

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2.1 di bawah ini adalah tabel mengenai penelitian terdahulu yang dapat dilihat.

Tabel 2. 1 Penelitian Terdahulu

1	Judul Jurnal	Sistem Informasi Penjualan Berbasis Web untuk Toko Ritel
	Nama Peneliti	Angga Wahyu Surya Putra, Suprianto
	Tahun	2024
	Nama Jurnal	<i>Indonesian Journal of Applied Technology</i>
	Volume dan Halaman	1(2):1-13
	Masalah Penelitian	Toko ritel yang menjadi fokus penelitian ini masih mengelola transaksi penjualan dan pencatatan inventaris secara manual. Hal ini menyebabkan sering terjadinya kesalahan dalam pengumpulan dan pencatatan data, serta kesulitan dalam melakukan <i>monitoring</i> stok barang dan transaksi. Untuk itu, diperlukan sistem informasi berbasis <i>web</i> yang dapat meningkatkan akurasi dan efisiensi dalam pengelolaan data penjualan dan inventaris.
	Metode Penelitian	Metode yang digunakan dalam penelitian ini adalah pengembangan aplikasi berbasis <i>web</i> dengan mengintegrasikan berbagai teknologi seperti <i>HTML</i> , <i>CSS</i> , <i>JavaScript</i> , <i>PHP</i> , dan <i>MySQL</i> untuk <i>backend database</i> . Selain itu, mereka menggunakan <i>Notepad++</i> sebagai <i>editor</i> teks dan <i>XAMPP</i> sebagai lingkungan server lokal. <i>Framework Bootstrap</i> juga digunakan untuk memudahkan responsivitas dan <i>styling</i> pada tampilan <i>frontend</i> .
	Hasil Penelitian	Pengembangan sistem informasi penjualan berbasis <i>web</i> yang berhasil diimplementasikan di toko ritel. Sistem ini mempermudah proses pencatatan transaksi,

		<i>monitoring</i> stok barang, serta pembuatan laporan penjualan secara lebih akurat dan efisien. Selain itu, sistem ini juga mengurangi kesalahan yang sering terjadi pada pengelolaan data manual dan memberikan kemudahan bagi pegawai dalam mengelola data penjualan.
2	Judul Jurnal	Sistem Informasi Manajemen Bengkel Modul <i>Point of Sales</i> Berbasis <i>Web</i>
	Nama Peneliti	Gede Angga Kusuma Putra, A.A. KOMPIANG OKA SUDANA, I MADE SUNIA RAHARJA
	Tahun	2021
	Nama Jurnal	Jurnal Ilmiah Teknologi dan Komputer
	Volume dan Halaman	2(3):1-12
	Masalah Penelitian	Bengkel kendaraan masih banyak yang menggunakan proses bisnis konvensional, terutama pada bagian kasir yang menangani transaksi layanan dan penjualan suku cadang. Proses ini rentan terhadap risiko kerusakan dan kehilangan data, serta kurang efisien. Oleh karena itu, diperlukan sistem yang terkomputerisasi untuk mempermudah proses pembayaran dan pencatatan transaksi di bengkel kendaraan.
	Metode Penelitian	Metodologi yang digunakan adalah metode <i>waterfall</i> , yang mencakup analisis kebutuhan, desain sistem dan perangkat lunak, implementasi, pengujian sistem, serta operasi dan pemeliharaan. Dalam tahap pengujian, digunakan Pengujian <i>Black-Box</i> untuk mengevaluasi fitur-fitur sistem berdasarkan input dan output tanpa mempertimbangkan struktur internal sistem. <i>Tools</i> yang digunakan dalam pengembangan sistem ini meliputi <i>Visual Studio Code</i> , dengan bahasa pemrograman <i>HTML</i> , <i>CSS</i> , <i>PHP</i> , dan <i>JavaScript</i> , serta <i>framework Laravel</i> untuk membangun dan mengelola database menggunakan <i>MySQL</i> .
	Hasil Penelitian	Pengembangan Sistem Informasi Manajemen Bengkel Modul <i>Point of Sales</i> berbasis <i>web</i> yang memudahkan proses pembayaran layanan dan penjualan suku cadang di bengkel kendaraan. Sistem ini juga mampu menghasilkan laporan harian

		untuk pemilik bengkel, sehingga memudahkan mereka dalam pengambilan keputusan yang cepat, tepat, dan akurat. Pengujian sistem dengan metode <i>Black-Box</i> menunjukkan bahwa fitur-fitur yang diuji berjalan sesuai dengan yang diharapkan.
3	Judul Jurnal	Sistem Informasi <i>Point of Sales</i> Berbasis <i>Web</i> di Perusahaan Bangunan Andalas Jaya
	Nama Peneliti	Enrico Chrislie, Sigit Birowo
	Tahun	2024
	Nama Jurnal	Jurnal Informatika dan Bisnis
	Volume dan Halaman	13(1):60-71
	Masalah Penelitian	Perusahaan Bangunan Andalas Jaya mengalami masalah dalam pencatatan penjualan dan persediaan barang yang masih dilakukan secara manual menggunakan sistem nota. Hal ini menyebabkan pengelolaan data yang tidak efisien, berisiko kehilangan data, dan kurang efektif dalam memantau stok barang, sehingga membutuhkan sistem yang terkomputerisasi untuk mengatasi masalah ini.
	Metode Penelitian	Metode kualitatif dengan pendekatan observasi dan wawancara untuk mengumpulkan data yang diperlukan dalam merancang sistem informasi <i>Point of Sales</i> . Pengembangan sistem menggunakan metode <i>Waterfall</i> , yang mencakup tahapan <i>Requirement Definition, System and Software Design, Implementation and Unit Testing, Integration and System Testing</i> , serta <i>Maintenance</i> . Selama proses ini, peneliti menerapkan berbagai tools seperti <i>Visual Studio Code</i> untuk pemrograman dan <i>MySQL</i> untuk manajemen <i>database</i> , mendesain antarmuka dengan <i>UML</i> , dan mengimplementasikan sistem pada platform <i>web</i> yang menghasilkan rancangan sistem yang efektif
	Hasil Penelitian	Pengembangan Sistem Informasi <i>Point of Sales</i> berbasis <i>web</i> yang dirancang untuk mempermudah pengelolaan transaksi penjualan dan persediaan barang di Perusahaan Bangunan Andalas Jaya. Sistem ini mampu mengotomatisasi proses yang

