

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Sebelum perancangan sistem dilakukan, beberapa penelitian sebelumnya ditelaah untuk melihat keterkaitan tema dengan penelitian ini, baik dari sisi objek kajian (penjualan dan persediaan *sparepart*) maupun dari sisi teknologi yang digunakan (integrasi payment *gateway*). Telaah ini digunakan sebagai bahan perbandingan sekaligus acuan dalam menentukan metode, kekurangan yang perlu diperbaiki, dan celah penelitian (research gap) yang dapat diisi oleh penelitian ini . Ringkasan penelitian terdahulu tersebut disajikan pada Tabel 2.1.

Tabel 2.1 Penelitian Terkait

No.	Penulis & Tahun	Judul/ Fokus Penelitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
1	Halawa & Kurniawan (2025) [27].	Perancangan Sistem Informasi Penjualan (POS) Berbasis Website dengan Metode Agile	PHP/MySQL berbasis web, metode Agile, Apache JMeter untuk performance testing	Berdasarkan pengujian performance testing: • Average Response Time: 2,3 detik • Peak Response Time: 4,1 detik • Error Rate: 0,5% •	Fokus penelitian hanya pada performa sistem (load/stress testing), belum menguji fungsionalitas fitur secara rinci (blackbox/UT); juga belum ada

No.	Penulis & Tahun	Judul/ Fokus Penelitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
				Throughput : 45 request/detik • Max Concurrent User: 200 user (passed), mengalami penurunan performa pada spike 300 use	integrasi payment gateway.
2	Raharjo et al., (2022) [28].	Perancangan Sistem Informasi dengan PHP dan MySQL untuk Pendaftaran Sekolah di Masa Pandemi — sistem PPDB online (pendaftaran, pembayaran, konfirmasi, cek status) untuk TK di Jakarta Utara	PHP, HTML, CSS, JavaScript ; MySQL; metode pengembangan Waterfall; UML (Use Case, ERD, LRS)	Black Box Testing pada 7 skenario pengujian (form pendaftaran, upload bukti bayar, publish artikel, kelola pengguna) — 7 dari 7 test case (100%) berstatus "Valid/Sesuai"	Evaluasi hanya menguji fungsionalitas (input-output), belum menguji keamanan sistem, performa/beban server, maupun kepuasan pengguna (UAT/kuesioner); studi kasus terbatas pada satu sekolah (skala kecil)

No.	Penulis & Tahun	Judul/ Fokus Penelitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
3	Ranti & Putra (2024) [29].	Penerapan Framework Laravel pada Sistem E-Booking (Studi Kasus: 4People Studio) — sistem e-booking berbasis web untuk pemesanan paket foto studio secara online, terintegrasi dengan pembayaran digital.	Laravel (Blade, Eloquent ORM, Composer); PHP; payment gateway Xendit; NGROK; metode pengembangan Agile.	Black Box Testing 6 skenario — 6/6 (100%) Valid; (2) System Usability Scale (SUS) skor 87,19 — grade B, rating "Excellent", kategori "Acceptable"; (3) Lighthouse Test: Performance 57 (sedang), Accessibility 98 (sangat baik), Best Practices 96, SEO 91 (sangat baik); (4) Survei kepuasan fitur (skala 1–10) ke 5 pengguna & 5 admin, dianalisis dengan rumus keputusan Upper/Lower terhadap nilai	Skor performa (kecepatan muat halaman) masih tergolong sedang (57/100) dan perlu optimasi; pelanggan belum dapat mengubah jadwal booking yang sudah dipesan; sistem belum mampu menangani pemesanan dengan jumlah orang tambahan yang melebihi kapasitas paket foto.

