

BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait

Penelitian terkait sistem informasi peminjaman alat dan pengelolaan inventaris telah banyak dilakukan untuk mengatasi permasalahan pencatatan manual yang kurang efisien. Salah satu penelitian yang dilakukan oleh Adithya et al. (2025) mengembangkan sistem informasi inventaris barang berbasis web untuk meningkatkan akurasi pengelolaan data inventaris. Penelitian tersebut bertujuan mengurangi kesalahan pencatatan serta mempermudah proses peminjaman dan pengembalian barang melalui sistem yang terkomputerisasi. Metode pengembangan yang digunakan adalah Waterfall, sehingga proses pengembangan sistem dilakukan secara terstruktur mulai dari analisis kebutuhan hingga implementasi [3].

Penelitian yang dilakukan oleh Halifah & Subrata (2025) menunjukkan bahwa penerapan sistem informasi berbasis web mampu meningkatkan efektivitas pengelolaan inventaris dan transaksi peminjaman. Sistem yang dikembangkan berhasil mengurangi kesalahan pencatatan, mempercepat proses administrasi, serta memudahkan pengelolaan data inventaris. Penelitian tersebut juga menekankan pentingnya analisis kebutuhan dan perancangan sistem yang terstruktur agar aplikasi yang dihasilkan sesuai dengan kebutuhan pengguna [5].

Selanjutnya, penelitian oleh Desfira Ayu Namira, Amroni, dan Hartiwi (2023) membahas pemanfaatan teknologi informasi dalam meningkatkan kualitas pengelolaan data di lingkungan sekolah. Hasil penelitian menunjukkan bahwa sistem informasi berbasis web mampu meningkatkan kecepatan akses informasi, ketepatan pengolahan data, serta efisiensi proses administrasi. Temuan tersebut menjadi salah satu dasar dalam pengembangan sistem informasi inventaris dan peminjaman alat berbasis website [6].

Berdasarkan penelitian-penelitian tersebut, dapat disimpulkan bahwa penerapan sistem informasi berbasis web mampu meningkatkan efektivitas pengelolaan inventaris, mempermudah proses peminjaman dan pengembalian alat, serta meningkatkan akurasi penyimpanan data. Oleh karena itu, penelitian ini mengembangkan Sistem Informasi Peminjaman Alat Bengkel Berbasis Website menggunakan Framework Laravel di SMK Ma'arif Talang. Framework Laravel dipilih karena menerapkan arsitektur Model-View-Controller (MVC) yang mendukung pengembangan aplikasi secara terstruktur, aman, dan mudah dikembangkan. Selain itu, sistem yang dibangun menerapkan Role-Based Access Control (RBAC) dengan tiga jenis pengguna, yaitu SuperAdmin, Admin, dan Siswa, sehingga pengelolaan inventaris, transaksi peminjaman, dan hak akses pengguna dapat dilakukan secara lebih efektif sesuai kebutuhan operasional bengkel.

2.2 Landasan Teori

Berbagai studi menunjukkan bahwa pembuatan *website* bisa dilakukan dengan bantuan *framework modern* yang dibuat untuk mempercepat proses pembuatan sistem serta meningkatkan keamanan dan struktur aplikasi. Salah satu *framework* PHP yang sering digunakan adalah Laravel. Laravel mendukung arsitektur MVC, memiliki fitur ORM *Eloquent* yang membuat pengelolaan *database* lebih mudah, dan juga menyediakan sistem *routing* dan *middleware* yang sangat baik.[7].

Framework lain seperti CodeIgniter dan Django juga terbukti berguna dalam membuat sistem informasi berbasis *web* karena mereka menyediakan komponen yang siap digunakan dan standar pengembangan yang lebih teratur. Selain itu, untuk membuat *website*, kita perlu *database* sebagai tempat menyimpan data yang teratur. MySQL, Laragon, dan PostgreSQL adalah *database* relasional yang sering digunakan karena stabil, cepat, dan mudah dihubungkan dengan berbagai *framework web*[8].