		sebelumnya dilakukan secara manual, termasuk pencatatan penjualan, pengelolaan stok barang, dan pembuatan laporan penjualan. Pengujian sistem menggunakan metode <i>Alpha Testing</i> menunjukkan bahwa sistem berjalan sesuai dengan yang diharapkan, dan sistem <i>POS</i> ini berhasil meningkatkan efisiensi pengelolaan data dan transaksi perusahaan.
4	Judul Jurnal	Perancangan Sistem Antrian Menggunakan Metode <i>RAD</i> Berbasis <i>Web</i>
	Nama Peneliti	Dini Febrianti, Bias Yulisa Geni
	Tahun	2024
	Nama Jurnal	Jurnal Mahasiswa Teknik Informatika
	Volume dan Halaman	8(1):1021-1028
	Masalah Penelitian	PT IT Service Center (ITSC) Cabang Cideng menghadapi masalah dalam sistem pelayanan karena tidak adanya sistem antrian yang jelas. Pelanggan merasa kurang puas dengan pelayanan yang lambat, serta kesulitan membedakan jenis layanan (<i>onsite</i> dan <i>pickup</i>). Hal ini menyebabkan penumpukan antrian dan ketidakpuasan pelanggan akibat tidak adanya sistem yang terorganisir dengan baik untuk mengelola antrian layanan.
	Metode Penelitian	Menggunakan Metode <i>RAD</i> , pengembangan sistem menggunakan bahasa pemrograman <i>PHP</i> dan basis data <i>MySQL</i> , dengan kerangka kerja <i>Laravel</i> . Sistem antrian dirancang untuk membedakan antara layanan <i>pickup</i> dan <i>onsite</i> , dengan implementasi menggunakan metode antrian <i>multiple</i> . Pengujian sistem dilakukan dengan metode <i>black box</i> , yang menunjukkan bahwa sistem dapat beroperasi dengan baik dan memenuhi kebutuhan perusahaan dalam mengelola antrian layanan secara efisien.
	Hasil Penelitian	Pengembangan sistem antrian berbasis <i>web</i> yang menggunakan metode antrian <i>multiple</i> (<i>onsite</i> dan <i>pickup</i>). Sistem ini berhasil mempermudah pengelolaan antrian layanan di PT ITSC, meningkatkan efisiensi pelayanan, dan mengurangi kesalahan dalam penanganan pelanggan. Pengujian

		sistem menggunakan metode <i>black box</i> menunjukkan bahwa sistem berjalan dengan baik dan mendapatkan tanggapan positif dari pengguna. Fitur-fitur yang dikembangkan meliputi pendaftaran antrian, pembagian nomor antrian, pengelolaan data pelanggan, dan manajemen data brand serta perangkat yang dapat di-service.
5	Judul Jurnal	Rancang Bangun Sistem Informasi Perpustakaan SMAN 1 Bayan Berbasis <i>Website</i> Dengan <i>PHP MySQL</i>
	Nama Peneliti	Mely Handayani, Ahmad Zafrullah Mardiansyah, Jatradi
	Tahun	2021
	Nama Jurnal	Jurnal Begawe Teknologi Informasi
	Volume dan Halaman	2(1):1-12
	Masalah Penelitian	SMAN 1 Bayan menghadapi masalah dalam pengelolaan perpustakaan yang masih dilakukan secara manual, termasuk dalam transaksi peminjaman dan pengembalian buku serta pengelolaan data buku. Sistem manual ini menyebabkan proses pengelolaan data menjadi lambat, tidak efisien, dan kurang akurat, sehingga diperlukan solusi berupa sistem informasi berbasis web untuk meningkatkan efisiensi dan akurasi dalam pengelolaan perpustakaan.
	Metode Penelitian	Menggunakan metode <i>waterfall</i> untuk pengembangan sistem. Metode ini meliputi analisis kebutuhan, desain sistem, implementasi, pengujian, dan pemeliharaan. Proses pengembangan ini dilakukan secara sistematis, dimulai dari pengumpulan data administrasi perpustakaan dan keinginan klien yang kemudian dianalisis untuk desain aplikasi. Implementasi desain melibatkan pemrograman dan pengujian untuk memastikan sistem berfungsi sesuai kebutuhan. Pengujian sistem menggunakan kuesioner menghasilkan persentase tinggi kepuasan pengguna, menunjukkan efektivitas sistem dalam mempermudah transaksi peminjaman dan pengembalian buku serta pengelolaan data perpustakaan.

	Hasil Penelitian	Pengembangan Sistem Informasi Perpustakaan berbasis <i>web</i> yang sudah terimplementasi dengan baik di SMAN 1 Bayan. Sistem ini menyediakan berbagai fitur seperti peminjaman, pengembalian buku, pengelolaan data buku, pencatatan laporan, dan pengelolaan data <i>user</i> . Berdasarkan hasil uji coba menggunakan kuesioner, 72% responden sangat setuju dan 28% responden setuju bahwa sistem ini telah mempermudah pekerjaan petugas perpustakaan, menghasilkan laporan yang akurat, dan menunjang kinerja perpustakaan secara keseluruhan.
6	Judul Jurnal	Rancang Bangun Aplikasi Inventaris Data Barang Sekolah Berbasis Web Pada SMK Negeri 1 Tanah Abang Kabupaten Pali
	Nama Peneliti	Iwan Setiawan, Suhartini, Fajriyah
	Tahun	2-23
	Nama Jurnal	Jurnal Sistem Informasi dan Teknik Komputer
	Volume dan Halaman	8(1):61-64
	Masalah Penelitian	SMK Negeri 1 Tanah Abang menghadapi masalah dalam pengelolaan data inventaris barang yang masih dilakukan secara manual menggunakan lembaran kertas. Pendekatan ini rentan terhadap kehilangan data dan kurang efisien dalam menyimpan serta mengelola inventaris. Sistem manual ini tidak mampu memenuhi kebutuhan sekolah dalam pengelolaan data inventaris yang lebih terstruktur dan aman.
	Metode Penelitian	Metode penelitian yang digunakan adalah deskriptif kualitatif, yang mengutamakan pengumpulan data melalui observasi langsung, wawancara dengan pihak terkait, dan studi literatur. Dalam pengembangan sistem, digunakan metode <i>prototyping</i> yang melibatkan pengkodean iteratif dan pengujian berkelanjutan untuk memastikan sistem sesuai dengan kebutuhan pengguna. Untuk pengembangan aplikasi, digunakan <i>Visual Studio Code</i> sebagai lingkungan pengembangan, <i>MySQL</i> untuk manajemen <i>database</i> , dan <i>XAMPP</i> sebagai <i>server</i> lokal

		yang mendukung implementasi dan pengujian aplikasi secara langsung.
	Hasil Penelitian	Pengembangan aplikasi inventaris data barang berbasis web yang diimplementasikan di SMK Negeri 1 Tanah Abang. Aplikasi ini berhasil membantu mempermudah proses pencatatan dan pengelolaan inventaris, meningkatkan keamanan penyimpanan data, serta mengubah sistem pencatatan manual menjadi sistem komputerisasi. Dengan adanya aplikasi ini, kinerja sekolah dalam mengelola inventaris menjadi lebih efisien dan terorganisir.
7	Judul Jurnal	<i>The Development of Web-based Sales Reporting Information Systems using Rapid Application Development Method</i>
	Nama Peneliti	Suryasari, Jansen Wiratama, Ririn Ikana Desanti
	Tahun	2023
	Nama Jurnal	Ultima Infosys: Jurnal Ilmu Sistem Informasi
	Volume dan Halaman	13(2):110-116
	Masalah Penelitian	PT Artindo Pratama Sejahtera, sebuah perusahaan yang bergerak di bidang produksi cat, menghadapi masalah dalam proses pelaporan penjualan. Pelaporan yang dilakukan oleh salesperson ke bagian admin gudang masih menggunakan aplikasi pesan singkat atau chat, sehingga data sering menumpuk dan tidak tercatat dengan baik. Hal ini menyebabkan proses pelaporan penjualan menjadi kurang efisien dan berpotensi menimbulkan kesalahan.
	Metode Penelitian	Penelitian ini menggunakan metode <i>Rapid Application Development (RAD)</i> untuk mengembangkan Sistem Informasi Pelaporan Penjualan berbasis <i>web</i> . <i>RAD</i> dipilih karena memungkinkan pengembangan yang lebih cepat dan hasil yang berkualitas dengan fokus pada setiap kebutuhan pengembangan secara bertahap. Prosesnya meliputi perencanaan kebutuhan, desain pengguna, konstruksi, dan <i>cutover</i> , dengan keterlibatan pengguna yang aktif dalam setiap tahapan. Alat yang digunakan