No.	Penulis & Tahun	Judul/ Fokus Penelitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
				ambang batas (T=7) — 7/8 fitur pengguna kategori "Upper" (baik), 1 fitur ("Ketersediaan Jadwal") kategori "Lower" (perlu perbaikan); seluruh fitur admin kategori "Upper"	
4	Rahmawati & Sumarsono (2024) [30]	Desain Pengembangan Website dengan Arsitektur Model View Controller pada Framework Laravel — analisis konseptual penerapan pola arsitektur MVC (Model, View,	Laravel (OOP); PHP; struktur direktori Laravel (App/Models, App/Http/Controllers, resources/views); Blade templating engine; PHPUnit; metode SDLC Waterfall;	Bersifat deskriptif-konseptual; black box testing (unit, functional, usability) disebutkan sebagai tahapan metodologi, namun tidak ada hasil pengujian atau skor evaluasi	Penelitian tidak membangun studi kasus aplikasi nyata dengan pengguna akhir (hanya mendemonstrasikan struktur kode generik User Model/Controller/View); tidak ada evaluasi performa, keamanan,

No.	Penulis & Tahun	Judul/ Fokus Penelitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
		Controller) pada framework Laravel untuk pengorganisasian kode program.	teknik observasi.	yang dilaporkan.	atau UX aktual — baru sebatas kerangka teoretis, bukan implementasi dan pengujian penuh.
5	Sumandito et al., (2024) [31]	Sistem Informasi Perpustakaan dengan Menggunakan Framework Laravel di Yayasan PB. Soedirman — sistem informasi berbasis web untuk pengelolaan koleksi buku, peminjaman, dan pengembalian buku perpustakaan.	Laravel; PHP; MySQL; metode pengembangan Waterfall; UML (Use Case Diagram, Activity Diagram, ERD, LRS)	Black Box Testing pada 2 proses bisnis utama: Form Login (3 skenario) dan Form Absensi (6 skenario) — total 9 dari 9 test case (100%) berstatus "Valid/Sesuai harapan"	Belum ada aplikasi mobile untuk akses perpustakaan; fitur pencarian belum memiliki filter spesifik (genre, penulis, tahun terbit); belum terintegrasi dengan sistem pembelajaran online; sistem keamanan (enkripsi data) belum diperkuat; koleksi digital (e-book, jurnal elektronik, multimedia)

No.	Penulis & Tahun	Judul/ FokusPen elitian	Teknologi & Alat	Matriks Evaluasi	Research Gap
					belum tersedia.

2.2 Landasan Teori

2.2.1 Website

Website adalah sekumpulan halaman informasi dalam bentuk digital baik berupa teks, gambar, video, maupun audio yang disediakan melalui jalur koneksi internet sehingga dapat diakses oleh pengguna di seluruh dunia [32]. Dalam dunia bisnis modern, *Website* berfungsi sebagai sarana untuk mengolah, menyimpan, dan menyampaikan informasi secara cepat guna mendukung efisiensi transaksi dan promosi produk secara luas tanpa batasan waktu [33]. Adapun gambaran visual mengenai website disajikan pada Gambar 2.1.



Gambar 2.1 *Website*

2.2.2 HTML (*Hypertext Markup Language*)

HTML merupakan bahasa pemrograman dasar dalam pengembangan sistem informasi berbasis *Website* modern yang

digunakan bersama dengan teknologi lain seperti CSS dan Javascript untuk membangun antarmuka sistem yang lengkap dan responsif [34]. Dalam arsitektur aplikasi berbasis web, *HTML* dikolaborasikan dengan bahasa pemrograman *PHP* di sisi *server* yang mampu menghasilkan dokumen *HTML* secara dinamis saat dieksekusi guna memfasilitasi akses dan pengolahan data dari *database* sebelum ditampilkan kepada pengguna melalui *web browser* [35]. Adapun gambaran visual mengenai *HTML* disajikan pada Gambar 2.2.



Gambar 2.2 *HTML*

2.2.3 *PHP (Hypertext Preprocessor)*

PHP merupakan bahasa pemrograman web atau *scripting language* yang didesain khusus untuk membangun aplikasi berbasis web [36]. *PHP* adalah bahasa *script server-side* yang bersifat *open source* dan dirancang untuk dapat bekerja sama dengan *database server* [37]. Hal ini dibuat sedemikian rupa guna mempermudah proses pembuatan dokumen *HTML* yang memerlukan akses langsung ke dalam basis data [38]. Adapun gambaran visual mengenai *PHP* disajikan pada Gambar 2.3.

