

RANCANG BANGUN APLIKASI MOBILE WAYERHOUSE BERBASIS FLUTTER DAN LARAVEL UNTUK MANAJEMEN INVENTARIS DAN PEMFAKTURAN TERINTEGRASI PADA CV WARAJAYA SOLVE TECHINDO

Made Hanindia Prami Swari ¹, Rico Satyadharma Putra Suryawan ²,
Ferry Irwansyah ³, Moch. Abshar Maulana ⁴, Jagad Samudra ⁵

¹⁻⁵Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional
"Veteran" Jawa Timur

e-mail: madehanindia.fik@upnjatim.ac.id, 23081010257@student.upnjatim.ac.id,
23081010044@student.upnjatim.ac.id, 23081010253@student.upnjatim.ac.id,
23081010259@student.upnjatim.ac.id

Abstract

The rapid advancement of information technology requires Internet Service Providers (ISPs) to continuously improve the quality of their operational services, particularly in terms of centralized company data management. CV Warajaya Solve Techindo, which operates both as an internet service provider and a software house, still faces challenges in inventory recording, client project monitoring, and invoicing processes that remain manual and decentralized. This study aims to design and build a mobile application called WayerHouse as an integrated digital solution to manage inventory, projects, and invoices in real time. The development method applied is the Iterative Agile method, divided into three phases: the foundation phase (database and system architecture design), the construction phase (development of business logic through REST API), and the integration phase (mobile interface development and system testing). The application was built using the Flutter framework for the user interface, Laravel 12 for the REST API, and PostgreSQL hosted on Supabase as the database. Testing was carried out using Postman to verify API endpoints, accompanied by functional testing on physical Android devices. The results show that WayerHouse successfully integrates the inventory, project management, and invoicing modules into a single centralized platform supported by role-based access control and audit logging, thereby improving the efficiency, transparency, and reliability of operational data management at CV Warajaya Solve Techindo.

Article History

Submitted: 15 Juni 2026
Accepted: 18 Juni 2026
Published: 19 Juni 2026

Key Words

Mobile Application;
Flutter; Laravel;
Inventory; Invoicing; Data
Warehouse

Abstrak

Perkembangan teknologi informasi menuntut penyedia layanan internet (Internet Service Provider/ISP) untuk terus meningkatkan kualitas layanan operasional, khususnya dalam hal pengelolaan data perusahaan secara terpusat. CV Warajaya Solve Techindo, sebagai perusahaan penyedia layanan internet sekaligus software house, masih menghadapi kendala berupa proses pendataan inventaris, pemantauan proyek klien, dan pemfakturan yang belum terpusat serta dilakukan secara manual. Penelitian ini bertujuan merancang dan membangun aplikasi mobile bernama WayerHouse sebagai solusi digital terintegrasi untuk mengelola inventaris, proyek, dan invoice secara real-time. Metode pengembangan yang digunakan adalah Agile Iteratif yang dibagi menjadi tiga fase, yaitu fase fondasi (perancangan basis data dan arsitektur sistem), fase konstruksi (pembangunan logika bisnis berbasis REST API), dan fase integrasi (pengembangan antarmuka serta pengujian sistem). Aplikasi dibangun menggunakan framework Flutter untuk antarmuka pengguna, Laravel 12 untuk REST API, dan PostgreSQL yang dihosting pada Supabase sebagai basis data. Pengujian dilakukan menggunakan Postman untuk memverifikasi seluruh endpoint API serta pengujian fungsional langsung pada perangkat Android. Hasil penelitian menunjukkan bahwa aplikasi WayerHouse berhasil mengintegrasikan modul inventaris, manajemen proyek, dan pemfakturan dalam satu platform terpusat yang didukung oleh kontrol akses berbasis peran (role-based access control) serta pencatatan audit log, sehingga mampu meningkatkan efisiensi, transparansi, dan keandalan pengelolaan data operasional pada CV Warajaya Solve Techindo.

Sejarah Artikel

Submitted: 15 Juni 2026
Accepted: 18 Juni 2026
Published: 19 Juni 2026

Kata Kunci

Aplikasi Mobile; Flutter;
Laravel; Inventaris;
Pemfakturan; Data
Warehouse

1. PENDAHULUAN

Kebutuhan akan akses internet yang cepat dan stabil telah menjadi bagian yang tidak terpisahkan dari aktivitas masyarakat modern. Kondisi ini mendorong penyedia layanan internet (Internet Service Provider/ISP) untuk tidak hanya berfokus pada kualitas jaringan, tetapi juga pada optimalisasi layanan pendataan dan kemudahan operasional internal. Pendataan yang terstruktur, mulai dari pembuatan proyek, penerbitan invoice, hingga pelaporan data, menjadi faktor krusial dalam menjaga efisiensi dan keamanan data perusahaan.

CV Warajaya Solve Techindo merupakan perusahaan yang bergerak di bidang penyedia layanan internet sekaligus software house. Dalam praktiknya, proses keluar masuk data produk perusahaan, pembuatan hingga pemantauan status proyek klien, serta proses pemfakturan masih dilakukan tanpa platform terpusat yang dapat diakses pegawai secara real-time. Tanpa adanya sistem yang terintegrasi, pegawai kerap mengalami kesulitan dalam melacak status proyek maupun melakukan pendataan produk perusahaan secara praktis, sehingga proses pelaporan menjadi tidak fleksibel dan harus melalui koordinasi panjang antar-divisi.

Permasalahan tersebut dapat diklasifikasikan ke dalam empat aspek utama, yaitu: (1) belum adanya platform terpusat bagi pegawai untuk memantau status proyek klien; (2) proses pelaporan periodik yang tidak fleksibel karena melibatkan banyak divisi; (3) belum tersedianya sistem pengelolaan inventaris terstruktur untuk memantau pergerakan barang masuk dan keluar serta proses stock opname; dan (4) tidak tersedianya sistem pemfakturan yang terhubung dengan data survei lapangan maupun data inventaris, sehingga proses penagihan rawan terjadi kesalahan pencatatan.

Berangkat dari permasalahan tersebut, penelitian ini bertujuan merancang dan mengembangkan aplikasi mobile bernama WayerHouse, yang dirancang khusus untuk menjawab kebutuhan operasional CV Warajaya Solve Techindo dalam memfasilitasi manajemen inventaris, pemfakturan, dan manajemen proyek secara terintegrasi. Perancangan antarmuka (User Interface) dan pengalaman pengguna (User Experience) WayerHouse mengadopsi prinsip dari platform manajemen bisnis yang telah mapan, seperti Majoo dan Olsera, guna menghadirkan navigasi yang intuitif, familier, dan profesional bagi pegawai (Tidwell, Brewer, & Valencia-Brooks, 2020).

Secara teknis, aplikasi dibangun menggunakan framework Flutter yang memungkinkan satu basis kode (codebase) tunggal menghasilkan antarmuka dengan performa setara aplikasi native pada berbagai perangkat (Kim & Lee, 2023). Antarmuka tersebut diintegrasikan dengan REST API berbasis Laravel sebagai pusat logika bisnis, serta basis data PostgreSQL untuk menjamin keandalan dan skalabilitas penyimpanan data. Proses rekayasa perangkat lunak dilaksanakan dengan pendekatan Agile Iteratif (Acuna, Gomez, & Juristo, 2018), yang memungkinkan pengembangan berjalan secara adaptif dan terstruktur melalui evaluasi berkala pada setiap iterasi. Melalui penelitian ini, aplikasi WayerHouse diharapkan dapat menjadi solusi digital yang meningkatkan efisiensi serta kualitas tata kelola data operasional pada CV Warajaya Solve Techindo.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan studi kasus pengembangan perangkat lunak (case study research and development) yang dilaksanakan secara langsung pada lingkungan kerja CV Warajaya Solve Techindo. Tahapan penelitian meliputi analisis kebutuhan sistem, perancangan arsitektur dan basis data, implementasi, hingga pengujian sistem.

