlip', showing a production progress page for an 'ELECTRIC BUS 18 SEATS (MARLIP E-BUS S.18)'. The page is divided into several sections. At the top, there's a header with the Marlip logo, navigation links (Beranda, Katalog, Tentang), and a user profile (Budi Santoso). Below the header, the order ID 'PO-2028-0016' is displayed. The main title is 'ELECTRIC BUS 18 SEATS (MARLIP E-BUS S.18)' with a subtitle 'Dipesan pada 26 Juni 2026'. There are two buttons: 'Lihat Dokumen PO' and 'Pembayaran'. The 'Status Pemesanan' section shows a progress bar with four stages: 'PO Divalidasi' (completed), 'DP Diterima' (completed), 'Proses Perakitan' (80% progress), and 'Slap Dikirim' (pending). The 'Ringkasan Finansial' section shows 'Termin 1 (DP 50%)' for Rp 201.465.000 and 'Termin 2 (Pelunasan)' for Rp 201.465.000, with a total price of Rp 402.930.000. The 'Dokumentasi Tahapan Produksi (Tugas Akhir)' section shows four steps: 'Frame Assembly' (completed), 'Painting' (completed), 'Finishing' (in progress), and 'Delivery' (pending). The bottom section contains company information, products, and contact details.

Tabel 4.30 Halaman Dashboard Admin

Marlip

Administrator

MENU UTAMA

Overview

Manajemen Produk

Konsumen

Pengaturan

←

Kembali ke Website

ADMIN

Admin

AAdmin Marlip

Selamat datang, Admin

Berikut ringkasan data platform Marlip hari ini.

TOTAL PESANAN

2

Keseluruhan transaksi masuk

MENUNGGU REVIEW

1

Butuh tindakan segera

DP DITERIMA (PROGRES)

1

Sedang diproses di pabrik

SELESAI

0

Kendaraan siap dikirim

Pesanan Terbaru

Menampilkan 4 data terakhir

Order ID	Konsumen	Produk	Tanggal	Total (Rp)	Status	Aksi
PO-2026-0002	Budi Santoso (Dinas Lingkungan Hidup Bandung)	Marlip EV Truck	3 Jun 2026	Rp 285.000.000	Menunggu Review	Review PO
PO-2026-0001	Budi Santoso (Dinas Lingkungan Hidup Bandung)	MESA — EV City Car	3 Jun 2026	Rp 192.500.000	Proses Perakitan	Lacak Progres

Deskripsi

Halaman ini menampilkan ringkasan data produk dan pesanan yang perlu ditangani oleh Admin. Admin dapat mengakses fitur pengelolaan produk, melihat daftar pengajuan pesanan, serta membuka halaman peninjauan dan penerbitan draf PO.

Gambar 4.30 Halaman Dashboard Admin

Tabel 4.31 Halaman Manajemen Produk

Halaman Manajemen Produk

Gambar 4.31 Halaman Manajemen Produk

Deskripsi

Halaman ini digunakan Admin untuk menambahkan, melihat, memperbarui, dan menonaktifkan data produk. Informasi produk meliputi nama, kategori, harga dasar, spesifikasi, deskripsi, dan foto. Produk yang berstatus aktif akan ditampilkan pada katalog Konsumen.

Tabel 4.32 Halaman Pengelolaan Pesanan Admin

Halaman Pengelolaan Pesanan Admin

Marlip

Beranda
Katalog
Tentang

Admin Marlip

Tinjauan Kustomisasi Konsumen

Konsumen baru saja mengajukan spesifikasi kustomisasi di bawah. Silakan tinjau dan klik "Kirim Draft PO" jika sudah sesuai.

Formulir Penawaran Admin

Tentukan harga final dan estimasi pengerjaan untuk pesanan ini sebelum menerbitkan draft PO resmi.

INFORMASI RINCIAN PESANAN

Kategori / Varian: Vehicle / Standar

Jumlah Unit: 1 Unit

Harga Dasar Satuan: Rp 363.000.000

PPN 11% Satuan: Rp 38.930.000

Total Harga Standar: Rp 402.930.000

HARGA FINAL (STANDAR - READONLY)

Rp 402.930.000

Varian standar menggunakan harga katalog standar 0% secara default.

ESTIMASI WAKTU PENGERJAAN (STANDAR - READONLY)

40 Hari

Varian standar memiliki estimasi default 40 hari kerja.

CATATAN PENAWARAN ADMIN (OPTIONAL)

Masukkan catatan pendukung penawaran...

MARLIP
PT MARLIP 2020 PNEC20
Email: sales@marlip.tech
Telp: 0811-2391777

PURCHASES ORDER

Referensi : PO-2024-0034

Tanggal : 4 Juni 2024

Ship to (Buyer):

Budi Santoso

PTN : Budi Santoso

To (Seller):

PT Marlip Indo Mandiri

PTN : Admin Marlip

NO	MODEL	UNIT	QTY	PRICE	TOTAL PRICE
1	NESA - EV CITY CAR	Unit	5	166.666.667	833.333.333

Subtotal 833.333.333

PPN 11% 91.666.667

Discount -

Total (Curr.) Rp 925.000.000

NOTE

Down Payment:

Payment term:

Delivery Location:

SUPPLIER/VENDOR

Batas Disamping

PURCHASING

AUTHORIZED SIGNATURE

Batas Disamping

Budi Santoso

APPROVED

Batas Disamping

FINANCE

Nota: Dokumen ini adalah dokumen Purchase Order resmi berstandar PT Marlip Indo Mandiri yang diterbitkan secara elektronik.

[Unduh PDF](#)
✗ Tolak Kustomisasi
➤ Setujui & Kirim Draft PO

Marlip

PT MARLIP 2020 PNEC20

Email: sales@marlip.tech

Telp: 0811-2391777

PRODUK KAMI

Kendaraan Listrik (4 Roda)

EV Bike

Alat Pembelajaran EV

Suku Cadang

PERUSAHAAN

Tentang Kami

Galeri Produk

Berita & Event

Karir

Hubungi Kami

INFORMASI KONTAK

Pla Jl. Moch. Toha No.225,
Pasawahan, Kec. Dayeuhkolot,
Kabupaten Bandung,
Jawa Barat 40256

Tel +62 811 2391777

Email contact@marlip.tech

© 2020 PT Marlip Indo Mandiri. Hak cipta dilindungi.

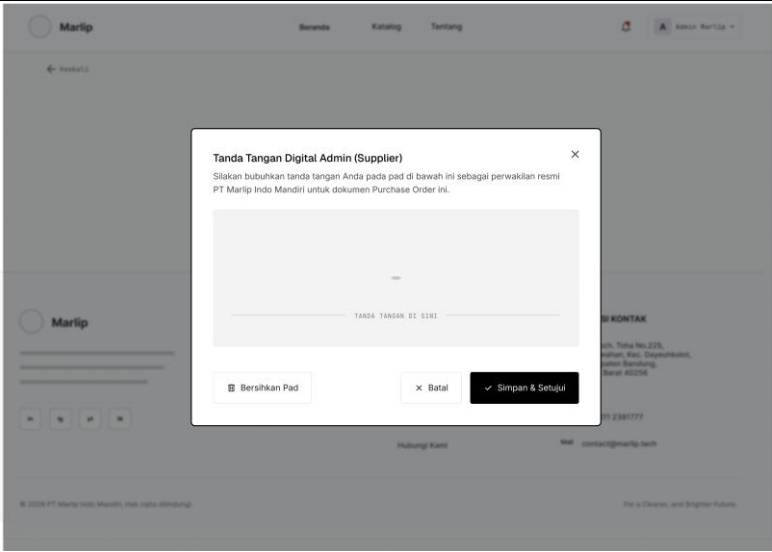
For a Cleaner, and Brighter Future.

Gambar 4.32 Halaman Pengelolaan Pesanan Admin

Deskripsi

Halaman ini menampilkan daftar pengajuan pesanan Konsumen. Admin dapat melihat rincian pesanan Standar atau Kustom, menolak pengajuan yang tidak dapat dipenuhi, menentukan harga final dan estimasi pengerjaan untuk varian Kustom, serta menandatangani dan menerbitkan draf PO.

Tabel 4.33 Halaman Tanda Tangan Elektronik Admin

Halaman Tanda Tangan Elektronik Admin 
Deskripsi Halaman atau modal ini digunakan Admin untuk memberikan tanda tangan elektronik sebelum menerbitkan draf Purchase Order. Tanda tangan Admin menjadi bagian dari dokumen PO yang selanjutnya ditinjau dan disetujui oleh Konsumen.

Tabel 4.34 Halaman Dashboard Kepala Produksi

Marlip

Tan Produkki

MENU UTAMA

- Antrean Produkki
- Profil Saya

Halaman Dashboard Kepala Produksi

Pesanan Dalam Proses

Pemburuan tahapan produksi kendaraan aktif.

Cari no. PO, konsumen, produk... 14 Aktif

PO-2028-0022

TENGKAT WAKTU
Sep 2028

Electric Blind Van (Marlip Electric Cargo - Blind Van)

Budi Santoso (Dinas Lingkungan Hidup Bandung)

Checklist Tahapan Produkki

1/4 Tahapan Selesai

1 Persiapan Produkki Selesai

27 Jul 2028

Dikirim 20.08

By Kassa Produkki

27 Jul 2028

Selesai 20.08

By Kepala Produkki

"BANKIR"

FOTO DOCUMENTASI

2 Perakitan Kendaraan Selesai

Perakitan rangka, motor listrik, baterai, dan sistem kelistrikan.

3 Pemeriksaan Akhir Selesai

Memeriksa kualitas ISO, uji fungsi elektronik, dan pengujian jalan.

4 Produkki Selesai Selesai

Seluruh tahapan produksi telah selesai dan unit siap dipasarkan/marketing.

PO-2028-0022

TENGKAT WAKTU
31 Agu 2028

Electric Blind Van (Marlip Electric Cargo - Blind Van)

Daffas (pt jaya abadi)

Checklist Tahapan Produkki

1/4 Tahapan Selesai

1 Persiapan Produkki Selesai

2 Perakitan Kendaraan Selesai

3 Pemeriksaan Akhir Selesai

4 Produkki Selesai Selesai

Merupakan 1-6 dari 14 data

Riwayat Produkki Selesai

Daftar kendaraan yang seluruh tahapan produksinya telah selesai.

PO-2028-0025

Selesai: 1 Juli 2028

Electric Bus 14 Seats - Standard (Marlip E-BUS 5.14 Standard)

Budi Santoso (Dinas Lingkungan Hidup Bandung)

STATUS PO: Siap Dikirim

PO-2028-0018

Selesai: 23 Jun 2028

Marlip Angkutan (Marlip MBS-S141 (14 + 1 Seats))

Budi Santoso (Dinas Lingkungan Hidup Bandung)

STATUS PO: Siap Dikirim

Kembali ke Website

Gambar 4.34 Halaman Dashboard Kepala Produksi

Deskripsi

Halaman ini menampilkan daftar pesanan yang telah memasuki proses produksi. Kepala Produksi dapat melihat informasi pesanan, status pembayaran, status tahapan produksi, serta mengakses halaman pembaruan tahapan produksi.

Tabel 4.35 Halaman Pembaruan Tahapan Produksi

Marlip

Tin Produksi

MENU UTAMA

Antrean Produksi

Profil Saya

Produksi

Pesanan Dalam Proses

Pembaruan tahapan produksi kendaraan aktif.

Carli no. PO, konsumen, produk...

14 AANF

PO-2026-0023

Electric Blind Van (Marlip Electric Cargo - Blind Van)

Budi Santoso (Dinas Lingkungan Hidup Bandung)

5 Sep 2026

Checklist Tahapan Produksi

1/4 Tahapan Selesai

1

Persiapan Produksi

Selesai

27 Jul 2026, 20:08

By Kepala Produksi

27 Jul 2026, 20:08

By Kepala Produksi

X

"Sesuai"

FOTO DOKUMENTASI

2

Perakitan Kendaraan

Selesai

Dimulai

MULAI TAHAPAN

Perakitan rangka, motor listrik, baterai, dan sistem kendalikannya.

3

Pemeriksaan Akhir

Selesai

Dimulai

MULAI TAHAPAN

Pemeriksaan kualitas RCT, uji fungsi kendalikannya, dan pengisian ulang.

4

Produksi Selesai

Selesai

Dimulai

MULAI TAHAPAN

Seluruh tahapan produksi telah selesai dan siap untuk diserahkan.

Checklist Tahapan Produksi

1/4 Tahapan Selesai

1

Persiapan Produksi

Selesai

2

Perakitan Kendaraan

Selesai

Dimulai

MULAI TAHAPAN

3

Pemeriksaan Akhir

Selesai

Dimulai

MULAI TAHAPAN

4

Produksi Selesai

Selesai

Dimulai

MULAI TAHAPAN

Merampikan 1-6 dan 14 data

1

2

3

Kembali ke Website

Gambar 4.35 Halaman Pembaruan Tahapan Produksi

Deskripsi

Halaman ini digunakan Kepala Produksi untuk memulai dan menyelesaikan tahapan produksi secara berurutan. Kepala Produksi dapat menambahkan catatan dan foto dokumentasi pada setiap tahapan. Tahapan yang telah selesai tidak dapat diubah kembali, sedangkan *invoice* pelunasan diterbitkan setelah seluruh tahapan produksi selesai.

BAB 5

IMPLEMENTASI DAN PENGUJIAN

Bab ini menjelaskan tahapan implementasi dari hasil perancangan yang telah diuraikan pada Bab 4 ke dalam bentuk aplikasi berbasis web. Implementasi dilakukan dengan menerjemahkan rancangan proses bisnis, basis data, antarmuka, dan kebutuhan fungsional ke dalam kode program. Selain itu, bab ini juga membahas pengujian perangkat lunak menggunakan metode Black-Box Testing untuk memastikan bahwa seluruh fungsionalitas sistem berjalan sesuai dengan spesifikasi kebutuhan pengguna. Pada bagian akhir, bab ini menguraikan tahapan deployment aplikasi agar sistem dapat diakses oleh pihak PT Marlip Indo Mandiri dan pengguna terkait.

5.1 Implementasi Sistem

Tahap implementasi sistem merupakan proses penerapan rancangan pemodelan sistem, arsitektur basis data, dan desain antarmuka menjadi aplikasi e-commerce pemesanan kendaraan listrik yang fungsional. Aplikasi ini dibangun menggunakan framework Next.js dengan pendekatan App Router, Prisma ORM sebagai penghubung basis data PostgreSQL, serta integrasi Midtrans Snap untuk mendukung pembayaran dua termin pada lingkungan pengujian.

5.1.1 Lingkungan Implementasi

Dalam proses pengembangan aplikasi pemesanan kendaraan listrik PT Marlip Indo Mandiri, penulis menggunakan perangkat keras dan perangkat lunak sebagai lingkungan implementasi sistem. Perangkat keras digunakan untuk proses pengkodean, pengujian lokal, dan dokumentasi. Sementara itu, perangkat lunak digunakan untuk mendukung proses perancangan antarmuka, pembuatan diagram, pengembangan aplikasi, pengelolaan basis data, integrasi pembayaran, serta deployment sistem.

Tabel 5.1 Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Laptop	Dell Latitude 7300
2	Prosesor	Intel(R) Core(TM) i7-8665U CPU @ 1.90GHz, 8 CPUs, ~2.1GHz
3	RAM	16 GB
4	Penyimpanan	SSD NVMe 239 GB

Tabel 5.2 Kebutuhan Perangkat Lunak

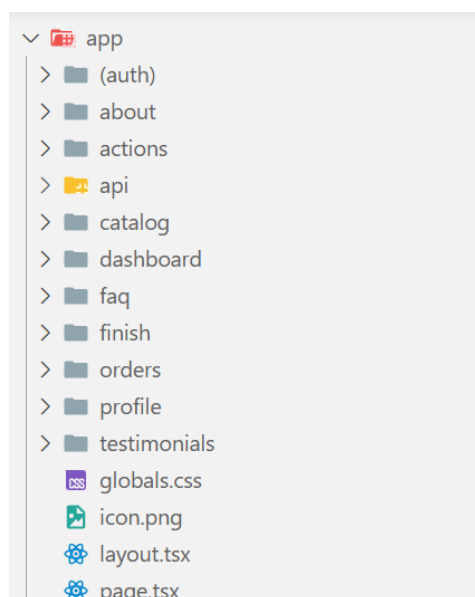
No	Perangkat Lunak	Keterangan
1	Windows 11 Pro 64-bit	Sistem operasi yang digunakan sebagai lingkungan utama dalam proses pengembangan aplikasi.
2	Visual Studio Code	Code editor yang digunakan untuk menulis, mengelola, dan melakukan penyuntingan source code aplikasi.
3	Node.js	Runtime environment JavaScript yang digunakan untuk menjalankan aplikasi berbasis Next.js pada lingkungan pengembangan.
4	TypeScript	Bahasa pemrograman yang digunakan untuk mengembangkan aplikasi dengan dukungan tipe data statis sehingga struktur kode lebih terjaga.
5	Next.js App Router	Framework utama berbasis React yang digunakan untuk membangun aplikasi web, mengelola routing, serta menangani proses server-side melalui Route Handler.
6	React	Library JavaScript yang digunakan untuk membangun komponen antarmuka pengguna pada aplikasi.
7	Tailwind CSS	Framework CSS yang digunakan untuk membangun tampilan antarmuka aplikasi secara responsif dan konsisten.