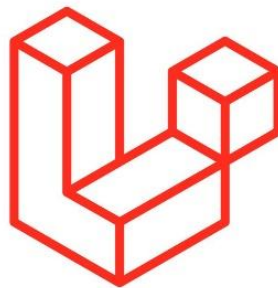
Gambar 2.3 *PHP*

Berikut contoh *script PHP*:

```
<?php
use Illuminate\Foundation\Inspiring;
use Illuminate\Support\Facades\Artisan;
Artisan::command('inspire', function () {
    $this->comment(Inspiring::quote());
})->purpose('Display an inspiring quote');
```

2.2.4 *Framework Laravel*

Laravel merupakan kerangka kerja (*Framework*) bahasa pemrograman *PHP* yang didesain untuk meningkatkan kualitas aplikasi dengan mengurangi beban biaya pengembangan dan memudahkan proses pemeliharaan (*maintenance*) [39]. *Framework* ini sangat membantu pengembang dalam merancang aplikasi berbasis *Website* karena menghasilkan kode program yang rapi, singkat, dan terstruktur sehingga meningkatkan produktivitas kerja [40]. Selain itu, *Laravel* memiliki beberapa keunggulan teknis diantaranya penggunaan *Command Line Interface* *Artisan*, dukungan pengelola paket *PHP Composer*, serta penulisan sintaks yang mudah dimengerti oleh pengembang [41]. Adapun gambaran visual mengenai *Framework Laravel* disajikan pada Gambar 2.4.

Gambar 2.4 *Framework Laravel*

2.2.5 *Laragon*

Laragon adalah perangkat lunak yang sangat berguna dalam lingkungan pengembangan *Website* berbasis *PHP* dan *MySQL* [42]. Perangkat lunak ini memiliki keunggulan utama yaitu kemampuannya untuk berperan sebagai server web Apache, sehingga pengembang dapat melakukan simulasi pengembangan *Website* secara lokal di komputer sebelum sistem diimplementasikan secara daring [43]. Adapun gambaran visual mengenai *Laragon* disajikan pada Gambar 2.5.

Gambar 2.5 *Laragon*

2.2.6 *MySQL*

MySQL merupakan suatu perangkat lunak sistem manajemen basis data *SQL* (*Database Management System* atau *DBMS*) yang memiliki karakteristik multithread serta multi-user [44]. *MySQL* berperan sebagai database engine atau server basis data yang

mendukung bahasa SQL (Structured Query Language) sebagai bahasa interaktif utama dalam mengelola, menyimpan, dan mengorganisir data secara sistematis [45]. Dalam pengembangan aplikasi berbasis *Website*, *MySQL* sangat populer digunakan karena keandalannya dalam menangani pengolahan data pada basis data sehingga informasi dapat ditampilkan secara dinamis melalui bahasa pemrograman seperti *PHP* [46]. Adapun gambaran visual mengenai *MySQL* disajikan pada Gambar 2.6.



Gambar 2.6 *MySQL*

2.2.7 *Visual Studio Code*

Visual Studio Code digunakan sebagai editor teks utama dalam penulisan dan pengelolaan kode program. Dalam pengembangan sistem berbasis web ini, perangkat lunak tersebut berperan sebagai alat bantu implementasi rancangan sistem ke dalam bahasa pemrograman [47]. Fleksibilitasnya dalam mendukung Framework Laravel dan integrasi dengan basis data *MySQL* menjadikannya komponen penting dalam lingkungan pengembangan aplikasi [48]. Adapun gambaran visual mengenai *Visual Studio Code* website disajikan pada Gambar 2.7.



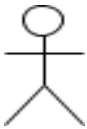
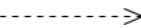
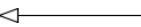
Gambar 2.7 *Visual Studio Code*

2.2.8 UML (*Unified Modeling Language*)

Unified Modeling Language merupakan bahasa pemodelan visual yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan rancangan dari suatu sistem perangkat lunak [49]. Pemodelan ini memberikan gambaran yang jelas mengenai sistem yang akan dibangun, baik dari sisi struktural maupun fungsional [50]. Beberapa diagram *UML* yang digunakan dalam perancangan sistem meliputi:

1. *Use Case Diagram*: Merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem informasi yang akan dibuat [51]. Notasi yang digunakan dalam perancangan *use case diagram* pada penelitian ini dapat dilihat pada Tabel 2.2.

