

**PENGEMBANGAN WEBSITE RENTAL MOBIL
BERBASIS NEXT.JS
(STUDI KASUS: RAHAYU TRANS TRAVEL BANDUNG)**

TUGAS AKHIR

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

oleh:

Miftah Fauzi
nrp.22.304.0039



**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 : Miftah Fauzi

NRP: 22.304.0039

Dengan judul:

“Pengembangan Website Rental Mobil Berbasis Next.js (Studi Kasus: Rahayu Trans Travel
Bandung)”

Bandung, 23 Juli 2026

Menyetujui,

Pembimbing utama,

(Fajar Darmawan, S.T., 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 plagiasi 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, 23 Juli 2026
Yang membuat pernyataan,
Meterai

(Miftah Fauzi)
NRP. 22.304.0039

ABSTRAK

Rahayu Trans Travel Bandung masih mengelola informasi armada, jadwal penyewaan, dan transaksi melalui *website* statis, WhatsApp dengan format bebas, kalender, serta Microsoft Excel yang terpisah. Kondisi ini menyebabkan data pelanggan kurang lengkap, komunikasi berulang, dan risiko pesanan ganda. Tugas akhir ini bertujuan mengembangkan *website* rental mobil berbasis Next.js yang berfungsi sebagai sistem informasi reservasi kendaraan untuk menyajikan katalog dinamis, menstandarkan data reservasi, dan memusatkan pengelolaan operasional.

Pengembangan menggunakan metode *Waterfall* melalui tahap komunikasi, perencanaan, pemodelan, konstruksi, dan *deployment*. Sistem diimplementasikan dengan Next.js, TypeScript, PostgreSQL, dan Prisma ORM, serta diterapkan pada Vercel dan Neon PostgreSQL. *Booking* disimpan berstatus *pending* dengan batas *expiresAt* 24 jam. *Booking* yang kedaluwarsa dibatalkan saat data *booking* atau ketersediaan diakses, sedangkan Vercel Cron harian digunakan sebagai pemeriksaan cadangan. Pengujian dilakukan menggunakan *Black Box Testing*.

Sistem menyediakan pencarian dan penyaringan armada, formulir reservasi, pemeriksaan ketersediaan tanggal, validasi minimal hari berdasarkan zona, estimasi biaya berdasarkan tarif zona dengan harga dasar sebagai cadangan, WhatsApp *Click to Chat*, serta *dashboard* admin. *Dashboard* mendukung pengelolaan armada, zona, harga, *booking*, pembayaran, testimoni, notifikasi, ketersediaan, dan ekspor Excel. Pengujian terhadap 50 *test case* menunjukkan seluruh skenario berhasil. Dengan demikian, sistem mendukung reservasi yang lebih terstruktur dan pengelolaan data yang lebih terpusat.

Kata kunci: sistem informasi reservasi, rental mobil, Next.js, WhatsApp *Click to Chat*, PostgreSQL, *Black Box Testing*.

ABSTRACT

Rahayu Trans Travel Bandung managed fleet information, rental schedules, and transactions using a static website, free-form WhatsApp messages, a calendar, and Excel files. This caused incomplete customer data, repetitive communication, and double-booking risks. This project develops a Next.js car rental website functioning as a reservation information system to provide a dynamic fleet catalog, standardize reservation data, and centralize operations.

The system was developed using Waterfall through communication, planning, modeling, construction, and deployment. It uses Next.js, TypeScript, PostgreSQL, and Prisma ORM, and is deployed on Vercel with Neon PostgreSQL. Each booking receives a pending status and a 24-hour expiresAt deadline. Expired bookings are canceled when booking or availability data is accessed, while a daily Vercel Cron job provides a fallback check. Functional testing applied Black Box Testing.

The system provides fleet search and filtering, a reservation form, date availability checking, minimum rental-day validation by zone, zone-based cost estimation with base rates as a fallback, WhatsApp Click to Chat, and an admin dashboard. The dashboard manages fleets, zones, rates, bookings, payments, testimonials, notifications, availability, and Excel exports. Testing of 50 test cases showed that all scenarios passed. Therefore, the system supports a more structured reservation process and centralized data management.

Keywords: *reservation information system, car rental, Next.js, WhatsApp Click to Chat, PostgreSQL, 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 “Pengembangan *Website* Rental Mobil Berbasis Next.js (Studi Kasus: Rahayu Trans Travel Bandung)”.

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 dan pengelolaan armada rental mobil agar proses reservasi menjadi lebih efisien, terstandarisasi, 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. Kedua orang tua, kakak, serta seluruh keluarga tercinta yang senantiasa memberikan doa, kasih sayang, serta dukungan moral maupun materiil yang tak terhingga.
3. Bapak Fajar Darmawan, 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 dan seluruh Dosen Program Studi Teknik Informatika Universitas Pasundan yang telah memberikan bekal ilmu pengetahuan selama masa perkuliahan.
5. Pihak Rahayu Trans Travel yang telah memberikan izin penelitian serta membantu dalam penyediaan data yang diperlukan.
6. Rekan-rekan seperjuangan angkatan 2022 Program Studi Teknik Informatika Universitas Pasundan yang saling memberikan semangat dan motivasi.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan dikarenakan 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 jasa transportasi atau penyewaan kendaraan.

Bandung, 23 Juli 2026

Penulis

(Miftah Fauzi)

DAFTAR ISI

ABSTRAK	i
ABSTRACT.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	iv
DAFTAR ISTILAH.....	viii
DAFTAR TABEL	ix
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN	xiv
DAFTAR SIMBOL.....	xv
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	1-2
1.5 Metodologi Tugas Akhir	1-3
1.6 Standar dan Kakas.....	1-4
1.7 Sistematika Penulisan	1-5
BAB 2 LANDASAN TEORI.....	2-1
2.1 Teori yang Digunakan.....	2-1
2.1.1 Metode <i>Waterfall</i>	2-1
2.1.2 Sistem Informasi Berbasis Web.....	2-2
2.1.3 <i>User Experience</i> (UX) dan <i>User Interface</i> (UI)	2-2
2.1.4 <i>Framework</i> Next.js	2-3
2.1.5 React	2-3
2.1.6 TypeScript.....	2-4
2.1.7 PostgreSQL.....	2-4

2.1.8	Prisma ORM	2-5
2.1.9	NextAuth.js (Autentikasi Berbasis <i>Token</i>)	2-5
2.1.10	Zod dan React Hook Form (Validasi Data).....	2-6
2.1.11	Recharts (Visualisasi Data)	2-6
2.1.12	Motion (Animasi Antarmuka)	2-6
2.1.13	SheetJS (xlsx)	2-6
2.1.14	Tailwind CSS.....	2-7
2.1.15	Integrasi WhatsApp <i>Click to Chat</i>	2-7
2.1.16	Pengujian Black-Box (<i>Black-Box Testing</i>).....	2-8
2.1.17	Unified <i>Modeling</i> Language (UML)	2-8
2.1.18	<i>Entity Relationship Diagram</i> (ERD)	2-9
2.1.19	Vercel	2-9
2.2	Penelitian Terdahulu	2-9
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
3.4	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.....	3-9
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	4-4
4.1.3	Analisis Kebutuhan Pengguna (<i>User Requirement</i>)	4-6
4.1.4	Analisis Kebutuhan Perangkat Lunak (<i>Software Requirement</i>)	4-6
4.1.5	Analisis Kebutuhan Non-Fungsional	4-8
4.2	Perancangan Sistem	4-8
4.2.1	<i>Use Case Diagram</i>	4-8
4.2.2	Skenario <i>Use Case</i>	4-11
4.2.3	<i>Activity Diagram</i>	4-20
4.2.4	<i>Sequence Diagram</i>	4-26
4.2.5	Perancangan Basis Data	4-43
4.2.6	Perancangan Antarmuka	4-47
BAB 5	IMPLEMENTASI DAN PENGUJIAN	5-1
5.1	Implementasi Sistem	5-1
5.1.1	Lingkungan Implementasi	5-1
5.1.2	Struktur <i>Folder</i> dan Berkas	5-2
5.1.3	Konfigurasi Aplikasi	5-2
5.1.4	Implementasi Basis Data	5-3
5.1.5	Implementasi <i>Route Handler</i> dan API	5-3
5.2	Implementasi Antarmuka Berdasarkan <i>Role</i>	5-4
5.2.1	Antarmuka Pelanggan	5-4
5.2.2	Antarmuka Admin	5-10
5.3	Implementasi Fitur Utama	5-17
5.3.1	Implementasi Katalog dan Detail Armada	5-17
5.3.2	Implementasi Formulir Reservasi dan Ringkasan Reservasi	5-18
5.3.3	Implementasi Validasi Ketersediaan Armada	5-18
5.3.4	Implementasi Perhitungan Harga Berdasarkan Zona	5-19
5.3.5	Implementasi Integrasi WhatsApp <i>Click to Chat</i>	5-19

5.3.6	Implementasi Pembatalan Otomatis <i>Booking Pending</i>	5-20
5.3.7	Implementasi Cek Status Pesanan	5-21
5.3.8	Implementasi Manajemen <i>Booking</i> dan Ekspor Excel	5-22
5.3.9	Implementasi <i>Monitoring</i> Ketersediaan Armada.....	5-22
5.3.10	Implementasi Testimoni, Notifikasi, dan Pengaturan	5-23
5.4	Pengujian Fungsional <i>Black Box</i>	5-24
5.4.1	Pengujian Katalog dan Reservasi	5-24
5.4.2	Pengujian Formulir dan Ringkasan Reservasi	5-25
5.4.3	Pengujian Validasi Ketersediaan Armada	5-25
5.4.4	Pengujian Perhitungan Harga Berdasarkan Zona	5-26
5.4.5	Pengujian Integrasi WhatsApp <i>Click to Chat</i>	5-26
5.4.6	Pengujian Pembatalan Otomatis <i>Booking Pending</i>	5-27
5.4.7	Pengujian Cek Status Pesanan	5-27
5.4.8	Pengujian Manajemen <i>Booking</i> dan Ekspor Excel	5-28
5.4.9	Pengujian <i>Monitoring</i> Ketersediaan	5-28
5.4.10	Pengujian Testimoni, Notifikasi, dan Pengaturan	5-29
5.4.11	Rekapitulasi Hasil Pengujian <i>Black Box</i>	5-30
5.5	<i>Deployment</i>	5-31
BAB 6	KESIMPULAN DAN SARAN.....	6-1
6.1	Kesimpulan	6-1
6.2	Saran.....	6-2
	DAFTAR PUSTAKA.....	xv
	LAMPIRAN.....	A

DAFTAR ISTILAH

No	Istilah	Penjelasan / Keterangan
1	Armada	Unit kendaraan yang disediakan Rahayu Trans Travel untuk disewakan kepada pelanggan.
2	Booking	Data transaksi reservasi yang mencatat pelanggan, armada, tanggal sewa, zona tujuan, dan status pesanan.
3	Double booking	Keadaan ketika satu armada menerima lebih dari satu booking pada rentang tanggal yang bertabrakan.
4	Pending	Status awal booking yang telah tersimpan tetapi belum dikonfirmasi oleh admin.
5	expiresAt	Penanda waktu kedaluwarsa booking, yaitu 24 jam setelah booking dibuat. Booking yang tetap pending setelah waktu ini dapat dibatalkan.
6	HargaZona	Tarif sewa per hari berdasarkan kombinasi armada dan zona tujuan. Harga dasar armada digunakan apabila tarif zona belum tersedia.
7	WhatsApp Click to Chat	Mekanisme yang membuka percakapan WhatsApp admin beserta draf pesan reservasi tanpa pelanggan harus menyimpan nomor terlebih dahulu.
8	Vercel Cron	Layanan penjadwalan yang memanggil endpoint pemeriksaan booking kedaluwarsa satu kali setiap hari sebagai proses cadangan.

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	2-9
Tabel 3.1 Alur Penyelesaian Tugas Akhir	3-1
Tabel 3.2 Deskripsi Skema Analisis	3-7
Tabel 4.1 Analisis Kebutuhan Pengguna	4-6
Tabel 4.2 Analisis Kebutuhan Perangkat Lunak	4-6
Tabel 4.3 Analisis Kebutuhan Non-Fungsional	4-8
Tabel 4.4 Deskripsi Aktor pada <i>Use Case Diagram</i>	4-10
Tabel 4.5 Deskripsi <i>Use Case Diagram</i>	4-10
Tabel 4.6 Skenario UC-01: Melihat Beranda	4-11
Tabel 4.7 Skenario UC-02: Melihat Katalog Armada	4-11
Tabel 4.8 Skenario UC-03: Melihat Detail Armada	4-12
Tabel 4.9 Skenario UC-04: Melihat Harga dan Galeri	4-12
Tabel 4.10 Skenario UC-05: Mengisi Formulir Reservasi	4-13
Tabel 4.11 Skenario UC-06: Mengirim Pesanan via WhatsApp	4-13
Tabel 4.12 Skenario UC-07: Mengecek Status Pesanan	4-14
Tabel 4.13 Skenario UC-08: Memberikan Testimoni	4-14
Tabel 4.14 Skenario UC-09: <i>Login</i> Admin	4-15
Tabel 4.15 Skenario UC-10: Melihat <i>Dashboard</i>	4-15
Tabel 4.16 Skenario UC-11: Mengelola Armada	4-16
Tabel 4.17 Skenario UC-12: Mengelola Zona Wilayah	4-16
Tabel 4.18 Skenario UC-13: Mengelola Harga Zona	4-17
Tabel 4.19 Skenario UC-14: Mengelola Pesanan	4-17
Tabel 4.20 Skenario UC-15: Mengelola Testimoni	4-18
Tabel 4.21 Skenario UC-16: Mengelola Pengaturan Situs	4-18
Tabel 4.22 Skenario UC-17: Melihat Notifikasi	4-19
Tabel 4.23 Skenario UC-18: Mengekspor Data Excel	4-19
Tabel 5.1 Kebutuhan Perangkat Keras	5-1
Tabel 5.2 Kebutuhan Perangkat Lunak	5-1
Tabel 5.3 Struktur <i>Folder</i> Aplikasi	5-2
Tabel 5.4 Konfigurasi <i>Environment Variable</i>	5-2
Tabel 5.5 Implementasi Entitas Basis Data	5-3
Tabel 5.6 Implementasi <i>Route Handler</i> dan API	5-3
Tabel 5.7 Pengujian Katalog dan Reservasi	5-24

Tabel 5.8 Pengujian Formulir dan Ringkasan Reservasi.....	5-25
Tabel 5.9 Pengujian Validasi Ketersediaan Armada	5-25
Tabel 5.10 Pengujian Perhitungan Harga Berdasarkan Zona	5-26
Tabel 5.11 Pengujian Integrasi WhatsApp <i>Click to Chat</i>	5-26
Tabel 5.12 Pengujian Pembatalan Otomatis <i>Booking Pending</i>	5-27
Tabel 5.13 Pengujian Cek Status Pesanan.....	5-27
Tabel 5.14 Pengujian Manajemen <i>Booking</i> dan Ekspor Excel.....	5-28
Tabel 5.15 Pengujian <i>Monitoring</i> Ketersediaan	5-28
Tabel 5.16 Pengujian Testimoni, Notifikasi, dan Pengaturan	5-29
Tabel 5.17 Rekapitulasi Hasil Pengujian <i>Black Box</i>	5-30

DAFTAR GAMBAR

Gambar 1.1 Metodologi Tugas Akhir	1-3
Gambar 2.1 Metode <i>Waterfall</i>	2-1
Gambar 2.2 Logo <i>Framework</i> Next.js.....	2-3
Gambar 2.3 Logo React.....	2-3
Gambar 2.4 TypeScript	2-4
Gambar 2.5 PostgreSQL.....	2-4
Gambar 2.6 Logo Prisma ORM.....	2-5
Gambar 2.7 Logo Tailwind CSS	2-7
Gambar 2.8 Ilustrasi Pengujian Black-Box	2-8
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 Rahayu Trans Travel Bandung.....	3-8
Gambar 4.1 Flowmap Alur Bisnis yang Berjalan	4-3
Gambar 4.2 Flowmap Alur Bisnis Usulan	4-5
Gambar 4.3 <i>Use Case Diagram</i>	4-9
Gambar 4.4 <i>Activity Diagram</i> A.....	4-21
Gambar 4.5 <i>Activity Diagram</i> B.....	4-23
Gambar 4.6 <i>Activity Diagram</i> C.....	4-25
Gambar 4.7 Diagram Sequence UC-01 - Melihat Beranda	4-26
Gambar 4.8 Diagram Sequence UC-02 - Melihat Katalog Armada.....	4-27
Gambar 4.9 Diagram Sequence UC-03 - Melihat Detail Armada.....	4-28
Gambar 4.10 Diagram Sequence UC-04 - Melihat Harga dan Galeri.....	4-29
Gambar 4.11 Diagram Sequence UC-05 - Mengisi Formulir Reservasi	4-30
Gambar 4.12 Diagram Sequence UC-06 - Mengirim Pesanan via WhatsApp.....	4-31
Gambar 4.13 Diagram Sequence UC-07 - Mengecek Status Pesanan	4-32
Gambar 4.14 Diagram Sequence UC-08 - Memberikan Testimoni	4-33
Gambar 4.15 Diagram Sequence UC-09 - <i>Login</i> Admin	4-34
Gambar 4.16 Diagram Sequence UC-10 - Melihat <i>Dashboard</i>	4-35
Gambar 4.17 Diagram Sequence UC-11 - Mengelola Armada.....	4-36
Gambar 4.18 Diagram Sequence UC-12 - Mengelola Zona Wilayah	4-37
Gambar 4.19 Diagram Sequence UC-13 - Mengelola Harga Zona.....	4-38
Gambar 4.20 Diagram Sequence UC-14 - Mengelola Pesanan	4-39

Gambar 4.21 Diagram Sequence UC-15 - Mengelola Testimoni	4-40
Gambar 4.22 Diagram Sequence UC-16 - Mengelola Pengaturan Situs.....	4-41
Gambar 4.23 Diagram Sequence UC-17 - Melihat Notifikasi	4-42
Gambar 4.24 Diagram Sequence UC-18 - Mengekspor Data Excel.....	4-43
Gambar 4.25 <i>Entity Relationship Diagram</i>	4-44
Gambar 4.26 Halaman Utama Beranda	4-47
Gambar 4.27 Halaman Katalog Armada	4-48
Gambar 4.28 Halaman Detail Armada	4-49
Gambar 4.29 Halaman Formulir Reservasi	4-50
Gambar 4.30 Halaman Formulir Ringkasan dan Konfirmasi Reservasi	4-51
Gambar 4.31 Halaman Pelacakan Status Reservasi dan Pemberian Testimoni	4-52
Gambar 4.32 Halaman <i>Login</i> Admin	4-53
Gambar 4.33 Halaman <i>Dashboard</i> Utama Admin	4-54
Gambar 4.34 Halaman Manajemen Armada	4-55
Gambar 4.35 Halaman Ketersediaan Armada	4-56
Gambar 4.36 Halaman Manajemen Zona Wilayah	4-57
Gambar 4.37 Halaman Manajemen <i>Booking</i>	4-58
Gambar 4.38 Halaman Manajemen Testimoni.....	4-59
Gambar 4.39 Halaman Pengaturan Situs	4-60
Gambar 5.1 Struktur <i>Folder</i> Aplikasi.....	5-2
Gambar 5.2 Halaman Beranda	5-5
Gambar 5.3 Halaman Katalog Armada	5-6
Gambar 5.4 Halaman Detail Armada	5-7
Gambar 5.5 Halaman Formulir Reservasi	5-8
Gambar 5.6 Halaman Ringkasan dan Pratinjau Pesan Reservasi	5-9
Gambar 5.7 Halaman Cek <i>Booking</i>	5-10
Gambar 5.8 Halaman <i>Login</i> Admin	5-11
Gambar 5.9 <i>Dashboard</i> Admin	5-12
Gambar 5.10 Halaman Manajemen Armada	5-12
Gambar 5.11 Halaman Pengaturan Harga Zona	5-13
Gambar 5.12 Halaman Manajemen Zona Wilayah	5-14
Gambar 5.13 Halaman Manajemen <i>Booking</i>	5-14
Gambar 5.14 Halaman <i>Monitoring</i> Ketersediaan Armada	5-15
Gambar 5.15 Halaman Manajemen Testimoni.....	5-16
Gambar 5.16 Panel Notifikasi Admin	5-16

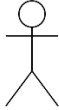
Gambar 5.17 Halaman Pengaturan Situs	5-17
--	------

DAFTAR LAMPIRAN

Lampiran A Berita acara	A-2
Lampiran B Berita acara.....	A-3
Lampiran C Dokumentasi Wawancara.....	4-4

DAFTAR SIMBOL

Use Case Diagram Simbol

No	Simbol	Nama	Deskripsi
1		Actor	Mewakili entitas eksternal (pengguna/konsumen atau sistem admin) yang berinteraksi langsung dengan sistem.
2		Use Case	Merepresentasikan fungsionalitas atau proses tertentu yang disediakan oleh sistem aplikasi kepada <i>actor</i> .
3		Association	Jalur komunikasi yang menghubungkan antara <i>actor</i> dengan <i>use case</i> yang diaksesnya.

Activity Diagram Simbol

No	Simbol	Nama	Deskripsi
1		Status Awal	Menandakan titik awal dimulainya suatu aliran pekerjaan atau aktivitas di dalam antarmuka sistem.
2		Status 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		Aliran Kontrol	Menunjukkan arah urutan eksekusi dari satu tahapan aktivitas ke aktivitas selanjutnya.

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

Rahayu Trans Travel Bandung, yang telah beroperasi sejak tahun 2020 dengan mengelola sekitar 24 armada kendaraan, saat ini masih menjalankan operasional penyewaan secara semi-manual. Meskipun telah memiliki *website* berbasis Google Sites, *website* tersebut bersifat statis dan belum menampilkan informasi krusial secara detail, seperti harga sewa dan ketersediaan kendaraan berdasarkan data *booking* terkini. Akibatnya, pelanggan tidak dapat memperoleh informasi secara mandiri dan tidak tersedia *dashboard* untuk pengelolaan data oleh admin.

Berdasarkan hasil observasi dan wawancara dengan pemilik Rahayu Trans Travel, ketiadaan informasi lengkap pada *website* menyebabkan proses pemesanan bergantung pada komunikasi WhatsApp. Mekanisme *input* pesanan masih berbasis teks bebas (*free text*) WhatsApp yang tidak terstandarisasi. Kondisi tersebut meningkatkan kemungkinan data tidak lengkap, seperti tanggal atau durasi sewa, sehingga percakapan antara pelanggan dan admin menjadi panjang dan berulang. Selain itu, pengecekan ketersediaan armada dilakukan melalui kalender, sedangkan data transaksi direkap pada Microsoft Excel. Pemisahan media pencatatan ini menimbulkan penumpukan pesan, keterlambatan respons, dan risiko pesanan ganda (*double booking*). Nalendra dan Swalaganata [NAL24] menunjukkan bahwa perancangan antarmuka pada aplikasi layanan kendaraan perlu memperhatikan kebutuhan pengguna agar informasi dan alur layanan dapat dipahami dengan mudah.

Sebagai solusi atas permasalahan tersebut, tugas akhir ini membangun *website* dinamis berbasis *framework* Next.js. Pemilihan Next.js didukung oleh penelitian Nugroho dan Sugandi [NUG24] yang melaporkan peningkatan performa *frontend* pada implementasi Next.js berdasarkan pengujian Lighthouse. Temuan tersebut digunakan sebagai rujukan pemilihan teknologi dari aspek performa, bukan sebagai hasil pengujian sistem Rahayu Trans Travel maupun bukti peningkatan SEO atau visibilitas bisnis. *Website* yang dikembangkan menyediakan katalog armada interaktif dengan fitur penyaringan dan informasi harga, formulir reservasi yang terintegrasi dengan WhatsApp *Click to Chat* menggunakan format data baku, serta *dashboard* admin untuk mengelola data armada, reservasi, dan jadwal secara terpusat.

1.2 Identifikasi Masalah

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

1. Bagaimana mengembangkan website rental mobil dinamis berbasis framework Next.js yang menyediakan katalog armada, detail harga, informasi ketersediaan, serta fungsi pencarian dan penyaringan data kendaraan?
2. Bagaimana memigrasikan formulir reservasi ke format digital yang dapat menstandarisasi data pemesanan pelanggan sebelum dikirimkan ke WhatsApp, sekaligus mengecek ketersediaan tanggal berdasarkan armada yang dipilih agar tidak terjadi jadwal bertabrakan?
3. Bagaimana menyediakan antarmuka *dashboard* bagi admin untuk melakukan pengelolaan data operasional (armada, kalender ketersediaan, zona wilayah dan harga, pemesanan), termasuk konfirmasi manual melalui WhatsApp dan pembatalan otomatis terhadap *booking pending* yang tidak dikonfirmasi dalam waktu 24 jam?

1.3 Tujuan Tugas Akhir

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

1. Mengembangkan website rental mobil dinamis berbasis framework Next.js yang menampilkan katalog armada, spesifikasi kendaraan, detail harga, informasi ketersediaan, serta mendukung fungsi pencarian dan penyaringan data kendaraan.
2. Mengimplementasikan formulir reservasi digital yang mampu menstandarisasi *input* data pemesanan pelanggan menjadi pesan terformat untuk dikirimkan melalui integrasi WhatsApp *Click to Chat*, serta melakukan pengecekan ketersediaan tanggal berdasarkan armada yang dipilih.
3. Membangun halaman *Dashboard* Admin yang mencakup kalender ketersediaan armada, indikator status *booking* dan pembayaran, manajemen zona wilayah dan harga sewa, pengelolaan data armada dan riwayat pemesanan, konfirmasi manual *booking* melalui WhatsApp, serta pembatalan otomatis terhadap *booking pending* yang tidak dikonfirmasi dalam waktu 24 jam.

1.4 Lingkup Tugas Akhir

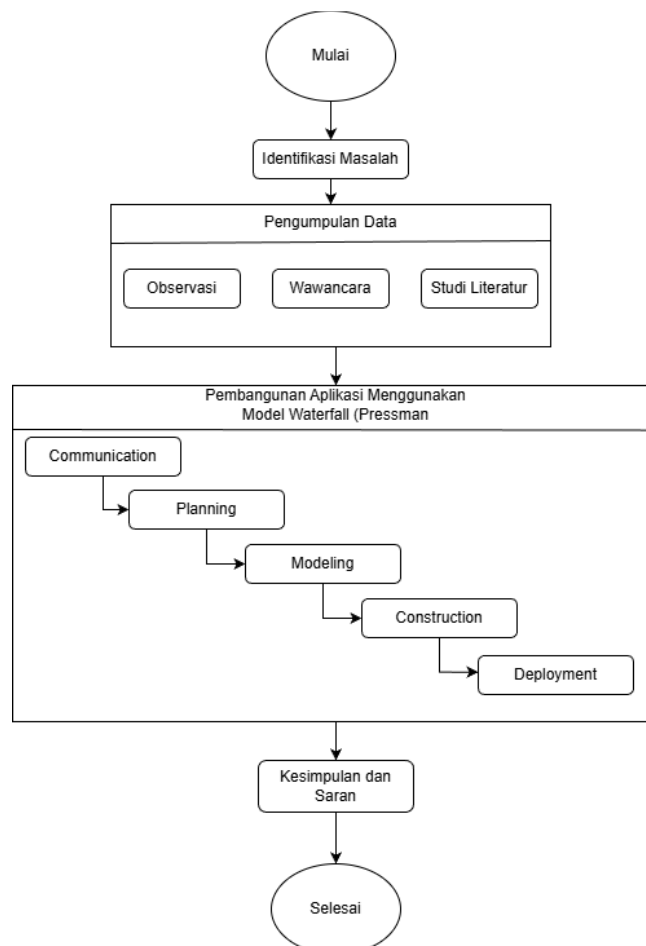
Berikut di bawah ini merupakan lingkup dari tugas akhir:

1. Aplikasi dikembangkan berbasis web menggunakan *framework* Next.js, dapat diakses melalui perangkat *desktop* maupun *mobile*.
2. Penelitian tidak mencakup pembuatan aplikasi *native* berbasis Android atau iOS.
3. Sistem menyediakan dua hak akses: Pelanggan dan Admin.
4. Fitur Pelanggan meliputi katalog kendaraan dengan fungsi filter, detail armada, dan *form booking* terintegrasi WhatsApp.
5. Fitur Admin meliputi manajemen data armada (CRUD), kalender ketersediaan armada, manajemen zona wilayah & harga, pembaruan status pemesanan, dan pengelolaan testimoni.
6. Sistem tidak mengimplementasikan integrasi *payment gateway* otomatis; transaksi pembayaran dilakukan secara manual sesuai prosedur mitra.

7. Pengembangan menggunakan bahasa pemrograman TypeScript/JavaScript dengan basis data relasional.

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 kebutuhan transformasi *website* dari statis menjadi dinamis, pemisahan pencatatan pesanan antara kalender dan Microsoft Excel, serta belum terstandarisasinya format pemesanan melalui WhatsApp. Identifikasi ini menjadi dasar penentuan tujuan penelitian.

2. Tahap Pengumpulan Data

- Observasi (Observation):** Pengamatan langsung terhadap *website* eksisting dan alur pemesanan manual.
- Wawancara (Interview):** Tanya jawab langsung dengan pemilik Rahayu Trans Travel untuk menggali kebutuhan fitur dan kendala operasional.
- Studi Literatur (Literature Review):** Kajian teori dari buku dan jurnal ilmiah terkait sistem informasi berbasis web, UI/UX, pengembangan dengan *Framework* Next.js, dan integrasi WhatsApp *Click to Chat*.

- d. **Dokumentasi:** Pengumpulan data pendukung berupa daftar armada, spesifikasi kendaraan, dan tarif sewa sebagai bahan perancangan basis data dan antarmuka sistem.

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 pengembangan sistem ini dikerjakan secara mandiri oleh satu orang pengembang dengan ruang lingkup dan kebutuhan sistem yang telah terdefinisi sejak awal. Tahapan pengembangan meliputi:

- a. **Communication (Komunikasi & Analisis):** Tahap ini berfokus pada pendefinisian seluruh kebutuhan fungsional dan non-fungsional sistem. Penulis menerjemahkan hasil observasi dan wawancara menjadi spesifikasi fitur, seperti antarmuka katalog armada, formulir reservasi *online*, dan halaman *dashboard* admin.
- b. **Planning (Perancangan):** Tahap perencanaan teknis yang meliputi estimasi waktu pengerjaan, penjadwalan, dan pemilihan teknologi. Pada tahap ini ditetapkan penggunaan *Framework* Next.js, bahasa TypeScript, Tailwind CSS, dan basis data PostgreSQL sebagai fondasi aplikasi.
- c. **Modeling (Pemodelan):** Tahap perancangan logika dan desain sistem. Penulis membuat model visual menggunakan diagram UML (*Use Case* dan *Activity Diagram*) untuk menggambarkan alur sistem, merancang skema *database* (ERD), serta mendesain antarmuka pengguna (UI/UX) menggunakan Figma.
- d. **Construction (Konstruksi):** Tahap penulisan kode program (*coding*). Penulis mengimplementasikan desain antarmuka ke dalam baris kode menggunakan Next.js dan Tailwind CSS. Pada tahap ini juga dilakukan penulisan logika integrasi WhatsApp *Click to Chat* untuk membentuk format pemesanan reservasi yang terstruktur.
- e. **Deployment (Penyerahan & Evaluasi):** Tahap ini berfokus pada pengujian fungsionalitas sistem menggunakan metode *Black-Box Testing* untuk memastikan setiap fitur berjalan sesuai spesifikasi. Setelah itu, dilakukan penerapan (*deployment*) sistem ke *server* publik menggunakan layanan Vercel agar dapat digunakan oleh mitra.

4. Tahap Kesimpulan dan Saran

Menarik kesimpulan dari seluruh proses pengembangan dan memberikan saran untuk pengembangan sistem lebih lanjut.

1.6 Standar dan Kakas

Standar dan kakas yang digunakan dalam tugas akhir ini adalah sebagai berikut:

1. Laptop Pribadi, sebagai perangkat keras utama dalam seluruh proses pengembangan.
2. Visual Studio Code, sebagai *code editor* utama.
3. *Framework* Next.js sebagai kerangka kerja utama (*full-stack framework*) untuk membangun antarmuka dan logika *server* dalam satu proyek.
4. React, sebagai pustaka antarmuka pengguna.

5. TypeScript, sebagai bahasa pemrograman utama.
6. Tailwind CSS, untuk *styling* antarmuka yang responsif.
7. PostgreSQL, sebagai sistem manajemen basis data relasional.
8. Prisma ORM, sebagai *Object-Relational Mapping* untuk akses basis data.
9. NextAuth.js (Auth.js), sebagai pustaka autentikasi.
10. Zod, sebagai validasi skema data.
11. Recharts, sebagai visualisasi data (grafik) pada *dashboard*.
12. Motion, sebagai animasi dan antarmuka pengguna
13. SheetJS (xlsx), pustaka untuk menghasilkan *file* Excel (.xlsx) di sisi klien.
14. React Hook Form, sebagai manajemen dan validasi formulir.
15. Node.js, sebagai lingkungan *runtime* JavaScript.
16. Postman, sebagai alat pengujian *endpoint* API.
17. Git dan GitHub, sebagai sistem kontrol versi.
18. Figma, untuk perancangan desain antarmuka (UI/UX).

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut:

BAB 1 PENDAHULUAN

Memuat latar belakang masalah, identifikasi masalah, tujuan, lingkup, metodologi, standar kakas, serta sistematika penulisan.

BAB 2 LANDASAN TEORI

Membahas teori pendukung yang digunakan dalam penelitian: sistem informasi berbasis web, UI/UX, *framework* Next.js, integrasi WhatsApp *Click to Chat*, metodologi pengembangan, serta tinjauan penelitian terdahulu.

BAB 3 SKEMA PENELITIAN

Menjelaskan profil objek penelitian, metode pengumpulan data, dan rincian tahapan pengembangan sistem.

BAB 4 ANALISIS DAN PERANCANGAN

Menyajikan analisis alur bisnis, kebutuhan sistem, pemodelan UML, perancangan basis data, dan rancangan antarmuka pengguna.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Memaparkan implementasi kode program, integrasi WhatsApp *Click to Chat*, serta hasil pengujian *Black-Box Testing*.

BAB 6 KESIMPULAN DAN SARAN

Berisi kesimpulan dari hasil penelitian dan saran untuk pengembangan sistem di masa mendatang.

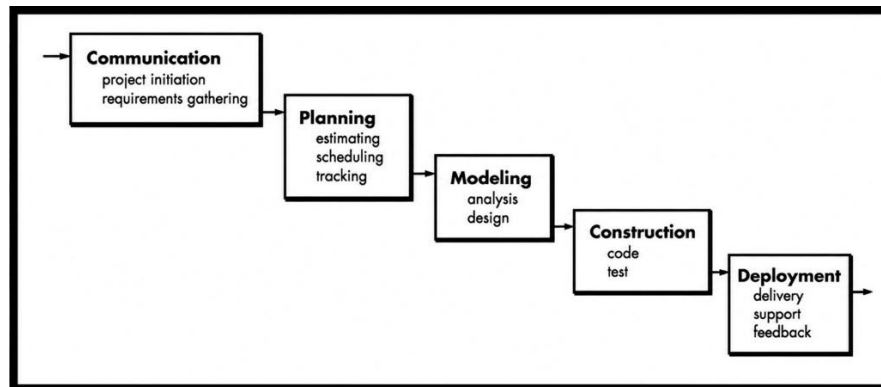
BAB 2

LANDASAN TEORI

Bab ini menguraikan teori-teori pendukung dan penelitian terdahulu yang relevan sebagai landasan pengembangan sistem.

2.1 Teori yang Digunakan

2.1.1 Metode *Waterfall*



Gambar 2.1 Metode *Waterfall*

Sumber: Pressman dan Maxim [PRE20].

Waterfall merupakan salah satu model pengembangan perangkat lunak paling awal yang diperkenalkan secara formal. Pressman dan Maxim [PRE20] mendefinisikan *Waterfall* sebagai model sekuensial klasik (*classic life cycle*) yang memandu pengembangan perangkat lunak melalui serangkaian fase yang terstruktur dan berurutan, di mana setiap fase harus diselesaikan secara tuntas sebelum fase berikutnya dimulai. Model ini juga dikenal sebagai linear sequential model karena alur pengerjaannya yang berjalan satu arah dari atas ke bawah layaknya air terjun.

Menurut Pressman dan Maxim [PRE20], model *Waterfall* terdiri atas lima fase utama yang saling bergantung, yaitu:

1. *Communication* (Komunikasi dan Analisis): Fase ini mencakup pengumpulan dan pendefinisian seluruh kebutuhan sistem secara lengkap sebelum proses perencanaan dimulai. Pada pengembangan *website* ini, fase tersebut meliputi identifikasi kebutuhan fungsional seperti fitur katalog armada, formulir reservasi, dan *dashboard* admin.
2. *Planning* (Perencanaan): Fase perencanaan teknis yang meliputi estimasi waktu pengerjaan, penjadwalan, dan pemilihan teknologi pendukung untuk pengembangan sistem.
3. *Modeling* (Pemodelan): Kebutuhan yang telah dianalisis diterjemahkan menjadi representasi teknis berupa arsitektur sistem, pemodelan UML (*Use Case* dan *Activity Diagram*), perancangan *Entity Relationship Diagram* (ERD), serta desain antarmuka pengguna (UI/UX) menggunakan Figma.
4. *Construction* (Konstruksi dan Pengujian): Tahap penulisan kode program sekaligus pengujian sistem. Pada pengembangan ini, tahap tersebut mencakup penulisan kode menggunakan Next.js, TypeScript, Tailwind CSS, serta pengujian menggunakan metode *Black-Box Testing* untuk memvalidasi setiap alur fungsional utama.

5. *Deployment* (Penerapan): Fase terakhir mencakup evaluasi, perbaikan kesalahan pasca-pengujian, serta penerapan sistem ke *server* publik menggunakan platform Vercel agar dapat digunakan oleh mitra.

Metode *Waterfall* dipilih dalam tugas akhir ini karena pengembangan sistem dikerjakan secara mandiri oleh satu pengembang dengan ruang lingkup kebutuhan yang telah terdefinisi sejak awal berdasarkan hasil wawancara dan observasi lapangan. Kondisi ini sesuai dengan karakteristik proyek yang cocok menggunakan *Waterfall*, yaitu ketika kebutuhan sistem bersifat stabil dan dapat didokumentasikan secara lengkap di awal pengerjaan [PRE20]. Pendekatan sekuensial ini memastikan setiap fase diselesaikan dan terdokumentasi dengan baik sebelum beralih ke fase berikutnya, sehingga menghasilkan produk akhir yang terstruktur dan terverifikasi.

2.1.2 Sistem Informasi Berbasis Web

Sistem informasi berbasis web adalah sistem yang dirancang untuk memproses, mengelola, dan menyajikan informasi melalui protokol HTTP. Pressman & Maxim [PRE20] menyatakan bahwa aplikasi web telah berevolusi dari penyedia informasi statis menjadi sistem komputasi kompleks yang memberikan fungsionalitas bisnis dan interaktivitas tinggi. Sistem ini memungkinkan akses data dari berbagai lokasi selama pengguna terhubung dengan internet.

O'Brien & Marakas [OBR10] mendefinisikan sistem informasi sebagai kombinasi teratur dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi. Dalam konteks bisnis jasa transportasi, sistem informasi berbasis web berperan sebagai jembatan antara perusahaan dan pelanggan, memfasilitasi pemasaran digital dan transaksi pemesanan yang lebih efisien dibandingkan metode konvensional.

2.1.3 *User Experience* (UX) dan *User Interface* (UI)

Norman [NOR13] menjelaskan bahwa *User Experience* (UX) mencakup keseluruhan aspek interaksi pengguna dengan sebuah produk atau sistem. Tujuan utama UX adalah menciptakan sistem yang tidak hanya dapat digunakan (*usable*), tetapi juga menyenangkan dan meminimalkan hambatan kognitif, misalnya ketika pengguna mencari dan memesan kendaraan sewa.

Nalendra dan Swalaganata [NAL24] menjelaskan bahwa perancangan UI/UX pada aplikasi layanan kendaraan perlu memperhatikan kebutuhan pengguna agar informasi dan alur layanan mudah dipahami. Dalam sistem Rahayu Trans Travel, prinsip tersebut diterapkan melalui navigasi yang konsisten, penyajian informasi armada yang jelas, serta alur reservasi yang terstruktur untuk mengurangi kebingungan pelanggan selama proses pemesanan.

2.1.4 Framework Next.js



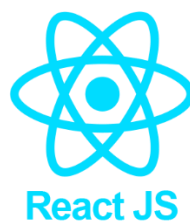
Gambar 2.2 Logo Framework Next.js

Sumber: Dokumentasi resmi Next.js [VER24].

Next.js adalah kerangka kerja pengembangan web berbasis React yang menyediakan kemampuan pemrosesan konten pada *server* serta komponen interaktif pada klien dalam satu aplikasi. Nugroho dan Sugandi [NUG24] melaporkan peningkatan performa *frontend* pada objek penelitian mereka berdasarkan pengujian Lighthouse, yang ditunjukkan melalui penurunan waktu *First Contentful Paint*, *Largest Contentful Paint*, dan *Speed Index* setelah penerapan Next.js. Hasil penelitian tersebut digunakan sebagai referensi mengenai potensi performa Next.js, bukan sebagai hasil pengujian sistem Rahayu Trans Travel.

Dokumentasi resmi Next.js menjelaskan bahwa *App Router* menggunakan *routing* berbasis sistem berkas serta mendukung *Server Components* dan *Client Components*. Next.js juga menyediakan komponen *Image* untuk optimasi gambar, *Metadata API* untuk menetapkan metadata halaman, serta *Route Handlers* untuk memproses permintaan HTTP [VER24]. Pada sistem Rahayu Trans Travel, fitur-fitur tersebut digunakan untuk menyusun halaman aplikasi, menampilkan gambar armada, menetapkan metadata halaman, dan menyediakan *endpoint API* untuk mengelola data armada serta reservasi.

2.1.5 React



Gambar 2.3 Logo React

Sumber: Dokumentasi resmi React [REA24].

React adalah pustaka (*library*) JavaScript *open-source* yang dikembangkan oleh Meta untuk membangun antarmuka pengguna, khususnya pada *Single Page Application* (SPA). React memperkenalkan konsep *Virtual DOM* yang memungkinkan pembaruan tampilan dilakukan secara efisien hanya pada komponen yang berubah tanpa memuat ulang seluruh halaman [FRE 19].

Arsitektur React berbasis komponen (*component-based*) membagi antarmuka menjadi bagian-bagian kecil yang mandiri dan dapat digunakan kembali (*reusable*). React juga mendukung JSX (JavaScript XML), ekstensi sintaks yang memudahkan penulisan struktur HTML di dalam kode JavaScript, sehingga struktur UI lebih mudah divisualisasikan [FRE 19].

2.1.6 TypeScript



Gambar 2.4 TypeScript

Sumber: Dokumentasi resmi TypeScript [MIC24].

TypeScript adalah bahasa pemrograman *open-source* yang dikembangkan oleh Microsoft sebagai *superset* dari JavaScript. TypeScript menambahkan fitur *static typing* yang memungkinkan pengembang mendefinisikan tipe data secara eksplisit, sehingga kesalahan tipe (*type errors*) dapat terdeteksi pada saat kompilasi (*compile-time*) sebelum aplikasi dijalankan [MIC24]. Hal ini meningkatkan keamanan dan stabilitas kode secara signifikan.

2.1.7 PostgreSQL

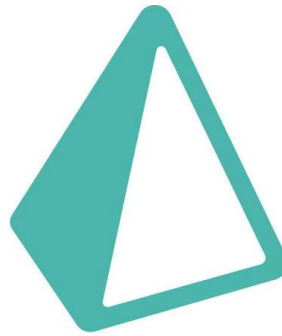


Gambar 2.5 PostgreSQL

Sumber: Dokumentasi resmi PostgreSQL [POS24].

PostgreSQL adalah sistem manajemen basis data relasional (RDBMS) tingkat lanjut yang bersifat *open-source*. PostgreSQL mendukung transaksi ACID (*Atomicity, Consistency, Isolation, Durability*) yang menjamin validitas data bahkan jika terjadi kegagalan sistem [POS24]. Struktur tabel relasional dengan *Primary Key* dan *Foreign Key* menjadikannya sangat cocok untuk menyimpan data transaksional yang kompleks seperti data pesanan dan armada kendaraan.

2.1.8 Prisma ORM



Gambar 2.6 Logo Prisma ORM

Sumber: Dokumentasi resmi Prisma ORM [PRI24].

Prisma adalah alat bantu (*library*) yang digunakan untuk menghubungkan aplikasi berbasis Node.js dan TypeScript dengan basis data relasional secara lebih mudah dan aman [PRI24]. Secara sederhana, Prisma berperan sebagai perantara antara kode program dan basis data, sehingga pengembang tidak perlu menulis perintah SQL secara langsung.

Prisma melibatkan tiga komponen utama yang saling mendukung. Pertama, Prisma Schema, yaitu *file schema.prisma* yang digunakan untuk mendefinisikan model data, relasi, dan konfigurasi sumber data. Kedua, Prisma Client, yaitu *client* akses basis data yang dihasilkan berdasarkan *schema* tersebut agar operasi baca dan tulis dapat digunakan secara *type-safe* dari kode TypeScript. Ketiga, Prisma Migrate, yaitu fitur untuk menghasilkan dan menerapkan perubahan struktur basis data ketika perintah migrasi dijalankan [PRI24].

Dalam pengembangan aplikasi ini, Prisma dipilih karena terintegrasi dengan TypeScript dan Next.js. Setelah *schema.prisma* diperbarui, Prisma Client perlu dibuat ulang melalui proses *generate* agar tipe dan operasi akses data mengikuti model terbaru. Perubahan struktur basis data diterapkan melalui perintah migrasi atau mekanisme penyelarasan skema yang dijalankan pengembang, bukan terjadi otomatis hanya karena *file schema* berubah [PRI24].

2.1.9 NextAuth.js (Autentikasi Berbasis Token)

NextAuth.js adalah pustaka autentikasi *open-source* untuk aplikasi Next.js yang menyediakan solusi lengkap dalam mengelola sesi pengguna, proses masuk (*sign-in*), dan perlindungan rute [AUT24]. Secara sederhana, NextAuth.js berperan sebagai sistem keamanan aplikasi yang memastikan hanya pengguna yang berwenang saja yang dapat mengakses halaman tertentu, seperti panel admin.

