

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Sebagai acuan perancangan sistem, kajian terhadap beberapa penelitian terdahulu yang relevan telah dirangkum secara komparatif pada

Tabel 2.1

No	Rujukan (Penulis/ Tahun)	Data/Datase t/Masukan	Model / Algoritma / Framework	Capaian / Hasil Evaluasi	Research Gap (Celah Penelitian)
1	Cepi Rahmat Hidayat (2023)	Data pelanggan, data pemesanan, data lokasi layanan towing	Sistem informasi berbasis web (PHP & MySQL)	Memperm udah pelanggan menemuka n layanan towing terdekat serta meningkat kan efektivitas proses	Belum menyediak an pelacakan posisi kendaraan secara real-time mengguna kan OpenStree tMap dan

				pemesana. n.	Leaflet serta belum mendukung pengelolaan operasional secara terintegrasi.
2	Rifqi, Muhammad Nawwal Izzur., Nindyasari, Ratih., Murti, Alif Catur (2025)	Data koleksi buku, anggota, transaksi peminjaman	Laravel Framework	Pengelolaan data perpustakaan menjadi lebih efektif dan terstruktur dibanding sistem manual.	Objek penelitian masih terbatas pada perpustakaan dan belum melakukan modifikasi layanan

					towing, pelacakan lokasi, serta pengelolaan armada.
3	Fanani, Khairul., Auliana, Sigit., Chafid, Nurul (2025)	Data pelayanan administrasi masyarakat	Laravel Framework, SDLC	Meningkatkan efisiensi pelayanan administrasi dan mengurangi kesalahan pencatatan .	Belum mendukung proses pemesanan layanan, pelacakan kendaraan, dan monitoring lokasi secara real-time.
4	Sujarwo., Fathoni, Muhammad	Data pengiriman barang,	Sistem informasi berbasis web	Memperbaiki pengelolaan	Tracking masih berorientasi pada

		status pengiriman		pengiriman serta penyajian informasi status kiriman kepada pelanggan.	status pengiriman barang dan belum menyediakan monitorin g posisi kendaraan towing secara real-time berbasis peta digital.
5	Sari Azhariyah., Muhamad Mukhlis (2023)	Data antarmuka (UI) website	Tailwind CSS, Bootstrap CSS	Tailwind CSS menghasilkan tampilan yang responsif,	Penelitian hanya membahas pengembangan antarmuka tanpa

	dan Miftah Faroq Santoso (2025)			konsisten, dan fleksibel dalam pengemba ngan antarmuka .	mengimpl ementasik an sistem informasi layanan towing maupun fitur tracking kendaraan.
6	Hilman Firmansy ah (2025)	Data koordinat GPS kendaraan	IoT (NodeMCU, GPS, GSM)	Sistem mampu melakukan pelacakan kendaraan secara real-time dan meningkat kan efisiensi	Memerluk an perangkat IoT tambahan sehingga implement asinya lebih kompleks dibanding kan

				pengelolaan armada.	pendekatan berbasis Geolocation API pada penelitian ini.
7	Angel Caroline Billan (2024)	Data GPS kendaraan operasional, data kondisi kendaraan	GPS Tracker, GSM	Monitoring armada menjadi lebih efektif serta membantu pengambilan keputusan operasional.	Fokus pada monitoring armada operasional, belum mendukung proses pemesanan layanan towing dan pelacakan driver berbasis web

					menggunakan OpenStreetMap dan Leaflet.
8	Marleni., Gunaryati (2023)	Data lokasi presensi karyawan	Laravel Framework, Leaflet JS	Berhasil menampilkan lokasi pengguna pada peta digital secara akurat.	Implementasi hanya pada sistem presensi karyawan dan belum mengembangkan sistem layanan towing yang mengintegrasikan pemesanan, tracking

					driver, dan pengelolaa n operasiona l.
--	--	--	--	--	--

2.2 Landasan Teori

2.2.1 Sistem Informasi Berbasis Web

Sistem informasi berbasis web adalah suatu sistem yang dirancang untuk mengumpulkan, memproses, menyimpan, dan mengelola data dengan memanfaatkan teknologi web sehingga dapat diakses secara online melalui *browser* internet. Sistem ini difungsikan untuk mempermudah penggunaan, pengolahan data, serta pengambilan keputusan dibandingkan dengan sistem manual karena prosesnya yang otomatis dan terstruktur. Teknologi web memungkinkan sistem informasi dapat digunakan oleh banyak pengguna tanpa dibatasi oleh lokasi geografis, sehingga informasi dapat diakses secara real-time kapan saja diperlukan [19].