Tabel 2.2 *Use Case Diagram*

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Merepresentasikan pengguna atau entitas luar yang berinteraksi dengan sistem.
2		<i>Generalization</i>	hubungan yang menunjukkan pewarisan (inheritance) antara aktor atau use case, di mana satu elemen merupakan turunan dari elemen yang lebih umum.
3		<i>Dependency</i>	Dependency adalah hubungan antar use case yang menunjukkan bahwa suatu proses membutuhkan proses lain agar bisa berjalan dengan baik.
4		<i>Extend</i>	hubungan antar use case yang menunjukkan adanya fitur tambahan (opsional) yang hanya dijalankan jika kondisi tertentu terpenuhi.
5		<i>Include</i>	hubungan <<include>> digunakan untuk menunjukkan bahwa suatu use case (sumber) secara otomatis akan memanggil atau membutuhkan fungsi dari use case lain (tujuan) agar dapat berjalan dengan lengkap.
6		<i>Association</i>	Menghubungkan Aktor dengan Use Case. Menunjukkan bahwa aktor tersebut berinteraksi atau

No	Gambar	Nama	Keterangan
			menggunakan fitur tersebut.
7		<i>Use Case</i>	Representasi dari fungsionalitas atau fitur utama yang disediakan oleh sistem.
8		<i>Collaboration</i>	Sekumpulan <i>use case</i> atau objek yang bekerja sama untuk menjalankan fungsi tertentu.
9		<i>Note</i>	Keterangan tambahan atau anotasi untuk memberikan detail lebih lanjut pada diagram.
10		<i>System</i>	Batas yang memisahkan antara sistem yang sedang dibangun dengan aktor luar.

2. *Activity Diagram*: Menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, keputusan (*decision*) yang mungkin terjadi, dan bagaimana alur tersebut berakhir [52]. Notasi yang digunakan dalam perancangan *activity diagram* pada penelitian ini dapat dilihat pada Tabel 2.3.

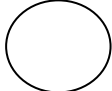
Tabel 2.3 *Activity Diagram*

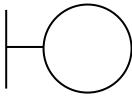
NO	Gambar	Nama	Keterangan
1		<i>Initial State</i>	Titik dimulainya suatu aliran kerja atau aktivitas dalam sistem.

NO	Gambar	Nama	Keterangan
2		<i>Final State</i>	Titik yang menandakan akhir dari seluruh rangkaian aktivitas.
3		<i>Activity</i>	Representasi dari sekumpulan aksi yang membentuk satu prosedur besar.
4		<i>Action</i>	Unit terkecil dari aktivitas yang dilakukan oleh sistem atau aktor.
5		<i>Decision</i>	Titik percabangan yang memiliki lebih dari satu jalur keluaran berdasarkan kondisi tertentu.

3. *Sequence Diagram*: Menggambarkan interaksi antar objek yang menunjukkan seorang aktor sedang melakukan pengelolaan data dalam satu skenario tertentu. *Sequence diagram* menggunakan beberapa notasi untuk menggambarkan interaksi antar objek berdasarkan urutan waktu, termasuk pengiriman pesan (*message*) antar objek dalam suatu proses [53]. Notasi-notasi tersebut disajikan pada Tabel 2.4.

Tabel 2.4 *Simbol Sequence Diagram*

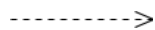
No	Gambar	Nama	Keterangan
1		<i>Entity Class</i>	Gambaran sistem sebagai landasan dalam menyusun basis data.
2		<i>Life Line</i>	Komponen yang digambarkan garis putus terhubung dengan objek (menunjukkan masa hidup objek).

No	Gambar	Nama	Keterangan
3		<i>Recursive</i>	Pesan untuk dirinya sendiri (proses internal pada objek).
4		<i>Boundary Class</i>	Menangani komunikasi antar lingkungan sistem (antarmuka/interface).

4. *Class Diagram*: Diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem atau perangkat lunak yang sedang digunakan, termasuk hubungan antar tabel pada basis data [54]. Notasi-notasi tersebut disajikan pada Tabel 2.5.