2.1 Metode Pengembangan Sistem

Mengingat kompleksitas kebutuhan operasional WayerHouse yang mencakup seluruh data perusahaan serta data proyek klien, pengembangan sistem dilaksanakan melalui pendekatan Agile Iteratif yang terbagi ke dalam tiga fase strategis utama, sebagaimana ditunjukkan pada penjelasan berikut.

- Fase Fondasi: berfokus pada perancangan skema basis data (database schema), pendefinisian relasi antar tabel, serta konfigurasi infrastruktur server dan koneksi ke basis data PostgreSQL. Pada fase ini dilakukan pula analisis mendalam terhadap alur operasional perusahaan, termasuk pembagian peran (role) pegawai dan alur data proyek hingga pemfakturan.

TABEL I
ANALISIS KEBUTUHAN APLIKASI

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
1. INFORMASI UMUM	Nama Proyek	Aplikasi Mobile Warehouse Inventory & Invoicing
	Platform	Android
	Versi Dokumen	1.0
2. PENDAHULUAN	2.1 Latar Belakang	CV Warajaya Solve Techindo bergerak di bidang infrastruktur IT dengan operasional yang melibatkan pengelolaan stok perangkat keras (hardware), komponen jaringan, dan aset IT secara intensif. Proses pencatatan stok masuk/keluar serta pembuatan invoice kepada klien selama ini masih dilakukan secara manual menggunakan spreadsheet atau dokumen fisik.
	2.2 Tujuan	Menyediakan sistem manajemen inventaris dan invoicing berbasis web yang dapat terintegrasi Mobile.
	2.3 Ruang Lingkup	Dibangun menggunakan framework Flutter (cross-platform Android), terintegrasi dengan RESTful API backend berbasis Laravel, dan terhubung ke database terpusat.
3. DESKRIPSI SISTEM	3.1 Aktor / Pengguna	Super Admin/Direktur dan Developer: Pengguna dengan akses penuh, mengelola master data barang, kategori, supplier, laporan, dan pengaturan akun dan semua aksi pada dashboard yang dimiliki masing-masing role. Admin/Manajer Gudang:

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		Pengguna dengan akses; mengelola master data barang, kategori, supplier, laporan. Finance: Pengguna yang membuat, mengedit, mengirim, dan memonitor status invoice kepada Super Admin, Admin, Project Manager, terdapat view piutang per client. Project Manager: Membuat Laporan Survey dan Project, Menerima dan mengelola Penugasan dari Super Admin, hanya melihat invoice dan data barang.
	3.2 Lingkungan Operasi	Dibangun menggunakan framework Flutter, serta pengembangan aplikasi mobile sisi user (frontend) yang terintegrasi dengan RESTful API backend perusahaan.
4. KEBUTUHAN FUNGSIONAL	REQ-F-01 (Autentikasi)	Pengguna dapat login menggunakan Email atau Nomor Telepon yang terdaftar, dengan verifikasi password terenkripsi. Tersedia fitur Lupa Password melalui email reset link. • Sistem menerapkan JWT (JSON Web Token) sebagai mekanisme autentikasi API. • Sesi pengguna otomatis kedaluwarsa setelah 8 jam tidak aktif (configurable). • Setiap percobaan login yang gagal dicatat dalam log sistem.
	REQ-F-02 (Dashboard Utama)	Sistem menampilkan ringkasan kondisi gudang secara real-time: • Total stok aset aktif di warehouse (jumlah item dan nilai estimasi). • Jumlah invoice aktif berdasarkan status: Draft, Menunggu Persetujuan, Disetujui, Ditolak, Lunas.

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		• Notifikasi terkini: invoice baru, persetujuan, stok menipis. • Shortcut menu utama: Inventaris, Invoice, Laporan Kebutuhan Survey, Surat Jalan. • Ringkasan proyek aktif yang sedang berjalan.
	REQ-F-03 (Manajemen Inventaris)	Pengguna dapat mengelola data inventaris secara lengkap: • Menambah, mengedit, dan menonaktifkan data master aset (nama, kode SKU, kategori, satuan, spesifikasi teknis, harga satuan, stok minimum). • Mengelola master data kategori aset (hardware, kabel fiber optik, komponen jaringan, alat instalasi, dll.). • Mengelola master data supplier beserta informasi kontak dan riwayat pengadaan. • Mencatat transaksi stok masuk: penerimaan aset dari supplier dengan detail tanggal, jumlah, nomor PO, dan keterangan. • Mencatat transaksi stok keluar: penggunaan aset untuk proyek yang terhubung dengan surat jalan terkait. • Mendukung scan barcode/QR code untuk identifikasi aset secara cepat di lapangan maupun gudang. • Menampilkan riwayat mutasi stok per item secara lengkap dan terfilter. • Sistem memberikan peringatan otomatis (notifikasi in-app dan push) apabila stok item berada di bawah batas minimum.

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
	REQ-F-04 (Manajemen Invoice)	Pengguna dapat membuat dan mengelola invoice digital: • Finance dapat membuat invoice baru berdasarkan Laporan Kebutuhan yang dikirim oleh Project Manager, atau secara manual. • Form pembuatan invoice: pilih klien, pilih proyek/penugasan terkait, tambahkan item dari laporan kebutuhan atau pilih manual dari inventaris, isi kuantitas, harga satuan, keterangan PPN, dan catatan. • Sistem melakukan kalkulasi otomatis: subtotal per item, total sebelum PPN, nilai PPN, dan grand total. • Invoice dapat disimpan sebagai Draft dan diedit sebelum diajukan. • Finance mengajukan invoice kepada Super Admin melalui tombol 'Ajukan Persetujuan'; status berubah menjadi 'Menunggu Persetujuan'. • Super Admin menerima notifikasi dan dapat menyetujui atau menolak invoice. Jika ditolak, Super Admin wajib mengisi catatan alasan penolakan. • Jika ditolak, Finance dapat merevisi invoice dan mengajukan kembali. • Invoice yang disetujui otomatis memicu penerbitan Surat Jalan digital. • Finance mengupdate status invoice menjadi 'Lunas' setelah pembayaran diterima dari klien. • Invoice dapat diekspor ke format PDF dan dikirimkan via email kepada Super

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		Admin, Admin, Project Manager, dan klien.
	REQ-F-05 (Manajemen Klien)	Admin dapat mengelola data klien perusahaan: • Admin/Manajer Gudang dapat menambah, mengedit, dan menonaktifkan data klien (nama perusahaan, alamat, telepon, email, NPWP). • Admin/Manajer Gudang dan Project Manager dapat membuat & mengelola data proyek yang terhubung ke klien (nama proyek, jenis pekerjaan, lokasi, tanggal, status). • Super Admin dapat membuat penugasan proyek yang dikirimkan sebagai notifikasi ke Project Manager yang ditunjuk. • Riwayat seluruh invoice dan surat jalan per klien dapat diakses oleh Super Admin, Admin, dan Finance.
	REQ-F-06 (Laporan & Ekspor)	Sistem menghasilkan laporan otomatis yang dapat difilter dan diekspor: • Laporan mutasi stok (masuk/keluar) berdasarkan rentang tanggal, kategori, atau proyek — untuk Admin dan Super Admin. • Laporan daftar invoice berdasarkan status, klien, atau rentang tanggal — untuk Finance, Admin, dan Super Admin. • Laporan penggunaan aset per proyek — untuk Admin dan Super Admin. • Laporan stok saat ini (rekap semua item, jumlah stok, dan nilai) — untuk Admin dan Super Admin.

