

# PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI MANAJEMEN DISTRIBUSI LPG BERBASIS WEB DENGAN ARSITEKTUR MULTI-TENANT

## DESIGN AND IMPLEMENTATION OF WEB-BASED LPG DISTRIBUTION MANAGEMENT INFORMATION SYSTEM WITH MULTI-TENANT ARCHITECTURE

Luthfi Alfaridz<sup>1</sup>, Siti Sarah<sup>2</sup>

<sup>123</sup>Program Studi Teknik Informatika, Fakultas Teknik

Universitas Suryakancana, Cianjur, Indonesia

Email: <sup>1</sup>luthfifahmi.alv@gmail.com

### ABSTRAK

Distribusi gas LPG merupakan kegiatan vital dalam rantai pasok energi nasional. Namun, banyak agen LPG yang masih menggunakan pencatatan manual sehingga menimbulkan permasalahan ketidakakuratan data, kesulitan pelacakan pesanan, sampai dengan lambatnya pembuatan laporan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem informasi manajemen distribusi LPG berbasis web dengan arsitektur multi-tenant. Metode pengembangan menggunakan Waterfall dengan perancangan UML. Teknologi yang digunakan meliputi React 18 untuk frontend, NestJS 10 untuk backend, dan PostgreSQL 15 untuk database. Hasil pengujian black-box menunjukkan seluruh 28 test case berfungsi sesuai spesifikasi dengan tingkat keberhasilan 100%. Sistem berhasil mengintegrasikan proses pencatatan pesanan, manajemen stok, pembayaran, dan pelaporan dalam satu platform yang dapat meningkatkan efisiensi operasional bisnis distribusi LPG.

**Kata Kunci:** Sistem Informasi, Distribusi LPG, Multi-Tenant, React, NestJS, PostgreSQL

### ABSTRACT

LPG gas distribution is a vital activity in the national energy supply chain. However, many LPG agents still use manual recording, causing problems of data inaccuracy, difficulty in tracking orders, and slow report generation. This research aims to design and implement a web-based LPG distribution management information system with multi-tenant architecture. The development method uses Waterfall with UML design. The technologies used include React 18 for frontend, NestJS 10 for backend, and PostgreSQL 15 for database. Black-box testing results show all 28 test cases function according to specifications with a 100% success rate. The system successfully integrates order recording, stock management, payment, and reporting processes into one integrated platform that can improve the operational efficiency of the LPG distribution business.

**Keywords:** Information System, LPG Distribution, Multi-Tenant, React, NestJS, PostgreSQL

### 1. PENDAHULUAN

## 1.1 Latar Belakang Masalah

Gas LPG (Liquefied Petroleum Gas) merupakan salah satu sumber energi strategis yang banyak digunakan masyarakat Indonesia untuk keperluan rumah tangga maupun komersial. Berdasarkan data Kementerian Energi dan Sumber Daya Mineral tahun 2023, konsumsi LPG nasional mencapai lebih dari 8,2 juta ton per tahun dengan tren peningkatan rata-rata 4-5% setiap tahunnya [1]. Peningkatan konsumsi ini mendorong kompleksitas dalam pengelolaan distribusi LPG dari produsen hingga ke tangan konsumen akhir.

Dalam rantai distribusi LPG di Indonesia, terdapat beberapa pelaku utama yang saling berkaitan. PT Pertamina sebagai produsen menyalurkan LPG ke SPBE (Stasiun Pengisian Bulk Elpiji), kemudian agen sebagai distributor resmi mengambil pasokan dari SPBE untuk didistribusikan ke pangkalan-pangkalan, dan akhirnya pangkalan menjual langsung ke konsumen akhir [2]. Seorang agen umumnya mengelola puluhan hingga ratusan pangkalan dengan volume transaksi harian yang signifikan, mencakup proses pemesanan, pengiriman, pembayaran, dan pelaporan yang kompleks.

Berdasarkan observasi dan wawancara yang dilakukan pada beberapa agen LPG di wilayah Kabupaten Cianjur pada periode September-Oktober 2025, ditemukan bahwa sebagian besar agen masih menggunakan sistem pencatatan manual berbasis buku tulis, kertas nota dan excel yang tidak terintegrasi satu sama lain. Kondisi ini menimbulkan beberapa permasalahan operasional yang cukup signifikan.

Pertama, **ketidakakuratan data stok** sering terjadi karena pencatatan manual rentan terhadap kesalahan input dan keterlambatan update. Perbedaan antara stok fisik dan stok tercatat dapat mencapai 5-10% setiap bulannya. Kedua, **kesulitan pelacakan pesanan** membuat operator kesulitan memantau status pengiriman secara real-time, sehingga koordinasi dengan driver menjadi tidak efisien. Ketiga, **lambatnya pembuatan laporan** menyebabkan proses rekapitulasi bulanan membutuhkan waktu 3-5 hari kerja yang seharusnya dapat dipersingkat. Keempat, **risiko kehilangan data** cukup tinggi karena catatan berbasis kertas rentan rusak, hilang, atau tidak terbaca.

SIM4LON (Sistem Informasi Manajemen Distribusi Untuk LPG Online) dirancang untuk menjawab permasalahan tersebut dengan mengotomatisasi seluruh alur kerja distribusi, mulai dari manajemen stok, pemesanan oleh pangkalan, pencatatan pembayaran, hingga pembuatan laporan secara otomatis. Dengan implementasi arsitektur multi-tenant, sistem ini memungkinkan satu aplikasi melayani banyak pangkalan dengan isolasi data yang ketat, sehingga setiap pangkalan hanya dapat mengakses data miliknya sendiri [3].

Penggunaan teknologi web modern menjadi pilihan utama karena memungkinkan akses dari berbagai perangkat tanpa perlu instalasi aplikasi khusus. React sebagai library frontend terpopuler menawarkan performa rendering yang efisien [4], sementara NestJS sebagai framework backend berbasis TypeScript menyediakan arsitektur yang modular dan scalable [5]. PostgreSQL dipilih sebagai database karena dukungan ACID compliance yang menjamin integritas data transaksional [6].

