BAB IV

ANALISA DAN PERANCANGAN APLIKASI

4.1 Analisa Kebutuhan

4.1.1 Analisis Kebutuhan Aplikasi

Analisis kebutuhan dilakukan untuk memastikan sistem "BookingYuk!" mampu menjawab permasalahan keamanan dan efisiensi dalam penyewaan mobil. Kebutuhan utama sistem ini dibagi menjadi dua sisi pengguna:

1. Sisi Penyewa (Renter):

a. Pengguna dapat mencari mobil berdasarkan lokasi terdekat menggunakan *Haversine* Formula.

b. Pengguna dapat menerima rekomendasi mobil terbaik berdasarkan kriteria harga, jarak, kapasitas, dan fasilitas menggunakan metode *Simple Additive Weighting (SAW).*

c. Pengguna dapat melakukan registrasi dan autentikasi akun secara aman.

d. Pengguna yang belum memiliki status verified wajib melakukan proses *Electronic Know Your Customer (e-KYC)* dengan mengunggah dokumen identitas berupa KTP, SIM, swafoto bersama KTP, foto rumah sebagai bukti domisili, serta melakukan verifikasi biometrik wajah.

e. Pengguna yang telah memiliki status *verified* hanya diwajibkan melakukan verifikasi biometrik wajah secara cepat pada setiap transaksi berikutnya.

f. Pengguna wajib memberikan persetujuan terhadap pemrosesan data pribadi sebelum proses verifikasi identitas dilakukan.

g. Pengguna dapat melakukan pemesanan kendaraan, melihat status pemesanan, serta melakukan pembayaran sesuai mekanisme yang disediakan sistem.

h. Pengguna dapat memperoleh rekomendasi dan bantuan melalui asisten berbasis *Generative Artificial Intelligence*.

2. Sisi Pemilik Rental (Owner):

a. Pengguna dapat mendaftarkan, mengubah, dan mengelola data armada kendaraan.

b. Pengguna dapat melihat dan memvalidasi hasil verifikasi identitas penyewa berdasarkan dokumen *e-KYC* dan hasil verifikasi biometrik wajah.

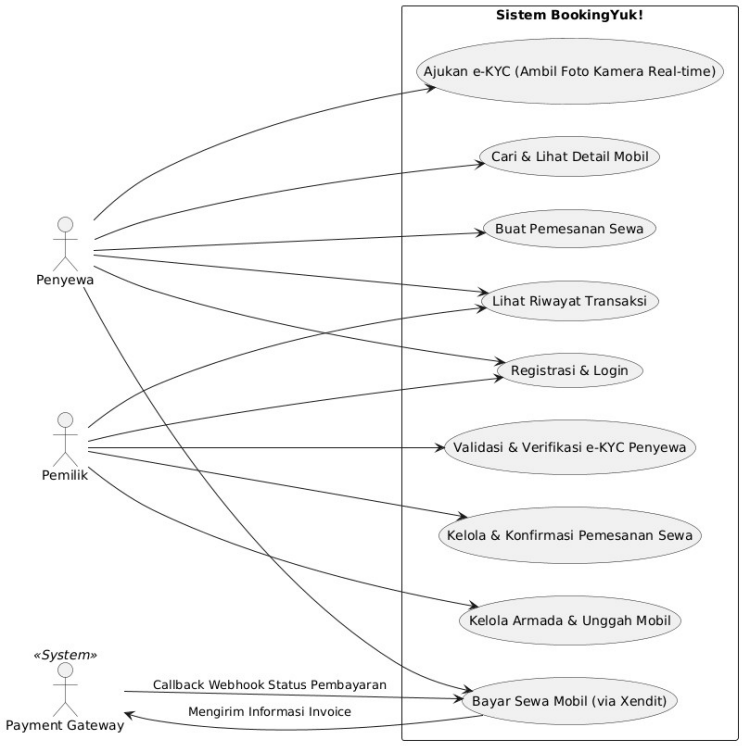
c. Pengguna dapat menyetujui atau menolak permohonan penyewaan berdasarkan hasil verifikasi identitas dan kelengkapan dokumen penyewa.

d. Pengguna dapat mengelola data transaksi penyewaan serta memantau status kendaraan.

e. Pengguna dapat melihat laporan penyewaan, analisis keuntungan, serta performa bisnis yang dihasilkan secara otomatis melalui *AI Gemini.*

f. Pengguna dapat mendaftarkan akun keuangan sub-account Xendit, mengonfigurasi rekening bank tujuan penarikan, memantau akumulasi saldo hasil transaksi persewaan secara real-time, serta melakukan penarikan dana (*withdrawal*) langsung ke rekening bank terdaftar.