		termasuk <i>UML Diagram</i> untuk desain dan <i>PHP</i> untuk pengkodean, dengan dukungan dari <i>Figma</i> untuk desain antarmuka sistem. Uji penerimaan pengguna (<i>User Acceptance Test - UAT</i>) dengan model pengujian <i>black box</i> digunakan untuk memvalidasi fungsionalitas sistem sebelum diimplementasikan sepenuhnya
	Hasil Penelitian	Sistem ini berhasil diujicobakan dengan mendapatkan skor rata-rata 82,5% dalam <i>UAT</i> , menunjukkan bahwa sistem tersebut mampu memenuhi kebutuhan perusahaan dan dapat meningkatkan efisiensi dalam pelaporan penjualan. Sistem yang dibangun juga menyediakan fitur otentikasi dan hak akses pengguna yang lebih aman dibandingkan sistem sebelumnya.
8	Judul Jurnal	<i>Enhancing sustainability integration in Sustainable Enterprise Resource Planning (S-ERP) system: Application of Transaction Cost Theory and case study analysis</i>
	Nama Peneliti	Kushal Anjaria
	Tahun	2024
	Nama Jurnal	<i>International Journal of Information Management Data Insights</i>
	Volume dan Halaman	4(2):1-23
	Masalah Penelitian	Ketidakmampuan sistem <i>Enterprise Resource Planning</i> yang berorientasi keberlanjutan (<i>S-ERP</i>) untuk secara holistik mengintegrasikan metrik keberlanjutan ke dalam semua fungsi bisnis. Hal ini menciptakan kesenjangan dalam memahami dampak lingkungan dari proses bisnis secara real-time dan menghambat pengambilan keputusan yang optimal, serta menghadapi tantangan dalam menyesuaikan sistem dengan peraturan keberlanjutan yang terus berubah.
	Metode Penelitian	Pendekatan <i>Transaction Cost Theory (TCT)</i> . Studi kasus dilakukan untuk mengevaluasi proses bisnis berkelanjutan melalui analisis <i>input-output</i> , sementara <i>TCT</i> digunakan untuk mengukur dan mengoptimalkan biaya transaksi yang terkait dengan integrasi keberlanjutan dalam sistem <i>Sustainable Enterprise Resource Planning (S-ERP)</i> .

	Hasil Penelitian	Penerapan <i>Transaction Cost Theory (TCT)</i> dalam sistem <i>Sustainable Enterprise Resource Planning (S-ERP)</i> berhasil meningkatkan integrasi metrik keberlanjutan pada berbagai proses bisnis. Studi kasus yang dilakukan menunjukkan bahwa penggunaan matriks keberlanjutan berbasis <i>input-output</i> dapat mengidentifikasi area untuk pengurangan biaya transaksi, meningkatkan efisiensi sumber daya, dan mengurangi dampak lingkungan. Pendekatan ini juga membantu organisasi dalam menyusun strategi keberlanjutan yang lebih komprehensif dan sesuai dengan kebutuhan industri, seperti desain produk berkelanjutan dan metode konstruksi modular.
9	Judul Jurnal	<i>Investigating ERP System Customization: A Focus on Cloud-ERP</i>
	Nama Peneliti	Hans Fredrik Hansen, Moutaz Haddara, Marius Langseth
	Tahun	2024
	Nama Jurnal	<i>Procedia Computer Science</i>
	Volume dan Halaman	219:915-923
	Masalah Penelitian	Keterbatasan kustomisasi pada sistem <i>Cloud-ERP</i> yang sering kali dianggap sebagai penghalang utama dalam adopsi sistem tersebut oleh organisasi. Kustomisasi diperlukan untuk memenuhi kebutuhan spesifik proses bisnis organisasi, tetapi banyak penyedia <i>Cloud-ERP</i> membatasi opsi kustomisasi, yang dapat berdampak pada kemampuan organisasi untuk memperoleh keunggulan kompetitif.
	Metode Penelitian	Review literatur sistematis yang mencakup artikel-artikel yang diterbitkan selama 12 tahun terakhir (2010–2022). Artikel yang direview dipilih berdasarkan relevansi dengan topik kustomisasi pada <i>Cloud-ERP</i> , dengan kata kunci seperti " <i>cloud</i> ," " <i>ERP</i> ," dan " <i>customization</i> ." Proses seleksi melibatkan pencarian di <i>database</i> seperti <i>Google Scholar</i> , <i>IEEE Xplore</i> , <i>Web of Science</i> , dan <i>ACM</i> , serta telaah referensi dari artikel yang direview untuk

		memastikan cakupan penelitian yang komprehensif.
	Hasil Penelitian	Meskipun memberikan fleksibilitas untuk memenuhi kebutuhan bisnis spesifik dan meningkatkan keunggulan kompetitif, kustomisasi juga menambah kompleksitas dalam implementasi dan pembaruan sistem. Sebagian besar penyedia <i>Cloud-ERP</i> membatasi opsi kustomisasi, terutama untuk klien kecil hingga menengah, yang menciptakan hambatan adopsi. Namun, bagi organisasi besar, kustomisasi yang tepat dapat mendukung efisiensi operasional, pengambilan keputusan manajerial, dan inovasi strategis.
10	Judul Jurnal	<i>Customizing ERP-systems: A framework to support the decision-making process</i>
	Nama Peneliti	Eli Hustad, Johan Stensholt
	Tahun	2023
	Nama Jurnal	Procedia Computer Science
	Volume dan Halaman	219:789-796
	Masalah Penelitian	Bagaimana perusahaan dapat mengambil keputusan yang tepat terkait kustomisasi sistem <i>ERP</i> . Meskipun kustomisasi sering kali diperlukan untuk memenuhi kebutuhan spesifik perusahaan, hal ini dapat meningkatkan kompleksitas, biaya, dan risiko dalam implementasi serta pemeliharaan sistem. Selain itu, banyak perusahaan menghadapi kesulitan dalam menentukan apakah lebih baik menyesuaikan sistem <i>ERP</i> dengan kebutuhan mereka atau mengadaptasi proses bisnis mereka agar sesuai dengan standar sistem <i>ERP</i> . Kurangnya pedoman yang jelas dan struktur pengambilan keputusan sering mengakibatkan kustomisasi yang tidak perlu, yang kemudian menjadi beban di masa depan.
	Metode Penelitian	Studi kasus interpretatif yang melibatkan 10 wawancara semi-terstruktur dengan pakar <i>ERP</i> , baik dari pihak konsultan maupun pelanggan. Studi ini juga menggunakan dokumen sekunder seperti spesifikasi kebutuhan dan dokumen proyek untuk mendukung analisis. Fokusnya adalah pada

		implementasi kustomisasi sistem <i>ERP</i> di organisasi menengah hingga besar. Wawancara dilakukan untuk menggali pengalaman langsung terkait tantangan, risiko, dan keputusan kustomisasi <i>ERP</i> .
	Hasil Penelitian	Kustomisasi <i>ERP</i> dapat memberikan manfaat seperti peningkatan fungsi, kepuasan pengguna, dan mendukung proses bisnis yang unik, tetapi juga membawa risiko signifikan, termasuk peningkatan biaya, kompleksitas, dan tantangan dalam pemeliharaan serta pembaruan sistem. Studi ini menghasilkan sebuah kerangka kerja untuk mendukung pengambilan keputusan kustomisasi <i>ERP</i> , yang menekankan pentingnya analisis risiko, evaluasi biaya dan manfaat, serta pemahaman mendalam tentang fitur standar sistem sebelum melakukan kustomisasi.