Dalam konteks tugas akhir ini, Auth.js dikonfigurasi menggunakan tiga mekanisme utama. Pertama, *Credentials Provider* untuk memeriksa *username* dan *password* admin. Kedua, *JWT Session Strategy* untuk mengelola sesi tanpa memerlukan tabel sesi tambahan pada basis data. *Token* sesi dikelola oleh Auth.js melalui *cookie* sesi dan digunakan saat sistem memeriksa akses pengguna. Ketiga, perlindungan rute admin diterapkan melalui *file proxy.ts* dengan matcher */admin/:path** dan *callback authorized*, sehingga pengguna yang belum terautentikasi diarahkan ke halaman *login* [AUT24].

Strategi JWT dipilih karena sesuai dengan kebutuhan aplikasi yang hanya memiliki akses admin terbatas. Validasi kredensial, pembentukan sesi, dan pemeriksaan akses tetap dilakukan oleh konfigurasi Auth.js pada sisi *server*.

2.1.10 Zod dan React Hook Form (Validasi Data)

Zod adalah pustaka validasi skema (*schema validation*) untuk TypeScript yang memungkinkan pengembang mendefinisikan aturan validasi data secara deklaratif dan *type-safe* [ZOD24]. Setiap skema Zod secara otomatis menghasilkan tipe TypeScript yang sesuai, sehingga konsistensi antara validasi *runtime* dan pengecekan tipe *compile-time* terjamin tanpa duplikasi kode.

React Hook Form adalah pustaka manajemen formulir untuk React yang mengutamakan performa tinggi dengan meminimalkan *re-render* komponen [RHF24]. Pustaka ini menggunakan pendekatan *uncontrolled components* dan referensi DOM langsung, sehingga formulir dengan banyak *field* tetap responsif. Integrasi antara Zod dan React Hook Form melalui `@hookform/resolvers` memungkinkan validasi skema diterapkan secara otomatis pada setiap *field* formulir, menghasilkan pesan kesalahan yang konsisten dan terlokalisasi.

Dalam konteks tugas akhir ini, kombinasi Zod dan React Hook Form digunakan pada formulir reservasi untuk memastikan data pemesanan pelanggan (nama, nomor telepon, tanggal, dan pilihan armada) telah tervalidasi sebelum dikirimkan ke WhatsApp *Click to Chat*.

2.1.11 Recharts (Visualisasi Data)

Recharts adalah pustaka visualisasi data berbasis React yang dibangun di atas D3.js dan menyediakan komponen grafik deklaratif seperti *Bar Chart*, *Line Chart*, dan *Pie Chart* [REC24]. Pendekatan berbasis komponen React memungkinkan grafik diintegrasikan secara alami ke dalam antarmuka aplikasi tanpa manipulasi DOM secara manual.

Dalam tugas akhir ini, Recharts digunakan pada halaman *Dashboard* Admin untuk menampilkan statistik operasional seperti jumlah pemesanan per periode dan distribusi status *booking*, sehingga admin dapat memantau kinerja bisnis secara visual.

2.1.12 Motion (Animasi Antarmuka)

Motion (sebelumnya dikenal sebagai Framer Motion) adalah pustaka animasi untuk React yang menyediakan API deklaratif untuk membuat transisi, gestur, dan animasi tata letak (*layout animations*) [MOT24]. Pustaka ini mengabstraksi kompleksitas Web Animations API dan CSS transitions ke dalam properti komponen yang mudah dibaca, seperti `initial`, `animate`, dan `exit`.

Animasi antarmuka yang digunakan secara tepat dapat membantu memperjelas perubahan keadaan dan menjaga kesinambungan interaksi pengguna. Dalam tugas akhir ini, Motion digunakan pada transisi halaman, efek tampil elemen katalog, dan interaksi kartu armada sebagai dukungan visual, bukan sebagai ukuran langsung peningkatan kepuasan pengguna [NOR13][MOT24].

2.1.13 SheetJS (xlsx)

SheetJS merupakan pustaka JavaScript untuk membaca dan menghasilkan berkas spreadsheet, seperti Excel, CSV, dan ODS, secara client-side [SHE23]. Pada sistem ini, SheetJS digunakan untuk mengekspor data armada, booking, dan testimoni dari panel admin ke berkas Excel untuk pelaporan dan arsip.

2.1.14 Tailwind CSS



Gambar 2.7 Logo Tailwind CSS

Sumber: Dokumentasi resmi Tailwind CSS [TAI24].

Tailwind CSS adalah kerangka kerja CSS yang mengutamakan utilitas (*utility-first framework*) untuk membangun desain antarmuka kustom. Berbeda dengan Bootstrap yang menyediakan komponen siap pakai, Tailwind memberikan kelas utilitas tingkat rendah yang dikombinasikan langsung di dalam elemen HTML, sehingga pengembang memiliki kontrol penuh terhadap desain tanpa menulis CSS kustom yang berlebihan [TAI24].

Tailwind mendukung desain responsif dengan pendekatan *mobile-first*. Pada versi 4 yang digunakan dalam tugas akhir ini, Tailwind CSS memperkenalkan *engine* baru, konfigurasi CSS-*first*, dan pendeteksian sumber kelas secara otomatis [TAI25]. Pada proyek Rahayu Trans Travel, Tailwind CSS v4 digunakan melalui paket `tailwindcss` dan `@tailwindcss/postcss` untuk membangun antarmuka responsif. Tailwind telah ditulis ulang sepenuhnya (*ground-up rewrite*) dengan mesin baru berperforma tinggi yang mampu mendeteksi kelas utilitas secara otomatis dari seluruh berkas proyek tanpa memerlukan konfigurasi manual, sehingga menghasilkan berkas CSS yang sangat kecil dan meningkatkan kecepatan muat halaman.

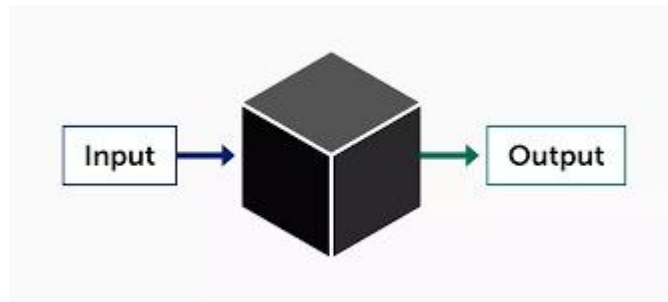
2.1.15 Integrasi WhatsApp *Click to Chat*

WhatsApp Click to Chat memungkinkan percakapan dibuka melalui tautan yang memuat nomor tujuan dan teks pesan yang telah disiapkan [WHA24]. Pada sistem Rahayu Trans Travel, data reservasi yang telah divalidasi dikonversi menjadi teks baku, kemudian disisipkan ke parameter URL `https://api.whatsapp.com/send` sebelum pelanggan diarahkan ke percakapan admin. Pesan tetap dikirim oleh pelanggan melalui aplikasi WhatsApp.

Wana dkk. [WAN24] menunjukkan bahwa integrasi WhatsApp pada sistem rental mobil dapat membantu komunikasi dan notifikasi antara pelanggan dan pengelola. Penelitian tersebut menggunakan WhatsApp Gateway, sedangkan sistem Rahayu Trans Travel menggunakan *Click to Chat*. Oleh karena itu, WAN24 digunakan sebagai referensi manfaat komunikasi WhatsApp secara umum, bukan sebagai bukti mekanisme *Click to Chat* atau pengiriman data otomatis yang bebas kesalahan.

Pendekatan ini dipilih berdasarkan kebutuhan mitra yang masih melakukan konfirmasi dan pembayaran secara manual melalui admin. *Click to Chat* digunakan untuk menstandarkan pesan reservasi dan membuka percakapan WhatsApp, sedangkan keputusan konfirmasi *booking* serta verifikasi pembayaran tetap dilakukan oleh admin.

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



Gambar 2.8 Ilustrasi Pengujian Black-Box

Sumber: Ilustrasi konsep berdasarkan Pressman dan Maxim [PRE20] serta Nidhra dan Dondeti [NID12].

Black-Box Testing adalah metode pengujian perangkat lunak yang memvalidasi fungsionalitas sistem tanpa melihat struktur kode internal. Pressman & Maxim [PRE20] mendeskripsikan pendekatan ini sebagai metode uji coba untuk menemukan kesalahan pada fungsi yang tidak benar atau hilang, kesalahan antarmuka (*interface errors*), dan kesalahan kinerja. Penguji hanya berfokus pada apakah *input* yang diberikan menghasilkan *output* yang sesuai spesifikasi. Nidhra dan Dondeti [NID12] dalam kajiannya menambahkan bahwa teknik *Black-Box Testing* mencakup beberapa pendekatan seperti *Equivalence Partitioning*, *Boundary Value Analysis*, dan *Decision Table Testing*, yang masing-masing efektif untuk mendeteksi jenis kesalahan yang berbeda pada antarmuka sistem.

Mustaqbal dkk. [MUS15] menunjukkan bahwa *Boundary Value Analysis* dapat digunakan untuk menguji nilai batas masukan dan menemukan kekurangan validasi tanpa meninjau struktur kode program. Dalam tugas akhir ini, *Black-Box Testing* digunakan untuk memverifikasi keluaran setiap fungsi berdasarkan masukan dan skenario yang telah ditetapkan, termasuk filter armada, formulir reservasi, integrasi WhatsApp, dan operasi pengelolaan data oleh admin.

2.1.17 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan artefak dari sebuah sistem perangkat lunak. UML menyediakan sekumpulan diagram logis yang membantu pengembang dalam menerjemahkan rancangan sistem yang kompleks ke dalam bentuk draf cetak biru (*blueprint*) yang terstandarisasi. Menurut Pressman dan Maxim [PRE20], aplikasi modem memerlukan pemodelan yang kuat untuk memastikan seluruh fungsionalitas bisnis dan interaktivitas tinggi dapat terpenuhi tanpa kendala komunikasi teknis antara analis sistem dan pemrogram.

Dalam tahapan perancangan sistem, pemodelan berbasis objek ini membagi diagramnya ke dalam beberapa jenis sesuai dengan sudut pandang kebutuhan sistem [PRE20]. Pada pengembangan aplikasi berbasis web ini, pemodelan difokuskan pada *Use Case Diagram* untuk menggambarkan fungsionalitas sistem dari sudut pandang interaksi aktor dengan aplikasi, serta *Activity Diagram* untuk memvisualisasikan alur kerja (*workflow*) dari setiap proses bisnis yang terjadi. Penggunaan kedua diagram ini memastikan bahwa seluruh spesifikasi operasional, mulai dari prosedur reservasi pelanggan hingga manajemen *dashboard* oleh admin Rahayu Trans Travel, dapat terpetakan dengan jelas dan sistematis.

2.1.18 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan sebuah model konseptual yang digunakan untuk merancang dan mendeskripsikan struktur logis dari sebuah basis data relasional. Model ini memvisualisasikan bagaimana data disimpan dan saling terkait di dalam sistem melalui penggunaan tiga komponen utama, yaitu entitas, atribut, dan relasi. Menurut Pressman dan Maxim [PRE20], perancangan model data yang terstruktur melalui diagram hubungan entitas ini menjadi tahapan krusial untuk memetakan kebutuhan penyimpanan informasi sebelum sebuah sistem memasuki fase konstruksi atau penulisan kode program.

Dalam sistem informasi reservasi yang menangani banyak data transaksi secara simultan, perancangan ERD yang presisi sangat dibutuhkan untuk mencegah terjadinya redundansi atau anomali data. Skema relasional yang dihasilkan dari ERD ini nantinya akan diimplementasikan ke dalam sistem manajemen basis data relasional (RDBMS) seperti PostgreSQL untuk menjamin pemrosesan transaksi yang valid dan konsisten [PRE20]. Melalui pemodelan yang akurat, setiap entitas operasional mulai dari data pengguna, jadwal armada kendaraan, hingga riwayat reservasi dapat terhubung dengan aman melalui relasi kunci (*primary key* dan *foreign key*).

2.1.19 Vercel

Vercel merupakan platform *cloud* yang digunakan untuk *deployment* aplikasi Next.js. Dokumentasi Vercel menjelaskan bahwa CDN mendistribusikan konten melalui jaringan global dan menangani *caching* pada *edge network* [VCD26]. Dalam tugas akhir ini, Vercel digunakan untuk menjalankan aplikasi web dan *Route Handlers* pada lingkungan produksi.

Integrasi Git pada Vercel dapat membuat *deployment* baru ketika perubahan dikirim ke repositori yang terhubung [VGT26]. Mekanisme ini digunakan pada tahap *deployment* setelah implementasi dan pengujian aplikasi selesai. Keberhasilan *deployment* tetap bergantung pada konfigurasi proyek, *environment variable*, proses *build*, dan ketersediaan layanan.

2.2 Penelitian Terdahulu

Berikut adalah penelitian-penelitian terdahulu yang relevan dengan topik tugas akhir ini:

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Hasil Penelitian	Alasan Digunakan
1	Raihan, N., & Eviyanti, A. (2022), ZONAsi: Jurnal Sistem Informasi, Vol. 4, No. 1.	Penelitian menghasilkan sistem informasi penyewaan mobil berbasis web menggunakan <i>framework</i> CodeIgniter untuk membantu pengelolaan data pelanggan, kendaraan, pemesanan, dan transaksi secara terkomputerisasi.	Menjadi referensi perancangan basis data dan alur reservasi kendaraan. Alur tersebut diadaptasi sesuai kebijakan Rahayu Trans Travel yang menyediakan layanan kendaraan dengan sopir. [NAU22].
2	Nugroho, R. P. A., & Sugandi, A. (2024), Journal of Cyber Health and Computer, Vol. 2, No. 2.	Penelitian melaporkan peningkatan performa sebesar 62,5% berdasarkan pengujian Lighthouse, terutama pada <i>First Contentful Paint</i> , <i>Largest Contentful Paint</i> , dan <i>Speed Index</i> .	Menjadi referensi pendukung pemilihan Next.js dari aspek performa <i>frontend</i> . Penelitian ini tidak digunakan sebagai bukti langsung peningkatan SEO sistem Rahayu Trans Travel. [NUG24].
3	Maulana, R. R., Sumaryana, Y., & Anwar, D. S. (2024), Jurnal Informatika dan Teknik Elektro Terapan, Vol. 12, No. 3.	Menghasilkan rancangan <i>User Interface</i> (UI) dan <i>User Experience</i> (UX) untuk aplikasi penyewaan mobil dengan pendekatan yang berpusat pada pengguna, sehingga alur pemesanan menjadi lebih efisien.	Menjadi referensi dalam merancang UI/UX sistem rental mobil, khususnya alur pengguna, katalog, dan formulir reservasi. Penelitian ini digunakan sebagai acuan antarmuka, bukan sebagai dasar penerapan Design Thinking [MAU24].

No	Peneliti	Hasil Penelitian	Alasan Digunakan
4	Siregar, N. M., & Fatmawati (2024), <i>Information Management for Educators and Professionals</i> , Vol. 9, No. 1.	Penelitian menghasilkan rancangan sistem informasi penyewaan mobil berbasis web yang memusatkan pengelolaan data pelanggan, kendaraan, dan transaksi serta menyediakan informasi ketersediaan armada.	Menjadi referensi pendukung dalam memodelkan <i>use case</i> dan alur fungsionalitas dasar sistem penyewaan kendaraan, khususnya pada bagian pengelolaan katalog armada dan pencatatan transaksi penyewaan berbasis web untuk diterapkan pada sistem Rahayu Trans Travel. [SIR24]
5	Pangestu, B. R., & Voutama, A. (2024) <i>JATI (Jurnal Mahasiswa Teknik Informatika)</i> , Vol. 8, No. 3	Menghasilkan sistem informasi rental mobil berbasis <i>website</i> yang dikembangkan menggunakan metode <i>Waterfall</i> . Penelitian menghasilkan sistem yang dikembangkan secara terstruktur menggunakan <i>Waterfall</i> .	Menjadi referensi utama dalam penerapan tahapan <i>Software Development Life Cycle</i> (SDLC) menggunakan metode <i>Waterfall</i> . Jurnal ini dijadikan acuan agar proses perancangan sistem reservasi Rahayu Trans Travel (dari tahap analisis kebutuhan, desain, <i>coding</i> , hingga pengujian) berjalan secara sistematis, terstruktur, dan terukur. [PAN24]
6	Wana, S., Chandra, A., & Iryani, J. (2024) <i>Lajutek: Journal of Informatics, Multimedia, And Information</i>	Menghasilkan sistem informasi rental mobil berbasis web yang mengintegrasikan layanan pesan singkat melalui WhatsApp Gateway untuk mempermudah komunikasi dan notifikasi.	Menjadi referensi manfaat integrasi WhatsApp pada proses komunikasi rental mobil. Mekanisme teknis penelitian berbeda karena sistem Rahayu Trans Travel menggunakan <i>Click to Chat</i> . [WAN24]
7	Geasela, Y. M., dkk. (2023) <i>Infotech: Journal of Technology Information</i> , Vol. 9, No. 2	Penelitian merancang dan membangun aplikasi penyewaan mobil <i>online</i> berbasis <i>website</i> bernama Eazyrent yang menyediakan pencarian kendaraan, pemesanan <i>online</i> , dan manajemen data oleh admin.	Menjadi perbandingan konsep pengembangan aplikasi rental mobil <i>online</i> berbasis <i>website</i> , khususnya pada fitur pencarian armada dan alur pemesanan yang serupa dengan sistem yang dibangun pada tugas akhir ini. [GEA23]

BAB 3

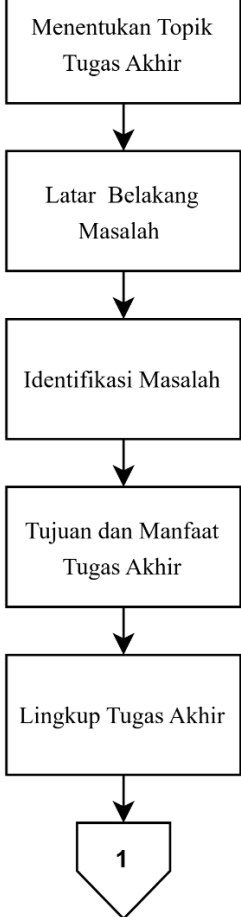
SKEMA PENELITIAN

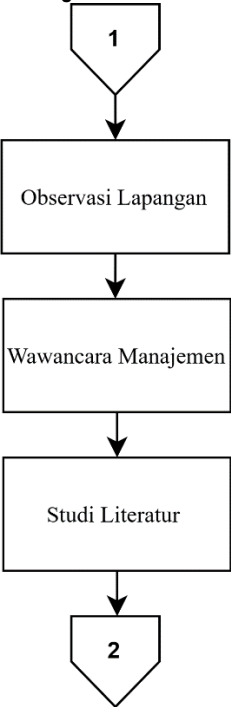
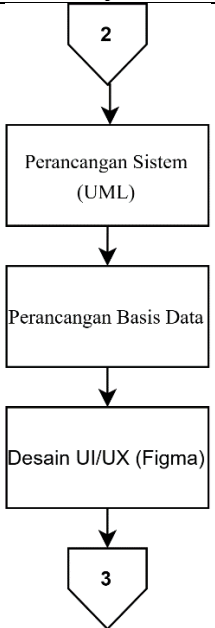
Bab ini berisi penjelasan alur penyelesaian tugas akhir, perumusan masalah, kerangka pemikiran teoretis, dan profil tempat penelitian. Skema penelitian berfungsi sebagai panduan awal bagi pembaca untuk memahami tujuan, ruang lingkup, dan landasan teoretis penelitian yang akan dilakukan. Dengan menyediakan 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.

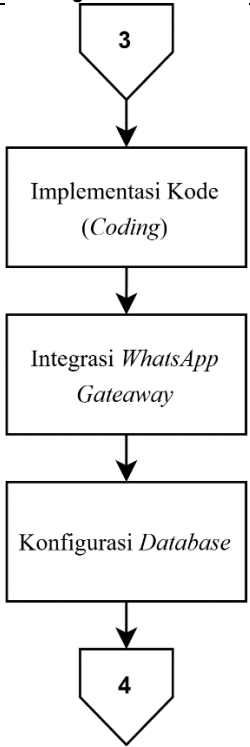
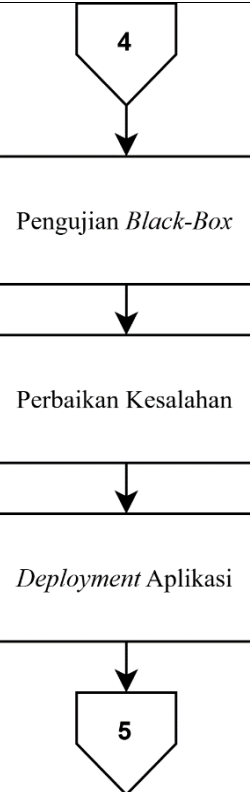
3.1 Alur Penyelesaian Tugas Akhir

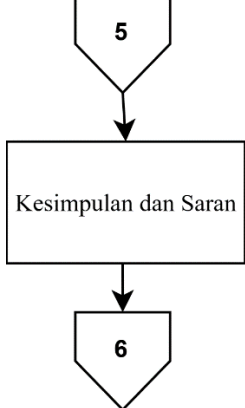
Alur penyelesaian tugas akhir ini dilakukan berdasarkan diagram alur penyelesaian yang sistematis. Berikut ini merupakan tabel alur penyelesaian tugas akhir:

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 reservasi kendaraan <i>online</i>. Selain itu, dari proses identifikasi masalah dapat mengetahui kendala pencatatan manual (Excel) dan pemesanan via WhatsApp yang sedang terjadi, serta menganalisis solusi integrasi WhatsApp <i>Click to Chat</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 WhatsApp <i>Click to Chat</i> dan <i>Framework</i> Next.js 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 Rahayu Trans Travel 2. Teori Sistem Informasi 3. Metodologi Penelitian

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 2: Pengumpulan data. Hasil: Memahami tentang teori yang akan digunakan dalam pembuatan <i>website</i> reservasi, data spesifikasi kendaraan, serta dokumentasi integrasi WhatsApp <i>Click to Chat</i>. Kontribusi: Berguna untuk fondasi pembuatan <i>database</i> dan alur logika pemesanan pada <i>website</i>.	 <pre> graph TD 1{{1}} --> A[Observasi Lapangan] A --> B[Wawancara Manajemen] B --> C[Studi Literatur] C --> 2{{2}} </pre>	1. Dokumentasi Next.js 2. Metode <i>Waterfall</i> 3. Jurnal Terkait
Tahap 3: <i>Pra-Production</i> (Perancangan). Hasil: Gambaran atau <i>mockup</i> desain antarmuka <i>website</i>, 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 <i>Waterfall</i> 2. Teori UML & ERD 3. Konsep UI/UX Design

Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 4: Production (Pembangunan). Hasil: <i>Prototype website</i> reservasi kendaraan yang sudah selesai atau sudah jadi secara fungsional (dapat melakukan pencarian katalog dan pemesanan). Kontribusi: Menghasilkan produk perangkat lunak sebagai solusi utama tugas akhir.	 <pre> graph TD 3{{3}} --> A[Implementasi Kode (Coding)] A --> B[Integrasi WhatsApp Gateway] B --> C[Konfigurasi Database] C --> 4{{4}} </pre>	Literatur dan Referensi 1. Bahasa Pemrograman TypeScript 2. Framework Next.js & Tailwind 3. Database PostgreSQL
Tahap 5: Deploy dan Pengujian. Hasil: <i>Prototype website</i> yang sudah teruji bebas <i>bug</i> (<i>Black-Box Testing</i>) dan berhasil di-deploy ke server publik agar dapat diakses. Kontribusi: Memastikan sistem berjalan lancar saat digunakan oleh pengguna (Rahayu Trans Travel).	 <pre> graph TD 4{{4}} --> A[Pengujian Black-Box] A --> B[Perbaikan Kesalahan] B --> C[Deployment Aplikasi] C --> 5{{5}} </pre>	1. Teori <i>Black-Box Testing</i> 2. Dokumentasi Vercel

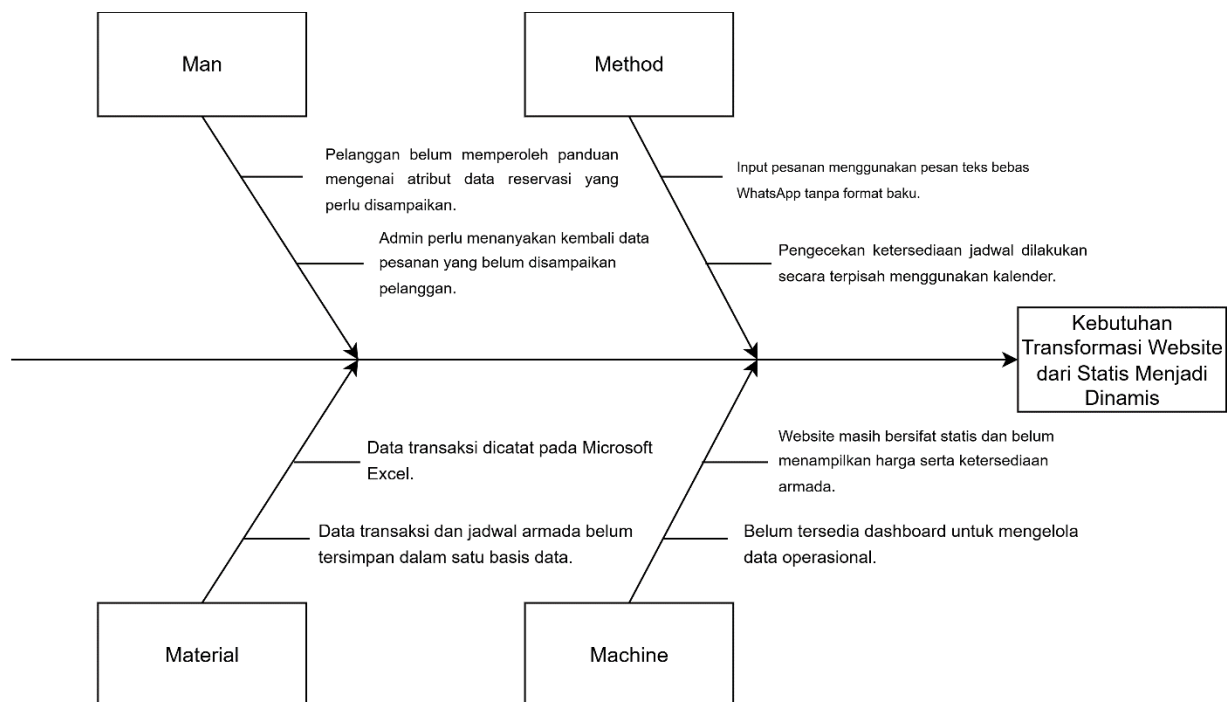
Tahap dan Hasil	Langkah Penelitian	Literatur dan Referensi
Tahap 6: Simpulan. Hasil: Kesimpulan dan saran. Kontribusi: Kesimpulan dan saran berdasarkan hasil yang telah dicapai dalam pembuatan <i>website</i> reservasi kendaraan.		1. Evaluasi Tujuan Awal

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 pemilik Rahayu Trans Travel, akar masalah yang ditemukan adalah "Kebutuhan Transformasi *Website* dari Statis Menjadi Dinamis". Faktor penyebabnya dianalisis menggunakan diagram Fishbone sebagai berikut:



Gambar 3.1 Fishbone Masalah

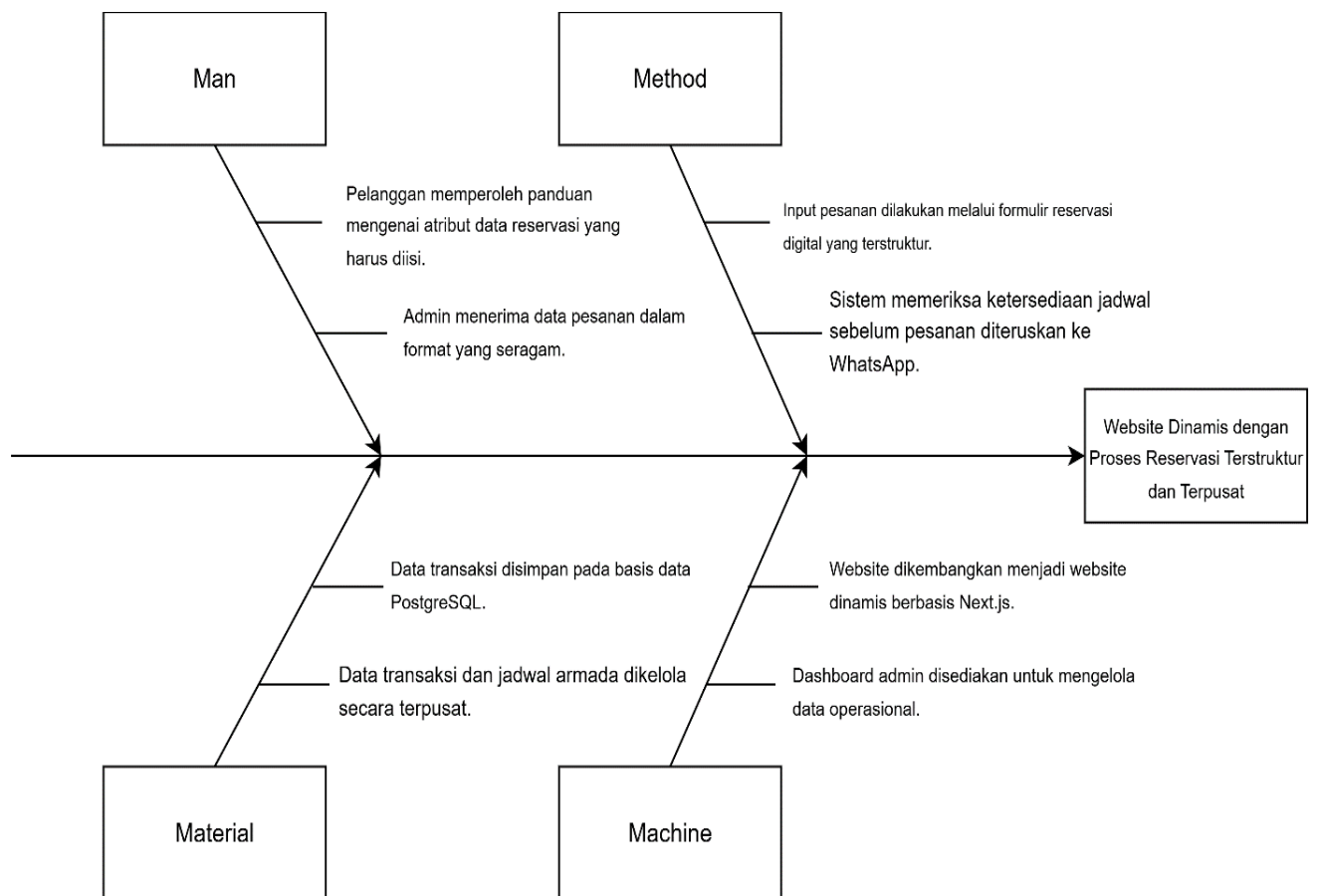
Berdasarkan Gambar 3.1, berikut adalah penjelasan mengenai faktor-faktor penyebab kebutuhan transformasi *website* Rahayu Trans Travel dari statis menjadi dinamis:

1. **Manusia (Man):** Pelanggan belum memperoleh panduan mengenai atribut data yang perlu disampaikan ketika melakukan reservasi, seperti nama pemesan, jenis armada, tanggal mulai, dan tanggal selesai. Oleh karena itu, admin perlu menanyakan kembali data pesanan yang belum disampaikan sebelum melanjutkan proses reservasi.

2. **Metode (Method):** Mekanisme *input* pesanan masih menggunakan pesan teks bebas (*free text*) WhatsApp tanpa format baku. Pengecekan ketersediaan armada juga dilakukan secara terpisah menggunakan kalender sehingga proses *input* pesanan dan pemeriksaan jadwal belum berada dalam satu alur yang terintegrasi.
3. **Material (Data):** Data transaksi dicatat pada Microsoft Excel, sedangkan jadwal penggunaan armada dicatat menggunakan kalender. Kedua media tersebut tidak terhubung sehingga data transaksi dan jadwal armada belum tersimpan dalam satu basis data. Kondisi ini menimbulkan risiko ketidaksesuaian data dan pesanan ganda (*double booking*).
4. **Teknologi (Machine):** Website Rahayu Trans Travel masih bersifat statis dan belum menyajikan informasi harga serta ketersediaan armada. Selain itu, belum tersedia *dashboard* dan basis data reservasi terpusat yang dapat digunakan admin untuk mengelola data armada, jadwal, dan transaksi.

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 sistem.



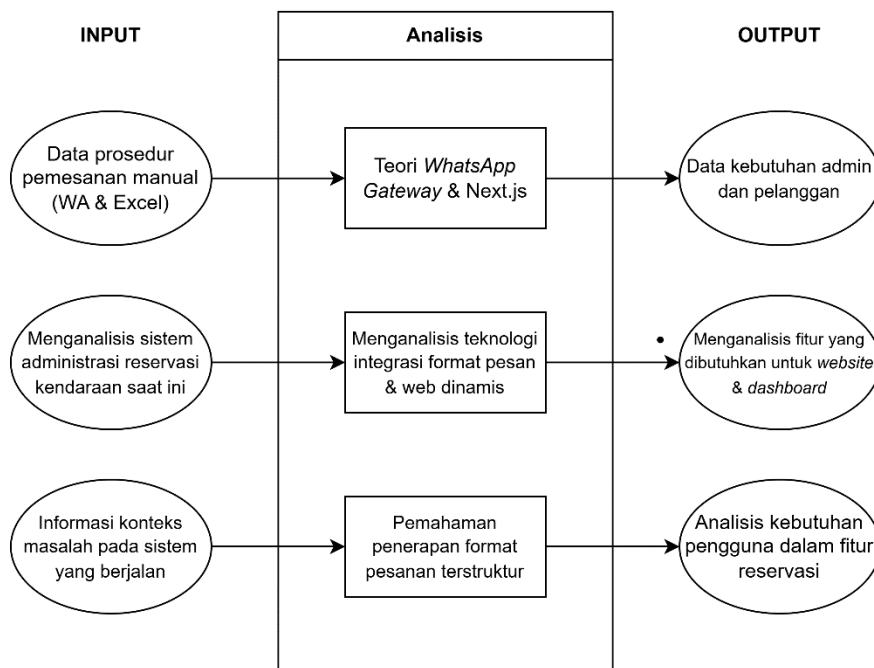
Gambar 3.2 Fishbone Solusi

Berdasarkan Gambar 3.2, berikut adalah penjelasan mengenai usulan solusi untuk mentransformasikan *website* statis menjadi *website* dinamis:

1. **Manusia (Man):** Pelanggan memperoleh panduan mengenai atribut data reservasi melalui formulir digital yang memuat Nama Pemesan, Nomor WhatsApp, Jenis Armada, Tanggal Mulai, dan Tanggal Selesai. Dengan format tersebut, admin menerima data pesanan dalam susunan yang seragam untuk diperiksa dan diproses lebih lanjut.
2. **Metode (Method):** Proses *input* pesanan dimigrasikan dari pesan teks bebas WhatsApp ke formulir reservasi digital yang terstruktur. Sistem memvalidasi kelengkapan data dan memeriksa ketersediaan jadwal pada tabel *Booking*. Jika jadwal bertabrakan, tombol pemesanan dinonaktifkan dan sistem menampilkan peringatan. Sebelum pelanggan diarahkan ke WhatsApp, sistem melakukan pemeriksaan ulang untuk memastikan jadwal masih tersedia.
3. **Material (Data):** Data pemesanan, armada, jadwal, biaya, dan status reservasi disimpan secara terpusat pada basis data relasional PostgreSQL. Tabel *Booking* digunakan sebagai acuan untuk memeriksa ketersediaan armada berdasarkan jenis armada serta rentang tanggal yang dipilih sehingga risiko *double booking* dapat dicegah oleh sistem.
4. **Teknologi (Machine):** *Website* statis ditransformasikan menjadi aplikasi web dinamis berbasis *framework* Next.js yang terhubung dengan PostgreSQL melalui Prisma ORM. Sistem menyediakan katalog armada, informasi harga, formulir reservasi, pemeriksaan jadwal, integrasi WhatsApp *Click to Chat*, serta *dashboard* admin untuk mengelola data operasional secara terpusat.

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 proses reservasi yang sedang berjalan.	Prosedur reservasi melalui WhatsApp serta pencatatan jadwal dan transaksi menggunakan kalender dan Microsoft Excel.	Gambaran alur reservasi, atribut data pesanan, media pencatatan, dan proses pemeriksaan ketersediaan armada.	Mengidentifikasi kebutuhan transformasi <i>website</i> dari statis menjadi dinamis serta kebutuhan pengelolaan data reservasi secara terstruktur dan terpusat.
Menganalisis teknologi pengembangan dan integrasi sistem.	Dokumentasi <i>framework</i> Next.js dan WhatsApp <i>Click to Chat</i> .	Pemahaman mengenai pengembangan <i>website</i> dinamis serta pembentukan pesan reservasi terstruktur untuk diteruskan ke WhatsApp.	Menentukan teknologi yang digunakan untuk membangun <i>website</i> dinamis dan mengintegrasikan data reservasi dengan WhatsApp <i>Click to Chat</i> .
Mengumpulkan data mengenai fitur yang dibutuhkan admin dan pelanggan.	Kebutuhan admin dan pelanggan.	Analisis kebutuhan pengguna dalam fitur reservasi kendaraan dan panel <i>dashboard</i> .	Untuk mendapatkan gambaran informasi tentang fitur yang akan dirancang pada antarmuka pengguna.
Mengumpulkan data mengenai kebutuhan penjadwalan armada	Jadwal penyewaan yang dicatat di kalender fisik	Informasi tentang kebutuhan visualisasi ketersediaan armada per tanggal	Untuk merancang fitur kalender ketersediaan berbasis web

3.3 Profil Tempat Penelitian

Penelitian ini dilaksanakan di Rahayu Trans Travel, sebuah perusahaan yang bergerak di bidang jasa transportasi dan penyewaan kendaraan (*rental* mobil). Pemahaman terhadap profil instansi diperlukan untuk memastikan solusi sistem reservasi yang dirancang sesuai dengan karakteristik operasional dan kebutuhan perusahaan.

3.3.1 Objek Penelitian

Objek penelitian dalam penyusunan tugas akhir ini adalah sistem administrasi reservasi dan pengelolaan jadwal armada di Rahayu Trans Travel. Pemilihan objek penelitian ini didasarkan pada kebutuhan digitalisasi proses pemesanan. Pada sistem yang sedang berjalan, pelanggan menyampaikan data reservasi melalui pesan teks bebas WhatsApp tanpa panduan atribut dan format isian yang baku. Pencatatan transaksi dilakukan menggunakan Microsoft Excel, sedangkan ketersediaan armada diperiksa melalui kalender. Pemisahan media tersebut menyebabkan data transaksi dan jadwal armada belum tersimpan dalam satu basis data reservasi. Adapun identitas objek penelitian adalah sebagai berikut:



Gambar 3.4 Logo Rahayu Trans Travel Bandung

- Nama Instansi: Rahayu Trans Travel Bandung.
- Alamat Instansi: Jl. Perumahan Bukit Arcamanik Tahap I No.2 blok A3, Sindanglaya, Kec. Cimenyan, Kabupaten Bandung, Jawa Barat 40195, Indonesia
- Fokus Penelitian: Pengembangan *website* rental mobil dinamis berbasis Next.js yang terintegrasi dengan WhatsApp *Click to Chat* serta *dashboard* admin untuk standardisasi format pemesanan dan pengelolaan ketersediaan armada.

3.4 Teknik Pengumpulan Data

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

1. Studi Literatur (*Literature Review*):

Penulis mengumpulkan dan mempelajari referensi teoritis dari buku teks, jurnal ilmiah, dan dokumentasi teknis. Fokus studi literatur meliputi sistem informasi berbasis web, integrasi WhatsApp *Click to Chat*, metodologi *Waterfall*, desain UI/UX, dan pengembangan menggunakan *Framework* Next.js.

2. Wawancara (*Interview*):

Teknik ini dilakukan melalui tanya jawab secara langsung dengan pemilik Rahayu Trans Travel. Penulis menggali informasi mengenai kendala operasional seperti keterlambatan respons akibat melayani pertanyaan berulang, serta risiko *double booking* pada pengarsipan data.

3. Observasi (*Observation*):

Penulis melakukan pengamatan langsung untuk memahami prosedur operasional yang sedang berjalan. Objek observasi meliputi alur pemesanan berbasis teks bebas (*free text*) via pesan WhatsApp, prosedur pengecekan jadwal armada pada alat bantu fisik (kalender), dan pengarsipan data riwayat pesanan menggunakan lembar kerja *spreadsheet* (Microsoft Excel).

4. Dokumentasi (*Documentation*):

Pengumpulan data pendukung berupa daftar armada, spesifikasi kendaraan, dan tarif sewa berdasarkan zona wilayah tujuan. Data ini digunakan sebagai landasan perancangan skema basis data dan antarmuka katalog 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 kendala prosedur reservasi dan kebutuhan fitur fungsional aplikasi.
2. **Lembar Observasi (*Observation Sheet*):** Daftar periksa untuk mengamati prosedur rekapitulasi data menggunakan *file* Microsoft Excel dan pengecekan jadwal pada kalender fisik.

3.5.2 Alat Pengembangan Sistem

Untuk merancang dan membangun platform *website* ini, penulis menggunakan perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut:

a. Perangkat Keras (*Hardware*):

- Laptop MSI Modern 14 B11MOU dengan prosesor Intel Core i5-1155G7.

b. Perangkat Lunak (*Software*):

- Sistem Operasi: Windows 11.
- *Code Editor*: Visual Studio Code.
- Desain UI/UX: Figma.
- *Web Browser*: Google Chrome, Edge.
- *Runtime Environment*: Node.js.
- *Framework & Library*: Next.js, React, dan Tailwind CSS.
- *Database*: PostgreSQL.

3.6 Metode Pengembangan Perangkat Lunak

Sesuai dengan kerangka kerja yang telah ditetapkan pada Bab 1, metode pembangunan aplikasi yang digunakan dalam penelitian ini adalah Model *Waterfall* menurut Roger S. Pressman (2020). Model ini dipilih karena pengembangan sistem dikerjakan secara mandiri oleh satu pengembang dengan ruang lingkup dan spesifikasi kebutuhan yang telah terdefinisi secara utuh sejak awal. Tahapan pengembangan meliputi:

1. **Communication (Komunikasi & Analisis):** Tahap ini berfokus pada analisis kebutuhan perangkat lunak (*Software Requirement Analysis*). Penulis menganalisis kendala dari prosedur *input* pelanggan yang berbasis teks bebas (*free text*) serta risiko *double booking* akibat penyimpanan data yang terpisah. Berdasarkan analisis tersebut, penulis merumuskan spesifikasi kebutuhan fungsional sistem seperti fitur katalog transparan, *form* tervalidasi, dan *dashboard* manajemen.
2. **Planning (Perancangan Teknis):** Tahap perencanaan yang meliputi penjadwalan pengerjaan tugas akhir dan pemilihan arsitektur teknologi (*tech stack*). Pada tahap ini ditetapkan penggunaan *Framework* Next.js sebagai kerangka antarmuka dinamis dan PostgreSQL sebagai basis data relasional tersentralisasi.
3. **Modeling (Pemodelan):** Tahap perancangan logika bisnis dan desain visual sistem. Penulis membuat model rancangan sistem menggunakan diagram UML (*Use Case Diagram* dan *Activity Diagram*), merancang skema relasi antar-tabel (*Entity Relationship Diagram* / ERD), serta membangun purwarupa antarmuka pengguna (*User Interface*) dengan bantuan kakas Figma.
4. **Construction (Konstruksi):** Tahap penulisan kode program (*coding*). Penulis mengimplementasikan desain antarmuka ke dalam baris kode menggunakan *Framework* Next.js dan Tailwind CSS. Tahap ini juga mencakup integrasi sistem WhatsApp *Click to Chat* untuk standardisasi format formulir dan implementasi Prisma ORM untuk operasi basis data.
5. **Deployment (Penyerahan & Evaluasi):** Tahap terakhir yang mencakup pengujian (*testing*) dan peluncuran sistem. Sistem dievaluasi menggunakan teknik *Black-Box Testing* guna memverifikasi bahwa alur reservasi, fungsi penyaringan katalog, dan manajemen data berjalan sesuai spesifikasi. Setelah divalidasi, aplikasi akan dipublikasikan (*deployment*) ke *server cloud* menggunakan platform Vercel agar dapat diakses penuh oleh pengguna.

BAB 4

ANALISIS DAN PERANCANGAN

Bab ini menjelaskan hasil analisis dan perancangan sistem *website* rental mobil Rahayu Trans Travel Bandung. Pembahasan disusun berdasarkan kondisi operasional yang berjalan, kebutuhan pengguna, kebutuhan perangkat lunak, rancangan proses bisnis, rancangan basis data, serta rancangan antarmuka sistem. Perancangan pada bab ini mengacu pada lingkup tugas akhir, yaitu membangun *website* rental mobil berbasis Next.js yang menyediakan katalog armada, formulir reservasi *online* terintegrasi WhatsApp, pengecekan ketersediaan kendaraan, serta *dashboard* admin untuk mengelola data operasional.

4.1 Analisis Sistem

Analisis sistem dilakukan untuk memahami proses bisnis penyewaan kendaraan yang saat ini berjalan di Rahayu Trans Travel, mengidentifikasi kendala utama, serta merumuskan kebutuhan sistem yang akan dibangun. Hasil analisis ini menjadi dasar dalam menentukan fitur, alur kerja, struktur data, dan tampilan antarmuka yang sesuai dengan kebutuhan pengguna.