4.1.2 Rancangan Diagram *Use Case*



Gambar 4.1 *Diagram Use*
Sumber : hasil penelitian (2026)

Use Case ini mendeskripsikan interaksi antara aktor (Penyewa dan Pemilik) dengan sistem BookingYuk!.

Aktor: Penyewa, Pemilik (Owner) Sistem *Payment Gateway*.

Fungsi Utama Penyewa:

1. Melakukan registrasi dan login akun.
2. Mengajukan proses *Electronic Know Your Customer (e-KYC)* melalui pengambilan foto menggunakan kamera secara *real-time*.

3. Melakukan pencarian dan melihat detail kendaraan.

4. Membuat pemesanan penyewaan kendaraan.

5. Melihat riwayat transaksi penyewaan.

6. Melakukan pembayaran sewa kendaraan melalui *Payment Gateway (Xendit).*

Fungsi Utama Pemilik :

1. Melakukan registrasi dan login akun.

2. Memvalidasi dan memverifikasi hasil *e-KYC* penyewa.

3. Mengelola armada kendaraan serta mengunggah data mobil.

4. Mengelola dan mengonfirmasi permintaan penyewaan kendaraan.

Fungsi Sistem *Payment Gateway*:

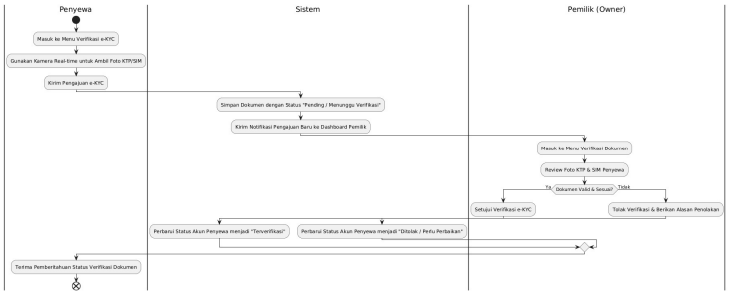
1. Menerima permintaan pembuatan *invoice* pembayaran dari sistem BookingYuk!.

2. Mengirimkan informasi *invoice* kepada pengguna.

3. Mengirimkan *callback webhook* sebagai informasi status pembayaran sehingga sistem dapat memperbarui status transaksi secara otomatis.

4.1.3 Rancangan Diagram Aktivitas

1. Verifikasi e-KYC



Gambar 4.2 *Diagram Aktivitas e-KYC*
Sumber : hasil penelitian (2026)

Diagram aktivitas menggambarkan alur kerja sistem, khususnya pada proses penyewaan:

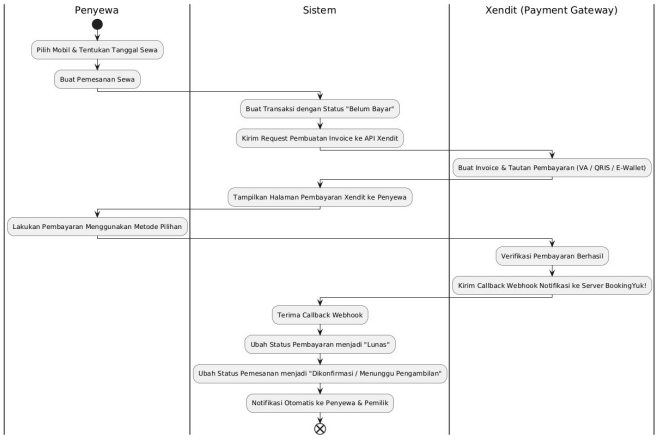
a. Penyewa membuka menu verifikasi *e-KYC*, kemudian mengambil foto KTP, SIM, dan swafoto menggunakan kamera secara real-time.