Berdasarkan uraian di atas, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem informasi manajemen distribusi LPG berbasis web yang dapat meningkatkan efisiensi operasional, akurasi data, dan kemudahan pembuatan laporan bagi pengguna aplikasi SIM4LON.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana cara merancang sistem informasi manajemen distribusi LPG berbasis web dengan pendekatan UML yang dapat mengakomodasi arsitektur multi-tenant?

2. Bagaimana mengimplementasikan sistem ini menggunakan teknologi web modern yaitu React untuk frontend, NestJS untuk backend, dan PostgreSQL untuk database?
3. Bagaimana hasil pengujian fungsionalitas sistem yang telah dibangun menggunakan metode black-box testing?

### 1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Merancang sistem informasi manajemen distribusi LPG menggunakan pendekatan Unified Modeling Language (UML) yang mencakup Use Case Diagram, Class Diagram, Activity Diagram, Sequence Diagram, dan Entity Relationship Diagram dengan arsitektur multi-tenant.
2. Mengimplementasikan rancangan sistem menjadi aplikasi web yang fungsional menggunakan React 18 sebagai library frontend, NestJS 10 sebagai framework backend, Prisma sebagai ORM, dan PostgreSQL 15 sebagai database.
3. Menguji fungsionalitas sistem menggunakan metode black-box testing untuk memastikan seluruh fitur berjalan sesuai dengan spesifikasi kebutuhan yang telah ditentukan.

### 1.4 Manfaat Penelitian

**Manfaat Teoritis:** Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan ilmu pengetahuan di bidang sistem informasi, khususnya dalam penerapan arsitektur multi-tenant pada aplikasi distribusi berbasis web. Selain itu, penelitian ini dapat menjadi referensi bagi peneliti selanjutnya yang ingin mengembangkan sistem serupa di domain bisnis lainnya.

**Manfaat Praktis:**

1. Bagi agen LPG, sistem ini dapat membantu mengotomatisasi proses operasional harian sehingga meningkatkan efisiensi waktu dan mengurangi kesalahan pencatatan manual.
2. Bagi pangkalan, sistem ini memberikan kemudahan dalam melakukan pemesanan dan memantau status pesanan secara real-time.
3. Bagi peneliti, penelitian ini menambah wawasan dan pengalaman dalam merancang serta mengimplementasikan sistem informasi dengan teknologi modern.

### 1.5 Batasan Masalah

Agar penelitian ini terarah, ditetapkanlah batasan masalah diantaranya yaitu:

1. Sistem dikembangkan untuk kebutuhan agen LPG skala menengah yang mengelola 10 hingga 100 pangkalan.
2. Platform yang dibangun berbasis web dengan desain responsif yang dapat diakses dari browser desktop maupun mobile.
3. Fitur yang dikembangkan mencakup manajemen pengguna, manajemen pangkalan dan supir, manajemen pesanan, manajemen pembayaran, manajemen stok, perencanaan dan penyaluran, serta laporan dan dashboard.
4. Sistem menggunakan autentikasi JWT dengan role admin, pangkalan, dan supir, serta mekanisme single-session login.
5. Pengujian dilakukan dengan metode black-box testing tanpa melibatkan unit testing atau integration testing.

6. Aspek keamanan tingkat lanjut seperti rate limiting, penetration testing, dan audit keamanan tidak termasuk dalam lingkup penelitian ini.
- 

## 2. TINJAUAN PUSTAKA

### 2.1 Sistem Informasi Manajemen

Sistem informasi manajemen (SIM) adalah suatu sistem berbasis komputer yang menyediakan informasi bagi beberapa pengguna yang mempunyai kebutuhan yang serupa dalam suatu organisasi [7]. Menurut O'Brien dan Marakas, sistem informasi adalah kombinasi terorganisasi dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi, sumber data, serta kebijakan dan prosedur yang menyimpan, mengambil, mengubah, dan menyebarkan informasi dalam organisasi [8].

Laudon dan Laudon mendefinisikan sistem informasi sebagai sekumpulan komponen yang saling terhubung untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian dalam organisasi [9]. Komponen utama sistem informasi meliputi: (1) hardware, (2) software, (3) data, (4) prosedur, dan (5) manusia.

Dalam konteks distribusi LPG, sistem informasi manajemen berperan penting untuk mengelola data pesanan, stok, pembayaran, dan pelaporan secara terintegrasi. Dengan adanya SIM, proses pengambilan keputusan dapat dilakukan lebih cepat dan akurat berdasarkan data real-time.

### 2.2 Distribusi LPG di Indonesia

LPG (Liquefied Petroleum Gas) adalah campuran gas hidrokarbon yang dicairkan dengan tekanan untuk memudahkan penyimpanan, penanganan, dan pengangkutan [2]. Di Indonesia, LPG didistribusikan melalui dua jalur utama: LPG Subsidi (tabung 3 kg untuk rumah tangga berpendapatan rendah) dan LPG Non-Subsidi (tabung 5,5 kg, 12 kg, dan 50 kg untuk komersial).

Rantai distribusi LPG di Indonesia melibatkan beberapa pihak [10]:

1. **PT Pertamina** - Produsen dan pemilik merek LPG
2. **SPBE (Stasiun Pengisian Bulk Elpiji)** - Tempat pengisian ulang tabung LPG
3. **Agen** - Distributor resmi yang ditunjuk Pertamina untuk menyalurkan LPG ke pangkalan
4. **Pangkalan** - Pengecer yang menjual LPG langsung ke konsumen

Setiap agen memiliki kuota penyaluran bulanan yang ditetapkan oleh Pertamina dan wajib melaporkan realisasi penyaluran secara berkala. Proses pelaporan ini membutuhkan pencatatan yang akurat dan sistematis.