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		- Laporan proyek dan penugasan Project Manager — untuk Super Admin dan PM yang bersangkutan. - Semua laporan dapat diekspor ke format PDF dan Excel (.xlsx).
	REQ-F-07 (Notifikasi)	Sistem mengirimkan notifikasi push dan in-app: - Ke Finance/Admin Invoicing: laporan kebutuhan baru dikirim oleh Project Manager. - Ke Super Admin: invoice baru diajukan untuk persetujuan oleh Finance. - Ke Finance/Admin Invoicing: invoice disetujui atau ditolak oleh Super Admin (beserta catatan jika ditolak). - Ke Project Manager: penugasan proyek baru dari Super Admin. - Ke Project Manager: surat jalan baru diterbitkan untuk proyeknya. - Ke Admin & Super Admin: stok item di bawah batas minimum.
	REQ-F-08 (Manajemen Pengguna)	Admin dapat mengelola akun pengguna aplikasi: - Super Admin dan Developer dapat membuat, mengedit, mengaktifkan, dan menonaktifkan akun pengguna. - Setiap akun ditetapkan satu dari empat role: Super Admin/Developer, Admin/Manajer Gudang, Finance/Admin Invoicing, atau Project Manager. - Hak akses setiap modul dikontrol secara otomatis berdasarkan role yang ditetapkan.

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		• Super Admin dapat melihat log aktivitas seluruh pengguna (login, aksi kritis, export data).
	REQ-F-09 (Mode Offline)	Aplikasi mendukung mode offline terbatas: • Project Manager dapat mengisi form laporan kebutuhan aset secara offline; data tersinkronisasi otomatis saat koneksi internet tersedia kembali. • Admin/Gudang dapat melihat daftar aset dan melakukan input transaksi stok secara offline dengan sinkronisasi otomatis. • Pengguna dapat melihat data yang telah dimuat sebelumnya (cached) dalam mode offline. • Fitur yang memerlukan koneksi (submit invoice, approval, notifikasi real-time, ekspor PDF) tidak tersedia dalam mode offline.
	REQ-F-10 (Validasi & Error Handling)	• Semua field input wajib divalidasi di sisi client (Flutter) sebelum dikirim ke server. • Pesan error yang jelas dan informatif dalam Bahasa Indonesia ditampilkan kepada pengguna ketika terjadi kegagalan input atau koneksi. • Sistem mencegah duplikasi data: nomor invoice, kode SKU aset, nomor surat jalan. • Setiap transaksi kritis (persetujuan invoice, penerbitan surat jalan, penghapusan data master) meminta konfirmasi dari pengguna sebelum dieksekusi. • Server mengembalikan HTTP status code yang sesuai dan pesan error terstruktur dalam format JSON.

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
5. KEBUTUHAN NON-FUNGSIONAL	5.1 Kinerja (Performance)	. Waktu respon API untuk pengambilan data (dashboard, daftar inventaris, daftar invoice) maksimal 3 detik dalam kondisi jaringan normal. . Proses penyimpanan data transaksi (stok masuk/keluar dan pembuatan invoice) tidak melebihi 5 detik sejak pengguna menekan tombol simpan. . Sistem mampu menangani minimal 1000 data inventaris aktif tanpa penurunan performa yang signifikan pada sisi aplikasi mobile. . Aplikasi dapat berjalan stabil pada perangkat Android minimal versi 10.0 dengan RAM minimal 4GB. . Sistem mendukung pemrosesan beberapa permintaan (request) API secara bersamaan tanpa menyebabkan crash atau force close pada aplikasi.
	5.2 Keamanan (Security)	. Sistem menggunakan autentikasi berbasis JWT (JSON Web Token) untuk setiap akses API. . Password pengguna disimpan dalam bentuk hash terenkripsi menggunakan algoritma bcrypt pada sisi server. . Setiap endpoint API dilindungi oleh mekanisme role-based access control (RBAC) sesuai dengan hak akses (Super Admin, Admin, Finance, Project Manager). . Seluruh komunikasi data antara aplikasi mobile dan server menggunakan protokol HTTPS (SSL/TLS). . Sistem mencatat log aktivitas penting seperti login, perubahan data inventaris, dan

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		pembuatan atau perubahan invoice. . Sesi pengguna otomatis berakhir setelah 8 jam tidak aktif (configurable sesuai kebijakan perusahaan).
	5.3 Usability	. Antarmuka aplikasi dirancang menggunakan prinsip Material Design agar konsisten dengan standar aplikasi Android. . Menu utama dapat diakses maksimal dalam 3 langkah navigasi dari halaman Dashboard. . Sistem menyediakan validasi input secara real-time untuk mencegah kesalahan pengisian data (misalnya jumlah stok tidak boleh negatif). . Setiap proses transaksi (stok masuk/keluar, pembuatan invoice) menampilkan notifikasi visual berupa pesan sukses atau gagal. . Desain tampilan dibuat sederhana, dengan ikon dan label yang jelas agar mudah digunakan oleh pengguna non-teknis. . Fitur pencarian dan filter tersedia pada halaman inventaris dan invoice untuk mempermudah pencarian data.
	5.4 Ketersediaan (Availability)	. Sistem memiliki tingkat ketersediaan layanan minimal 99% dalam periode operasional bulanan. . Database sistem dilakukan backup otomatis minimal satu kali setiap hari. . Jika terjadi gangguan server, sistem dapat dipulihkan dalam waktu maksimal 24 jam. . Aplikasi mendukung mode offline terbatas untuk pencatatan sementara transaksi

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		yang akan disinkronkan ketika koneksi internet tersedia kembali. . Sistem menampilkan pesan kesalahan (error message) yang informatif apabila terjadi gangguan koneksi atau kegagalan proses.
6. PEMODELAN SISTEM	6.1 Use Case Diagram	Aktor utama dan use case yang terhubung: Super Admin (Akses Penuh): . Manajemen Master Data: Mengelola data barang, kategori, dan supplier secara penuh. . Manajemen Pengguna: Mengatur akun pengguna lain dan menetapkan hak akses (Role Management). . Monitoring Laporan & Dashboard: Melihat ringkasan kondisi gudang, status invoice, dan laporan proyek secara real-time. . Otorisasi Transaksi: Melakukan persetujuan (approval) terhadap invoice atau laporan yang masuk dari Admin atau PM. Admin (Manajer Gudang): . Manajemen Inventaris: Mengelola data stok barang, termasuk mencatat stok masuk dan keluar. . Manajemen Klien: Mengelola basis data klien perusahaan yang akan dihubungkan dengan invoice. . Pelaporan Gudang: Menghasilkan laporan stok dan mutasi barang untuk dilaporkan kepada Super Admin. Project Manager (PM): . Manajemen Survey & Proyek: Membuat laporan hasil survey lapangan dan

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		mengelola detail proyek yang sedang berjalan. . Manajemen Penugasan: Menerima dan mengelola tugas yang diberikan oleh Super Admin terkait proyek tertentu. . Akses Data Invoice & Barang: Melihat status invoice terkait proyeknya dan mengecek ketersediaan data barang di gudang untuk kebutuhan proyek. . Laporan Kebutuhan: Membuat permintaan atau laporan kebutuhan barang berdasarkan hasil survey di lapangan.
	6.2 Activity Diagram	Alur Pembuatan Invoice: Login → Dashboard → Menu Invoice → Buat Invoice Baru → Pilih Klien → Tambah Item dari Inventaris → Isi Qty & Harga → Sistem Auto-Kalkulasi Total → Preview Invoice → Simpan sebagai Draft → Kirim via Email/WhatsApp → Status Berubah menjadi "Terkirim" → Klien Membayar → Admin Update Status → "Lunas". Alur Catat Stok Masuk: Login → Dashboard → Menu Inventaris → Tambah Transaksi Masuk → Scan Barcode / Pilih Item Manual → Isi Jumlah & Keterangan → Konfirmasi → Sistem Update Stok → Notifikasi Berhasil.
	6.3 Sequence Diagram	Skenario: Buat Invoice Baru 1. [User/Finance] → tap "Buat Invoice" 2. [UI/View] → kirim request GET /clients & GET /inventory ke [Controller] 3. [Controller] → panggil [API Service] dengan JWT Token