b. Sistem menyimpan dokumen dengan status Pending dan mengirimkan notifikasi pengajuan kepada Pemilik (Owner) untuk dilakukan proses verifikasi.

c. Pemilik memeriksa dokumen identitas yang diunggah. Jika dokumen valid, sistem mengubah status akun penyewa menjadi Terverifikasi . Jika tidak valid, sistem mengubah status menjadi Ditolak atau Perlu Perbaikan.

d. Setelah akun terverifikasi, penyewa dapat melanjutkan proses penyewaan kendaraan, mulai dari mencari mobil, membuat pemesanan, melakukan pembayaran, hingga transaksi penyewaan dapat diproses oleh pemilik.

2. Diagram aktivitas pembayaran



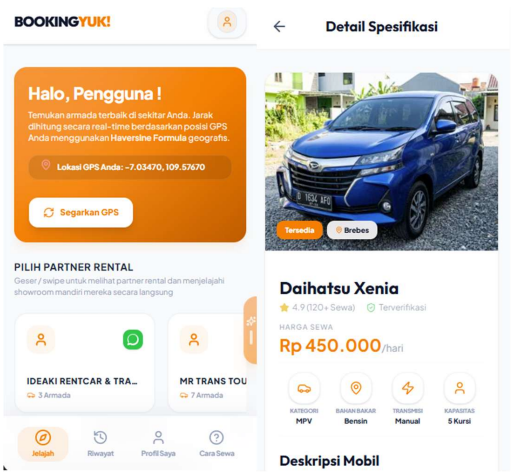
Gambar 4.3 *Diagram Aktivitas Pembayaran*
Sumber : hasil penelitian (2026)

Diagram aktivitas pembayaran menggambarkan alur proses pembayaran sewa kendaraan melalui layanan *Payment Gateway Xendit*, mulai dari pembuatan transaksi hingga pembayaran berhasil dikonfirmasi oleh system, berikut alurnya

a. Penyewa memilih mobil, menentukan tanggal penyewaan, kemudian membuat pemesanan sewa.
b. Sistem membuat data transaksi dengan status "Belum Bayar" dan mengirimkan permintaan pembuatan *invoice ke API Xendit*.
c. Xendit menghasilkan invoice beserta tautan pembayaran yang dapat menggunakan berbagai metode, seperti *Virtual Account (VA), QRIS*, atau *E-Wallet*, kemudian mengirimkan informasi tersebut kembali ke sistem.
d. Sistem menampilkan halaman pembayaran kepada penyewa, kemudian penyewa melakukan pembayaran menggunakan metode yang dipilih.
e. Setelah pembayaran berhasil, *Xendit* melakukan verifikasi transaksi dan mengirimkan callback webhook ke server BookingYuk! sebagai pemberitahuan bahwa pembayaran telah diterima.
f. Sistem menerima *callback webhook*, kemudian memperbarui status pembayaran menjadi "Lunas" dan mengubah status pemesanan menjadi "Dikonfirmasi / Menunggu Pengambilan".
g. Terakhir, sistem secara otomatis mengirimkan notifikasi kepada penyewa dan pemilik (owner) sebagai informasi bahwa pembayaran telah berhasil diproses dan kendaraan siap untuk proses pengambilan.

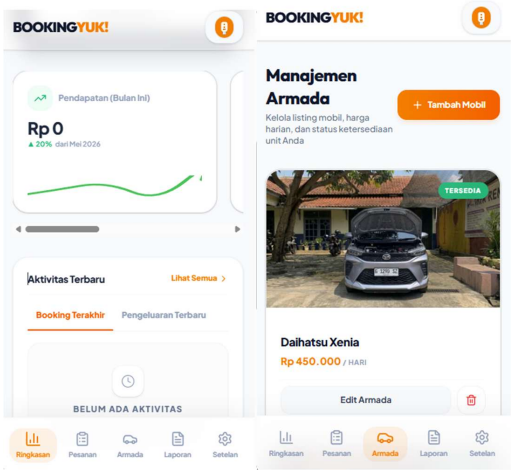
4.1.4 Rancangan User Interface (MockUp)