### 2.3 Arsitektur Multi-Tenant

Multi-tenancy adalah arsitektur perangkat lunak di mana satu instance software melayani banyak pelanggan (tenant) secara bersamaan dengan tetap menjaga isolasi data antar tenant [3]. Berbeda dengan single-tenant di mana setiap pelanggan memiliki instance terpisah, multi-tenant lebih efisien dalam penggunaan resource dan maintenance karena hanya ada satu codebase yang perlu dikelola.

Terdapat tiga pendekatan utama dalam implementasi multi-tenant pada database [11]:

1. **Separate Database** - Setiap tenant memiliki database terpisah. Isolasi tinggi namun biaya tinggi.

2. **Shared Database, Separate Schema** - Semua tenant berbagi database namun memiliki schema terpisah.
3. **Shared Database, Shared Schema** - Semua tenant berbagi database dan schema, dibedakan dengan identifier (tenant\_id). Paling efisien namun memerlukan implementasi row-level security yang ketat.

Dalam penelitian ini, pendekatan yang digunakan adalah Shared Database, Shared Schema dengan kolom **pangkalan\_id** sebagai tenant identifier. Setiap query database difilter berdasarkan pangkalan\_id user yang sedang login untuk memastikan isolasi data.

## 2.4 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan artefak sistem perangkat lunak [12]. UML dikembangkan oleh Object Management Group (OMG) dan telah menjadi standar de facto dalam pengembangan perangkat lunak berorientasi objek.

Diagram UML yang digunakan dalam penelitian ini meliputi:

Diagram	Fungsi
Use Case Diagram	Menggambarkan fungsionalitas sistem dari perspektif aktor
Class Diagram	Menunjukkan struktur statis kelas dan relasi antar kelas
Activity Diagram	Menggambarkan alur aktivitas proses bisnis
Sequence Diagram	Menunjukkan interaksi antar objek dalam urutan waktu
State Machine Diagram	Menggambarkan perubahan state suatu objek
ERD (Entity Relationship Diagram)	Menunjukkan struktur tabel database dan relasinya

## 2.5 Teknologi Pengembangan Web Modern

### React

React adalah library JavaScript open-source untuk membangun user interface yang dikembangkan oleh Meta (Facebook) [4]. React menggunakan pendekatan component-based architecture di mana UI dipecah menjadi komponen-komponen kecil yang dapat digunakan kembali (reusable). Virtual DOM yang digunakan React memungkinkan rendering yang efisien dengan hanya memperbarui bagian yang berubah.

Keunggulan React antara lain:

- Performa tinggi dengan Virtual DOM
- Komponen reusable dan modular
- Ekosistem library yang luas
- Didukung oleh komunitas besar

### NestJS

NestJS adalah framework Node.js progresif untuk membangun aplikasi server-side yang efisien dan scalable [5]. NestJS menggunakan TypeScript secara default dan mengadopsi arsitektur modular yang terinspirasi dari

Angular. Framework ini mendukung berbagai pattern seperti MVC, microservices, dan REST API.

Fitur utama NestJS meliputi:

- Arsitektur modular dengan dependency injection
- Support TypeScript native
- Decorator-based routing
- Integrasi mudah dengan berbagai database melalui TypeORM atau Prisma

## PostgreSQL

PostgreSQL adalah sistem manajemen database relasional open-source yang powerful dengan dukungan penuh terhadap ACID (Atomicity, Consistency, Isolation, Durability) compliance [6]. PostgreSQL mendukung berbagai tipe data advanced, JSON/JSONB, full-text search, dan extensibility.

Alasan penggunaan PostgreSQL:

- ACID compliance untuk integritas transaksi
- Performa tinggi untuk data relasional
- Support JSON untuk data semi-structured
- Mature dan stabil untuk production

## Prisma ORM

Prisma adalah Object-Relational Mapping (ORM) modern untuk Node.js dan TypeScript yang menyediakan type-safe database access [13]. Prisma menggunakan schema definition language untuk mendeskripsikan model data dan secara otomatis generate TypeScript types.

## 2.6 Metode Pengembangan Waterfall

Model Waterfall adalah model klasik dalam pengembangan perangkat lunak yang bersifat sekuensial dan linear [14]. Setiap fase harus diselesaikan sebelum fase berikutnya dimulai, sehingga model ini cocok untuk proyek dengan kebutuhan yang sudah terdefinisi dengan jelas.

Tahapan dalam model Waterfall:

1. **Requirements Analysis** - Pengumpulan dan analisis kebutuhan
2. **System Design** - Perancangan arsitektur dan desain sistem
3. **Implementation** - Penulisan kode program
4. **Testing** - Pengujian fungsionalitas sistem
5. **Deployment** - Pemasangan sistem ke lingkungan production
6. **Maintenance** - Pemeliharaan dan perbaikan pasca deployment

Model ini dipilih karena kebutuhan sistem sudah teridentifikasi dengan jelas berdasarkan observasi lapangan dan wawancara dengan stakeholder.

## 2.7 Penelitian Terkait

Beberapa penelitian sebelumnya yang relevan dengan topik ini antara lain:

**Tabel 1. Perbandingan Penelitian Terkait**

No	Peneliti (Tahun)	Judul	Teknologi	Hasil
1	Pratama (2022) [15]	Sistem Informasi Penjualan LPG Berbasis Web	PHP, MySQL	Mencatat transaksi penjualan
2	Hidayat (2021) [16]	Aplikasi Distribusi Gas dengan Laravel	Laravel, MySQL	Mempercepat proses distribusi
3	Santoso (2023) [17]	SI Manajemen Agen LPG	CodeIgniter, MySQL	Meningkatkan akurasi stok
4	Wijaya (2022) [18]	Aplikasi Monitoring Stok LPG	Java, PostgreSQL	Monitoring real-time