No	Perangkat Lunak	Keterangan
8	Prisma ORM	Object-Relational Mapping yang digunakan sebagai penghubung antara aplikasi dengan basis data PostgreSQL.
9	PostgreSQL	Sistem manajemen basis data relasional yang digunakan untuk menyimpan data pengguna, produk, pesanan, detail pesanan, dan invoice.
10	Neon	Platform serverless PostgreSQL yang digunakan sebagai layanan basis data cloud untuk aplikasi.
11	NextAuth	Library autentikasi yang digunakan untuk mengelola sesi login, otorisasi, dan pembagian hak akses pengguna berdasarkan role.
12	Midtrans Snap API Sandbox	Layanan payment gateway yang digunakan untuk simulasi pembayaran dua termin, yaitu Down Payment (DP) dan pelunasan.
13	Vercel	Platform deployment yang digunakan untuk menerapkan aplikasi web agar dapat diakses secara daring.
14	Git dan GitHub	Sistem version control dan repository yang digunakan untuk menyimpan, mengelola, serta melacak perubahan source code aplikasi.
15	Figma	Perangkat lunak desain yang digunakan untuk membuat rancangan antarmuka pengguna sebelum diimplementasikan ke dalam aplikasi.
16	Draw.io / diagrams.net	Perangkat lunak yang digunakan untuk membuat diagram perancangan sistem, seperti flowmap, use case diagram, activity diagram, sequence diagram, dan ERD.
17	Google Chrome	Web browser yang digunakan untuk menjalankan, meninjau, dan menguji aplikasi selama proses pengembangan.

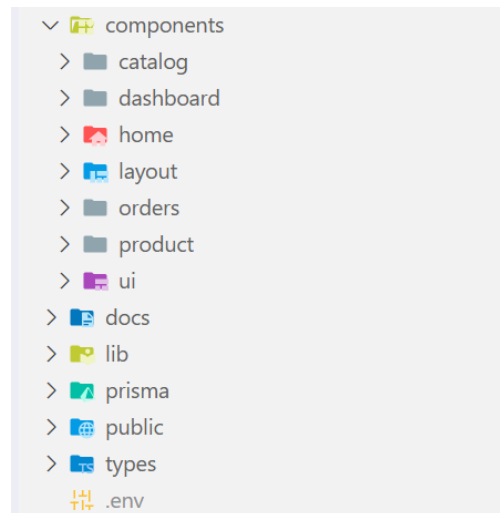
Perangkat keras dan perangkat lunak tersebut digunakan untuk mendukung proses implementasi sistem secara menyeluruh, mulai dari pengkodean, pengujian lokal, pengelolaan basis data, hingga penerapan aplikasi ke lingkungan deployment.

5.1.2 Struktur Folder dan Berkas

Struktur folder dan berkas aplikasi disusun mengikuti pola Next.js App Router. Pada struktur ini, direktori app digunakan untuk mengelola halaman, layout, dan Route Handler. Direktori components digunakan untuk menyimpan komponen antarmuka yang digunakan secara berulang. Direktori lib digunakan untuk menyimpan fungsi pendukung seperti konfigurasi Prisma dan autentikasi. Sementara itu, direktori prisma digunakan untuk menyimpan skema basis data dan konfigurasi model data.



Gambar 5.1 Struktur Folder App Router



Gambar 5.2 Struktur Folder Komponen dan Konfigurasi Aplikasi

Tabel 5.3 Struktur Folder Aplikasi

No	Folder/Berkas	Keterangan
1	app	Direktori utama Next.js App Router yang berisi halaman aplikasi, layout, route handler API, serta pengaturan routing berbasis folder.
2	app/api	Direktori yang berisi Route Handler untuk menangani proses backend seperti pesanan, produk, pembayaran, upload file, dan webhook Midtrans.
3	app/catalog	Direktori halaman katalog dan detail produk yang dapat diakses oleh pengunjung maupun pengguna terdaftar.
4	app/dashboard	Direktori halaman dashboard berdasarkan hak akses pengguna, yaitu Konsumen, Admin, dan Kepala Produksi.
5	app/orders	Direktori halaman yang berkaitan dengan detail pesanan, dokumen PO, pembayaran, dan pelacakan tahapan produksi.
6	app/(auth)	Direktori halaman autentikasi, seperti login dan registrasi pengguna.
7	components	Direktori yang berisi komponen antarmuka pengguna yang digunakan secara berulang pada berbagai halaman aplikasi.
8	components/dashboard	Direktori komponen khusus untuk dashboard Admin dan Kepala Produksi, seperti manajemen produk, tabel pesanan, dan checklist tahapan produksi.
9	components/orders	Direktori komponen yang berkaitan dengan dokumen transaksi, seperti Digital PO, Invoice, dan pelacakan tahapan produksi.
10	components/product	Direktori komponen yang berkaitan dengan detail produk dan formulir pemesanan.
11	lib	Direktori yang berisi fungsi pendukung aplikasi, seperti konfigurasi autentikasi, koneksi Prisma, konfigurasi tahapan produksi, dan utilitas lainnya.
12	prisma	Direktori yang berisi skema basis data Prisma, konfigurasi model data, dan berkas migrasi basis data.
13	public	Direktori yang digunakan untuk menyimpan aset statis aplikasi, seperti gambar, ikon, dan file pendukung lainnya.
14	types	Direktori yang berisi definisi tipe data TypeScript yang digunakan pada aplikasi.
15	proxy.ts	Berkas pengatur perlindungan rute berdasarkan status login dan role pengguna.
16	.env	Berkas konfigurasi environment variable seperti koneksi database, konfigurasi NextAuth, dan kredensial Midtrans.

Struktur folder tersebut membantu pemisahan tanggung jawab antarbagian aplikasi. Halaman dan Route Handler dikelola melalui direktori app, komponen tampilan dikelola melalui components, konfigurasi basis data dikelola melalui prisma, sedangkan fungsi pendukung aplikasi dikelola melalui lib.

5.1.3 Konfigurasi Aplikasi

Konfigurasi aplikasi dilakukan untuk menghubungkan sistem dengan layanan yang digunakan, seperti basis data, autentikasi, payment gateway, dan deployment. Konfigurasi utama disimpan pada berkas `.env` dalam bentuk environment variable. Nilai asli dari environment variable tidak ditampilkan dalam laporan karena berisi data sensitif, seperti kredensial basis data, secret autentikasi, dan kunci Midtrans.

Tabel 5.4 Konfigurasi Environment Variable

No	Konfigurasi	Keterangan
1	DATABASE_URL	Digunakan sebagai URL koneksi utama aplikasi ke basis data PostgreSQL pada Neon.
2	DIRECT_URL	Digunakan sebagai URL koneksi langsung ke basis data untuk kebutuhan Prisma dan migrasi basis data.
3	AUTH_SECRET	Digunakan sebagai secret key untuk proses autentikasi dan pengamanan sesi pengguna.
4	NEXTAUTH_URL	Digunakan sebagai URL dasar aplikasi pada tautan dan kebutuhan autentikasi.
5	MIDTRANS_SERVER_KEY	Digunakan sebagai kunci server untuk membuat transaksi Snap dan memvalidasi webhook Midtrans.
6	MIDTRANS_IS_PRODUCTION	Digunakan untuk menentukan mode Midtrans pada sisi server, yaitu sandbox atau production.
7	NEXT_PUBLIC_MIDTRANS_CLIENT_KEY	Digunakan untuk menyediakan client key Midtrans yang dapat diakses komponen frontend saat menampilkan Snap.
8	NEXT_PUBLIC_MIDTRANS_IS_PRODUCTION	Digunakan untuk menentukan mode Midtrans pada komponen frontend saat memuat Snap.

Konfigurasi aplikasi tidak hanya dilakukan melalui berkas `.env`, tetapi juga melalui berkas pendukung seperti `schema.prisma`, konfigurasi autentikasi, dan konfigurasi payment gateway. Konfigurasi tersebut digunakan untuk menghubungkan aplikasi dengan basis data, mengatur sesi pengguna, menjalankan integrasi Midtrans, serta menyiapkan aplikasi pada lingkungan deployment.

5.1.4 Implementasi Basis Data

Implementasi basis data pada aplikasi pemesanan kendaraan listrik PT Marlip Indo Mandiri dilakukan menggunakan PostgreSQL yang dikelola melalui layanan Neon. Akses basis data dari aplikasi dilakukan menggunakan Prisma ORM. Prisma digunakan untuk mendefinisikan struktur data, relasi antarentitas, serta tipe data yang digunakan pada sistem.

Skema basis data didefinisikan pada berkas `schema.prisma`. Berkas tersebut memuat enam entitas utama, yaitu User, Produk, PurchaseOrder, DetailPesanan, Invoice, dan TahapanProduksi. Entitas-entitas tersebut merupakan hasil implementasi dari rancangan Entity Relationship Diagram (ERD) yang telah dijelaskan pada Bab 4.

Tabel 5.5 Implementasi Entitas Basis Data

No	Entitas	Fungsi
1	User	Menyimpan data pengguna sistem, seperti Konsumen, Admin, dan Kepala Produksi.
2	Produk	Menyimpan data katalog produk, kategori, spesifikasi, harga dasar, foto produk, dan status aktif produk.
3	PurchaseOrder	Menyimpan data transaksi pemesanan, status PO, harga, estimasi pengerjaan, tanda tangan elektronik, dan relasi tahapan produksi
4	DetailPesanan	Menyimpan rincian produk yang dipesan, jumlah pesanan, kustomisasi, harga satuan, dan subtotal.
5	Invoice	Menyimpan data tagihan pembayaran DP dan pelunasan, status pembayaran, Snap Token, dan Midtrans Order ID.
6	TahapanProduksi	Menyimpan data tahapan produksi pada setiap pesanan, meliputi nama dan urutan tahapan, status, waktu serta pengguna yang memulai atau menyelesaikan tahapan, catatan, dan foto dokumentasi.

Entitas User digunakan untuk menyimpan data pengguna sistem beserta role yang dimiliki. Entitas Produk digunakan untuk menyimpan data katalog produk yang ditampilkan pada halaman katalog. Entitas PurchaseOrder digunakan sebagai tabel utama transaksi pemesanan. Entitas DetailPesanan digunakan untuk menyimpan rincian produk yang dipesan, sedangkan entitas Invoice digunakan untuk menyimpan tagihan pembayaran berdasarkan termin DP dan pelunasan. Entitas TahapanProduksi digunakan untuk menyimpan status dan riwayat pembaruan setiap tahapan produksi pada Purchase Order. Setiap pesanan memiliki empat tahapan, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai.

Pada implementasinya, relasi antarentitas dibentuk melalui *foreign key*. Satu pengguna dapat memiliki banyak *PurchaseOrder*. Satu *PurchaseOrder* dapat memiliki satu atau lebih *DetailPesanan*. Satu Produk dapat digunakan pada banyak *DetailPesanan*. Selain itu, satu *PurchaseOrder* dapat memiliki maksimal dua *Invoice*, yaitu *Invoice* DP dan *Invoice* pelunasan. Satu *PurchaseOrder* dapat memiliki empat *TahapanProduksi*. Entitas User juga berelasi dengan *TahapanProduksi* melalui atribut *startedBy* dan *completedBy* untuk mencatat pengguna yang memulai dan menyelesaikan setiap tahapan.

Untuk menjaga ketepatan nilai transaksi, field nominal uang seperti *hargaDasar*, *totalHarga*, *hargaFinal*, *hargaSatuan*, *subtotal*, dan *totalTagihan* menggunakan tipe data *Decimal(18,0)*. Penggunaan tipe data tersebut bertujuan untuk menghindari kesalahan pembulatan pada nilai mata uang, khususnya pada proses perhitungan PPN, DP, dan pelunasan.

Potongan kode implementasi model User dan Produk pada Prisma ORM dapat dilihat pada Lampiran D, sedangkan model transaksi yang terdiri dari *PurchaseOrder*, *DetailPesanan*, *Invoice*, dan *TahapanProduksi* dapat dilihat pada Lampiran E.

5.1.5 Implementasi Route Handler dan API

Implementasi Route Handler digunakan sebagai penghubung antara antarmuka pengguna dan proses pengolahan data pada sisi server. Pada aplikasi ini, Route Handler ditempatkan pada direktori *app/api* dengan berkas *route.ts* sesuai pendekatan Next.js App Router. Route Handler berfungsi untuk menerima permintaan HTTP, memvalidasi data masukan, memeriksa autentikasi dan otorisasi pengguna, menjalankan operasi basis data menggunakan Prisma ORM, serta menghubungkan sistem dengan layanan eksternal seperti Midtrans.

Route Handler pada sistem ini digunakan untuk beberapa kebutuhan utama, yaitu pengelolaan produk, pengajuan pesanan, peninjauan Purchase Order, pembayaran, validasi status transaksi, webhook Midtrans, dan unggah berkas. Setiap endpoint memiliki fungsi yang berbeda sesuai dengan kebutuhan sistem dan hak akses pengguna.

Tabel 5.6 Implementasi Route Handler

No	Endpoint	Metode	Fungsi
1	/api/products	GET, POST	Mengambil data katalog produk aktif yang dapat diakses publik dan menambahkan produk baru oleh Admin.
2	/api/products/[id]	GET, PATCH, DELETE	Mengambil detail produk, memperbarui data produk, serta menonaktifkan produk dari katalog.
3	/api/orders	GET, POST	Mengambil daftar pesanan sesuai <i>role</i> pengguna dan menyimpan pengajuan pesanan baru dari Konsumen.
4	/api/orders/[id]	GET, PATCH	Mengambil detail <i>Purchase Order</i> serta memperbarui data pesanan dan status PO sesuai hak akses pengguna.
5	/api/orders/[id]/tahapan	GET, PATCH	Mengambil data tahapan produksi serta memulai atau menyelesaikan tahapan produksi secara berurutan oleh Kepala Produksi.
6	/api/orders/[id]/pay	POST	Memproses pembayaran pesanan pada termin tertentu serta memperbarui status pembayaran dan status pesanan sesuai hasil validasi.
7	/api/payments/create-token	POST	Membuat atau mengambil <i>Snap Token</i> dari layanan Midtrans berdasarkan data <i>Invoice</i> .
8	/api/payments/status/[invoiceId]	GET	Melakukan pemeriksaan dan sinkronisasi status pembayaran antara sistem aplikasi dan layanan transaksi Midtrans.
9	/api/payments/webhook	POST	Menerima <i>webhook</i> dari Midtrans dan memvalidasi <i>signature key</i> , <i>order ID</i> , serta <i>gross amount</i> sebelum memperbarui data pembayaran.
10	/api/upload	POST	Menangani unggah foto produk oleh Admin dan dokumentasi produksi oleh Kepala Produksi.
11	/api/auth/[...nextauth]	GET, POST	Menangani autentikasi dan manajemen sesi pengguna menggunakan NextAuth.

Pada endpoint produk, sistem menampilkan produk yang masih aktif pada katalog dan membatasi proses penambahan, pembaruan, serta penonaktifan produk hanya untuk pengguna dengan role Admin. Validasi dilakukan pada sisi server, seperti validasi kategori produk dan harga dasar agar data yang tersimpan tetap sesuai dengan kebutuhan sistem.

Pada endpoint pesanan, sistem membatasi pengajuan pesanan hanya untuk pengguna dengan role Konsumen. Data pesanan divalidasi sebelum disimpan, meliputi produk aktif, warna utama, varian pesanan, jumlah pesanan, dan catatan kebutuhan khusus untuk varian Kustom. Endpoint pesanan juga digunakan untuk mengatur transisi status PO dan memvalidasi tanda tangan elektronik. Sementara itu, pembaruan tahapan produksi dilakukan oleh Kepala Produksi melalui endpoint /api/orders/[id]/tahapan.

Pada endpoint pembayaran, sistem melakukan integrasi dengan Midtrans Snap untuk menghasilkan Snap Token pembayaran. Pembayaran dibagi menjadi dua termin, yaitu Down Payment (DP) dan pelunasan. Pembayaran DP hanya dapat dilakukan setelah Konsumen menyetujui draf PO, sedangkan pembayaran pelunasan hanya dapat dilakukan setelah DP berstatus paid dan seluruh tahapan produksi berstatus Selesai.

Webhook Midtrans digunakan untuk menerima notifikasi status transaksi dari Midtrans. Sebelum memperbarui data Invoice dan Purchase Order, sistem memvalidasi signature key, order ID, dan gross amount. Validasi tersebut bertujuan untuk memastikan bahwa notifikasi pembayaran yang diterima benar-

benar berasal dari transaksi yang sesuai. Setelah valid, sistem memperbarui status Invoice dan status Purchase Order sesuai termin pembayaran.

Potongan kode implementasi Route Handler untuk validasi pengajuan pesanan, pembuatan Purchase Order, validasi webhook Midtrans, serta pembaruan status Invoice dan Purchase Order dapat dilihat pada Lampiran F sampai Lampiran I.