Selain itu, penelitian dalam konteks *e-commerce* dan pelacakan pengiriman berbasis web menunjukkan bahwa sistem informasi web dapat meningkatkan efektivitas operasional dengan menyediakan fitur seperti pelacakan status, notifikasi, dan laporan

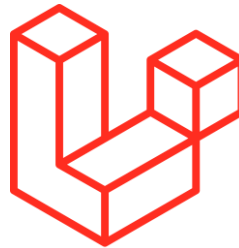
secara otomatis yang bertujuan meningkatkan efisiensi pelayanan dan kepuasan pelanggan. Pengembangan sistem pelacakan pengiriman logistik berbasis web mampu memberikan informasi yang akurat dan terperinci mengenai status barang serta mempercepat proses bisnis perusahaan jasa pengiriman [20].

2.2.2 Framework Laravel

Laravel adalah *framework* aplikasi web berbasis bahasa pemrograman PHP yang menerapkan arsitektur Model-View-Controller (MVC). Framework ini bersifat *open-source* dan dirancang untuk mempermudah pengembangan aplikasi web secara terstruktur, cepat, dan aman. Laravel menyediakan berbagai fitur modern seperti routing, *middleware*, ORM (*Eloquent*), sistem autentikasi, manajemen basis data dengan migrasi, serta dukungan testing otomatis yang membantu *developer* dalam membangun aplikasi web dengan kode yang bersih dan mudah dipelihara [21].

Dalam penelitian mengenai sistem informasi manajemen stok berbasis web, Laravel digunakan sebagai basis pengembangan untuk menghasilkan aplikasi yang terintegrasi, responsif, dan efisien dalam pengelolaan data, serta meningkatkan akurasi pencatatan dan kepuasan pengguna. Penelitian tersebut menunjukkan bahwa penggunaan Laravel dapat meminimalkan kesalahan pelaporan, meningkatkan responsivitas sistem, serta memaksimalkan struktur

kode melalui fitur-fiturnya [22]. Gambar 2.1 ini merupakan contoh logo dari Framework Laravel.



Gambar 2.1 Logo Laravel

(Sumber: <https://www.logo.wine/logo/Laravel>)

2.2.3 Metode Pengembangan Sistem

Metode pengembangan sistem merupakan suatu pendekatan sistematis yang digunakan dalam perancangan dan pembangunan sistem informasi agar sistem yang dihasilkan sesuai dengan kebutuhan pengguna. Salah satu metode pengembangan yang banyak digunakan dalam penelitian sistem informasi adalah *Software Development Life Cycle* (SDLC). SDLC terdiri dari beberapa tahapan, yaitu analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Metode ini bertujuan untuk menghasilkan perangkat lunak yang berkualitas melalui proses yang terstruktur dan terdokumentasi dengan baik.

2.2.4 Penerapan Laravel Dalam Sistem Informasi

Laravel merupakan framework PHP berbasis arsitektur Model–View–Controller (MVC) yang dirancang untuk mempermudah pengembangan aplikasi web secara terstruktur, aman, dan efisien. Laravel menyediakan fitur seperti *routing*, ORM *Eloquent*, autentikasi, dan manajemen basis data yang mendukung pengembangan sistem informasi berbasis web.

Selain itu, pengembangan sistem informasi profil perusahaan berbasis website menggunakan Laravel berhasil menghasilkan aplikasi web yang dinamis dan responsif untuk menyajikan informasi publik perusahaan dengan efektif, menunjukkan bahwa Laravel cocok digunakan dalam konteks informasi layanan kepada stakeholder [23].

2.2.5 Framework CSS Tailwind

Tailwind CSS adalah sebuah *utility-first CSS framework* yang digunakan untuk membangun antarmuka pengguna (*user interface*) yang responsif, fleksibel, dan mudah disesuaikan tanpa harus menulis banyak kode CSS kustom. Pendekatan *utility-first* pada Tailwind CSS berarti bahwa pengembang dapat langsung menambahkan kelas utilitas ke elemen HTML untuk menentukan gaya tampilan, seperti ukuran, warna, spasi, dan responsivitas, yang dapat mempercepat proses desain dan meningkatkan konsistensi

antarmuka aplikasi web. Hal ini membuat Tailwind CSS sangat cocok digunakan pada pengembangan sistem informasi berbasis web yang memerlukan antarmuka yang adaptif dan mudah diatur [14]. Gambar 2.2 di bawah ini merupakan contoh dari logo Tailwind CSS.