Analisis terhadap jurnal-jurnal yang telah dipelajari menunjukkan keterkaitan yang signifikan dengan topik skripsi dalam aspek metodologi, jurnal terkait penggunaan metode prototipe dan *Rapid Application Development (RAD)* pada pengembangan sistem, seperti yang diterapkan pada sistem antrian berbasis web dan sistem informasi perpustakaan, menegaskan efektivitas pendekatan keberlanjutan. Metode ini memungkinkan pengembang untuk menghadirkan prototipe awal yang dapat diuji langsung oleh pengguna, sehingga kebutuhan spesifik dapat diidentifikasi lebih cepat dan solusi yang sesuai dapat diterapkan secara optimal [25], [26], [27]. Selanjutnya, pengujian *blackbox* yang dibahas pada beberapa jurnal, seperti sistem informasi manajemen bengkel dan *point of sales*, menunjukkan bahwa metode ini sangat cocok untuk menguji fungsi sistem tanpa perlu memeriksa detail kode internal. Dalam konteks *ERP* akuntansi, pengujian *blackbox* menjadi langkah penting untuk memastikan bahwa setiap fitur, mulai dari pencatatan jurnal hingga pembuatan laporan keuangan, berjalan sesuai kebutuhan pengguna [28], [29]. Jurnal tentang *ERP* dan *Cloud-ERP* mengangkat isu pentingnya kustomisasi sistem, yang sering kali diperlukan untuk memenuhi kebutuhan spesifik

organisasi. Namun, kustomisasi juga membawa risiko, seperti kesulitan dalam pembaruan sistem di masa depan. Hal ini relevan dengan *ERP* akuntansi berbasis *web*, di mana fleksibilitas dan kemampuan adaptasi terhadap kebutuhan bisnis menjadi aspek krusial dalam pengembangannya [30], [31]. Selain itu, beberapa jurnal menunjukkan bagaimana digitalisasi proses, seperti pada sistem informasi perpustakaan SMAN 1 Bayan dan sistem penjualan berbasis *web*, dapat meningkatkan efisiensi operasional dan akurasi data. Sistem berbasis *web* memungkinkan pengelolaan data yang lebih cepat, mengurangi kesalahan manual, serta menghasilkan laporan yang akurat. Aspek ini sejalan dengan tujuan *ERP* akuntansi yang dirancang untuk menyederhanakan dan mengotomasi pengelolaan data keuangan [27], [32], [33]. Jurnal tentang *Sustainable ERP (S-ERP)* menambahkan dimensi baru berupa integrasi keberlanjutan ke dalam *ERP*. Walaupun fokus utama skripsi pada modul akuntansi, pengelolaan data keberlanjutan dapat menjadi fitur tambahan yang mendukung pengambilan keputusan strategis berbasis data [34]. Terakhir, penerapan teknologi informasi berbasis *web*, seperti yang diuraikan pada jurnal sistem perpustakaan dan *point of sales*, memperkuat argumen bahwa sistem berbasis *web* menawarkan fleksibilitas tinggi, akses mudah, serta pengelolaan data yang efisien. Hal ini memberikan dasar kuat untuk pengembangan *ERP* akuntansi berbasis *web*, di mana aksesibilitas dan integrasi data *real-time* menjadi kebutuhan utama pengguna sistem [27], [32]. Banyak penelitian sebelumnya berfokus pada penerapan *ERP* untuk tujuan spesifik, seperti *Point of Sales*, namun belum banyak yang secara mendalam mengintegrasikan modul akuntansi dalam satu *platform* berbasis *web* dengan pendekatan prototipe. Pada skripsi ini mengadopsi pendekatan prototipe yang memungkinkan iterasi berkelanjutan berdasarkan umpan balik user, serta metode pengujian *blackbox* untuk memastikan keandalan modul akuntansi tanpa harus mengakses struktur internal kode. Kebaruannya terletak pada integrasi modul akuntansi berbasis *web* yang dirancang untuk mendukung efisiensi operasional sekaligus keakuratan data untuk pengambilan keputusan yang lebih baik.

2.2 Teori tentang Skripsi

2.2.1 ERP

Enterprise Resource Planning (ERP) adalah sistem perangkat lunak terintegrasi yang dirancang untuk mengelola dan mengotomatisasi berbagai proses bisnis dalam organisasi. *ERP* berfungsi sebagai platform sentral yang mengintegrasikan fungsi-fungsi utama seperti keuangan, sumber daya manusia, produksi, logistik, penjualan, dan layanan pelanggan ke dalam satu sistem terpadu. Dengan integrasi ini, data yang dihasilkan oleh satu departemen dapat diakses dan dimanfaatkan oleh departemen lain secara *real-time*, yang membantu organisasi dalam mengurangi duplikasi data dan meningkatkan efisiensi operasional [22]. Selain itu, *ERP* juga memberikan visibilitas yang lebih baik terhadap keseluruhan proses bisnis, memungkinkan pengambilan keputusan yang lebih cepat dan tepat [35].

Meskipun implementasi *ERP* sering kali menimbulkan tantangan karena memerlukan perubahan besar dalam proses bisnis dan adaptasi organisasi terhadap sistem baru, manfaat yang diperoleh sangat signifikan. Peningkatan produktivitas, pengambilan keputusan yang lebih cepat dan tepat, serta pengurangan biaya operasional adalah beberapa keuntungan utama dari penerapan *ERP* [36], [37]. Selain itu, *ERP* memungkinkan perusahaan untuk lebih fleksibel dalam menanggapi perubahan pasar dan kebutuhan bisnis, yang pada akhirnya dapat memberikan keunggulan kompetitif dalam lingkungan bisnis yang semakin dinamis [38], [39].

2.2.2 Accounting Management

Manajemen akuntansi adalah sebuah cabang dari akuntansi yang berfokus pada proses pengumpulan, analisis, interpretasi, dan presentasi informasi keuangan untuk membantu manajer dalam membuat keputusan yang efektif dalam manajemen dan kontrol operasional organisasi [40], [41]. Aspek kunci dari manajemen akuntansi adalah perencanaan dan penganggaran, dimana manajer menggunakan informasi akuntansi untuk membuat rencana strategis dan operasional jangka pendek maupun panjang [40], [42].

Perbedaan utama antara manajemen akuntansi dengan akuntansi keuangan terletak pada tujuan dan fokusnya. Akuntansi keuangan lebih difokuskan pada penyusunan laporan keuangan yang ditujukan kepada pihak eksternal seperti *investor*, *kreditor*, dan *regulator*. Sebaliknya, manajemen akuntansi berfokus pada penyediaan informasi yang digunakan secara internal oleh pihak manajemen. Oleh sebab itu, manajemen akuntansi kerap melibatkan penggunaan berbagai teknik seperti analisis biaya-volume-laba, penganggaran, dan analisis varians, yang berfungsi untuk membantu pengambilan keputusan operasional sehari-hari [40]. Informasi ini kemudian dapat dimanfaatkan untuk mengidentifikasi area yang memerlukan perbaikan, mengoptimalkan efisiensi operasional, dan mengarahkan sumber daya ke area yang paling produktif [42].

Keberadaan manajemen akuntansi yang efektif dalam sebuah organisasi sangat penting, karena informasi yang dihasilkannya berperan krusial dalam menentukan arah strategi bisnis serta kebijakan operasional perusahaan. Dengan sistem manajemen akuntansi yang baik, manajemen dapat merespons perubahan pasar dengan lebih proaktif, mengelola risiko dengan lebih baik, serta memaksimalkan keuntungan perusahaan [41]. Selain itu, manajemen akuntansi juga mendukung pencapaian tujuan jangka panjang perusahaan melalui perencanaan keuangan yang matang, pengendalian biaya yang efektif, serta evaluasi kinerja yang berkelanjutan [40], [42].