Bab / Bagian	Sub-Bagian	Aplikasi Mobile Warehouse Inventory & Invoicing
		4. [API Service] → query [Database] → kembalikan JSON list klien & barang 5. [Controller] → teruskan data ke [UI/View] → tampilkan form Invoice 6. [User] → isi form & tap "Simpan" 7. [UI/View] → kirim POST /invoices dengan payload data invoice 8. [API Service] → validasi data → simpan ke [Database] → kembalikan response 201 Created 9. [UI/View] → tampilkan konfirmasi sukses & opsi Kirim Invoice

- Fase Konstruksi: merupakan pembangunan logika bisnis di sisi server (backend) menggunakan Laravel. Setiap endpoint REST API dirancang dengan prinsip separation of concerns, yaitu setiap Controller menangani satu domain bisnis tertentu (autentikasi, inventaris, proyek, invoice, dan notifikasi), sementara logika pemrosesan data didelegasikan ke Model melalui Eloquent ORM.

CUPLIKAN KODE API.PHP

```

routes > api.php > ...
Rico Satyadharma P.S, last week | 2 authors (maul-j099 and one other)
1 <?php
2
3 use App\Http\Controllers\Api\Auth\LoginController;
4 use App\Http\Controllers\Api\Auth\LogoutController;
5 use App\Http\Controllers\Api\Auth\MeController;
6 use App\Http\Controllers\Api\DashboardController;
7 use App\Http\Controllers\Api\InvoiceController;
8 use App\Http\Controllers\Api\ItemController;
9 use App\Http\Controllers\Api\ProjectController;
10 use App\Http\Controllers\Api\SurveyController;
11 use App\Http\Controllers\Api\StockOpnameController;
12 use App\Http\Controllers\Api\UserController;
13 use Illuminate\Support\Facades\Route;
14
15 Route::post('/login', LoginController::class);
16
17 Route::middleware('auth:sanctum')->group(function () { void {
18     Route::post('/logout', LogoutController::class);
19     Route::get('/me', MeController::class);
20
21     Route::get('/dashboard', DashboardController::class)
22         // Permission: dashboard.view
23         // Allowed roles: super_admin, admin, finance, project_manager
24         ->middleware('permission:dashboard.view');
25

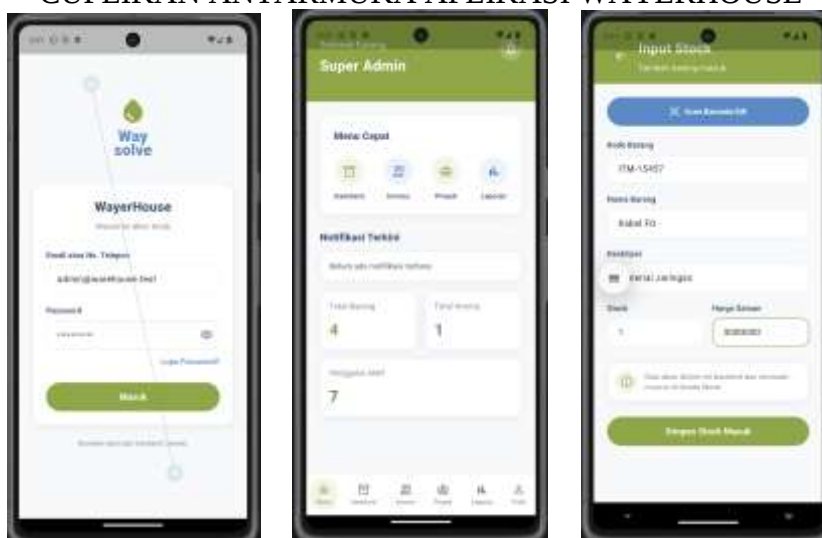
```

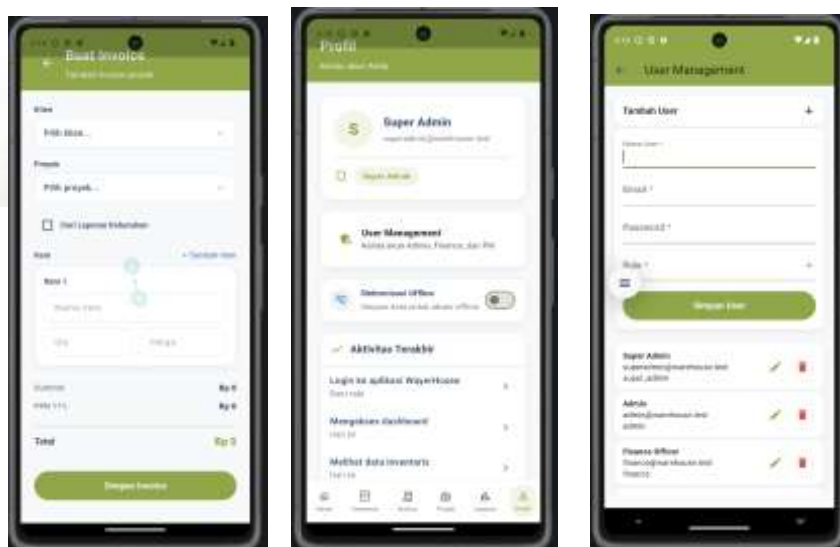
- Fase Integrasi: meliputi pembangunan antarmuka pengguna (frontend) menggunakan Flutter, integrasi seluruh komponen frontend dengan API backend, serta pengujian menyeluruh untuk memastikan setiap interaksi pengguna menghasilkan respons yang tepat dari server.

CUPLIKAN KODE PROVIDER

```
lib > features > dashboard > dashboard_provider.dart
7  class DashboardStats {
8    final int itemsTotal;
9    final int invoicesTotal;
10   final int activeUsersTotal;
11   final String role;
12   final List<String> widgets;
13
14   const DashboardStats({
15     required this.itemsTotal,
16     required this.invoicesTotal,
17     required this.activeUsersTotal,
18     required this.role,
19     required this.widgets,
20   });
21
22   factory DashboardStats.fromJson(Map<String, dynamic> json) {
23     final data = json['data'] as Map<String, dynamic>;
24     final stats = data['stats'] as Map<String, dynamic>;
25     return DashboardStats(
26       itemsTotal: stats['items_total'] ?? 0,
27       invoicesTotal: stats['invoices_total'] ?? 0,
28       activeUsersTotal: stats['active_users_total'] ?? 0,
29       role: data['role'] ?? '',
30       widgets: List<String>.from(data['widgets'] ?? []),
31     );
32   }
33 }
```