5.2 Implementasi Antarmuka Berdasarkan Role

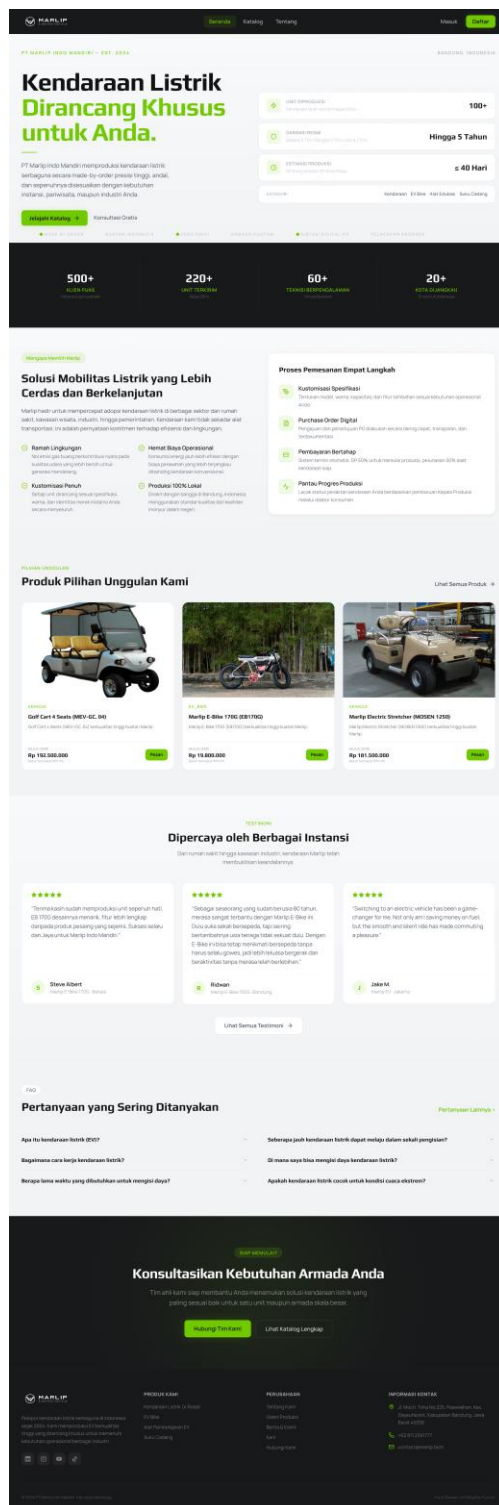
Implementasi antarmuka pengguna dilakukan berdasarkan rancangan antarmuka yang telah dijelaskan pada Bab 4. Antarmuka aplikasi dibangun menggunakan React dan Tailwind CSS agar tampilan dapat menyesuaikan ukuran layar serta tetap konsisten pada setiap halaman. Dalam sistem ini, antarmuka dibedakan berdasarkan hak akses pengguna, yaitu Pengunjung, Konsumen, Admin, dan Kepala Produksi.

Pembagian antarmuka berdasarkan role bertujuan agar setiap pengguna hanya dapat mengakses fitur yang sesuai dengan kewenangannya. Pengunjung dapat mengakses katalog dan detail produk tanpa melakukan login. Konsumen dapat mengajukan pesanan, meninjau draf PO, melakukan pembayaran, dan memantau tahapan produksi. Admin dapat mengelola produk dan meninjau pesanan, sedangkan Kepala Produksi dapat memperbarui tahapan produksi serta mengunggah dokumentasi produksi.

5.2.1 Antarmuka Pengunjung dan Konsumen

Antarmuka Pengunjung dan Konsumen dirancang untuk memudahkan pengguna dalam melihat informasi produk, mengajukan pesanan, menyetujui draf Purchase Order, melakukan pembayaran, serta memantau tahapan produksi. Pengunjung dapat mengakses halaman beranda, katalog, dan detail produk tanpa login. Namun, ketika akan mengajukan pesanan, pengguna diarahkan untuk login sebagai Konsumen terlebih dahulu.

Setelah login, Konsumen dapat menggunakan Formulir Pemesanan pada halaman detail produk. Formulir tersebut menyediakan pilihan warna utama, varian Standar atau Kustom, jumlah pesanan, serta catatan kebutuhan khusus apabila memilih varian Kustom. Setelah pesanan diajukan, Konsumen dapat meninjau draf PO, memberikan tanda tangan elektronik, melakukan pembayaran DP dan pelunasan, serta melihat tahapan produksi melalui dashboard.



Gambar 5.3 Antarmuka Halaman Beranda


[Beranda](#)
[Katalog](#)
[Tentang](#)
[Masuk](#)
[Daftar](#)

[Semua Produk](#)
[Kendaraan](#)
[EV Bike](#)
[Alat Pembastaraan](#)
[Suku Cadang](#)

Katalog Kendaraan Listrik

Keunggulan berbagai pilihan kendaraan listrik yang dapat disesuaikan sesuai kebutuhan operasional industri Anda.



KENDARAAN EV
Electric Blind Van (Marlip Electric Cargo - Blind Van)
Electric Blind Van (Marlip Electric Cargo - Blind Van) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 291.500.000

Penawaran



KENDARAAN EV
Electric Bus 14 Seats - Luxury Shuttle (Marlip E-BUS 5.14 - New)
Electric Bus 14 Seats - Luxury Shuttle (Marlip E-BUS 5.14 - New) berkapasitas hingga 14 orang penumpang.

SEKELAH DARI
Rp 323.000.000

Penawaran



KENDARAAN EV
Electric Bus 14 Seats - Standard (Marlip E-BUS 5.14 Standard)
Electric Bus 14 Seats - Standard (Marlip E-BUS 5.14 Standard) berkapasitas hingga 14 orang penumpang.

SEKELAH DARI
Rp 303.000.000

Penawaran



KENDARAAN EV
Electric Bus 15 Seats Full Cover (Marlip E-BUS 5.15 Full Cover)
Electric Bus 15 Seats Full Cover (Marlip E-BUS 5.15 Full Cover) berkapasitas hingga 15 orang penumpang.

SEKELAH DARI
Rp 370.000.000

Penawaran



KENDARAAN EV
Electric Bus 18 Seats (Marlip E-BUS 5.18)
Electric Bus 18 Seats (Marlip E-BUS 5.18) berkapasitas hingga 18 orang penumpang.

SEKELAH DARI
Rp 363.000.000

Penawaran



KENDARAAN EV
Electric Bus 23 Seats (Marlip E-BUS 5.23)
Electric Bus 23 Seats (Marlip E-BUS 5.23) berkapasitas hingga 23 orang penumpang.

SEKELAH DARI
Rp 494.000.000

Penawaran



KENDARAAN EV
Electric Bus 23 Seats Full Cover (Marlip E-BUS 5.23 Full Cover)
Electric Bus 23 Seats Full Cover (Marlip E-BUS 5.23 Full Cover) berkapasitas hingga 23 orang penumpang.

SEKELAH DARI
Rp 527.000.000

Penawaran



KENDARAAN EV
Electric Cargo Box (Marlip Electric Cargo - Box)
Electric Cargo Box (Marlip Electric Cargo - Box) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 259.500.000

Penawaran



KENDARAAN EV
Electric Cargo Flat Deck (Marlip Electric Cargo - Flat Deck)
Electric Cargo Flat Deck (Marlip Electric Cargo - Flat Deck) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 247.500.000

Penawaran



KENDARAAN EV
Electric Cargo Pick Up (Marlip Electric Cargo - Pick Up)
Electric Cargo Pick Up (Marlip Electric Cargo - Pick Up) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 247.500.000

Penawaran



KENDARAAN EV
Electric Dump Truck (Marlip Electric Cargo - Dump Truck)
Electric Dump Truck (Marlip Electric Cargo - Dump Truck) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 291.500.000

Penawaran



KENDARAAN EV
Electric Truck - Box/Blind Van 2 Ton (E-Truck Box 2 Ton)
Electric Truck - Box/Blind Van 2 Ton (E-Truck Box 2 Ton) berkapasitas hingga 10 orang penumpang.

SEKELAH DARI
Rp 359.700.000

Penawaran

[1](#)
[2](#)
[3](#)
[4](#)



Produk Kami

- Kendaraan Listrik (EV Bus)
- EV Bike
- Alat Pembastaraan EV
- Suku Cadang

Perusahaan

- Teknikal Kami
- Galeri Produk
- Baris & Lantai
- Rak
- Hubungi Kami

Informasi Kontak

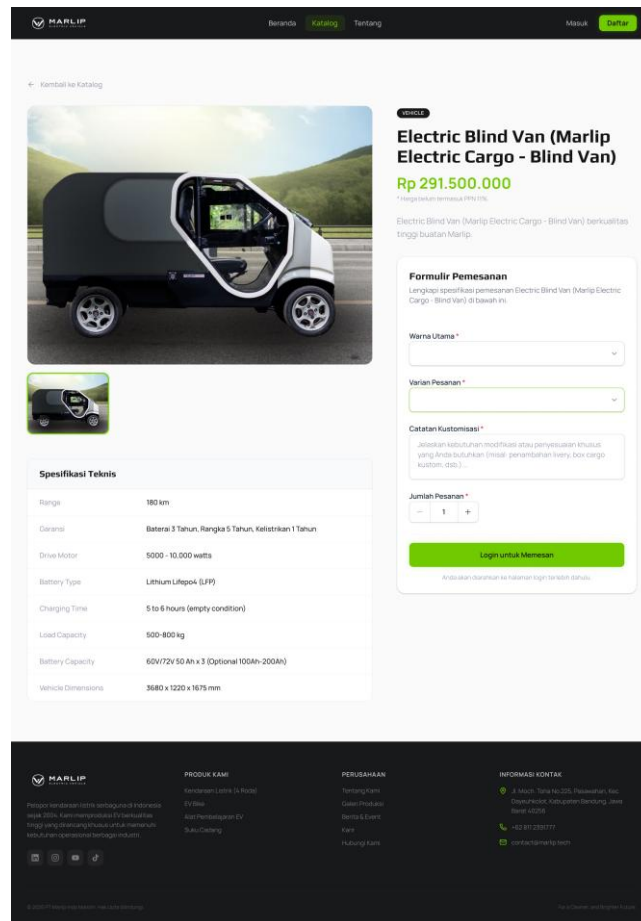

 Jl. Moch. Tohir, 100, Pekanbaru, Riau
 28111, Indonesia


 +62 811 250777

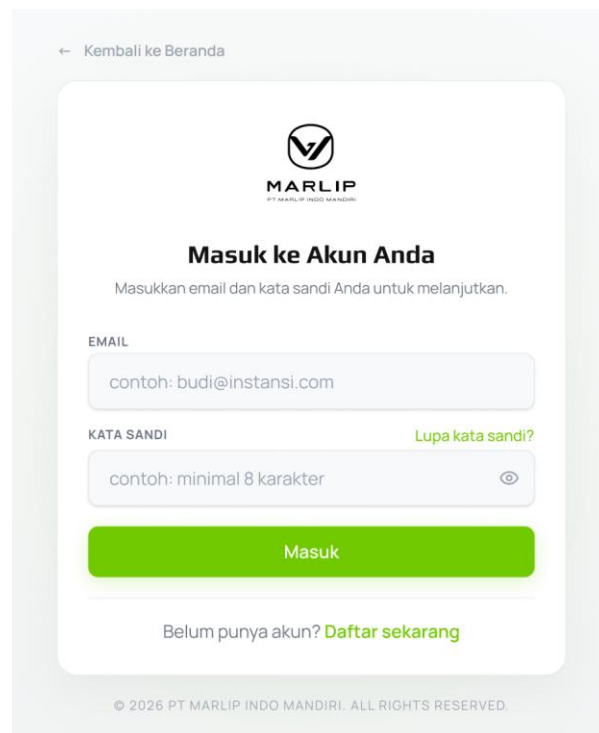

 info@marlip.co.id

© 2023 Marlip Indonesia. All rights reserved.

Gambar 5.4 Antarmuka Halaman Katalog Produk



Gambar 5.5 Antarmuka Halaman Detail Produk dan Formulir Pemesanan



Gambar 5.6 Antarmuka Halaman Login

← Kembali ke Beranda



Buat Akun Baru

Lengkapi data diri Anda untuk mulai memesan kendaraan listrik Marlip.

NAMA LENGKAP

contoh: Budi Santoso

INSTANSI / PERUSAHAAN

contoh: PT Maju Jaya atau Dinas Perhubungan

NOMOR WHATSAPP

contoh: 081234567890

EMAIL

contoh: budi@instansi.com

ALAMAT LENGKAP

contoh: Jl. Moch. Toha No.225, Pasawahan, Dayeuhkolot, Bandung

KATA SANDI

contoh: minimal 8 karakter

Daftar Sekarang

Dengan mendaftar, Anda menyetujui Syarat & Ketentuan serta Kebijakan Privasi Marlip.

Sudah punya akun? [Masuk di sini](#)

© 2026 PT MARLIP INDO MANDIRI. ALL RIGHTS RESERVED.

Gambar 5.7 Antarmuka Halaman Registrasi



Menu Utama

- Pesanan Saya**
- Profil Saya

DAFTAR PESANAN

Konsumen

Terdapat Pesanan: **15** | Dalam Proses: **9** | Selesai: **1**

Carilah PO atau nama produk...

Pesanan Saya

Electric Bus 18 Seats (Marlip E-BUS 5.18)

Dipesan: 26 Jun 2026 | Jumlah: 1 unit

Progres Perakitan: 80%

FOTO DOKUMENTASI PERAKITAN (1)

Lacak Progres Perakitan

Electric Bus 14 Seats - Luxury Shuttle (Marlip E-BUS 5.14 - New)

Dipesan: 23 Jun 2026 | Jumlah: 2 unit

Progres Perakitan: 75%

FOTO DOKUMENTASI PERAKITAN (3)

Lacak Progres Perakitan

Marlip Angklung (Marlip MBS-5141 (14 + 1 Seats))

Dipesan: 23 Jun 2026 | Jumlah: 2 unit

Progres Perakitan: 100%

FOTO DOKUMENTASI PERAKITAN (1)

Lihat Status Siap Dikirim

Electric Dump Truck (Marlip Electric Cargo - Dump Truck)

Dipesan: 22 Jun 2026 | Jumlah: 1 unit

Progres Perakitan: 100%

Bayar Pelunasan

Electric Blind Van (Marlip Electric Cargo - Blind Van)

Dipesan: 22 Jun 2026 | Jumlah: 1 unit

Progres Perakitan: 15%

Lacak Progres Perakitan

Electric Dump Truck (Marlip Electric Cargo - Dump Truck)

Dipesan: 22 Jun 2026 | Jumlah: 20 unit

Alasan Pembatalan: tidak ada

Lihat Detail Pembatalan

Menampilkan 1-6 dari 16 data

Gambar 5.8 Antarmuka Dashboard Konsumen

[Beranda](#)
[Katalog](#)
[Tentang Kami](#)

Budi Santoso

[Kembali](#)

Tinjau Draft Digital PO Anda

Admin telah memvalidasi informasi pesanan Anda. Silakan tinjau draft PO resmi di bawah. Klik 'Setujui Draft PO' untuk menerbitkan invoice DP 50% dan mulai produksi.

PT Marlipindo
 Jl. Moch. Toha No. 225, Klaten, Sukoharjo, Jawa Tengah, Indonesia
 57131 - 0291-767150

PT Marlipindo
 Jl. Moch. Toha No. 225, Klaten, Sukoharjo, Jawa Tengah, Indonesia
 57131 - 0291-767150

No	Detail	Unit	Qty	Price	Total Price
1	ELECTRIC BLIND VAN (MARLIP ELECTRIC CARO - BLIND VAN) Material: Alum Model: Standard Harga Satuan: Rp. 231.190.000 PPN 10%: Rp. 23.469.900 Harga Diskon: Rp. 23.469.900	1000	1	231.190.000	231.190.000
Subtotal					231.190.000
PPN 10%					23.469.900
Diskon					-
Total (Curr.)					Rp. 323.568.000

NOTE

Down Payment: **(Rp. 161.762.000 (50%))**
 Pemasangan: **(Rp. 161.762.000 (50%))**
 Pemasangan Term: **(Rp. 50% & Pemasangan 50%)**
 Sistem Pengiriman: **100% Klaten**
 Delivery Location: **C-1**

SURVEY/ON-SITE

Marlipindo

AUTHORIZED SIGNATURE

Budi Santoso

APPROVED

PT Marlipindo

Nota: Informasi di bawah ini adalah informasi yang akan digunakan PT Marlipindo untuk memproduksi dan mengirimkan barang.