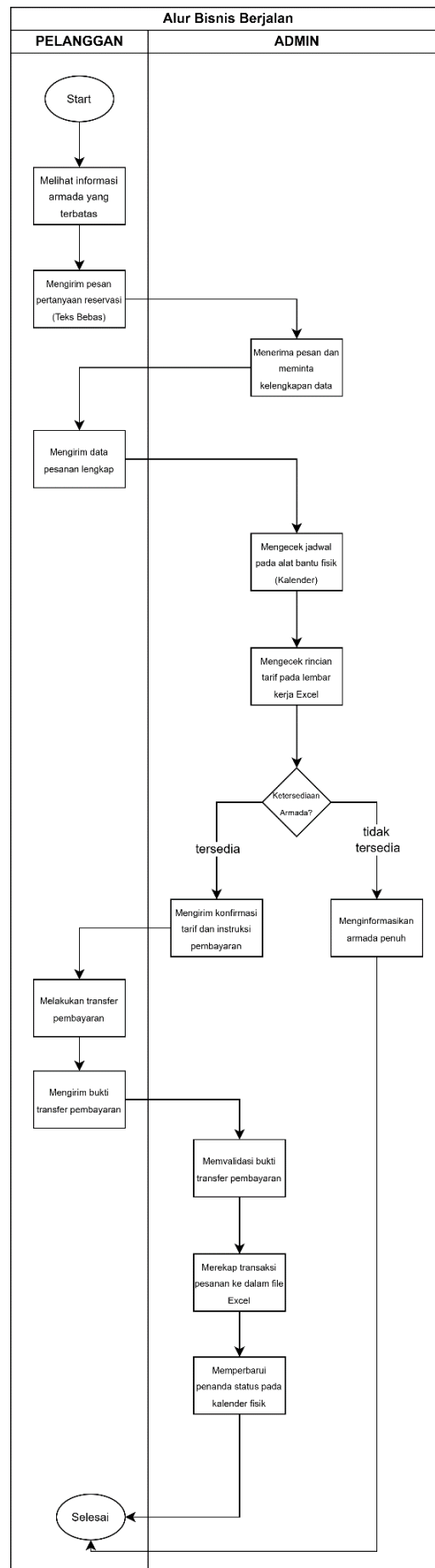
4.1.1 Analisis Alur Bisnis yang Berjalan

Berdasarkan hasil observasi dan wawancara, proses operasional penyewaan kendaraan di Rahayu Trans Travel masih mengandalkan media konvensional yang tidak terintegrasi. *Website* yang tersedia sebelumnya bersifat statis, sehingga informasi armada, harga, dan ketersediaan kendaraan belum dapat diperbarui secara mandiri oleh admin melalui *dashboard*. Proses komunikasi utama masih bergantung pada WhatsApp dengan format teks bebas, sedangkan pencatatan jadwal dan transaksi dilakukan secara terpisah melalui kalender fisik dan lembar kerja Microsoft Excel.

Alur bisnis yang berjalan saat ini adalah sebagai berikut:

1. Pelanggan membuka *website* atau media informasi Rahayu Trans Travel untuk melihat informasi umum layanan.
2. Pelanggan menghubungi admin melalui WhatsApp untuk menanyakan kendaraan, harga, dan ketersediaan jadwal.
3. Admin menanyakan kembali data yang belum lengkap, seperti nama pemesan, nomor HP, tanggal mulai, tanggal selesai, tujuan, dan pilihan armada.
4. Admin mengecek ketersediaan kendaraan menggunakan alat bantu fisik berupa kalender secara terpisah.
5. Jika armada tersedia, admin mengonfirmasi harga, ketentuan sewa, dan instruksi mekanisme pembayaran melalui pesan teks balasan.
6. Admin mencatat data pemesanan ke dalam lembar kerja Excel.
7. Apabila terjadi perubahan jadwal atau status pembayaran, admin harus memperbarui penanda status pada kalender fisik dan dokumen rekapitulasi secara bertahap.

Alur tersebut masih dapat berjalan, tetapi memiliki beberapa kelemahan. Pelanggan harus bertanya langsung untuk mendapatkan informasi harga dan ketersediaan. Admin juga harus melakukan tanya jawab berulang karena format data pemesanan dari pelanggan tidak baku (*free text*). Selain itu, pengarsipan data yang terputus dari sistem utama berisiko menimbulkan keterlambatan pembaruan data, kesalahan rekapitulasi, dan potensi pesanan ganda (*double booking*) pada tanggal yang sama.



Gambar 4.1 Flowmap Alur Bisnis yang Berjalan

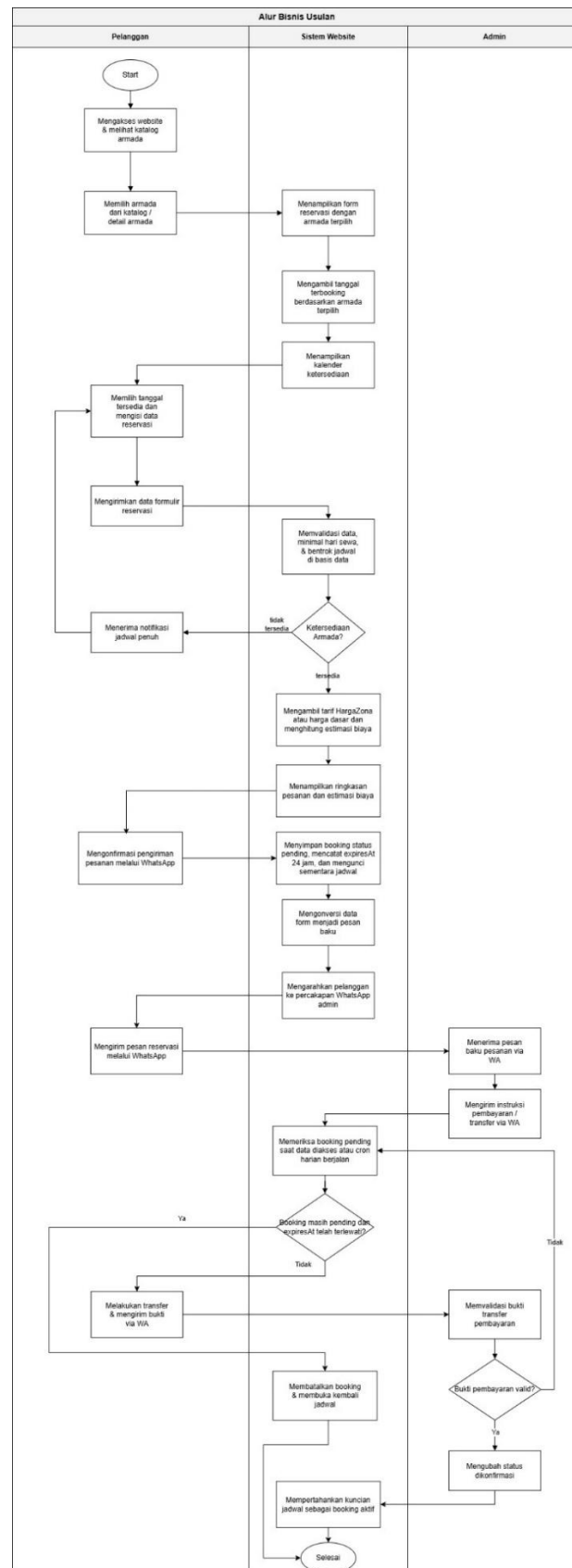
4.1.2 Analisis Alur Bisnis Usulan

Sistem informasi usulan dirancang sebagai aplikasi web dinamis berbasis *framework* Next.js dengan dukungan basis data relasional PostgreSQL melalui Prisma ORM. Sistem ini menyediakan antarmuka publik untuk pelanggan serta panel *dashboard* khusus untuk admin selaku pengelola. Pelanggan dapat mengakses katalog armada, memeriksa detail spesifikasi kendaraan, melihat transparansi harga, mengisi formulir reservasi terstruktur, serta melacak status pemesanan secara mandiri. Di sisi lain, admin dapat melakukan autentikasi hak akses untuk mengelola data armada, zona wilayah tujuan, tarif sewa, testimoni, serta memantau jadwal operasional melalui kalender web terpusat.

Alur bisnis usulan dirancang berjalan sebagai berikut:

1. Pelanggan mengakses aplikasi web Rahayu Trans Travel.
2. Pelanggan memeriksa katalog armada dan dapat menggunakan fitur penyaringan (filter) berdasarkan kategori atau kapasitas kendaraan.
3. Pelanggan membuka halaman detail armada untuk melihat dokumentasi foto, spesifikasi teknis, rincian harga sewa, dan melanjutkan ke formulir reservasi.
4. Pelanggan mengisi formulir reservasi terstruktur yang memuat data nama, nomor telepon, armada terpilih, rentang tanggal sewa, zona wilayah tujuan, dan keterangan tambahan.
5. Sistem *website* melakukan validasi data masukan secara otomatis, memeriksa ketersediaan tanggal pada basis data untuk mencegah pesanan ganda (*double booking*), serta memverifikasi batasan minimal hari sewa berdasarkan ketentuan zona tujuan.
6. Sistem menampilkan ringkasan pesanan dan menghitung estimasi biaya berdasarkan tarif kombinasi armada dan zona tujuan serta durasi sewa. Apabila tarif zona belum tersedia, sistem menggunakan harga dasar armada.
7. Setelah ringkasan pesanan sesuai, pelanggan mengonfirmasi pengiriman pesanan melalui WhatsApp.
8. Sistem menyimpan *booking* dengan status *pending*, mencatat *expiresAt* selama 24 jam, membuat notifikasi admin, mengonversi data reservasi menjadi pesan baku, dan mengarahkan pelanggan ke percakapan WhatsApp admin.
9. Pelanggan mengirimkan pesan reservasi melalui WhatsApp. Admin menerima pesan tersebut dan mengirimkan instruksi pembayaran kepada pelanggan.
10. Pelanggan melakukan transfer dan mengirimkan bukti pembayaran melalui WhatsApp sebelum batas waktu *expiresAt*.
11. Admin memvalidasi bukti pembayaran. Apabila pembayaran valid, admin memperbarui status *booking* menjadi dikonfirmasi dan memperbarui status pembayaran menjadi *booking*, DP, lunas, atau penuh sesuai pembayaran yang diterima.
12. Sistem memeriksa *booking* berstatus *pending* yang melewati *expiresAt* ketika data *booking* atau ketersediaan armada diakses. Sistem juga menjalankan pemeriksaan melalui *cron* harian sebagai proses cadangan. *Booking* yang telah kedaluwarsa diubah menjadi dibatalkan agar tanggal armada tersedia kembali.

13. Pelanggan dapat memeriksa pembaruan status transaksi melalui halaman lacak reservasi menggunakan nomor telepon dan PIN verifikasi yang valid.



Gambar 4.2 Flowmap Alur Bisnis Usulan

4.1.3 Analisis Kebutuhan Pengguna (*User Requirement*)

Kebutuhan pengguna menjelaskan hal yang dibutuhkan oleh setiap aktor agar proses penyewaan kendaraan dapat dilakukan lebih efisien dan terstruktur. Kebutuhan pengguna dideskripsikan pada Tabel 4.1 berikut:

Tabel 4.1 Analisis Kebutuhan Pengguna

Kode	Deskripsi Kebutuhan Pengguna (<i>User Requirement</i>)
UR-01	Pelanggan ingin dapat melihat daftar armada beserta kategori, kapasitas, transmisi, tahun, foto, dan harga sewa.
UR-02	Pelanggan ingin dapat melakukan pencarian, penyaringan (filter), atau perbandingan armada sebelum melakukan pemesanan.
UR-03	Pelanggan ingin dapat melihat detail armada, galeri foto, spesifikasi, dan informasi harga.
UR-04	Pelanggan ingin dapat mengisi formulir reservasi dengan format data yang jelas dan tervalidasi.
UR-05	Pelanggan ingin dapat mengetahui tanggal yang tidak tersedia berdasarkan armada yang dipilih agar tidak memilih jadwal yang bertabrakan dengan <i>booking</i> yang belum berstatus dibatalkan.
UR-06	Pelanggan ingin melihat ringkasan pesanan dan estimasi biaya sewa berdasarkan armada, zona tujuan, dan durasi sewa sebelum mengirimkan pesanan melalui WhatsApp.
UR-07	Pelanggan ingin dapat mengirim data reservasi ke WhatsApp admin dalam format pesan yang rapi.
UR-08	Pelanggan ingin dapat mengecek status reservasi secara mandiri tanpa harus selalu menanyakan ulang ke admin.
UR-09	Pelanggan ingin dapat memberikan ulasan dan membagikan pengalaman perjalanan melalui halaman lacak pesanan setelah pesanan berstatus selesai.
UR-10	Admin ingin dapat melakukan autentikasi (<i>login</i>) ke <i>dashboard</i> dengan akun yang aman.
UR-11	Admin ingin dapat mengelola data armada, termasuk foto, kategori, kapasitas, harga, transmisi, tahun, deskripsi, dan status ketersediaan.
UR-12	Admin ingin dapat mengatur zona wilayah, minimal hari sewa, dan harga sewa per zona.
UR-13	Admin ingin dapat melihat daftar reservasi, memfilter status, memperbarui status pesanan, memperbarui status pembayaran, serta memastikan <i>booking pending</i> yang tidak dikonfirmasi melalui WhatsApp dalam waktu 24 jam dapat dibatalkan agar tidak mengunci jadwal armada terlalu lama.
UR-14	Admin ingin dapat mengelola testimoni pelanggan dan pengaturan situs.
UR-15	Admin ingin dapat menerima notifikasi ketika ada pesanan baru yang masuk.
UR-16	Admin ingin dapat mengekspor data armada dan riwayat reservasi ke Excel untuk kebutuhan rekapitulasi.

4.1.4 Analisis Kebutuhan Perangkat Lunak (*Software Requirement*)

Software requirement adalah deskripsi terperinci mengenai fungsi spesifik yang harus disediakan oleh perangkat lunak untuk memenuhi kebutuhan pengguna (UR). Identifikasi kebutuhan fungsional perangkat lunak disajikan pada tabel 4.2 berikut:

Tabel 4.2 Analisis Kebutuhan Perangkat Lunak

Kode UR	User Requirement	Kode SR	Software Requirement
UR-01	Pelanggan ingin dapat melihat daftar armada beserta kategori, kapasitas, transmisi, tahun, foto, dan harga sewa.	SR-01	Sistem harus menampilkan halaman beranda yang berisi profil singkat, armada unggulan, alur pemesanan, dan testimoni.
		SR-02	Sistem harus menampilkan katalog armada dengan informasi foto, nama, kategori, kapasitas, transmisi, dan harga.
UR-02	Pelanggan ingin dapat melakukan pencarian, penyaringan (filter), atau perbandingan armada sebelum melakukan pemesanan.	SR-03	Sistem harus menyediakan fitur filter dan <i>sorting</i> pada katalog armada.
UR-03	Pelanggan ingin dapat melihat detail armada, galeri foto, spesifikasi, dan informasi harga.	SR-04	Sistem harus menyediakan halaman detail armada berdasarkan slug kendaraan.
		SR-05	Sistem harus menyediakan halaman harga dan informasi syarat ketentuan sewa.
		SR-06	Sistem harus menyediakan halaman galeri foto armada.
UR-04	Pelanggan ingin dapat mengisi formulir reservasi dengan format data yang jelas dan tervalidasi.	SR-07	Sistem harus menyediakan formulir reservasi dengan validasi nama, nomor HP, armada, tanggal mulai, tanggal selesai, zona tujuan, dan keterangan.

Kode UR	User Requirement	Kode SR	Software Requirement
		SR-08	Sistem harus menyimpan <i>booking</i> setelah pelanggan mengonfirmasi pengiriman pesanan melalui WhatsApp dengan status awal <i>pending</i> dan mencatat <i>expiresAt</i> selama 24 jam.
UR-05	Pelanggan ingin dapat mengetahui tanggal yang tidak tersedia berdasarkan armada yang dipilih agar tidak memilih jadwal yang bertabrakan dengan <i>booking</i> yang belum berstatus dibatalkan.	SR-09	Sistem harus mengambil data tanggal yang sudah dipesan berdasarkan armada yang dipilih dan hanya menghitung <i>booking</i> yang belum berstatus dibatalkan untuk mencegah pemilihan jadwal yang bertabrakan.
UR-06	Pelanggan ingin dapat melihat ringkasan pesanan dan estimasi biaya sewa berdasarkan armada, zona tujuan, dan durasi sewa sebelum mengirimkan pesanan melalui WhatsApp.	SR-10	Sistem harus mengambil tarif berdasarkan kombinasi armada dan zona tujuan dari data HargaZona untuk menghitung estimasi biaya sesuai durasi sewa. Apabila tarif HargaZona belum tersedia, sistem menggunakan harga dasar armada sebagai tarif cadangan.
UR-07	Pelanggan ingin dapat mengirim data reservasi ke WhatsApp admin dalam format pesan yang rapi.	SR-11	Sistem harus membuat format pesan WhatsApp otomatis berdasarkan data reservasi pelanggan.
UR-08	Pelanggan ingin dapat mengecek status reservasi secara mandiri tanpa harus selalu menanyakan ulang ke admin.	SR-12	Sistem harus menyediakan halaman cek pesanan berdasarkan nomor HP dan PIN verifikasi.
UR-09	Pelanggan ingin dapat memberikan ulasan dan membagikan pengalaman perjalanan melalui halaman lacak pesanan setelah pesanan berstatus selesai.	SR-13	Sistem harus menyediakan formulir pengisian testimoni pada halaman cek pesanan yang hanya dapat diakses apabila status operasional pesanan telah selesai.
UR-10	Admin ingin dapat melakukan autentikasi (<i>login</i>) ke <i>dashboard</i> dengan akun yang aman.	SR-14	Sistem harus menyediakan fitur autentikasi admin menggunakan <i>username</i> dan <i>password</i> .
		SR-15	Sistem harus melindungi halaman admin dari akses pengguna yang belum melakukan <i>login</i> .
		SR-16	Sistem harus menyediakan <i>dashboard</i> statistik admin.
UR-11	Admin ingin dapat mengelola data armada, termasuk foto, kategori, kapasitas, harga, transmisi, tahun, deskripsi, dan status ketersediaan.	SR-17	Sistem harus menyediakan fungsi pengelolaan (CRUD) armada, termasuk unggah foto kendaraan.
UR-12	Admin ingin dapat mengatur zona wilayah, minimal hari sewa, dan harga sewa per zona.	SR-18	Sistem harus menyediakan pengaturan harga armada per zona wilayah.
		SR-19	Sistem harus menyediakan fungsi pengelolaan (CRUD) zona wilayah.
UR-13	Admin ingin dapat melihat daftar reservasi, memfilter status, memperbarui status pesanan, memperbarui status pembayaran, serta memastikan <i>booking pending</i> yang tidak dikonfirmasi melalui WhatsApp dalam waktu 24 jam dapat dibatalkan agar tidak mengunci jadwal armada terlalu lama.	SR-20	Sistem harus memeriksa <i>booking</i> berstatus <i>pending</i> yang melewati <i>expiresAt</i> ketika data <i>booking</i> atau ketersediaan armada diakses serta melalui <i>cron</i> harian sebagai pemeriksaan cadangan, kemudian mengubah <i>booking</i> tersebut menjadi dibatalkan agar jadwal armada tersedia kembali.
UR-14	Admin ingin dapat mengelola testimoni pelanggan dan pengaturan situs.	SR-21	Sistem harus menyediakan fungsi moderasi (persetujuan, pembatalan, dan penghapusan) terhadap testimoni yang dikirimkan oleh pelanggan.
		SR-22	Sistem harus menyediakan pengaturan situs, seperti nomor WhatsApp admin dan informasi kontak.
UR-15	Admin ingin dapat menerima notifikasi ketika ada pesanan baru yang masuk.	SR-23	Sistem harus membuat notifikasi admin ketika ada pesanan baru masuk.
UR-16	Admin ingin dapat mengekspor data armada dan riwayat reservasi ke Excel untuk kebutuhan rekapitulasi.	SR-24	Sistem harus menyediakan fungsi ekspor data armada dan riwayat reservasi ke file Excel.

4.1.5 Analisis Kebutuhan Non-Fungsional

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

Tabel 4.3 Analisis Kebutuhan Non-Fungsional

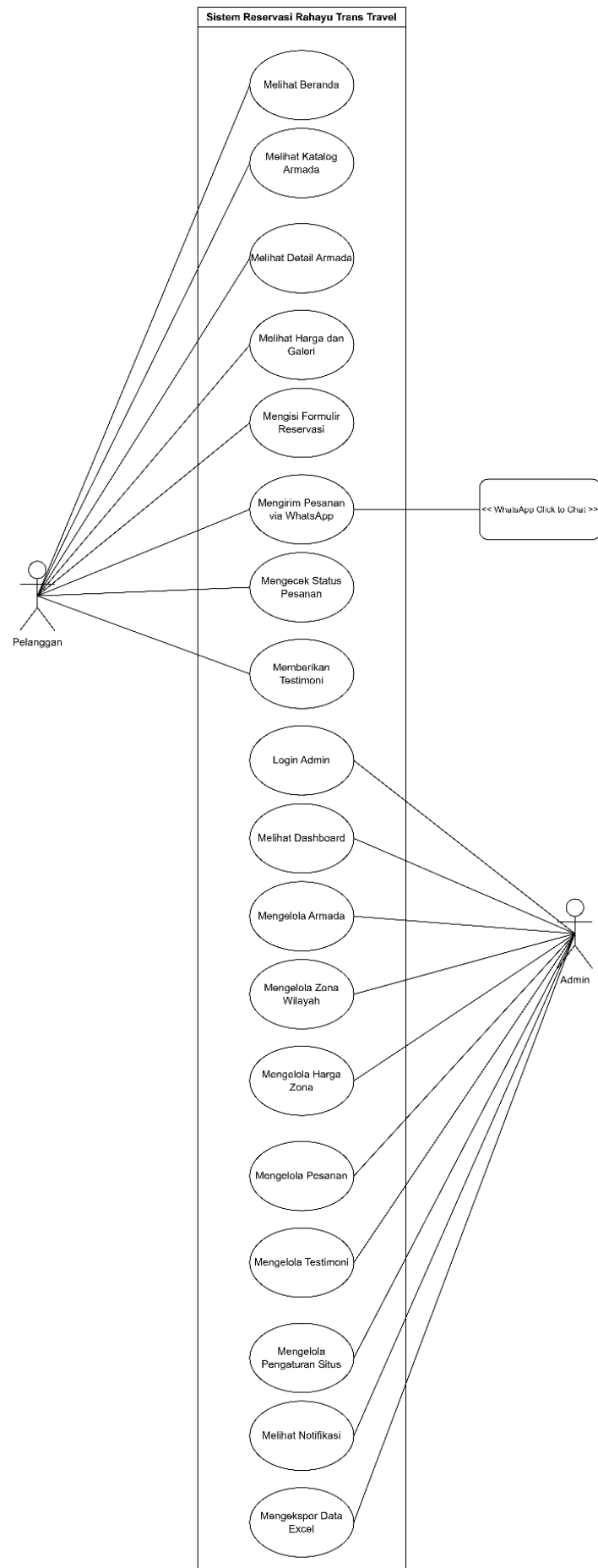
Kode	Kategori	Deskripsi Kebutuhan
NFR-01	Usability	Antarmuka sistem harus mudah digunakan dan dipahami oleh pelanggan awam maupun admin operasional.
NFR-02	Responsiveness	Tampilan antarmuka harus dapat menyesuaikan secara optimal pada berbagai ukuran layar (<i>desktop</i> , <i>tablet</i> , maupun <i>mobile</i>).
NFR-03	Security	Halaman admin harus dilindungi oleh proses autentikasi. Kata sandi (<i>password</i>) admin harus disimpan di dalam basis data menggunakan metode <i>hashing</i> .
NFR-04	Data Integrity	Data transaksi reservasi harus terhubung dengan data entitas armada melalui relasi basis data yang kuat untuk mencegah redundansi data.
NFR-05	Reliability	Sistem harus mampu secara otomatis mencegah pemilihan rentang tanggal yang bertabrakan dengan jadwal <i>booking</i> yang belum berstatus dibatalkan.
NFR-06	Maintainability	Struktur kode sumber harus dipisahkan secara modular antara komponen halaman (<i>page</i>), jalur sistem (<i>API route</i>), akses data (<i>service</i>), dan skema basis data.
NFR-07	Performance	Aplikasi web harus memanfaatkan kemampuan <i>rendering</i> dan optimasi gambar Next.js untuk mendukung pemuatan halaman yang efisien.
NFR-08	Portability	Sistem harus dapat dijalankan secara optimal pada lingkungan <i>runtime</i> Node.js dan basis data PostgreSQL.

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 Reservasi Rahayu Trans Travel.



Gambar 4.3 Use Case Diagram

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

Tabel 4.4 Deskripsi Aktor pada *Use Case Diagram*

No	Nama Aktor	Deskripsi
1	Pelanggan	Pengguna publik yang mengakses <i>website</i> untuk mencari informasi kendaraan, mengisi formulir reservasi, dan mengecek status pesanan secara mandiri.
2	Admin	Pengelola Rahayu Trans Travel yang melakukan autentikasi (<i>login</i>) ke <i>dashboard</i> untuk mengelola data armada, zona wilayah, tarif zona, memantau pesanan, serta mengelola testimoni.
3	WhatsApp <i>Click to Chat</i>	Sistem eksternal yang bertugas menerima parameter format pesan baku dari <i>website</i> dan membuka ruang obrolan (<i>chat</i>) untuk diteruskan ke admin.

Tabel 4.5 Deskripsi *Use Case Diagram*

No	Kode	Nama Use Case	Deskripsi
1	UC-01	Melihat Beranda	<i>Use case</i> yang digunakan oleh pelanggan untuk melihat informasi umum layanan, <i>form</i> pencarian armada, armada unggulan, panduan pemesanan, dan testimoni.
2	UC-02	Melihat Katalog Armada	<i>Use case</i> yang digunakan oleh pelanggan untuk melihat daftar armada dan melakukan pencarian atau penyaringan berdasarkan tanggal berangkat-sampai, tujuan, jumlah penumpang, kategori, serta kapasitas kendaraan sebelum melanjutkan ke proses reservasi.
3	UC-03	Melihat Detail Armada	<i>Use case</i> yang digunakan oleh pelanggan untuk mengeksplorasi foto, spesifikasi, harga, dan mengakses tombol reservasi pada satu kendaraan.
4	UC-04	Melihat Harga dan Galeri	<i>Use case</i> yang digunakan oleh pelanggan untuk melihat daftar harga, syarat ketentuan, dan dokumentasi foto armada.
5	UC-05	Mengisi Formulir Reservasi	<i>Use case</i> yang digunakan oleh pelanggan untuk mengisi data pemesanan dan melihat ringkasan pesanan beserta kalkulasi biaya.
6	UC-06	Mengirim Pesanan via WhatsApp	<i>Use case</i> yang berjalan saat pelanggan selesai mengisi formulir, di mana sistem menyimpan data dan mengarahkan ke WhatsApp Click to Chat (https://api.whatsapp.com/send) dengan format pesan otomatis.
7	UC-07	Memeriksa Status Pesanan	<i>Use case</i> yang digunakan oleh pelanggan untuk melacak status transaksinya menggunakan nomor telepon dan PIN verifikasi.
8	UC-08	Memberikan Testimoni	<i>Use case</i> yang digunakan oleh pelanggan untuk mengirimkan ulasan perjalanan melalui halaman lacak pesanan khusus untuk transaksi yang telah selesai.
9	UC-09	<i>Login Admin</i>	<i>Use case</i> yang digunakan oleh admin untuk melakukan autentikasi ke dalam sistem <i>dashboard</i> menggunakan <i>username</i> dan <i>password</i> .
10	UC-10	Melihat <i>Dashboard</i>	<i>Use case</i> yang digunakan oleh admin untuk memantau ringkasan data operasional secara keseluruhan.
11	UC-11	Mengelola Armada	<i>Use case</i> yang digunakan oleh admin untuk menambah, mengubah, menghapus, mengunggah foto, dan memperbarui status kendaraan.
12	UC-12	Mengelola Zona Wilayah	<i>Use case</i> yang digunakan oleh admin untuk mengatur nama zona, minimal hari sewa, dan urutan tampil zona tujuan.
13	UC-13	Mengelola Harga Zona	<i>Use case</i> yang digunakan oleh admin untuk menetapkan harga sewa per hari berdasarkan kombinasi armada dan zona wilayah.
14	UC-14	Mengelola Pesanan	<i>Use case</i> yang digunakan oleh admin untuk melihat, memfilter, serta memperbarui status transaksi dan verifikasi pembayaran.
15	UC-15	Mengelola Testimoni	<i>Use case</i> yang digunakan oleh admin untuk memoderasi (menyetujui atau menghapus) ulasan yang masuk dari pelanggan sebelum ditampilkan di halaman publik.
16	UC-16	Mengelola Pengaturan Situs	<i>Use case</i> yang digunakan oleh admin untuk memperbarui informasi kontak dan nomor tujuan WhatsApp utama.
17	UC-17	Melihat Notifikasi	<i>Use case</i> yang digunakan oleh admin untuk menerima peringatan ketika terdapat transaksi pesanan baru masuk ke dalam sistem.
18	UC-18	Mengekspor Data Excel	<i>Use case</i> yang digunakan oleh admin untuk mengunduh rekapitulasi data armada maupun riwayat transaksi ke dalam format <i>file</i> Excel.

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 *website* reservasi rental mobil di Rahayu Trans Travel Bandung:

Tabel 4.6 Skenario UC-01: Melihat Beranda

Identifikasi		
Use Case ID	UC-01	
Nama Use Case	Melihat Beranda	
Aktor	Pelanggan	
Prekondisi	Pelanggan mengakses website Rahayu Trans Travel.	
Postkondisi	Informasi utama layanan tampil pada halaman beranda.	
Tujuan	Memberikan gambaran awal mengenai layanan rental mobil, armada unggulan, alur pemesanan, dan testimoni.	
Skenario Utama		
Kondisi Awal: Halaman Beranda		
No	Aktor	Sistem
1	Pelanggan membuka alamat website.	Sistem menampilkan halaman beranda.
2	Pelanggan melihat profil singkat layanan.	Sistem menampilkan informasi Rahayu Trans Travel.
3	Pelanggan melihat armada unggulan.	Sistem mengambil dan menampilkan data armada yang aktif.
4	Pelanggan melihat alur pemesanan dan testimoni.	Sistem menampilkan konten informasi layanan dan testimoni pelanggan.
5	Pelanggan memilih menu atau tombol menuju halaman lain.	Sistem mengarahkan pelanggan ke halaman yang dipilih.
Kondisi Akhir: Halaman Beranda tampil dengan data aktif.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Data armada unggulan belum tersedia.	Sistem tetap menampilkan informasi beranda tanpa daftar armada unggulan.
4a	Data testimoni belum tersedia.	Sistem menyembunyikan atau menampilkan bagian testimoni dalam kondisi kosong.

Tabel 4.7 Skenario UC-02: Melihat Katalog Armada

Identifikasi		
Use Case ID	UC-02	
Nama Use Case	Melihat Katalog Armada	
Aktor	Pelanggan	
Prekondisi	Pelanggan berada pada <i>website</i> dan memilih menu armada.	
Postkondisi	Daftar armada tampil sesuai data dan filter yang dipilih.	
Tujuan	Memudahkan pelanggan mencari kendaraan berdasarkan tanggal sewa, tujuan, jumlah penumpang, kategori, kapasitas, atau harga.	
Skenario Utama		
Kondisi Awal: Halaman Katalog Armada		
No	Aktor	Sistem
1	Pelanggan membuka halaman armada.	Sistem mengambil daftar armada dari <i>database</i> .
2		Sistem menampilkan foto, nama, kategori, kapasitas, transmisi, dan harga.
3	Pelanggan mengisi tanggal berangkat-sampai, tujuan, jumlah penumpang, atau memilih filter armada.	Sistem memperbarui daftar armada sesuai parameter pencarian dan filter yang dipilih.
4	Pelanggan memilih pengurutan data.	Sistem mengurutkan armada berdasarkan pilihan pelanggan.
5	Pelanggan memilih salah satu armada.	Sistem mengarahkan pelanggan ke halaman detail armada.
Kondisi Akhir: Pelanggan memperoleh daftar armada yang sesuai atau masuk ke halaman detail kendaraan.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Data armada tidak ditemukan.	Sistem menampilkan pesan peringatan bahwa armada belum tersedia.
3a	Filter tidak menghasilkan data.	Sistem menampilkan pesan hasil pencarian kosong dan menyediakan opsi menghapus filter.

Tabel 4.8 Skenario UC-03: Melihat Detail Armada

Identifikasi		
Use Case ID	UC-03	
Nama Use Case	Melihat Detail Armada	
Aktor	Pelanggan	
Prekondisi	Pelanggan memilih salah satu armada dari katalog.	
Postkondisi	Detail kendaraan tampil dan pelanggan dapat melanjutkan ke reservasi.	
Tujuan	Menampilkan informasi lengkap kendaraan sebelum pelanggan melakukan reservasi.	
Skenario Utama		
Kondisi Awal: Halaman Detail Armada		
No	Aktor	Sistem
1	Pelanggan membuka halaman detail armada.	Sistem mengambil data armada berdasarkan slug kendaraan.
2	Pelanggan melihat foto, spesifikasi, harga, dan ketentuan sewa.	Sistem menampilkan galeri, kapasitas, transmisi, tahun, harga, dan deskripsi.
		Sistem menampilkan harga dasar dan informasi terkait sewa.
3	Pelanggan menekan tombol "Reservasi".	Sistem mengarahkan pelanggan ke formulir reservasi dengan armada terpilih.
Kondisi Akhir: Halaman Detail Armada menampilkan informasi kendaraan secara lengkap.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Slug armada tidak ditemukan.	Sistem mengarahkan ke halaman tidak ditemukan (404 Not Found).
4a	Armada tidak aktif.	Sistem menyembunyikan tombol reservasi atau memberi keterangan tidak tersedia.

Tabel 4.9 Skenario UC-04: Melihat Harga dan Galeri

Identifikasi		
Use Case ID	UC-04	
Nama Use Case	Melihat Harga dan Galeri	
Aktor	Pelanggan	
Prekondisi	Pelanggan memilih menu harga atau galeri.	
Postkondisi	Informasi harga, ketentuan sewa, atau dokumentasi foto tampil.	
Tujuan	Memberikan informasi pendukung sebelum pelanggan melakukan pemesanan.	
Skenario Utama		
Kondisi Awal: Halaman Harga atau Galeri.		
No	Aktor	Sistem
1	Pelanggan membuka halaman harga.	Sistem menampilkan daftar harga dan ketentuan sewa.
2	Pelanggan membuka halaman galeri.	Sistem menampilkan dokumentasi foto armada atau layanan.
3	Pelanggan memilih tombol reservasi atau armada.	Sistem mengarahkan pelanggan ke halaman terkait.
Kondisi Akhir: Halaman Harga atau Halaman Galeri berhasil dirender oleh sistem.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Data harga belum lengkap.	Sistem menampilkan harga yang tersedia dan informasi untuk menghubungi admin.
2a	Data galeri belum tersedia.	Sistem menampilkan pesan bahwa dokumentasi belum tersedia.

Tabel 4.10 Skenario UC-05: Mengisi Formulir Reservasi

Identifikasi		
Use Case ID	UC-05	
Nama Use Case	Mengisi Formulir Reservasi	
Aktor	Pelanggan	
Prekondisi	Pelanggan berada pada halaman pesanan.	
Postkondisi	Data pesanan tervalidasi dan ringkasan pesanan tampil.	
Tujuan	Menstandarisasi data pemesanan pelanggan sebelum diteruskan ke admin melalui WhatsApp.	
Skenario Utama		
Kondisi Awal: Halaman Formulir Reservasi.		
No	Aktor	Sistem
1	Pelanggan membuka formulir reservasi dari armada yang telah dipilih.	Sistem menampilkan armada terpilih dan mengambil tanggal yang telah dipesan.
2	Pelanggan memilih tanggal mulai dan selesai sewa.	Sistem menghitung durasi sewa.
3	Pelanggan memilih zona tujuan.	Sistem mengambil ketentuan minimal hari dan tarif HargaZona berdasarkan armada dan zona tujuan.
4	Pelanggan mengisi nama, nomor HP, dan keterangan tambahan.	Sistem memvalidasi kelengkapan serta format data.
5	Pelanggan menekan tombol lanjut ke ringkasan.	Sistem memeriksa bentrok jadwal dan minimal hari sewa.
6	Pelanggan membuka ringkasan pesanan.	Sistem menghitung estimasi berdasarkan tarif zona dan durasi, kemudian menampilkan ringkasan pesanan.
Kondisi Akhir: Halaman Formulir Reservasi menampilkan ringkasan pesanan dan estimasi biaya.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
4a	Data wajib belum lengkap	Sistem menampilkan pesan validasi pada field yang bermasalah.
3a	Tarif HargaZona belum tersedia.	Sistem menggunakan harga dasar armada sebagai tarif cadangan.
5a	Tanggal bertabrakan dengan booking yang belum berstatus dibatalkan	Sistem menolak tanggal tersebut dan meminta pelanggan memilih jadwal lain.
5b	Durasi sewa kurang dari minimal hari zona	Sistem menampilkan pesan bahwa durasi belum memenuhi aturan minimal.

Tabel 4.11 Skenario UC-06: Mengirim Pesanan via WhatsApp

Identifikasi		
Use Case ID	UC-06	
Nama Use Case	Mengirim Pesanan via WhatsApp	
Aktor	Pelanggan, WhatsApp <i>Click to Chat</i>	
Prekondisi	Pelanggan sudah melengkapi formulir reservasi dan membuka ringkasan pesanan.	
Postkondisi	Pesanan tersimpan dengan status <i>pending</i> , batas waktu expiresAt 24 jam tercatat, dan WhatsApp terbuka dengan pesan reservasi otomatis.	
Tujuan	Meneruskan data pesanan ke admin dalam format pesan yang rapi dan konsisten.	
Skenario Utama		
Kondisi Awal: Halaman Ringkasan Pesanan terbuka dan data tervalidasi		
No	Aktor	Sistem
1	Pelanggan mengonfirmasi pengiriman pesanan melalui WhatsApp.	Sistem mengirimkan data reservasi ke proses penyimpanan <i>booking</i> .
2	Pelanggan menunggu proses pengiriman.	Sistem menyimpan data pesanan ke <i>database</i> dengan status <i>pending</i> dan expiresAt 24 jam.
3		Sistem membuat notifikasi pesanan baru untuk admin.
4		Sistem menyusun pesan WhatsApp berdasarkan data pemesanan.
5		Sistem membuka URL <i>https://api.whatsapp.com/send</i> menuju nomor WhatsApp admin.
6	Pelanggan mengirim pesan pada aplikasi WhatsApp.	WhatsApp <i>Click to Chat</i> meneruskan pesan yang sudah disusun sistem ke nomor admin.
Kondisi Akhir Halaman aplikasi WhatsApp pelanggan terbuka dengan draf pesan otomatis.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	Data pesanan gagal tersimpan	Sistem menampilkan pesan kegagalan dan meminta pelanggan mencoba kembali sebelum membuka WhatsApp.
2b	<i>Booking</i> masih <i>pending</i> setelah melewati expiresAt	Sistem membatalkan <i>booking</i> ketika data <i>booking</i> atau ketersediaan diakses atau ketika <i>cron</i> harian berjalan, kemudian membuka kembali jadwal armada.
5a	Nomor WhatsApp admin belum sesuai	Sistem tetap membuka tautan berdasarkan pengaturan yang tersimpan, kemudian admin perlu memperbarui nomor pada pengaturan situs.

Tabel 4.12 Skenario UC-07: Mengecek Status Pesanan

Identifikasi		
Use Case ID	UC-07	
Nama Use Case	Mengecek Status Pesanan	
Aktor	Pelanggan	
Prekondisi	Pelanggan memiliki nomor HP yang digunakan saat pesanan dan PIN verifikasi.	
Postkondisi	Data pesanan yang cocok ditampilkan dengan pembatasan informasi sensitif.	
Tujuan	Memudahkan pelanggan memperoleh status reservasi tanpa harus mengulang pertanyaan ke admin.	
Skenario Utama		
Kondisi Awal: Halaman Cek Pesanan		
No	Aktor	Sistem
1	Pelanggan membuka halaman cek pesanan.	Sistem menampilkan <i>form</i> pencarian pesanan.
2	Pelanggan memasukkan nomor HP dan PIN.	Sistem mengirim data pencarian ke <i>endpoint lookup</i> .
3	Pelanggan menekan tombol cek status.	Sistem mencari pesanan berdasarkan nomor HP.
4		Sistem mencocokkan PIN dengan aturan verifikasi yang ditentukan.
5		Sistem menampilkan status pesanan, armada, tanggal sewa, dan status pembayaran yang relevan.
Kondisi Akhir: Halaman Cek Pesanan menampilkan informasi status transaksi pelanggan		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	Nomor HP atau PIN belum diisi	Sistem menampilkan pesan validasi.
4a	Nomor HP atau PIN tidak cocok	Sistem menampilkan informasi bahwa pesanan tidak ditemukan.
5a	Tidak ada data pesanan aktif	Sistem menampilkan pesan kosong dan meminta pelanggan memeriksa kembali <i>input</i> .

Tabel 4.13 Skenario UC-08: Memberikan Testimoni

Identifikasi		
Use Case ID	UC-08	
Nama Use Case	Memberikan Testimoni	
Aktor	Pelanggan	
Prekondisi	Pelanggan sedang melacak pesanan dan status pesannya sudah "Selesai".	
Postkondisi	Ulasan berhasil terkirim ke basis data dengan status "Menunggu" untuk dimoderasi admin.	
Tujuan	Menyediakan fasilitas bagi pelanggan untuk memberikan ulasan dan membagikan pengalaman perjalanan mereka.	
Skenario Utama		
Kondisi Awal: Halaman Lacak Pesanan		
No	Aktor	Sistem
1	Pelanggan menekan tombol "Tulis Ulasan Perjalanan"	Sistem menampilkan formulir pengisian ulasan (<i>rating</i> bintang dan teks)
2	Pelanggan mengisi <i>rating</i> dan teks ulasan, lalu menekan tombol kirim.	Sistem memvalidasi kelengkapan data masukan.
3		Sistem menyimpan data ulasan ke basis data dengan status awal "Menunggu"
4		Sistem memperbarui komponen menjadi pesan "Ulasan Terkirim" dan menampilkan <i>pop-up</i> notifikasi berhasil
Kondisi Akhir: Ulasan pelanggan berhasil tersimpan di sistem dan menunggu persetujuan admin.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Status pesanan belum selesai (contoh: Menunggu Konfirmasi).	Sistem mengunci fitur ulasan dan menampilkan pesan informasi bahwa formulir akan tersedia setelah perjalanan berstatus selesai
2a	Pelanggan menekan tombol kirim tetapi belum memberikan <i>rating</i> bintang atau teks ulasan	Sistem menampilkan pesan validasi kesalahan (<i>error</i>) dan menahan proses pengiriman data

Tabel 4.14 Skenario UC-09: *Login Admin*

Identifikasi		
Use Case ID	UC-09	
Nama Use Case	Login Admin	
Aktor	Admin	
Prekondisi	Admin memiliki akun dan membuka halaman <i>login</i> .	
Postkondisi	Admin berhasil masuk ke <i>dashboard</i> .	
Tujuan	Melindungi halaman admin dari pengguna yang tidak berwenang.	
Skenario Utama		
Kondisi Awal: Halaman <i>Login Admin</i>		
No	Aktor	Sistem
1	Admin membuka halaman <i>login</i> .	Sistem menampilkan <i>form username</i> dan <i>password</i> .
2	Admin mengisi <i>username</i> dan <i>password</i> .	Sistem memvalidasi kelengkapan data.
3	Admin menekan tombol masuk.	Sistem memeriksa kecocokan akun dan <i>hash password</i> .
4		Sistem membuat sesi <i>login</i> admin.
5		Sistem mengarahkan admin ke <i>dashboard</i> .
Kondisi Akhir: Halaman <i>Dashboard</i> utama tampil setelah sesi <i>login</i> berhasil		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	<i>Username</i> atau <i>password</i> belum diisi	Sistem menampilkan pesan validasi.
3a	<i>Username</i> atau <i>password</i> salah	Sistem menampilkan pesan <i>login</i> gagal.
1a	Admin sudah <i>login</i>	Sistem mengarahkan admin langsung ke <i>dashboard</i> .

Tabel 4.15 Skenario UC-10: *Melihat Dashboard*

Identifikasi		
Use Case ID	UC-10	
Nama Use Case	Melihat <i>Dashboard</i>	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> .	
Postkondisi	Ringkasan data operasional tampil pada <i>dashboard</i> .	
Tujuan	Memberikan gambaran cepat mengenai kondisi armada, pesanan, pembayaran, dan aktivitas terbaru.	
Skenario Utama		
Kondisi Awal: Halaman <i>Dashboard</i> utama terbuka		
No	Aktor	Sistem
1	Admin membuka <i>dashboard</i> .	Sistem memverifikasi sesi <i>login</i> admin.
2		Sistem mengambil data ringkasan armada, pesanan, dan status pembayaran.
3		Sistem menampilkan kartu statistik, grafik/ringkasan, dan aktivitas terbaru.
4	Admin memilih menu manajemen.	Sistem mengarahkan admin ke halaman yang dipilih.
Kondisi Akhir: Halaman <i>Dashboard</i> menampilkan ringkasan data operasional secara lengkap		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Admin belum <i>login</i>	Sistem mengarahkan admin ke halaman <i>login</i> .
2a	Data ringkasan gagal dimuat	Sistem menampilkan pesan kesalahan atau data kosong.

Tabel 4.16 Skenario UC-11: Mengelola Armada

Identifikasi		
Use Case ID	UC-11	
Nama Use Case	Mengelola Armada	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan membuka halaman armada.	
Postkondisi	Data armada tersimpan, diperbarui, dinonaktifkan, atau dihapus sesuai aksi admin.	
Tujuan	Memudahkan admin memperbarui data kendaraan yang tampil pada <i>website</i> .	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Armada		
No	Aktor	Sistem
1	Admin membuka halaman armada.	Sistem menampilkan daftar armada.
2	Admin menekan tombol tambah atau edit armada.	Sistem menampilkan <i>modal form</i> armada.
3	Admin mengisi data kendaraan dan mengunggah foto.	Sistem memvalidasi nama, kategori, kapasitas, harga, tahun, transmisi, deskripsi, status, dan <i>file</i> foto.
4	Admin menyimpan data.	Sistem menyimpan data melalui <i>endpoint</i> armada.
5	Admin memilih aksi hapus atau nonaktifkan armada.	Sistem meminta konfirmasi dan memproses aksi jika disetujui.
Kondisi Akhir: Halaman Manajemen Armada menampilkan daftar kendaraan yang telah diperbarui.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Data wajib tidak lengkap	Sistem menolak penyimpanan dan menampilkan pesan validasi.
5a	Armada masih terkait data pesanan aktif	Sistem mencegah penghapusan dan meminta admin menonaktifkan armada apabila tidak ingin ditampilkan.

