BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Berikut ini adalah beberapa penelitian terhadulu yang memiliki topik serupa dengan penelitian ini:

Tabel 2.1 Penelitian Terdahulu

PENELITIAN TERDAHULU 1 [6]	
Judul Jurnal	Analisa dan Perancangan Sistem Informasi Penggajian Karyawan Berbasis Website Menggunakan Metode Prototype CV. Muda Jeans
Nama Jurnal	EXPLORE – Volume 13 No 2 Tahun 2023
Penulis & Tahun	Erik Rahman, 2023
Metode	Penelitian ini menggunakan metode deskriptif untuk memahami sistem manual yang ada di CV. Muda Jeans dan metode *prototyping* untuk mengembangkan sistem informasi penggajian berbasis web. Metode deskriptif digunakan untuk menganalisis masalah yang ada melalui wawancara, observasi, dan pengumpulan dokumen terkait proses penggajian. Dengan metode *prototyping*, pengembang dan pengguna bekerja sama dalam beberapa iterasi yang mencakup tahap pengumpulan kebutuhan, perancangan mockup awal, evaluasi, dan revisi prototipe. Proses iteratif ini memungkinkan prototipe menjadi semakin sesuai dengan kebutuhan pengguna. Tahapan ini diakhiri dengan pengujian sistem menggunakan metode black box dan white box untuk memastikan semua fitur berfungsi sesuai rencana.

Kesimpulan	Prototipe sistem penggajian karyawan berbasis web yang dikembangkan untuk CV. Muda Jeans berhasil meningkatkan efisiensi dan akurasi proses penggajian. Sistem ini mempermudah pencatatan absensi dan penghitungan gaji, sehingga mengurangi potensi kesalahan dan duplikasi data. Selain itu, sistem ini mempercepat proses pembuatan slip gaji dan laporan penggajian, menjadikannya lebih praktis dan dapat diandalkan. Hasil penelitian menunjukkan bahwa metode *prototyping* efektif dalam merancang sistem yang sesuai dengan kebutuhan pengguna.

PENELITIAN TERDAHULU 2 [7]	
Judul Jurnal	Penerapan Model Prototype pada Sistem Penggajian Karyawan PT. Sutera Agung Properti
Nama Jurnal	PROSISKO Vol. 9 No. 1, Maret 2022
Penulis & Tahun	Reni Widyastuti, Vincent Luis, 2022
Metode	Penelitian ini menggunakan metode *prototyping* untuk merancang sistem penggajian berbasis web di PT. Sutera Agung Properti. Data dikumpulkan melalui wawancara dan observasi untuk mengidentifikasi masalah dalam sistem manual, seperti kesalahan rekap absensi dan keterlambatan penghitungan gaji. Tahapan *prototyping* meliputi analisis kebutuhan pengguna, pembuatan mockup awal, dan iterasi berulang hingga sistem memenuhi kebutuhan pengguna. Bahasa pemrograman PHP dan MySQL digunakan untuk membangun sistem ini, sementara diagram UML seperti use case, activity, dan class digunakan untuk mendesain

	struktur sistem. Evaluasi dilakukan melalui pengujian black box untuk memastikan setiap fitur berjalan sesuai spesifikasi.
Kesimpulan	Sistem penggajian berbasis web yang dirancang untuk PT. Sutera Agung Properti berhasil mengatasi kelemahan sistem manual, seperti kesalahan dalam rekapitulasi absensi dan penghitungan gaji. Sistem ini meningkatkan efisiensi dan akurasi dalam pengelolaan data gaji, mempermudah proses operasional, dan mendukung transparansi. Metode *prototyping* terbukti efektif dalam memahami kebutuhan pengguna dan menghasilkan sistem yang dapat diandalkan untuk pengelolaan penggajian.

PENELITIAN TERDAHULU 3 [8]	
Judul Jurnal	Penerapan Metode Prototype Pada Perancangan Sistem Informasi Penggajian Karyawan Berbasis Web
Nama Jurnal	Paradigma, Vol. 23, No. 2, September 2021
Penulis & Tahun	Eka Wulansari Fridayanthie, Haryanto, Tsamarah Tsabitah, 2021
Metode	Penelitian ini menggunakan metode deskriptif untuk menganalisis proses bisnis di PT. Bina Angkasa dan mengidentifikasi kendala dalam sistem penggajian manual. Pengembangan sistem dilakukan dengan metode *prototyping*, yang mencakup tahap pengumpulan kebutuhan pengguna, desain prototipe, evaluasi, dan iterasi perbaikan. Data absensi karyawan dikumpulkan secara otomatis