[Unduh PDF](#)

[X Tolak Draft PO](#)

[Setujui Draft PO & Terbitkan Invoice](#)

PRODUK KAMI

Handmade Lotion (14.700)

EV (Blue)

Asal Pemasangan EV

Buku Coding

PESUSUKAN

Tentang Kami

Global Products

Baru & Event

Rare

Hubungi Kami

INFORMASI KONTAK

Jl. Moch. Toha No. 225, Sukoharjo, Jawa Tengah 57131

0291-767150

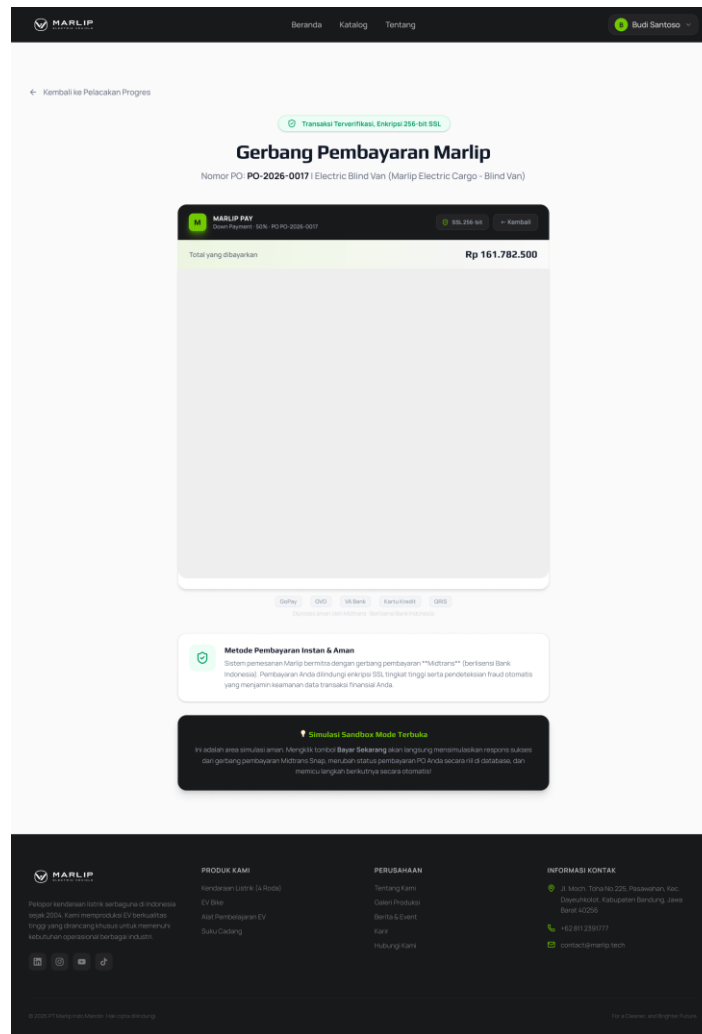
0291-767150

contact@marlipindo.com

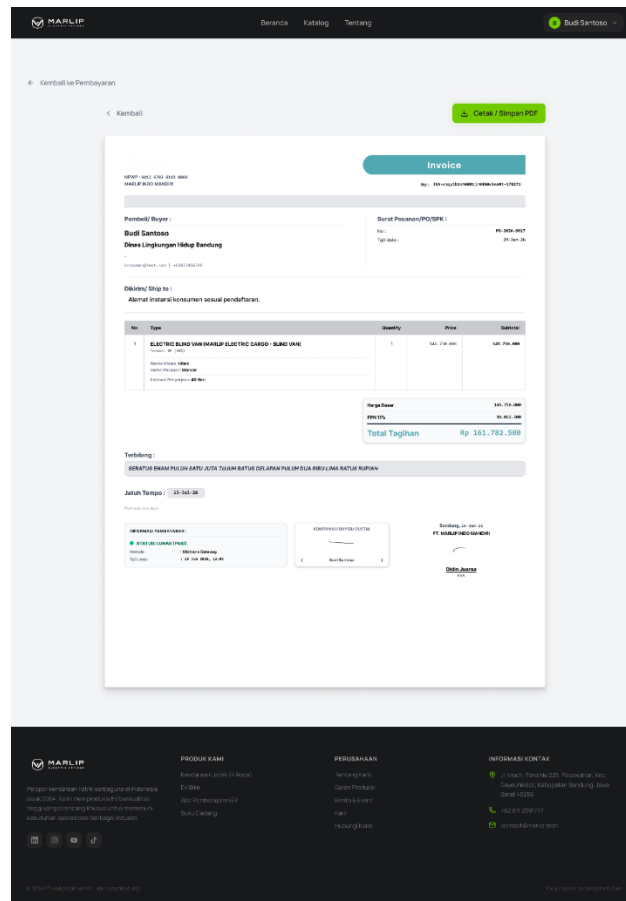
Produk kami telah terdistribusi ke seluruh Indonesia. Untuk informasi lebih lanjut, silakan hubungi kami melalui WhatsApp atau email kami.

© 2023 PT Marlipindo. All rights reserved.

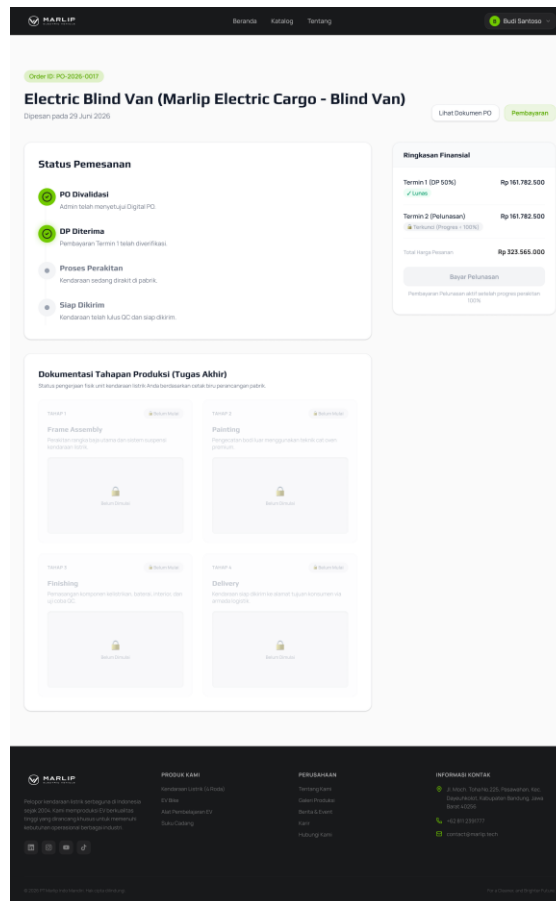
Gambar 5.9 Antarmuka Persetujuan Draf Purchase Order Konsumen



Gambar 5.10 Antarmuka Pembayaran Midtrans Snap



Gambar 5.11 Antarmuka Invoice Pembayaran

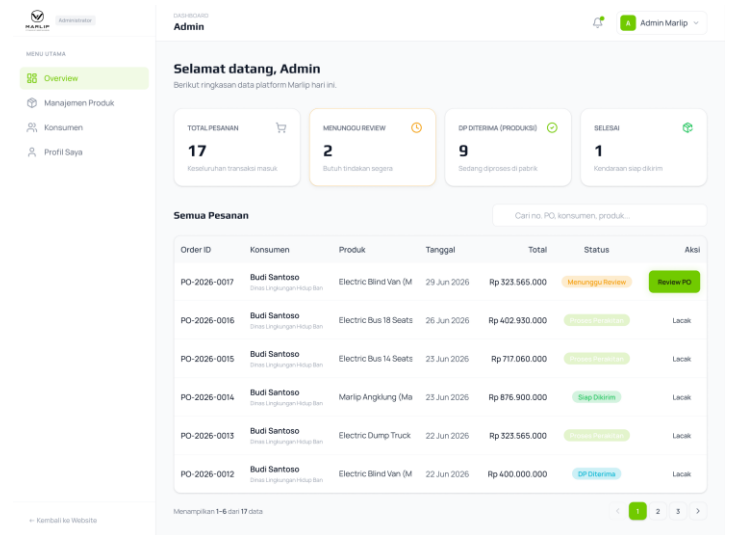


Gambar 5.12 Antarmuka Pelacakan Tahapan Produksi

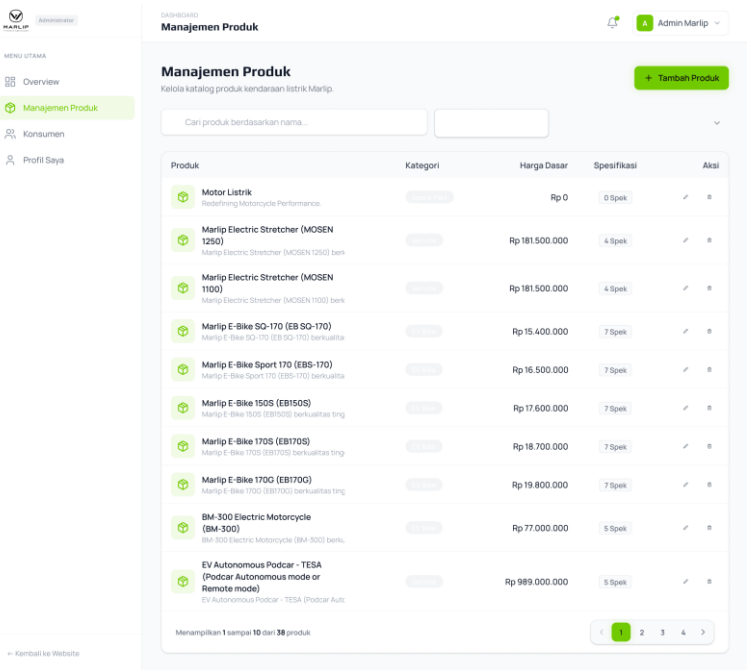
5.2.2 Antarmuka Admin

Antarmuka Admin digunakan untuk mengelola proses operasional sistem, khususnya pengelolaan katalog produk dan peninjauan pesanan yang diajukan oleh Konsumen. Melalui dashboard Admin, pengguna dengan role Admin dapat melihat daftar pesanan, memeriksa detail pesanan, menentukan harga final dan estimasi pengerjaan untuk pesanan Kustom, serta menerbitkan draf Purchase Order.

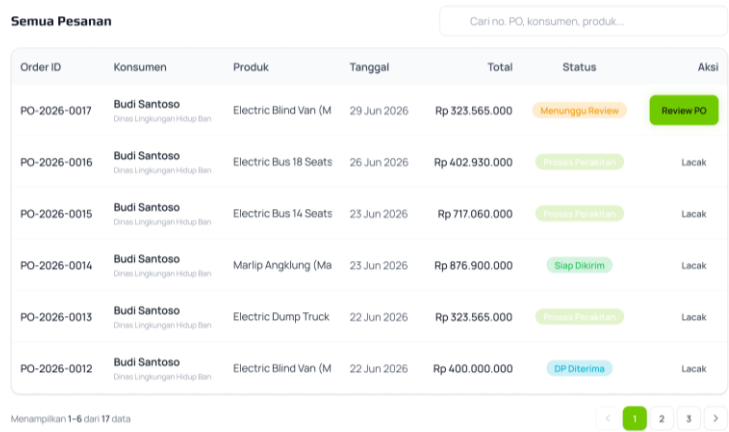
Pada pengelolaan produk, Admin dapat menambahkan produk baru, memperbarui data produk, mengunggah foto produk, dan menonaktifkan produk yang tidak ingin ditampilkan pada katalog. Selain itu, Admin juga memiliki fitur tanda tangan elektronik yang digunakan sebelum draf PO diterbitkan kepada Konsumen.



Gambar 5.13 Antarmuka Dashboard Admin



Gambar 5.14 Antarmuka Manajemen Produk



Gambar 5.15 Antarmuka Pengelolaan Pesanan Admin

[illegible]

Gambar 5.16 Antarmuka Peninjauan Pesanan dan Draf PO

Tanda Tangan Elektronik Admin (Supplier) ✕

Silakan bubuhkan tanda tangan Anda pada pad di bawah ini sebagai perwakilan resmi PT Marlip Indo Mandiri untuk dokumen Purchase Order ini.

TANDA TANGAN DI SINI

 Bersihkan Pad

✕ Batal

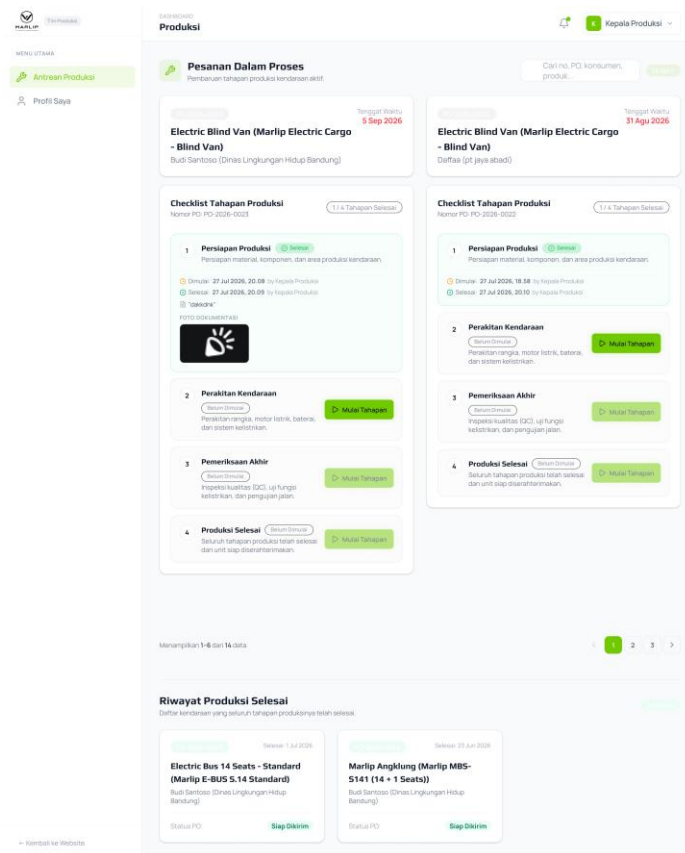
✓ Simpan & Setujui

Gambar 5.17 Antarmuka Tanda Tangan Elektronik Admin

5.2.3 Antarmuka Kepala Produksi

Antarmuka Kepala Produksi digunakan untuk menampilkan daftar pesanan yang telah memasuki tahap produksi serta memperbarui tahapan produksinya. Setelah pembayaran DP berhasil diverifikasi, sistem menyiapkan empat tahapan produksi, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai.

Melalui halaman ini, Kepala Produksi dapat memulai dan menyelesaikan setiap tahapan secara berurutan, serta menambahkan catatan dan foto dokumentasi produksi. Sistem mencatat waktu dan pengguna yang melakukan pembaruan. Tahapan yang telah selesai tidak dapat diubah kembali. Setelah seluruh tahapan produksi selesai, sistem menerbitkan *Invoice* pelunasan, sedangkan status pesanan tetap berada pada tahap Proses Perakitan sampai pembayaran pelunasan berhasil diverifikasi.



Gambar 5.18 Antarmuka Dashboard Kepala Produksi

PO-2026-0023

Tenggat Waktu
5 Sep 2026

Electric Blind Van (Marlip Electric Cargo
- Blind Van)

Budi Santoso (Dinas Lingkungan Hidup Bandung)

Checklist Tahapan Produksi

Nomor PO: PO-2026-0023

1 / 4 Tahapan Selesai

1

Persiapan Produksi Selesai

Persiapan material, komponen, dan area produksi kendaraan.

Dimulai: 27 Jul 2026, 20.08 by Kepala Produksi

Selesai: 27 Jul 2026, 20.09 by Kepala Produksi

"dakdknk"

FOTO DOKUMENTASI

2

Perakitan Kendaraan

Belum Dimulai

Mulai Tahapan

Perakitan rangka, motor listrik, baterai, dan sistem kelistrikan.

3

Pemeriksaan Akhir

Belum Dimulai

Mulai Tahapan

Inspeksi kualitas (QC), uji fungsi kelistrikan, dan pengujian jalan.

4

Produksi Selesai

Belum Dimulai

Mulai Tahapan

Seluruh tahapan produksi telah selesai dan unit siap diserahkan.

Gambar 5.19 Antarmuka Pembaruan Tahapan Produksi

5.3 Implementasi Fitur Utama

Implementasi fitur utama menjelaskan penerapan fungsi inti yang menjadi fokus pada aplikasi pemesanan kendaraan listrik PT Marlip Indo Mandiri. Fitur utama yang diimplementasikan meliputi katalog produk dan formulir pemesanan, Digital Purchase Order, tanda tangan elektronik, pembayaran melalui Midtrans Snap, invoice, validasi pembayaran, serta pelacakan tahapan produksi. Setiap fitur dijelaskan berdasarkan alur yang berjalan pada aplikasi dan ditampilkan melalui hasil implementasi antarmuka.

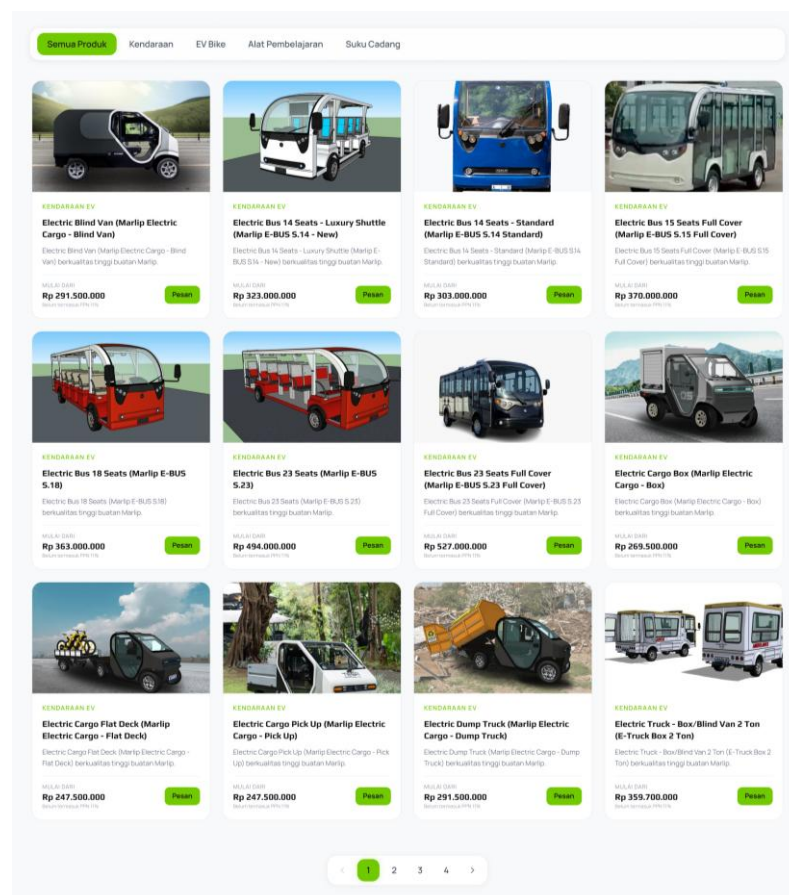
5.3.1 Implementasi Katalog Produk dan Formulir Pemesanan

Fitur katalog produk digunakan untuk menampilkan daftar produk yang tersimpan pada basis data. Produk yang ditampilkan pada katalog merupakan produk yang masih berstatus aktif, sehingga produk yang telah dinonaktifkan oleh Admin tidak akan muncul pada halaman katalog. Data produk ditampilkan secara dinamis dari basis data PostgreSQL melalui Prisma ORM.