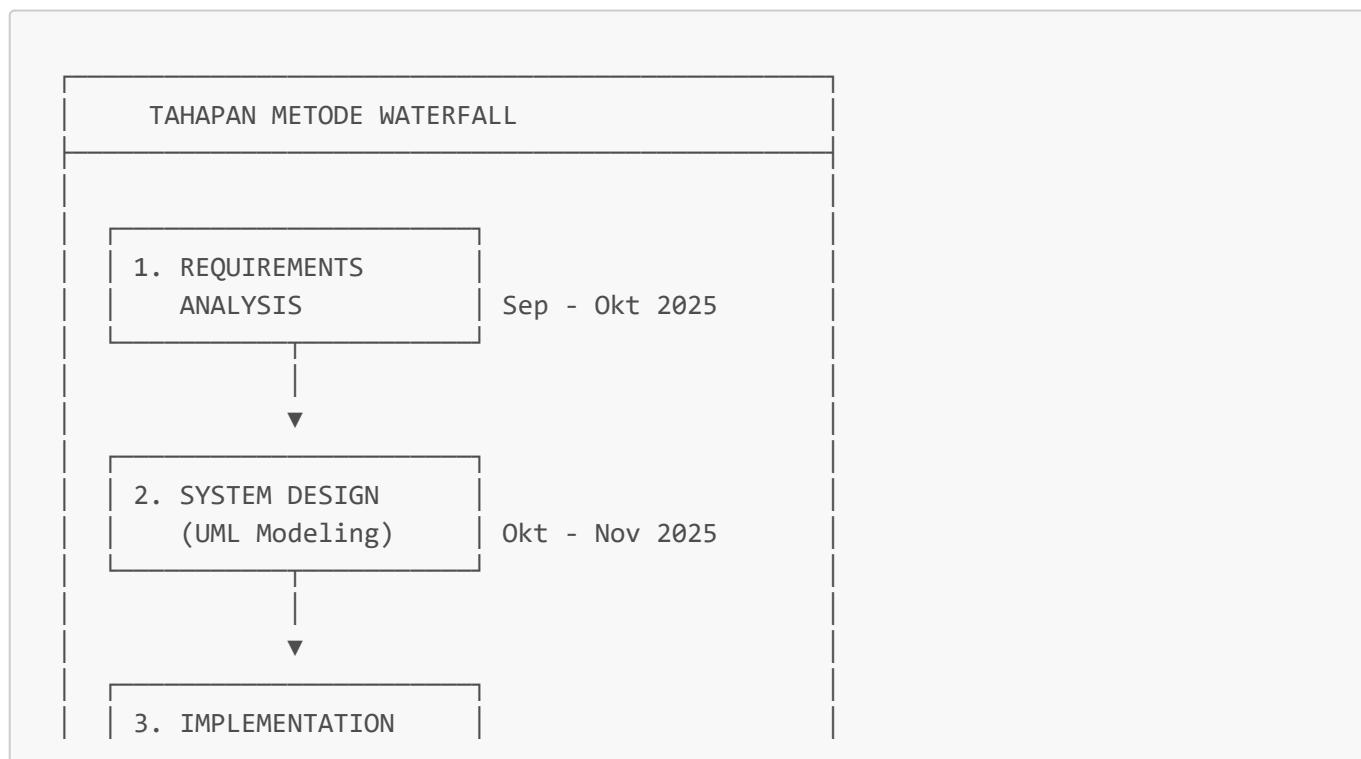
Perbedaan penelitian ini dengan penelitian sebelumnya terletak pada:

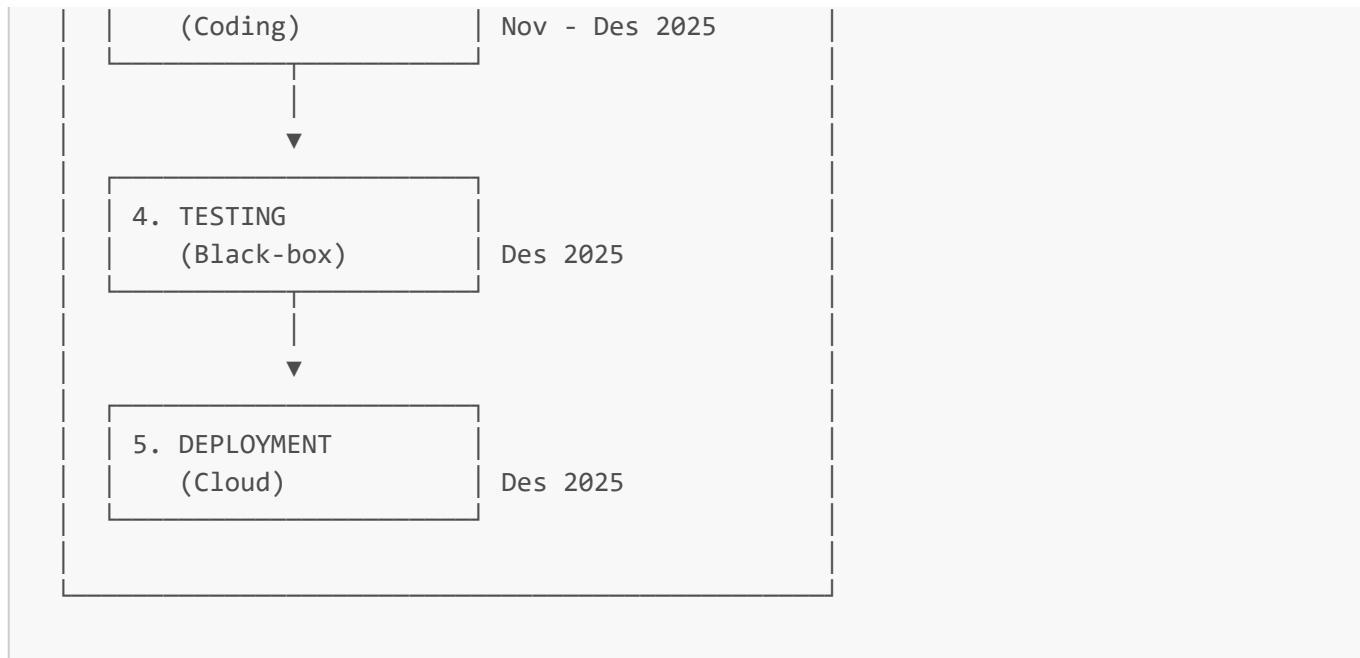
- Teknologi modern** - Menggunakan React + NestJS + Prisma yang lebih modern dan scalable dibanding PHP/CodeIgniter
- Arsitektur multi-tenant** - Mendukung isolasi data antar pangkalan dalam satu aplikasi
- Dokumentasi UML lengkap** - 49 diagram UML untuk dokumentasi yang komprehensif
- Cloud deployment** - Langsung di-deploy ke cloud (Vercel + Railway) untuk akses publik
- Fitur komprehensif** - Mencakup pesanan, pembayaran, stok, perencanaan, penyaluran, dan SAAS pangkalan