Tabel 2.5 *Simbol Class Diagram*

NO	Gambar	Nama	Keterangan
1		<i>Class</i>	Struktur utama sistem yang membungkus data (<i>attributes</i>) dan perilaku (<i>methods</i>) dalam satu kesatuan logika.
2		<i>Collaboration</i>	Menggambarkan interaksi kolektif antara sekumpulan objek yang bekerja sama untuk merealisasikan fungsi tertentu dalam sistem.
3		<i>Nary Association</i>	Menghubungkan lebih dari dua kelas dalam satu relasi tunggal untuk menunjukkan ketergantungan yang kompleks antar entitas.
4		<i>Generalization</i>	Menunjukkan hubungan pewarisan (<i>inheritance</i>) di mana kelas anak mewarisi seluruh atribut

NO	Gambar	Nama	Keterangan
			dan metode dari kelas induk.
5		<i>Association</i>	Hubungan struktural statis yang mendefinisikan koneksi antar objek.
6		<i>Dependency</i>	Menunjukkan bahwa suatu kelas menggunakan atau bergantung pada kelas lain, sehingga perubahan pada kelas tujuan dapat memengaruhi kelas sumber.
7		<i>Realization</i>	Hubungan yang menunjukkan bahwa suatu kelas mengimplementasikan seluruh kontrak atau operasi yang didefinisikan pada sebuah <i>Interface</i> .

BAB III

METODOLOGI PENELITIAN

3.1 Prosedur Penelitian

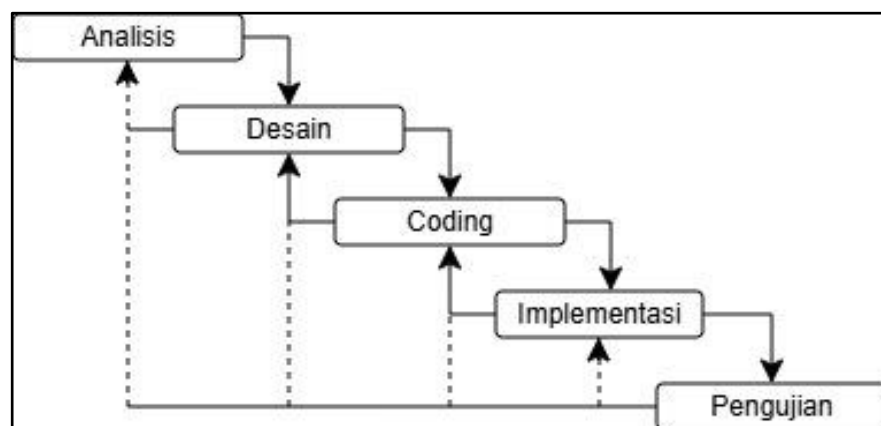
Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode Waterfall, yaitu pendekatan yang dilakukan secara bertahap dan berurutan mulai dari perencanaan, analisis, perancangan, implementasi, hingga pengujian, di mana setiap tahap harus diselesaikan terlebih dahulu sebelum berlanjut ke tahap berikutnya. Metode ini dipilih karena kebutuhan Sistem Informasi Penjualan Suku Cadang Motor berbasis web pada Hery Motor telah teridentifikasi secara jelas sejak tahap observasi awal, meliputi kebutuhan pengelolaan produk, transaksi penjualan ke reseller, hingga integrasi *payment gateway*, sehingga pendekatan yang terstruktur dan berurutan ini memudahkan peneliti dalam mengembangkan, mengendalikan, serta mendokumentasikan setiap tahapan pembangunan sistem secara sistematis.

Berdasarkan kajian terhadap penelitian terdahulu yang membahas pengembangan aplikasi *e-commerce* berbasis Laravel, ditemukan bahwa penelitian Mahendra et al. (2024) dalam "Pemodelan Sistem Penjualan Suku Cadang Motor dan Layanan Cepat Berbasis Website di Bengkel Meca" masih berfokus pada studi literatur mengenai urgensi website sebagai sarana promosi, tanpa mengimplementasikan fitur transaksi digital seperti *payment gateway* maupun manajemen mitra reseller. Sementara itu, penelitian