Gambar 2.2 Logo Tailwind CSS

(Sumber: <https://s11nk.com/tailwindcss>)

2.2.6 MySQL

MySQL adalah salah satu Sistem Manajemen Basis Data Relasional (RDBMS) yang umum digunakan dalam pengembangan aplikasi web modern. MySQL memanfaatkan *Structured Query Language* (SQL) sebagai bahasa standar untuk pengelolaan data, sehingga memudahkan pengembang dalam melakukan operasi penyimpanan, pencarian, pembaruan, dan penghapusan data secara efisien. Database ini memiliki sifat *open-source*, mendukung banyak pengguna (*multi-user*) dan pemrosesan paralel (*multi-threaded*), serta dapat diintegrasikan dengan berbagai bahasa pemrograman seperti PHP, Python, dan Java. MySQL banyak digunakan pada aplikasi web karena kemampuannya dalam

menangani data dalam jumlah besar dengan kinerja yang stabil serta dukungan komunitas yang luas [24]. Gambar 2.3 di bawah ini merupakan contoh dari logo MySQL.



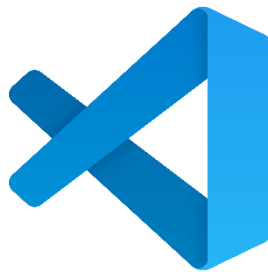
Gambar 2.3 Logo MySQL

(Sumber: <https://sllnk.com/logomysql>)

2.2.7 Visual Studio Code

Visual Studio Code (*VS Code*) adalah salah satu editor kode sumber (*source code editor*) yang ringan, fleksibel, dan populer di kalangan pengembang perangkat lunak modern. VS Code mendukung berbagai bahasa pemrograman seperti PHP, JavaScript, HTML, dan CSS, serta memiliki fitur penting seperti penyorotan sintaks (*syntax highlighting*), pelengkapan kode otomatis (*auto-completion*), debugging terintegrasi, serta pengelolaan versi melalui *extension*. Fitur-fitur ini membantu programmer dalam mempercepat proses penulisan, pemeliharaan, dan pengujian kode secara efisien, terutama dalam pengembangan aplikasi berbasis web. Penelitian ini menunjukkan bahwa penggunaan Visual Studio Code

dapat memudahkan pengembang dalam menyusun kode program dan membantu menjamin konsistensi penulisan selama proses pembuatan sistem [25]. Gambar 2.4 di bawah ini merupakan contoh dari logo Visual Studio Code.



Gambar 2.4 Logo Visual Studio Code

(Sumber: <https://11nq.com/visualstudiocode>)

2.2.8 Web Browser

Web browser adalah perangkat lunak aplikasi yang berfungsi sebagai antarmuka utama antara pengguna dan aplikasi berbasis web, karena browser merupakan media yang digunakan untuk mengakses, menampilkan, dan berinteraksi dengan konten serta sistem informasi yang disajikan melalui jaringan internet. Tanpa web browser, pengguna tidak dapat mengakses aplikasi web karena browser menerjemahkan elemen kode seperti HTML, CSS, dan JavaScript sehingga dapat dilihat dan digunakan oleh pengguna secara visual. Dengan adanya browser modern seperti Google Chrome, Mozilla Firefox, dan Microsoft Edge, pengguna dapat mengakses sistem berbasis web kapan saja dan dari berbagai

perangkat seperti komputer atau perangkat mobile secara real-time [26].

2.2.9 UML (Unified Modeling Language)

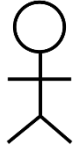
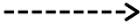
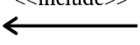
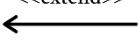
Unified Modeling Language (UML) merupakan bahasa pemodelan yang dipakai untuk merancang, menggambarkan, dan mendokumentasikan sistem perangkat lunak, terutama yang berorientasi objek. UML berfungsi membantu pengembang maupun analis sistem dalam menjelaskan struktur sistem, alur kerja, serta hubungan antar komponen secara teratur dan mudah dipahami sebelum proses implementasi dilakukan. Penggunaan UML juga mempermudah analisis kebutuhan sistem, mengurangi kemungkinan kesalahan saat pengembangan, serta meningkatkan komunikasi antara pengembang, pengguna, dan pihak terkait lainnya.