CUPLIKAN ANTARMUKA APLIKASI WAYERHOUSE





2.2 Arsitektur Sistem

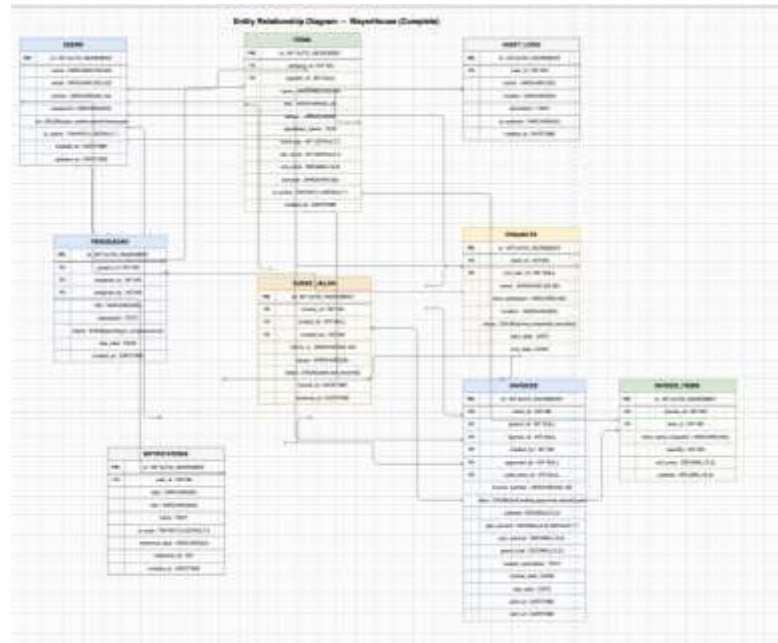
Aplikasi WayerHouse dirancang dengan arsitektur tiga lapis (three-tier architecture) yang terdiri atas client layer, network layer, dan backend layer yang terhubung dengan basis data. Pada client layer, pengguna berinteraksi dengan antarmuka Flutter yang status datanya dikendalikan melalui state management, sementara data sesi disimpan secara lokal (local data). Permintaan data dikirimkan ke server melalui REST API transmission menggunakan HTTP client asinkron, dengan payload dalam format JSON yang menyertakan bearer token pada setiap permintaan ke rute yang terproteksi. Pada backend layer, permintaan yang masuk diproses oleh router dan disaring melalui middleware untuk validasi token sebelum diteruskan ke Controller terkait, yang selanjutnya berinteraksi dengan basis data melalui Eloquent ORM. Hasil pemrosesan data kemudian dikembalikan dalam bentuk respons JSON yang ditampilkan kembali pada aplikasi mobile.

2.3 Perancangan Basis Data

Perancangan basis data dimodelkan menggunakan Entity Relationship Diagram (ERD) sebelum proses penulisan kode dimulai. Entitas utama yang dirancang meliputi Users (data seluruh pengguna sistem, yaitu admin, super admin, finance, dan project manager), Items (data barang/inventaris beserta kategori dan pemasok), Projects (data proyek yang dijalankan perusahaan), Assignment/Penugasan (tugas yang diberikan kepada pegawai), Surat Jalan (dokumen pengiriman barang untuk proyek), Invoices dan Invoice_Items (data faktur beserta rincian item dan status pembayaran), Audit Logs (pencatatan aktivitas pengguna pada sistem), serta Notifications (notifikasi yang dikirim kepada setiap pengguna).

Relasi antarentitas dirancang agar setiap proyek dapat memiliki banyak penugasan, surat jalan, dan invoice; setiap invoice dapat memiliki banyak invoice item yang masing-masing terhubung ke data item; serta setiap aktivitas pengguna tercatat pada audit log. Integritas relasi dijaga melalui kunci primer dan kunci asing pada setiap tabel, sehingga seluruh aktivitas pengguna, pengelolaan barang, proyek, dan dokumen terkait dapat ditelusuri secara lengkap (Inmon, 2020).

Jurnal Ilmiah Sain dan Teknologi
ENTITY RELATIONSHIP DIAGRAM



2.4 Alat dan Teknologi

Pengembangan aplikasi WayerHouse memanfaatkan beberapa perangkat bantu dan teknologi yang dipilih berdasarkan kebutuhan proyek serta standar industri pengembangan aplikasi mobile dan backend API modern, sebagaimana dirangkum pada Tabel II.

TABEL II
ALAT DAN TEKNOLOGI YANG DIGUNAKAN

Alat / Teknologi	Fungsi dalam Pengembangan
Visual Studio Code	Lingkungan utama penulisan kode backend (Laravel/PHP) dan frontend (Flutter/Dart).
Android Studio	Pengujian antarmuka melalui emulator dan perangkat fisik Android.
Laravel 12 (PHP 8.2+)	Framework backend untuk REST API, migration basis data, Eloquent ORM, task scheduling, dan autentikasi berbasis token (Sanctum).
Flutter (Dart)	Framework cross-platform untuk membangun antarmuka mobile dengan satu basis kode.
PostgreSQL (Supabase)	Basis data relasional yang dihosting pada platform Backend-as-a-Service (Supabase).
Postman	Pengujian REST API, verifikasi format respons JSON, serta validasi autentikasi dan otorisasi.
Git & GitHub	Kontrol versi kode sumber dan kolaborasi pengembangan melalui mekanisme branching.
Figma	Perancangan wireframe dan prototipe high-fidelity antarmuka sebelum implementasi kode.
draw.io / dbdiagram.io	Perancangan Use Case Diagram, Activity Diagram, dan Entity Relationship Diagram (ERD).