	melalui sistem berbasis web, diproses oleh admin untuk menghitung gaji, dan menghasilkan slip gaji serta laporan penggajian. Tahapan *prototyping* memastikan bahwa sistem dirancang berdasarkan masukan pengguna melalui evaluasi berulang.
Kesimpulan	Hasil penelitian menunjukkan bahwa sistem penggajian berbasis web yang dikembangkan untuk PT. Bina Angkasa meningkatkan efisiensi operasional, akurasi data, dan transparansi dalam pengelolaan gaji karyawan. Sistem ini mempermudah pengelolaan absensi dan penghitungan gaji, serta mengurangi risiko kesalahan yang sering terjadi pada sistem manual. Proses iterasi dalam metode *prototyping* memungkinkan sistem memenuhi kebutuhan pengguna secara optimal.

PENELITIAN TERDAHULU 4 [9]	
Judul Jurnal	Critical Success Factor Based Resource Allocation in ERP Implementation: A Nonlinear Programming Model
Nama Jurnal	International Journal of Research, Volume 8, Issue 8
Penulis & Tahun	Ying Xie, Colin Allen, dan Mahmood Ali, 2022
Metode	Penelitian ini menggunakan metode *prototyping* untuk mengembangkan sistem informasi desa berbasis web di Desa Lau Gumba. Data dikumpulkan melalui wawancara, observasi, dan studi pustaka untuk memahami kebutuhan sistem dan tantangan yang dihadapi perangkat desa. Tahapan pengembangan meliputi analisis kebutuhan, desain awal menggunakan diagram hubungan entitas

	(ERD), pengkodean, dan pengujian. Sistem dirancang untuk menyediakan informasi pemerintah desa secara efektif dan efisien, dengan fitur tambahan seperti publikasi kegiatan desa. Prototipe diuji dan dievaluasi secara iteratif untuk memastikan kesesuaian dengan kebutuhan pengguna.
Kesimpulan	Sistem informasi desa berbasis web yang dikembangkan meningkatkan efisiensi pengelolaan data di Desa Lau Gumba. Sistem ini membantu perangkat desa dalam menyebarkan informasi secara luas, mengurangi hambatan komunikasi, dan meningkatkan transparansi pemerintahan. Metode *prototyping* memungkinkan sistem dirancang dengan masukan langsung dari pengguna, menghasilkan aplikasi yang user-friendly dan relevan.
PENELITIAN TERDAHULU 5 [10]	
Judul Jurnal	Implementasi E-Commerce Berbasis Web pada Toko Denia Donuts Menggunakan Metode Prototype
Nama Jurnal	JUSIFO (Jurnal Sistem Informasi), Vol. 6, No. 2, Desember 2020
Penulis & Tahun	F. J. Molina-Castillo, R. Rodríguez, C. López-Nicolas, and H. Bouwman, 2022
Metode	Penelitian ini menggunakan metode *prototyping* untuk mengembangkan sistem e-commerce berbasis web di Toko Denia Donuts. Tahapan awal melibatkan wawancara dengan pemilik toko untuk memahami kebutuhan sistem, seperti informasi stok produk, laporan penjualan, dan kemudahan transaksi konsumen. Berdasarkan informasi ini, pengembang membuat prototipe awal yang mencakup fitur dasar. Prototipe kemudian diuji oleh pengguna, dan umpan balik digunakan untuk memperbaiki sistem melalui iterasi berulang. Framework CodeIgniter dipilih sebagai platform pengembangan

	karena sifatnya yang ringan dan mudah digunakan. Pengujian dilakukan menggunakan metode black box untuk memastikan sistem memenuhi spesifikasi teknis dan fungsional.
Kesimpulan	Sistem e-commerce berbasis web yang dikembangkan untuk Toko Denia Donuts mempermudah proses transaksi konsumen dan pengelolaan toko. Fitur-fitur seperti pencatatan stok otomatis, laporan penjualan, dan tampilan produk meningkatkan efisiensi operasional. Metode *prototyping* memungkinkan pengembang merespons kebutuhan pengguna secara iteratif, menghasilkan sistem yang user-friendly dan sesuai dengan ekspektasi pemilik toko. Hasilnya, toko dapat mengelola bisnis secara lebih profesional dan meningkatkan pengalaman pelanggan.