Yulvianda & Ismail (2024) pada "Rancangan Sistem Pembayaran pada Toko Pempek Mama Tika Menggunakan Laravel dengan Payment Gateway Midtrans" telah berhasil mengatasi masalah pencatatan manual dan antrean kasir melalui integrasi Midtrans, namun ruang lingkupnya masih terbatas pada satu jenis pengguna, yaitu konsumen langsung, tanpa skema harga khusus untuk mitra bisnis. Demikian pula penelitian Ratino et al. (2023) dalam "Implementasi Framework Laravel dalam Pengembangan Aplikasi E-Commerce untuk Toko Jago Software" berhasil mendigitalisasi proses jual beli produk secara daring, tetapi belum menyertakan pengujian performa sistem maupun integrasi dengan aplikasi mobile pihak ketiga untuk sinkronisasi data secara *real-time*. Ketiga penelitian tersebut menunjukkan bahwa pengembangan *e-commerce* berbasis Laravel umumnya masih dirancang untuk skema penjualan langsung ke konsumen akhir (*business-to-consumer*) tanpa mengakomodasi kebutuhan kemitraan reseller. Kondisi ini sejalan dengan permasalahan yang ditemukan di lapangan pada Hery Motor, di mana proses penjualan ke reseller sebelumnya masih dilakukan melalui komunikasi manual (telepon/pesan) tanpa katalog harga khusus reseller maupun status transaksi yang transparan, sehingga rawan menimbulkan kesalahan pencatatan dan keterlambatan konfirmasi pembayaran. Berdasarkan kesenjangan tersebut, penelitian ini mengembangkan sistem informasi penjualan berbasis web yang mengintegrasikan *payment gateway* Midtrans, menyediakan panel khusus reseller dengan skema harga tersendiri, alur retur/refund tiga tahap, serta integrasi data dengan aplikasi mobile Flutter

untuk sinkronisasi stok antar gudang secara *real-time* — sebuah cakupan yang belum diakomodasi secara menyeluruh pada penelitian-penelitian *e-commerce* berbasis Laravel sebelumnya.

Berdasarkan hasil observasi pada Hery Motor, ditemukan bahwa proses penjualan suku cadang kepada reseller masih dilakukan secara manual melalui komunikasi telepon atau pesan singkat, sehingga menyebabkan keterlambatan konfirmasi pembayaran, kesulitan admin dalam memantau riwayat dan status transaksi tiap reseller, serta rawan terjadi kesalahan pencatatan akibat tidak adanya sistem yang mencatat transaksi secara terpusat. Oleh karena itu, penelitian ini melakukan pengembangan sistem dengan pendekatan berbeda, yaitu menggunakan framework Laravel sebagai basis aplikasi web serta integrasi *payment gateway* Midtrans untuk memvalidasi status pembayaran secara otomatis dan *real-time*, guna mendukung proses transaksi dan pelayanan reseller yang lebih cepat, akurat, dan transparan. Tahapan-tahapan tersebut digambarkan dalam bentuk diagram alur penelitian seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Prosedur Penelitian

1. Perencanaan

Perencanaan (*Planning*) Tahap awal ini berfokus pada identifikasi masalah dan penentuan lingkup penelitian. Peneliti melakukan pengumpulan data awal melalui wawancara dengan pemilik atau pihak terkait untuk memahami kendala yang ada pada sistem yang sedang berjalan serta merumuskan tujuan pengembangan sistem baru.

2. Analisis

Pada tahap ini, dilakukan analisis mendalam terhadap kebutuhan pengguna (*user requirements*). Proses ini mencakup pengumpulan spesifikasi fungsional (apa saja yang bisa dilakukan sistem) dan non-fungsional (perangkat keras/lunak yang dibutuhkan). Hasilnya biasanya dituangkan dalam dokumen spesifikasi kebutuhan perangkat lunak (*Software Requirement Specification*) sebagai acuan pengembangan.

3. Perancangan

Perancangan (*Design*) Tahap desain bertujuan untuk mengubah hasil analisis kebutuhan menjadi model teknis atau "cetak biru" (*blueprint*) sistem sebelum pengkodean dimulai. Perancangan ini meliputi arsitektur sistem, struktur basis data (ERD), dan antarmuka pengguna menggunakan diagram UML (*Use Case, Activity, Class, dan Sequence Diagram*) agar sistem dapat dipahami secara visual dan teknis.