Pemilihan *framework* dan *database* yang baik sangat berpengaruh terhadap kualitas sistem informasi yang dibuat, khususnya dalam hal kinerja, keamanan, dan kemudahan dalam mengelola data [9].

2.2.1 Website

Website adalah kumpulan halaman yang dipakai untuk memberikan berbagai macam informasi, seperti tulisan, foto, animasi, suara, atau semua itu sekaligus. Situs *web* bisa dibuat dengan cara yang tetap sama atau bisa berubah-ubah, tergantung apa yang

diperlukan. Untuk membuka *website*, orang memerlukan koneksi internet sebagai jalur penghubung [10].

2.2.2 Framework

Framework merupakan kerangka kerja yang berisi kumpulan pustaka, komponen, serta kode dasar yang disediakan untuk mempermudah dan mempercepat proses pengembangan aplikasi. Dengan adanya *framework*, pengembang tidak perlu membangun sistem dari awal karena telah tersedia struktur dan fungsionalitas utama yang dapat langsung dimanfaatkan. Pada penelitian ini, *framework* yang digunakan adalah Laravel, yaitu *framework* PHP berbasis arsitektur *Model-View-Controller* (MVC) yang membantu menciptakan aplikasi dengan struktur yang lebih terorganisir melalui pemisahan logika bisnis dan tampilan. Laravel dipilih karena menyediakan berbagai fitur bawaan seperti routing, middleware, dan Blade templating engine yang mendukung proses pengembangan sistem informasi secara efisien. Dari sisi keamanan, Laravel memiliki perlindungan terhadap serangan seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, dan *Cross-Site Request Forgery (CSRF)*. Selain itu, Laravel dilengkapi *Eloquent ORM* untuk mempermudah pengelolaan database serta fitur migrasi yang memungkinkan perubahan struktur data dilakukan tanpa mengganggu data yang telah tersimpan. Dalam pembangunan sistem informasi peminjaman alat bengkel dan buku perpustakaan ini, Laravel digunakan untuk mengatur proses

peminjaman, pengembalian, manajemen stok, serta autentikasi Admin dan pengguna, sehingga sistem yang dihasilkan lebih aman, terstruktur, dan mudah dikembangkan.

2.2.3 Laragon



Laragon

Gambar 1. 1 Logo Laragon

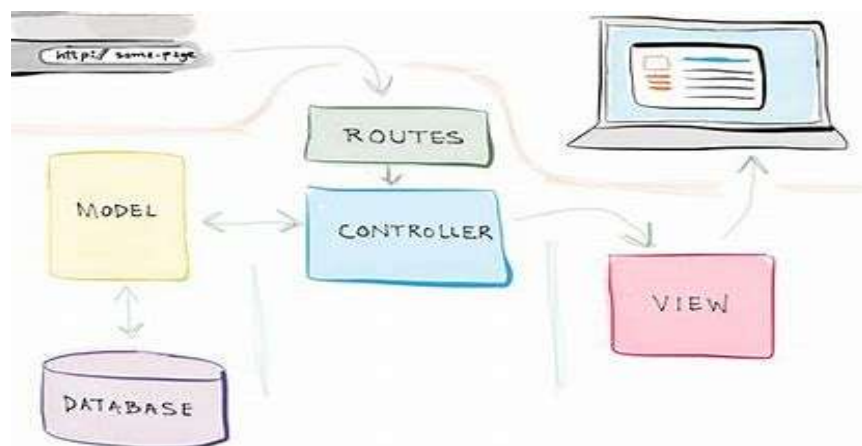
Laragon bukan database, tetapi local development environment yang menyediakan paket lengkap untuk menjalankan server web secara lokal. Laragon mendukung *MySQL*, *PHP*, *Apache*, *Nginx*, dan *Node.js*, serta memiliki fitur auto-virtual host dan kecepatan tinggi dalam menjalankan proyek web. Laragon sering digunakan untuk pengembangan aplikasi Laravel karena instalasinya mudah dan *performanya* ringan [10].