Tabel 4.17 Skenario UC-12: Mengelola Zona Wilayah

Identifikasi		
Use Case ID	UC-12	
Nama Use Case	Mengelola Zona Wilayah	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan membuka halaman zona wilayah.	
Postkondisi	Data zona wilayah tersimpan, diperbarui, atau dihapus.	
Tujuan	Mengatur pembagian wilayah tujuan dan minimal hari sewa yang digunakan pada formulir reservasi.	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Zona Wilayah		
No	Aktor	Sistem
1	Admin membuka halaman zona wilayah.	Sistem menampilkan daftar zona.
2	Admin menambah atau mengubah zona.	Sistem menampilkan <i>form</i> zona wilayah.
3	Admin mengisi nama zona, minimal hari, dan urutan tampil.	Sistem memvalidasi data masukan.
4	Admin menekan tombol simpan.	Sistem menyimpan perubahan zona ke <i>database</i> dan memperbarui daftar zona.
5	Admin memilih opsi hapus pada zona tertentu.	Sistem memunculkan dialog konfirmasi penghapusan.
6	Admin menyetujui konfirmasi penghapusan.	Sistem menghapus baris data zona dari <i>database</i> dan memperbarui daftar zona.
Kondisi Akhir: Halaman Manajemen Zona Wilayah menampilkan daftar zona yang telah diperbarui		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
3a	Minimal hari tidak valid	Sistem menolak penyimpanan dan menampilkan pesan validasi.
5a	Zona masih digunakan pada harga zona atau pesanan	Sistem mencegah penghapusan data yang masih berelasi.
6a	Admin membatalkan konfirmasi penghapusan	Sistem menutup dialog konfirmasi dan membatalkan proses hapus data.

Tabel 4.18 Skenario UC-13: Mengelola Harga Zona

Identifikasi		
Use Case ID	UC-13	
Nama Use Case	Mengelola Harga Zona	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> , data armada tersedia, dan data zona wilayah tersedia.	
Postkondisi	Harga sewa per kombinasi armada dan zona berhasil disimpan.	
Tujuan	Menentukan tarif sewa berdasarkan kendaraan dan wilayah tujuan.	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Armada		
No	Aktor	Sistem
1	Admin menekan tombol harga zona pada salah satu baris armada.	Sistem mengambil data zona wilayah beserta harga yang sudah tersimpan untuk armada tersebut.
2		Sistem menampilkan jendela <i>pop-up</i> formulir Harga per Zona.
3	Admin mengisi atau mengubah nominal harga sewa harian pada kolom zona yang tersedia.	Sistem memvalidasi format masukan angka.
4	Admin menekan tombol simpan harga.	Sistem menyimpan pembaruan tarif seluruh zona ke tabel terkait di basis data.
5		Sistem menutup jendela <i>pop-up</i> dan merender notifikasi sukses.
Kondisi Akhir: Halaman Manajemen Armada tetap terbuka dan harga zona kendaraan terkait telah diperbarui.		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Data zona wilayah belum dikonfigurasi	Sistem menampilkan <i>pop-up</i> peringatan untuk mengatur zona wilayah terlebih dahulu.
3a	Input nominal harga tidak valid atau menggunakan huruf	Sistem menolak proses penyimpanan dan mempertahankan tampilan <i>pop-up</i> beserta pesan validasi kesalahan.
4a	Admin membatalkan proses (menutup <i>pop-up</i>)	Sistem menutup jendela <i>pop-up</i> dan membatalkan seluruh perubahan harga.

Tabel 4.19 Skenario UC-14: Mengelola Pesanan

Identifikasi		
Use Case ID	UC-14	
Nama Use Case	Mengelola Pesanan	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan berada pada halaman manajemen pesanan	
Postkondisi	Status pesanan, status verifikasi pembayaran, dan ketersediaan jadwal armada berhasil diperbarui di <i>database</i> .	
Tujuan	Membantu admin memantau riwayat pesanan serta memperbarui status transaksi secara cepat	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Pesanan		
No	Aktor	Sistem
1	Admin membuka halaman manajemen pesanan.	Sistem memeriksa dan membatalkan <i>booking pending</i> yang telah melewati <i>expiresAt</i> , kemudian memuat kartu ringkasan dan daftar pesanan terbaru.
2	Admin melihat <i>booking pending</i> beserta batas waktu <i>expiresAt</i> .	Sistem menampilkan indikator batas waktu pada <i>booking pending</i> .
3	Admin menekan <i>tab</i> filter status di atas tabel (Semua, <i>Pending</i> , Dikonfirmasi, Selesai, Dibatalkan).	Sistem menyaring dan merender ulang baris data sesuai kategori yang dipilih.
4	Admin mengubah opsi pada <i>dropdown</i> di kolom Pembayaran (<i>Booking</i> , DP, Lunas, atau Penuh).	Sistem langsung menyimpan status pembayaran baru ke basis data dan memperbarui warna label.
5	Admin menekan tombol "Ubah Status" pada kolom Aksi dan memilih status operasional baru (Dikonfirmasi, Selesai, atau Dibatalkan).	Sistem menyimpan status operasional pesanan ke basis data dan memperbarui warna label status.
Kondisi Akhir: Halaman Manajemen Pesanan tetap terbuka dan baris data transaksi telah <i>ter-update</i>		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Data pesanan pada filter yang dipilih kosong	Sistem menampilkan pesan bahwa tidak ada data pesanan pada kategori tersebut.
1b	Ditemukan <i>booking pending</i> yang melewati <i>expiresAt</i>	Sistem mengubah status menjadi dibatalkan dan membuka kembali jadwal armada.
5a	Admin memilih status operasional "Dibatalkan"	Sistem mengubah status pesanan dan otomatis melepas kunci tanggal armada di sistem agar dapat dipesan pelanggan lain.

Tabel 4.20 Skenario UC-15: Mengelola Testimoni

Identifikasi		
Use Case ID	UC-15	
Nama Use Case	Mengelola Testimoni	
Aktor	Admin	
Prekondisi	Admin telah <i>login</i> dan pelanggan telah mengirimkan ulasan baru ke dalam sistem.	
Postkondisi	Status ulasan pelanggan berhasil diperbarui menjadi "Disetujui" (tampil di publik) atau dihapus dari basis data	
Tujuan	Memoderasi ulasan pelanggan untuk memastikan hanya ulasan yang valid dan pantas yang ditampilkan pada halaman utama <i>website</i> .	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Testimoni (Tab "Menunggu")		
No	Aktor	Sistem
1	Admin menekan tab "Menunggu" pada halaman manajemen testimoni.	Sistem mengambil data dari basis data dan menampilkan daftar ulasan pelanggan yang berstatus menunggu moderasi.
2	Admin memeriksa isi ulasan dan menekan tombol "Setujui".	Sistem memperbarui status ulasan tersebut menjadi "Disetujui" di dalam basis data.
3		Sistem merender ulang tampilan, memindahkan kartu ulasan ke tab "Disetujui", dan memunculkan ulasan tersebut di halaman utama publik.
Kondisi Akhir: Ulasan pelanggan berhasil disetujui dan tayang di halaman utama <i>website</i> .		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Admin menilai isi ulasan tidak pantas/spam dan menekan ikon hapus (tempat sampah).	Sistem menghapus data ulasan tersebut dari basis data secara permanen dan merender ulang daftar ulasan.
2a	Admin menekan tombol "Batalan" pada ulasan yang sudah terlanjur berada di tab "Disetujui".	Sistem mengembalikan status ulasan menjadi "Menunggu" dan langsung menyembunyikannya dari halaman utama publik.

Tabel 4.21 Skenario UC-16: Mengelola Pengaturan Situs

Identifikasi		
Use Case ID	UC-16	
Nama Use Case	Mengelola Pengaturan Situs	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan berada pada menu Pengaturan.	
Postkondisi	Data informasi kontak dan identitas bisnis berhasil diperbarui di basis data.	
Tujuan	Memastikan informasi yang tampil pada halaman publik <i>website</i> sesuai dengan data operasional terbaru.	
Skenario Utama		
Kondisi Awal: Halaman Pengaturan Situs		
No	Aktor	Sistem
1	Admin membuka halaman pengaturan situs.	Sistem memuat dan menampilkan data <i>form</i> pengaturan situs saat ini.
2	Admin mengisi atau mengubah kolom <i>input</i> pada bagian Kontak dan Informasi Bisnis.	Sistem memvalidasi kelengkapan data masukan, khususnya pada kolom wajib (Nama Bisnis).
3	Admin menekan tombol "Simpan Pengaturan".	Sistem memproses dan menyimpan pembaruan konfigurasi ke basis data.
4		Sistem menampilkan pesan indikator "Pengaturan berhasil disimpan!" di sebelah tombol.
Skenario Alternatif		
No	Kondisi	Aksi Sistem
2a	Kolom data wajib (seperti Nama Bisnis) dikosongkan	Sistem menolak proses penyimpanan dan menampilkan pesan validasi kesalahan pada kolom tersebut.

Tabel 4.22 Skenario UC-17: Melihat Notifikasi

Identifikasi		
Use Case ID	UC-17	
Nama Use Case	Melihat Notifikasi	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan berada di halaman mana saja pada <i>dashboard</i> . Terdapat data pesanan baru masuk dari pelanggan	
Postkondisi	Notifikasi dibaca, dihapus, atau admin berhasil dialihkan ke detail pesanan yang tersorot	
Tujuan	Memberikan informasi <i>real-time</i> kepada admin terkait pesanan baru dan mempermudah navigasi langsung ke data tersebut	
Skenario Utama		
Kondisi Awal: Halaman <i>dashboard</i> (atau menu lainnya) terbuka dengan indikator lonceng notifikasi menyala		
No	Aktor	Sistem
1	Admin menekan menu atau ikon notifikasi pada bilah navigasi samping (<i>sidebar</i>)	Sistem menampilkan <i>dropdown</i> atau panel daftar notifikasi pesanan masuk yang ditandai dengan indikator titik biru (belum dibaca)
2	Admin menekan salah satu item notifikasi pada daftar tersebut.	Sistem mengubah status notifikasi menjadi sudah dibaca (indikator titik biru hilang)
3		Sistem mengarahkan admin ke Halaman Manajemen Pesanan
4		Sistem memberikan sorotan (garis penanda biru) pada sisi kiri baris tabel pesanan yang sesuai dengan notifikasi
Kondisi Akhir: Halaman Manajemen Pesanan terbuka dan baris transaksi yang bersangkutan berhasil disorot		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Admin menekan tombol teks "Tandai dibaca" di bagian atas daftar notifikasi	Sistem mengubah seluruh status notifikasi menjadi terbaca dan memunculkan indikator pesan "Semua notifikasi ditandai dibaca"
1b	Admin menekan tombol teks "Hapus semua" di bagian bawah daftar notifikasi	Sistem membersihkan seluruh riwayat pesan dari daftar notifikasi
2a	Tidak ada riwayat notifikasi sama sekali	Sistem menampilkan daftar notifikasi dalam keadaan kosong

Tabel 4.23 Skenario UC-18: Mengekspor Data Excel

Identifikasi		
Use Case ID	UC-18	
Nama Use Case	Mengekspor Data Excel	
Aktor	Admin	
Prekondisi	Admin sudah <i>login</i> dan berada pada Halaman Manajemen Pesanan (atau halaman data lainnya).	
Postkondisi	Berkas laporan rekapitulasi berformat .xlsx berhasil diunduh ke penyimpanan lokal admin.	
Tujuan	Memfasilitasi admin untuk merekapitulasi data riwayat transaksi operasional ke dalam format <i>spreadsheet</i> yang mudah dicetak atau diolah.	
Skenario Utama		
Kondisi Awal: Halaman Manajemen Pesanan terbuka dan daftar baris data transaksi tersedia.		
No	Aktor	Sistem
1	Admin menekan tombol hijau "Export Excel" yang berada di atas tabel data.	Sistem mengambil baris data yang sedang ditampilkan pada tabel (termasuk menyesuaikan jika ada <i>filter</i> status yang sedang aktif).
2		Sistem memetakan data tersebut ke dalam format kolom <i>spreadsheet</i> terstruktur (No, Nama Pelanggan, No. HP, Armada, Tanggal Mulai, Tanggal Selesai, Durasi, Dengan Sopir, Status, Pembayaran, Keterangan, dan Tanggal <i>Booking</i>).
3		Sistem membentuk nama berkas secara dinamis berdasarkan tanggal sistem (contoh: <i>booking_YYYY-MM-DD.xlsx</i>).
4		Sistem merespons dengan memicu proses unduhan (<i>download</i>) otomatis berkas tersebut pada <i>browser</i> admin.
Kondisi Akhir: Halaman Manajemen Pesanan tetap terbuka dan berkas Excel hasil ekspor berhasil tersimpan di perangkat admin		
Skenario Alternatif		
No	Kondisi	Aksi Sistem
1a	Tabel data pesanan sedang dalam keadaan kosong (tidak ada riwayat sama sekali)	Sistem tetap memicu unduhan, namun berkas Excel yang dihasilkan hanya memuat <i>header</i> judul kolom tanpa baris data transaksi.

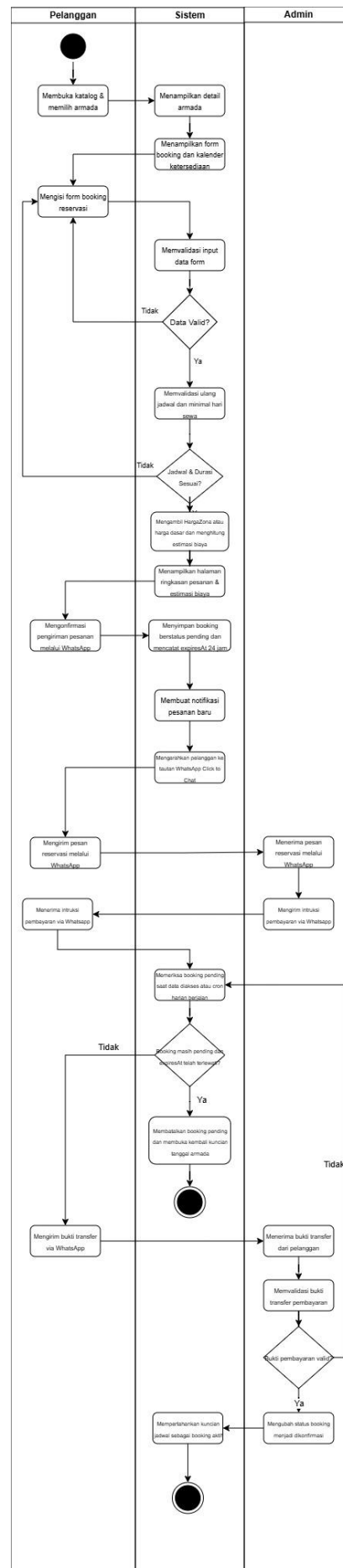
4.2.3 *Activity Diagram*

Activity diagram menggambarkan aliran kerja (*workflow*) atau urutan aktivitas yang terjadi di dalam sistem secara runtut. Pada pengembangan *website* reservasi Rahayu Trans Travel Bandung ini, *activity diagram* dirancang untuk memetakan interaksi antara dua aktor utama, yaitu Pelanggan dan Admin.

Alur aktivitas dimulai dari sisi pelanggan yang melakukan pemilihan armada kendaraan, mengisi formulir reservasi data diri dan jadwal sewa secara manual, hingga melakukan pengiriman format pesanan terstruktur ke WhatsApp admin. Sementara itu, alur aktivitas di sisi admin berfokus pada pengelolaan data armada, verifikasi bukti pembayaran manual, pembaruan status operasional transaksi, hingga moderasi terhadap ulasan atau testimoni yang dikirimkan oleh pelanggan melalui halaman lacak pesanan.

A. *Activity Diagram* Konfirmasi Reservasi via WhatsApp dan Verifikasi Pembayaran

Alur ini menggambarkan proses ketika pelanggan memilih armada kendaraan pada halaman katalog dan mengisi formulir reservasi. Sistem kemudian akan memvalidasi ketersediaan jadwal serta durasi sewa, merender ringkasan pesanan, dan menyimpannya ke dalam basis data. Pada akhir alur, pelanggan akan diarahkan ke integrasi WhatsApp *Click to Chat* untuk mengirimkan pesan terformat otomatis kepada admin.

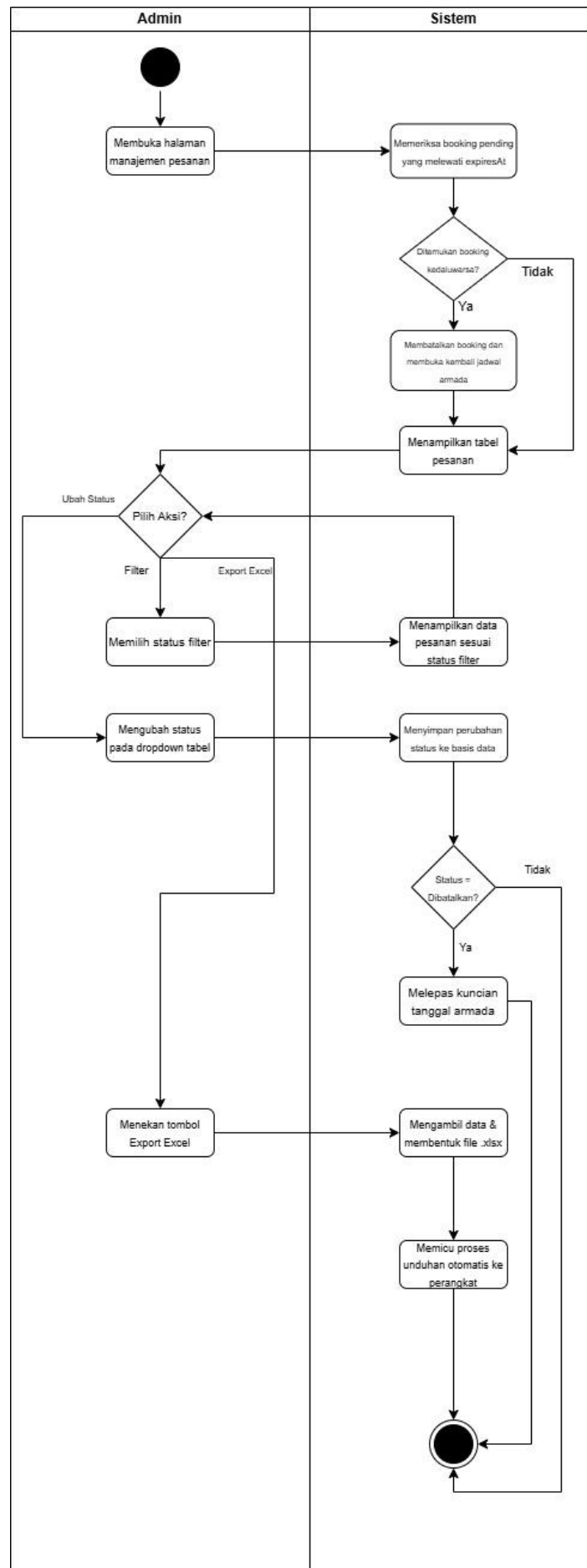


Gambar 4.4 Activity Diagram A

Berdasarkan Gambar 4.4, aktivitas dimulai ketika pelanggan memilih armada dan melengkapi formulir reservasi. Sistem memvalidasi data, ketersediaan jadwal, dan minimal hari sewa, kemudian menghitung estimasi biaya menggunakan tarif Harga Zona atau harga dasar armada sebagai cadangan. Setelah pelanggan mengonfirmasi pengiriman melalui WhatsApp, sistem menyimpan *booking* berstatus *pending* dengan *expiresAt* selama 24 jam. Admin memvalidasi bukti pembayaran yang dikirim melalui WhatsApp, sedangkan sistem membatalkan *booking* yang tetap *pending* setelah melewati *expiresAt*.

B. Activity Diagram Manajemen Pesanan dan Ekspor Data

Alur ini memodelkan tahapan yang dilakukan oleh admin dalam mengelola riwayat transaksi pada panel *dashboard*. Admin dapat menyaring (filter) data pesanan, memperbarui status operasional secara dinamis, serta melakukan pengunduhan (*export*) seluruh riwayat transaksi ke dalam format berkas Excel (.xlsx) untuk kebutuhan pelaporan atau rekapitulasi data.

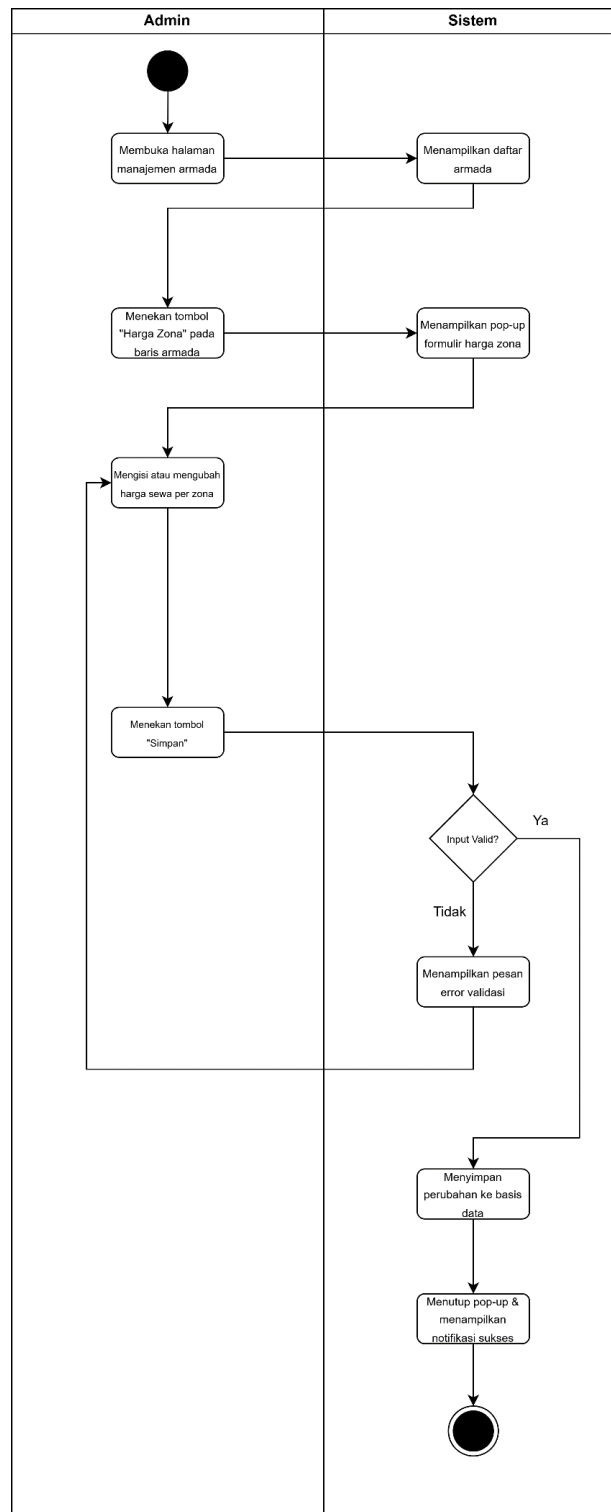


Gambar 4.5 Activity Diagram B

Berdasarkan Gambar 4.5, aktivitas dimulai ketika admin membuka halaman manajemen pesanan. Sistem memeriksa *booking* berstatus *pending* yang telah melewati *expiresAt* dan membatalkannya agar jadwal armada tersedia kembali, kemudian menampilkan daftar pesanan. Admin dapat memfilter data berdasarkan status, memperbarui status pesanan dan pembayaran, serta mengeksport data ke dalam format Excel. Apabila status pesanan diubah menjadi dibatalkan, sistem melepaskan kunci tanggal armada agar dapat dipesan kembali.

C. Activity Diagram Pengelolaan Armada dan Harga Zona

Alur ini menggambarkan interaksi admin dalam menetapkan dan memperbarui tarif sewa kendaraan. Admin dapat memilih armada pada halaman manajemen, kemudian membuka formulir *pop-up* untuk memasukkan rentang harga sewa harian yang disesuaikan dengan masing-masing zona wilayah tujuan. Sistem akan memvalidasi masukan dan menyimpan pembaruan harga tersebut ke dalam basis data.



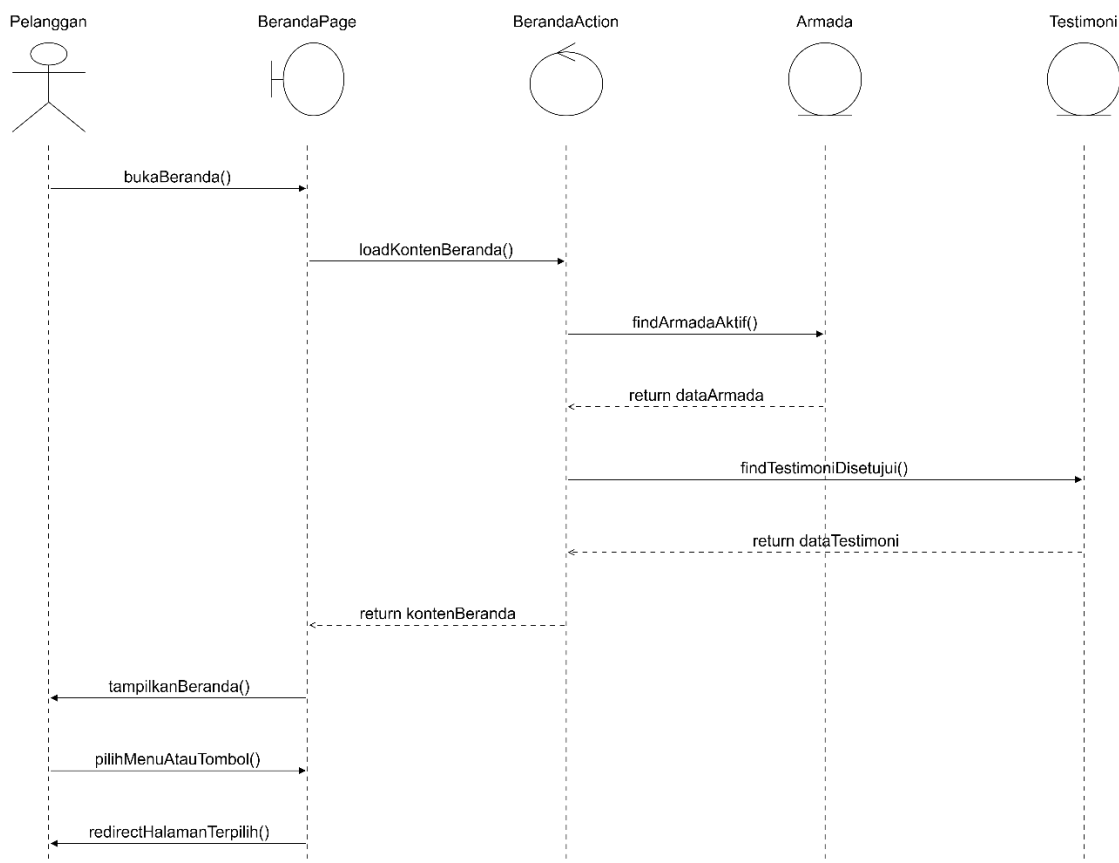
Gambar 4.6 Activity Diagram C

Berdasarkan Gambar 4.6 di atas, alur aktivitas menggambarkan interaksi admin dalam menetapkan dan memperbarui tarif sewa kendaraan. Admin dapat memilih armada pada halaman manajemen, kemudian membuka formulir *pop-up* untuk memasukkan rentang harga sewa harian yang disesuaikan dengan masing-masing zona wilayah tujuan. Sistem selanjutnya akan memvalidasi masukan tersebut dan menyimpan pembaruan data harga ke dalam basis data.

4.2.4 Sequence Diagram

Pada sistem reservasi Rahayu Trans Travel, *Sequence Diagram* digunakan untuk menggambarkan urutan komunikasi yang terjadi ketika pelanggan atau admin menjalankan fitur pada aplikasi. Objek *boundary* merepresentasikan halaman dan komponen antarmuka pada direktori *app*, objek *control* merepresentasikan proses validasi serta *route handler* pada direktori *app/api*, sedangkan objek *entity* merepresentasikan model data yang dikelola melalui Prisma ORM dan PostgreSQL. Diagram disusun berdasarkan alur fitur yang terdapat pada *project*, seperti pencarian armada, reservasi, perhitungan harga zona, pengiriman pesan melalui WhatsApp, pembatalan *booking* yang melewati *expiresAt*, serta pengelolaan data oleh admin. Penamaan pesan atau *method* pada diagram bersifat konseptual, tetapi tetap mengacu pada proses dan perilaku yang diterapkan dalam *source code* sistem Rahayu Trans Travel.

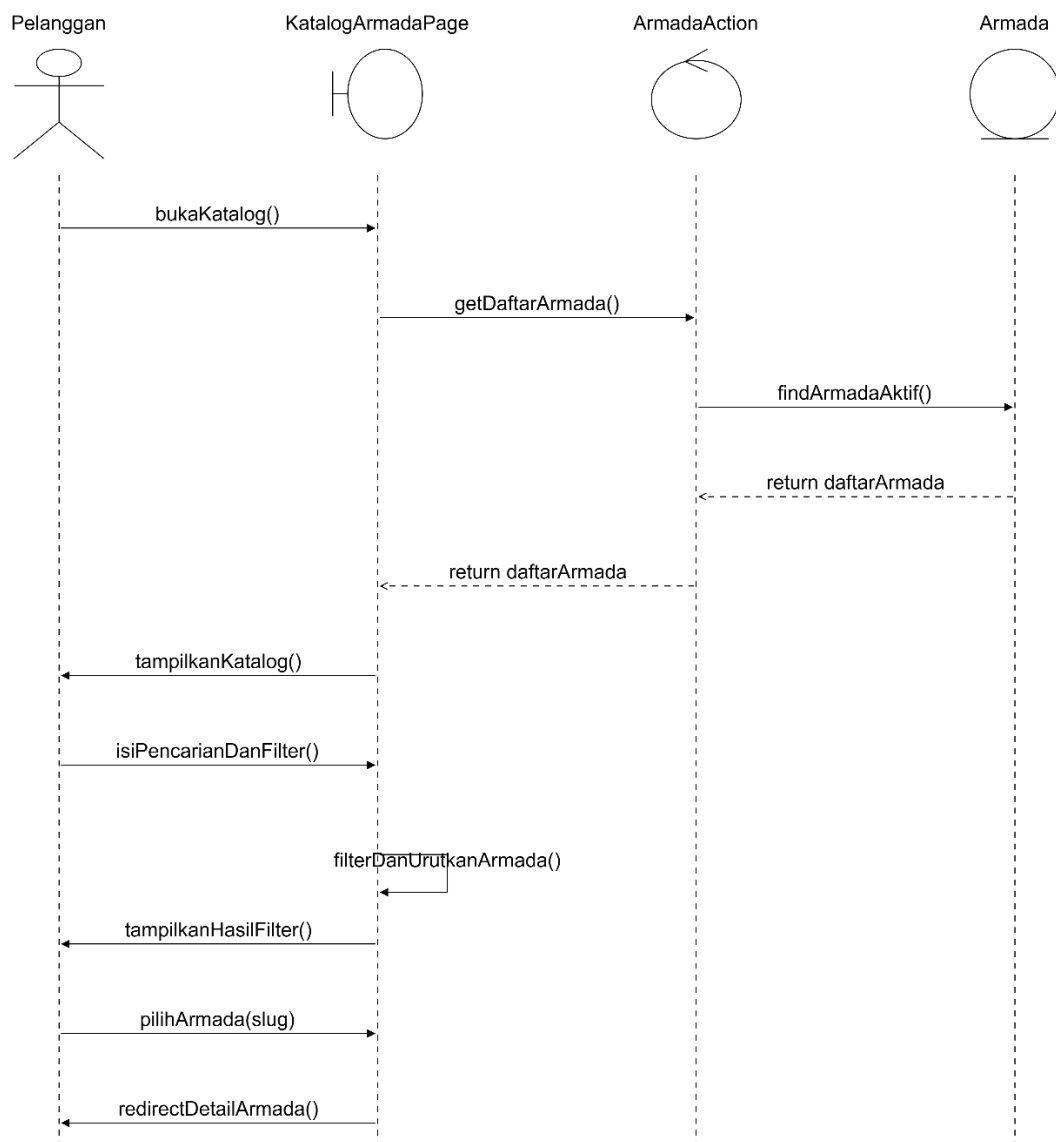
4.2.4.1 Diagram Sequence UC-01 - Melihat Beranda



Gambar 4.7 Diagram Sequence UC-01 - Melihat Beranda

Penjelasan Alur: Pelanggan mengakses halaman beranda Rahayu Trans Travel. Sistem mengambil data armada yang tersedia dan testimoni yang telah disetujui, kemudian menampilkan formulir pencarian, armada unggulan, cara pemesanan, testimoni, dan informasi kontak. Pelanggan dapat menggunakan formulir pencarian atau memilih menu untuk menuju halaman terkait.

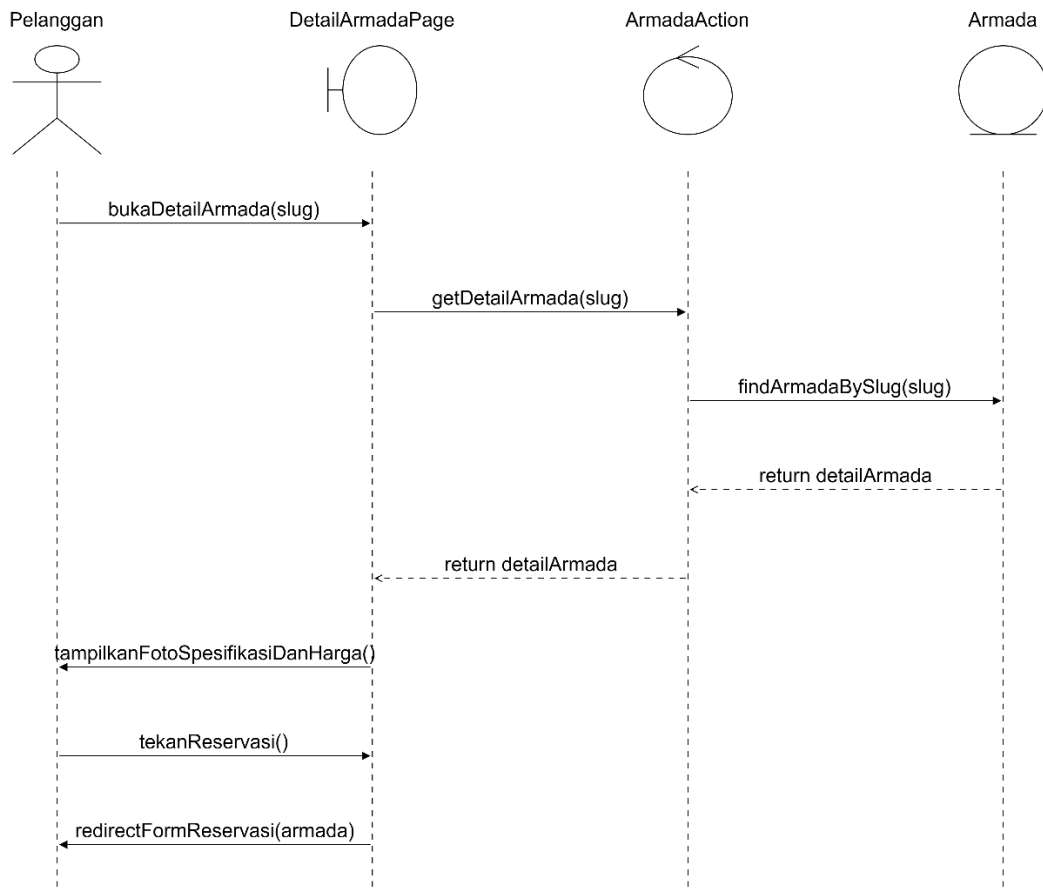
4.2.4.2 Diagram Sequence UC-02 - Melihat Katalog Armada



Gambar 4.8 Diagram Sequence UC-02 - Melihat Katalog Armada

Penjelasan Alur: Pelanggan membuka halaman katalog dan sistem mengambil data armada dari basis data. Pelanggan dapat memasukkan tanggal perjalanan, tujuan, dan jumlah penumpang serta menggunakan filter kategori atau kapasitas. Sistem memeriksa armada yang sesuai dan tidak memiliki *booking* bertabrakan, kemudian menampilkan hasil pencarian. Pelanggan dapat memilih armada untuk membuka halaman detail.

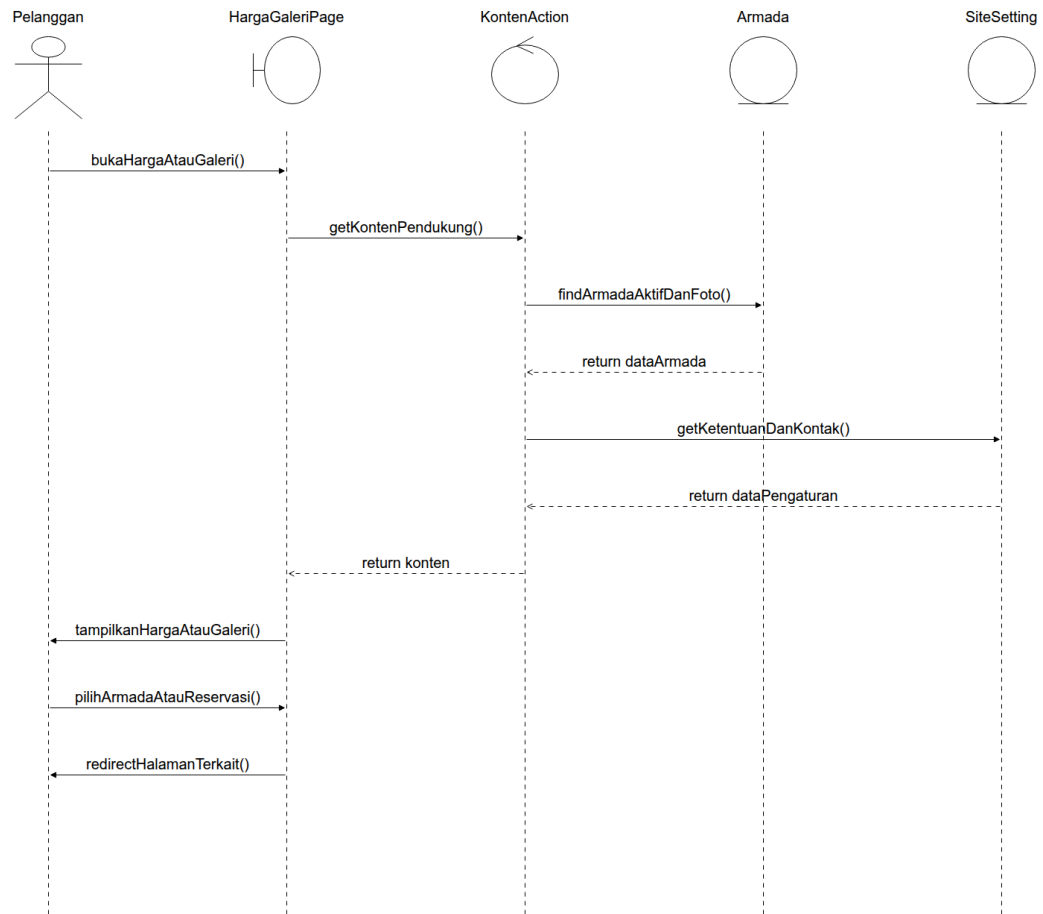
4.2.4.3 Diagram Sequence UC-03 - Melihat Detail Armada



Gambar 4.9 Diagram Sequence UC-03 - Melihat Detail Armada

Penjelasan Alur: Pelanggan memilih armada dari halaman katalog. Sistem menggunakan slug kendaraan untuk mengambil data armada yang meliputi foto, spesifikasi, kapasitas, transmisi, tahun, deskripsi, dan harga dasar per hari. Setelah detail ditampilkan, pelanggan dapat menekan tombol reservasi untuk membuka formulir dengan armada tersebut sebagai pilihan awal.

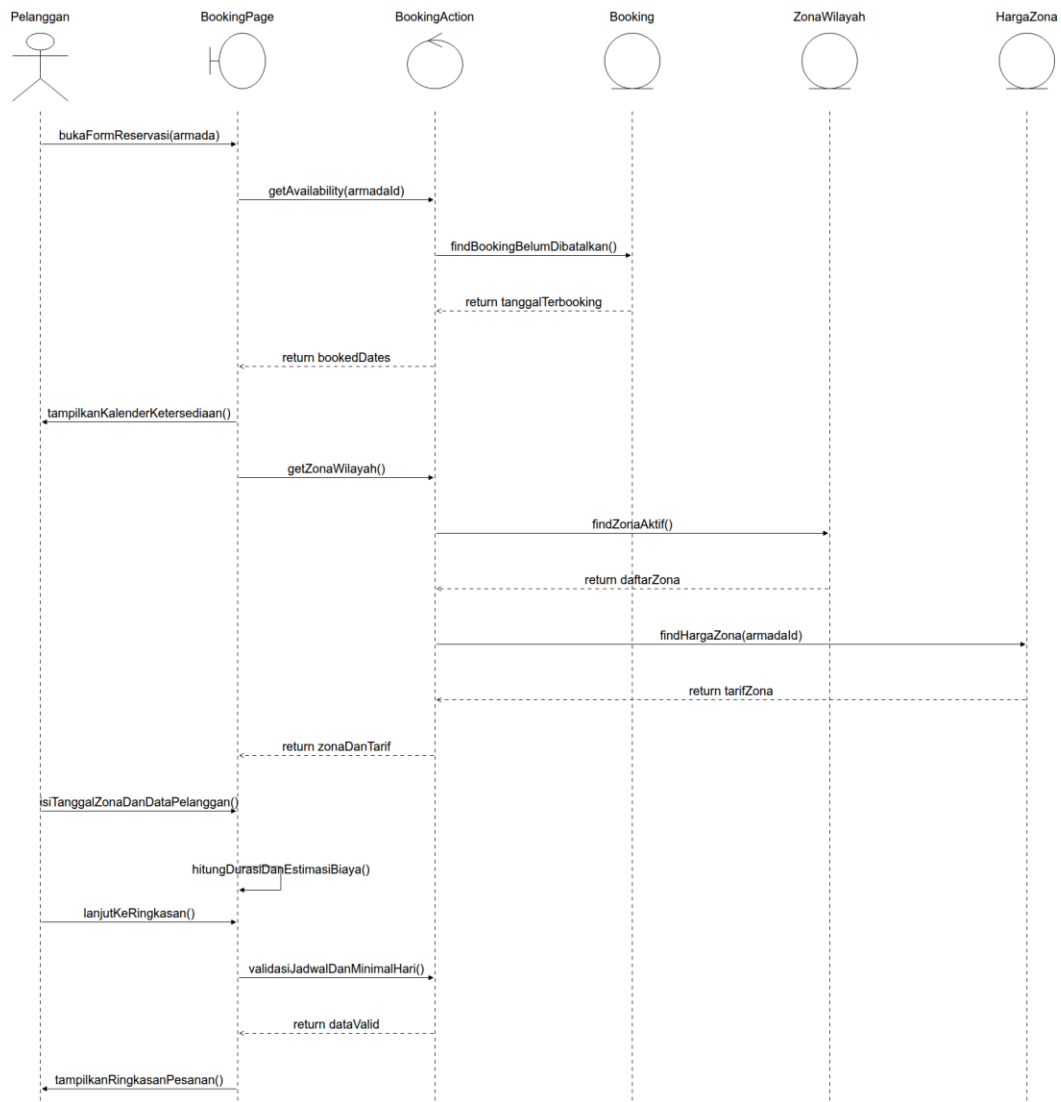
4.2.4.4 Diagram Sequence UC-04 - Melihat Harga dan Galeri



Gambar 4.10 Diagram Sequence UC-04 - Melihat Harga dan Galeri

Penjelasan Alur: Pelanggan membuka halaman harga atau galeri. Sistem mengambil data armada, foto kendaraan, ketentuan penyewaan, dan informasi pendukung yang tersimpan pada sistem. Data tersebut ditampilkan agar pelanggan dapat melihat referensi harga dasar dan dokumentasi armada sebelum melanjutkan ke halaman detail atau formulir reservasi.