### 3. METODOLOGI PENELITIAN

#### 3.1 Metode Pengembangan Sistem

Penelitian ini menggunakan metode pengembangan **Waterfall** yang terdiri dari lima tahapan utama sebagaimana ditunjukkan pada Gambar 1.





**Gambar 1. Tahapan Metode Waterfall**

Penjelasan setiap tahapan:

- 1. Requirements Analysis (September - Oktober 2025)** Pada tahap ini dilakukan pengumpulan data melalui observasi dan wawancara untuk mengidentifikasi kebutuhan sistem. Hasil analisis didokumentasikan dalam bentuk kebutuhan fungsional dan non-fungsional.
- 2. System Design (Oktober - November 2025)** Tahap perancangan sistem menggunakan UML yang mencakup Use Case Diagram, Class Diagram, Activity Diagram, Sequence Diagram, State Machine Diagram, dan Entity Relationship Diagram.
- 3. Implementation (November - Desember 2025)** Tahap implementasi meliputi pengembangan frontend dengan React, backend dengan NestJS, dan database dengan PostgreSQL. Prisma digunakan sebagai ORM untuk interaksi database.
- 4. Testing (Desember 2025)** Pengujian dilakukan dengan metode black-box testing untuk memastikan seluruh fungsionalitas berjalan sesuai spesifikasi.
- 5. Deployment (Desember 2025)** Sistem di-deploy ke cloud dengan Vercel untuk frontend dan Railway untuk backend dan database.

### 3.2 Teknik Pengumpulan Data

Pengumpulan data dilakukan melalui tiga teknik:

**3.2.1 Observasi** Observasi dilakukan dengan mengamati langsung proses bisnis di agen LPG wilayah Cianjur selama dua minggu (14-28 September 2025). Aspek yang diamati meliputi:

- Alur pemesanan dari pangkalan ke agen
- Proses pencatatan dan administrasi
- Mekanisme pengiriman dan distribusi
- Proses pembayaran dan penagihan
- Pembuatan laporan bulanan

### 3.2.2 Wawancara

Wawancara dilakukan dengan beberapa narasumber:

- Pemilik agen LPG (2 orang)
- Operator/admin agen (3 orang)
- Pemilik pangkalan (5 orang)

Pertanyaan wawancara difokuskan pada permasalahan operasional sehari-hari, kebutuhan sistem yang diinginkan, dan ekspektasi terhadap sistem baru.

### 3.2.3 Studi Pustaka

Studi pustaka dilakukan untuk mengkaji teori-teori terkait:

- Sistem informasi manajemen
- Distribusi LPG di Indonesia
- Arsitektur multi-tenant
- Teknologi pengembangan web modern (React, NestJS, PostgreSQL)
- Metode pengembangan Waterfall
- Unified Modeling Language (UML)

## 3.3 Alat dan Bahan Penelitian

**Tabel 2. Perangkat Keras yang Digunakan**

No	Perangkat	Spesifikasi
1	Laptop	ASUS ROG, Intel Core i7, RAM 16GB, SSD 512GB
2	Monitor	External 24" Full HD
3	Koneksi Internet	IndiHome 30 Mbps

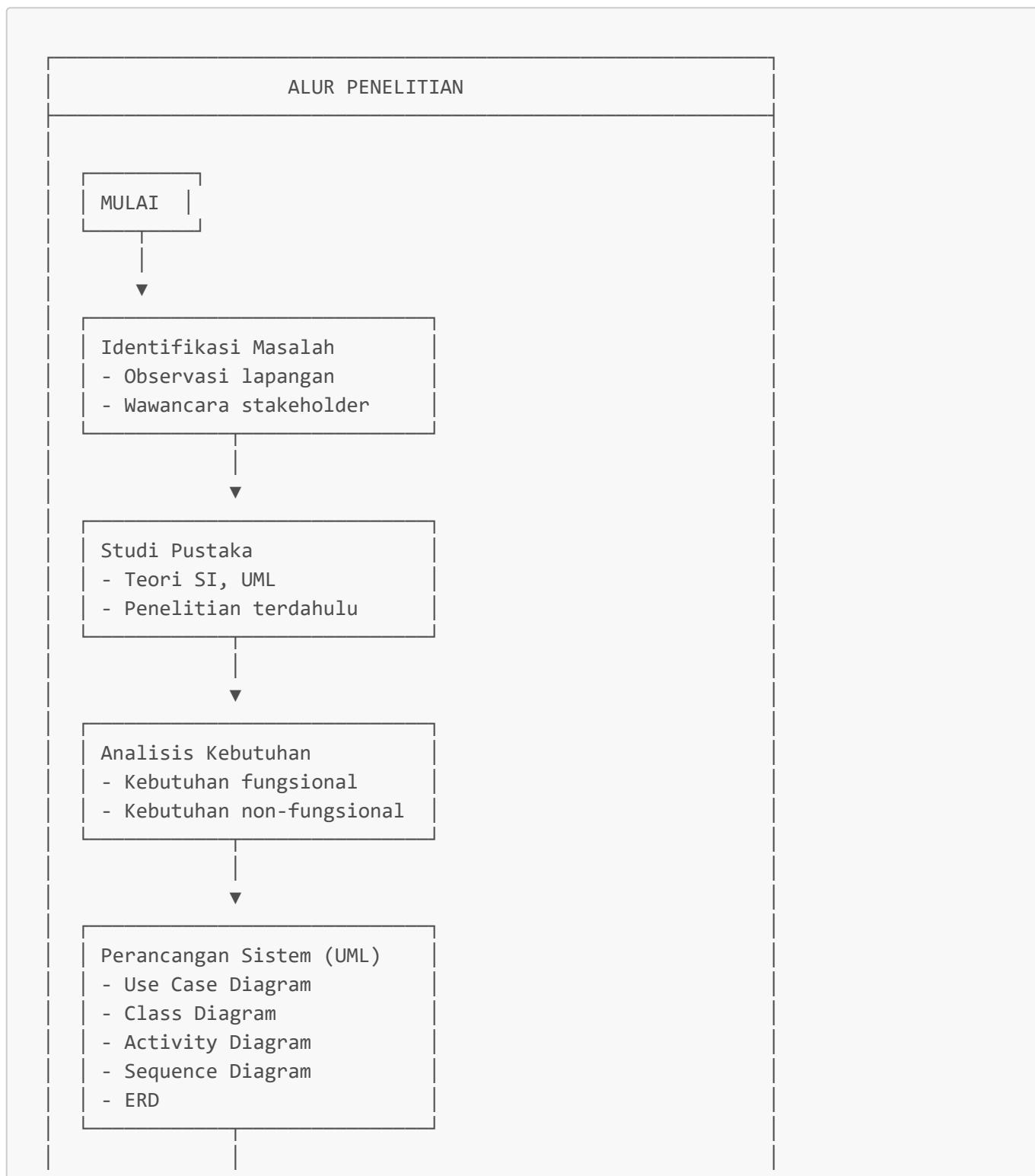
**Tabel 3. Perangkat Lunak yang Digunakan**