2.2.3 Efektif

Efektivitas mengacu pada kemampuan untuk mencapai tujuan atau hasil yang diinginkan dengan memanfaatkan sumber daya yang tersedia secara optimal. Dalam konteks manajemen dan organisasi, efektivitas diukur berdasarkan seberapa baik suatu tindakan atau proses dapat memenuhi target yang telah ditentukan. Suatu proses dianggap efektif jika berhasil mencapai hasil yang diharapkan dalam waktu yang ditetapkan, dengan penggunaan sumber daya yang minimal namun tepat. Oleh karena itu, efektivitas berfokus pada output dan hasil akhir, menjadi salah satu faktor penting dalam penilaian kinerja individu maupun organisasi [43], [44].

Selain itu, efektivitas juga berkaitan erat dengan kemampuan beradaptasi terhadap perubahan dan tantangan dalam mencapai hasil yang optimal. Dalam dunia bisnis yang dinamis, perusahaan yang efektif adalah mereka yang mampu menyesuaikan proses operasionalnya sesuai dengan perkembangan pasar, teknologi, dan kebutuhan pelanggan. Peningkatan efektivitas sering kali diperoleh melalui penerapan strategi berbasis data, inovasi, dan pengelolaan sumber daya manusia yang efisien [45]. Dengan demikian, efektivitas tidak hanya tentang bekerja keras, tetapi juga bekerja secara cerdas, yaitu melakukan hal yang benar dengan cara yang tepat untuk mencapai hasil yang maksimal [46].

2.2.4 Efisien

Efisiensi merujuk pada kemampuan untuk mencapai hasil yang diinginkan dengan menggunakan sumber daya yang seminimal mungkin. Dalam konteks manajemen, hal ini berarti melakukan tugas atau proses dengan memanfaatkan waktu, energi, dan biaya secara optimal tanpa ada pemborosan. Sebuah sistem yang efisien dapat diukur dari seberapa baik proses tersebut meminimalisir input sambil memaksimalkan *output*. Dengan kata lain, efisiensi memastikan bahwa setiap tindakan yang dilakukan memberikan hasil maksimal dengan penggunaan sumber daya yang paling sedikit, serta mengurangi potensi terjadinya pemborosan atau duplikasi kerja. Efisiensi sering kali menjadi indikator utama dalam menilai kinerja sebuah organisasi atau sistem [47].

Selain itu, efisiensi juga memiliki peran penting dalam meningkatkan daya saing organisasi, terutama dalam pengelolaan operasional dan pengambilan keputusan strategis. Organisasi yang efisien cenderung memiliki proses yang lebih sederhana dan terstruktur, yang pada akhirnya memungkinkan mereka untuk beroperasi dengan lebih cepat dan responsif terhadap perubahan pasar atau kebutuhan pelanggan. Penerapan teknologi informasi, sistem otomatisasi, serta praktik terbaik dalam manajemen adalah beberapa cara yang sering digunakan untuk meningkatkan efisiensi, dengan tujuan utama meningkatkan produktivitas dan menurunkan biaya operasional dalam jangka panjang [47].

2.2.5 Transparansi Keuangan

Transparansi keuangan merupakan suatu konsep yang mengedepankan pentingnya keterbukaan dalam penyampaian informasi keuangan suatu organisasi. Konsep ini bertujuan untuk memastikan bahwa seluruh data dan laporan keuangan disampaikan secara jelas, jujur, dan dapat diakses oleh seluruh pemangku kepentingan, termasuk investor, pemerintah, dan masyarakat. Dengan transparansi keuangan yang baik, perusahaan dapat meminimalisir potensi terjadinya kecurangan, manipulasi data, atau penyalahgunaan dana [48]. Lebih dari itu, transparansi juga berperan penting dalam memastikan bahwa seluruh keputusan finansial didasarkan pada data yang akurat dan dapat diverifikasi [49].

Dalam penerapannya, transparansi keuangan dapat membangun kepercayaan publik dan memperkuat hubungan perusahaan dengan para pemangku kepentingan. Keterbukaan informasi yang diberikan oleh perusahaan mengenai kondisi keuangan mereka memungkinkan pihak eksternal untuk menilai kesehatan finansial perusahaan secara objektif [50]. Hal ini juga dapat mendukung pengambilan keputusan yang lebih baik dan berbasis data oleh para manajer perusahaan. Pada akhirnya, transparansi yang baik tidak hanya mendukung tata kelola yang lebih akuntabel tetapi juga dapat meningkatkan daya tarik perusahaan di mata investor dan mitra bisnis [51].

2.3 Teori tentang Framework yang digunakan

2.3.1 SDLC

Software Development Life Cycle (SDLC) adalah kerangka kerja yang digunakan untuk mengembangkan perangkat lunak dengan cara yang terstruktur dan sistematis. *SDLC* biasanya terdiri dari beberapa tahap yang saling berhubungan, seperti perencanaan, analisis kebutuhan, desain, pengembangan, pengujian, implementasi, dan pemeliharaan. Setiap tahap ini memiliki tujuan spesifik dan *output* yang menjadi input untuk tahap berikutnya. Misalnya, pada tahap perencanaan, tim proyek menetapkan tujuan dan lingkup proyek, sementara pada tahap analisis kebutuhan, mereka mengidentifikasi dan

mendokumentasikan kebutuhan pengguna dan sistem [52]. Hasil dari tahap-tahap awal ini menjadi dasar untuk mendesain arsitektur dan fitur perangkat lunak, yang kemudian dikembangkan oleh tim pengembang [53].

Setelah proses pengembangan selesai, perangkat lunak akan memasuki tahap pengujian untuk memastikan bahwa perangkat lunak tersebut telah memenuhi kebutuhan yang telah ditentukan dan bebas dari kesalahan atau *bug* [54]. Tahap implementasi adalah saat perangkat lunak di-*deploy* ke lingkungan produksi dan siap digunakan oleh pengguna akhir. Meskipun demikian, siklus hidup perangkat lunak tidak berhenti sampai di situ. Pada tahap pemeliharaan, perangkat lunak akan terus diperbarui, dilakukan perbaikan *bug*, serta penyesuaian terhadap perubahan kebutuhan pengguna atau lingkungan operasional [55]. *SDLC* berperan penting dalam memastikan bahwa pengembangan perangkat lunak dilakukan secara efisien, sesuai dengan kebutuhan pengguna, serta tetap relevan dalam jangka panjang. Berbagai metodologi *SDLC*, seperti model *Waterfall* yang bersifat *linear* dan *Agile* yang lebih iteratif dan fleksibel, dapat digunakan sesuai dengan kebutuhan proyek, masing-masing dengan kelebihan dan kekurangannya [56]. Pemilihan model yang tepat sangat bergantung pada karakteristik proyek dan tim pengembang [52], [53].

2.3.2 RAD (*Rapid Application Development*)

Rapid Application Development adalah cara membuat perangkat lunak yang tidak memerlukan banyak perencanaan dan memungkinkan kita membuat prototipe dengan cepat. Keuntungan dari metode ini adalah proses pengembangan sistem relatif singkat dan proses pengembangannya fleksibel, karena kita dapat mendesain ulang situs *web* saat membuatnya [57], [58].