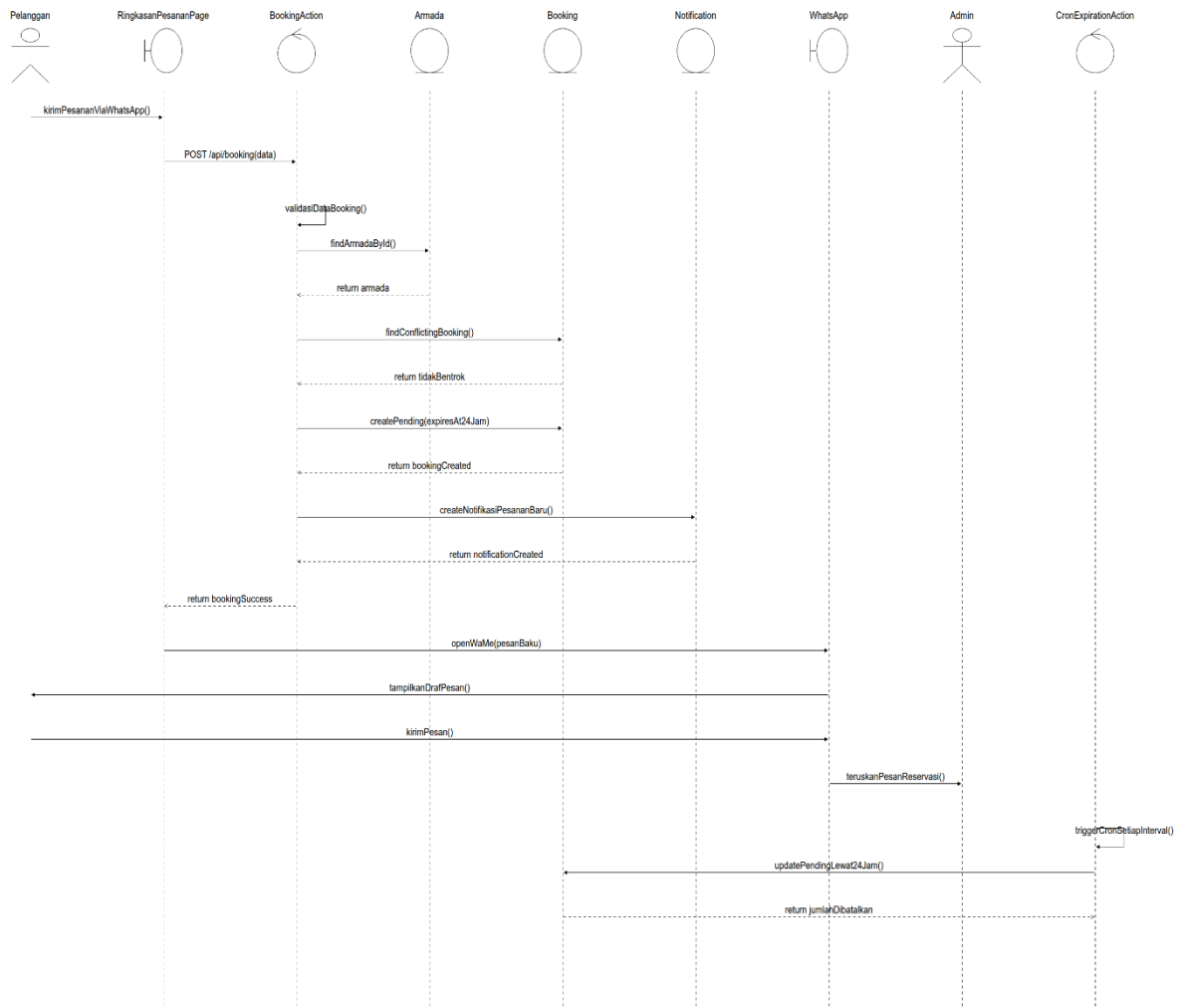
4.2.4.5 Diagram Sequence UC-05 - Mengisi Formulir Reservasi



Gambar 4.11 Diagram Sequence UC-05 - Mengisi Formulir Reservasi

Penjelasan Alur: Pelanggan membuka formulir reservasi dengan armada yang telah dipilih. Sistem terlebih dahulu memeriksa *booking pending* yang melewati *expiresAt*, kemudian mengambil tanggal dari *booking* yang belum dibatalkan untuk ditampilkan pada kalender. Sistem juga mengambil zona wilayah dan tarif berdasarkan armada. Setelah pelanggan memilih tanggal dan zona, sistem menggunakan harga zona atau harga dasar armada apabila tarif zona belum tersedia. Sistem memvalidasi data, minimal hari sewa, dan bentrok jadwal sebelum menampilkan ringkasan serta estimasi biaya.

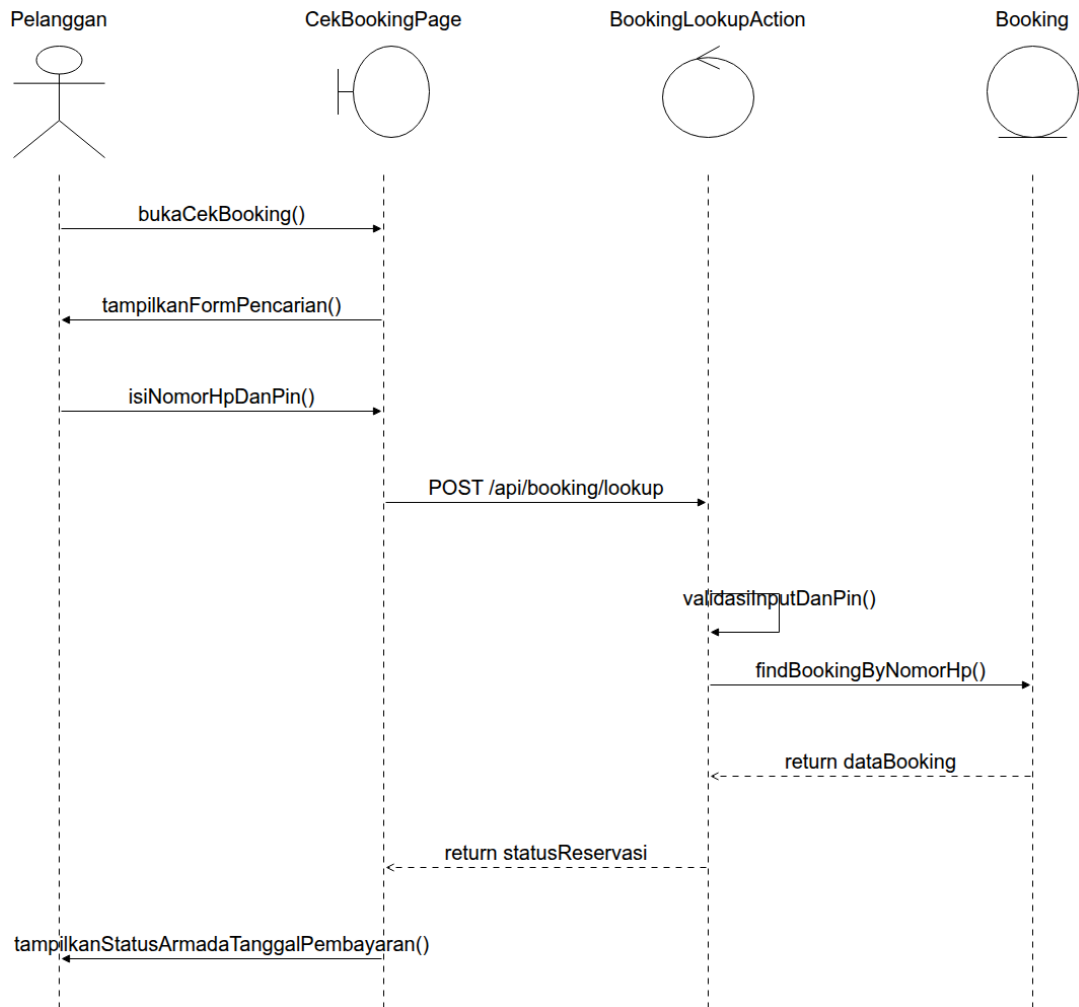
4.2.4.6 Diagram Sequence UC-06 - Mengirim Pesanan via WhatsApp



Gambar 4.12 Diagram Sequence UC-06 - Mengirim Pesanan via WhatsApp

Penjelasan Alur: Pelanggan menekan tombol kirim pesan via WhatsApp pada halaman ringkasan. Sistem memvalidasi kembali data armada dan tanggal, kemudian menyimpan *booking* dengan status *pending* serta *expiresAt* selama 24 jam. Sistem membuat notifikasi pesan baru untuk admin, membentuk pesan berdasarkan data reservasi, dan membuka WhatsApp *Click to Chat*. Pesan tetap dikirim oleh pelanggan melalui aplikasi WhatsApp.

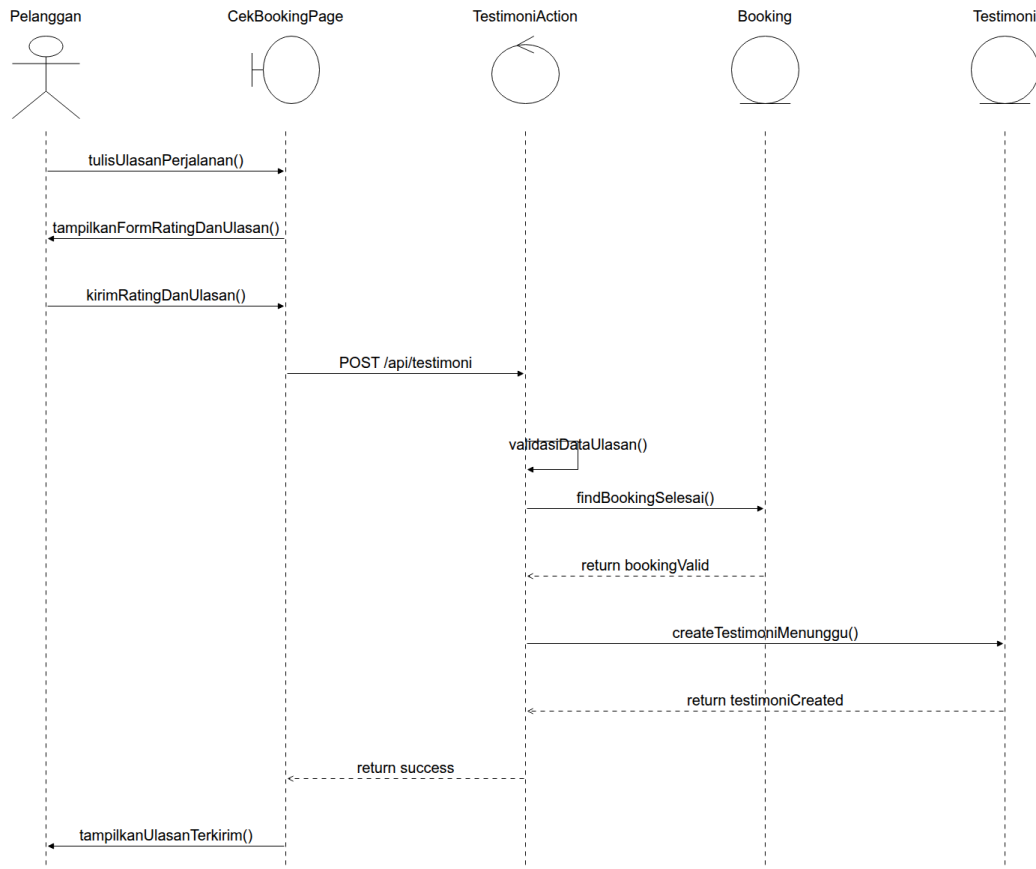
4.2.4.7 Diagram Sequence UC-07 - Mengecek Status Pesanan



Gambar 4.13 Diagram Sequence UC-07 - Mengecek Status Pesanan

Penjelasan Alur: Pelanggan membuka halaman cek *booking* dan memasukkan nomor HP serta PIN verifikasi. Sistem memeriksa *booking pending* yang telah kedaluwarsa, mencari transaksi berdasarkan nomor HP, dan mencocokkan PIN. Apabila valid, sistem menampilkan data reservasi, armada, tanggal sewa, status operasional, dan status pembayaran dengan informasi pribadi yang disamarkan.

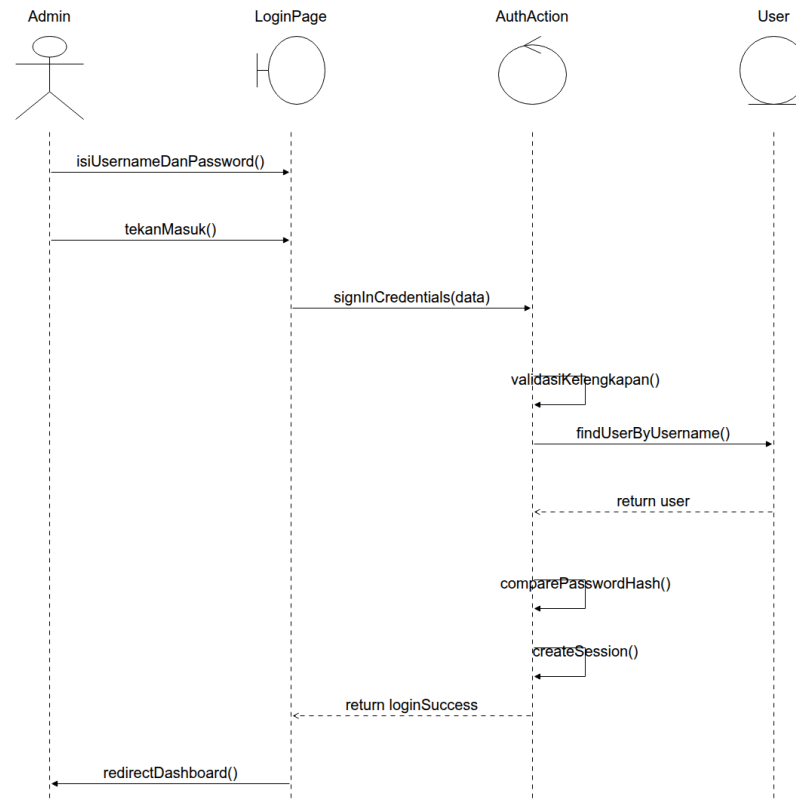
4.2.4.8 Diagram Sequence UC-08 - Memberikan Testimoni



Gambar 4.14 Diagram Sequence UC-08 - Memberikan Testimoni

Penjelasan Alur: Pelanggan membuka hasil pencarian *booking* yang telah berstatus selesai. Sistem memeriksa bahwa *booking* belum memiliki testimoni, kemudian menampilkan formulir *rating* dan ulasan. Setelah data dikirim, sistem menyimpan testimoni dengan status belum disetujui. Testimoni dapat ditampilkan pada halaman publik setelah melalui persetujuan admin.

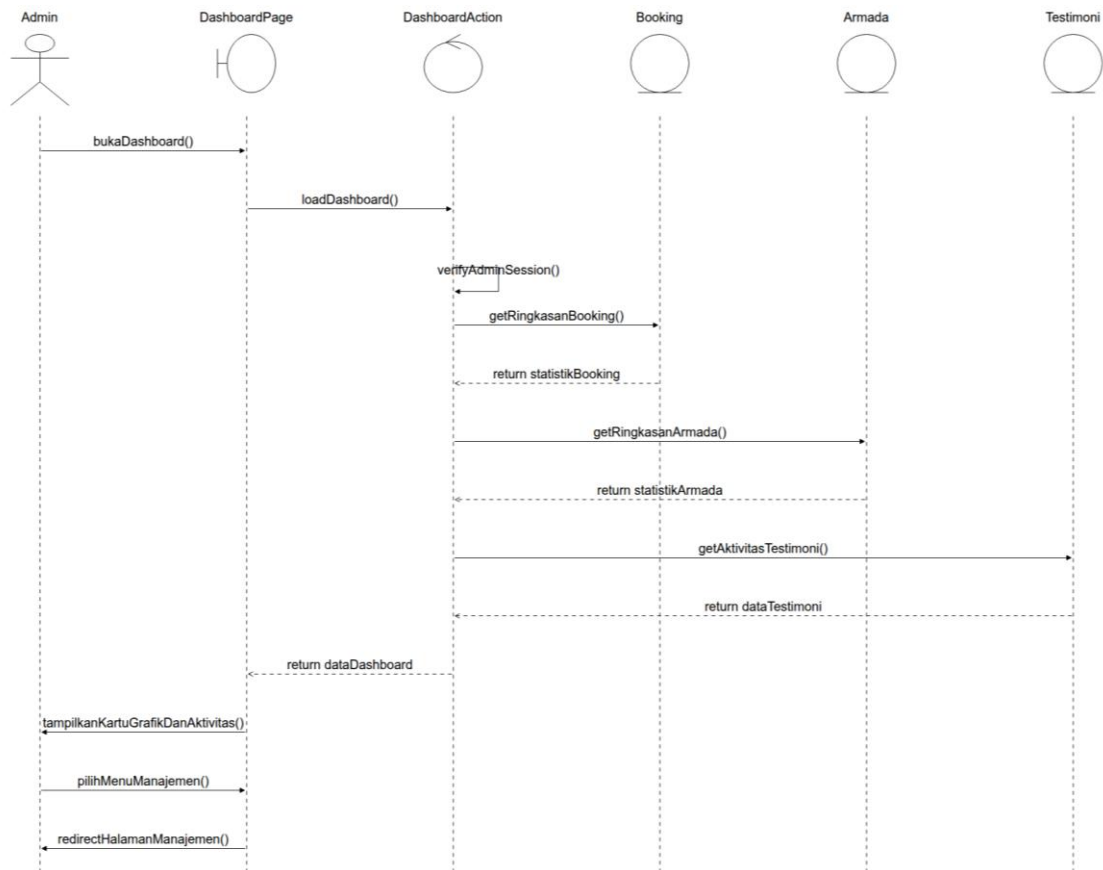
4.2.4.9 Diagram Sequence UC-09 - *Login Admin*



Gambar 4.15 Diagram Sequence UC-09 - *Login Admin*

Penjelasan Alur: Admin memasukkan *username* dan *password* pada halaman *login*. Sistem mencari akun berdasarkan *username* dan membandingkan *password* dengan nilai *passwordHash* pada entitas User. Apabila kredensial valid, sistem membuat sesi autentikasi dan mengarahkan admin menuju *dashboard*. Jika tidak valid, sistem menolak autentikasi dan menampilkan pesan kesalahan.

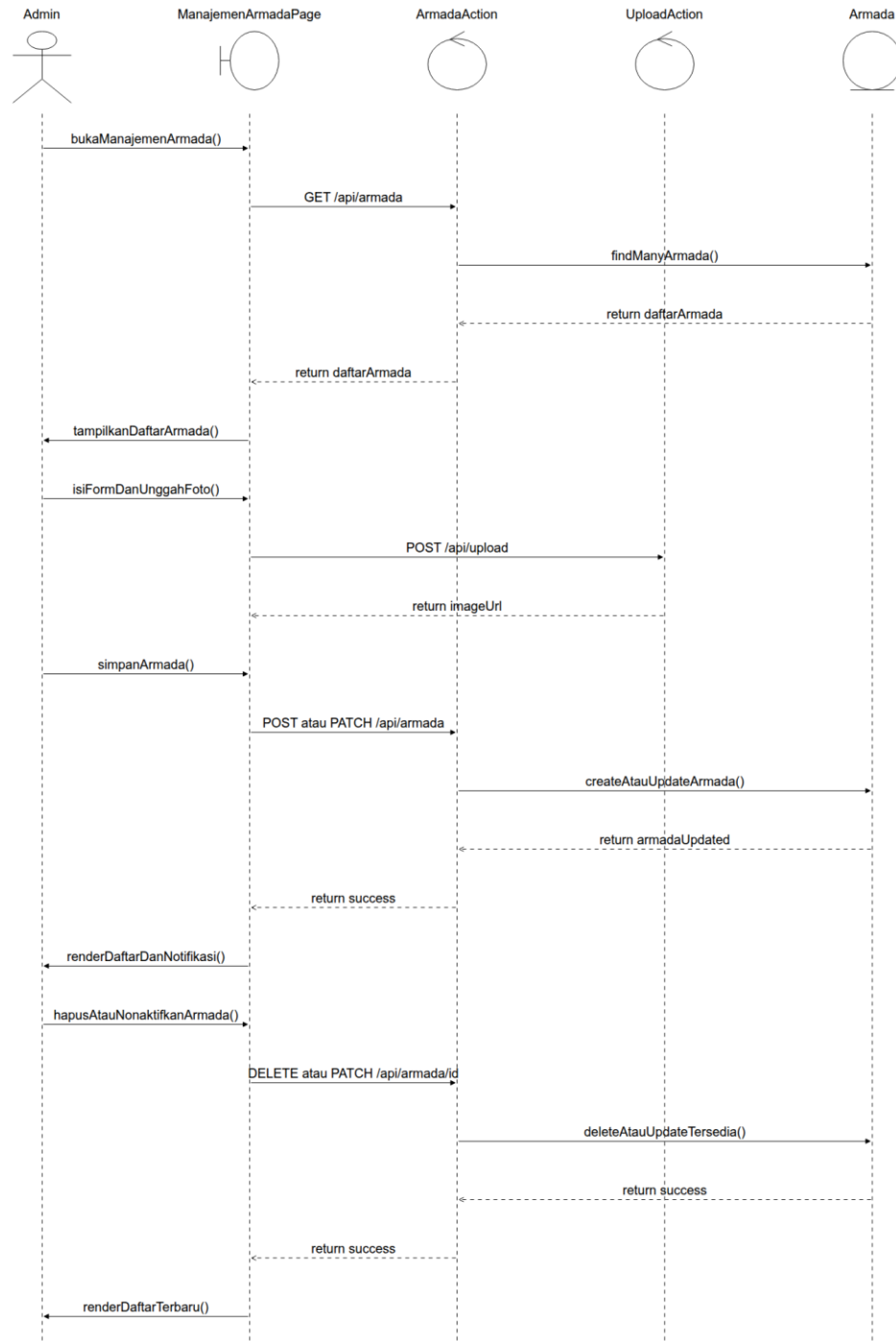
4.2.4.10 Diagram Sequence UC-10 - Melihat *Dashboard*



Gambar 4.16 Diagram Sequence UC-10 - Melihat *Dashboard*

Penjelasan Alur: Admin yang telah *login* membuka halaman *dashboard*. Sistem memeriksa sesi autentikasi, menjalankan pemeriksaan *booking* kedaluwarsa, dan mengambil ringkasan data armada, *booking*, pembayaran, serta testimoni. Data tersebut ditampilkan dalam bentuk statistik, grafik, dan aktivitas terbaru untuk membantu admin memantau kondisi operasional.

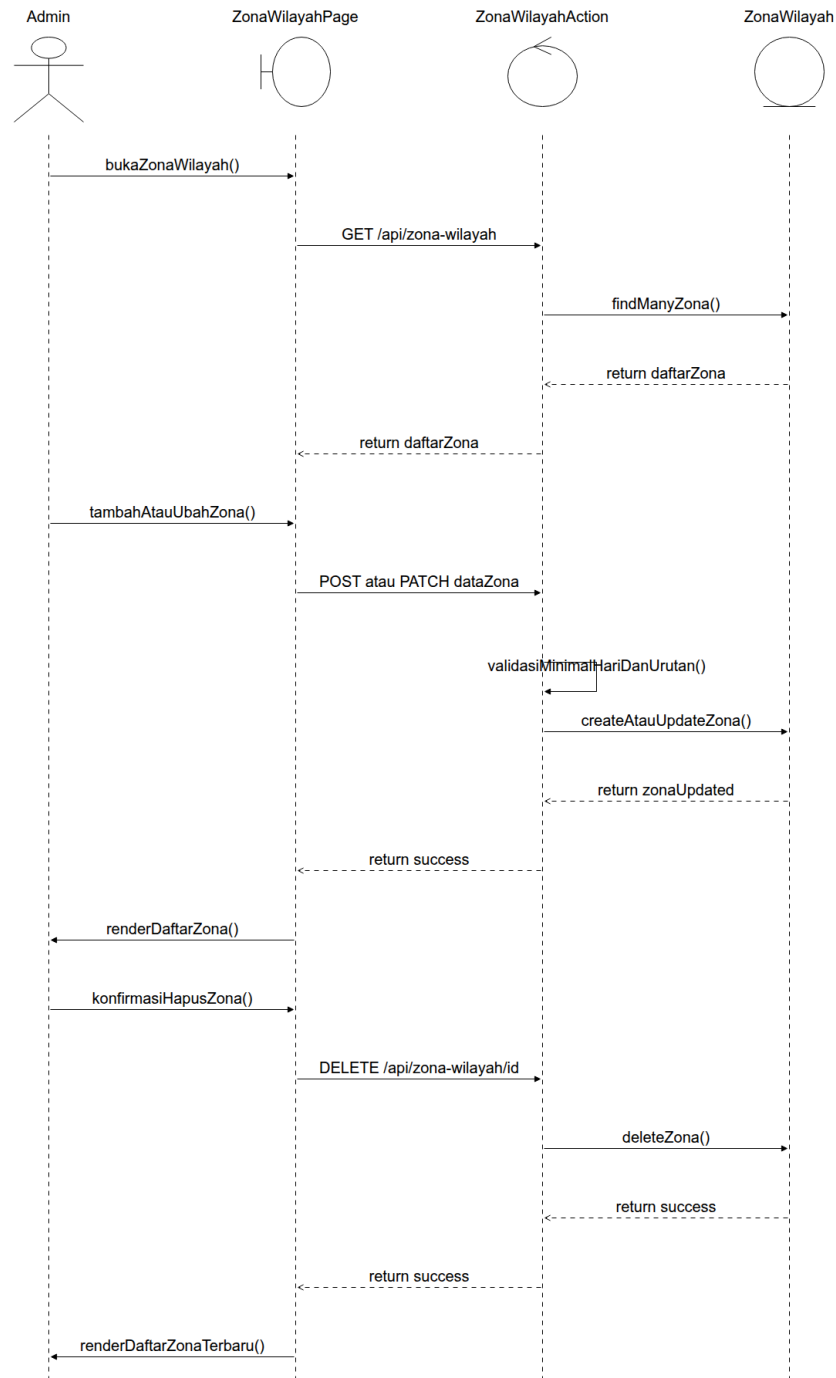
4.2.4.11 Diagram Sequence UC-11 - Mengelola Armada



Gambar 4.17 Diagram Sequence UC-11 - Mengelola Armada

Penjelasan Alur: Admin membuka halaman manajemen armada dan sistem mengambil seluruh data kendaraan. Admin dapat menambah, mengubah, menghapus, mengunggah foto, atau memperbarui status ketersediaan kendaraan. Sistem memvalidasi data masukan dan menyimpan perubahan pada entitas Armada. Setelah proses berhasil, daftar armada ditampilkan kembali dalam kondisi terbaru.

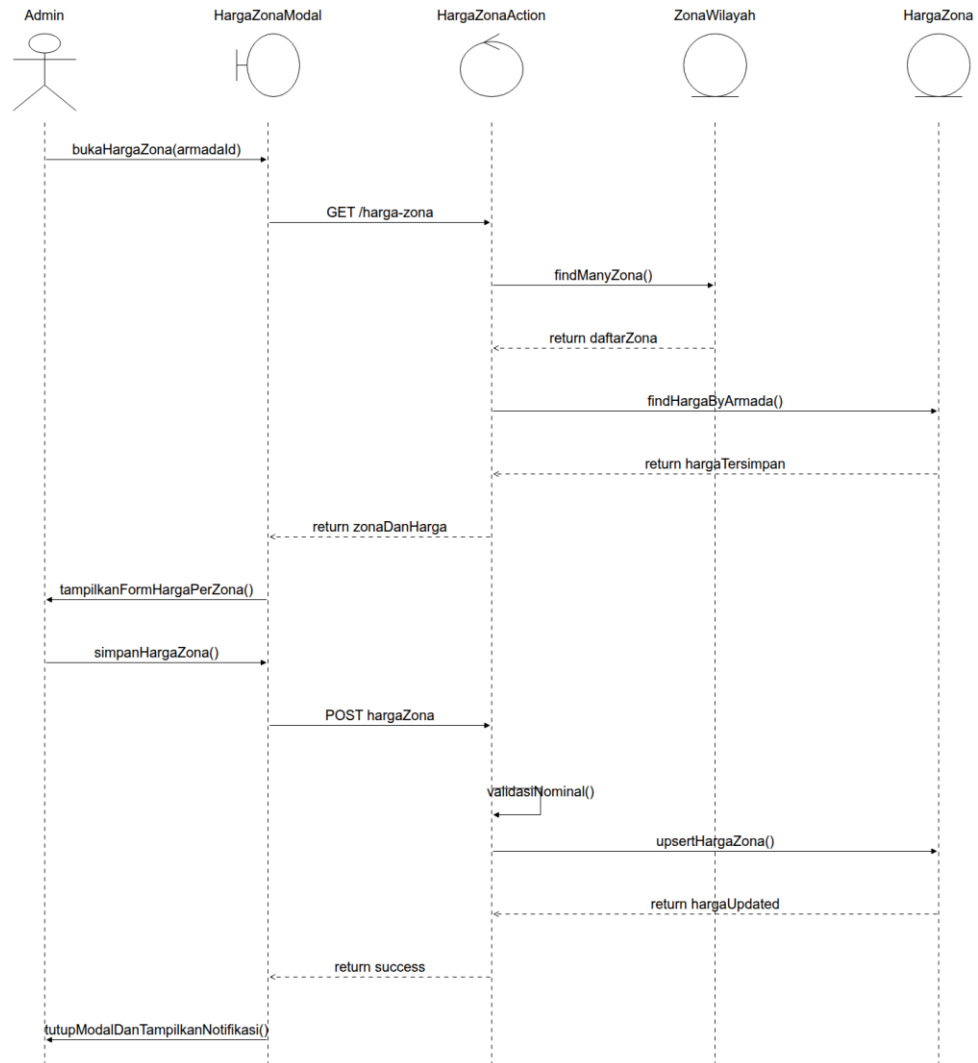
4.2.4.12 Diagram Sequence UC-12 - Mengelola Zona Wilayah



Gambar 4.18 Diagram Sequence UC-12 - Mengelola Zona Wilayah

Penjelasan Alur: Admin membuka halaman zona wilayah dan sistem mengambil seluruh data ZonaWilayah. Admin dapat menambah, mengubah, atau menghapus zona beserta nilai minimal hari sewa dan urutan tampilnya. Sistem memvalidasi masukan sebelum menjalankan operasi penyimpanan, kemudian menampilkan daftar zona yang telah diperbarui.

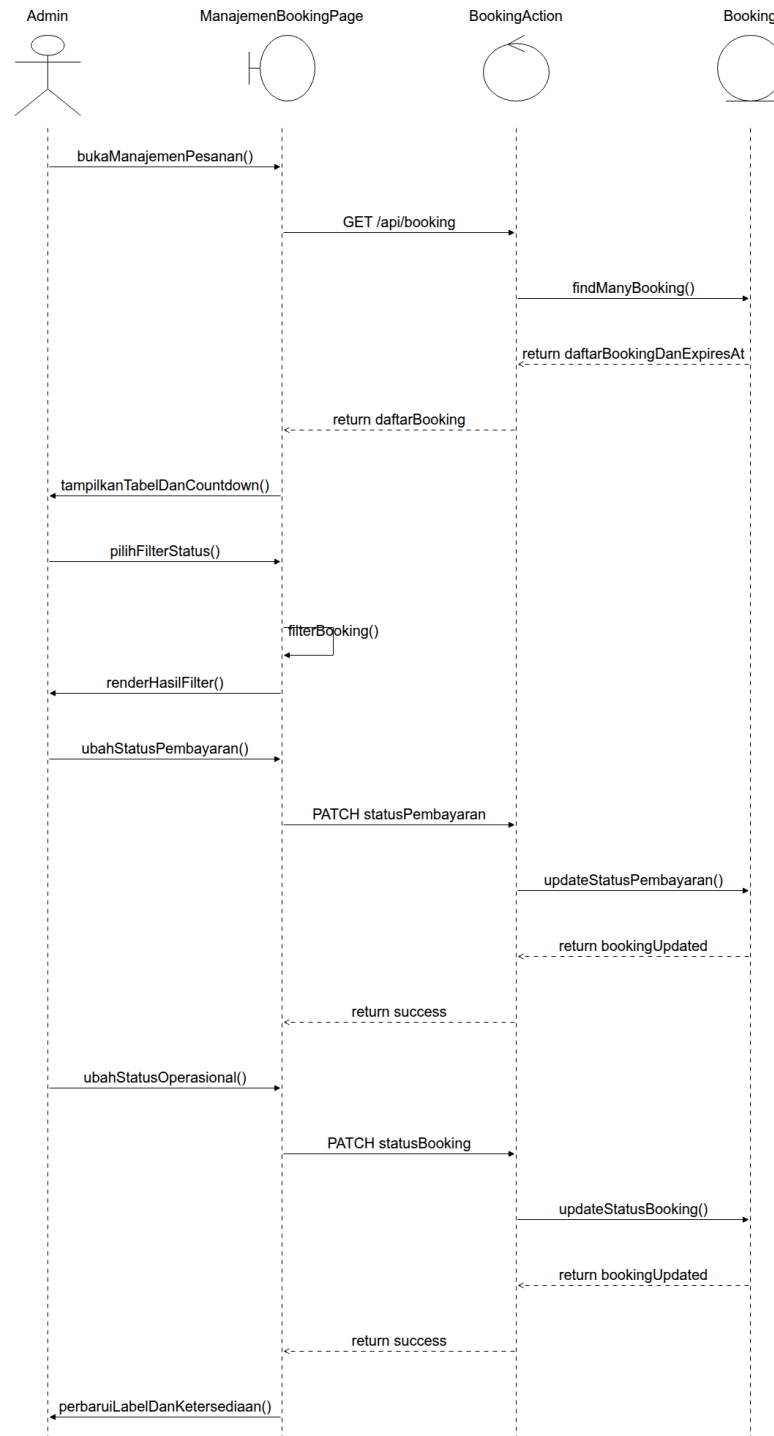
4.2.4.13 Diagram Sequence UC-13 - Mengelola Harga Zona



Gambar 4.19 Diagram Sequence UC-13 - Mengelola Harga Zona

Penjelasan Alur: Admin memilih pengaturan harga zona pada salah satu armada. Sistem mengambil daftar zona wilayah dan data HargaZona yang berhubungan dengan armada tersebut. Admin memasukkan tarif per hari untuk setiap zona, kemudian sistem memvalidasi nominal dan menyimpan data menggunakan proses *upsert*. Setiap kombinasi armada dan zona hanya memiliki satu tarif.

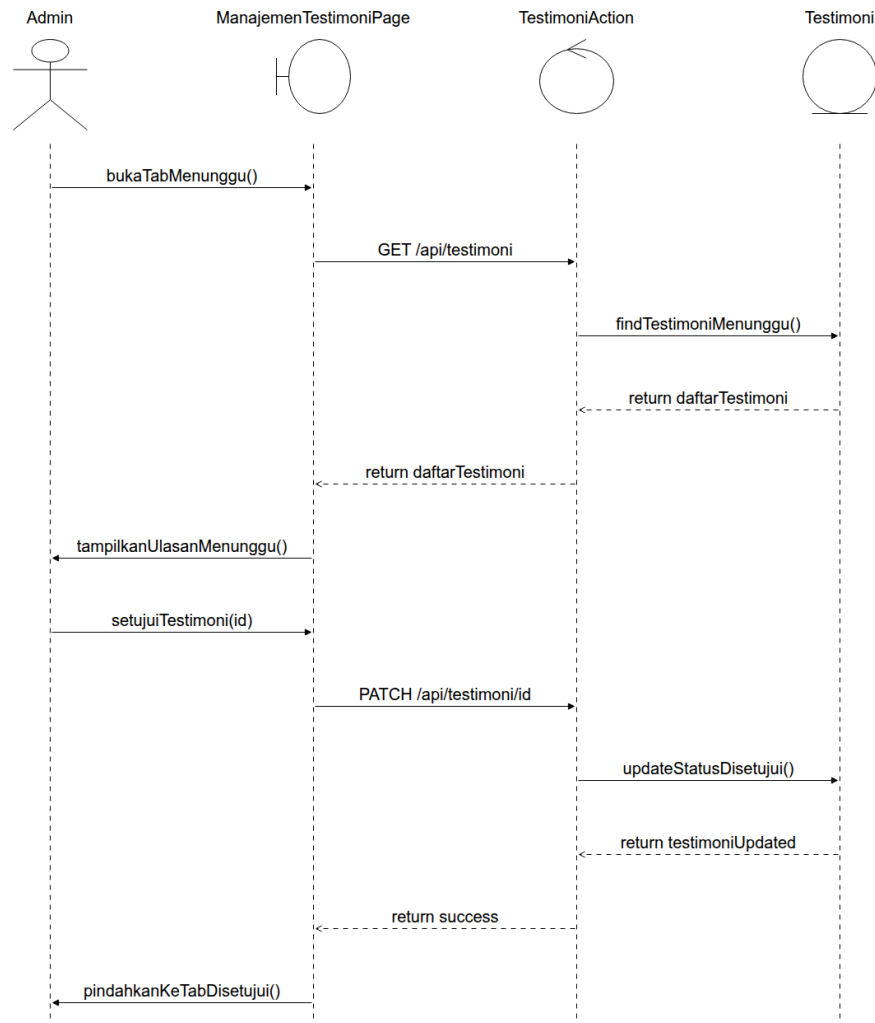
4.2.4.14 Diagram Sequence UC-14 - Mengelola Pesanan



Gambar 4.20 Diagram Sequence UC-14 - Mengelola Pesanan

Penjelasan Alur: Admin membuka halaman manajemen *booking*. Sistem memeriksa *booking pending* dengan *expiresAt* yang telah terlewat dan mengubahnya menjadi dibatalkan sebelum mengambil daftar *booking* terbaru. Admin dapat memfilter pesanan serta memperbarui status *booking* dan status pembayaran. Vercel *Cron* menjalankan pemeriksaan harian sebagai proses cadangan terhadap *booking* yang telah kedaluwarsa.

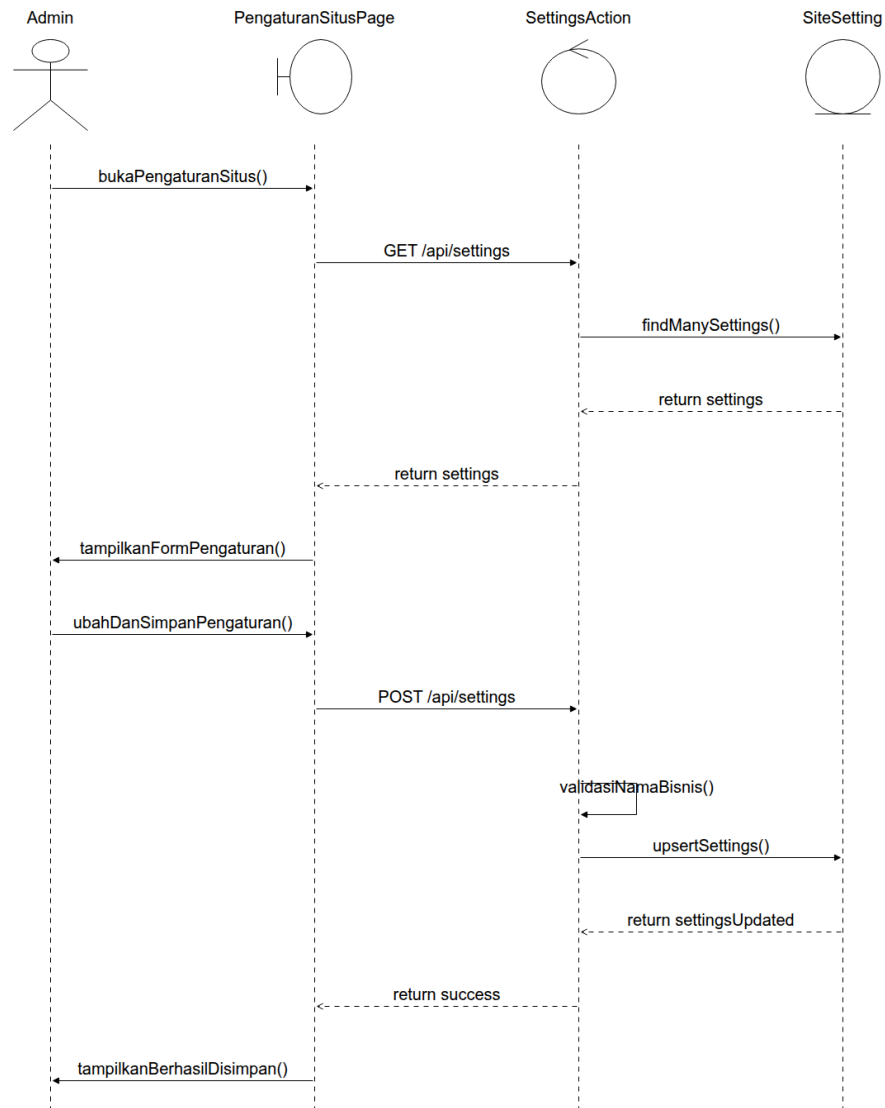
4.2.4.15 Diagram Sequence UC-15 - Mengelola Testimoni



Gambar 4.21 Diagram Sequence UC-15 - Mengelola Testimoni

Penjelasan Alur: Admin membuka halaman manajemen testimoni dan sistem mengambil daftar ulasan pelanggan. Admin dapat meninjau isi dan *rating*, menyetujui testimoni agar tampil pada halaman publik, atau menghapus testimoni yang tidak sesuai. Sistem menyimpan perubahan dan menampilkan daftar testimoni terbaru.

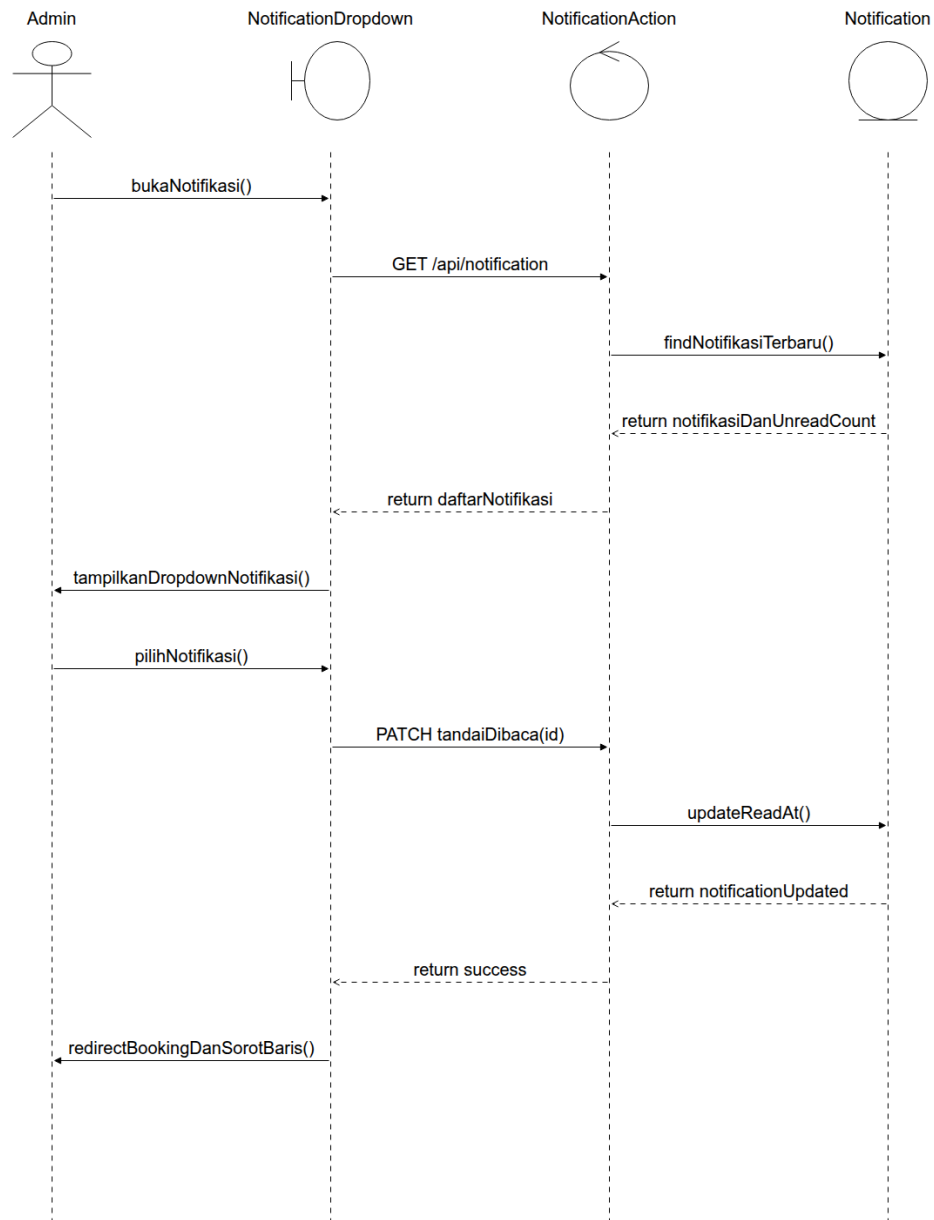
4.2.4.16 Diagram Sequence UC-16 - Mengelola Pengaturan Situs



Gambar 4.22 Diagram Sequence UC-16 - Mengelola Pengaturan Situs

Penjelasan Alur: Admin membuka halaman pengaturan situs dan sistem mengambil data dari entitas SiteSetting. Admin dapat memperbaiki nama bisnis, nomor WhatsApp, alamat, jam operasional, dan informasi kontak lainnya. Sistem memvalidasi data dan menyimpannya dalam bentuk pasangan key dan value, kemudian menampilkan notifikasi bahwa pengaturan berhasil diperbarui.

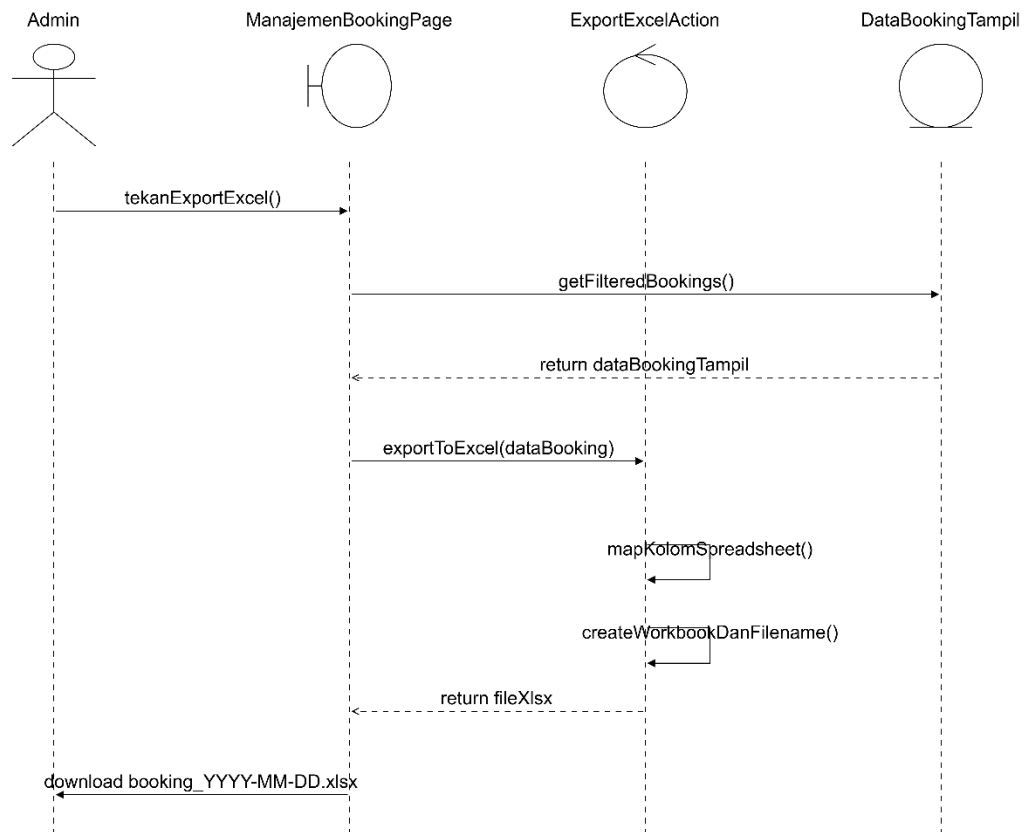
4.2.4.17 Diagram Sequence UC-17 - Melihat Notifikasi



Gambar 4.23 Diagram Sequence UC-17 - Melihat Notifikasi

Penjelasan Alur: Admin membuka menu notifikasi pada *dashboard*. Sistem mengambil notifikasi terbaru dan menghitung jumlah pesan yang belum dibaca. Ketika admin memilih notifikasi *booking*, sistem mengubah statusnya menjadi telah dibaca dan mengarahkan admin menuju halaman manajemen *booking* dengan pesanan terkait dalam keadaan disorot.