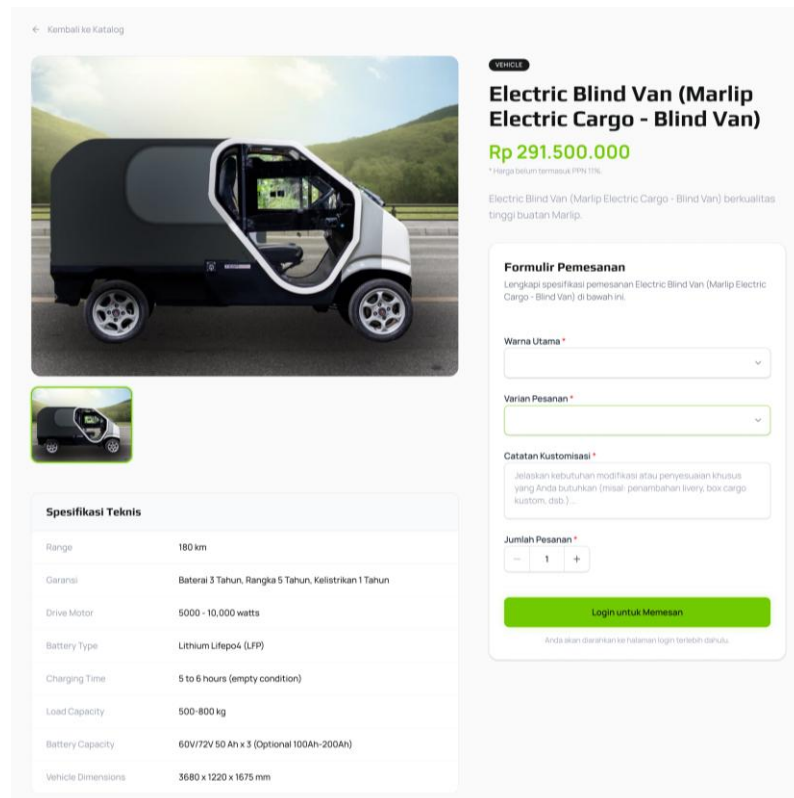
Pengunjung dapat melihat halaman katalog dan detail produk tanpa harus login terlebih dahulu. Namun, ketika pengguna akan mengajukan pesanan, sistem mengarahkan pengguna untuk login sebagai Konsumen. Pembatasan ini bertujuan agar data pesanan selalu terhubung dengan akun Konsumen yang valid.

Pada halaman detail produk, Konsumen dapat mengisi Formulir Pemesanan. Formulir tersebut memuat pilihan warna utama Hitam atau Putih, varian pesanan Standar atau Kustom, serta jumlah pesanan 1 sampai 20 unit. Jika Konsumen memilih varian Kustom, sistem menampilkan kolom catatan kebutuhan khusus yang wajib diisi. Sementara itu, pada varian Standar, Konsumen tidak perlu mengisi catatan tambahan.

Untuk produk kategori kendaraan dan EV Bike, sistem menyediakan Formulir Pemesanan langsung pada halaman detail produk. Sementara itu, produk kategori Alat Pembelajaran dan Suku Cadang diarahkan melalui pemesanan manual agar dapat dikonfirmasi lebih lanjut oleh pihak perusahaan.



Gambar 5.20 Tampilan Katalog Produk



← Kembali ke Katalog

VEHICLE

Electric Blind Van (Marlip Electric Cargo - Blind Van)

Rp 291.500.000
*Harga belum termasuk PPN 10%.

Electric Blind Van (Marlip Electric Cargo - Blind Van) berkualitas tinggi buatan Marlip.

Formulir Pemesanan
Lengkapi spesifikasi pemesanan Electric Blind Van (Marlip Electric Cargo - Blind Van) di bawah ini.

Warna Utama *

Varian Pesanan *

Catatan Kustomisasi *

Jumlah Pesanan *

1

Login untuk Memesan

Anda akan diarahkan ke halaman login terlebih dahulu.

Spesifikasi Teknis

Range	180 km
Garansi	Baterai 3 Tahun, Rangka 5 Tahun, Kelistrikan 1 Tahun
Drive Motor	5000 - 10,000 watts
Battery Type	Lithium Lifepo4 (LFP)
Charging Time	5 to 6 hours (empty condition)
Load Capacity	500-800 kg
Battery Capacity	60V/72V 50 Ah x 3 (Optional 100Ah-200Ah)
Vehicle Dimensions	3680 x 1220 x 1675 mm

Gambar 5.21 Tampilan Detail Produk dan Formulir Pemesanan

5.3.2 Implementasi Digital Purchase Order dan Tanda Tangan Elektronik

Digital Purchase Order digunakan sebagai dokumen transaksi yang memuat rincian pesanan, data Konsumen, produk yang dipesan, harga, estimasi pengerjaan, serta syarat dan ketentuan. Setelah Konsumen mengajukan pesanan, data pesanan akan masuk ke dashboard Admin dengan status Menunggu Review.

Admin kemudian meninjau rincian pesanan tersebut. Untuk pesanan varian Standar, sistem menggunakan harga setelah PPN dan estimasi pengerjaan 40 hari secara otomatis. Untuk pesanan varian Kustom, Admin menentukan harga final dan estimasi pengerjaan berdasarkan hasil peninjauan kebutuhan Konsumen.

Setelah data penawaran lengkap, Admin memberikan tanda tangan elektronik dan menerbitkan draf Digital Purchase Order. Konsumen kemudian meninjau draf PO tersebut. Apabila Konsumen menyetujui isi PO, Konsumen memberikan tanda tangan elektronik sebagai bentuk persetujuan. Setelah PO disetujui, sistem mengubah status pesanan menjadi Menunggu DP dan menerbitkan Invoice DP sebesar 50%.

← Kembali

Tinjau Draft Digital PO Anda
Admin telah memvalidasi rincian pesanan Anda. Silakan tinjau draft PO resmi di bawah. Klik 'Setuju Draft PO' untuk menerbitkan invoice DP 50% dan mulai pengiriman.

marlipindo@gmail.com
0811.2184.400

PT Marlip Indo Mandiri
Jl. Moch. Taha No. 223 Kab. Bandung, Jawa Barat, Indonesia
www.marlipindo.com

PURCHASE ORDER

Nomor : PO-2020-0017
Tanggal : 01 Juni 2020

Ship to (Buyer):
Budi Santoso
Dinas Lingkungan Hidup Bandung
Jawa - Bandung

To (Seller):
PT Marlip Indo Mandiri
Jl. Moch. Taha No. 223 Kab. Bandung, Jawa Barat, Indonesia
www.marlipindo.com

No	Model	Unit	Qty	Price	Total Price
1	ELECTRIC BLIND VAN (MARLIP/ELECTRIC CARGO - BLIND VAN) Warna/Model: Hitam Warna/Pemilihan: Standar Harga Dasar (termasuk ongkos kirim) Rp 251.500.000 PPN 1% Seller Rp 25.150.000 Harga sebelum diskon PPN Rp 276.650.000	Unit	1	251.500.000	251.500.000

Subtotal 251.500.000
PPN 1% 25.150.000
Discount -
Total (Curr.) Rp 323.565.000

NOTE

Down Payment: Rp 161.782.500 (50%)
Pembayaran: Rp 161.782.500 (50%)
Payment Term: (DP 50% & Pembayaran 50%)
Estimasi Pengiriman: 1-2 Hari
Gedung: L001001

SUPPLIER/VERSION
Purchasing

AUTHORIZED SIGNATURE
Budi Santoso

APPROVED
Finance

Nota: Dokumen ini adalah dokumen Purchase Order resmi dari PT Marlip Indo Mandiri yang diterbitkan secara elektronik.

Unduh PDF

Tolak Draft PO

Setuju Draft PO & Terbitkan Invoice

Gambar 5.22 Tampilan Digital Purchase Order

Tanda Tangan Elektronik Admin (Supplier) ✕

Silakan bubuhkan tanda tangan Anda pada pad di bawah ini sebagai perwakilan resmi PT Marlip Indo Mandiri untuk dokumen Purchase Order ini.

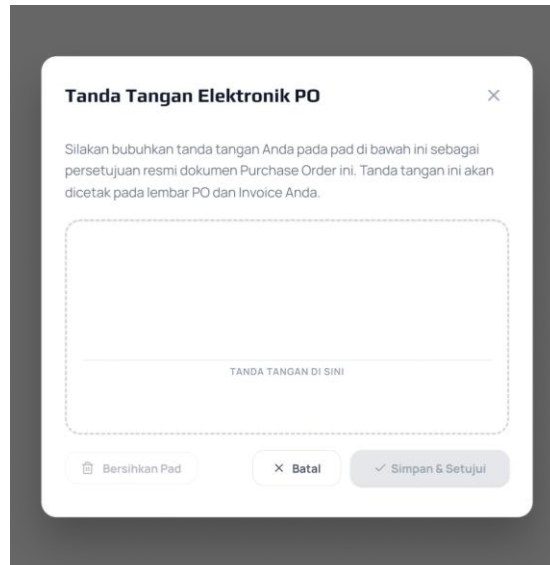
TANDA TANGAN DI SINI

Bersihkan Pad

Batal

Simpan & Setujui

Gambar 5.23 Tampilan Tanda Tangan Elektronik Admin



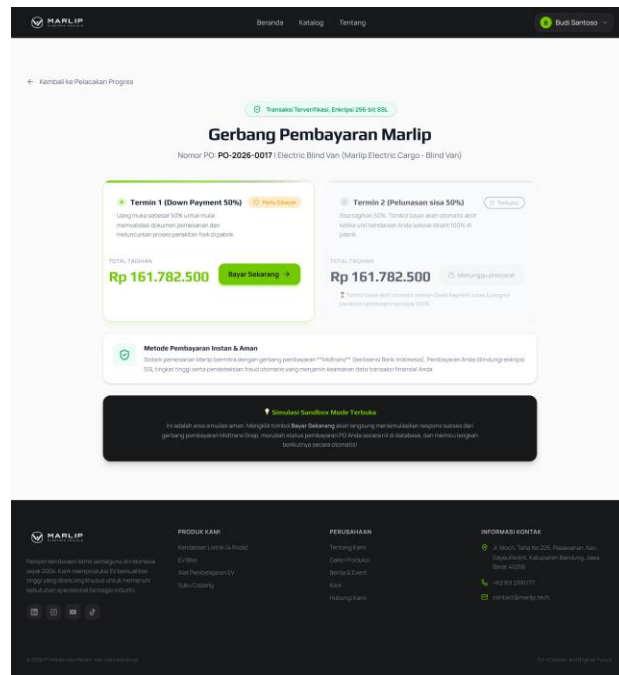
Gambar 5.24 Tampilan Tanda Tangan Elektronik Konsumen

5.3.3 Implementasi Pembayaran Midtrans Snap

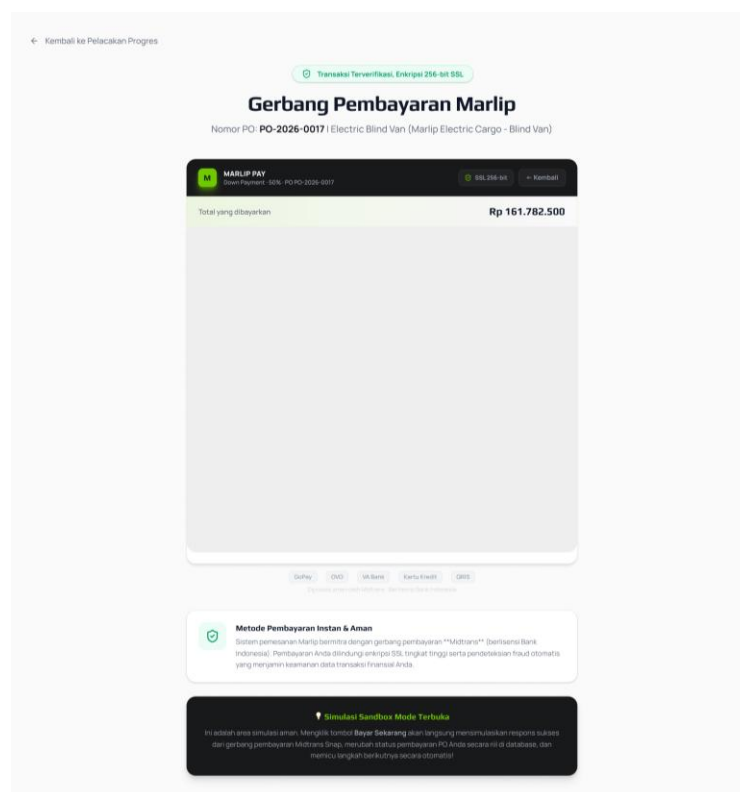
Pembayaran pada aplikasi dilakukan menggunakan Midtrans Snap API pada lingkungan *sandbox*. Sistem menerapkan pembayaran dua termin, yaitu *Down Payment* (DP) sebesar 50% dan pelunasan sebesar sisa tagihan. Pembayaran DP dapat dilakukan setelah Konsumen menyetujui draf PO, sedangkan pembayaran pelunasan tersedia setelah pembayaran DP berstatus *paid* dan seluruh tahapan produksi selesai.

Saat Konsumen menekan tombol pembayaran, sistem membuat atau mengambil data *Invoice* sesuai termin pembayaran. Setelah itu, sistem mengirimkan permintaan ke Midtrans untuk memperoleh *Snap Token*. Token tersebut digunakan untuk menampilkan antarmuka Midtrans Snap agar Konsumen dapat memilih metode pembayaran yang tersedia.

Sistem membatasi proses pembayaran sesuai status pesanan. Pembayaran DP hanya dapat dilakukan ketika pesanan berstatus Menunggu DP. Pembayaran pelunasan hanya dapat dilakukan apabila pembayaran DP telah berstatus *paid* dan seluruh tahapan produksi telah selesai.



Gambar 5.25 Tampilan Halaman Pembayaran Dua Termin



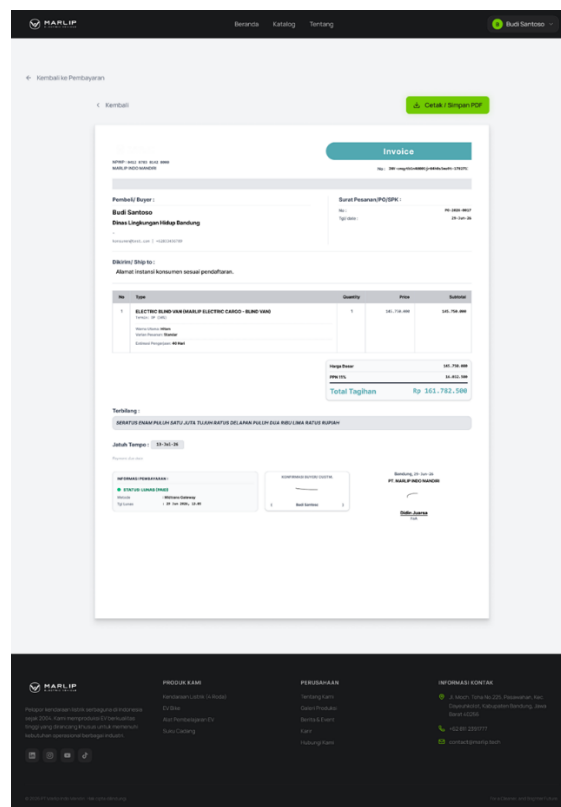
Gambar 5.26 Tampilan Midtrans Snap

5.3.4 Implementasi Invoice dan Validasi Pembayaran Webhook

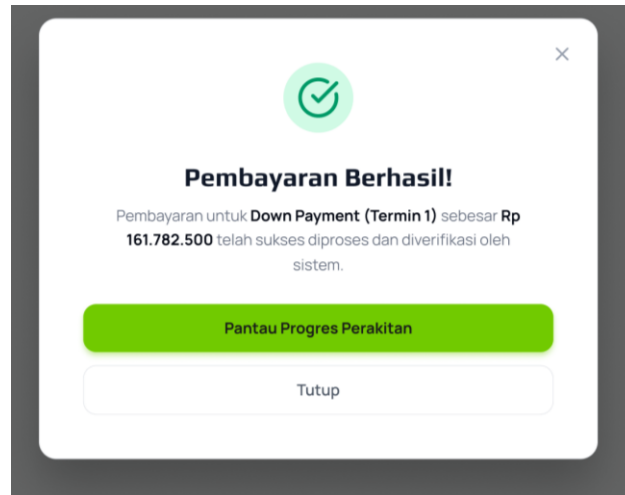
Invoice digunakan untuk menyimpan data tagihan pembayaran berdasarkan termin, yaitu DP dan pelunasan. Setiap Invoice menyimpan informasi termin pembayaran, total tagihan, status pembayaran, Midtrans Order ID, Snap Token, serta waktu pembayaran. Invoice DP diterbitkan setelah Konsumen menyetujui draf PO, sedangkan Invoice pelunasan diterbitkan setelah seluruh tahapan produksi selesai.

Validasi pembayaran dilakukan melalui webhook Midtrans. Ketika transaksi pembayaran diproses oleh Midtrans, sistem menerima notifikasi pembayaran melalui Route Handler webhook. Sebelum memperbarui data pembayaran, sistem melakukan validasi signature key, order ID, dan gross amount. Validasi tersebut dilakukan untuk memastikan bahwa notifikasi pembayaran benar-benar sesuai dengan transaksi yang tersimpan di basis data.