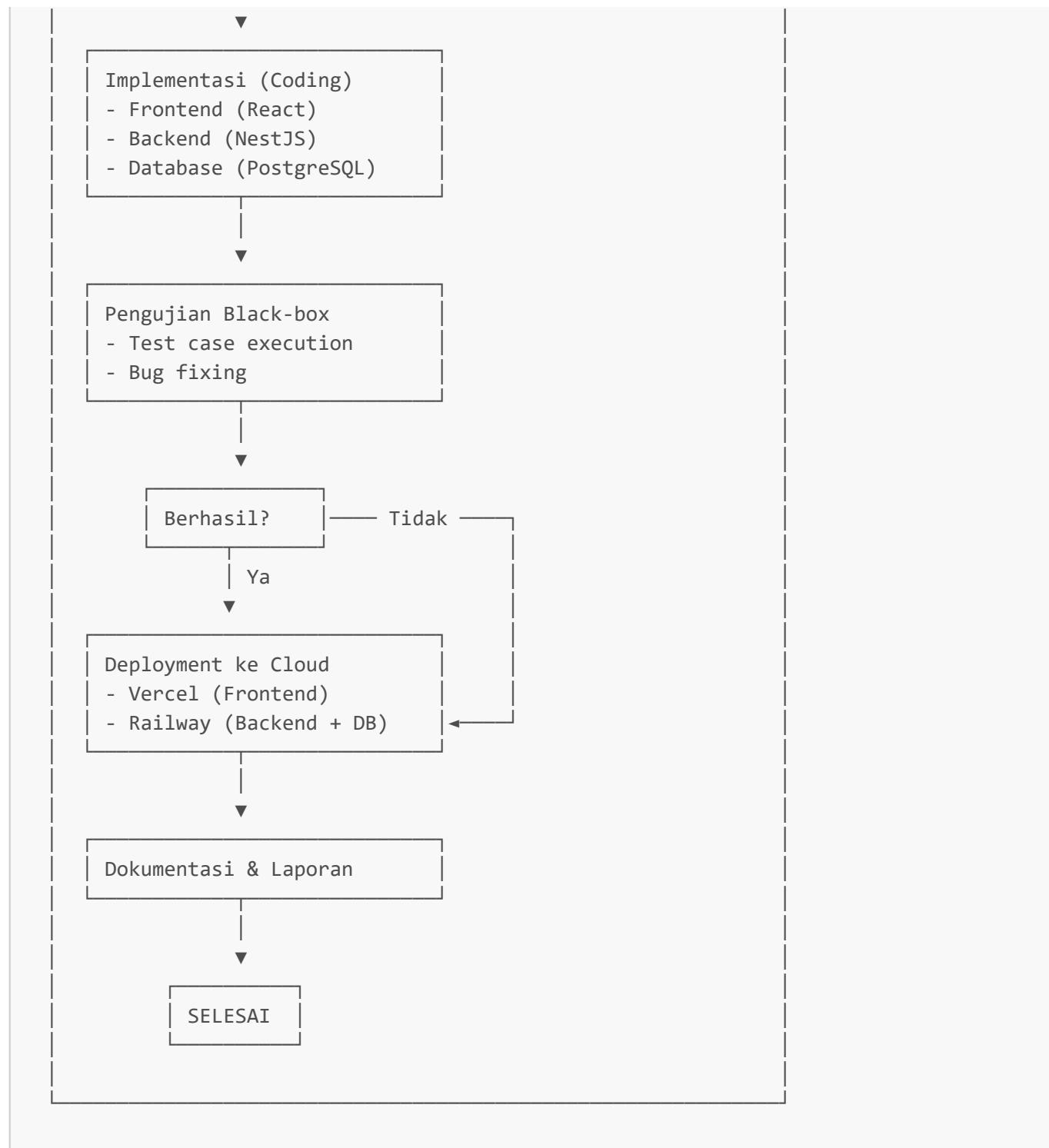
No	Kategori	Tools	Versi	Fungsi
1	IDE	Visual Studio Code	1.85	Code editor
2	Frontend Framework	React	18.2	UI library
3	Build Tool	Vite	5.0	Frontend bundler
4	CSS Framework	Tailwind CSS	3.4	Styling
5	UI Components	Shadcn/UI	Latest	Component library
6	Backend Framework	NestJS	10.2	REST API
7	ORM	Prisma	5.7	Database access
8	Database	PostgreSQL	15	Data storage
9	UML Tool	PlantUML	Latest	UML diagrams
10	Version Control	Git + GitHub	Latest	Source control
11	Frontend Hosting	Vercel	-	Cloud deployment