4.2.4.18 Diagram Sequence UC-18 - Mengekspor Data Excel



Gambar 4.24 Diagram Sequence UC-18 - Mengekspor Data Excel

Penjelasan Alur: Admin menekan tombol ekspor pada halaman manajemen armada atau *booking*. Sistem mengambil data yang sedang ditampilkan dan memetakan atributnya menjadi kolom *spreadsheet*. *Library SheetJS/xlsx* membentuk workbook dan nama berkas, kemudian sistem memulai pengunduhan *file* Excel secara otomatis ke perangkat admin.

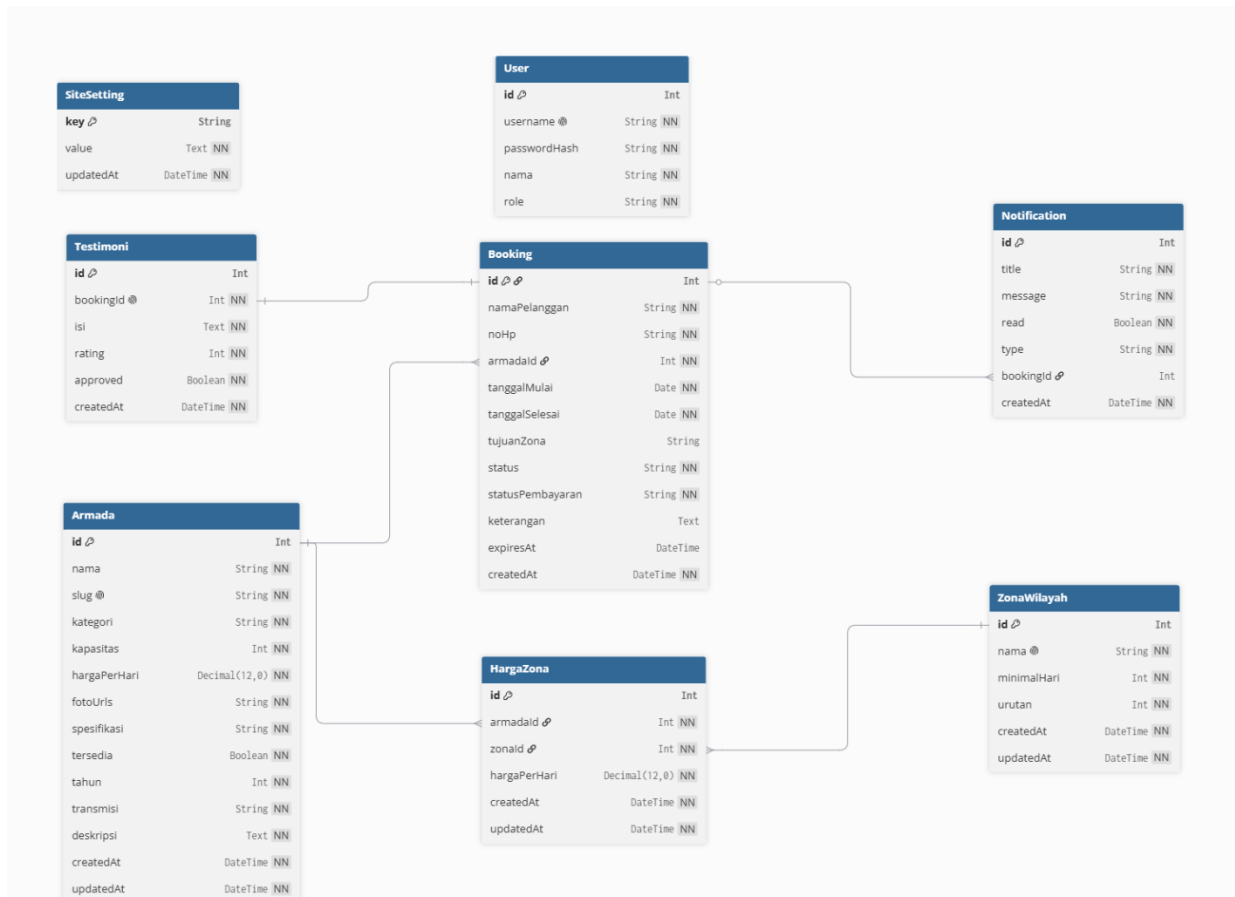
4.2.5 Perancangan Basis Data

Perancangan basis data merupakan tahapan untuk memodelkan struktur penyimpanan data beserta hubungan antarentitas yang dibutuhkan oleh sistem. Pada pengembangan *website* Rahayu Trans Travel, basis data dirancang menggunakan PostgreSQL sebagai sistem manajemen basis data relasional atau Relational *Database* Management System (RDBMS).

Infrastruktur basis data menggunakan layanan Neon Serverless Postgres sebagai penyimpanan berbasis komputasi awan. Penggunaan Neon memungkinkan basis data diakses oleh aplikasi yang telah di-*deploy* serta mendukung pengelolaan sumber daya secara fleksibel. Sistem juga menggunakan Prisma ORM (*Object-Relational Mapping*) sebagai penghubung antara *backend* Next.js dan basis data PostgreSQL. Prisma ORM digunakan untuk mendefinisikan skema, relasi, tipe data, dan menjalankan operasi kueri secara terstruktur.

4.2.5.1 Entity Relationship Diagram

Entity Relationship Diagram (ERD) digunakan untuk memvisualisasikan struktur entitas, atribut, kunci utama, kunci tamu, dan kardinalitas hubungan antarentitas pada basis data sistem.



Gambar 4.25 Entity Relationship Diagram

Berdasarkan rancangan basis data pada Gambar 4.25, sistem memiliki delapan entitas dengan fungsi sebagai berikut:

1. Entitas User

Entitas User digunakan untuk menyimpan data autentikasi pengelola sistem. Atribut yang disimpan meliputi id, username, passwordHash, nama, dan role. Atribut username bersifat unik, sedangkan passwordHash menyimpan kata sandi yang telah melalui proses *hashing*.

2. Entitas Armada

Entitas Armada digunakan untuk menyimpan data kendaraan yang disewakan. Atributnya meliputi id, nama, slug, kategori, kapasitas, hargaPerHari, fotoUrls, spesifikasi, tersedia, tahun, transmisi, deskripsi, createdAt, dan updatedAt. Atribut hargaPerHari berfungsi sebagai harga dasar yang digunakan apabila tarif untuk kombinasi armada dan zona belum tersedia.

3. Entitas ZonaWilayah

Entitas ZonaWilayah digunakan untuk menyimpan referensi wilayah tujuan penyewaan. Atributnya meliputi id, nama, minimalHari, urutan, createdAt, dan updatedAt. Atribut minimalHari digunakan untuk menentukan batas minimal durasi sewa berdasarkan zona tujuan.

4. Entitas HargaZona

Entitas HargaZona merupakan entitas penghubung antara Armada dan ZonaWilayah. Entitas ini menyimpan tarif sewa khusus berdasarkan kombinasi armada dan zona tujuan melalui atribut *armadaId*, *zonaId*, dan *hargaPerHari*. Kombinasi *armadaId* dan *zonaId* bersifat unik agar satu armada hanya memiliki satu tarif untuk setiap zona.

5. Entitas Booking

Entitas *Booking* digunakan untuk menyimpan transaksi reservasi pelanggan. Atributnya meliputi *id*, *namaPelanggan*, *noHp*, *armadaId*, *tanggalMulai*, *tanggalSelesai*, *tujuanZona*, *status*, *statusPembayaran*, *keterangan*, *expiresAt*, dan *createdAt*. Atribut *expiresAt* menyimpan batas waktu konfirmasi selama 24 jam. *Booking* yang masih berstatus *pending* setelah melewati waktu tersebut dapat diubah menjadi dibatalkan agar jadwal armada tersedia kembali.

6. Entitas Testimoni

Entitas Testimoni digunakan untuk menyimpan ulasan pelanggan yang terhubung dengan transaksi *Booking*. Data yang disimpan meliputi *bookingId*, *isi*, *rating*, *approved*, dan *createdAt*. Atribut *bookingId* bersifat unik sehingga satu *booking* hanya dapat memiliki satu testimoni.

7. Entitas Notification

Entitas Notification digunakan untuk menyimpan pemberitahuan kepada admin, terutama ketika terdapat *booking* baru. Atributnya meliputi *id*, *title*, *message*, *read*, *type*, *bookingId*, dan *createdAt*. Atribut *bookingId* bersifat opsional sehingga notifikasi dapat dihubungkan dengan *booking* tertentu atau digunakan sebagai notifikasi sistem umum.

8. Entitas SiteSetting

Entitas SiteSetting digunakan untuk menyimpan konfigurasi dinamis *website* dalam bentuk pasangan *key* dan *value*. Data yang disimpan dapat berupa nomor WhatsApp admin, informasi kontak, nama bisnis, alamat, dan konfigurasi lainnya tanpa harus mengubah kode aplikasi.

4.2.5.2 Relasi Antar Entitas

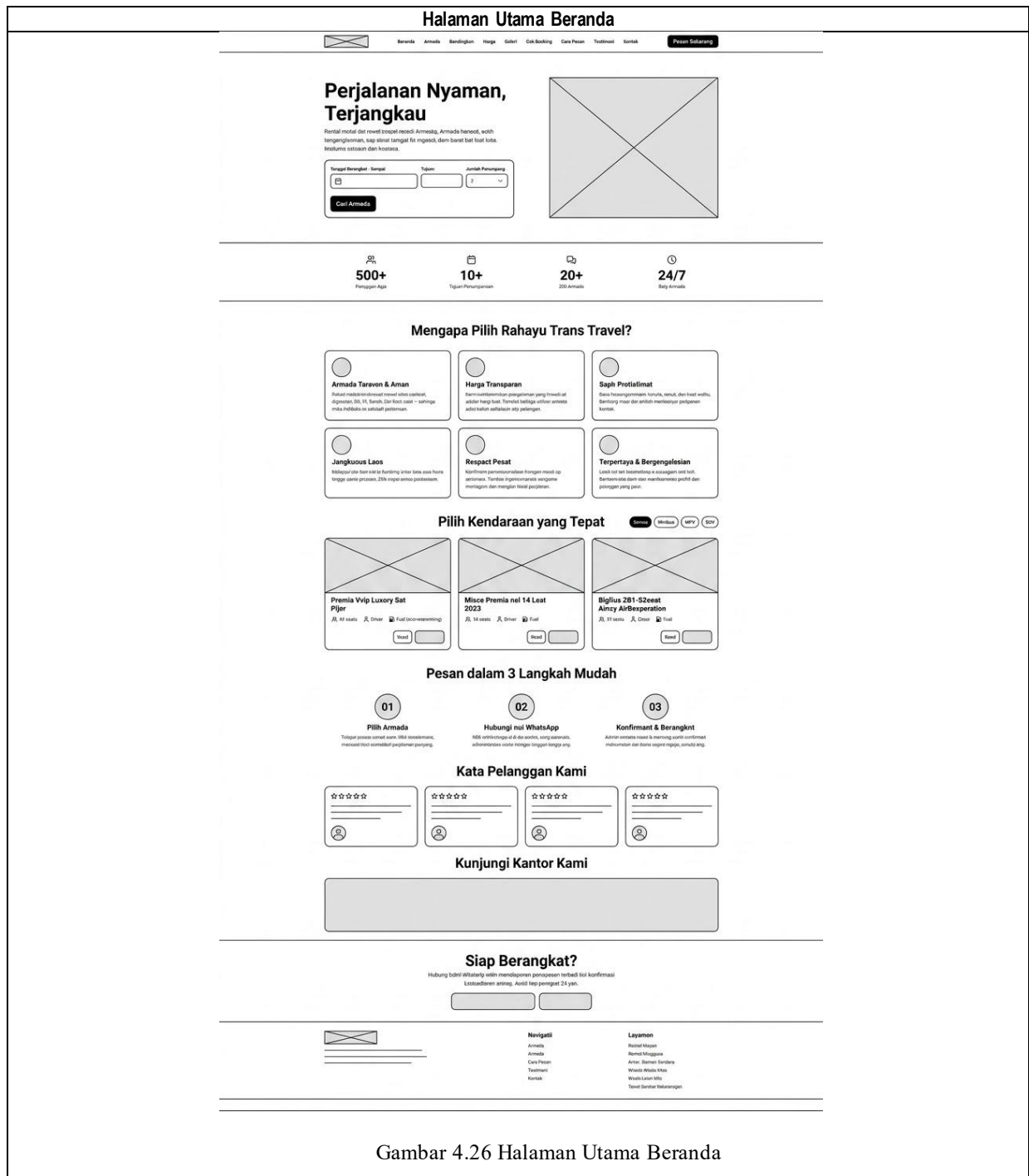
Relasi antarentitas dibentuk menggunakan *primary key* dan *foreign key* untuk menjaga integritas serta konsistensi data. Relasi yang terdapat pada rancangan basis data adalah sebagai berikut:

1. Entitas Armada memiliki relasi *One-to-Many* dengan *Booking*. Satu armada dapat memiliki banyak riwayat *booking*, sedangkan satu *booking* hanya menggunakan satu armada.
2. Entitas Armada memiliki relasi *One-to-Many* dengan HargaZona. Satu armada dapat memiliki banyak tarif berdasarkan zona tujuan.
3. Entitas ZonaWilayah memiliki relasi *One-to-Many* dengan HargaZona. Satu zona wilayah dapat digunakan pada pengaturan harga untuk banyak armada.
4. Entitas Armada dan ZonaWilayah secara tidak langsung memiliki relasi *Many-to-Many* melalui entitas penghubung HargaZona. Relasi ini memungkinkan setiap armada memiliki tarif berbeda pada setiap zona.

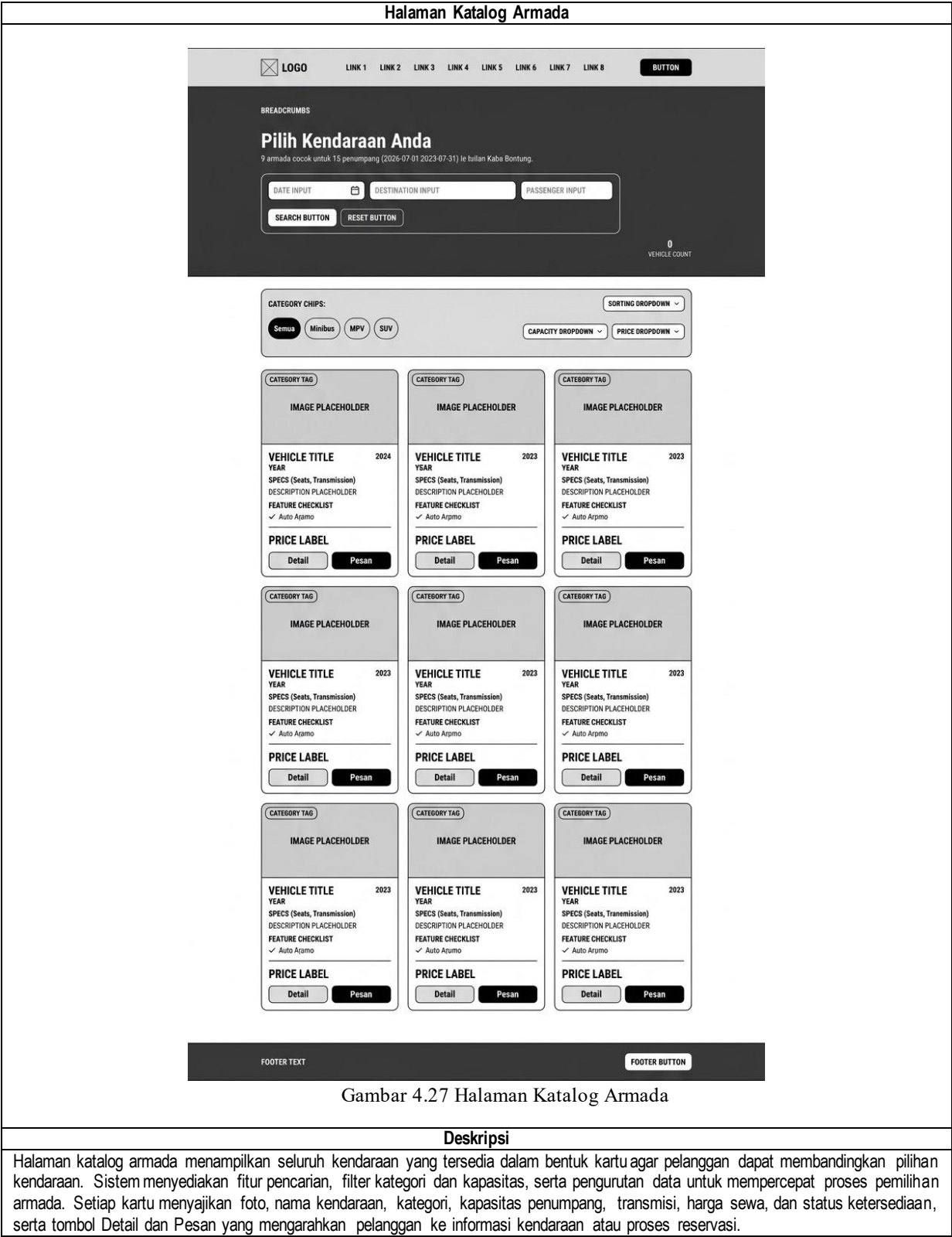
5. Entitas *Booking* memiliki relasi *One-to-Zero-or-One* dengan Testimoni. Satu *booking* dapat memiliki maksimal satu testimoni, sedangkan setiap testimoni harus terhubung dengan satu *booking*.
6. Entitas *Booking* memiliki relasi *One-to-Many* dengan Notification. Satu *booking* dapat menghasilkan beberapa notifikasi, sedangkan satu notifikasi hanya dapat terhubung dengan satu *booking* atau tidak memiliki hubungan *booking* apabila berupa notifikasi umum.
7. Entitas User dan SiteSetting tidak memiliki *foreign key* dengan entitas lain. User digunakan secara mandiri untuk autentikasi admin, sedangkan SiteSetting digunakan sebagai penyimpanan konfigurasi *website*.

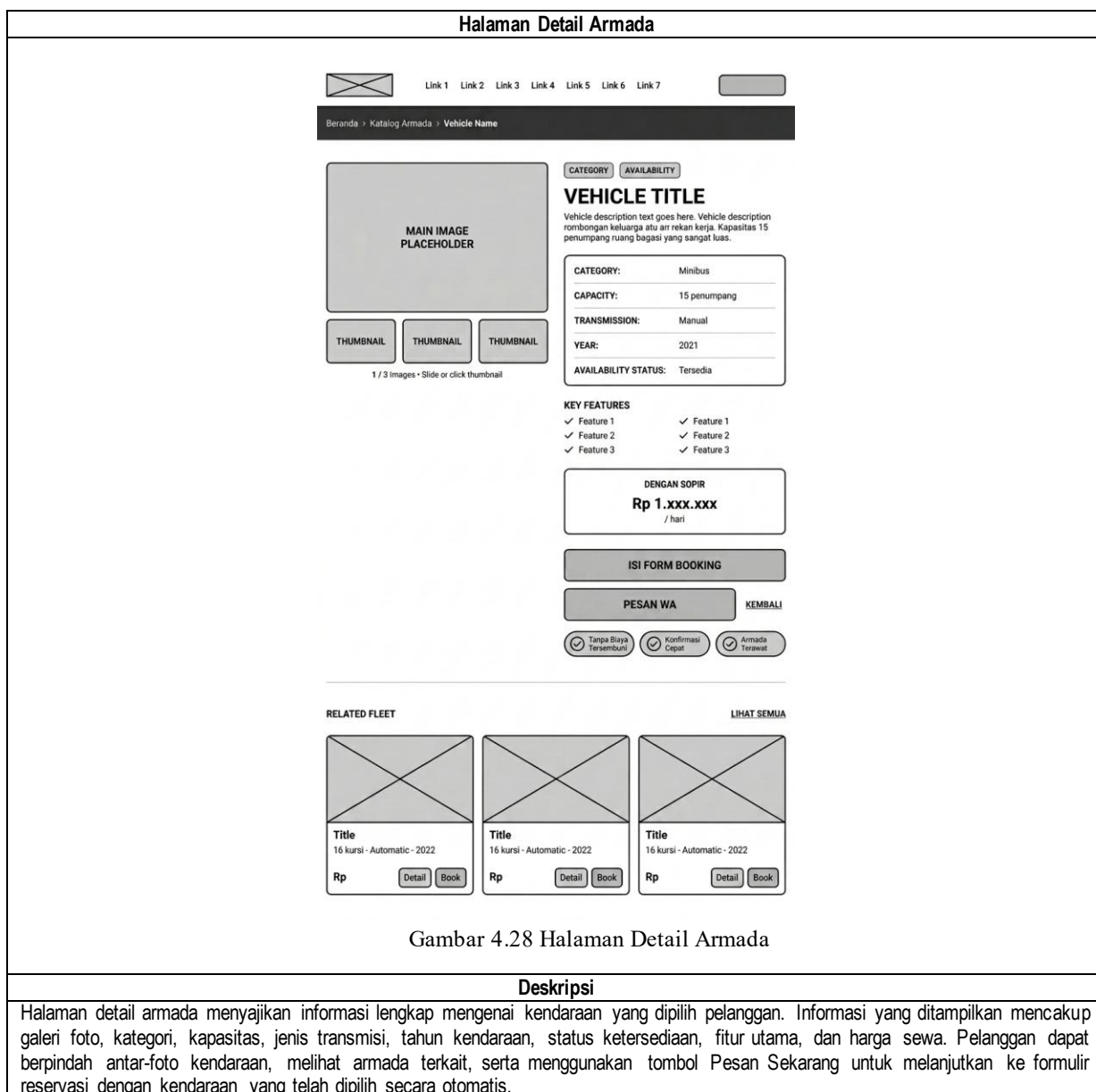
Pada rancangan ini, tujuanZona pada entitas *Booking* disimpan dalam bentuk teks sebagai rekaman zona tujuan ketika transaksi dibuat. Sementara itu, perhitungan harga dilakukan dengan mengambil tarif dari entitas HargaZona berdasarkan kombinasi armada dan zona yang dipilih.

4.2.6 Perancangan Antarmuka



Halaman utama beranda merupakan halaman awal yang diakses pelanggan untuk memperoleh gambaran umum layanan Rahayu Trans Travel. Bagian hero menampilkan pesan utama layanan dan tombol ajakan untuk melihat armada atau melakukan pemesanan. Di bawahnya tersedia daftar armada unggulan yang memuat foto, nama kendaraan, kapasitas, transmisi, harga sewa, serta tombol untuk melihat detail dan memesan kendaraan. Halaman ini juga menyediakan navigasi menuju informasi armada, harga, galeri, pengecekan booking, testimoni, dan kontak sehingga pelanggan dapat mengakses fitur utama secara cepat.





Halaman Formulir Reservasi

1

 Isi Data

2

 Konfirmasi

01 Pilih Kendaraan & Jadwal

Kendaraan *

Tanggal Sewa (Mulai & Selesai) *

Zona Wilayah Tujuan *

02 Data Pemesan

Nama Lengkap *

Nomor HP / WhatsApp *

Keterangan (opsional)

Lanjut ke Ringkasan

PRATINJAU PESANAN

Kendaraan	Toyota Hiace (15 kursi)
Sopir	Dengan Sopir (wajib)
Mulai	9 Juni 2026
Selesai	11 Juni 2026
Durasi	3 hari
Zona Tujuan	Dalam Kota Bandung

Rp 3.600.000

belum termasuk BBM & tol

Gambar 4.29 Halaman Formulir Reservasi

Deskripsi

Halaman Formulir Ringkasan dan Konfirmasi Reservasi

LOGO

[Beranda](#) > [Kantor Bank](#)

Formulir Reservasi

In data di bawah, sistem ini akan diinformasikan dan di kirim ke WhatsApp.com untuk southwood user.

Step 1

2 Konfirmasi

Booking Details

Vehicle Model	VehiclePacar
Category	Monary
Capacity	3 orang
Transmission	Bangirai Rojo (semirak)
Driver	Delam Rota Bandung
Zone	5 Jan 0015
Start Date	01 Mar 2024
End Date	April 2023
Duration	1 IR
Customer Phone	07000000000

Total Price: Rp X,XXX,XXX

Edit Pesanan

Kirim ke WhatsApp

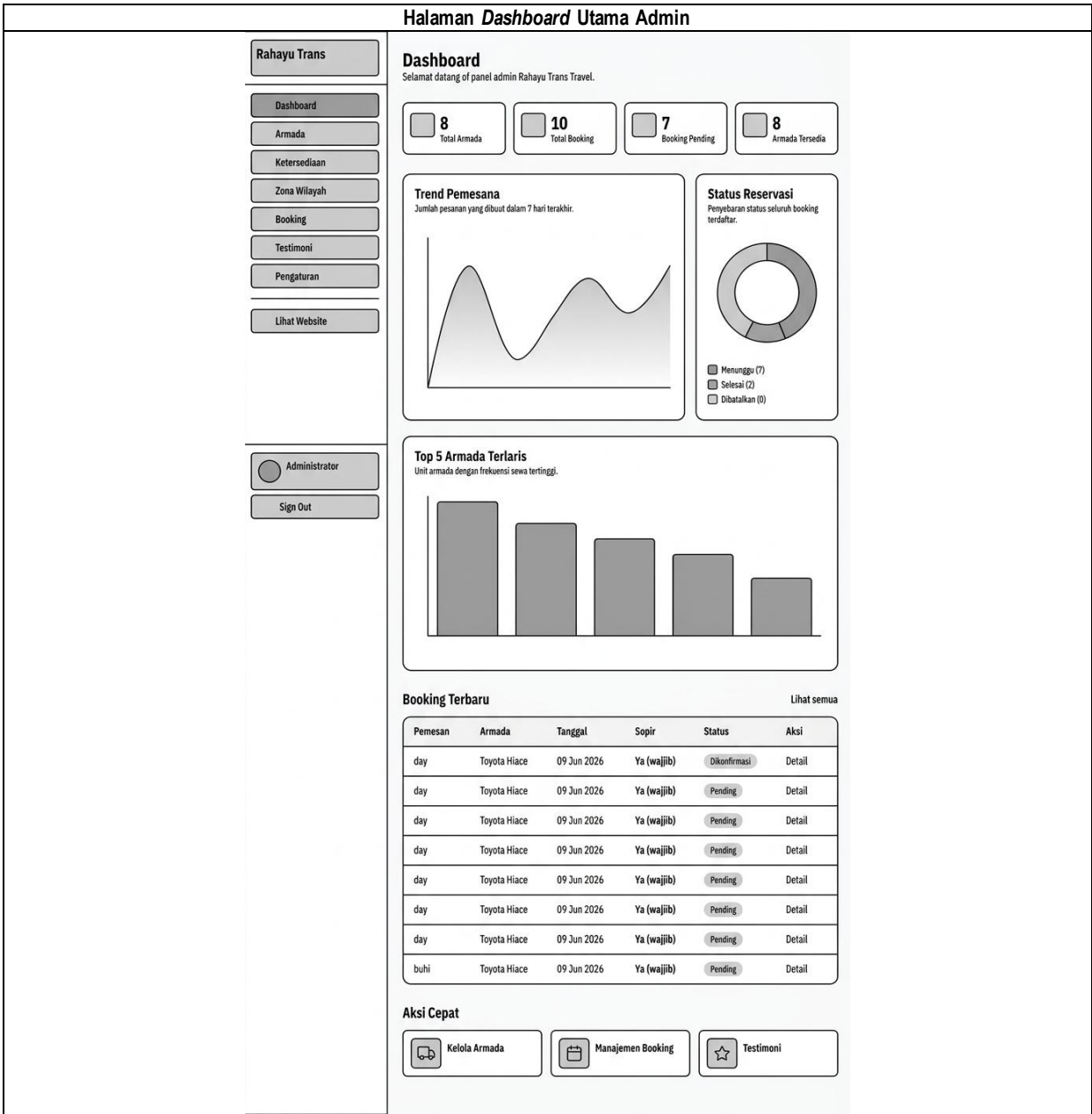
Gambar 4.30 Halaman Formulir Ringkasan dan Konfirmasi Reservasi

Deskripsi

Halaman ringkasan dan konfirmasi reservasi merupakan tahap kedua sebelum data pemesanan dikirim kepada admin. Sistem menampilkan kembali detail kendaraan, kategori, kapasitas, transmisi, zona tujuan, tanggal sewa, durasi, nomor pelanggan, serta total estimasi biaya agar dapat diperiksa terlebih dahulu. Sistem juga membentuk pratinjau pesan WhatsApp dalam format yang baku. Pelanggan dapat memilih Edit Pesanan untuk memperbaiki data atau Kirim ke WhatsApp untuk menyimpan reservasi dan membuka percakapan dengan admin.

Halaman cek status pesanan memungkinkan pelanggan melacak reservasi secara mandiri menggunakan nomor HP pemesan dan PIN verifikasi *booking*. Jika data sesuai, sistem menampilkan identitas *booking*, armada, periode sewa, tujuan, status pesanan, serta tahapan proses yang telah berlangsung. Untuk pesanan yang telah selesai, halaman ini menyediakan formulir penilaian berupa *rating* bintang dan ulasan. Testimoni yang dikirim akan disimpan dengan status menunggu persetujuan admin sebelum ditampilkan pada halaman publik.

Halaman <i>Login Admin</i>
 Rahayu Trans Travel Panel Administrasi Masuk ke Akun Masukkan kredensial Anda untuk melanjutkan. Username Password Masuk  Halaman ini hanya dapat diakses oleh staf yang berwenang. Hubungi administrator jika Anda membutuhkan akses. © 2026 Rahayu Trans Travel. Hak cipta dilindungi.



Halaman Manajemen Armada

Dashboard
Armada

Ketersediaan
Zona Wilayah

Booking
Testimoni

Pengaturan
Lihat Website

Administrator
Sign Out

Armada
8 kendaraan terdaftar

Export Excel

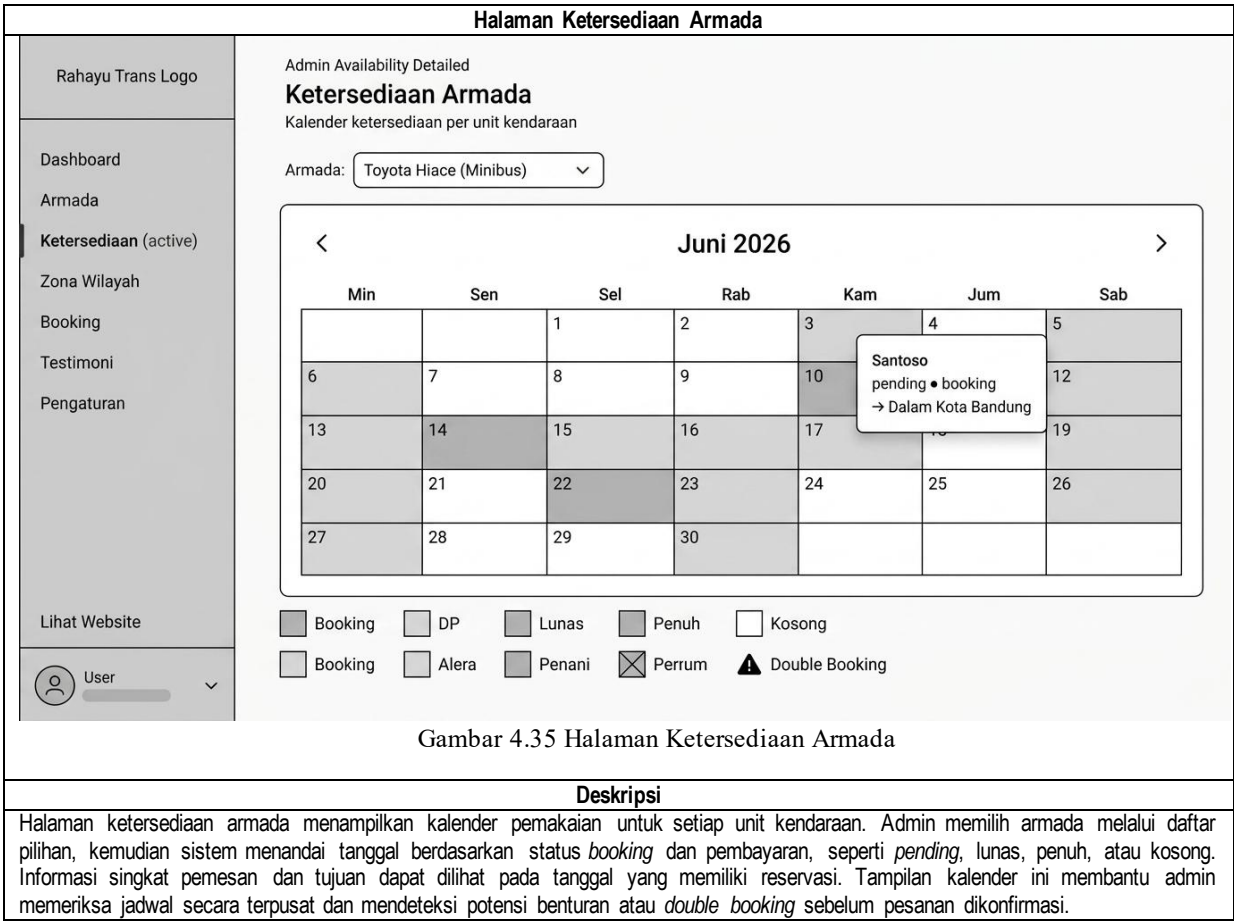
+ Tambah Armada

Nama	Kategori	Kapasitas	Harga/Hari	Status	Aksi
Toyota Hiace 2021 - Manual	Minibus	15 orang	Rp 1.200.000	☒ Tersedia	Harga Zona Edit Hapus
Mercedes Sprinter 2022 - Automatic	Minibus	16 orang	Rp 2.000.000	☒ Tersedia	Harga Zona Edit Hapus
Isuzu Elf 2022 - Manual	Minibus	12 orang	Rp 850.000	☒ Tersedia	Harga Zona Edit Hapus
Toyota Alphard 2023 - Automatic	MPV	7 orang	Rp 1.500.000	☒ Tersedia	Harga Zona Edit Hapus
Toyota Innova Reborn 2022 - Automatic	MPV	7 orang	Rp 800.000	☒ Tersedia	Harga Zona Edit Hapus
Suzuki Ertiga 2021 - Manual	MPV	7 orang	Rp 550.000	☒ Tersedia	Harga Zona Edit Hapus
Toyota Fortuner 2022 - Automatic	SUV	7 orang	Rp 900.000	☒ Tersedia	Harga Zona Edit Hapus
Mitsubishi Pajero Sport 2023 - Automatic	SUV	7 orang	Rp 1.100.000	☒ Tersedia	Harga Zona Edit Hapus

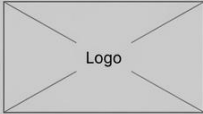
Gambar 4.34 Halaman Manajemen Armada

Deskripsi

Halaman manajemen armada digunakan admin untuk mengelola data kendaraan yang ditampilkan pada *website*. Tabel memuat nama kendaraan, kategori, kapasitas, harga per hari, dan status ketersediaan. Admin dapat menambah armada baru, mengubah informasi kendaraan, mengatur tarif berdasarkan zona, menonaktifkan atau mengaktifkan status armada, serta menghapus data yang tidak digunakan. Fitur ekspor Excel disediakan untuk menghasilkan rekap data armada.



Halaman Manajemen Booking



☐ Dashboard
☐ Armada
☐ Ketersediaan
☐ Zona Wilayah
☒ **Booking**
☐ Testimoni
☐ Pengaturan

[Lihat Website](#)

 Administrator
[Sign Out](#)

Manajemen Booking

10 total pemesanan

☐ Pending: [N]
☐ Dikonfirmasi: [N]
☐ Selesai: [N]
☐ Dibatalkan: [N]

Semua
Pending
Dikonfirmasi
Selesai
Dibatalkan

[Export Excel](#)

ID	Pemesan	No. HP	Armada	Tanggal Mulai	Tanggal Selesai	Durasi	Sopir	Status	Pembayaran
H01	Pemesan	073 4245677	Armada	28 Juni 2021	21 Juni 2021	3 durr	Sopir	Dikonfirmasi	Lunas
H02	Pemesan	073 4245677	Armada	25 Juni 2021	31 Juni 2021	3 durr	Sopir	Pending	Lunas
H03	Pemesan	073 4245677	Armada	28 Juni 2021	21 Juni 2021	3 durr	Sopir	Pending	Lunas
H04	Pemesan	073 4245677	Armada	29 Juni 2021	31 Juni 2021	3 durr	Sopir	Selesai	Lunas
H05	Pemesan	073 4245677	Armada	29 Juni 2021	31 Juni 2021	2 durr	Sopir	Selesai	Lunas
H06	Pemesan	073 4245677	Armada	29 Juni 2021	31 Juni 2021	3 durr	Sopir	Selesai	Lunas
H07	Pemesan	073 4245677	Armada	28 Juni 2021	21 Juni 2021	3 durr	Sopir	Selesai	Lunas
H08	Pemesan	073 4245677	Armada	39 Juni 2021	31 Juni 2021	3 durr	Sopir	Belum Lunas	Lunas
H09	Pemesan	073 4245677	Armada	28 Juni 2021	31 Juni 2021	3 durr	Sopir	Belum Lunas	Lunas

Gambar 4.37 Halaman Manajemen Booking

Deskripsi

Halaman manajemen *booking* menampilkan seluruh transaksi reservasi yang masuk ke dalam sistem. Admin dapat melihat ringkasan jumlah pesanan berdasarkan status, memfilter data *pending*, dikonfirmasi, selesai, atau dibatalkan, serta memeriksa identitas pemesan, nomor HP, armada, tanggal sewa, durasi, sopir, status *booking*, dan pembayaran. Melalui halaman ini admin dapat memperbaiki status operasional maupun verifikasi pembayaran serta mengeksport data *booking* ke dalam berkas Excel untuk kebutuhan rekapitulasi. *Booking pending* yang melewati batas *expiresAt* akan diubah menjadi dibatalkan sehingga tanggal armada dapat tersedia kembali.

Dashboard

Armada

Ketersediaan

Zona Wilayah

Booking

Testimoni

Pengaturan

Lihat Website

Administrator

Administrator

Sign Out

Halaman Manajemen Testimoni

Testimoni

3 ulasan - 1 menunggu persetujuan

Export Excel

Semua 3

Menunggu 1

Disetujui 2

Testimoni dari pelanggan masuk dengan status **Menunggu**. Klik **Setujui** agar tampil di halaman publik.

★★★★★

Disetujui

“Pesan via WhatsApp sangat mudah, langsung dibalas das cepat. Mobil Innova-nyaman u nyaman untuk perjalanan Bandung-Yogyakarta. Pasti bakal lagi!”

Armada: Toyota Innova Reborn

Tujuan: Dalam Kota Bandung

Ahmad Fauzi

08 Juni 2026

Batalan

★★★★☆

Menunggu

“Sudah beberapa kali pakai Rahayu Trans untur keperluan keluarga, jujur, tidak da biaya tiba-tiba. Sangat rekomendasi untuk cari rental terpercaya.

Armada: Toyota Innova Reborn

Tujuan: Dalam Kota Bandung

Siti Rahayu

08 Juni 2026

Setujui

★★★★☆

Disetujui

“Pelayanan sangat memuaskan! Sopirnya dan tepat waktu. Mobil dan ber-AC. Kami sewe Hiace untuk 12 orang, semua Puncak, semua nyaman dan aman.

Armada: Toyota Hiace

Tujuan: Dalam Kota Bandung

Budi Santoso

08 Juni 2026

Batalan

Gambar 4.38 Halaman Manajemen Testimoni

Deskripsi

Halaman manajemen testimoni digunakan admin untuk memoderasi ulasan yang dikirim pelanggan setelah pesanan selesai. Setiap kartu menampilkan *rating*, isi ulasan, nama pelanggan, armada, tujuan, tanggal, dan status publikasi. Admin dapat memfilter testimoni yang masih menunggu atau telah disetujui, kemudian memilih tindakan Setujui agar ulasan tampil pada *website*, Batalkan untuk menarik persetujuan, atau Hapus untuk menghilangkan data yang tidak layak. Data testimoni juga dapat diekspor ke Excel.

Halaman Pengaturan Situs	
LOGO Dashboard Armada Ketersediaan Zona Wilayah Booking Testimoni Pengaturan Lihat Website Sign Out	Pengaturan Situs Kelola informasi kontak dan identitas bisnis Anda. Kontak Nomor WhatsApp (format internasional) Tampilan Nomor WhatsApp Alamat Google Map Embed URL (src) Jam Operasional Informasi Bisnis Nama Bisnis Tagline Email Simpan Pengaturan
Gambar 4.39 Halaman Pengaturan Situs	
Deskripsi Halaman pengaturan situs digunakan admin untuk memperbarui informasi umum yang ditampilkan pada <i>website</i> tanpa mengubah kode program. Data yang dapat dikelola meliputi nomor WhatsApp, teks tampilan kontak, alamat, tautan Google Maps, jam operasional, nama bisnis, <i>tagline</i>, dan email. Setelah tombol Simpan Pengaturan dipilih, perubahan disimpan ke basis data dan digunakan pada halaman publik maupun proses WhatsApp <i>Click to Chat</i> sehingga informasi kontak selalu sesuai dengan kondisi operasional terbaru.	

BAB 5

IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas implementasi dan pengujian sistem informasi reservasi kendaraan pada Rahayu Trans Travel Bandung. Implementasi dilakukan berdasarkan hasil analisis dan perancangan pada Bab 4, meliputi lingkungan pengembangan, struktur aplikasi, basis data, *route handler*, antarmuka pelanggan, *dashboard* admin, serta fitur utama sistem. Fitur tersebut mencakup pencarian dan katalog armada, validasi ketersediaan berdasarkan armada, perhitungan harga berdasarkan zona, integrasi WhatsApp *Click to Chat*, pembatalan otomatis *booking pending* setelah 24 jam, pengecekan status pesanan, pengelolaan data oleh admin, dan ekspor data ke Excel. Pengujian fungsional dirancang menggunakan metode *Black Box Testing* untuk memastikan keluaran sistem sesuai dengan kebutuhan pengguna dan perangkat lunak yang telah ditetapkan.

5.1 Implementasi Sistem

Implementasi sistem merupakan tahap penerjemahan hasil analisis dan perancangan menjadi aplikasi web yang dapat digunakan oleh pelanggan dan admin Rahayu Trans Travel. Sistem dibangun menggunakan *framework* Next.js dengan bahasa TypeScript, basis data PostgreSQL, serta Prisma ORM sebagai penghubung antara aplikasi dan basis data. Sistem menyediakan halaman publik untuk pelanggan serta *dashboard* admin untuk pengelolaan data operasional.

5.1.1 Lingkungan Implementasi

Lingkungan implementasi terdiri dari perangkat keras dan perangkat lunak yang digunakan selama proses pengembangan sistem.

Tabel 5.1 Kebutuhan Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	Laptop	MSI Modern 14 B11MOU
2	Prosesor	I5 1155G7
3	Memori	16 GB
4	Penyimpanan	SSD 512 GB

Tabel 5.2 Kebutuhan Perangkat Lunak

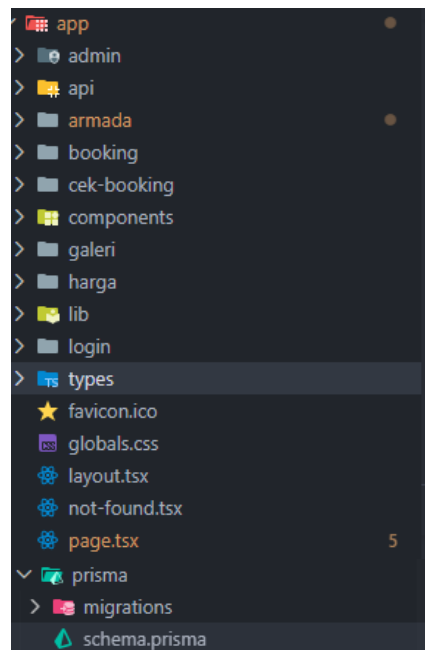
No	Perangkat Lunak	Keterangan
1	Windows	Sistem operasi pengembangan
2	Visual Studio Code	Code editor
3	Node.js	Runtime JavaScript
4	Next.js 16.2.5	Framework utama aplikasi web
5	React 19.2.4	Library antarmuka pengguna
6	TypeScript	Bahasa pemrograman utama
7	Tailwind CSS	Styling antarmuka
8	PostgreSQL	Sistem manajemen basis data
9	Prisma ORM	Pengelolaan skema dan query basis data
10	Auth.js / NextAuth	Autentikasi admin
11	Zod	Validasi data
12	SheetJS / xlsx	Ekspor data ke Excel
13	Vercel	Deployment aplikasi web
14	Neon PostgreSQL	Deployment basis data cloud

5.1.2 Struktur *Folder* dan Berkas

Struktur *folder* aplikasi disusun berdasarkan pendekatan *App Router* pada Next.js. Pembagian *folder* dilakukan untuk memisahkan halaman publik, halaman admin, API, komponen antarmuka, pustaka pendukung, dan konfigurasi basis data.

Tabel 5.3 Struktur *Folder* Aplikasi

No	Folder/Berkas	Fungsi
1	app/	Direktori utama aplikasi Next.js
2	app/admin/	Halaman <i>dashboard</i> admin
3	app/api/	<i>Route handler</i> /API aplikasi
4	app/armada/	Halaman katalog dan detail armada
5	app/booking/	Halaman formulir reservasi
6	app/cek-booking/	Halaman pengecekan status pesanan
7	app/components/	Komponen antarmuka yang digunakan berulang
8	app/lib/	Konfigurasi, helper, dan koneksi Prisma
9	app/login/	Halaman <i>login</i> admin
10	prisma/schema.prisma	Skema basis data
11	auth.ts	Konfigurasi autentikasi admin
12	vercel.json	Konfigurasi <i>cron</i> untuk expiration <i>booking</i>



Gambar 5.1 Struktur *Folder* Aplikasi

5.1.3 Konfigurasi Aplikasi

Konfigurasi aplikasi dilakukan melalui *environment variable* agar data sensitif tidak ditulis langsung di dalam *source code*. Konfigurasi tersebut digunakan untuk koneksi basis data, autentikasi, dan kebutuhan *runtime* aplikasi.