1. Desain Antarmuka Penyewa



Gambar 4.4. Desain antar muka penyewa
Sumber : hasil penelitian (2026)

2. Desain Antarmuka Pemilik Rental



Gambar 4.5. Desain Antarmuka Pemilik Rental
Sumber : hasil penelitian (2026)

Desain antarmuka pengguna (UI) pada aplikasi pemesanan rental mobil dirancang agar mudah digunakan, responsif, dan mampu memberikan pengalaman penggunaan yang intuitif bagi pelanggan maupun admin. Setiap elemen antarmuka dibuat dengan mempertimbangkan prinsip user-*centered design*, kesederhanaan navigasi, serta konsistensi tampilan sehingga pengguna dapat mengoperasikan aplikasi tanpa memerlukan pengetahuan teknis yang mendalam.

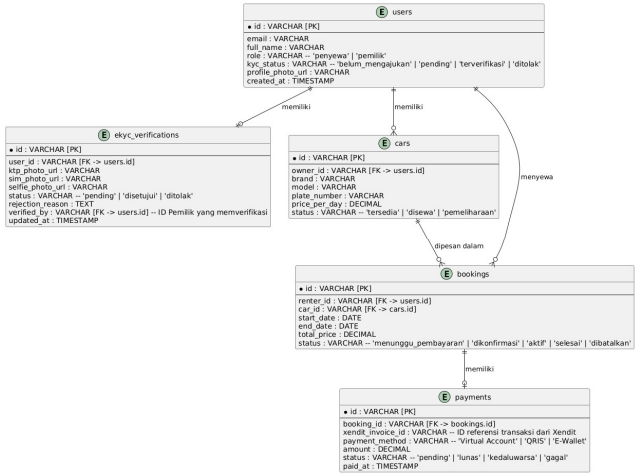
Antarmuka pelanggan terdiri dari beberapa halaman utama, dimulai dari halaman login dan registrasi yang menyediakan formulir sederhana untuk masuk dan membuat akun baru. Setelah berhasil masuk, pengguna diarahkan ke halaman beranda yang menampilkan menu navigasi serta rekomendasi mobil. Halaman inti adalah halaman katalog mobil, di mana daftar mobil ditampilkan dalam bentuk kartu berisi foto kendaraan, harga sewa, kapasitas, dan status ketersediaan. Pengguna dapat memilih mobil dan diarahkan ke halaman detail mobil, yang berisi informasi lengkap serta tombol untuk melakukan pemesanan. Proses pemesanan berlangsung di halaman formulir pemesanan, di mana pengguna memilih tanggal sewa, tanggal kembali, melihat total biaya, dan mengunggah bukti pembayaran. Setelah itu, pengguna dapat memantau status pemesanan melalui halaman riwayat pemesanan yang menampilkan daftar transaksi beserta status verifikasinya.

Antarmuka bagi admin memiliki struktur yang lebih berfokus pada pengelolaan data. Admin diarahkan ke dashboard Admin, yang menampilkan ringkasan jumlah mobil, pesanan masuk, dan laporan penyewaan. Pada halaman kelola mobil, admin dapat menambah mobil baru, mengubah data mobil, atau mengubah status ketersediaannya. Selain itu, terdapat halaman kelola pemesanan yang menampilkan seluruh pesanan dari pelanggan. Admin dapat memverifikasi pembayaran, mengubah status pesanan, serta menandai mobil sebagai “Disewa” atau “Tersedia”. Sementara

itu, halaman laporan memungkinkan admin melihat ringkasan transaksi serta riwayat penyewaan untuk tujuan evaluasi operasional.

Secara keseluruhan, desain antarmuka pengguna dibuat dengan tata letak yang sederhana, penggunaan warna yang konsisten, ikon yang mudah dikenali, serta elemen navigasi yang jelas. Pendekatan ini memastikan bahwa aplikasi dapat digunakan dengan nyaman oleh semua kalangan, baik pelanggan yang membutuhkan proses penyewaan yang cepat dan jelas, maupun admin yang memerlukan sistem pengelolaan data yang efisien.