Apabila pembayaran DP berhasil diverifikasi, sistem memperbarui status Invoice menjadi paid, mengubah status pesanan menjadi DP Diterima, dan menyiapkan empat tahapan produksi. Apabila pembayaran pelunasan berhasil diverifikasi setelah seluruh tahapan produksi selesai, sistem memperbarui status pesanan menjadi Siap Dikirim.



Gambar 5.27 Tampilan Invoice Pembayaran



Gambar 5.28 Tampilan Status Pembayaran Berhasil

5.3.5 Implementasi Pelacakan Tahapan Produksi

Fitur pelacakan tahapan produksi digunakan untuk menampilkan perkembangan pengerjaan kendaraan kepada Konsumen. Informasi yang ditampilkan bukan berasal dari pelacakan *real-time* otomatis, melainkan berdasarkan pembaruan yang dilakukan oleh Kepala Produksi melalui dashboard produksi.

Setelah pembayaran DP berhasil diverifikasi, sistem menyiapkan empat tahapan produksi, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai. Konsumen dapat melihat status setiap tahapan dalam bentuk *checklist*, beserta riwayat waktu pembaruan, pengguna yang melakukan pembaruan, catatan, dan foto dokumentasi produksi.

Setelah seluruh tahapan produksi selesai, sistem menerbitkan *Invoice* pelunasan. Status pesanan tetap berada pada tahap Proses Perakitan sampai pembayaran pelunasan berhasil diverifikasi. Setelah pelunasan berhasil, status pesanan berubah menjadi Siap Dikirim.

Beranda
Katalog
Tentang
Buji Santoso

Order ID: PO-2025-0017

Electric Blind Van (Marlip Electric Cargo - Blind Van)
Dipesan pada 29 Juni 2025

[Lihat Dokumen PO](#)
[Pembayaran](#)

Status Pemesanan

- PO Divalidasi**
Admin telah menyetujui Digital PO.
- DP Diterima**
Pembayaran Termin 1 telah diverifikasi.
- Proses Perakitan**
Kendaraan sedang dirakit di pabrik.
- Siap Dikirim**
Kendaraan telah lulus QC dan siap dikirim.

Ringkasan Finansial

Termin 1 (DP 50%)	Rp 161.782.500
✓ Lunas	
Termin 2 (Pelunasan)	Rp 161.782.500
⌚ Termino (Progress: 100%)	
Total Harga Pemesanan	Rp 323.565.000
Bayar Pelunasan	
Pembayaran Pelunasan akan menambah progress perakitan 50%.	

Dokumentasi Tahapan Produksi (Tugas Akhir)

Status pengisian foto untuk kendaraan listrik Anda berdasarkan urutan below rincian gambar pabrik.

Tahap 1
[Dokumentasi](#)

Frame Assembly

Pengisian rangka body utama dan sistem suspensi kendaraan listrik.

Tahap 2
[Dokumentasi](#)

Painting

Pengisian body car menggunakan teknik cat oven produksi.

Tahap 3
[Dokumentasi](#)

Finishing

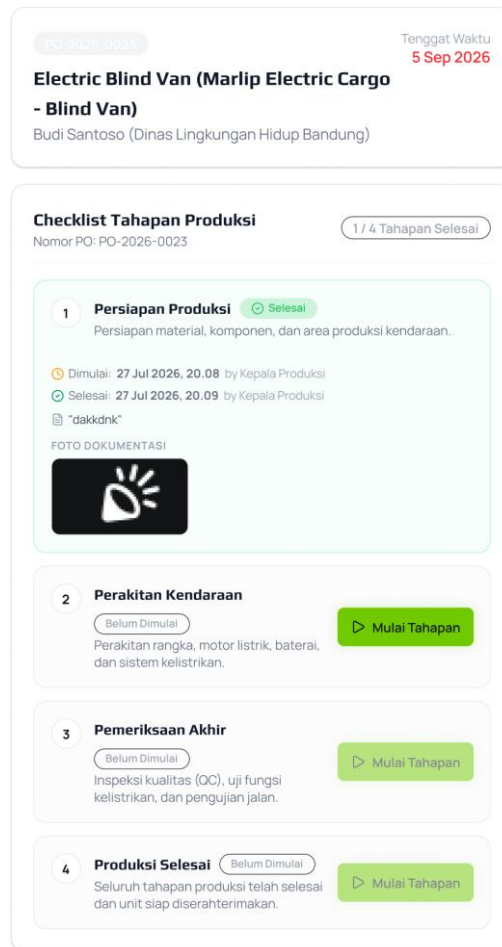
Pemasangan komponen interior bus, busang, interior, dan set motor DC.

Tahap 4
[Dokumentasi](#)

Delivery

Kendaraan siap dikirim ke dealer tujuan konsumen via armada logistik.

Gambar 5.29 Tampilan Pelacakan Tahapan Produksi Konsumen



Gambar 5.30 Tampilan Pembaruan Tahapan Produksi Kepala Produksi

5.4 Pengujian Fungsional Black Box

Pengujian fungsional dilakukan menggunakan metode Black Box Testing. Pengujian ini berfokus pada kesesuaian keluaran sistem terhadap masukan yang diberikan tanpa melihat struktur kode program secara langsung. Tujuan dari pengujian ini adalah memastikan bahwa setiap fitur utama pada aplikasi pemesanan kendaraan listrik PT Marlip Indo Mandiri berjalan sesuai dengan kebutuhan fungsional yang telah dirancang pada Bab 4.

Pengujian dilakukan berdasarkan alur penggunaan sistem oleh Pengunjung, Konsumen, Admin, dan Kepala Produksi. Aspek yang diuji meliputi akses katalog, pengajuan pesanan, pengelolaan produk, peninjauan PO, tanda tangan elektronik, pembayaran dua termin melalui Midtrans Snap, validasi pembayaran, pembaruan tahapan

produksi, serta pembatasan hak akses pengguna.

5.4.1 Pengujian Katalog dan Pemesanan

Tabel 5.7 Pengujian Katalog dan Pemesanan

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-01-01	Akses katalog tanpa login	Pengunjung membuka halaman katalog produk tanpa melakukan login	Sistem menampilkan daftar produk aktif yang tersedia	Sistem menampilkan daftar produk aktif yang tersedia	Pass
TC-01-02	Akses detail produk tanpa login	Pengunjung memilih salah satu produk pada halaman katalog	Sistem menampilkan halaman detail produk	Sistem menampilkan halaman detail produk	Pass
TC-01-03	Pengunjung mencoba mengajukan pesanan	Pengunjung menekan tombol pemesanan pada halaman detail produk	Sistem mengarahkan pengguna ke halaman login	Sistem mengarahkan pengguna ke halaman login	Pass
TC-01-04	Konsumen mengajukan pesanan varian Standar	Konsumen login, memilih warna, varian Standar, jumlah pesanan, lalu menekan tombol ajukan pesanan	Sistem menyimpan pesanan tanpa mewajibkan catatan kebutuhan khusus	Sistem menyimpan pesanan tanpa mewajibkan catatan kebutuhan khusus	Pass
TC-01-05	Konsumen mengajukan pesanan varian Kustom	Konsumen memilih varian Kustom, mengisi catatan kebutuhan khusus, lalu mengajukan pesanan	Sistem menyimpan pesanan berstatus Menunggu Review	Sistem menyimpan pesanan berstatus Menunggu Review	Pass
TC-01-06	Catatan Kustom dikosongkan	Konsumen memilih varian Kustom tetapi tidak mengisi catatan kebutuhan khusus	Sistem menolak pengajuan dan menampilkan validasi bahwa catatan wajib diisi	Sistem menolak pengajuan dan menampilkan validasi bahwa catatan wajib diisi	Pass
TC-01-07	Jumlah pesanan di luar batas	Konsumen mengisi jumlah pesanan kurang dari 1 atau lebih dari 20 unit	Sistem menolak input jumlah pesanan yang tidak sesuai	Sistem menolak input jumlah pesanan yang tidak sesuai	Pass

5.4.2 Pengujian Pengelolaan Produk Admin

Tabel 5.8 Pengujian Pengelolaan Produk Admin

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-02-01	Admin menambahkan produk baru	Admin membuka halaman manajemen produk, mengisi data produk, lalu menyimpan data	Sistem menyimpan produk baru ke basis data	Sistem menyimpan produk baru ke basis data	Pass
TC-02-02	Validasi kategori produk	Admin mengisi kategori produk yang tidak sesuai dengan kategori yang tersedia	Sistem menolak data produk dan menampilkan pesan validasi kategori	Sistem menolak data produk dan menampilkan pesan validasi kategori	Pass
TC-02-03	Validasi harga dasar produk	Admin mengisi harga dasar tidak valid pada produk kendaraan atau EV Bike	Sistem menolak data produk dan menampilkan pesan validasi harga dasar	Sistem menolak data produk dan menampilkan pesan validasi harga dasar	Pass
TC-02-04	Admin memperbaiki data produk	Admin memilih produk, mengubah data produk, lalu menyimpan perubahan	Sistem memperbaiki data produk pada basis data	Sistem memperbaiki data produk pada basis data	Pass
TC-02-05	Admin menonaktifkan produk	Admin menekan aksi nonaktifkan pada salah satu produk	Sistem mengubah status produk menjadi tidak aktif dan produk tidak tampil pada katalog	Sistem mengubah status produk menjadi tidak aktif dan produk tidak tampil pada katalog	Pass
TC-02-06	Pengunjung membuka produk nonaktif	Pengunjung membuka detail produk yang sudah dinonaktifkan	Sistem tidak menampilkan produk nonaktif pada katalog atau detail produk	Sistem tidak menampilkan produk nonaktif pada katalog atau detail produk	Pass

5.4.3 Pengujian Peninjauan dan Persetujuan PO

Tabel 5.9 Pengujian Peninjauan dan Persetujuan PO

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-03-01	Admin membuka pesanan masuk	Admin membuka dashboard Admin dan memilih pesanan berstatus Menunggu Review	Sistem menampilkan rincian pesanan Konsumen	Sistem menampilkan rincian pesanan Konsumen	Pass
TC-03-02	Admin menerbitkan PO varian Standar	Admin meninjau pesanan Standar dan memberikan tanda tangan elektronik	Sistem menggunakan harga setelah PPN dan estimasi 40 hari secara otomatis	Sistem menggunakan harga setelah PPN dan estimasi 40 hari secara otomatis	Pass
TC-03-03	Admin menerbitkan PO varian Kustom	Admin mengisi harga final, estimasi pengerjaan, catatan penawaran, dan tanda tangan elektronik	Sistem menerbitkan draf PO untuk ditinjau Konsumen	Sistem menerbitkan draf PO untuk ditinjau Konsumen	Pass
TC-03-04	Harga final Kustom tidak diisi	Admin mencoba menerbitkan draf PO Kustom tanpa mengisi harga final	Sistem menolak penerbitan draf PO dan menampilkan validasi	Sistem menolak penerbitan draf PO dan menampilkan validasi	Pass
TC-03-05	Estimasi Kustom tidak diisi	Admin mencoba menerbitkan draf PO Kustom tanpa mengisi estimasi pengerjaan	Sistem menolak penerbitan draf PO dan menampilkan validasi	Sistem menolak penerbitan draf PO dan menampilkan validasi	Pass
TC-03-06	Konsumen menyetujui draf PO	Konsumen membuka draf PO, memberikan tanda tangan elektronik, lalu menyetujui PO	Sistem mengubah status pesanan menjadi Menunggu DP dan menerbitkan Invoice DP	Sistem mengubah status pesanan menjadi Menunggu DP dan menerbitkan Invoice DP	Pass
TC-03-07	Konsumen menyetujui PO tanpa tanda tangan	Konsumen mencoba menyetujui draf PO tanpa tanda tangan elektronik	Sistem menolak persetujuan dan menampilkan validasi tanda tangan	Sistem menolak persetujuan dan menampilkan validasi tanda tangan	Pass
TC-03-08	Cetak atau simpan Digital PO	Konsumen membuka dokumen PO dan menekan fitur cetak/simpan PDF	Sistem membuka dialog cetak peramban	Sistem membuka dialog cetak peramban	Pass
TC-03-09	Cetak atau simpan Invoice	Konsumen membuka Invoice dan menekan fitur cetak/simpan PDF	Sistem membuka dialog cetak peramban	Sistem membuka dialog cetak peramban	Pass

5.4.4 Pengujian Pembayaran DP dan Pelunasan

Tabel 5.10 Pengujian Pembayaran DP dan Pelunasan

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-04-01	Konsumen membuka halaman pembayaran DP	Konsumen membuka halaman pembayaran setelah PO disetujui	Sistem menampilkan tagihan DP sebesar 50%	Sistem menampilkan tagihan DP sebesar 50%	Pass
TC-04-02	Membuat Snap Token DP	Konsumen menekan tombol bayar DP	Sistem menampilkan antarmuka Midtrans Snap	Sistem menampilkan antarmuka Midtrans Snap	Pass
TC-04-03	Pembayaran DP pada status tidak sesuai	Konsumen melakukan pembayaran DP pada status pesanan yang tidak sesuai	Sistem menolak pembayaran dan tidak membuat Snap Token	Sistem menolak pembayaran dan tidak membuat Snap Token	Pass
TC-04-04	Webhook pembayaran DP berhasil	Midtrans mengirim webhook pembayaran DP berstatus paid	Sistem mengubah status Invoice menjadi paid, status pesanan menjadi DP Diterima, dan sistem menyiapkan empat tahapan produksi	Sistem mengubah status Invoice menjadi paid, status pesanan menjadi DP Diterima, dan sistem menyiapkan empat tahapan produksi	Pass

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-04-05	Pelunasan sebelum seluruh tahapan selesai	Konsumen mencoba melakukan pelunasan ketika masih terdapat tahapan produksi yang belum berstatus Selesai.	Sistem menolak pembayaran pelunasan	Sistem menolak pembayaran pelunasan	Pass
TC-04-06	Konsumen membuka pembayaran pelunasan	Konsumen membuka halaman pembayaran setelah seluruh tahapan produksi selesai	Sistem menampilkan tagihan pelunasan sebesar sisa tagihan	Sistem menampilkan tagihan pelunasan sebesar sisa tagihan	Pass
TC-04-07	Webhook pembayaran pelunasan berhasil	Midtrans mengirim webhook pembayaran pelunasan berstatus paid	Sistem mengubah status Invoice menjadi paid dan status pesanan menjadi Siap Dikirim	Sistem mengubah status Invoice menjadi paid dan status pesanan menjadi Siap Dikirim	Pass
TC-04-08	Validasi gross amount webhook	Sistem menerima webhook dengan nilai gross amount yang tidak sesuai	Sistem menolak webhook dan tidak memperbarui data pembayaran	Sistem menolak webhook dan tidak memperbarui data pembayaran	Pass
TC-04-09	Validasi signature key webhook	Sistem menerima webhook dengan signature key tidak valid	Sistem menolak webhook dan tidak memperbarui status transaksi	Sistem menolak webhook dan tidak memperbarui status transaksi	Pass

5.4.5 Pengujian Pembaruan Tahapan Produksi

Tabel 5.11 Pengujian Pembaruan Tahapan Produksi

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-05-01	Kepala Produksi membuka dashboard produksi	Kepala Produksi login dan membuka dashboard produksi	Sistem menampilkan daftar pesanan yang telah memasuki tahap produksi beserta status tahapannya	Sistem menampilkan daftar pesanan dan status tahapannya	Pass
TC-05-02	Memulai tahapan produksi pertama	Kepala Produksi memilih tahapan Persiapan Produksi dan menekan tombol Mulai Tahapan	Status tahapan berubah menjadi Sedang Diproses, waktu serta pengguna yang memulai tersimpan, dan status PO menjadi Proses Perakitan	Status tahapan, waktu, pengguna, dan status PO berhasil diperbarui	Pass
TC-05-03	Memulai tahapan tanpa memenuhi urutan	Kepala Produksi mencoba memulai tahap Perakitan kendaraan sebelum Persiapan Produksi selesai	Sistem menolak aksi karena tahapan sebelumnya belum selesai	Sistem menolak aksi dan menampilkan pesan validasi	Pass
TC-05-04	Menyelesaikan tahapan produksi	Kepala Produksi menambahkan catatan dan foto, kemudian menekan tombol Selesaikan Tahapan	Status tahapan berubah menjadi Selesai serta catatan, foto, waktu, dan pengguna penyelesaian tersimpan	Data tahapan dan dokumentasi berhasil disimpan	Pass
TC-05-05	Menyelesaikan tahapan yang belum dimulai	Kepala Produksi mencoba menyelesaikan tahapan berstatus Belum Dimulai	Sistem menolak aksi karena tahapan belum dimulai	Sistem menolak aksi dan menampilkan pesan validasi	Pass
TC-05-06	Mengubah tahapan yang telah selesai	Kepala Produksi mencoba mengubah kembali tahapan berstatus Selesai	Sistem menolak perubahan karena tahapan yang selesai bersifat tetap	Sistem menolak perubahan tahapan	Pass
TC-05-07	Menyelesaikan seluruh tahapan produksi	Kepala Produksi menyelesaikan tahapan Produksi Selesai setelah tiga tahapan sebelumnya selesai	Sistem menerbitkan Invoice pelunasan satu kali dan status pesanan tetap Proses Perakitan sampai pelunasan dibayar	Invoice pelunasan berhasil diterbitkan dan status pesanan tetap Proses Perakitan	Pass

5.4.6 Pengujian Hak Akses Pengguna

Tabel 5.12 Pengujian Hak Akses Pengguna

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-06-01	Pengunjung membuka dashboard	Pengunjung membuka URL dashboard tanpa login	Sistem mengarahkan pengguna ke halaman login	Sistem mengarahkan pengguna ke halaman login	Pass
TC-06-02	Konsumen membuka dashboard Admin	Konsumen mencoba mengakses halaman dashboard Admin	Sistem menolak akses dan mengarahkan pengguna sesuai role	Sistem menolak akses dan mengarahkan pengguna sesuai role	Pass
TC-06-03	Admin mengakses dashboard Konsumen	Admin mencoba membuka dashboard Konsumen	Sistem menolak akses dan mengarahkan Admin ke dashboard Admin	Sistem menolak akses dan mengarahkan Admin ke dashboard Admin	Pass
TC-06-04	Kepala Produksi membuka dashboard Admin	Kepala Produksi mencoba membuka dashboard Admin	Sistem menolak akses dan mengarahkan pengguna ke dashboard Produksi	Sistem menolak akses dan mengarahkan pengguna ke dashboard Produksi	Pass
TC-06-05	Konsumen membuka pesanan milik pengguna lain	Konsumen membuka URL pesanan yang bukan miliknya	Sistem menolak akses terhadap data pesanan tersebut	Sistem menolak akses terhadap data pesanan tersebut	Pass
TC-06-06	Admin mengelola produk	Admin membuka halaman manajemen produk	Sistem mengizinkan Admin mengelola produk	Sistem mengizinkan Admin mengelola produk	Pass
TC-06-07	Kepala Produksi mengubah harga PO	Kepala Produksi mencoba mengubah data harga atau penawaran PO	Sistem menolak akses karena perubahan harga hanya dapat dilakukan Admin	Sistem menolak akses karena perubahan harga hanya dapat dilakukan Admin	Pass
TC-06-08	Pengunjung mengakses endpoint pemesanan	Pengunjung mencoba mengirim pengajuan pesanan tanpa login	Sistem menolak permintaan dengan status tidak terautentikasi	Sistem menolak permintaan dengan status tidak terautentikasi	Pass

5.4.7 Rekapitulasi Hasil Pengujian Black Box

Berdasarkan rekapitulasi pada Tabel 5.13, seluruh skenario pengujian fungsional berhasil dijalankan dengan persentase keberhasilan sebesar 100%. Hasil ini menunjukkan bahwa fitur utama aplikasi pemesanan kendaraan listrik PT Marlip Indo Mandiri telah berjalan sesuai dengan kebutuhan fungsional yang dirancang pada Bab 4.