A) *Requirements Planning*

Pengguna dan analis berkolaborasi untuk menentukan hasil yang diinginkan dari suatu aplikasi atau sistem. Untuk mencapai kesepakatan, kedua belah pihak harus bekerja sama. Pengguna dari seluruh hierarki organisasi juga dibutuhkan selama tahap ini. Selama tahap ini, kami

fokus pada menemukan solusi yang layak untuk masalah yang dihadapi Organisasi secara keseluruhan.

B) *Design Workshop*

Proses perancangan dimulai dengan tahap Analisis Kebutuhan dan berlanjut melalui tahap Desain, Arsitektur, dan Pemodelan Data, dan terakhir Desain Antarmuka Pengguna, yang semuanya diinformasikan oleh hasil Perencanaan Kebutuhan. Pada titik inilah penulis mengulangi desain modul berdasarkan umpan balik dari pengguna, yang sekarang dapat memberikan masukan pada prototipe yang berfungsi penuh.

2.3.3 *Prototype*

Metode *prototyping* merupakan salah satu pendekatan dalam pengembangan sistem yang memungkinkan tim pengembang untuk menciptakan model awal dari sistem atau aplikasi yang sedang dikembangkan. Model awal ini, yang disebut prototipe, berfungsi sebagai representasi sementara yang dapat diuji dan dievaluasi oleh pengguna akhir sebelum sistem yang sesungguhnya dibangun sepenuhnya. Salah satu keunggulan utama dari metode ini adalah kemampuannya untuk memperoleh umpan balik dari pengguna pada tahap awal pengembangan. Dengan demikian, pengembang dapat mengidentifikasi kebutuhan dan keinginan pengguna secara lebih akurat serta menangani potensi masalah sejak awal, sehingga mengurangi risiko kesalahan besar pada tahap akhir pengembangan [59], [60], [61].

Selain itu, metode *prototyping* juga memperlancar komunikasi antara pengembang dan pengguna, karena pengguna dapat melihat dan berinteraksi langsung dengan prototipe yang mendekati produk akhir. Hal ini membantu dalam menjelaskan dan memvalidasi kebutuhan sistem yang mungkin sulit dipahami hanya melalui dokumentasi. Namun, tantangan dalam metode ini adalah risiko fokus yang berlebihan pada aspek tertentu dari prototipe, yang dapat menyebabkan pengembang mengabaikan elemen penting lainnya. Oleh karena itu, penting bagi *prototyping* dilakukan dengan perencanaan yang

matang dan tetap konsisten dengan tujuan keseluruhan dari proyek pengembangan sistem [60], [61].

2.3.4 Blackbox Testing

Black box testing adalah metode pengujian perangkat lunak yang berfokus pada evaluasi fungsionalitas aplikasi tanpa memerlukan pemahaman tentang kode sumber yang mendasarinya. Metode ini, yang juga dikenal sebagai *behavioral testing*, bertujuan untuk menganalisis bagaimana sistem merespons berbagai *input* dan memeriksa hasil *output* yang dihasilkan. Dalam konteks ini, *black box testing* sangat penting untuk memastikan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang telah ditentukan [62], [63]. Pengujian ini umumnya dilakukan pada tahap akhir pengembangan perangkat lunak, di mana penguji dapat memastikan bahwa semua fitur berfungsi dengan baik sebelum aplikasi dirilis kepada pengguna akhir [64].

Salah satu aspek utama dari *black box testing* adalah fokusnya pada aspek eksternal dari aplikasi. Penguji tidak perlu memahami logika *internal* atau struktur kode, melainkan cukup memastikan bahwa setiap fungsi aplikasi berjalan sesuai dengan harapan [65]. Hal ini memungkinkan penguji untuk mendeteksi kesalahan pada antarmuka pengguna, validasi data yang masuk, serta respons aplikasi terhadap data yang tidak valid [66]. Dengan demikian, *black box testing* berperan penting dalam meningkatkan kualitas perangkat lunak dengan mengidentifikasi dan memperbaiki kesalahan sebelum aplikasi digunakan oleh pengguna.

Standar kualitas yang relevan untuk pengujian fungsionalitas adalah ISO/IEC 25010:2011, yang menekankan pada dimensi *Functional Suitability* untuk memastikan perangkat lunak memenuhi kebutuhan pengguna sesuai spesifikasi [67]. Dimensi ini mencakup tiga elemen utama: *Functional Completeness*, untuk mengevaluasi apakah semua fungsi yang dibutuhkan telah diimplementasikan; *Functional Correctness*, yang memastikan bahwa hasil atau *output* sesuai dengan *input* yang diberikan; dan *Functional Appropriateness*, yang menilai apakah fungsi yang disediakan relevan dan

efektif untuk mendukung tujuan pengguna [67]. Standar ini membantu memastikan perangkat lunak tidak hanya berfungsi dengan benar tetapi juga memberikan pengalaman yang sesuai bagi pengguna.

2.4 Teori tentang *Tools/Software* yang digunakan

2.4.1 *Visual Studio Code*

Visual Studio Code (VSCode) adalah sebuah *Integrated Development Environment (IDE)* yang banyak digunakan karena fleksibilitas dan ekosistemnya yang kaya akan ekstensi. Dikembangkan oleh *Microsoft*, *VSCode* merupakan alat yang ringan namun memiliki kemampuan yang kuat dalam mendukung berbagai bahasa pemrograman dan alur kerja pengembangan perangkat lunak. Dengan antarmuka yang intuitif dan fitur bawaan seperti *debugging*, penyorotan sintaksis, integrasi pengendalian versi, serta pelengkapan kode cerdas melalui *IntelliSense*, *VSCode* menjadi pilihan utama para pengembang untuk menangani proyek-proyek mulai dari skrip sederhana hingga pengembangan perangkat lunak yang kompleks [68], [69]. Selain itu, *VSCode* sangat dapat dikustomisasi, memungkinkan pengguna untuk menyesuaikan *editor* dengan kebutuhan alur kerja mereka sendiri melalui instalasi ekstensi dari marketplace yang menyediakan alat linting, pemformatan, *debugging* khusus bahasa, dan banyak lagi [68].

Salah satu kekuatan utama *VSCode* adalah sifatnya yang lintas *platform*, tersedia untuk *Windows*, *macOS*, dan *Linux*, yang memungkinkan pengembang untuk mempertahankan lingkungan pengembangan yang konsisten tanpa tergantung pada sistem operasi yang digunakan [70]. Efisiensi dari *VSCode* semakin ditingkatkan dengan integrasi *Git* yang memungkinkan pengelolaan kode sumber langsung dari editor, menciptakan alur kerja yang mulus untuk kolaborasi dan manajemen proyek [68]. Seiring dengan meningkatnya popularitas lingkungan pengembangan berbasis *cloud*, *VSCode* telah mengintegrasikan dukungan untuk pengembangan jarak jauh dan alur kerja berbasis *cloud*, seperti melalui *Visual Studio Codespaces*. Ini memungkinkan pengembang untuk menulis, menjalankan, dan *debugging* kode pada *server*

jarak jauh tanpa harus meninggalkan *editor*, menjadikannya alat yang penting dalam praktik pengembangan perangkat lunak modern [68], [70].