4.1.5 Rancangan Database



Gambar 4.6 Rancangan database
Sumber : hasil penelitian (2026)

Struktur data :

1. Tabel Users:

   Menyimpan informasi akun pengguna, meliputi ID, Email, Nama Lengkap, Peran (Penyewa/Pemilik), Status *e-KYC*, Foto Profil, serta Tanggal Pembuatan Akun.

2. Tabel eKYC_Verifications:

Menyimpan data hasil pengajuan *Electronic Know Your Customer (e-KYC)* berupa ID Pengguna, Foto KTP, Foto SIM, Swafoto bersama KTP, Status Verifikasi, Alasan Penolakan, ID Pemilik yang Memverifikasi, dan Waktu Pembaruan Data.

3. Tabel Cars:

   Menyimpan informasi kendaraan yang disewakan, meliputi ID Pemilik, Merek, Model, Nomor Polisi, Harga Sewa per Hari, serta Status Kendaraan (tersedia, disewa, atau pemeliharaan).

4. Tabel Bookings:

   Menyimpan data transaksi penyewaan kendaraan, meliputi ID Penyewa, ID Kendaraan, Tanggal Mulai, Tanggal Selesai, Total Harga, serta Status Pemesanan (menunggu pembayaran, dikonfirmasi, aktif, selesai, atau dibatalkan).

5. Tabel Payments:

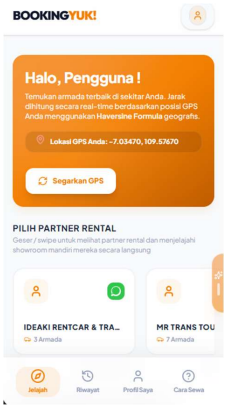
   Menyimpan informasi pembayaran yang terintegrasi dengan *Payment Gateway Xendit*, meliputi *ID Booking*, *ID Invoice Xendit*, Metode Pembayaran, Nominal Pembayaran, Status Pembayaran, dan Waktu Pembayaran.

4.2 Implementasi

4.2.1 Proses Implementasi

Bagian ini menampilkan hasil akhir aplikasi yang telah dikembangkan:

1. Tampilan Halaman Utama: Menampilkan antarmuka bersih dengan peta lokasi.



Gambar 4.7 dashboard penyewa
Sumber : hasil penelitian (2026)

2. Tampilan Fitur AI: Menampilkan chat asisten Gemini yang memberikan saran pemilihan mobil.



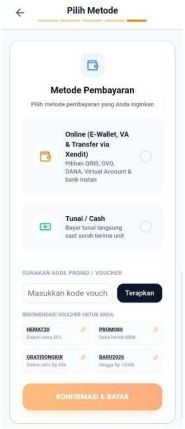
Gambar 4.8 Asisten *AI*
Sumber : hasil penelitian (2026)

3. Tampilan Verifikasi: Menampilkan proses *scanning* wajah yang berhasil divalidasi oleh face-api.js.



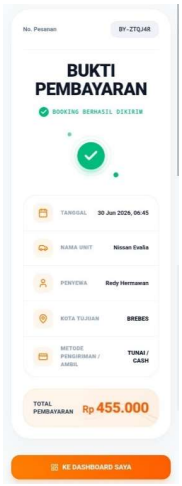
Gambar 4.9 verivikasi
Sumber : hasil penelitian (2026)

4. Tampilan Metode pembayaran : Menampilkan beberapa metode pembayaran yang bisa di pilih oleh penyewa



Gambar 4.10 Pembayaran
Sumber : hasil penelitian (2026)

5. Tampilan Bukti pembayaran : Menampilkan Bukti pembayaran yang bisa di *screenshot* pilih oleh penyewa sebagai bukti.



Gambar 4.11 Bukti Pembayaran
Sumber : hasil penelitian (2026)

4.2.2 Hasil Implementasi

Aplikasi "BookingYuk!" telah dikonfigurasi sebagai Aplikasi yang dapat diinstal langsung melalui di perangkat Android/iOS dan siap pakai.