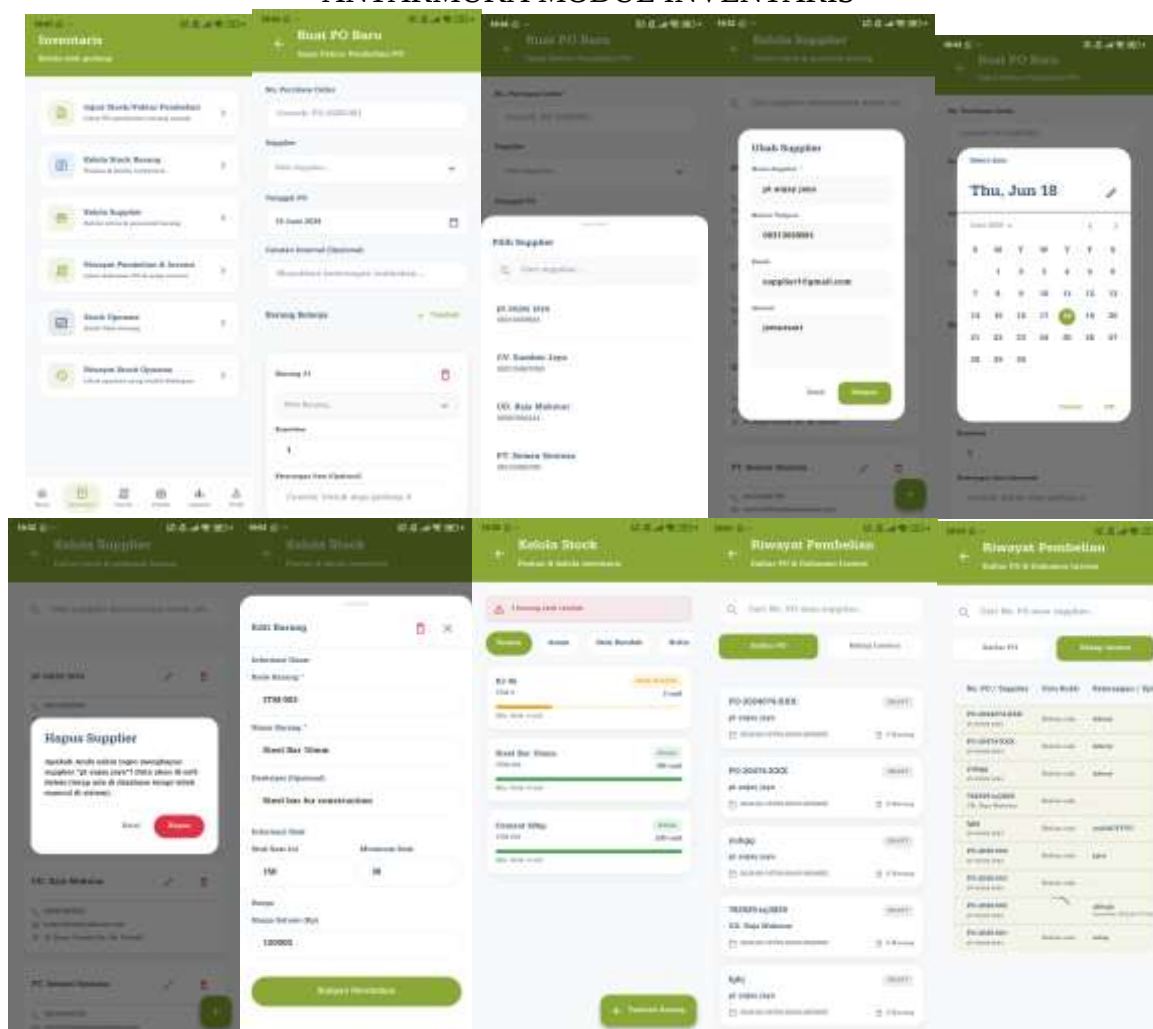
3. HASIL DAN PEMBAHASAN

Pelaksanaan penelitian menghasilkan sebuah aplikasi mobile terintegrasi bernama WayerHouse, yang mencakup modul inventaris, manajemen proyek, pemfakturan, tata kelola layanan teknologi informasi, serta konsolidasi data lintas modul yang merepresentasikan prinsip data warehouse. Hasil implementasi dari setiap modul dijabarkan pada subbagian berikut.

3.1 Implementasi Modul Inventaris

Modul inventaris dikembangkan untuk menjawab permasalahan proses keluar-masuk produk perusahaan yang sebelumnya tidak melalui satu pintu yang jelas. Fitur input stok diintegrasikan sebagai satu pintu barang masuk yang disesuaikan dengan faktur pembelian, dilengkapi dengan fungsi CRUD (create, read, update, delete) untuk data barang, serta fitur stock opname beserta riwayatnya sebagai mekanisme penyesuaian antara data sistem dan kondisi fisik barang pada periode tertentu. Dengan demikian, pergerakan dan deskripsi setiap produk perusahaan dapat dipantau secara real-time dan terintegrasi.

ANTARMUKA MODUL INVENTARIS

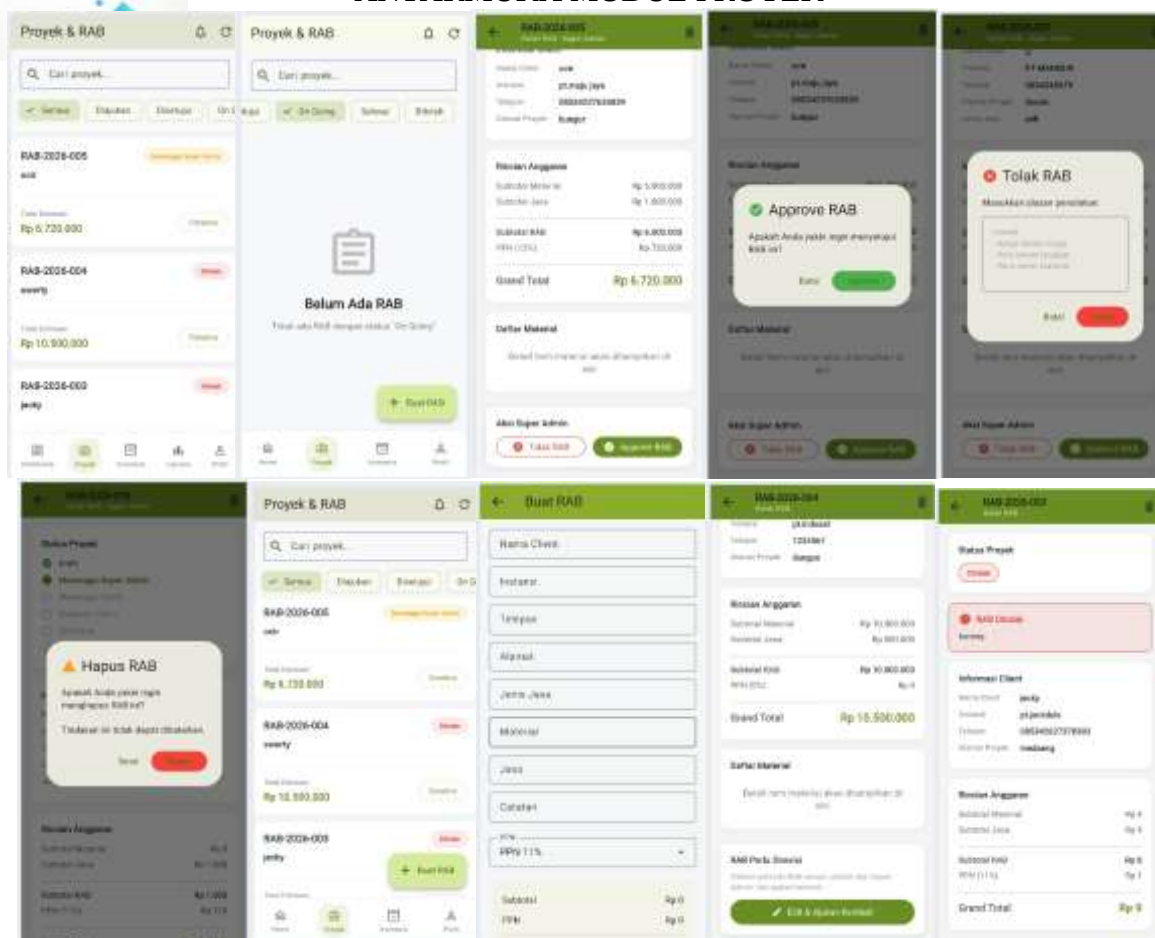


3.2 Implementasi Modul Manajemen Proyek

Modul manajemen proyek dikembangkan untuk mengatasi proses pembuatan proyek yang sebelumnya dilakukan secara manual oleh admin setelah survei lapangan. Alur proyek pada WayerHouse diawali dengan survei oleh pegawai lapangan, dilanjutkan dengan pengajuan Rencana Anggaran Biaya (RAB) (McConnell, 2019), hingga proses persetujuan proyek yang

juga memanfaatkan data dari modul inventaris untuk memastikan ketersediaan barang yang dibutuhkan dalam pengerjaan proyek.