Tabel 5.4 Konfigurasi *Environment Variable*

No	Konfigurasi	Fungsi
1	DATABASE_URL	URL koneksi PostgreSQL
2	AUTH_SECRET	<i>Secret key</i> untuk autentikasi
3	AUTH_URL	URL aplikasi untuk kebutuhan Auth.js
4	CRON_SECRET	<i>Token</i> untuk melindungi <i>endpoint</i> expiration <i>booking</i>
5	NEXT_PUBLIC_*	Konfigurasi publik sisi <i>client</i> apabila diperlukan

5.1.4 Implementasi Basis Data

Basis data diimplementasikan menggunakan PostgreSQL dan Prisma ORM. Skema yang digunakan terdiri atas delapan entitas utama. Entitas *Booking* menyimpan *expiresAt* sebagai batas konfirmasi selama 24 jam. Entitas *HargaZona* menghubungkan *Armada* dan *ZonaWilayah* untuk menyimpan tarif harian per kombinasi armada dan zona.

Tabel 5.5 Implementasi Entitas Basis Data

No	Entitas	Implementasi
1	Armada	Menyimpan identitas, kategori, kapasitas, harga dasar, foto, spesifikasi, tahun, transmisi, dan status ketersediaan kendaraan.
2	Booking	Menyimpan data pelanggan, armada, tanggal sewa, zona tujuan, status operasional, status pembayaran, keterangan, <i>createdAt</i> , dan <i>expiresAt</i> .
3	ZonaWilayah	Menyimpan nama zona, minimal hari sewa, dan urutan tampilan.
4	HargaZona	Menyimpan tarif per hari berdasarkan kombinasi armada dan zona.
5	Testimoni	Menyimpan <i>rating</i> dan ulasan yang terhubung dengan satu <i>booking</i> .
6	User	Menyimpan akun admin dan <i>password</i> yang telah di-hash.
7	SiteSetting	Menyimpan pengaturan situs dalam bentuk key dan value.
8	Notification	Menyimpan notifikasi sistem, terutama <i>booking</i> baru untuk admin.

5.1.5 Implementasi Route Handler dan API

Route handler menjadi penghubung antara antarmuka pengguna dan basis data. *Endpoint* publik digunakan untuk menampilkan data serta membuat reservasi, sedangkan operasi pengelolaan dilindungi sesi admin.

Tabel 5.6 Implementasi Route Handler dan API

No	Endpoint	Method HTTP	Fungsi
1	/api/armada	GET, POST	GET mengambil daftar armada untuk admin, sedangkan POST menambahkan armada baru.
2	/api/armada/[id]	GET, PUT, DELETE	Mengambil, memperbaiki, atau menghapus data armada tertentu berdasarkan ID.
3	/api/armada/[id]/availability	GET	Mengambil booking dan tanggal yang telah dipesan berdasarkan armada serta menjalankan pemeriksaan booking kedaluwarsa.
4	/api/armada/[id]/harga-zona	GET, PUT	Mengambil dan menyimpan tarif per zona untuk armada tertentu.
5	/api/booking	GET, POST	GET mengambil daftar booking untuk admin, sedangkan POST memvalidasi dan menyimpan reservasi pelanggan.
6	/api/booking/[id]	PATCH, DELETE	Memperbarui status operasional atau pembayaran serta menghapus booking tertentu.
7	/api/booking/lookup	POST	Mencari status booking menggunakan nomor HP dan PIN verifikasi yang dikirim melalui request body.
8	/api/cron/expire-bookings	GET	Menjalankan pembatalan booking berstatus pending yang telah melewati <i>expiresAt</i> .
9	/api/zona-wilayah	GET, POST	Mengambil daftar zona wilayah dan menambahkan zona baru beserta minimal hari sewa.
10	/api/zona-wilayah/[id]	PUT, DELETE	Memperbarui atau menghapus zona wilayah tertentu berdasarkan ID.
11	/api/testimoni	GET, POST	Mengambil daftar testimoni dan menyimpan testimoni baru dari pelanggan.
12	/api/testimoni/[id]	PATCH, DELETE	Memperbarui status moderasi, isi, atau rating serta menghapus testimoni tertentu.
13	/api/notification	GET, PATCH, DELETE	Mengambil notifikasi, menandainya sebagai telah dibaca, atau menghapus seluruh notifikasi.
14	/api/settings	GET, POST	Mengambil dan menyimpan pengaturan situs menggunakan mekanisme <i>upsert</i> .
15	/api/auth/[...nextauth]	GET, POST	Menangani autentikasi, pembuatan sesi, pemeriksaan sesi, login, dan logout admin melalui Auth.js.
16	/api/upload	POST	Memvalidasi dan mengunggah foto armada dalam format JPG, PNG, atau WebP dengan ukuran maksimal 5 MB.

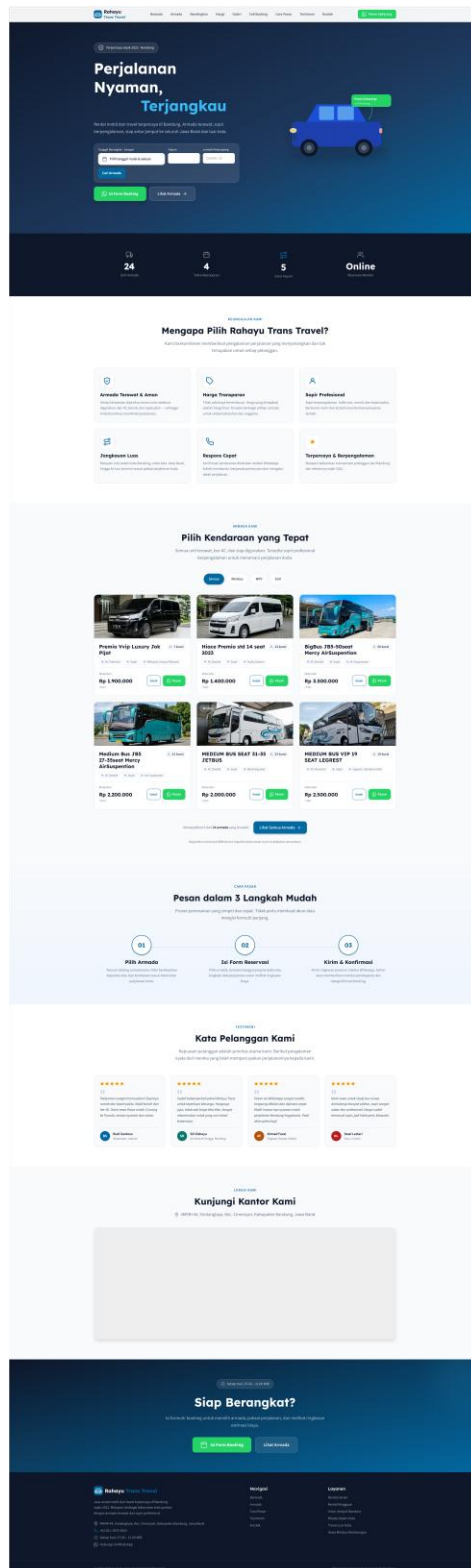
5.2 Implementasi Antarmuka Berdasarkan *Role*

Implementasi antarmuka dibagi berdasarkan *role* pengguna, yaitu pelanggan sebagai pengguna publik dan admin sebagai pengelola sistem. Antarmuka pelanggan difokuskan pada kemudahan memperoleh informasi armada dan melakukan reservasi, sedangkan antarmuka admin difokuskan pada pengelolaan data operasional.

5.2.1 Antarmuka Pelanggan

5.2.1.1 Halaman Beranda

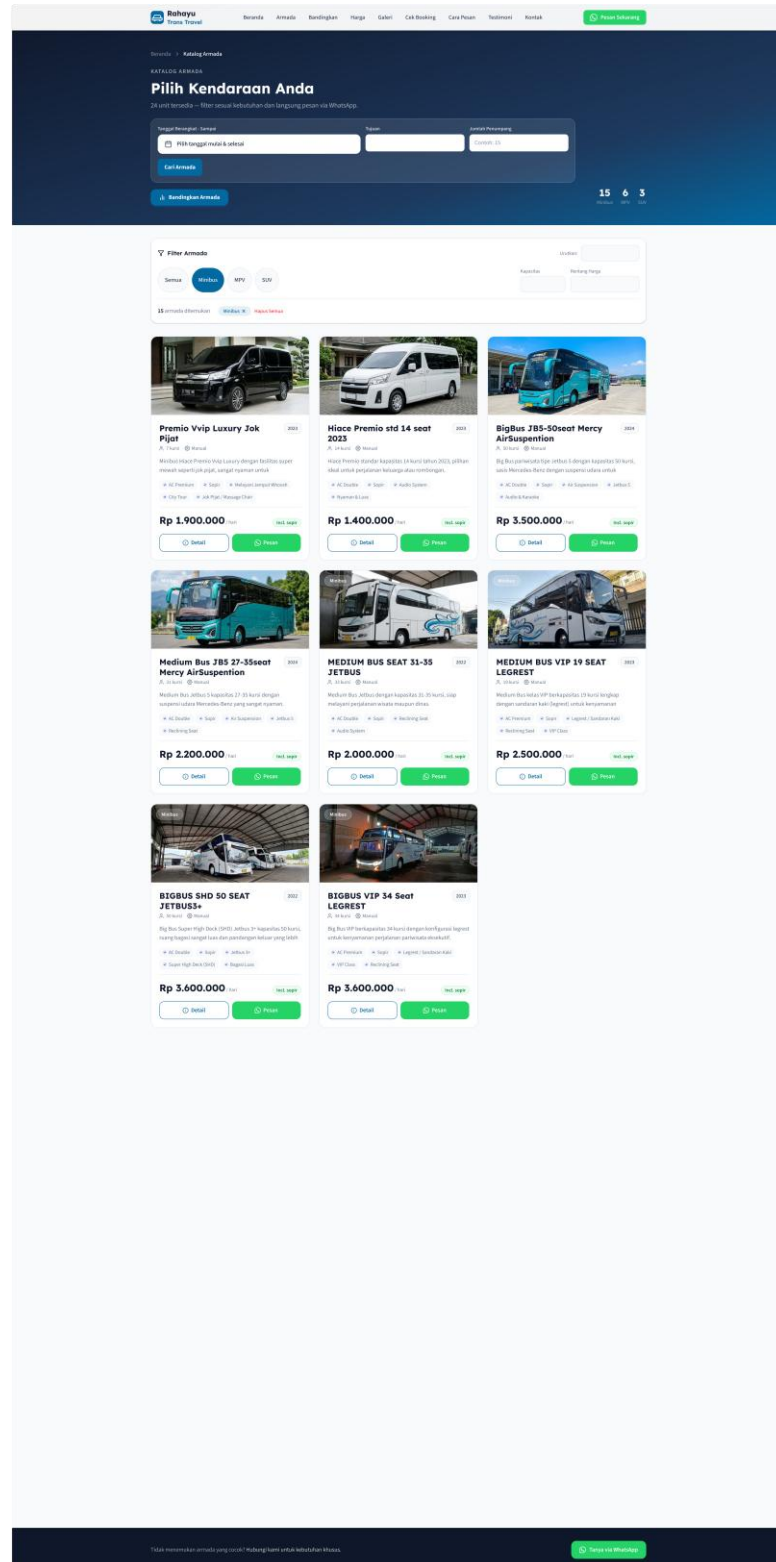
Halaman beranda menampilkan pencarian berdasarkan tanggal perjalanan, zona tujuan, dan jumlah penumpang. Pelanggan juga dapat langsung membuka katalog tanpa menggunakan pencarian tersebut. Hasil pencarian pada katalog menyaring armada berdasarkan kapasitas dan bentrok jadwal, sedangkan filter kategori, kapasitas, rentang harga, dan pengurutan tersedia untuk mempersempit daftar kendaraan.



Gambar 5.2 Halaman Beranda

5.2.1.2 Halaman Armada

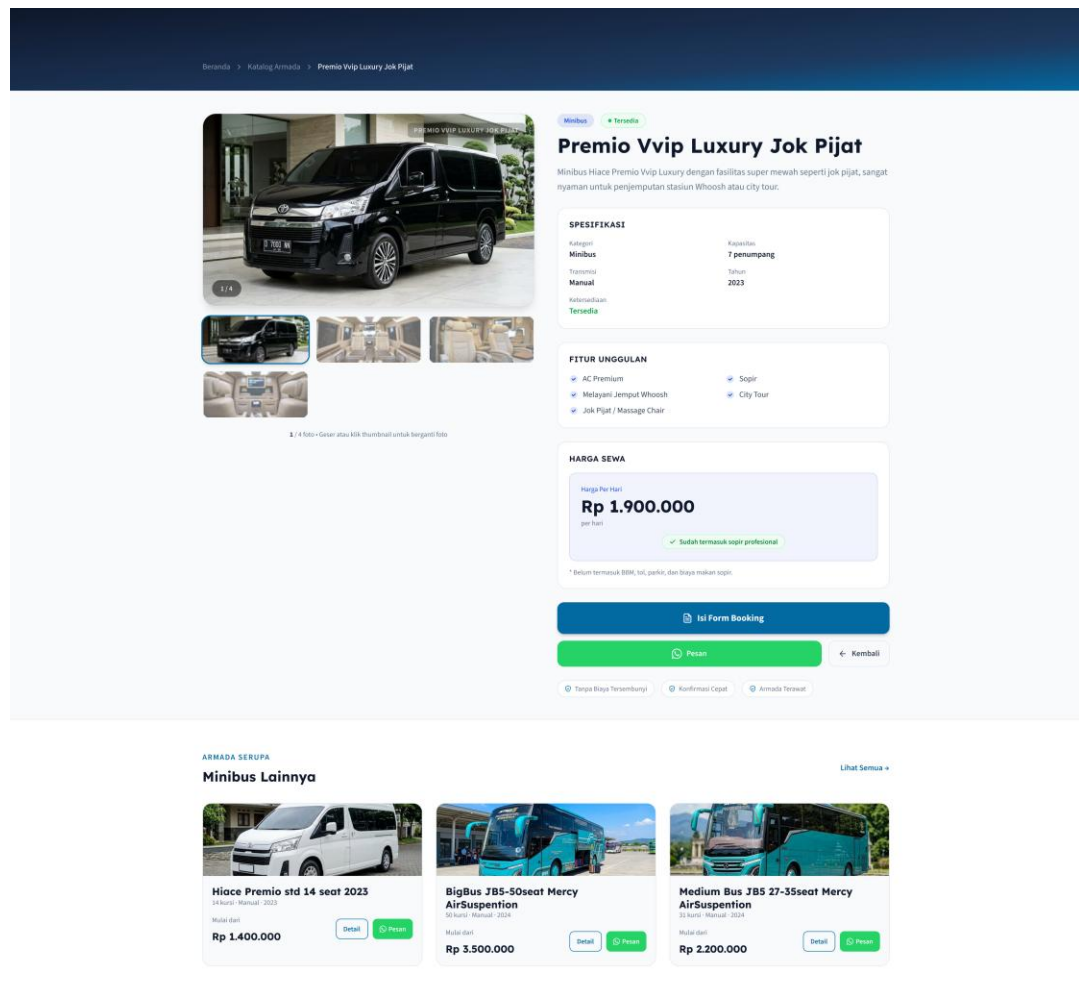
Halaman Armada menampilkan daftar armada yang tersedia. Pelanggan dapat melakukan pencarian berdasarkan rentang tanggal sewa, zona tujuan, dan jumlah penumpang. Sistem juga menyediakan filter kategori, kapasitas, harga, serta pengurutan armada untuk mempermudah pelanggan memilih kendaraan yang sesuai.



Gambar 5.3 Halaman Katalog Armada

5.2.1.3 Halaman Detail

Halaman detail menampilkan foto, spesifikasi, kapasitas, transmisi, tahun, harga dasar, dan tautan menuju formulir reservasi. Armada yang dipilih dikirim melalui parameter URL sehingga terpilih secara otomatis pada formulir. Pada implementasi saat ini, pelanggan masih dapat mengganti armada melalui pilihan kendaraan pada formulir.



Gambar 5.4 Halaman Detail Armada

5.2.1.4 Halaman Formulir Reservasi

Formulir reservasi memuat armada, rentang tanggal, zona tujuan, nama pelanggan, nomor HP, dan keterangan. Kalender menandai tanggal yang telah digunakan berdasarkan armada terpilih. Setelah validasi berhasil, sistem menampilkan ringkasan pesanan, estimasi biaya, dan pratinjau pesan WhatsApp.

Rahayu Trans Travel Beranda Armada Bandingkan Harga Galeri Cek Booking Cara Pesan Testimoni Kontak [Pesan Sekarang](#)

Beranda > Form Booking

Formulir Reservasi

Isi data di bawah. Pesanan akan otomatis diformat dan dikirim ke WhatsApp kami untuk konfirmasi cepat.

1 Isi Data 2 Konfirmasi

01. Pilih Kendaraan & Jadwal
Tentukan armada dan tanggal perjalanan Anda.

Kendaraan *

Tanggal Sewa (Mulai & Selesai) *

19 Jul 2026 - 22 Jul 2026

Durasi sewa: 4 hari

Zona Wilayah Tujuan *

Minimal hari sewa berbeda per zona wilayah
 ▲ Zona ini minimal sewa 3 hari

PRATINJAU PESANAN

Kendaraan	Premio Vlp Luxury Jsk Pijat (7 kursi)
Sopir	Dengan Sopir (wajib)
Mulai	19 Juli 2026
Selesai	22 Juli 2026
Durasi	4 hari
Zona Tujuan	Jawa Tengah
Pemesan	Miftah Fauzi
No. HP	+6282118703834
Estimasi Total	Rp 8.800.000
	Rp 2.200.000 x 4 hari

02. Data Pemesan
Informasi kontak untuk konfirmasi pemesanan.

Nama Lengkap *

Miftah Fauzi

Nomor HP / WhatsApp *

+6282118703834

Format: 08xx atau +62xx (untuk konfirmasi via WhatsApp)

Keterangan (opsional)

cth: penjemputan di Bandara Husein, wisata keluarga, acara pernikahan...

[Lanjut ke Ringkasan](#)

Gambar 5.5 Halaman Formulir Reservasi

5.2.1.5 Halaman Ringkasan Reservasi

Halaman ringkasan reservasi menampilkan kembali data pemesanan yang telah diisi pelanggan, meliputi armada terpilih, kapasitas, transmisi, zona tujuan, rentang tanggal, durasi sewa, identitas pemesan, serta estimasi total biaya. Sistem juga menampilkan pratinjau pesan WhatsApp dalam format baku agar pelanggan dapat memeriksa informasi sebelum dikirimkan. Apabila data belum sesuai, pelanggan dapat kembali mengubah pesanan. Setelah pelanggan mengonfirmasi pengiriman, sistem menyimpan *booking* dengan status *pending*, mencatat *expiresAt* selama 24 jam, membuat notifikasi untuk admin, dan mengarahkan pelanggan ke percakapan WhatsApp admin dengan draf pesan reservasi yang telah terisi.

Rahayu Trans Travel

Beranda Armada Bandingkan Harga Galeri Cek Booking Cara Pesan Testimoni Kontak [Pesan Sekarang](#)

Beranda > Form Booking

Formulir Reservasi

Isi data di bawah. Pesanan akan otomatis diformat dan dikirim ke WhatsApp kami untuk konfirmasi cepat.

1 Isi Data 2 Konfirmasi

RINGKASAN PESANAN
[Edit](#)

Kendaraan	Premio Vvip Luxury Jok Pijat
Kategori	Minibus
Kapasitas	7 orang
Transmisi	Manual
Sopir	Dengan Sopir (wajib)
Zona Tujuan	Jawa Barat
Tanggal Mulai	22 Juli 2026
Tanggal Selesai	29 Juli 2026
Durasi	8 hari
Pemesan	miifah
No. HP/WA	082118703834
Rp 2.300.000 / hari x 8 hari *Belum termasuk BBM, tol, dan parkir	Rp 16.800.000 (Estimasi Total)

PRATINJAU PESAN WHATSAPP

"PEMESANAN RAHAYU TRANS TRAVEL"

"Data Pemesan"

Nama: miifah
No. HP/WA: 082118703834

"Detail Kendaraan"

Armada: Premio Vvip Luxury Jok Pijat
Kapasitas: 7 orang
Sopir: Dengan sopir (wajib)

"Jadwal Perjalanan"

Tanggal mulai: 22 Juli 2026
Tanggal selesai: 29 Juli 2026
Durasi: 8 hari
Zona tujuan: Jawa Barat
Keterangan: jemput di bandara husein

"Estimasi Biaya"

Rp 2.300.000 x 8 hari = Rp 16.800.000
Belum termasuk BBM, tol, parkir, dan makan sopir

Mohon dikonfirmasi ketersediaan armada.
Terima Kasih.

[← Edit Pesanan](#) [Kirim via WhatsApp](#)

Dengan mengirim, WhatsApp akan terbuka dan pesan di atas akan terisi otomatis. Admin kami akan membalas dalam < 5 menit untuk konfirmasi.

Gambar 5.6 Halaman Ringkasan dan Pratinjau Pesan Reservasi

5.2.1.6 Halaman Cek Booking

Halaman cek *booking* digunakan untuk mencari reservasi berdasarkan nomor HP dan PIN. Sistem menyamarkan nama serta nomor HP sebelum hasil ditampilkan. Formulir testimoni tersedia ketika *booking* telah berstatus selesai dan belum memiliki testimoni.

Rahayu Trans Travel

Beranda Armada Bandingkan Harga Galeri Cek Booking Cara Pesan Testimoni Kontak [Pesan Sekarang](#)

Beranda > Cek Booking

Cek Status Pemesanan

Lacak status terbaru booking Anda secara langsung tanpa harus menghubungi admin.

Cari Reservasi

Masukkan nomor HP Anda saat memesan, beserta PIN verifikasi. PIN verifikasi dapat berupa 4 digit terakhir nomor HP Anda atau tanggal-bulan-mulai sewa (DDMMYY).

Nomor HP Pemesan: 082118703834 PIN Verifikasi Booking: 2207

[Cari Sekarang](#)

Hasil Pencarian

BOOKING ID #18 [Selesai](#)

Premio Vvip Luxury Jok Pijat

Pemesan: m****h No. HP: 0821****3834 Layanan: Dengan Sopir (wajib)	Tanggal Sewa: 22 Jul - 30 Jul 2026 (9 hari) Tujuan/Catatan: ****g*****	TIMELINE STATUS Permintaan Diterima 24 Jul 2025 Dikonfirmasi & Diterima Armada telah di-blok & siap digunakan Perjalanan Selesai Mobi telah dikembalikan dengan aman
---	--	---

[Tulis Ulasan Perjalanan](#)

Bagikan pengalaman perjalanan Anda untuk membantu pelanggan lain.

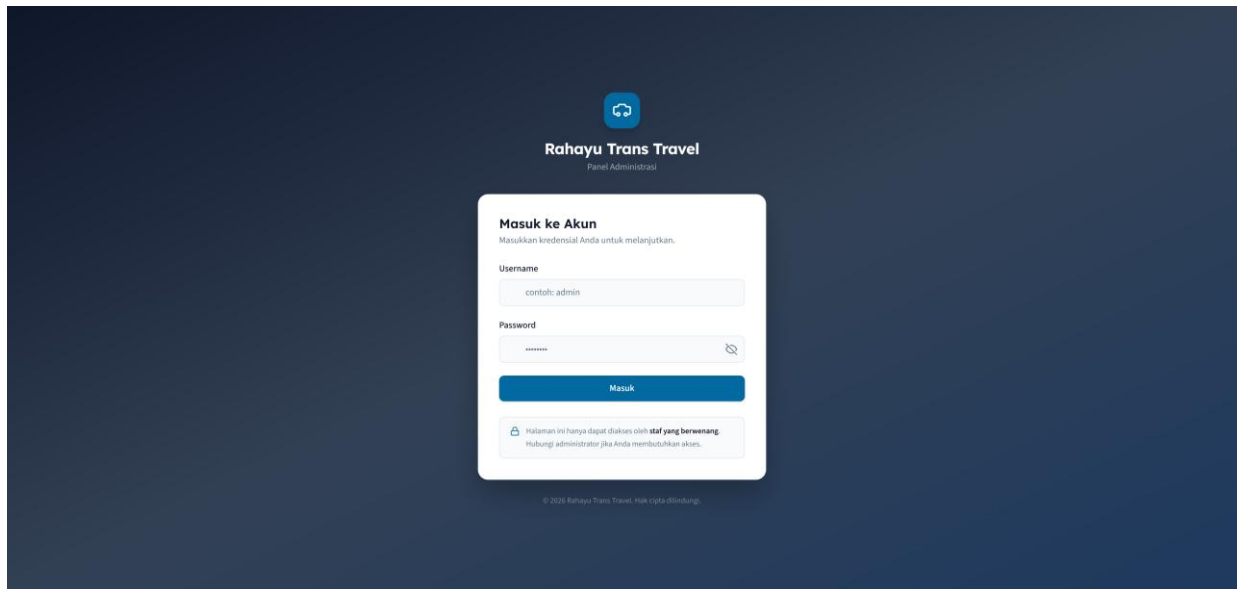
Gambar 5.7 Halaman Cek *Booking*

5.2.2 Antarmuka Admin

Antarmuka admin digunakan untuk mengelola data operasional *website*. Admin harus melakukan *login* terlebih dahulu sebelum dapat mengakses *dashboard*. Setelah berhasil *login*, admin dapat melihat ringkasan statistik, mengelola data armada, mengelola zona wilayah, mengatur harga berdasarkan zona, memantau *booking*, memperbarui status *booking* dan pembayaran, memoderasi testimoni, mengelola pengaturan situs, serta melihat notifikasi *booking* baru.

5.2.2.1 Halaman Login Admin

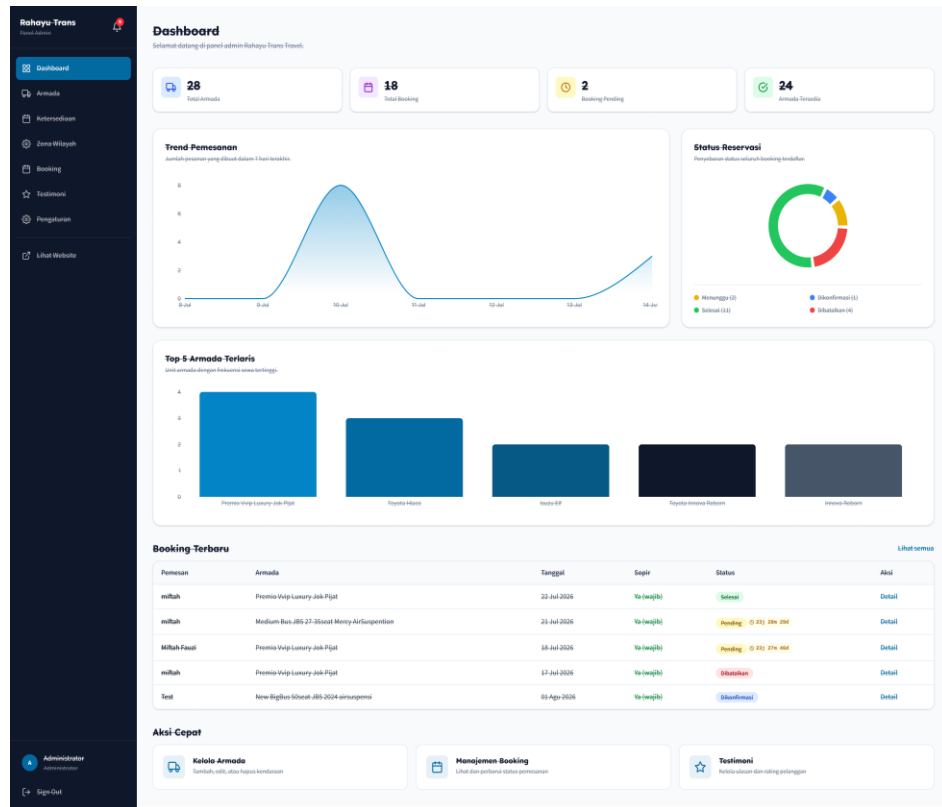
Halaman *login* digunakan untuk membatasi akses ke fitur pengelolaan sistem. *Admin* memasukkan *username* dan *password*, kemudian sistem memvalidasi kredensial melalui *Auth.js/NextAuth*. Jika data valid, admin diarahkan ke halaman *dashboard*. jika tidak, sistem menampilkan pesan kesalahan.



Gambar 5.8 Halaman *Login* Admin

5.2.2.2 Halaman Dashboard Admin

Halaman *dashboard* admin menampilkan ringkasan data operasional sistem, seperti jumlah armada, *booking*, dan testimoni. Halaman ini juga menyediakan akses menuju pengelolaan armada, zona wilayah, harga zona, *booking*, ketersediaan jadwal, testimoni, notifikasi, dan pengaturan situs.



Gambar 5.9 Dashboard Admin

5.2.2.3 Halaman Manajemen Armada

Halaman manajemen armada menyediakan fungsi tambah, ubah, unggah foto, nonaktifkan, hapus, ekspor Excel, dan pengaturan harga zona. Halaman zona wilayah digunakan untuk mengatur nama zona, minimal hari, dan urutan tampilan.

Armada

28 kendaraan terdaftar

[Export Excel](#) [+ Tambah Armada](#)

Nama	Kategori	Kapasitas	Harga/Hari	Status	Aksi
BIGBUS SHD 50 SEAT JETBUS3+ 2022 - Manual	Minibus	50 orang	Rp 3.600.000	☒ Tersedia	Harga Zona Edit Hapus
BIGBUS VIP 34 Seat LEGREST 2020 - Manual	Minibus	34 orang	Rp 3.600.000	☒ Tersedia	Harga Zona Edit Hapus
Rubicon 2023 - Automatic	SUV	5 orang	Rp 5.500.000	☒ Tersedia	Harga Zona Edit Hapus
Defender 2023 - Automatic	SUV	5 orang	Rp 6.500.000	☒ Tersedia	Harga Zona Edit Hapus
Lexus LX570 2021 - Automatic	SUV	7 orang	Rp 12.000.000	☒ Tersedia	Harga Zona Edit Hapus
Avanza 2023 2023 - Automatic	MPV	7 orang	Rp 650.000	☒ Tersedia	Harga Zona Edit Hapus
Xpander 2023 2023 - Automatic	MPV	7 orang	Rp 700.000	☒ Tersedia	Harga Zona Edit Hapus
Innova Zenix 2020 - Automatic	MPV	7 orang	Rp 1.000.000	☒ Tersedia	Harga Zona Edit Hapus
New BigBus Soleat JBS 2024 airsuspensi	Minibus	50 orang	Rp 3.800.000	☒ Tersedia	Harga Zona Edit Hapus
Innova Zenix Q Modelista 2023 - Automatic	MPV	7 orang	Rp 1.300.000	☒ Tersedia	Harga Zona Edit Hapus

Menampilkan 11 - 28 dari 28 armada

Sebelumnya 1 2 3 Berikutnya

Gambar 5.10 Halaman Manajemen Armada

5.2.2.4 Halaman Pengaturan Harga Zona

Halaman pengaturan harga zona digunakan admin untuk menetapkan tarif sewa per hari berdasarkan kombinasi armada dan zona tujuan. Admin memilih armada, kemudian memasukkan harga untuk setiap zona wilayah yang tersedia. Sistem menyimpan tarif tersebut pada entitas HargaZona menggunakan kombinasi armadaId dan zonaId. Tarif ini digunakan untuk menghitung estimasi biaya reservasi. Apabila tarif zona untuk armada yang dipilih belum tersedia, sistem menggunakan harga dasar armada sebagai tarif cadangan.

The screenshot shows the 'Armada' management interface. A modal for setting prices for the 'BIGBUS VIP 34 Seat LEGREST' is active. The modal lists the following regions and their minimum rental durations:

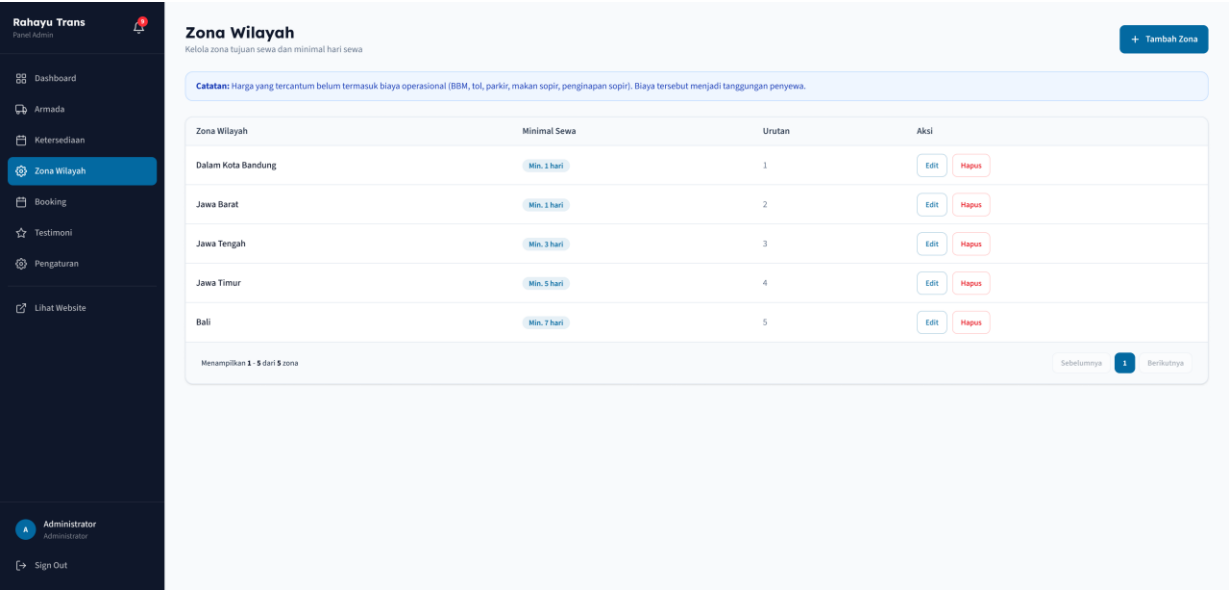
- Dalam Kota Bandung (min. 1 hari)
- Jawa Barat (min. 1 hari)
- Jawa Tengah (min. 3 hari)
- Jawa Timur (min. 5 hari)
- Bali (min. 7 hari)

The modal also includes a 'Batal' (Cancel) button and a 'Simpan Harga' (Save Price) button.

Gambar 5.11 Halaman Pengaturan Harga Zona

5.2.2.5 Halaman Manajemen Zona Wilayah

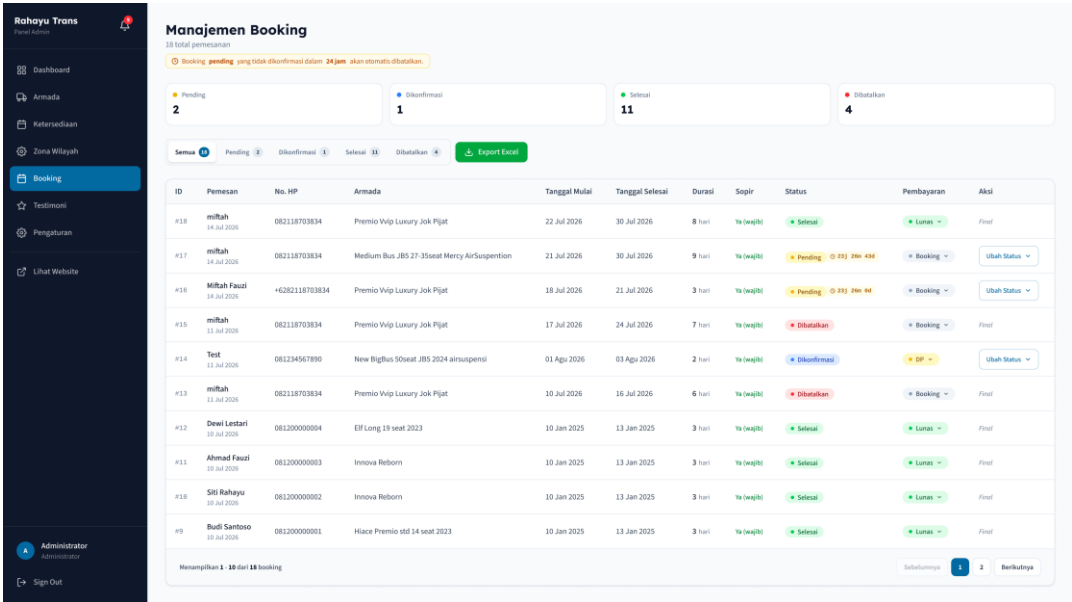
Halaman manajemen zona wilayah digunakan admin untuk mengelola daftar wilayah tujuan penyewaan. Admin dapat menambahkan, memperbarui, dan menghapus zona, serta menentukan minimal hari sewa dan urutan tampil setiap zona. Data minimal hari digunakan sistem untuk memvalidasi durasi reservasi, sedangkan urutan tampil menentukan susunan pilihan zona pada formulir reservasi pelanggan.



Gambar 5.12 Halaman Manajemen Zona Wilayah

5.2.2.6 Halaman Manajemen Booking

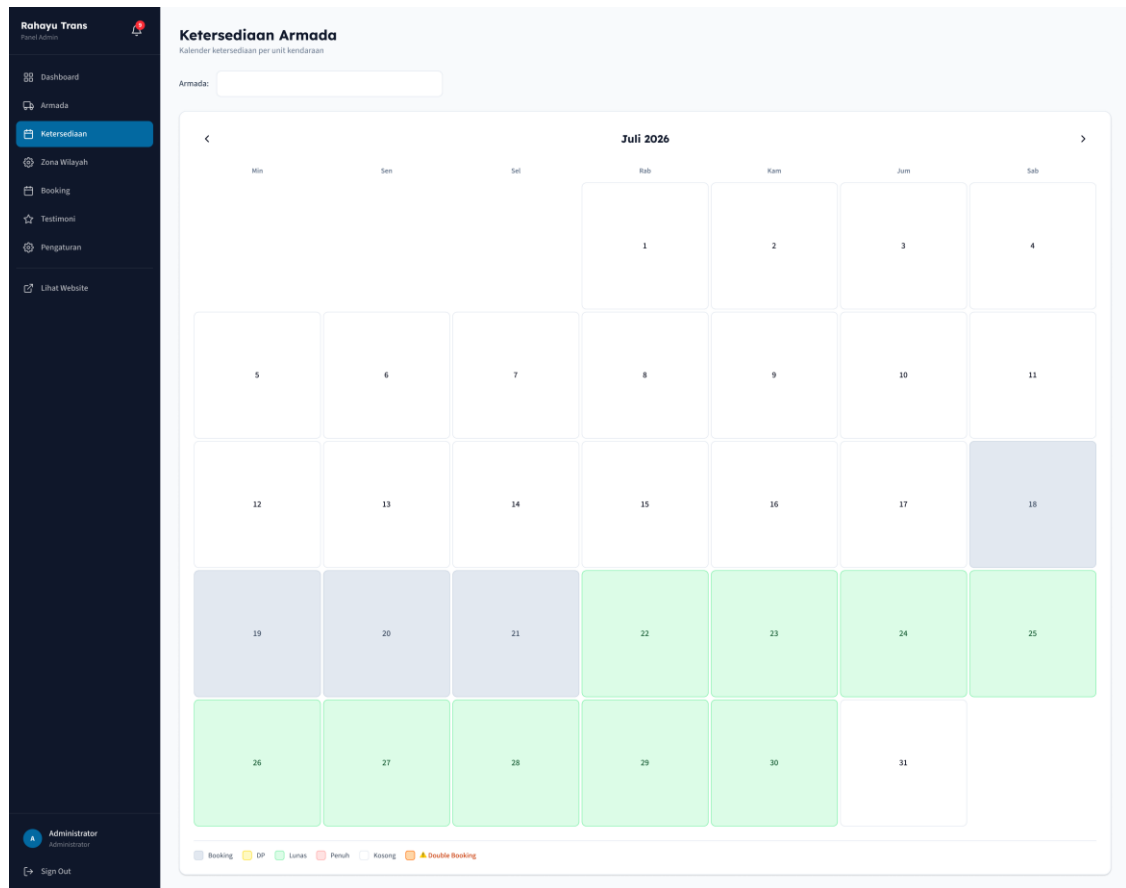
Halaman manajemen *booking* menampilkan ringkasan status, filter, *countdown* expiresAt untuk *booking pending*, status pembayaran, perubahan status operasional, dan ekspor data. Data expired diperiksa sebelum daftar terbaru ditampilkan. Halaman testimoni digunakan untuk menyetujui, membatalkan persetujuan, atau menghapus ulasan. Pengaturan situs menyimpan informasi dinamis seperti kontak WhatsApp admin.



Gambar 5.13 Halaman Manajemen Booking

5.2.2.7 Halaman *Monitoring Ketersediaan Armada*

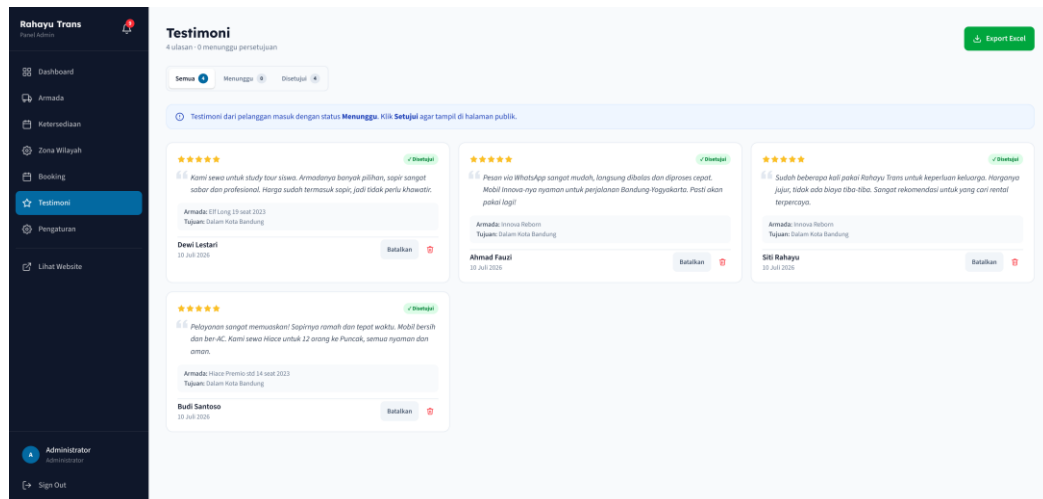
Halaman *monitoring* ketersediaan armada menampilkan kalender penggunaan untuk setiap unit kendaraan. Admin memilih armada dan bulan yang ingin diperiksa, kemudian sistem mengambil *booking* yang belum dibatalkan melalui *endpoint* ketersediaan. Setiap tanggal diberi warna berdasarkan status pembayaran, yaitu *booking*, DP, lunas, atau penuh. Apabila ditemukan lebih dari satu *booking* pada armada dan tanggal yang sama, sistem menampilkan penanda bentrok sebagai peringatan bagi admin.



Gambar 5.14 Halaman *Monitoring Ketersediaan Armada*

5.2.2.8 Halaman Manajemen Testimoni

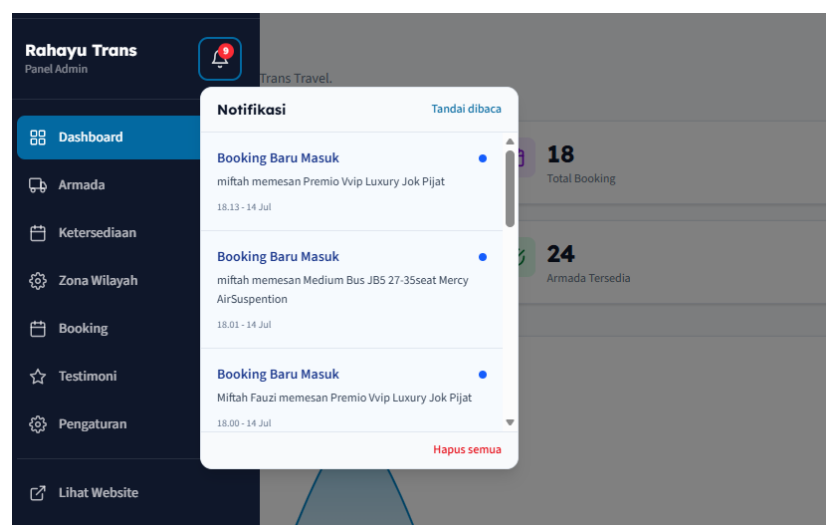
Halaman manajemen testimoni digunakan untuk memoderasi ulasan yang dikirim pelanggan setelah *booking* selesai. Admin dapat memfilter testimoni berdasarkan status moderasi, menyetujui testimoni agar tampil pada halaman publik, membatalkan persetujuan, atau menghapus testimoni. Informasi armada dan pelanggan ditampilkan untuk membantu admin memeriksa keterkaitan ulasan dengan transaksi.



Gambar 5.15 Halaman Manajemen Testimoni

5.2.2.9 Halaman Panel notifikasi

Panel notifikasi menampilkan pemberitahuan *booking* baru yang dibuat oleh sistem setelah reservasi berhasil disimpan. Admin dapat melihat jumlah notifikasi yang belum dibaca, membuka informasi *booking* terkait, menandai notifikasi sebagai telah dibaca, dan menghapus notifikasi. Fitur ini membantu admin mengetahui adanya reservasi baru tanpa harus memeriksa tabel *booking* secara terus-menerus.



Gambar 5.16 Panel Notifikasi Admin

5.2.2.10 Halaman pengaturan situs

Halaman pengaturan situs digunakan untuk menyimpan informasi bisnis yang ditampilkan secara dinamis pada *website*. Data yang dapat diperbarui meliputi nomor dan tampilan WhatsApp, alamat, lokasi Google Maps, jam operasional, nama bisnis, *tagline*, dan email. Sistem menyimpan setiap pengaturan menggunakan metode *upsert* pada entitas *SiteSetting* sehingga nilai lama diperbarui apabila key telah tersedia.

Pengaturan Situs
Kebila informasi kontak dan identitas bisnis Anda.