PENELITIAN TERDAHULU 6 [11]	
Judul Jurnal	Implementasi *Prototyping* dalam Perancangan Sistem Informasi Ikatan Keluarga Alumni Santri Berbasis Web
Nama Jurnal	JATIKA, Vol. 2, No. 2, Juni 2021
Penulis & Tahun	Laila Nur Auliani, 2023
Metode	Penelitian ini menggunakan metode *prototyping* untuk merancang sistem informasi berbasis web bagi Ikatan Keluarga Alumni Santri Pondok Pesantren Walisongo (ITTMAWAS). Penelitian dimulai dengan pengumpulan data melalui observasi, wawancara, dan studi literatur untuk mengidentifikasi kebutuhan pengguna, seperti publikasi kegiatan, pengolahan data anggota, dan penyebaran informasi. Prototipe awal dirancang menggunakan

	Use Case Diagram, Activity Diagram, dan Class Diagram untuk memetakan alur sistem. Setelah prototipe selesai, dilakukan evaluasi oleh pengguna dengan mengumpulkan masukan untuk perbaikan lebih lanjut. Pengembangan iteratif ini memastikan sistem memenuhi kebutuhan organisasi secara menyeluruh.
Kesimpulan	Hasil penelitian menunjukkan bahwa sistem informasi berbasis web untuk ITTMAWAS meningkatkan efisiensi dalam pengelolaan data anggota dan publikasi kegiatan. Sistem ini memungkinkan penyebaran informasi secara luas dengan biaya lebih rendah dibandingkan metode konvensional. Dengan menggunakan metode *prototyping*, pengembang dapat menghasilkan sistem yang responsif terhadap kebutuhan pengguna, membantu organisasi dalam mengoptimalkan operasional dan memperluas jangkauan informasinya.

PENELITIAN TERDAHULU 7 [12]	
Judul Jurnal	Sistem Informasi Penggajian Karyawan Berbasis Web dengan Metode Prototype (Studi Kasus: Minuman Tay-Tay)
Nama Jurnal	JORAPI, Vol. 1, No. 3, Juli 2023
Penulis & Tahun	Intra Swadaya Hidayat, Eko Setiawan, Yanti Efendi, Taufik Ihsan, 2023
Metode	Penelitian ini menggunakan metode *prototyping* untuk mengembangkan sistem informasi penggajian berbasis web di

	Minuman Tay-Tay. Data kebutuhan sistem dikumpulkan melalui wawancara dengan pemilik usaha untuk memahami proses penggajian manual yang digunakan, seperti pencatatan absensi, gaji pokok, dan lembur. Prototipe awal dirancang untuk memetakan fitur utama, termasuk perhitungan gaji otomatis dan laporan keuangan. Evaluasi prototipe dilakukan secara iteratif dengan melibatkan pengguna untuk memastikan bahwa sistem memenuhi kebutuhan. Sistem dirancang menggunakan teknologi berbasis web untuk kemudahan akses dan integrasi data.
Kesimpulan	Sistem informasi penggajian berbasis web yang dikembangkan untuk Minuman Tay-Tay mempermudah proses penghitungan gaji karyawan, mengurangi kesalahan manual, dan meningkatkan efisiensi waktu. Sistem ini memungkinkan pencatatan absensi otomatis dan menghasilkan laporan penggajian yang lebih akurat. Metode *prototyping* membantu pengembang merespons masukan pengguna dengan cepat, menghasilkan sistem yang relevan dan mudah digunakan oleh pemilik usaha. Hasilnya, pengelolaan data penggajian menjadi lebih terstruktur dan profesional.

PENELITIAN TERDAHULU 8 [13]	
Judul Jurnal	Exploring the Impact of Prototype Fidelity on Crowdfunding Performance
Nama Jurnal	Journal of Business Venturing, Vol. 37

Penulis & Tahun	M. Wessel et al, Ferdinand Thies, Alexander Benlian, 2022
Metode	Penelitian ini menggunakan analisis empiris berbasis data survei yang diterapkan pada berbagai kampanye crowdfunding untuk mengeksplorasi hubungan antara tingkat fidelitas *prototype* dan keberhasilan kampanye. Analisis dilakukan melalui regresi probit dan uji moderasi untuk mengevaluasi efek variabel terkait, seperti kualitas presentasi dan jenis hadiah yang ditawarkan.
Kesimpulan	Penelitian menunjukkan bahwa ada hubungan berbentuk U-terbalik antara fidelitas *prototype* dan keberhasilan crowdfunding, di mana tingkat fidelitas yang moderat lebih efektif dalam menarik dukungan finansial. Faktor moderasi, seperti kualitas presentasi dan materialitas hadiah, memengaruhi efektivitas fidelitas *prototype*. Hasil ini menekankan pentingnya mencapai keseimbangan optimal dalam fidelitas *prototype* untuk meningkatkan performa kampanye crowdfunding.