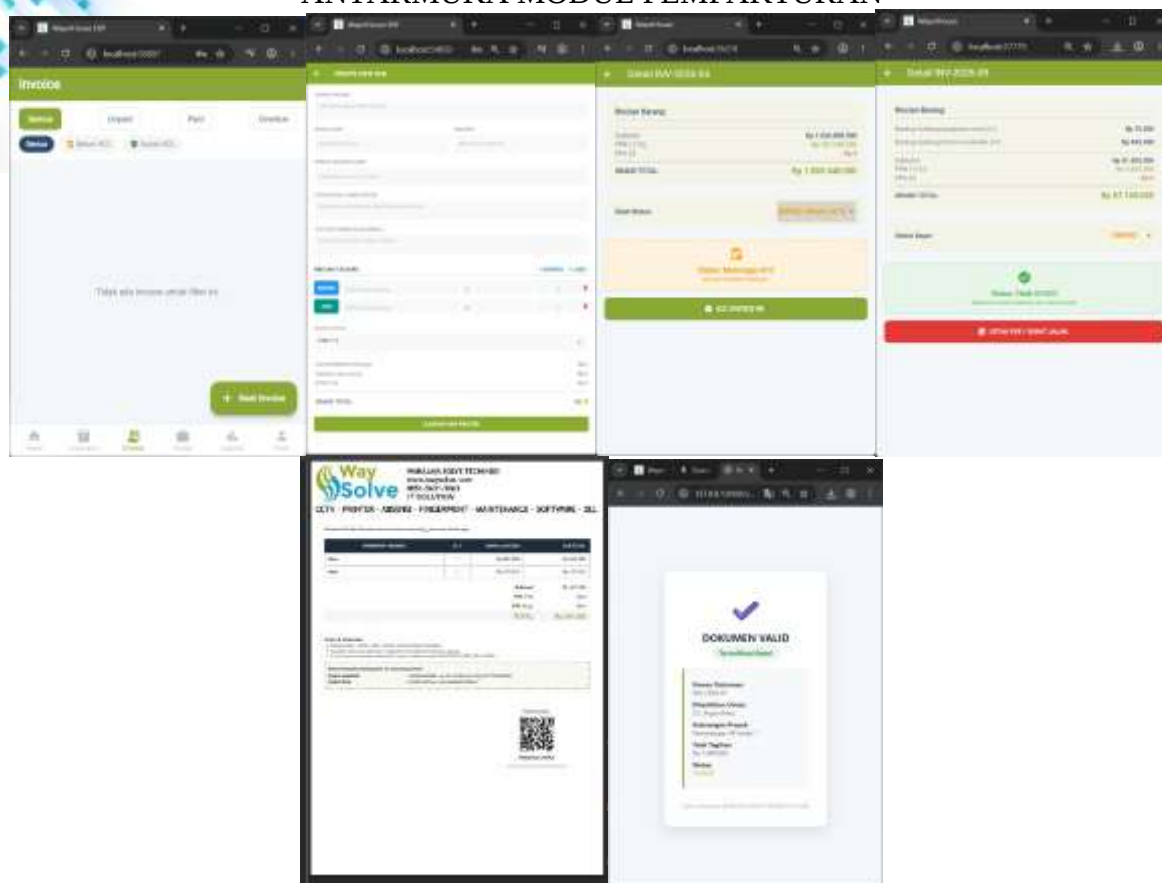
ANTARMUKA MODUL PROYEK



3.3 Implementasi Modul Pemfakturan (Invoicing)

Modul pemfakturan dirancang sebagai platform komunikasi dua arah (two-way communication) yang terhubung dengan modul proyek/RAB dan modul inventaris, mengingat proses pendataan faktor sebelumnya dilakukan secara manual. Setiap faktur memiliki status yang dapat dipantau secara berjenjang, yaitu draft, waiting (menunggu persetujuan), approved (disetujui), rejected (ditolak), dan paid (lunas). Pada halaman pemfakturan, pegawai dapat membuat faktur yang secara otomatis menghitung subtotal beserta Pajak Pertambahan Nilai (PPN), serta menyaring status faktur untuk mempermudah pemantauan progres penagihan.

Jurnal Ilmiah Sain dan Teknologi
ANTARMUKA MODUL PEMFAKTURAN



3.4 Implementasi Tata Kelola Layanan Teknologi Informasi

Dari aspek tata kelola sistem informasi, WayerHouse dirancang untuk menyelaraskan keandalan infrastruktur perusahaan dengan kepatuhan administratif dan keamanan data internal. Sistem menerapkan autentikasi berbasis token yang membatasi akses data sesuai peran pengguna (Admin, Finance Officer, Project Manager, dan Super Admin), dilengkapi dengan validasi data secara real-time pada form input untuk mencegah duplikasi maupun data kosong pada field wajib. Seluruh aktivitas pengguna, termasuk login, perubahan data, serta pembuatan proyek atau invoice, tercatat pada audit log untuk mendukung pengawasan dan ketertelusuran (traceability) aktivitas sistem.

The screenshot displays two slides from a presentation, each showing an inventory and transaction report for a warehouse. The slides are titled 'LAPORAN INVENTARIS & TRANSAKSI GUDANG'.

Slide 1 (Left): The report is for 'Majalah' (Magazines). It includes a sidebar with navigation options: Home, Menu, Tentang, and Kontak. The main content area shows a table of transactions with columns: Tanggal (Date), Tipe (Type), Barang (Item), Qty (Quantity), and Keterangan (Description). The table lists several transactions, including 'Barang Gudang awal es' and 'Barang Gudang akhir es'.

Slide 2 (Right): The report is for 'Koran' (Newspapers). It also includes a sidebar with navigation options: Home, Menu, Tentang, and Kontak. The main content area shows a table of transactions with columns: Tanggal (Date), Tipe (Type), Barang (Item), Qty (Quantity), and Keterangan (Description). The table lists several transactions, including 'Barang Gudang awal es' and 'Barang Gudang akhir es'.

3.5 Implementasi Konsep Data Warehouse

Pengembangan WayerHouse turut mengonsolidasikan data dari berbagai modul internal, meliputi users, items, projects, penugasan, invoices, surat jalan, dan notifikasi, ke dalam struktur basis data relasional yang saling terhubung sesuai prinsip data warehouse (Inmon, 2020). Data master pengguna, barang, dan proyek menjadi fondasi single source of truth (Bizarro & Sampaio, 2020) yang dapat diakses secara real-time oleh admin, finance officer, maupun project manager tanpa perlu mengakses beberapa sistem terpisah. Modul invoice menyimpan data finansial historis berupa subtotal, PPN, total, dan status pembayaran, sementara audit log menyediakan jejak historis aktivitas pengguna. Kontrol akses berbasis peran berfungsi sebagai mekanisme data governance, dan seluruh akses data dipusatkan melalui satu lapisan REST API (API centralization) sehingga konsistensi data tetap terjaga.

3.6 Penqujian Sistem

Pengujian sistem dilakukan melalui dua pendekatan, yaitu pengujian REST API menggunakan Postman dan pengujian fungsional pada perangkat Android fisik. Pengujian melalui Postman digunakan untuk memverifikasi setiap endpoint, termasuk format respons JSON serta mekanisme autentikasi dan otorisasi pada rute yang terproteksi token, sebagaimana ditunjukkan pada pengujian endpoint pengambilan data pengguna (GET user). Pengujian fungsional pada perangkat fisik dilakukan untuk memastikan antarmuka aplikasi berjalan responsif pada berbagai resolusi layar serta seluruh alur bisnis, mulai dari login, pengelolaan inventaris, pembuatan proyek, hingga penerbitan invoice, berjalan sesuai dengan spesifikasi yang telah dirancang tanpa ditemukan kesalahan fungsional yang signifikan.