Kontak

Nomor WhatsApp (format internasional)
6282118703834
Digunakan untuk link tombol WhatsApp. Contoh: 6282204567890

Tampilan Nomor WhatsApp
+62 821-1870 3834
Nomor yang ditampilkan kepada pengunjung website.

Alamat
Jl. Contoh No. 123, Bandung, Jawa Barat

Google Map Embed URL (url)
<https://www.google.com/maps/embed?pb=...>
Paste URL yang ada di dalam atribut <iframe>... pada kode frame Google Maps Anda.

Jam Operasional
Setiap hari, 07.00 - 21.00 WIB

Informasi Bisnis

Nama Bisnis *
Rahayu Trans Travel

Tagline
Perjalanan Nyaman, Harga Terjangkau
Slogan singkat yang muncul di bawah nama bisnis.

Email
info@rahayutrans.com

Simpan Pengaturan

Gambar 5.17 Halaman Pengaturan Situs

5.3 Implementasi Fitur Utama

Implementasi fitur utama menjelaskan fungsi inti sistem yang berhubungan langsung dengan kebutuhan pengguna dan permasalahan pada sistem berjalan.

5.3.1 Implementasi Katalog dan Detail Armada

Fitur katalog armada digunakan untuk menampilkan daftar kendaraan yang tersedia pada Rahayu Trans Travel. Data armada diambil dari basis data dan ditampilkan dalam bentuk kartu yang memuat foto, nama kendaraan, kategori, kapasitas, transmisi, tahun, dan harga sewa. Pelanggan dapat membuka halaman detail armada untuk melihat informasi lebih lengkap sebelum melakukan reservasi.

Kode Program 5.1 Pencarian Armada Tersedia pada Basis Data

```
const whereBase = {
  tersedia: true,
  kapasitas: { gte: jumlahPenumpang },
};

const rows = await prisma.armada.findMany({
  where:
    tanggalStart && tanggalEnd
    ? {
      ...whereBase,
      bookings: {
        none: {
          status: { not: "dibatalkan" },
          tanggalMulai: { lte: tanggalEnd },
          tanggalSelesai: { gte: tanggalStart },
        },
      },
    }
    : whereBase,
  orderBy: { id: "asc" },
});
```

Kode Program 5.2 Filter dan Pengurutan Katalog pada *Client*

```
const filtered = useMemo(() => {
  const base = armadaList
    .filter((v) => category === "Semua" || v.category === category)
    .filter((v) => matchesCapacity(v.seats, capacity))
    .filter((v) => matchesPrice(v.pricePerDay, price));

  return applySort(base, sort);
}, [category, capacity, price, sort]);
```

5.3.2 Implementasi Formulir Reservasi dan Ringkasan Reservasi

Formulir menggunakan React Hook Form dan Zod untuk memvalidasi data. Durasi dihitung secara inklusif, sehingga tanggal mulai dan tanggal selesai sama dihitung sebagai satu hari. Ketika armada berasal dari halaman katalog atau detail, parameter armada digunakan sebagai nilai awal. Setelah validasi *form*, sistem memeriksa bentrok tanggal dan minimal hari zona sebelum berpindah ke ringkasan.

Kode Program 5.3 Perhitungan Durasi Inklusif

```
function hitungDurasi(from: Date | undefined, to: Date | undefined): number {
  if (!from || !to) return 0;
  const diff = to.getTime() - from.getTime();
  return Math.max(1, Math.ceil(diff / (1000 * 60 * 60 * 24)) + 1);
}
```

5.3.3 Implementasi Validasi Ketersediaan Armada

Validasi ketersediaan armada dilakukan dengan mengambil data *booking* berdasarkan armada dan rentang tanggal tertentu. Sistem mengecek *booking* yang statusnya belum dibatalkan agar pelanggan tidak memilih jadwal yang sudah digunakan. Dengan fitur ini, risiko *double booking* dapat dikurangi karena sistem menampilkan data ketersediaan berdasarkan transaksi yang tersimpan di basis data.

Kode Program 5.4 Pemeriksaan Bentrok *Booking* pada *Server*

```
const conflictingBooking = await prisma.booking.findFirst({
  where: {
    armadaId: parsedArmadaId,
    status: { not: "dibatalkan" },
    tanggalMulai: { lte: endDate },
    tanggalSelesai: { gte: startDate },
  },
});

if (conflictingBooking) {
  return badRequest("Tanggal reservasi bertabrakan dengan booking aktif");
}
```

5.3.4 Implementasi Perhitungan Harga Berdasarkan Zona

Perhitungan harga dilakukan berdasarkan armada yang dipilih, zona wilayah tujuan, dan durasi sewa. Setiap zona wilayah memiliki aturan minimal hari sewa. Harga armada juga dapat berbeda untuk setiap zona melalui tabel *HargaZona*. Sistem menghitung estimasi biaya agar pelanggan dapat mengetahui perkiraan total sewa sebelum mengirim pesan.

Kode Program 5.5 Pemilihan Tarif dan Harga Dasar Cadangan

```
const hargaPerHari = useMemo(() => {
  if (!selectedVehicle) return 0;
  if (!selectedZona) return selectedVehicle.pricePerDay;
  if (hargaZonaData?.armadaId !== selectedVehicle.id) {
    return selectedVehicle.pricePerDay;
  }
  return hargaZonaData.prices[selectedZona.id]
    ?? selectedVehicle.pricePerDay;
}, [selectedVehicle, selectedZona, hargaZonaData]);

const totalEstimasi = hargaPerHari * durasi;
```

5.3.5 Implementasi Integrasi WhatsApp *Click to Chat*

Integrasi WhatsApp *Click to Chat* digunakan untuk meneruskan data reservasi pelanggan kepada admin. Setelah pelanggan mengisi formulir dan menyimpan *booking*, sistem membentuk format pesan otomatis berisi nama pelanggan, nomor HP, armada, tanggal sewa, zona tujuan, dan keterangan. Pesan tersebut kemudian dikirim melalui tautan WhatsApp sehingga admin menerima data pesanan dengan format yang lebih konsisten.

Kode Program 5.6 Penyimpanan *Booking* sebelum Membuka WhatsApp

```
const response = await fetch("/api/booking", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify(dataReservasi),
});

if (!response.ok) return;

const message = buildWaMessage(dataReservasi);
window.location.assign(buildWaUrl(message, waNumber));

export function buildWaUrl(text: string, phone: string): string {
```

```
const digits = phone.replace(/\D/g, "");
const normalizedPhone = digits.startsWith("0")
  ? `62${digits.slice(1)}`
  : digits;

return `https://api.whatsapp.com/send?phone=${encodeURIComponent(
  normalizedPhone
)}&text=${encodeURIComponent(text)}`;
}
```

5.3.6 Implementasi Pembatalan Otomatis *Booking Pending*

Booking baru disimpan dengan status *pending* dan *expiresAt* yang bernilai 24 jam setelah waktu pembuatan. Selama status belum dibatalkan, rentang tanggal tetap mengunci armada. Fungsi *expireStalePendingBookings* hanya mengubah *booking* yang masih *pending* dan *expiresAt* telah terlewat menjadi dibatalkan. *Booking* yang telah dikonfirmasi tidak memenuhi kondisi tersebut sehingga tetap mengunci jadwal.

Kode Program 5.7 Penyimpanan *Booking Pending* dan *expiresAt*

```
const now = new Date();
const booking = await prisma.booking.create({
  data: {
    namaPelanggan,
    noHp,
    armadaId: parsedArmadaId,
    tanggalMulai: startDate,
    tanggalSelesai: endDate,
    tujuanZona: tujuanZona ?? null,
    status: "pending",
    expiresAt: getExpiresAt(now),
  },
});
```

Kode Program 5.8 Proses Pembatalan *Booking* Kedaluwarsa

```
export async function expireStalePendingBookings(now = new Date()) {
  return prisma.booking.updateMany({
    where: {
      status: "pending",
      expiresAt: { lte: now },
    },
    data: { status: "dibatalkan" },
  });
}
```

Pemeriksaan dijalankan ketika daftar *booking* admin, pembuatan *booking*, ketersediaan armada, atau halaman cek *booking* mengakses data. Sebagai pemeriksaan cadangan, Vercel *Cron* memanggil *endpoint* `/api/cron/expire-bookings` satu kali setiap hari berdasarkan konfigurasi `vercel.json`.

Kode Program 5.9 Konfigurasi Vercel *Cron* Harian

```
{
  "crons": [
    {
      "path": "/api/cron/expire-bookings",
      "schedule": "0 0 * * *"
    }
  ]
}
```

5.3.7 Implementasi Cek Status Pesanan

Fitur cek status pesanan memungkinkan pelanggan melihat status *booking* secara mandiri. Pelanggan memasukkan nomor HP dan PIN verifikasi. Sistem mencari data *booking* berdasarkan nomor HP, lalu mencocokkan PIN dengan beberapa kemungkinan seperti empat digit terakhir nomor HP, tanggal mulai sewa, atau ID *booking*. Data sensitif seperti nama dan nomor HP ditampilkan dalam bentuk tersamarkan untuk menjaga privasi pelanggan.

Kode Program 5.10 Pencarian *Booking* dengan Nomor HP dan PIN

```
await expireStalePendingBookings();

const bookings = await prisma.booking.findMany({
  where: {
    OR: [
      { noHp: { contains: cleanHp } },
      { noHp: { contains: noHp } },
    ],
  },
  include: {
    armada: { select: { nama: true, kategori: true, transmisi: true } },
    testimoni: { select: { id: true } },
  },
  orderBy: { createdAt: "desc" },
});

const filteredBookings = bookings.filter((b) => {
  const hpPin = cleanHp.slice(-4);
  const start = new Date(b.tanggalMulai);
  const datePin = `${String(start.getDate()).padStart(2, "0")}${
    String(start.getMonth() + 1).padStart(2, "0")
  }`;
  const datePinAlt = `${String(start.getMonth() + 1).padStart(2, "0")}${
    String(start.getDate()).padStart(2, "0")
  }`;
  const idPin = String(b.id);
  const targetPin = String(pin).trim();

  return targetPin === hpPin || targetPin === datePin ||
    targetPin === datePinAlt || targetPin === idPin ||
    idPin.endsWith(targetPin);
});

const maskedResult = filteredBookings.map((b) => ({
  id: b.id,
  namaPelanggan: maskName(b.namaPelanggan),
  noHp: maskPhone(b.noHp),
  armada: b.armada,
  tanggalMulai: b.tanggalMulai.toISOString(),
  tanggalSelesai: b.tanggalSelesai.toISOString(),
  tujuanZona: b.tujuanZona,
  status: b.status,
  hasReview: !!b.testimoni,
}));
```

5.3.8 Implementasi Manajemen *Booking* dan Ekspor Excel

Manajemen *booking* digunakan admin untuk melihat daftar pesanan yang masuk, memfilter berdasarkan status, memperbarui status *booking*, dan memperbarui status pembayaran. Data *booking* juga dapat diekspor ke format Excel menggunakan *library* SheetJS/xlsx agar admin dapat membuat rekapitulasi transaksi secara lebih mudah.

Kode Program 5.11 Pembaruan Status *Booking* dan Pembayaran

```
const body: {
  status?: BookingStatus;
  statusPembayaran?: PaymentStatus;
} = await req.json();

const data: Record<string, string> = {};

if (body.status) {
  if (!(BOOKING_STATUSES as readonly string[]).includes(body.status)) {
    return badRequest("Status tidak valid");
  }
  data.status = body.status;
}

if (body.statusPembayaran) {
  if (!(PAYMENT_STATUSES as readonly
string[]).includes(body.statusPembayaran)) {
    return badRequest("Status pembayaran tidak valid");
  }
  data.statusPembayaran = body.statusPembayaran;
}

const updated = await prisma.booking.update({ where: { id }, data });
```

Kode Program 5.12 Ekspor Data *Booking* ke Excel

```
export function exportToExcel({
  data,
  filename,
  sheetName = "Data",
}: ExportOptions) {
  const worksheet = XLSX.utils.json_to_sheet(data);
  const workbook = XLSX.utils.book_new();
  XLSX.utils.book_append_sheet(workbook, worksheet, sheetName);
  XLSX.writeFile(workbook, `${filename}.xlsx`);
}

const today = new Date().toISOString().split("T")[0];
exportToExcel({
  data: exportData,
  filename: `booking_${today}`,
  sheetName: "Data Booking",
});
```

5.3.9 Implementasi *Monitoring Ketersediaan Armada*

Monitoring ketersediaan armada digunakan admin untuk melihat pemakaian setiap kendaraan dalam bentuk kalender bulanan. Setelah admin memilih armada dan periode bulan, antarmuka mengirim permintaan ke *endpoint* availability dengan ID armada dan parameter bulan. *Endpoint* menjalankan pemeriksaan *booking pending* yang telah kedaluwarsa, kemudian mengambil *booking* yang belum berstatus dibatalkan pada rentang bulan tersebut. Data *booking* ditampilkan menggunakan warna

berdasarkan status pembayaran dan penanda khusus apabila ditemukan jadwal yang bertabrakan. Fitur ini merupakan bagian pendukung dari pengelolaan pesanan karena jadwal terbentuk dari data *booking* dan tidak diubah secara manual melalui kalender.

Kode Program 5.13 Pengambilan Data Ketersediaan Bulanan

```
const monthStr = `${year}-${String(month + 1).padStart(2, "0")}`;

fetch(`/api/armada/${selectedArmada}/availability?month=${monthStr}`)
  .then((response) => response.json())
  .then((data) => setBookings(data.bookings || []))
  .finally(() => setLoading(false));
```

5.3.10 Implementasi Testimoni, Notifikasi, dan Pengaturan

Testimoni hanya dapat dibuat untuk *booking* milik nomor HP yang sesuai dan telah berstatus selesai. Satu *booking* dibatasi memiliki satu testimoni. Testimoni baru berstatus belum disetujui hingga admin melakukan moderasi. Notification dibuat ketika *booking* baru berhasil disimpan dan dapat ditandai dibaca atau dihapus oleh admin. SiteSetting menggunakan metode *upsert* untuk menyimpan konfigurasi dinamis, termasuk nomor WhatsApp dan informasi bisnis.

Kode Program 5.14 Validasi dan Penyimpanan Testimoni

```
const booking = await prisma.booking.findFirst({
  where: {
    id: Number(bookingId),
    noHp: String(noHp),
    status: "selesai",
  },
});

if (!booking) return badRequest("Booking tidak valid atau belum selesai");

const existing = await prisma.testimoni.findUnique({
  where: { bookingId: Number(bookingId) },
});

if (existing) return badRequest("Booking ini sudah memiliki testimoni");

const testimoni = await prisma.testimoni.create({
  data: {
    bookingId: Number(bookingId),
    isi,
    rating: Number(rating),
    approved: false,
  },
});
```

Kode Program 5.15 Penyimpanan Notifikasi dan Pengaturan Situs

```

await prisma.notification.create({
  data: {
    title: "Booking Baru Masuk",
    message: `${namaPelanggan} memesan ${armada?.nama || "Kendaraan"}`,
    type: "booking",
    bookingId: booking.id,
  },
});

await Promise.all(
  Object.entries(body).map(([key, value]) =>
    prisma.siteSetting.upsert({
      where: { key },
      update: { value },
      create: { key, value },
    })
  ),
);

```

5.4 Pengujian Fungsional *Black Box*

Pengujian sistem dilakukan menggunakan metode *Black Box Testing*. Pengujian ini berfokus pada masukan dan keluaran sistem tanpa melihat struktur kode program secara langsung. Tujuan pengujian adalah memastikan setiap fitur utama berjalan sesuai kebutuhan fungsional yang telah dirancang pada Bab 4.

5.4.1 Pengujian Katalog dan Reservasi

Pengujian katalog dan reservasi dilakukan untuk memastikan pelanggan dapat melihat daftar armada serta menemukan kendaraan sesuai kebutuhan perjalanan. Pengujian mencakup pemuatan katalog, pencarian berdasarkan jadwal dan kapasitas, penyaringan kategori dan harga, serta pengurutan data armada. Skenario pengujian katalog dan reservasi disajikan pada Tabel 5.7

Tabel 5.7 Pengujian Katalog dan Reservasi

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-01-01	Membuka katalog armada	Pelanggan membuka halaman /armada tanpa parameter pencarian.	Sistem menampilkan seluruh armada yang berstatus tersedia.	Pass
TC-01-02	Pencarian berdasarkan jadwal dan kapasitas	Pelanggan mengisi tanggal mulai, tanggal selesai, zona tujuan, dan jumlah penumpang.	Sistem menampilkan armada yang kapasitasnya mencukupi dan tidak memiliki <i>booking</i> bentrok.	Pass
TC-01-03	Filter kategori	Pelanggan memilih salah satu kategori armada.	Daftar hanya menampilkan armada dengan kategori yang dipilih.	Pass
TC-01-04	Filter kapasitas dan harga	Pelanggan memilih kapasitas dan rentang harga tertentu.	Daftar armada berubah sesuai kapasitas dan rentang harga yang dipilih.	Pass
TC-01-05	Pengurutan katalog	Pelanggan memilih urutan relevansi, harga termurah, harga tertinggi, atau tahun terbaru.	Sistem mengurutkan daftar armada sesuai pilihan pelanggan.	Pass

5.4.2 Pengujian Formulir dan Ringkasan Reservasi

Pengujian formulir dan ringkasan reservasi dilakukan untuk memastikan data pemesanan dapat dimasukkan dan divalidasi dengan benar. Pengujian mencakup penggunaan armada terpilih sebagai nilai awal, validasi data wajib, perhitungan durasi secara inklusif, ketentuan minimal hari sewa, serta penyajian ringkasan dan estimasi biaya. Skenario pengujiannya disajikan pada Tabel 5.8

Tabel 5.8 Pengujian Formulir dan Ringkasan Reservasi

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-02-01	Armada sebagai nilai awal	Pelanggan membuka formulir dari katalog atau detail armada.	Armada yang sebelumnya dipilih tampil sebagai nilai awal formulir.	Pass
TC-02-02	Validasi field wajib	Pelanggan mengirim formulir dengan nama, nomor HP, tanggal, atau zona yang kosong.	Sistem menolak proses dan menampilkan pesan validasi pada field terkait.	Pass
TC-02-03	Durasi satu hari	Pelanggan memilih tanggal mulai dan selesai pada tanggal yang sama.	Sistem menghitung durasi sewa sebagai satu hari.	Pass
TC-02-04	Durasi kurang dari minimal zona	Pelanggan memilih durasi yang lebih pendek dari minimal hari zona tujuan.	Sistem menolak proses dan menampilkan ketentuan minimal hari sewa.	Pass
TC-02-05	Menampilkan ringkasan reservasi	Pelanggan mengisi seluruh data dengan valid dan melanjutkan proses.	Sistem menampilkan armada, tanggal, zona, durasi, data pelanggan, dan estimasi biaya.	Pass

5.4.3 Pengujian Validasi Ketersediaan Armada

Pengujian validasi ketersediaan armada dilakukan untuk memastikan sistem menampilkan jadwal berdasarkan armada yang dipilih dan mencegah terjadinya pemesanan pada tanggal yang bertabrakan. Pengujian juga memastikan *booking* yang dibatalkan tidak lagi mengunci jadwal dan tanggal yang sama tetap dapat digunakan oleh armada berbeda. Skenario pengujiannya disajikan pada Tabel 5.9

Tabel 5.9 Pengujian Validasi Ketersediaan Armada

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-03-01	Mengambil ketersediaan per armada	Pelanggan membuka kalender pada formulir untuk armada tertentu.	Sistem mengambil <i>bookedDates</i> berdasarkan ID armada yang sedang dipilih.	Pass
TC-03-02	Tanggal telah dipesan	Pelanggan membuka kalender yang memiliki <i>booking</i> belum dibatalkan.	Tanggal yang telah dipesan ditandai dan tidak dapat dipilih.	Pass
TC-03-03	Pencegahan <i>double booking</i>	Pelanggan memanipulasi permintaan dan mengirim rentang tanggal yang bertabrakan.	Server menolak penyimpanan <i>booking</i> dan mengembalikan pesan jadwal bertabrakan.	Pass
TC-03-04	Armada berbeda pada tanggal sama	Pelanggan memilih tanggal yang digunakan oleh armada lain.	Tanggal tetap dapat dipilih apabila armada yang dipilih tidak memiliki <i>booking</i> bentrok.	Pass
TC-03-05	Tanggal <i>booking</i> dibatalkan	Pelanggan membuka kalender pada <i>booking</i> berstatus dibatalkan.	Tanggal <i>booking</i> yang dibatalkan tersedia kembali untuk dipilih.	Pass

5.4.4 Pengujian Perhitungan Harga Berdasarkan Zona

Pengujian perhitungan harga berdasarkan zona dilakukan untuk memastikan sistem mengambil tarif dari HargaZona sesuai kombinasi armada dan zona tujuan. Pengujian mencakup penggunaan harga dasar armada sebagai tarif cadangan, perhitungan total berdasarkan durasi inklusif, serta perubahan tarif dan minimal hari ketika zona tujuan diganti. Skenario pengujiannya disajikan pada Tabel 5.10

Tabel 5.10 Pengujian Perhitungan Harga Berdasarkan Zona

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-04-01	Tarif HargaZona tersedia	Pelanggan memilih kombinasi armada dan zona yang memiliki HargaZona.	Sistem menggunakan harga per hari dari HargaZona.	Pass
TC-04-02	HargaZona tidak tersedia	Pelanggan memilih kombinasi armada dan zona tanpa data HargaZona.	Sistem menggunakan harga dasar armada sebagai tarif cadangan.	Pass
TC-04-03	Perhitungan total estimasi	Pelanggan memilih tarif dan rentang sewa yang valid.	Total estimasi sama dengan tarif per hari dikalikan durasi inklusif.	Pass
TC-04-04	Perubahan zona tujuan	Pelanggan mengganti zona setelah memilih armada dan tanggal.	Minimal hari, tarif per hari, dan total estimasi diperbarui sesuai zona baru.	Pass

5.4.5 Pengujian Integrasi WhatsApp *Click to Chat*

Pengujian integrasi WhatsApp *Click to Chat* dilakukan untuk memastikan sistem membentuk pesan reservasi dalam format baku dan menyimpan booking sebelum membuka percakapan WhatsApp admin. Pengujian juga mencakup penanganan kegagalan penyimpanan serta pembukaan <https://api.whatsapp.com/send> dengan draf pesan yang telah terisi. Skenario pengujiannya disajikan pada Tabel 5.11

Tabel 5.11 Pengujian Integrasi WhatsApp *Click to Chat*

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-05-01	Pratinjau pesan WhatsApp	Pelanggan menyelesaikan formulir dan membuka ringkasan.	Sistem membentuk pesan baku yang memuat data reservasi pelanggan.	Pass
TC-05-02	Menyimpan booking sebelum WhatsApp	Pelanggan mengonfirmasi pengiriman pesan melalui WhatsApp.	Sistem menyimpan <i>booking pending</i> dan membuat notifikasi sebelum membuka WhatsApp.	Pass
TC-05-03	Penyimpanan booking gagal	API <i>booking</i> mengembalikan respons gagal ketika pelanggan mengonfirmasi pengiriman.	Sistem menampilkan kesalahan dan tidak membuka percakapan WhatsApp.	Pass
TC-05-04	Membuka percakapan admin	Booking berhasil disimpan dan sistem membentuk tautan https://api.whatsapp.com/send	Percakapan WhatsApp admin terbuka dengan draf pesan terisi; pelanggan mengirimkannya secara manual.	Pass

5.4.6 Pengujian Pembatalan Otomatis *Booking Pending*

Pengujian pembatalan otomatis dilakukan untuk memastikan *booking* baru memperoleh status *pending* dan *expiresAt* selama 24 jam. Pengujian mencakup kondisi sebelum dan setelah batas waktu, perlakuan terhadap *booking* yang telah dikonfirmasi, serta pemeriksaan melalui Vercel *Cron* harian. Skenario pengujiannya disajikan pada Tabel 5.12

Tabel 5.12 Pengujian Pembatalan Otomatis *Booking Pending*

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-06-01	Pencatatan <i>expiresAt</i>	Pelanggan berhasil membuat <i>booking</i> baru.	Sistem menyimpan status <i>pending</i> dan <i>expiresAt</i> 24 jam setelah waktu pembuatan.	Pass
TC-06-02	<i>Pending</i> belum kedaluwarsa	Sistem memeriksa <i>booking pending</i> sebelum <i>expiresAt</i> .	<i>Booking</i> tetap <i>pending</i> dan tanggal armada tetap terkunci.	Pass
TC-06-03	<i>Pending</i> melewati <i>expiresAt</i>	Data <i>booking</i> atau ketersediaan diakses setelah <i>expiresAt</i> terlewati.	Sistem mengubah <i>booking</i> menjadi dibatalkan dan tanggal tersedia kembali.	Pass
TC-06-04	<i>Booking</i> telah dikonfirmasi	Sistem memeriksa <i>booking</i> dikonfirmasi yang <i>expiresAt</i> -nya telah lewat.	Status tidak berubah dan rentang tanggal tetap mengunci armada.	Pass
TC-06-05	Pemeriksaan <i>cron</i> harian	<i>Endpoint cron</i> dijalankan dengan otorisasi yang valid.	Seluruh <i>booking pending</i> yang kedaluwarsa dibatalkan dan jumlahnya dikembalikan pada respons.	Pass

5.4.7 Pengujian Cek Status Pesanan

Pengujian cek status pesanan dilakukan untuk memastikan pelanggan dapat menemukan reservasi menggunakan nomor HP dan PIN verifikasi. Pengujian mencakup beberapa format PIN, penanganan PIN tidak valid, penyamaran data pribadi, serta ketersediaan formulir testimoni untuk *booking* yang telah selesai. Skenario pengujiannya disajikan pada Tabel 5.13

Tabel 5.13 Pengujian Cek Status Pesanan

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-07-01	PIN empat digit nomor HP	Pelanggan memasukkan nomor HP dan empat digit terakhir nomor tersebut.	Sistem menampilkan <i>booking</i> yang sesuai.	Pass
TC-07-02	PIN tanggal mulai atau ID <i>booking</i>	Pelanggan memasukkan nomor HP beserta PIN DDMM, MMDD, atau ID <i>booking</i> .	Sistem menampilkan <i>booking</i> yang cocok dengan salah satu format PIN.	Pass
TC-07-03	PIN tidak valid	Pelanggan memasukkan nomor HP terdaftar dengan PIN yang tidak cocok.	Sistem tidak menampilkan data <i>booking</i> .	Pass
TC-07-04	Penyamaran data pribadi	Pelanggan berhasil menemukan <i>booking</i> .	Nama pelanggan dan nomor HP ditampilkan dalam bentuk tersamarkan.	Pass
TC-07-05	Hak mengirim testimoni	Pelanggan membuka <i>booking</i> selesai yang belum memiliki testimoni.	Sistem menampilkan formulir testimoni untuk <i>booking</i> tersebut.	Pass

5.4.8 Pengujian Manajemen *Booking* dan Ekspor Excel

Pengujian manajemen *booking* dan ekspor Excel dilakukan untuk memastikan admin dapat memfilter data, memperbarui status operasional dan pembayaran, serta mengekspor data yang sedang ditampilkan. Pengujian juga memastikan *server* menolak status yang tidak terdaftar agar konsistensi data tetap terjaga. Skenario pengujiannya disajikan pada Tabel 5.14

Tabel 5.14 Pengujian Manajemen *Booking* dan Ekspor Excel

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-08-01	Filter status <i>booking</i>	Admin memilih status <i>pending</i> , dikonfirmasi, selesai, atau dibatalkan.	Tabel hanya menampilkan <i>booking</i> sesuai status yang dipilih.	Pass
TC-08-02	Pembaruan status operasional	Admin memilih status operasional yang valid pada <i>booking</i> .	Sistem menyimpan status baru dan memuat ulang data <i>booking</i> .	Pass
TC-08-03	Status tidak valid	Pemintaan PATCH dikirim dengan status di luar daftar yang diizinkan.	Server menolak permintaan dan tidak mengubah data <i>booking</i> .	Pass
TC-08-04	Pembaruan status pembayaran	Admin memilih status <i>booking</i> , DP, lunas, atau penuh.	Sistem menyimpan status pembayaran yang dipilih.	Pass
TC-08-05	Ekspor data terfilter	Admin menerapkan filter lalu menjalankan ekspor Excel.	Sistem mengunduh file <i>booking_YYYY-MM-DD.xlsx</i> berisi data yang sedang terfilter.	Pass

5.4.9 Pengujian *Monitoring Ketersediaan*

Pengujian *monitoring* ketersediaan dilakukan untuk memastikan admin dapat memantau penggunaan setiap armada melalui kalender bulanan. Pengujian mencakup pemilihan armada, perubahan periode bulan, penandaan tanggal berdasarkan status pembayaran, dan pemberian indikator apabila ditemukan jadwal yang bertabrakan. Skenario pengujiannya disajikan pada Tabel 5.15

Tabel 5.15 Pengujian *Monitoring Ketersediaan*

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-09-01	Memilih armada	Admin memilih salah satu armada pada halaman ketersediaan.	Sistem menampilkan kalender <i>booking</i> milik armada yang dipilih.	Pass
TC-09-02	Mengganti periode bulan	Admin memilih bulan sebelumnya atau berikutnya.	Sistem mengambil dan menampilkan <i>booking</i> pada bulan yang dipilih.	Pass
TC-09-03	Status kalender	Kalender memiliki <i>booking</i> dengan status pembayaran <i>booking</i> , DP, lunas, atau penuh.	Tanggal ditampilkan dengan warna yang sesuai dengan legenda status pembayaran.	Pass
TC-09-04	Indikator jadwal bentrok	Terdapat lebih dari satu <i>booking</i> pada armada dan tanggal yang sama.	Sistem menampilkan penanda bentrok dan informasi <i>booking</i> terkait.	Pass

5.4.10 Pengujian Testimoni, Notifikasi, dan Pengaturan

Pengujian ini dilakukan untuk memastikan testimoni hanya dapat dikirim melalui *booking* yang valid dan telah selesai, serta satu *booking* hanya memiliki satu testimoni. Pengujian juga mencakup moderasi testimoni oleh admin, pembuatan dan pengelolaan notifikasi *booking* baru, serta penyimpanan pengaturan situs menggunakan metode *upsert*. Skenario pengujiannya disajikan pada Tabel 5.16

Tabel 5.16 Pengujian Testimoni, Notifikasi, dan Pengaturan

Test Case ID	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Status
TC-10-01	Mengirim testimoni valid	Pelanggan mengirim <i>rating</i> dan ulasan untuk <i>booking</i> selesai dengan nomor HP yang sesuai.	Sistem menyimpan testimoni dengan <i>approved</i> bernilai <i>false</i> .	Pass
TC-10-02	Testimoni <i>booking</i> belum selesai	Pelanggan mengirim testimoni untuk <i>booking pending</i> atau dikonfirmasi.	Sistem menolak testimoni dan menampilkan pesan validasi.	Pass
TC-10-03	Testimoni nomor HP tidak cocok	Pelanggan mengirim testimoni dengan nomor HP yang berbeda dari <i>booking</i> .	Sistem menolak penyimpanan testimoni.	Pass
TC-10-04	Testimoni ganda	Pelanggan kembali mengirim testimoni untuk <i>booking</i> yang sama.	Sistem menolak karena satu <i>booking</i> hanya dapat memiliki satu testimoni.	Pass
TC-10-05	Moderasi testimoni	Admin menyetujui testimoni yang masih menunggu moderasi.	Status <i>approved</i> berubah dan testimoni dapat ditampilkan pada halaman publik.	Pass
TC-10-06	Notifikasi <i>booking</i> baru	Pelanggan berhasil membuat <i>booking</i> baru.	Sistem membuat Notification yang terhubung dengan <i>booking</i> dan menampilkannya kepada admin.	Pass
TC-10-07	Mengelola notifikasi	Admin menandai notifikasi sebagai dibaca atau menghapus seluruh notifikasi.	Status <i>read</i> diperbarui atau data notifikasi terhapus sesuai aksi admin.	Pass
TC-10-08	Menyimpan pengaturan situs	Admin mengubah nomor WhatsApp atau informasi bisnis.	Sistem melakukan <i>upsert</i> SiteSetting berdasarkan key dan menyimpan value terbaru.	Pass

5.4.11 Rekapitulasi Hasil Pengujian *Black Box*

Berdasarkan hasil pengujian fungsional yang telah dilakukan, seluruh skenario pengujian memperoleh hasil sesuai dengan yang diharapkan.

Tabel 5.17 Rekapitulasi Hasil Pengujian *Black Box*

No	Kelompok Pengujian	Rentang Test Case	Jumlah Test Case	Berhasil	Gagal	Persentase Keberhasilan
1	Katalog dan Reservasi	TC-01-01 s.d TC-01-05	5	5	0	100%
2	Formulir dan Ringkasan Reservasi	TC-02-01 s.d TC-02-05	5	5	0	100%
3	Validasi Ketersediaan Armada	TC-03-01 s.d TC-03-05	5	5	0	100%
4	Perhitungan Harga Berdasarkan Zona	TC-04-01 s.d TC-04-04	4	4	0	100%
5	Integrasi WhatsApp <i>Click to Chat</i>	TC-05-01 s.d TC-05-04	4	4	0	100%
6	Pembatalan Otomatis <i>Booking Pending</i>	TC-06-01 s.d TC-06-05	5	5	0	100%
7	Cek Status Pesanan	TC-07-01 s.d TC-07-05	5	5	0	100%
8	Manajemen <i>Booking</i> dan Ekspor Excel	TC-08-01 s.d TC-08-05	5	5	0	100%
9	<i>Monitoring</i> Ketersediaan Armada	TC-09-01 s.d TC-09-04	4	4	0	100%
10	Testimoni, Notifikasi, dan Pengaturan	TC-10-01 s.d TC-10-08	8	8	0	100%
Total			50	50	0	100%

Berdasarkan rekapitulasi tersebut, sistem informasi reservasi kendaraan Rahayu Trans Travel Bandung telah berjalan sesuai kebutuhan fungsional yang telah dirancang. Fitur utama seperti katalog armada, reservasi, validasi jadwal, integrasi WhatsApp, *dashboard* admin, pengelolaan *booking*, testimoni, notifikasi, dan ekspor data dapat digunakan sesuai fungsi masing-masing.

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 dan autentikasi disimpan dalam bentuk *environment variable* pada *dashboard* Vercel. Hal ini dilakukan agar data sensitif seperti URL *database* dan *secret* autentikasi tidak ditulis langsung di dalam *source code*.

Setelah proses *deployment* selesai, aplikasi dapat diakses melalui URL *deployment*. Pelanggan dapat melihat katalog armada, mengisi reservasi, mengirim pesanan melalui WhatsApp, dan mengecek status *booking*. Admin dapat *login* ke *dashboard* untuk mengelola armada, zona wilayah, harga zona, *booking*, testimoni, notifikasi, dan pengaturan situs.

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil analisis, perancangan, implementasi, dan pengujian yang telah dilakukan, sistem informasi reservasi kendaraan berbasis web untuk Rahayu Trans Travel Bandung berhasil dibangun menggunakan Next.js, TypeScript, PostgreSQL, dan Prisma ORM. Sistem telah mengubah penyajian informasi armada yang sebelumnya bersifat statis menjadi katalog dinamis yang menampilkan foto, spesifikasi, kapasitas, tahun kendaraan, transmisi, harga, dan status ketersediaan. Pelanggan dapat melakukan pencarian armada berdasarkan jadwal, zona tujuan, dan jumlah penumpang, serta menggunakan filter kategori, kapasitas, harga, dan pengurutan untuk memperoleh kendaraan yang sesuai dengan kebutuhan perjalanan.

Proses reservasi telah diimplementasikan melalui formulir digital yang menstandarkan data pelanggan dan perjalanan sebelum diteruskan kepada admin melalui WhatsApp *Click to Chat*. Sistem memvalidasi kelengkapan data, minimal hari sewa berdasarkan zona tujuan, serta ketersediaan tanggal untuk setiap armada guna mencegah terjadinya *double booking*. Estimasi biaya dihitung berdasarkan tarif kombinasi armada dan zona melalui HargaZona dengan harga dasar armada sebagai nilai cadangan. *Booking* disimpan dengan status *pending* dan batas waktu *expiresAt* selama 24 jam. *Booking pending* yang melewati batas tersebut dibatalkan ketika data *booking* atau ketersediaan diakses, sedangkan Vercel *Cron* menjalankan pemeriksaan harian sebagai proses cadangan agar tanggal armada dapat tersedia kembali.

Dashboard admin telah menyediakan fungsi pengelolaan data armada, zona wilayah, harga zona, *booking*, status pembayaran, testimoni, notifikasi, pengaturan situs, dan ekspor data Excel. Admin juga dapat memantau kalender ketersediaan setiap armada berdasarkan data *booking* yang belum dibatalkan serta melihat indikator status pembayaran dan peringatan jadwal bentrok. Berdasarkan pengujian *Black Box* pada Bab 5, seluruh skenario yang telah dijalankan menghasilkan keluaran sesuai dengan hasil yang diharapkan. Dengan demikian, sistem yang dibangun telah memenuhi tujuan tugas akhir dalam menyediakan informasi armada, mendukung reservasi digital yang terstruktur, dan membantu admin mengelola kegiatan operasional reservasi secara terpusat.

6.2 Saran

Pengembangan selanjutnya disarankan untuk menambahkan fasilitas pembayaran digital atau unggah bukti pembayaran secara langsung pada *website*. Pengembangan tersebut dapat mengurangi ketergantungan terhadap pemeriksaan bukti transfer melalui percakapan WhatsApp dan membantu sistem mencatat waktu, nominal, serta status verifikasi pembayaran secara lebih terstruktur. Sistem juga dapat dilengkapi dengan notifikasi otomatis kepada pelanggan dan admin menjelang batas *expires*. Agar pelanggan memperoleh pengingat sebelum *booking* dibatalkan.

Pemeriksaan *booking* kedaluwarsa pada implementasi saat ini menggunakan pemeriksaan saat data diakses dan Vercel *Cron* harian sebagai proses cadangan. Pada pengembangan berikutnya, proses tersebut dapat menggunakan layanan penjadwalan dengan interval yang lebih sering agar pembatalan berlangsung lebih mendekati batas 24 jam. Selain itu, pengembangan dapat mencakup pencadangan basis data berkala, pencatatan aktivitas admin, laporan pendapatan berdasarkan periode, serta pengujian lebih lanjut yang melibatkan pengguna dan pemilik usaha untuk menilai kemudahan penggunaan, kecepatan layanan, dan kesesuaian sistem dengan proses operasional Rahayu Trans Travel.

DAFTAR PUSTAKA

- [AUT24] Auth.js Contributors, "Auth.js (NextAuth.js) Documentation", 2024. Tersedia: <https://authjs.dev/>
- [FRE19] Freeman, A., "Pro React 16", Apress, 2019. doi: 10.1007/978-1-4842-4451-7.
- [GEA23] Geasela, Y. M., Lijadi, T., Tanria, D., Drisella, T., & Dagara, B. R., "Rancang Bangun Aplikasi Penyewaan Mobil Online 'Eazyrent' Berbasis Website", Infotech: Journal of Technology Information, Vol. 9, No. 2, pp. 145-154, 2023. doi: 10.37365/jti.v9i2.193.
- [MAU24] Maulana, R. R., Sumaryana, Y., & Anwar, D. S., "Perancangan User Interface dan User Experience Aplikasi Rental Mobil Indocar Berbasis Mobile Menggunakan Metode Design Thinking", Jurnal Informatika dan Teknik Elektro Terapan, Vol. 12, No. 3, pp. 1796-1805, 2024. doi: 10.23960/jitet.v12i3.4442.
- [MIC24] Microsoft, "TypeScript Documentation: The Handbook", 2024. Tersedia: <https://www.typescriptlang.org/docs/>.
- [MOT24] Framer B.V., "Motion Documentation", 2024. Tersedia: <https://motion.dev/docs/>
- [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, 2015. doi: 10.33197/jitter.vol1.iss3.2015.62.
- [NAL24] Nalendra, I. P., & Swalaganata, G., "Perancangan Prototype UI/UX Aplikasi Home Service Kendaraan Berbasis Mobile dengan Pendekatan Design Thinking", Journal of Information System and Application Development, Vol. 2, No. 1, pp. 31-41, 2024. doi: 10.26905/jisad.v2i1.11044.
- [NAU22] Raihan, N., & Eviyanti, A., "Perancangan Sistem Informasi Penyewaan Mobil Berbasis Web Menggunakan Framework CodeIgniter", ZONAsi: Jurnal Sistem Informasi, Vol. 4, No. 1, pp. 31-41, 2022. doi: 10.31849/zn.v4i1.9547.
- [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, pp. 29-50, 2012. doi: 10.5121/ijesa.2012.2204.
- [NOR13] Norman, D., "The Design of Everyday Things: Revised and Expanded Edition", Basic Books, New York, 2013.
- [NUG24] Nugroho, R. P. A., & Sugandi, A., "Meningkatkan Performa Frontend dengan Menggunakan Framework Next.js dalam Pengembangan Website", Journal of Cyber Health and Computer, Vol. 2, No. 2, pp. 14-19, 2024. doi: 10.64163/jochac.v2i2.31.
- [OBR10] O'Brien, J. A., & Marakas, G. M., "Management Information Systems", Edisi 10, McGraw-Hill/Irwin, 2010.
- [PAN24] Pangestu, B. R., & Voutama, A., "Rancang Bangun Sistem Informasi Rental Mobil Berbasis Website", JATI (Jurnal Mahasiswa Teknik Informatika), Vol. 8, No. 3, pp. 2586-2592, 2024. doi: 10.36040/jati.v8i3.9497.
- [POS24] PostgreSQL Global Development Group, "PostgreSQL 16 Documentation", 2024. Tersedia: <https://www.postgresql.org/docs/>.
- [PRE20] Pressman, R. S., & Maxim, B. R., "Software Engineering: A Practitioner's Approach", Edisi 9, McGraw-Hill Education, New York, 2020.
- [PRI24] Prisma Data, Inc., "Prisma ORM Documentation", 2024. Tersedia: <https://www.prisma.io/docs/>
- [REA24] React Contributors, "React Documentation", 2024. Tersedia: <https://react.dev/>.
- [REC24] Recharts Contributors, "Recharts: A Composable Charting Library Built on React Components", 2024. Tersedia: <https://recharts.org/>
- [RHF24] React Hook Form Contributors, "React Hook Form Documentation", 2024. Tersedia: <https://react-hook-form.com/>
- [SHE23] SheetJS., "SheetJS Community Edition Documentation", 2023. Tersedia: <https://docs.sheetjs.com/>.
- [SIR24] Siregar, N. M., & Fatmawati, "Perancangan Sistem Informasi Penyewaan Mobil Berbasis Web pada Gumilar Rent and Tour", Information Management for Educators and Professionals, Vol. 9, No. 1, pp. 21-30, 2024. doi: 10.51211/imbi.v9i1.2890.

- [TAI24] Wathan, Adam, "Tailwind CSS Documentation", Tailwind Labs Inc., 2024. Tersedia: <https://tailwindcss.com/docs/>.
- [TAI25] Tailwind Labs, Inc., "Tailwind CSS v4.0", 2025. Tersedia: <https://tailwindcss.com/blog/tailwindcss-v4>.
- [VCD26] Vercel, Inc., "Vercel CDN", 2026. Tersedia: <https://vercel.com/docs/cdn>.
- [VER24] Vercel, Inc., "Next.js Documentation: App Router", 2024. Tersedia: <https://nextjs.org/docs/app>.
- [VGT26] Vercel, Inc., "Git Integrations", 2026. Tersedia: <https://vercel.com/docs/git>.
- [WAN24] Wana, S., Chandra, A., & Iryani, J., "Sistem Informasi Rental Mobil Berbasis Web Menggunakan WhatsApp Gateway pada Rental Mobil Sasha", Lajutek: Journal of Informatics, Multimedia, and Information, Vol. 2, No. 2, pp. 41-55, 2024. doi: 10.61912/lajutek.v2i2.106.
- [WHA24] WhatsApp LLC., "WhatsApp Click to Chat API Documentation", 2024. Tersedia: <https://faq.whatsapp.com/5913398998672934/>.
- [ZOD24] Zod Contributors, "Zod Documentation: TypeScript-first Schema Validation", 2024. Tersedia: <https://zod.dev/>.

LAMPIRAN

BERITA ACARA WAWANCARA

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

Nama : Miftah Fauzi

NRP : 223040039

Fakultas : Teknik Informatika

Universitas: Universitas Pasundan

Telah melakukan wawancara dengan:

Nama : Cecep Hery

Jabatan : Owner

Perusahaan: Rahayu Trans Travel Bandung

Dalam rangka penyusunan Tugas Akhir semester VII tahun akademik 2025-2026 yang berjudul "Pengembangan Website Rental Mobil Dengan Fitur Reservasi Online Berbasis Next.Js (Studi Kasus: Rahayu Trans Travel Bandung)".

Bandung, 31 Mei 2026

Pewawancara



Miftah Fauzi

Yang Diwawancara,



Catatan wawancara:

Rahayu Trans Travel telah beroperasi sejak tahun 2020 dan saat ini mengelola 24 Armada kendaraan yang aktif di wilayah

- Perusahaan Rahayu telah memiliki website (informasi: rute, nomor tidak menentu) dalam rangka dan proses pemesanan. Pada akhirnya telah sehingga bergantung pada tanya jawab via whatsapp, dan pencatatan operasional masih dilakukan manual menggunakan microsoft Excel dan juga kalender fisik.
- Permasalahan utama yang kerap terjadi meliputi Penutupan pesan pelanggan, pengumpulan informasi pemesanan yang tidak lengkap, serta pengalihan pesan terjadinya double booking.
- Pihak Rahayu Travel menambahkan pengembangan sistem baru yang menghasilkan katalog armada, rutean harga sewa, serta status ketersediaan kendaraan. Mayor pelanggan tidak perlu selalu bertanya dari awal.
- Formulir pesanan yang diisi pelanggan di website diharapkan bisa langsung dikirim ke whatsapp admin dengan format yang rapi dan detail.
- Proses pembayaran dari pelanggan akan tetap menggunakan metode transfer manual konvensional karena pemilik belum ingin menggunakan pembayaran otomatis pada website.
- Harapan utama narasumber adalah website yang akan dibangun mudah digunakan dan bisa membantu promosi usaha pemilik.

Bandung, 31 Mei 2026

Pewawancara



Miftah Fauzi

Yang Diwawancara,



Lampiran B Berita acara



Lampiran C Dokumentasi Wawancara