No	Kategori	Tools	Versi	Fungsi
12	Backend Hosting	Railway	-	Cloud deployment
13	Storage	Supabase	-	File storage
14	API Testing	Postman	10.x	API testing
15	Browser	Google Chrome	120	Testing

### 3.4 Alur Penelitian

Alur penelitian secara keseluruhan ditunjukkan pada Gambar 2.



**Gambar 2. Alur Penelitian**

### 3.5 Perancangan UML

Perancangan sistem menggunakan Unified Modeling Language (UML) dengan detail sebagai berikut:

**Tabel 4. Rincian Diagram UML yang Dibuat**

No	Jenis Diagram	Jumlah	Keterangan
1	Use Case Diagram	1	18 use case, 3 aktor
2	Class Diagram	1	22 class, 9 enum

No	Jenis Diagram	Jumlah	Keterangan
3	Activity Diagram	25	Per fitur utama
4	Sequence Diagram	18	Per use case
5	State Machine Diagram	2	Status order, payment
6	ERD	1	23 tabel
7	Deployment Diagram	1	Arsitektur cloud
<b>Total</b>		<b>49</b>	

### 3.6 Metode Pengujian

Pengujian sistem menggunakan metode **Black-box Testing** yang berfokus pada fungsionalitas tanpa memperhatikan struktur internal kode [19]. Black-box testing menguji apakah input menghasilkan output yang sesuai dengan spesifikasi.

Tahapan pengujian:

1. Menyusun test case berdasarkan kebutuhan fungsional
  2. Menjalankan setiap test case pada sistem
  3. Membandingkan hasil aktual dengan hasil yang diharapkan
  4. Mencatat status Pass/Fail untuk setiap test case
  5. Melakukan perbaikan jika ditemukan kegagalan
  6. Mengulangi pengujian hingga semua test case passed
- 

## 4. HASIL DAN PEMBAHASAN

### Analisis Sistem Berjalan

Proses bisnis agen LPG saat ini berjalan secara manual: pangkalan memesan via WhatsApp, operator mencatat di buku tulis, driver diberitahu manual untuk pengiriman, pembayaran dicatat terpisah di buku kas, dan rekapitulasi dikerjakan manual di Excel. Kelemahan sistem ini adalah tidak ada integrasi, rentan kesalahan, dan sulit monitoring real-time.

### Analisis Kebutuhan Sistem

Berdasarkan analisis, kebutuhan fungsional sistem meliputi: mengelola data pengguna dan pangkalan, mencatat dan mengubah status pesanan, mencatat pembayaran, mengelola stok, dan menghasilkan laporan. Kebutuhan non-fungsional meliputi: berbasis web responsif, keamanan login dengan JWT, mendukung minimal 100 pangkalan, dan waktu respon < 3 detik.

### Perancangan Sistem

**Use Case Diagram** menunjukkan sistem memiliki 3 aktor utama: Admin (mengelola master data), Operator (operasional pesanan, stok, pembayaran), dan Pangkalan (mengakses data milik sendiri). Total terdapat 18 use case yang teridentifikasi.

**Class Diagram** menunjukkan sistem memiliki 22 class yang terbagi dalam kategori: Master Data (users, agen, pangkalans, drivers, lpg\_products), Order Management (orders, order\_items, timeline\_tracks, invoices), Payment (order\_payment\_details, payment\_records), Stock (stock\_histories, penerimaan\_stok, penyaluran\_harian), dan Pangkalan Module (consumers, consumer\_orders, pangkalan\_stocks).

**ERD** menunjukkan database terdiri dari 23 tabel dengan relasi one-to-many dan one-to-one. Relasi utama meliputi: pangkalans (1) → orders (N), orders (1) → order\_items (N), dan orders (1) → order\_payment\_details (1).

## Implementasi Sistem

Arsitektur sistem menggunakan pendekatan three-tier: Presentation Layer (React di Vercel), Business Logic Layer (NestJS di Railway), dan Data Layer (PostgreSQL + Supabase Storage). Komunikasi antar layer menggunakan REST API dengan autentikasi JWT.

Implementasi fitur utama meliputi:

1. **Manajemen Pesanan** - CRUD pesanan dengan status tracking (DRAFT, DIPROSES, SIAP\_KIRIM, DIKIRIM, SELESAI)
2. **Manajemen Pembayaran** - Pencatatan DP, cicilan, pelunasan dengan cetak nota
3. **Manajemen Stok** - Penerimaan dari SPBE, penyaluran ke pangkalan, history pergerakan
4. **Pelaporan** - Dashboard KPI, export PDF/Excel format Pertamina
5. **Multi-Tenant** - Setiap pangkalan hanya dapat mengakses data miliknya

## Pengujian Sistem

Pengujian dilakukan dengan metode black-box testing untuk memastikan fungsionalitas sistem. Total 28 test case diuji meliputi autentikasi (5 test), manajemen pesanan (8 test), pembayaran (5 test), stok (6 test), dan laporan (4 test). Hasil pengujian menunjukkan seluruh test case passed dengan tingkat keberhasilan 100%.