TABEL III
BLACK BOX TESTING

ID	Modul/Fitur	Skenario Pengujian	Masukan Data	Hasil yang Diharapkan	Hasil Aktual	Status
BBAUT H-01	Autentikasi	Login dengan email dan kata sandi yang valid	Email terdaftar, password benar	Sistem memvalidasi kredensial dan mengarahkan pengguna ke Dashboard sesuai role	Login berhasil, redirect ke Dashboard sesuai role	PASS
BBAUT H-02	Autentikasi	Login dengan kata sandi yang salah	Email valid, password salah	Sistem menolak proses login dan menampilkan pesan kesalahan kredensial	Pesan error ditampilkan, akses login ditolak	PASS
BBAUT H-03	Autentikasi	Logout pengguna dari aplikasi	Menekan tombol logout	Sesi/token pengguna dihapus dan diarahkan kembali ke halaman Login	Sesi berhasil dihapus, kembali ke halaman Login	PASS
BBUSR-01	User Management	Super Admin menambahkan akun pengguna baru	Nama, email unik, dan role pengguna	Akun baru tersimpan ke database dan tampil pada daftar pengguna	Akun baru berhasil ditambahkan dan tampil di daftar	PASS
BBUSR-02	User Management	Menambahkan akun dengan email yang sudah terdaftar	Email yang sudah dipakai user lain	Sistem menolak input dan menampilkan pesan validasi duplikasi email	Validasi duplikasi berhasil, input ditolak	PASS

ID	Modul/Fitur	Skenario Pengujian	Masukan Data	Hasil yang Diharapkan	Hasil Aktual	Status
BBUSR-03	User Management	Mengirim form tanpa mengisi field wajib	Field nama/email dikosongkan	Sistem menampilkan pesan validasi bahwa field wajib harus diisi	Pesan validasi tampil, form tidak terkirim	PASS
BBUSR-04	User Management	Mengakses menu User Management dengan role selain Super Admin	Login sebagai Admin/Finance/Project Manager	Menu User Management tidak ditampilkan atau akses ditolak sesuai RBAC	Menu tersembunyi/akses ditolak sesuai role	PASS
BBINV-01	Inventaris	Menambahkan data barang baru ke sistem	Nama barang, kategori, supplier, harga	Data barang baru tersimpan dan tampil pada daftar inventaris	Barang baru berhasil tersimpan dan tampil	PASS
BBINV-02	Inventaris	Input stok masuk sesuai faktur pembelian	Jumlah barang dan nomor faktur	Jumlah stok bertambah sesuai input dan riwayat transaksi tercatat	Stok terupdate, riwayat stok masuk tersimpan	PASS
BBINV-03	Inventaris	Melakukan stock opname penyesuaian data fisik	Jumlah fisik data dibanding sistem	Sistem menyesuaikan selisih stok dan mencatat riwayat stock opname	Penyesuaian stok berhasil, riwayat tercatat	PASS
BBPRJ-01	Manajemen Proyek	Input data survei lapangan dan	Data lokasi/klien, daftar item, estimasi biaya	Data survei dan RAB tersimpan dengan status	Survei dan RAB berhasil diajukan	PASS

ID	Modul/Fitur	Skenario Pengujian	Masukan Data	Hasil yang Diharapkan	Hasil Aktual	Status
		pengajuan RAB		menunggu persetujuan		
BBPRJ-02	Manajemen Proyek	Admin/Supervisor Admin menyetujui pengajuan proyek	Memilih Setujui pada RAB yang diajukan	Status proyek berubah menjadi disetujui dan proyek menjadi aktif	Status berubah, proyek aktif untuk dikerjakan	PASS
BBINVC-01	Invoicing	Membuat invoice baru berdasarkan data proyek	Daftar item, kuantitas, dan harga satuan	Subtotal dan PPN terhitung otomatis, invoice tersimpan berstatus draft	Kalkulasi PPN otomatis benar, status draft	PASS
BBINVC-02	Invoicing	Mengajukan invoice untuk proses persetujuan	Menekan tombol Ajukan pada invoice draft	Status invoice berubah dari draft menjadi waiting (menunggu persetujuan)	Status berhasil berubah menjadi waiting	PASS
BBINVC-03	Invoicing	Admin menyetujui atau menolak invoice yang diajukan	Memilih Setujui atau Tolak beserta catatan	Status berubah menjadi approved (dengan surat jalan terbit) atau rejected dengan catatan tersimpan	Status approved/rejected sesuai aksi, catatan tersimpan	PASS
BBAUD IT-01	Audit Log	Sistem mencatat aktivitas perubahan	Melakukan perubahan pada data invoice/user	Audit log mencatat user, jenis aksi, modul	Log aktivitas tercatat lengkap dan dapat ditelusuri	PASS

ID	Modul/Fitur	Skenario Pengujian	Masukan Data	Hasil yang Diharapkan	Hasil Aktual	Status
		data oleh pengguna		terkait, dan waktu aktivitas		

3.7 Pembahasan

Hasil implementasi menunjukkan bahwa integrasi modul inventaris, manajemen proyek, dan pemfakturan ke dalam satu platform mobile berhasil menjawab permasalahan utama yang dihadapi CV Warajaya Solve Techindo, yaitu ketidaktersediaan platform terpusat dan alur kerja manual yang menghambat efisiensi operasional. Penerapan kontrol akses berbasis peran serta audit log meningkatkan akuntabilitas dan keamanan data, sementara konsolidasi data lintas modul melalui pendekatan data warehouse mempermudah pemantauan status proyek, inventaris, dan keuangan secara terpadu. Pemilihan Flutter sebagai framework frontend juga terbukti efisien dalam menghasilkan antarmuka yang konsisten pada satu basis kode untuk platform Android (Kim & Lee, 2023), sementara Laravel memberikan fondasi backend yang terstruktur melalui penerapan prinsip separation of concerns pada setiap Controller dan Model.

4. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa aplikasi mobile WayerHouse berhasil dikembangkan menggunakan framework Flutter, REST API berbasis Laravel 12, dan basis data PostgreSQL melalui pendekatan Agile Iteratif yang terbagi ke dalam fase fondasi, konstruksi, dan integrasi. Aplikasi ini berhasil mengintegrasikan modul inventaris, manajemen proyek, dan pemfakturan ke dalam satu platform terpusat yang didukung oleh kontrol akses berbasis peran serta pencatatan audit log, sehingga mampu menggantikan proses pendataan manual yang sebelumnya berjalan tanpa alur yang terstruktur pada CV Warajaya Solve Techindo. Seluruh fungsionalitas utama sistem yang diuji melalui Postman dan pengujian fungsional pada perangkat fisik telah berjalan sesuai dengan spesifikasi yang dirancang.

Penelitian ini masih memiliki ruang pengembangan lebih lanjut. Penambahan fitur laporan laba rugi yang menghitung selisih antara barang masuk dengan faktur penagihan keluar disarankan agar perusahaan dapat memantau kesehatan keuangan secara lebih komprehensif. Selain itu, optimasi performa dan responsivitas tampilan pada berbagai ukuran perangkat juga perlu menjadi perhatian pada pengembangan selanjutnya agar pengalaman pengguna semakin optimal.

DAFTAR PUSTAKA

- Acuna, S. T., Gomez, M. N., & Juristo, N. (2018). How do Agile methods affect software product quality? *IEEE Software*, 35(5), 56–64.
- Al-Sarayreh, K. T., & Al-Sarayreh, M. (2021). Software requirements specification (SRS) tailoring framework for software house environment. *Journal of Software Engineering and Applications*, 14(3), 85–103.
- Bizarro, P. A., & Sampaio, J. R. (2020). Single Source of Truth (SSOT) architecture for enterprise data integration. *International Journal of Computer Science and Information Security*, 18(4), 112–120.
- Google Developers. (2024). Material Design 3 guidelines for Android and cross-platform application interface. Google Open Source.
- Inmon, W. H. (2020). *Building the Data Warehouse* (5th ed.). John Wiley & Sons.

- Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia. (2020). Buku panduan merdeka belajar - Kampus merdeka. Direktorat Jenderal Pendidikan Tinggi Kemendikbud RI.
- Kim, Y., & Lee, S. (2023). A comparative evaluation of native and cross-platform mobile application development tools: Flutter vs Android native. *Applied Sciences*, 13(7), 4125.
- McConnell, S. (2019). *Software estimation: Demystifying the black art*. Microsoft Press.
- Tidwell, J., Brewer, C., & Valencia-Brooks, A. (2020). *Designing interfaces: Patterns for effective interaction design* (3rd ed.). O'Reilly Media.