Tabel 5.13 Rekapitulasi Hasil Pengujian Black Box

No	Kelompok Pengujian	Rentang Test Case	Jumlah Test Case	Berhasil	Gagal	Persentase Keberhasilan
1	Katalog dan Pemesanan	TC-01-01 s.d. TC-01-07	7	7	0	100%
2	Pengelolaan Produk Admin	TC-02-01 s.d. TC-02-06	6	6	0	100%
3	Peninjauan dan Persetujuan PO	TC-03-01 s.d. TC-03-09	9	9	0	100%
4	Pembayaran DP dan Pelunasan	TC-04-01 s.d. TC-04-09	9	9	0	100%
5	Pembaruan Tahapan Produksi	TC-05-01 s.d. TC-05-07	7	7	0	100%
6	Hak Akses Pengguna	TC-06-01 s.d. TC-06-08	8	8	0	100%
Total			46	46	0	100%

5.5 Deployment

Deployment merupakan tahap penerapan aplikasi dari lingkungan pengembangan lokal ke lingkungan daring agar sistem dapat diakses melalui internet. Pada penelitian ini, aplikasi diterapkan menggunakan Vercel sebagai platform hosting aplikasi web dan Neon PostgreSQL sebagai layanan basis data cloud. Source code aplikasi disimpan pada GitHub, kemudian dihubungkan ke Vercel untuk proses build dan deployment.

Konfigurasi layanan seperti koneksi basis data, autentikasi, dan Midtrans Snap API disimpan dalam bentuk environment variable pada dashboard Vercel. Hal ini dilakukan agar data sensitif seperti URL database, secret autentikasi, dan kunci Midtrans tidak ditulis langsung di dalam source code. Integrasi pembayaran pada tahap deployment tetap menggunakan Midtrans sandbox karena transaksi yang dilakukan masih bersifat simulasi pengujian.

Setelah proses deployment selesai, aplikasi dapat diakses melalui URL deployment dan fitur utama tetap dapat berjalan sesuai role pengguna. Pengunjung dapat melihat katalog produk, Konsumen dapat mengajukan pesanan dan melakukan pembayaran, Admin dapat mengelola produk serta meninjau pesanan, sedangkan Kepala Produksi dapat memperbarui tahapan produksi.

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi, dan pengujian yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Penelitian ini berhasil merancang dan membangun aplikasi pemesanan kendaraan listrik berbasis web pada PT Marlip Indo Mandiri menggunakan Next.js, Prisma ORM, PostgreSQL, dan Midtrans Snap API pada lingkungan *sandbox*.
2. Aplikasi yang dibangun menyediakan alur pemesanan kendaraan secara digital melalui katalog produk, formulir pemesanan varian Standar atau Kustom, peninjauan pesanan oleh Admin, penerbitan Digital Purchase Order, serta persetujuan PO menggunakan tanda tangan elektronik.
3. Aplikasi berhasil menyediakan pembayaran dua termin melalui Midtrans Snap, yaitu *Down Payment* (DP) sebesar 50% dan pelunasan sisa tagihan. Sistem juga dapat memperbarui status pembayaran dan status pesanan melalui validasi *webhook* Midtrans.
4. Aplikasi berhasil menyediakan fitur pelacakan tahapan produksi berdasarkan pembaruan Kepala Produksi. Sistem menampilkan empat tahapan produksi berbentuk *checklist*, yaitu Persiapan Produksi, Perakitan Kendaraan, Pemeriksaan Akhir, dan Produksi Selesai, beserta catatan dan foto dokumentasi.
5. Berdasarkan hasil pengujian *Black Box*, fitur utama aplikasi berjalan sesuai dengan kebutuhan yang telah dirancang, meliputi katalog produk, pemesanan, pengelolaan produk, persetujuan PO, pembayaran, validasi *webhook*, pelacakan tahapan produksi, dan pembatasan hak akses pengguna.

6.2 Saran

Berdasarkan kesimpulan dan hasil pengembangan sistem yang telah dilakukan, terdapat beberapa saran yang dapat digunakan untuk pengembangan penelitian selanjutnya, yaitu:

1. Integrasi Midtrans yang masih menggunakan lingkungan sandbox dapat dikembangkan ke mode production agar sistem dapat digunakan untuk transaksi pembayaran nyata oleh PT Marlip Indo Mandiri.
2. Sistem dapat dikembangkan dengan menambahkan integrasi notifikasi WhatsApp agar Konsumen, Admin, dan Kepala Produksi menerima informasi terbaru terkait status pesanan, pembayaran, dan tahapan produksi secara lebih cepat.
3. Tahapan produksi dapat dikembangkan lebih detail sesuai standar operasional perusahaan, misalnya melalui penambahan tahapan pemeriksaan komponen, pengecatan, atau pengujian kendaraan.
4. Sistem dapat dikembangkan dengan menambahkan audit log untuk mencatat aktivitas penting, seperti perubahan status pesanan, pembaruan harga, pembaruan tahapan produksi, dan validasi pembayaran.
5. Penelitian selanjutnya dapat memperluas sistem pada aspek manajemen stok komponen, integrasi pengiriman, layanan purna jual, serta pengelolaan garansi kendaraan.

DAFTAR PUSTAKA

- [ALB23] Albrecht, M. G., Green, M., & Hoffman, L., “Principles of Marketing”, OpenStax, Houston, 2023. Tersedia: https://assets.openstax.org/oscms-prodcms/media/documents/Principles_Marketing-WEB.pdf
- [ARR23] Arribe, E., Jayema, A., & Putra, F. W., “Perancangan Sistem Informasi Penjualan Motor Listrik PT. Talabu”, Pixel: Jurnal Ilmiah Komputer Grafis, Vol. 16, No. 2, pp. 26–38, 2023.
- [COD70] Codd, E. F., “A Relational Model for Large Shared Data Banks”, Communications of the ACM, Jilid 13, Nomor 6, Juni 1970.
- [FAT23] Fatman, Y., Nafisah, N. K., & Pambudi, P. B., “Implementasi Payment Gateway dengan Menggunakan Midtrans pada Website UMKM Geberco”, Jurnal KomtekInfo, Vol. 10, pp. 64–72, 2023.
- [FRE19] Freeman, A., “Pro React 16”, Edisi 1, Apress, Berkeley, 2019.
- [GIB24] Gibran, C., Dewi, A. R., & Hadinata, E., “Implementasi Framework Laravel untuk Pengembangan Website Penjualan Ayam Potong dengan Pemanfaatan Midtrans Menggunakan Metode FAST”, Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI), Vol. 7, No. 1, pp. 246–253, 2024. Tersedia: <https://ejournal.sisfokomtek.org/index.php/jikom/article/download/2920/2091/21379>
- [IND23] Indonesia, Pemerintah Republik, “Peraturan Presiden Republik Indonesia Nomor 79 Tahun 2023 tentang Perubahan atas Peraturan Presiden Nomor 55 Tahun 2019 tentang Percepatan Program Kendaraan Bermotor Listrik Berbasis Baterai untuk Transportasi Jalan”, Jakarta, 2023.
- [JON15] Jones, M., Bradley, J., & Sakimura, N., “JSON Web Token (JWT)”, RFC 7519, Internet Engineering Task Force, Mei 2015. Tersedia: <https://www.rfc-editor.org/rfc/rfc7519>
- [KAL23] Kalsid, E., & Septanto, H., “Rancang Bangun Aplikasi E-Document Purchase Order sebagai Media Pendistribusian dan Penyimpanan Data Digital Berbasis Web”, Jurnal Informatika dan Teknologi Komputer, Vol. 3, No. 2, pp. 146–154, 2023. Tersedia: <https://researchhub.id/index.php/jitek/article/download/1844/1552/6753>
- [KES25] Kesuma, W., & Ramadhan, M. I., “Perancangan Aplikasi Tracking Proses Produksi pada PT Makmur Bintang Plastindo Berbasis Web”, Jurnal Sistem Informasi Komputer (SIKOM), Vol. 2, No. 1, 2025. Tersedia: <https://journal.beaninstitute.id/index.php/sikom/article/download/47/64>
- [LAU16] Laudon, K. C., & Traver, C. G., “E-Commerce 2016: Business, Technology, Society”, Edisi 12, Pearson, Boston, 2016.
- [MAR26] PT Marlip Indo Mandiri, “Marlip – Electric Vehicle Manufacturing”, tersedia: Juli 2026, <https://marlip.tech/>, 2026.
- [MIC24] Microsoft, “TypeScript Documentation: The Handbook”, tersedia: Juli 2026, <https://www.typescriptlang.org/docs/>, 2024.
- [MID26] Midtrans, “Built-in Interface (Snap)”, Dokumentasi Resmi, tersedia: Juli 2026, <https://docs.midtrans.com/docs/snap>, 2026.
- [MUS15] Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H., “Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus: Aplikasi Prediksi Kelulusan SNMPTN)”, Jurnal Ilmiah Teknologi Informasi Terapan, Vol. 1, No. 3, pp. 31–36, Agustus 2015.
- [NEO26] Neon, “Connection Pooling”, Dokumentasi Resmi, tersedia: Juli 2026, <https://neon.com/docs/connect/connection-pooling>, Februari 2026.
- [NID12] Nidhra, S., & Dondeti, J., “Black Box and White Box Testing Techniques – A Literature Review”, International Journal of Embedded Systems and Applications, Vol. 2, No. 2, 2012.
- [POS24] PostgreSQL Global Development Group, “PostgreSQL 16 Documentation”, tersedia: Juli 2026, <https://www.postgresql.org/docs/16/>, 2024.
- [PRE20] Pressman, R. S., & Maxim, B. R., “Software Engineering: A Practitioner’s Approach”, Edisi 9, McGraw-Hill Education, New York, 2020.

- [PRI26] Prisma, “Prisma ORM”, Dokumentasi Resmi, tersedia: Juli 2026, <https://www.prisma.io/docs/orm>, 2026.
- [RAF14] Rafiei, H., Rabbani, M., & Kokabi, R., “Multi-site Production Planning in Hybrid Make-to-Stock/Make-to-Order Production Environment”, Journal of Industrial Engineering International, Vol. 10, Artikel 60, 2014. Tersedia: <https://doi.org/10.1007/s40092-014-0060-z>
- [RAI25] Raisinghani, M. S., “Foundations of Information Systems”, OpenStax, Houston, 2025. Tersedia: https://assets.openstax.org/oscms-prodcms/media/documents/Foundations_of_Information_Systems_-_WEB.pdf
- [ROC23] Rochman, F. N., Aziz, R. A., Setiawan, A. C., & Susilo, H., “Perancangan Sistem Penjualan Mobil Berbasis Web pada Dealer dengan Menggunakan Metode Waterfall”, Jurnal Minfo Polgan, Vol. 12, No. 2, 2023. Tersedia: <https://jurnal.polgan.ac.id/index.php/jmp/article/download/13153/2114/15003>
- [SAN00] Sandhu, R., Ferraiolo, D. F., & Kuhn, D. R., “The NIST Model for Role-Based Access Control: Towards a Unified Standard”, Proceedings of the Fifth ACM Workshop on Role-Based Access Control, pp. 47–63, 2000. Tersedia: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=916402
- [TAI24] Tailwind Labs, “Tailwind CSS Documentation”, tersedia: Juli 2026, <https://tailwindcss.com/docs>, 2024.
- [TUR18] Turban, E., Outland, J., King, D., Lee, J. K., Liang, T. P., & Turban, D. C., “Electronic Commerce 2018: A Managerial and Social Networks Perspective”, Edisi 9, Springer, Cham, 2018.
- [VER26] Vercel, Inc., “Getting Started: Route Handlers”, Next.js Documentation, tersedia: Juli 2026, <https://nextjs.org/docs/app/getting-started/route-handlers>, Februari 2026.
- [WAT14] Watt, A., “Database Design”, Edisi 2, BCcampus, Victoria, 2014. Tersedia: <https://open.umn.edu/opentextbooks/textbooks/database-design-2nd-edition>

LAMPIRAN

Lampiran A Berita Acara Wawancara

BERITA ACARA WAWANCARA

Dengan ini menerangkan bahwa mahasiswa yang bertanda tangan di bawah ini :

Nama : Daffaa Aprilino

NRP : 223040025

Fakultas : Teknik

Program Studi : Teknik Informatika

Universitas : Universitas Pasundan

Telah melakukan wawancara dengan

Nama : Ir. Masrah Marang, M.M.

Jabatan : Direktur Utama

Perusahaan : PT Marlip Indomandiri

Dalam rangka penyusunan Tugas Akhir Semester VII tahun akademik 2025-2026 yang berjudul "RANCANG BANGUN WEBSITE PEMESANAN KENDARAAN LISTRIK MENGGUNAKAN PAYMENT GATEWAY MIDTRANS" (Studi Kasus : PT Marlip Indomandiri).

Yang Diwawancara,



Ir. Masrah Marang, M.M.

Bandung, 2 Februari 2026

Pewawancara



Daffaa Aprilino

Catatan wawancara:

A. Sistem Pemesanan dan Pembayaran
 Alur Pemesanan: Menggunakan sistem pre-order karena spesifikasi kendaraan disesuaikan dengan permintaan konsumen memilih produk sesuai spesifikasi, lalu menyetor dokumen PD
 Pembayaran: wajib down payment (DP) sebesar 50% awal. Sisanya dilunasi sebelum barang dikirim. belum ada cicilan
 Masa produksi: 14 sampai 30 hari
 Dokumen: konsumen wajib menandatangani dan mengirimkan kembali dokumen PD
 B. manajemen stock dan batasi pesan
 stock produk: karena sistem "made by order", tidak ada istilah stock habis semua pesanan yang masuk otomatis pre-order
 Pembatasan: jika konsumen membatalkan pesanan secara sepihak sebelum membayar maka uang DP dinyatakan hangus
 C. kebutuhan admin
 Admin membuat laporan lengkap di sistem, meliputi data pengunjung, kategori produk

Bandung, 2 Februari 2026

Yang Diwawancara,



Ir. Masrah Marang, M.M.

Pewawancara



Daffaa Aprilino

Lampiran B Contoh Purchase Order

**MARLIP**
PT MARLIP INDO MANDIRI

PURCHASES ORDER

 marlipindo@gmail.com
 0811.2194.400

Nomor :
Tanggal :
To (Seller):

Ship to (Buyer):
PT Marlip Indo Mandiri
Jl. Moch. Toha No. 225 Kab. Bandung
Jawa Barat, Indonesia

Address :

attn :

attn :

No	Model	Unit	Qty	Price	Total Price
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

Subtotal
PPN 11%
Discount
Total (Curr.)