4. Implementasi

Tahap ini berfokus pada implementasi rancangan desain ke dalam bahasa pemrograman. Proses kompilasi sistem mengandalkan teknologi

PHP, Laravel, dan MySQL. Di dalamnya, fungsionalitas inti yang meliputi pengelolaan produk, pencatatan transaksi, hingga pembuatan laporan dikembangkan langsung menjadi modul program yang nyata.

5. Pengujian

Pengujian (*Testing*) Setelah pengkodean selesai, sistem harus diuji untuk memastikan seluruh fungsi berjalan dengan benar dan bebas dari kesalahan (*bug*). Metode yang umum digunakan adalah Black-Box Testing, yaitu pengujian yang berfokus pada input dan output sistem untuk memverifikasi apakah fungsionalitas aplikasi sudah sesuai dengan kebutuhan pengguna yang telah didefinisikan di awal.

3.2 Metode Pengumpulan Data

1. Observasi

Melalui observasi langsung di Kecamatan Talang, Kabupaten Tegal, peneliti mengamati jalannya proses bisnis, mekanisme transaksi manual, hingga hambatan teknis yang terjadi di lapangan. Pendekatan ini mempermudah peneliti dalam memetakan kendala nyata pada sistem konvensional secara objektif..

2. Wawancara

Teknik pengumpulan data dilakukan melalui tanya jawab secara langsung kepada pemilik usaha, Tujuan dari wawancara ini adalah untuk menggali informasi mendalam mengenai kebutuhan fungsional sistem

(apa saja yang harus bisa dilakukan aplikasi) serta mendapatkan data spesifik terkait produk, harga, dan manajemen stok.

3. Studi Literatur

Selain observasi dan wawancara, peneliti juga melakukan studi literatur dengan mengumpulkan serta mengkaji berbagai sumber referensi yang relevan, seperti buku, jurnal ilmiah, dan hasil penelitian terdahulu yang berkaitan dengan sistem informasi penjualan dan persediaan barang, penerapan *Framework* Laravel, serta integrasi *Payment Gateway*. Studi literatur ini bertujuan untuk memperkuat landasan teori, membandingkan metode maupun pendekatan yang telah digunakan pada penelitian sejenis sebelumnya, serta memastikan bahwa rancangan sistem yang dibangun sesuai dengan kaidah keilmuan dan praktik terbaik yang berlaku.

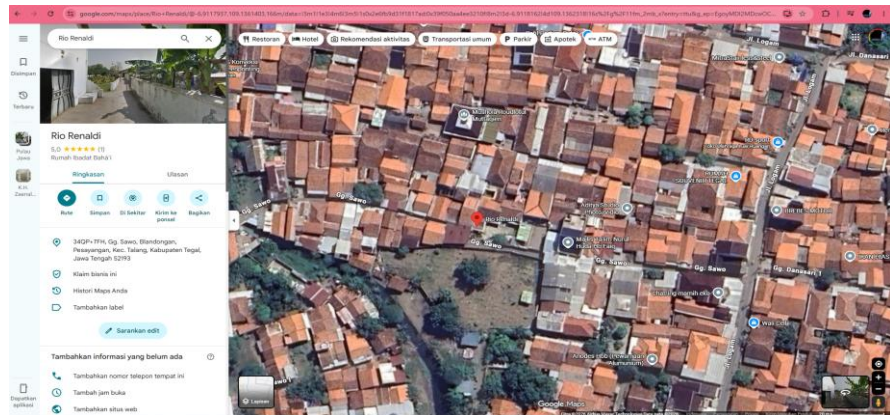
3.3 Waktu dan Tempat Penelitian

3.3.1 Waktu Penelitian

Penelitian ini dilaksanakan dalam kurun waktu kurang lebih enam bulan, terhitung sejak bulan Januari 2026 hingga bulan Juni 2026.

3.3.2 Tempat Penelitian

Tempat penelitian dilakukan pada Hery Motor, yang terletak di Jawa Tengah, Kabupaten Tegal, Kec. Talang, Desa Pesayangan Jln Logam RT06/RW01 52193. Tempat penelitian bisa dilihat pada gambar 3.2.



Gambar 3.2 Tempat Penelitian