2.2.4 Laravel



Gambar 1. 2 Logo Laravel

Laravel adalah *framework* PHP yang menggunakan arsitektur *MVC (Model–View–Controller)* untuk mempermudah proses pengembangan aplikasi web. Laravel menyediakan fitur seperti *Eloquent ORM*, *Blade Template Engine*, *routing yang fleksibel*, *middleware*, serta dukungan *migration* untuk mengelola struktur *database*. Laravel didesain agar pengembangan aplikasi menjadi lebih cepat, aman, dan terstruktur sehingga banyak digunakan dalam pembangunan sistem informasi modern [8].



Gambar 1. 3 Konsep Laravel

2.2.5 MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang banyak digunakan dalam aplikasi web. MySQL mendukung bahasa SQL untuk mengelola data dan terkenal karena kecepatan, stabilitas, serta kompatibilitasnya dengan berbagai *framework* seperti Laravel dan CodeIgniter. MySQL dinilai efisien untuk sistem informasi yang membutuhkan pengolahan data cepat dan struktur tabel yang terorganisasi [10].

Pada penelitian ini, website sistem informasi peminjaman alat bengkel dan buku perpustakaan akan dibangun menggunakan kombinasi Laragon sebagai lingkungan pengembangan, Laravel sebagai *framework* utama untuk merancang aplikasi web, serta MySQL sebagai basis data yang menyimpan seluruh informasi diperlukan. Pemilihan teknologi tersebut dilakukan karena ketiganya mampu mendukung pengembangan sistem yang terstruktur, aman, mudah dikelola, serta sesuai dengan kebutuhan operasional sekolah [11].



Gambar 1. 4 Logo MySQL

2.2.6 Database

Database merupakan kumpulan data yang saling berhubungan dan disimpan secara terorganisir dengan tingkat redundansi yang terkontrol agar dapat mendukung satu atau lebih aplikasi secara optimal. Penyimpanan data dirancang agar tidak bergantung pada program yang digunakan untuk mengaksesnya, sehingga proses penambahan, pengubahan, maupun pengambilan data dapat dilakukan dengan mudah dan efisien [12].

2.2.7 Visual Studio Code

Visual Studio Code adalah editor teks yang ringan dan kuat untuk pengembangan perangkat lunak. VS Code dikembangkan oleh *Microsoft* dan mendukung berbagai bahasa pemrograman, seperti *JavaScript*, *Python*, *PHP*, *HTML*, *CSS*, *C++*, dan banyak lagi. VS Code dirancang untuk menjadi editor yang cepat, dapat diperluas, dan mudah digunakan untuk menulis kode.



Gambar 1. 5 Visual Studio Code

2.2.8 Taildwind CSS

Tailwind CSS adalah editor teks yang ringan dan kuat untuk pengembangan perangkat lunak. VS Code dikembangkan oleh *Microsoft* dan mendukung berbagai bahasa pemrograman, seperti

JavaScript, Python, PHP, HTML, CSS, C++, dan banyak lagi. VS Code dirancang untuk menjadi editor yang cepat, dapat diperluas, dan mudah digunakan untuk menulis kode.



Gambar 1. 6 Tailwindcss

2.2.9 Java Script

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web interaktif. Bahasa ini memungkinkan pengembang untuk menambahkan elemen-elemen dinamis seperti animasi, *form* validasi, manipulasi elemen HTML, dan interaksi pengguna lainnya ke dalam situs web. JavaScript biasanya dijalankan di sisi klien (browser), namun dapat juga digunakan di sisi server melalui lingkungan seperti Node.js.