1. *Use Case Diagram*

Digunakan untuk menggambarkan fungsi-fungsi yang diinginkan dari sebuah sistem dan menggambarkan sebuah interaksi antara aktor dan sistem. Didalam *Use Case Diagram* terdapat 6 (enam) aktor yang menggambarkan entitas manusia maupun sebuah sistem yang melakukan pekerjaan di sistem.

Tabel 2.1 menunjukkan simbol-simbol *use case diagram*

Tabel 2.1 Simbol *Use Case* Diagram

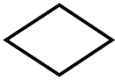
No	Gambar	Simbol	Keterangan
1.		Aktor	Mewakili peran seorang system yang lain, atau alat ketika berkomunikasi dengan Use Case.
2.		Use Case	Mewakili Abstraksi dan interaksi antara sebuah System dan Actor.
3.		Association	Abstraksi penghubung antar Actor dengan Use Case.
4.		Generalisasi	Menunjukkan spesialisasi actor untuk dapat berpartisipasi dengan Use Case.
5.		Include	Menunjukkan bahwa suatu Use Case seluruhnya merupakan fungsionalitas dari Use Case lainnya.
6.		Extend	Menunjukkan bahwa suatu Use Case tambahan fungsional

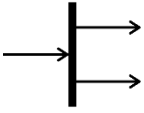
			dari Use Case lainnya.
--	--	--	------------------------

2. *Activity Diagram*

Digunakan untuk menggambarkan alur proses dan urutan aktivitas dalam suatu sistem yang menunjukkan perpindahan dari satu aktivitas ke aktivitas lainnya secara berkesinambungan. Pada *Activity Diagram* terdapat berbagai macam simbol yang digunakan untuk merepresentasikan alur dan proses kerja dalam sistem. Tabel 2.2 menunjukkan simbol-simbol *activity diagram*.

Tabel 2.2 Simbol *Activity Diagram*

No	Gambar	Simbol	Keterangan
1.		Initial node	Merupakan awal aktivitas yang akan dibuat
2.		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi
3.		Decision	Menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.

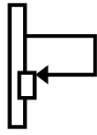
5.		Fork Join node	Digunakan untuk menunjukan kegiatan yang dilakukan secara paralel untuk dari dua kegiatan paralel jadi satu.
6.		Fork node	Satu aliran pada tahap yang berubah menjadi beberapa aliran.
7.		Activity Final node	Merupakan akhir aktivitas yang telah dibuat.

3. *Sequence Diagram*

Digunakan untuk menggambarkan interaksi antar objek didalam dan disekitar sistem yang berupa *message* yang digambarkan berdasarkan waktu. Di dalam *Sequence Diagram* terdapat beberapa simbol yang digunakan untuk menunjukan alur komunikasi pesan antar objek dalam sistem. Tabel 2.1 menunjukan simbol-simbol *sequence diagram*.

Tabel 2.3 Simbol *Sequence Diagram*

No	Gambar	Simbol	Keterangan
1.		Entity Class	Menggambarkan awal dari sebuah sistem, menjadi dasar penyusunan

			basisdata.
2.		Boundary Class	Menggambarkan komunikasi antara sistem seperti Form.
3.		Control Class	Penghubung antara boundary dengan table.
4.		Recursive	Mengirimkan pesan yang akan dikirim ke dirinya sendiri
5.		Activation	Menghubungkan proses durasi aktivasi sebuah proses
6.		Life Line	Menghubungkan tiap-tiap objek yang mana terdapat activation
7.		Message	Sebagai pengirim pesan ke tiap-tiap Class

4. Class Diagram

Digunakan untuk menggambaran struktur serta deskripsi dari *class*, *package* dan objek dalam suatu sistem, seperti hubungan pewarisan, asosiasi dan lainnya. Di dalam *Class Diagram* terdapat beberapa simbol yang digunakan untuk mempresentasikan xstruktur data dalam sistem. Tabel 2.1 menunjukkan simbol-simbol *class diagram*.

Tabel 2.4 Simbol Class Diagram

No	Gambar	Simbol	Keterangan
1.		Class	Menggambarkan kumpulan objek-objek dengan atribut dan operasi yang sama
2.		Nary Association	Sebagai upaya menghindari asosiasi dengan lebih dari 2 objek
3.		Collaboration	Urutan aksi-aksi yang ditampilkan oleh sistem yang menghasilkan suatu hasil yang tertukar bagi suatu actor
4.		Generalization	Menggambarkan dimana hubungan objek descendent berperilaku dan struktur data objek yang di atasnya objek induk ancestor
5.		Realization	Operasi yang benar-benar dilakukan oleh suatu objek