Note
Down Payment :
Payment Term : 100% after delivery to Marlip, pt. warehouse in Bandung
Delivery Location : PT Marlip Indo Mandiri, Jl. M Toha No. 225 Kab. Bandung, Jawa Barat, Indonesia

Supplier/ Vendor	Autorized Signature	Approved
	Purchasing	Finance

Note :

Lampiran C Contoh Invoice

 MARLIP PT. MARLIP INDO MANDIRI		Invoice																																																			
NPWP : 0412 8783 8142 8000 MARLIP INDO MANDIRI				No :																																																	
Pembeli/ Buyer :		Surat Pesanan/PO/SPK : No : - Tgl/ date :																																																			
Dikirim/ Ship to :																																																					
<table border="1"> <thead> <tr> <th>No</th> <th>Type</th> <th>Quantity</th> <th>Price</th> <th>Subtotal</th> </tr> </thead> <tbody> <tr><td>1</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr><td>2</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr><td>3</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr><td>4</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr><td>5</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr><td>6</td><td></td><td>-</td><td>-</td><td>-</td></tr> <tr> <td colspan="4">TOTAL</td> <td>-</td> </tr> <tr> <td colspan="4"></td> <td>PPN 11%</td> </tr> <tr> <td colspan="4"></td> <td>Jumlah</td> </tr> </tbody> </table>				No	Type	Quantity	Price	Subtotal	1		-	-	-	2		-	-	-	3		-	-	-	4		-	-	-	5		-	-	-	6		-	-	-	TOTAL				-					PPN 11%					Jumlah
No	Type	Quantity	Price	Subtotal																																																	
1		-	-	-																																																	
2		-	-	-																																																	
3		-	-	-																																																	
4		-	-	-																																																	
5		-	-	-																																																	
6		-	-	-																																																	
TOTAL				-																																																	
				PPN 11%																																																	
				Jumlah																																																	
Terbilang																																																					
Nol Rupiah																																																					
Jatuh Tempo 00-Jan-00 Payment due date		Bandung, 00-Jan-00 PT. MARLIP INDO MANDIRI																																																			
TUNAI atau Transfer Ke : Bank : BRI Cab Soekarno Hatta Bdg Ac No. : 040701001433309 Nama : Marlip Indo Mandiri		<table border="1"> <tr> <td>Konfirmasi Buyer/ Custm.</td> </tr> <tr> <td>()</td> </tr> </table> <u>Didin Juarsa</u> FaA		Konfirmasi Buyer/ Custm.	()																																																
Konfirmasi Buyer/ Custm.																																																					
()																																																					

Lampiran D Implementasi Model User dan Produk

```
1 model User {
2   id      String    @id @default(cuid())
3   nama     String
4   email    String    @unique
5   passwordHash String @map("password_hash")
6   role     Role      @default(konsumen)
7   whatsapp String?
8   instansi String?
9   alamat   String?
10  createdAt DateTime @default(now()) @map("created_at")
11  updatedAt DateTime @updatedAt @map("updated_at")
12
13  purchaseOrders PurchaseOrder[]
14  tahapanStarted TahapanProduksi[] @relation("tahapanStartedBy")
15  tahapanCompleted TahapanProduksi[] @relation("tahapanCompletedBy")
16
17  @@map("users")
18 }
19
20 model Produk {
21   id      String    @id @default(cuid())
22   nama     String
23   slug     String    @unique
24   kategori Kategori
25   deskripsi String    @db.Text
26   spesifikasi Json     @default("{}")
27   hargaDasar Decimal    @db.Decimal(18, 0) @map("harga_dasar")
28   fotoUrls String[]    @map("foto_urls")
29   isActive Boolean      @default(true) @map("is_active")
30   createdAt DateTime    @default(now()) @map("created_at")
31   updatedAt DateTime    @updatedAt @map("updated_at")
32
33   detailPesanan DetailPesanan[]
34
35   @@map("produk")
36 }
```

Lampiran E Implementasi Model Transaksi

```

1 model PurchaseOrder {
2   id          String          @id @default(cuid())
3   noPO        String          @unique @map("no_po")
4   userId      String          @map("user_id")
5   tglPesanan  DateTime        @default(now()) @map("tgl_pesanan")
6   totalHarga  Decimal         @db.Decimal(18, 0) @map("total_harga")
7   progressProduksi Int        @default(0) @map("progress_produksi") // 0-100 (Legacy)
8   statusPO    StatusPO       @default(menunggu_review) @map("status_po")
9   catatanAdmin String?       @map("catatan_admin") @db.Text
10  catatanKustomisasi String?  @map("catatan_kustomisasi") @db.Text
11  warnaUtama   String?       @map("warna_utama")
12  varianPesanan String?       @map("varian_pesanan")
13  hargaDasar   Decimal?       @db.Decimal(18, 0) @map("harga_dasar")
14  ppn          Decimal?       @db.Decimal(18, 0)
15  hargaSetelahPpn Decimal?    @db.Decimal(18, 0) @map("harga_setelah_ppn")
16  hargaFinal   Decimal?       @db.Decimal(18, 0) @map("harga_final")
17  estimasiHari Int?           @map("estimasi_hari")
18  catatanPenawaran String?    @map("catatan_penawaran") @db.Text
19  fotoProduk   String[]       @map("foto_produk") // Legacy
20  signatureUrl  String?       @map("signature_url") @db.Text
21  adminSignatureUrl String?   @map("admin_signature_url") @db.Text
22  createdAt    DateTime       @default(now()) @map("created_at")
23  updatedAt    DateTime       @updatedAt @map("updated_at")
24
25  user          User           @relation(fields: [userId], references: [id])
26  detailPesanan DetailPesanan[]
27  invoices      Invoice[]
28  tahapanProduksi TahapanProduksi[]
29
30  @@map("purchase_orders")
31 }
32
33 model DetailPesanan {
34   id          String @id @default(cuid())
35   poId        String @map("po_id")
36   produkId    String @map("produk_id")
37   qty         Int     @default(1)
38   kustomisasi Json    @default("{}")
39   hargaSatuan Decimal @db.Decimal(18, 0) @map("harga_satuan")
40   subtotal    Decimal @db.Decimal(18, 0)
41
42   purchaseOrder PurchaseOrder @relation(fields: [poId], references: [id], onDelete: Cascade)
43   produk         Produk       @relation(fields: [produkId], references: [id])
44
45   @@map("detail_pesanan")
46 }
47
48 model Invoice {
49   id          String @id @default(cuid())
50   poId        String @map("po_id")
51   termin     Termin
52   totalTagihan Decimal @db.Decimal(18, 0) @map("total_tagihan")
53   statusPembayaran StatusPembayaran @default(pending) @map("status_pembayaran")
54   midtransOrderId String? @unique @map("midtrans_order_id")
55   snapToken    String?    @map("snap_token")
56   paidAt       DateTime?   @map("paid_at")
57   createdAt    DateTime    @default(now()) @map("created_at")
58   updatedAt    DateTime    @updatedAt @map("updated_at")
59
60   purchaseOrder PurchaseOrder @relation(fields: [poId], references: [id], onDelete: Cascade)
61
62   @@unique([poId, termin])
63   @@map("invoices")
64 }
65
66 model TahapanProduksi {
67   id          String @id @default(cuid())
68   poId        String @map("po_id")
69   kodeTahapan String @map("kode_tahapan")
70   namaTahapan String @map("nama_tahapan")
71   urutan      Int
72   status       StatusTahapan @default(belum_dimulai)
73   startedAt    DateTime?     @map("started_at")
74   startedBy    String?       @map("started_by")
75   completedAt  DateTime?     @map("completed_at")
76   completedBy  String?       @map("completed_by")
77   catatan      String?       @db.Text
78   fotoUrl      String?       @map("foto_url")
79   createdAt    DateTime       @default(now()) @map("created_at")
80   updatedAt    DateTime       @updatedAt @map("updated_at")
81
82   purchaseOrder PurchaseOrder @relation(fields: [poId], references: [id], onDelete: Cascade)
83   startedByUser  User?         @relation("tahapanStartedBy", fields: [startedBy], references: [id])
84   completedByUser User?        @relation("tahapanCompletedBy", fields: [completedBy], references: [id])
85
86   @@unique([poId, kodeTahapan])
87   @@map("tahapan_produksi")
88 }

```

Lampiran F Implementasi Validasi Pengajuan Pesanan

```
1 // POST /api/orders - buat pesanan baru dari form pemesanan
2 export async function POST(request: NextRequest) {
3   try {
4     const session = await auth()
5     if (!session?.user?.id) {
6       return NextResponse.json({ error: "Tidak terautentikasi" }, { status: 401 })
7     }
8
9     if (session.user.role !== "konsumen") {
10      return NextResponse.json(
11        { error: "Akses ditolak: Hanya konsumen yang dapat membuat pesanan" },
12        { status: 403 }
13      )
14    }
15
16    const resolvedUserId = session.user.id
17
18    const body = await request.json()
19    const { produkId, qty = 1, warnaUtama, varianPesanan, catatanKustomisasi } = body
20
21    if (!produkId) {
22      return NextResponse.json({ error: "produkId wajib diisi" }, { status: 400 })
23    }
24
25    // Validasi warnaUtama
26    if (warnaUtama !== "Hitam" && warnaUtama !== "Putih") {
27      return NextResponse.json({ error: "Pilihan warna utama hanya Hitam atau Putih" }, { status: 400 })
28    }
29
30    // Validasi varianPesanan
31    if (varianPesanan !== "Standar" && varianPesanan !== "Kustom") {
32      return NextResponse.json({ error: "Pilihan varian pesanan hanya Standar atau Kustom" }, { status: 400 })
33    }
34
35    // Validasi catatanKustomisasi untuk varian Kustom
36    if (varianPesanan === "Kustom" && (!catatanKustomisasi || catatanKustomisasi.trim() === "")) {
37      return NextResponse.json({ error: "Catatan kustomisasi wajib diisi untuk varian Kustom" }, { status: 400 })
38    }
39
40    // Validasi qty integer 1-20
41    const qtyNum = Number(qty)
42    if (!Number.isInteger(qtyNum) || qtyNum < 1 || qtyNum > 20) {
43      return NextResponse.json({ error: "Jumlah (qty) harus berupa angka bulat antara 1 sampai 20" }, { status: 400 })
44    }
45  }
46 }
```

Lampiran G Implementasi Pembuatan Purchase Order

```

1 // Ambil data produk untuk kalkulasi harga dan validasi produk aktif
2 const produk = await prisma.produk.findUnique({ where: { id: produkId } })
3 if (!produk || !produk.isActive) {
4   return NextResponse.json({ error: "Produk tidak ditemukan atau tidak aktif" }, { status: 400 })
5 }
6
7 // Generate nomor PO unik: PO-2026-XXXX
8 const count = await prisma.purchaseOrder.count()
9 const noPO = `PO-${new Date().getFullYear()}-${String(count + 1).padStart(4, "0")}`
10
11 // Kalkulasi harga:
12 // PPN = harga dasar * 11%
13 // Harga setelah PPN = harga dasar + PPN
14 // Total = harga setelah PPN * qty
15 const hargaDasarDec = new Prisma.Decimal(produk.hargaDasar.toString())
16 const ppnDec = hargaDasarDec.mul(0.11).round()
17 const hargaSetelahPpnDec = hargaDasarDec.add(ppnDec)
18 const totalHargaDec = hargaSetelahPpnDec.mul(qtyNum)
19 const subtotalDec = hargaDasarDec.mul(qtyNum)
20
21 const estimasiHari = varianPesanan === "Standar" ? 40 : null
22
23 const po = await prisma.purchaseOrder.create({
24   data: {
25     noPO,
26     userId: resolvedUserId,
27     totalHarga: totalHargaDec,
28     statusPO: "menunggu_review",
29     warnaUtama,
30     varianPesanan,
31     catatanKustomisasi: varianPesanan === "Kustom" ? catatanKustomisasi : null,
32     hargaDasar: hargaDasarDec,
33     ppn: ppnDec,
34     hargaSetelahPpn: hargaSetelahPpnDec,
35     estimasiHari,
36     detailPesanan: {
37       create: {
38         produkId,
39         qty: qtyNum,
40         kustomisasi: {
41           warnaUtama,
42           varianPesanan,
43           catatanKustomisasi: varianPesanan === "Kustom" ? catatanKustomisasi : null,
44         },
45         hargaSatuan: hargaDasarDec,
46         subtotal: subtotalDec,
47       },
48     },
49   },
50   include: {
51     detailPesanan: { include: { produk: true } },
52     user: { select: { nama: true, email: true } },
53   },
54 })
55
56 // Kirim email notifikasi pesanan diterima secara async
57 if (po.user) {
58   sendOrderReceivedEmail(po.user.email, po.user.nama, po.noPO, po.totalHarga.toString()).catch((err) => {
59     console.error("[Order Received Email Error]", err)
60   })
61 }
62
63 return NextResponse.json({ data: po }, { status: 201 })
64 } catch (error: unknown) {
65   console.error("[POST /api/orders]", error)
66   return NextResponse.json({ error: "Gagal membuat pesanan" }, { status: 500 })
67 }
68 }

```

Lampiran H Implementasi Validasi Webhook Midtrans

```

1 import { NextRequest, NextResponse } from "next/server"
2 import { prisma } from "@lib/prisma"
3 import crypto from "crypto"
4 import { Prisma } from "@prisma/client"
5 import { sendDPPaidEmail, sendOrderPaidAndReadyEmail } from "@lib/mail"
6
7 export async function POST(request: NextRequest) {
8   try {
9     const body = await request.json()
10    const {
11      order_id,
12      status_code,
13      gross_amount,
14      signature_key,
15      transaction_status,
16      fraud_status
17    } = body
18
19    if (!order_id || !status_code || !gross_amount || !signature_key || !transaction_status) {
20      return NextResponse.json({ error: "Payload webhook tidak lengkap" }, { status: 400 })
21    }
22
23    // 1. Verifikasi Signature Key dari Midtrans
24    const serverKey = process.env.MIDTRANS_SERVER_KEY
25    if (!serverKey || serverKey.includes("xxxxxx") || serverKey === "") {
26      return NextResponse.json({ error: "Midtrans Server Key belum dikonfigurasi di server" }, { status: 500 })
27    }
28
29    const combinedString = `${order_id}${status_code}${gross_amount}${serverKey}`
30    const calculatedSignature = crypto.createHash("sha512").update(combinedString).digest("hex")
31
32    if (calculatedSignature !== signature_key) {
33      console.warn(`[Webhook Warning] Invalid signature for order ${order_id}`)
34      return NextResponse.json({ error: "Signature tidak valid" }, { status: 400 })
35    }
36
37    // 2. Cari Invoice berdasarkan midtransOrderId
38    const invoice = await prisma.invoice.findFirst({
39      where: { midtransOrderId: order_id },
40      include: {
41        purchaseOrder: true
42      }
43    })
44
45    if (!invoice) {
46      console.warn(`[Webhook Warning] Invoice dengan Midtrans Order ID ${order_id} tidak ditemukan`)
47      return NextResponse.json({ error: "Invoice tidak ditemukan" }, { status: 404 })
48    }
49
50    // Validasi gross_amount menggunakan Prisma.Decimal
51    const payloadGross = new Prisma.Decimal(gross_amount.toString())
52    if (!payloadGross.equals(invoice.totalTagihan)) {
53      return NextResponse.json({ error: "Gross amount tidak cocok" }, { status: 400 })
54    }

```

Lampiran I Implementasi Pembaruan Status Pembayaran dan Pesanan

```

1 if (statusPembayaran === "paid") {
2   if (txInvoice.termin === "dp") {
3     if (order.statusPO !== "menunggu_dp") {
4       throw new Error("TransitionInvalid: DP hanya boleh mengubah PO ke dp_diterima dari status
menunggu_dp")
5     }
6     nextStatus = "dp_diterima"
7
8     // Inisialisasi 4 tahapan produksi secara idempoten
9     const initialStages = [
10      { poId: order.id, kodeTahapan: "PERSIAPAN", namaTahapan: "Persiapan Produksi", urutan: 1 },
11      { poId: order.id, kodeTahapan: "PERAKITAN", namaTahapan: "Perakitan Kendaraan", urutan: 2 },
12      { poId: order.id, kodeTahapan: "PEMERIKSAAN", namaTahapan: "Pemeriksaan Akhir", urutan: 3 },
13      { poId: order.id, kodeTahapan: "SELESAI", namaTahapan: "Produksi Selesai", urutan: 4 },
14    ]
15
16    await tx.tahapanProduksi.createMany({
17      data: initialStages,
18      skipDuplicates: true,
19    })
20  } else if (txInvoice.termin === "lunas") {
21    const dpInvoice = await tx.invoice.findFirst({
22      where: { poId: order.id, termin: "dp" }
23    })
24    const dpPaid = dpInvoice?.statusPembayaran === "paid"
25
26    if (!dpPaid) {
27      throw new Error("TransitionInvalid: Pelunasan hanya boleh dilakukan jika DP sudah lunas")
28    }
29
30    const stages = await tx.tahapanProduksi.findMany({
31      where: { poId: order.id }
32    })
33    const allDone = stages.length >= 4 && stages.every(s => s.status === "selesai")
34
35    if (!allDone) {
36      throw new Error("TransitionInvalid: Pelunasan hanya boleh dilakukan jika seluruh tahapan
produksi telah selesai")
37    }
38    nextStatus = "siap_dikirim"
39  }
40 }
41
42 // Lakukan pembaruan invoice
43 await tx.invoice.update({
44   where: { id: txInvoice.id },
45   data: {
46     statusPembayaran,
47     paidAt: (statusPembayaran === "paid") ? new Date() : txInvoice.paidAt
48   }
49 })
50
51 // Lakukan pembaruan PO jika status pembayarannya paid
52 if (statusPembayaran === "paid") {
53   await tx.purchaseOrder.update({
54     where: { id: order.id },
55     data: {
56       statusPO: nextStatus,
57     }
58   })
59 }

```