2.4.2 XAMPP

XAMPP adalah perangkat lunak *open-source* yang menyediakan lingkungan pengembangan *web* lokal dengan mudah dan cepat. *XAMPP* adalah singkatan dari *Cross-Platform*, *Apache*, *MySQL*, *PHP*, dan *Perl*, yang mencerminkan komponen utama yang disertakan dalam paketnya. *XAMPP* memungkinkan pengguna untuk membuat dan menguji aplikasi *web* secara lokal di komputer mereka tanpa harus mengunggahnya ke *server web* yang sebenarnya. Hal ini sangat berguna bagi pengembang *web* yang ingin mengembangkan dan menguji aplikasi mereka sebelum meluncurkannya ke publik [71], [72]. Dengan menyediakan *server web Apache*, *database MySQL*, serta dukungan untuk bahasa pemrograman *PHP* dan *Perl*, *XAMPP* memungkinkan pengembang untuk memiliki semua alat yang mereka butuhkan dalam satu paket yang mudah diinstal dan digunakan.

Keunggulan *XAMPP* adalah kemudahan dalam pengaturan dan penggunaannya, bahkan untuk pengguna yang kurang berpengalaman dalam manajemen *server*. Paket ini mendukung berbagai sistem operasi seperti *Windows*, *Linux*, dan *macOS*, menjadikannya solusi lintas platform yang fleksibel [73], [74]. Selain itu, *XAMPP* memiliki fitur tambahan seperti *phpMyAdmin* untuk mengelola basis data *MySQL*, serta modul tambahan untuk mengelola komponen lain seperti *FileZilla FTP Server* dan *Mercury Mail Server* [74]. Hal ini membuat *XAMPP* menjadi alat yang sangat berguna bagi pengembang *web* yang membutuhkan lingkungan uji coba yang komprehensif namun tidak terlalu rumit [73], [75].

Secara keseluruhan, *XAMPP* merupakan alat yang sangat berguna bagi pengembang *web*, menyediakan semua komponen yang diperlukan untuk membangun dan menguji aplikasi *web* dalam satu paket yang mudah diinstal dan digunakan. Dengan dukungan untuk berbagai sistem operasi dan fitur

tambahan yang memperkaya fungsionalitasnya, *XAMPP* tetap menjadi pilihan utama dalam pengembangan aplikasi web lokal [73], [74], [75].

2.4.3 *PHP*

PHP merupakan bahasa pemrograman yang banyak digunakan dalam pengembangan *web*, khususnya untuk menghasilkan konten dinamis pada situs *web*. Diperkenalkan pertama kali oleh Rasmus Lerdorf pada tahun 1994, *PHP* awalnya dikembangkan untuk memantau lalu lintas pengunjung pada situs pribadinya. Seiring perkembangannya, *PHP* menjadi bahasa *scripting server-side* yang sangat andal dalam mengelola konten dinamis, basis data, serta pembuatan aplikasi *web* yang kompleks [76], [77]. Kemampuannya untuk berintegrasi dengan berbagai sistem basis data seperti *MySQL*, *PostgreSQL*, dan *SQLite*, serta kompatibilitas yang tinggi dengan *HTML*, *CSS*, dan *JavaScript*, menjadikan *PHP* sebagai pilihan yang efisien dan fleksibel dalam pengembangan aplikasi berbasis *web* [78], [79].

Selain itu, *PHP* juga dikenal karena komunitasnya yang luas dan dukungan dokumentasi yang melimpah, sehingga memudahkan pengembang dalam proses pembelajaran dan pemecahan masalah [76]. *PHP* merupakan bahasa pemrograman *open-source*, yang memungkinkan siapa saja untuk mengunduh, memodifikasi, dan menggunakan tanpa biaya. Walaupun popularitas *PHP* sedikit menurun dengan munculnya bahasa pemrograman lain seperti *Python* dan *Node.js*, *PHP* tetap menjadi salah satu bahasa utama dalam pengembangan *web*, terutama untuk proyek-proyek yang memerlukan solusi yang cepat dan dapat diskalakan [80]. Kelebihan lain dari *PHP* terletak pada dukungannya terhadap berbagai framework seperti *Laravel*, *CodeIgniter*, dan *Symfony*, yang membantu pengembang dalam menciptakan aplikasi *web* yang aman dan mudah dikelola [81], [82].

2.4.4 *HTML*

HTML (*Hypertext Markup Language*) adalah bahasa standar yang digunakan dalam pembuatan dan pengaturan halaman *web*. *HTML* berfungsi

dengan mendefinisikan struktur dasar sebuah halaman *web* melalui penggunaan elemen-elemen yang dikenal sebagai tag. Setiap tag memiliki peran spesifik, seperti menampilkan teks, gambar, tautan, serta elemen interaktif lainnya pada halaman *web*. Dengan demikian, *HTML* menjadi fondasi utama yang memungkinkan *browser* untuk menampilkan konten *web* secara tepat kepada pengguna [83]. Elemen-elemen ini juga memungkinkan penataan konten menjadi lebih terstruktur, seperti melalui penggunaan paragraf, *heading*, daftar, tabel, dan berbagai elemen lainnya [83].

Seiring berjalannya waktu, *HTML* mengalami perkembangan signifikan, dengan versi terbarunya adalah *HTML5*. *HTML5* hadir dengan fitur-fitur yang lebih canggih, termasuk dukungan untuk elemen multimedia seperti *audio* dan *video*, serta elemen interaktif seperti *canvas* dan *SVG* [83]. Selain itu, *HTML5* juga memperkenalkan struktur semantik yang lebih baik, seperti penggunaan *tag header*, *article*, dan *footer*, yang sangat membantu mesin pencari dan perangkat bantu dalam memahami konten halaman *web*. Dengan demikian, *HTML* tidak hanya memberikan struktur dasar pada halaman *web*, tetapi juga mendukung pengembangan *web* yang lebih interaktif, responsif, dan ramah terhadap mesin pencari.

Penerapan *HTML5* dalam pengembangan *web* memberikan keuntungan signifikan melalui integrasi yang lebih optimal dengan teknologi lain seperti *CSS* dan *JavaScript*, yang berkontribusi pada peningkatan pengalaman pengguna melalui antarmuka grafis yang lebih menarik dan fungsional [83]. Contohnya, fitur-fitur multimedia dan interaktif yang ada dalam *HTML5* memungkinkan pengembang untuk menciptakan aplikasi *web* yang lebih dinamis dan menarik, sehingga dapat meningkatkan tingkat keterlibatan pengguna [83]. Selain itu, *HTML5* dengan struktur semantiknya yang lebih baik juga memainkan peran penting dalam optimasi *SEO* (*Search Engine Optimization*), yang esensial untuk meningkatkan visibilitas halaman *web* di mesin pencari.

2.4.4 MySQL

MySQL, yang merupakan Sistem Manajemen Basis Data (DBMS) bersifat open-source, memanfaatkan *Structured Query Language (SQL)* sebagai bahasa utamanya, memiliki peran fundamental dalam pengelolaan data. Sistem ini krusial untuk aplikasi yang membutuhkan efisiensi dalam penyimpanan dan pengambilan data. Fungsionalitas *MySQL* yang memungkinkan operasi dasar seperti penyimpanan, pengambilan, dan manipulasi data dalam format tabel terstruktur, memberikan kemudahan dalam pengelolaan informasi [84], [85]. Keberadaan *MySQL* yang didukung oleh lisensi *GNU General Public License (GPL)* menawarkan kebebasan kepada pengguna untuk mengunduh, memodifikasi, dan menggunakan perangkat lunak ini tanpa biaya, sehingga meningkatkan popularitasnya [85].

Performa yang optimal dalam mengelola data terstruktur menjadi salah satu keunggulan *MySQL*. Namun, dengan kemunculan era *big data*, tantangan terkait dengan kompleksitas dan variasi data yang meningkat menjadi pertimbangan dalam penggunaan *MySQL*. Meskipun demikian, *MySQL* masih sering menjadi pilihan utama, sementara sistem manajemen basis data *NoSQL* juga mulai diperhatikan seiring dengan kebutuhan untuk menangani data yang tidak terstruktur [86].