Gambar 4.12 playstore
Sumber : hasil penelitian (2026)

4.3 Spesifikasi Aplikasi

4.3.1 *Hardware*

Tabel 4.1 Spesifikasi minimum & rekomendasi
Sumber : hasil penelitian (2026)

Kategori Spesifikasi	Spesifikasi Minimum	Spesifikasi Rekomendasi
Sistem Operasi (OS)	Android 7.0	Android 10.0 lebih baru
Memori Utama (RAM)	3 GB	4 GB atau lebih besar
Penyimpanan Internal	Sisa ruang kosong 150 MB	Sisa ruang kosong 500 MB+
Kamera Belakang	8 MP (Dengan dukung Autofokus)	12 MP atau lebih besar (Autofokus & LED Flash)
Kamera Depan	5 MP	8 MP atau lebih besar (Autofokus)
Konektivitas Jaringan	3G / HSPA / Wi-Fi Stabil	4G LTE / 5G / Wi-Fi Kecepatan Tinggi

4.3.2 Software

Tabel 4.2 Sofware yang digunakan
Sumber : hasil penelitian (2026)

No	**Software**	**Versi**
1	Windows 11 Pro 64-bit	24H2
2	Visual Studio Code	Terbaru
3	Node.js	22 LTS
4	npm	10.x
5	React Native	0.79.x
6	TypeScript	5.x
7	Android Studio	Terbaru

4.4 Pengujian Aplikasi

Pengujian Tahapan pengujian aplikasi dilakukan terhadap unit sistem yang telah dikembangkan dengan menggunakan metode Black Box dan White Box. Pengujian ini bertujuan untuk memverifikasi kesesuaian fungsi sistem, mengidentifikasi kesalahan pada logika program, serta memastikan aplikasi dapat beroperasi sesuai dengan kebutuhan pengguna.

4.4.1 Pengujian Black Box

Tabel 4.3 Pengujian Black Box
Sumber : hasil penelitian (2026)

Skenario Pengujian	Langkah Pengujian	Input	Hasil yang Diharapkan	Status
Autentikasi Pengguna & Role	Menekan tombol "Masuk dengan Google" atau beralih tipe peran di halaman registrasi.	Email Gmail Aktif, Pilihan Peran (Penyewa / Owner).	Sistem berhasil mendaftarkan pengguna dan mengarahkan Penyewa ke Dashboard Rental, sedangkan Owner ke Dashboard Mitra	Berhasil
Unggah Berkas & Kamera Verifikasi	Menekan tombol kamera di formulir sewa untuk mengambil foto KTP, SIM, Rumah, dan swafoto.	Tangkapan Kamera / Pilih File Foto	Kamera aktif sesuai tipe kamera dan foto berhasil dimuat di grid pratinjau. Tombol "Ulangi" berfungsi mereset gambar.	Berhasil
Biometric Face	Tombol "Kirim Kelengkapan Dokumen" ditekan untuk memicu analisis pencocokan	Tangkapan Kamera / Pilih File Foto (KTP, SIM, Selfie).	Sistem mendeteksi wajah di foto. Jika biometrik gagal akibat sandbox/CORS, sistem melakukan *graceful fallback* mencocokkan kredensial NIK & Nama, kemudian	Berhasil

	wajah selfie dengan KTP.		meloloskan status verifikasi	
Metode Rental & Form Validasi	Memilih metode Lepas Kunci dan mengosongkan salah satu dokumen wajib seperti SIM.	Pilihan: "Lepas Kunci", NIK 16 digit, SIM kosong.	Tombol "Bayar Sekarang" tetap dinonaktifkan (*disabled*) hingga seluruh dokumen (KTP, SIM, Foto Rumah) diunggah lengkap.	Berhasil
Integrasi Payment Gateway (Xendit)	Menekan tombol transaksi setelah seluruh formulir terisi lengkap.	Klik "Bayar Sekarang" & Trigger Invoice.	Sistem membuat data pesanan berstatus di Firestore, memicu API Xendit, dan membuka Pembayaran langsung di dalam aplikasi.	Berhasil
Pembayaran Selesai	Melakukan pembayaran via simulasi Xendit di dalam visual invoice.	Klik "Simulate Payment" atau memilih metode pembayaran pada gerbang Xendit.	Invoice terbayar, kerangka menutup otomatis, user diarahkan ke halaman Bukti Pembayaran, dan status pesanan di Firestore berubah.	Berhasil
Perhitungan Otomatis Penalty Denda Overtime	Owner melakukan konfirmasi selesai sewa pada mobil yang masa kembalinya melewati batas waktu (Late).	Klik "Selesaikan Sewa" pada pesanan overdue.	Sistem mendeteksi keterlambatan otomatis, memunculkan modal denda berlatar merah, menghitung biaya denda, dan menetapkan status mobil kembali ke Tersedia.	Berhasil