Gambar 1. 7 Java Script

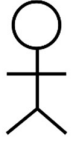
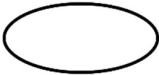
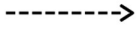
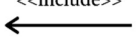
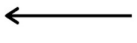
2.2.10 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual yang digunakan untuk memvisualisasikan, merancang, mendokumentasikan, dan menjelaskan struktur serta perilaku sistem perangkat lunak secara terstandarisasi [13]. Dengan UML, perancang sistem dapat membuat representasi grafis dari komponen sistem seperti aktor, proses, alur kerja, serta hubungan antar objek sehingga memudahkan komunikasi antara analis, pengembang, dan pemangku kepentingan. UML mencakup beberapa jenis *diagram* seperti *Use Case Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram* yang membantu memodelkan berbagai aspek sistem secara terstruktur dan sistematis. Penggunaan UML dalam perancangan sistem informasi terbukti efektif untuk memastikan bahwa model yang dibangun telah sesuai dengan kebutuhan pengguna dan siap diimplementasikan ke dalam kode program

1. *Use case diagram*

Merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat aktor yang merupakan sebuah gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem.

Tabel 2. 1 Simbol *Use Case Diagram*

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

2. *Class diagram*

Merupakan gambaran struktur dan deskripsi dari *class*, *package* dan objek yang saling berhubungan seperti diantaranya pewarisan, asosiasi dan lainnya.

Tabel 2. 2 Simbol *Class Diagram*

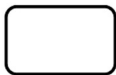
No	Gambar	Simbol	Keterangan
1.		<i>Class</i>	Menggambarkan kumpulan objek-objek dengan atribut dan operasi yang sama.

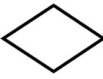
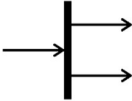
No	Gambar	Simbol	Keterangan
2.		<i>Nary Association</i>	Sebagai upaya menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Collaboration</i>	Urutan aksi-aksi yang ditampilkan oleh sistem yang menghasilkan suatu hasil yang tertukar bagi suatu <i>actor</i> .
4.		<i>Generalization</i>	Menggambarkan dimana hubungan objek <i>descendent</i> berperilaku dan struktur data objek yang di atasnya objek induk <i>ancestor</i> .
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang tidak mandiri.

3. Activity diagram

Sebuah *diagram* aktivitas UML menggambarkan perilaku dinamis dari suatu sistem atau bagian dari sistem melalui aliran kontrol antara aksi yang dilakukan sistem.

Tabel 2. 3 Simbol *Activity Diagram*

No	Gambar	Simbol	Keterangan
1.		<i>Initial node</i>	Merupakan awal aktivitas yang akan dibuat.
2.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas

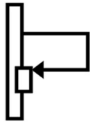
No	Gambar	Simbol	Keterangan
			antarmuka saling berinteraksi.
3.		<i>Decicion</i>	Menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.
4.		<i>Fork Join node</i>	Digunakan untuk menunjukan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
5.		<i>Fork node</i>	Satu aliran pada tahap yang berubah menjadi beberapa aliran.
6.		<i>Activity Final node</i>	Merupakan akhir aktivitas yang telah dibuat.

4. *Sequence diagram*

Menggambarkan interaksi antar objek didalam dan di sekitar sistem (termasuk pengguna, *display* dan sebagainya) berupa message yang digambarkan terhadap waktu. *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu.

Tabel 2. 4 Simbol *Sequence Diagram*

No	Gambar	Simbol	Keterangan
1.		<i>Entity Class</i>	Menggambarkan awal dari sebuah sistem, menjadi dasar penyusunan basisdata.

No	Gambar	Simbol	Keterangan
2.		<i>Boundary Class</i>	Menggambarkan komunikasi antara sistem seperti <i>Form</i> .
3.		<i>Control Class</i>	Penghubung antara <i>boundary</i> dengan <i>table</i> .
4.		<i>Recursive</i>	Mengirimkan pesan yang akan dikirim ke dirinya sendiri.
5.		<i>Activation</i>	Menghubungkan proses durasi aktivasi sebuah proses.
6.		<i>Life Line</i>	Menghubungkan tiap-tiap objek yang mana terdapat <i>activation</i> .
7.		<i>Message</i>	Sebagai pengirim pesan ke tiap-tiap <i>Class</i> .