**Tabel 1. Ringkasan Hasil Pengujian Black-Box**

Kategori	Jumlah Test	Pass	Fail	Percentase
Autentikasi	5	5	0	100%
Manajemen Pesanan	8	8	0	100%
Pembayaran	5	5	0	100%
Stok	6	6	0	100%
Laporan	4	4	0	100%
<b>Total</b>	<b>28</b>	<b>28</b>	<b>0</b>	<b>100%</b>

## Pembahasan

Sistem SIM4LON berhasil memenuhi kebutuhan yang telah diidentifikasi. Integrasi proses bisnis dalam satu platform mengurangi duplikasi pencatatan dan meningkatkan konsistensi data. Arsitektur multi-tenant berhasil mengisolasi data antar pangkalan dengan mekanisme row-level filtering berdasarkan pangkalan\_id.

Penggunaan teknologi modern (React + NestJS) memungkinkan pengembangan yang scalable dan maintainable. Dokumentasi UML yang lengkap (49 diagram) memudahkan pemeliharaan dan pengembangan lanjutan. Deployment cloud (Vercel + Railway) memungkinkan akses dari mana saja tanpa perlu infrastruktur server sendiri.

Keterbatasan sistem saat ini meliputi: belum ada rate limiting untuk pencegahan spam, backup database masih manual, dan belum mendukung mode offline. Hal ini menjadi rekomendasi pengembangan selanjutnya.

---

## 5. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa:

1. Sistem SIM4LON berhasil dirancang menggunakan pendekatan UML dengan arsitektur multi-tenant, terdiri dari 18 use case, 22 class, dan 23 tabel database.
2. Implementasi sistem berbasis web dengan React (frontend) dan NestJS (backend) dengan PostgreSQL sebagai database berhasil dilakukan dan telah di-deploy ke cloud.
3. Pengujian black-box menunjukkan seluruh 28 test case berfungsi sesuai spesifikasi dengan tingkat keberhasilan 100%.

Sistem ini dapat meningkatkan efisiensi operasional agen LPG melalui integrasi proses pencatatan, pelacakan real-time, dan otomatisasi laporan. Untuk pengembangan selanjutnya disarankan menambahkan rate limiting, backup otomatis, unit testing, dan mobile native application.

---

## DAFTAR PUSTAKA

- [1] Kementerian ESDM, "Statistik Minyak dan Gas Bumi 2023," Jakarta: Kementerian ESDM RI, 2023.
- [2] PT Pertamina, "Laporan Tahunan PT Pertamina (Persero) 2023," Jakarta: Pertamina, 2023.
- [3] C. J. Guo, W. Sun, Y. Huang, Z. H. Wang, and B. Gao, "A Framework for Native Multi-Tenancy Application Development and Management," in Proc. 9th IEEE Int. Conf. E-Commerce Technol. (CEC/EEE), 2007, pp. 551-558.
- [4] Meta Platforms, "React Documentation: A JavaScript library for building user interfaces," [Online]. Available: <https://react.dev>. [Accessed: Oct. 15, 2024].
- [5] NestJS, "NestJS Documentation: A progressive Node.js framework," [Online]. Available: <https://docs.nestjs.com>. [Accessed: Oct. 15, 2024].
- [6] PostgreSQL Global Development Group, "PostgreSQL 15 Documentation," [Online]. Available: <https://www.postgresql.org/docs/15>. [Accessed: Oct. 15, 2024].
- [7] G. B. Davis and M. H. Olson, Management Information Systems: Conceptual Foundations, Structure and Development, 2nd ed. New York: McGraw-Hill, 1985.
- [8] J. A. O'Brien and G. M. Marakas, Management Information Systems, 10th ed. New York: McGraw-Hill, 2017.
- [9] K. C. Laudon and J. P. Laudon, Management Information Systems: Managing the Digital Firm, 16th ed. London: Pearson, 2020.

- [10] Badan Pengatur Hilir Minyak dan Gas Bumi (BPH Migas), "Pedoman Distribusi LPG Bersubsidi," Jakarta: BPH Migas, 2022.
- [11] S. Aulakh, "Multi-Tenant Architecture: A Comprehensive Guide," Medium, 2023. [Online]. Available: <https://medium.com/@s.aulakh/multi-tenant-architecture>. [Accessed: Nov. 1, 2024].
- [12] J. Rumbaugh, I. Jacobson, and G. Booch, Unified Modeling Language Reference Manual, 2nd ed. Boston: Addison-Wesley, 2004.
- [13] Prisma, "Prisma Documentation: Next-generation Node.js and TypeScript ORM," [Online]. Available: <https://www.prisma.io/docs>. [Accessed: Oct. 20, 2024].
- [14] R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner's Approach, 9th ed. New York: McGraw-Hill, 2020.
- [15] A. Pratama, "Sistem Informasi Penjualan LPG Berbasis Web," Jurnal Teknologi Informasi, vol. 8, no. 2, pp. 45-56, 2022.
- [16] R. Hidayat, "Aplikasi Distribusi Gas Berbasis Web dengan Framework Laravel," Jurnal Sistem Informasi, vol. 5, no. 1, pp. 23-34, 2021.
- [17] B. Santoso, "Sistem Informasi Manajemen Agen LPG dengan Metode Prototype," Jurnal Informatika, vol. 10, no. 3, pp. 112-125, 2023.
- [18] D. Wijaya, "Aplikasi Monitoring Stok LPG Berbasis Java," Jurnal Teknologi Komputer, vol. 7, no. 2, pp. 78-89, 2022.
- [19] G. J. Myers, C. Sandler, and T. Badgett, The Art of Software Testing, 3rd ed. Hoboken: John Wiley & Sons, 2011.

---

## BIODATA PENULIS

**Luthfi Alfaridz** lahir di [Kota], [Tanggal]. Penulis menempuh pendidikan S1 di Program Studi Teknik Informatika, Fakultas Teknik, Universitas Suryakancana sejak tahun [tahun masuk]. Minat penelitian penulis meliputi pengembangan web, sistem informasi, dan cloud computing. Penulis dapat dihubungi melalui email: luthfi.alfaridz@student.unsur.ac.id.

---

*Artikel ini merupakan hasil penelitian Tugas Akhir  
Program Studi Teknik Informatika, Universitas Suryakancana  
© 2025*