4.4.2 Pengujian White Box

1. Kasus Uji Validasi Cabang

Kasus 1: Model Gagal Dimuat & Data Teks Valid

a. Input: modelsLoaded = false, renterData = { fullName: "Ahmad Dani", NIK: "3275012345678901" }

b. Jalur Logika (Path): Start modelsLoaded == false Tampilkan Peringatan Fallback Validasi Sukses? == YES Beri Skor Simulasi 88%-93% Set Verifikasi Sukses Finish.

Hasil: Lolos verifikasi biometrik dengan aman tanpa crash program.

Kasus 2: Model Berhasil Dimuat & Wajah Tidak Terdeteksi

a. Input: modelsLoaded = true, capturedImageTemp adalah gambar blur kosong/tanpa wajah.

b. Jalur Logika (Path): Start modelsLoaded == true Deteksi Wajah KTP & Selfie Terjadi Error / No face detected Bagian catch (error) terpicu Mengevaluasi Validitas Data Teks (YES) Set Verifikasi Sukses Finish.

Hasil: Sistem bertransisi dengan mulus ke validasi teks agar perjalanan sewa pelanggan tidak tersendat.

2. Pengujian Keputusan Rumus Overtime Denda

Tabel 4.4 pengujian logika
Sumber : hasil penelitian (2026)

Skenario Jalur Uji	Nilai Input Waktu	Kondisi Keputusan	Jalur Kode yang Terkesusi	Nilai
Jalur A (Tepat Waktu)	nowTime <dTime	isLate == false	Blok if (isLate) dilewati.	Rp 0 (Tidak ada denda, muncul banner hijau "TEPAT WAKTU")
Jalur B (Selesai Tepat Batas)	nowTime == endTime	isLate == false	Blok if (isLate) dilewati.	Rp 0 (Tidak ada denda)

Jalur C (Lambat 1 Menit)	now lebih lambat 1 menit dari end	isLate == true	Evaluasi pembulatan ke atas Math.ceil d ari 0.016 jam.	Terhitung 1 jam denda Tarif Denda (misal: Rp 50.000)
Jalur D (Terlamb at 5 Jam)	now lebih lambat 5 jam dari end	isLate == true	Evaluasi pembulatan ke atas Math.ceil d ari 5 jam.	Terhitung 5 jam denda Tarif Denda (misal: 5 Rp 50.000 = Rp 250.000)

3. Ringkasan perbandingan metode pengujian

Tabel 4.5 perbadingan pengujian
Sumber : hasil penelitian (2026)

Karakteristik	Pengujian Black Box	Pengujian White Box
Fokus Utama	Evaluasi antarmuka/UI, respons tombol kamera, integrasi Xendit, dan alur database Firestore.	Evaluasi struktur logika transisi state pesanan, rumus denda, dan blok antisipasi *failure* API Swafoto.
Kriteria Keberhasilan	Tidak ada tombol meluap di mobile screen, transaksi sukses dikonfirmasi, dan data tersimpan di database.	Seluruh cabang kondisi (if, else, catch, fallback) teruji mengeksekusi variabel tanpa memicu pembekuan aplikasi (*crash/freeze*).
Pelaksana Pengujian	Pengguna akhir, tim QA (*Quality Assurance*), klien.	Pengembang sistem (*Software Engineer*), *System Analyst*.