MySQL memiliki sintaks yang relatif mudah untuk dipahami, yang banyak mengadopsi elemen dari bahasa pemrograman *C* dan *C++*, memudahkan akses bagi pengembang yang familiar dengan bahasa tersebut. Sistem ini juga mendukung konsep pemrograman terstruktur dan prosedural, yang memungkinkan pengguna untuk mengembangkan fungsi dan prosedur yang dapat digunakan kembali, menjadikan *MySQL* tidak hanya sebagai media penyimpanan data tetapi juga sebagai pendukung dalam pengembangan aplikasi yang lebih kompleks.

2.4.5 UML

Unified Modeling Language (UML) merupakan sebuah bahasa pemodelan yang digunakan untuk mendesain dan mendokumentasikan sistem perangkat lunak. *UML* berfungsi sebagai alat untuk memvisualisasikan berbagai aspek

dari sistem, baik dari segi struktur, perilaku, maupun interaksi antar komponen. Dalam pengembangan perangkat lunak, *UML* seringkali digunakan untuk menggambarkan desain sistem yang kompleks, sehingga memudahkan pengembang dan pihak terkait lainnya untuk memahami dan mengubah sistem sesuai kebutuhan [87], [88].

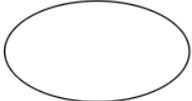
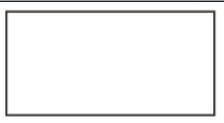
UML memiliki berbagai jenis diagram yang masing-masing dirancang untuk tujuan tertentu. Beberapa diagram yang umum digunakan meliputi diagram kelas, yang menggambarkan struktur objek dalam sistem, *diagram use case*, yang menunjukkan bagaimana pengguna berinteraksi dengan sistem, serta diagram aktivitas, yang memodelkan alur kerja dalam sistem. Dengan memanfaatkan beragam diagram ini, *UML* mampu merepresentasikan aspek statis maupun dinamis dari suatu sistem, sehingga berperan penting dalam perencanaan dan pengembangan perangkat lunak yang lebih terstruktur [87], [88], [89].

Kehadiran *UML* dalam pengembangan perangkat lunak sangatlah penting, karena memungkinkan komunikasi yang lebih baik antar anggota tim pengembang, memfasilitasi proses analisis dan perancangan sistem, serta mendukung dokumentasi yang terstruktur. Dengan adanya standar yang diterima secara luas, *UML* memfasilitasi berbagai pihak yang terlibat dalam proyek untuk menggunakan "bahasa" yang sama, yang pada gilirannya mengurangi risiko miskomunikasi dan kesalahan dalam proses pengembangan [89]. Selain itu, *UML* juga memiliki peran penting dalam pemeliharaan perangkat lunak, karena memberikan gambaran *visual* yang membantu dalam memahami struktur dan perilaku sistem yang kompleks [87].

2.4.5.1 Use Case Diagram

Use Case Diagram yang dapat dilihat pada Tabel 2.2 di bawah mengidentifikasi berbagai kasus penggunaan atau situasi yang dapat terjadi dalam penggunaan sistem. Kasus penggunaan (*use case*) menggambarkan fungsionalitas sistem dari perspektif pengguna. Setiap kasus penggunaan

memiliki skenario atau langkah-langkah yang menggambarkan bagaimana pengguna berinteraksi dengan sistem untuk mencapai tujuan tertentu.

No	Simbol	Keterangan
1		Seseorang atau sesuatu yang berpartisipasi dalam sistem sebagai “actor”
2		Istilah "Use Case" mengacu pada prosedur menyeluruh yang akan dijalankan oleh sistem; dalam konteks inilah aktor dipekerjakan.
3		kotak yang secara akurat mewakili sistem dengan menggambarkan komponen <i>eksternal</i> atau <i>internal</i> sistem dalam sebuah diagram. “System Boundar”
4		Mengkoneksikan Actor Dengan Use Case “Association”
5		Mengkoneksikan 2 actor & case. “Generalization”

Tabel 2. 2 Simbol-simbol Use Case Diagram [90]

2.4.5.2 Activity Diagram

Activity diagram yang dapat dilihat pada Tabel 2.3 di bawah adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan alur proses bisnis atau alur kerja. Diagram ini menggambarkan berbagai tahapan dalam suatu proses, termasuk aktivitas, pengambilan keputusan, serta aliran kontrol yang menghubungkan aktivitas-aktivitas tersebut. Dalam konteks penggunaannya, *activity diagram* berfungsi sebagai representasi visual yang jelas mengenai bagaimana berbagai aktivitas berinteraksi dan saling berurutan dalam suatu sistem atau proses tertentu.

Activity diagram memiliki struktur yang menyerupai *flowchart*, namun lebih berfokus pada perilaku dan interaksi antara aktivitas-aktivitas. Diagram ini dapat mencakup komponen seperti node awal, node akhir, aktivitas, serta transisi yang menghubungkan setiap aktivitas[91]. *Activity diagram* tidak hanya digunakan untuk memodelkan aktivitas, tetapi juga dapat diperluas untuk analisis dan implementasi arsitektur sistem, yang menunjukkan fleksibilitas dan penerapannya dalam berbagai konteks[91].

Tabel 2. 3 *Activity Diagram* [90]

No	Simbol	Keterangan
1		Digunakan untuk menandai awal dari suatu proses didalam aktivitas.
2		Aktivitas yang dilakukan sistem, yan biasanya diawali kata kerja
3		Digunakan untuk menunjukkan titik keputusan yang dimana kondisi harus dinilai untuk menentukan jalur yang akan diikuti.
4		Digunakan untuk mengakhiri suatu proses.
5		<i>Swimlane</i> memisahkan yang bertanggung jawab terhadap aktivitas yang terjadi.

2.4.5.3 *Class Diagram*

Class diagram pada Tabel 2.4 di bawah berfungsi untuk memodelkan struktur statis dari sistem perangkat lunak. Diagram ini menggambarkan berbagai kelas yang terdapat dalam sistem, lengkap dengan atribut dan metode

yang dimiliki oleh masing-masing kelas, serta hubungan antar kelas seperti asosiasi, agregasi, komposisi, pewarisan, dan realisasi. *Class diagram* membantu dalam perancangan sistem dengan menyediakan visualisasi yang jelas mengenai entitas-entitas dan interaksi antar kelas, sehingga memudahkan pengembang dalam merencanakan, mengkomunikasikan, serta mendokumentasikan desain sistem sebelum implementasi dilakukan. Dengan demikian, *class diagram* berperan penting dalam memastikan bahwa struktur sistem perangkat lunak dirancang secara efektif dan efisien[92].

Tabel 2. 4 *Class Diagram* [90]

No	Simbol	Keterangan
1		Simbol tabel ini memiliki fungsi untuk memetakan himpunan dari objek-objek yang berbagai atribut serta operasi yang sama.
2		Simbol belah ketupat ini memiliki fungsi sebagai upaya menghindari asosiasi dengan lebih dari 2 objek.
3		Simbol panah ke arah kiri dengan garis putus-putus ini merupakan operasi yang benar-benar hanya dilakukan oleh suatu objek.
4		Simbol anak panah ke arah kanan dengan garis putus-putus ini memetakan hubungan yang dimana perubahan terjadi pada satu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri.
5		Simbol garis ini memetakan apa yang menghubungkan antara objek satu dengan objek lainnya.