PENELITIAN TERDAHULU 9 [14]	
Judul Jurnal	Integrated System Design Of Sales And Production Module Using RAD Method (Case Study: PT Shafira Putri Kreatif)
Nama Jurnal	International Journal of New Media Technology, Vol. 10, No. 1

Penulis & Tahun	A. Dzaki Hernindyaputra, M. E. Johan, and D. Tjahjana, 2023
Metode	Penelitian ini menggunakan metode Rapid Application Development (RAD) untuk merancang dan mengembangkan modul penjualan dan produksi yang terintegrasi. Prototipe sistem dibuat melalui beberapa iterasi, melibatkan umpan balik pengguna dan pengujian fungsional. Pengumpulan data dilakukan melalui wawancara, observasi, dan analisis dokumen perusahaan. Sistem diuji untuk memastikan kompatibilitas dan efisiensi operasionalnya.
Kesimpulan	Implementasi modul terintegrasi menghasilkan peningkatan efisiensi proses penjualan dan produksi. Sistem ini berhasil mengurangi redundansi data, meningkatkan transparansi, dan mempersingkat waktu proses. Namun, implementasi memerlukan pelatihan pengguna yang intensif dan perbaikan dokumentasi untuk memastikan keberhasilan adopsi sistem. Selain itu, perlu adanya integrasi lebih lanjut untuk memastikan skalabilitas sistem di masa depan.

PENELITIAN TERDAHULU 10 [15]	
Judul Jurnal	*The Development of Web-based Sales Reporting Information Systems using Rapid Application Development Method*
Nama Jurnal	Ultima Infosys: Jurnal Ilmu Sistem Informasi

Penulis & Tahun	Suryasari, Jansen Wiratama, Ririn Ikana Desanti, 2023
Metode	Penelitian ini menggunakan metode Rapid Application Development (RAD) untuk mengembangkan Sistem Informasi Pelaporan Penjualan berbasis web. Tahapan yang dilakukan mencakup perencanaan kebutuhan, desain menggunakan UML Diagram dan Figma, pengkodean dengan PHP, serta pengujian sistem menggunakan metode black-box melalui User Acceptance Test (UAT). Pengujian ini dilakukan untuk memvalidasi fungsionalitas sistem sebelum implementasi penuh. Metode RAD dipilih karena memungkinkan pengembangan yang cepat, iteratif, dan efisien tanpa mengorbankan kualitas sistem.
Kesimpulan	Sistem berhasil diuji dengan skor rata-rata 82,5% dalam UAT, yang menunjukkan sistem dapat memenuhi kebutuhan perusahaan dalam pelaporan penjualan, meningkatkan efisiensi, dan memberikan keamanan lebih baik melalui fitur otentikasi dan hak akses pengguna. Sistem ini mampu meminimalkan kesalahan serta mendukung pengelolaan data secara *real-time* dibandingkan dengan sistem sebelumnya.

Penelitian tentang pengembangan sistem informasi berbasis web dengan menggunakan metode *prototyping* telah banyak dilakukan, menunjukkan bahwa metode ini efektif untuk menghasilkan sistem yang sesuai dengan kebutuhan pengguna. Sebagai contoh, penelitian oleh Erik Rahman di CV. Muda Jeans menggunakan metode *prototyping* untuk merancang sistem informasi penggajian berbasis web [9]. Penelitian ini berhasil meningkatkan efisiensi dalam penghitungan gaji dan mengurangi kesalahan dalam pencatatan manual. Pendekatan iteratif memungkinkan pengguna berpartisipasi aktif dalam proses

pengembangan, menghasilkan sistem yang responsif terhadap kebutuhan bisnis. Temuan ini relevan dengan penelitian Anda, yang juga mengadopsi metode *prototyping* untuk mengembangkan modul ERP berbasis web pada sektor penjualan dan keuangan, dengan tujuan utama meningkatkan efisiensi operasional dan akurasi data [10].

Penelitian lainnya oleh Reni Widyastuti dan Vincent Luis di PT. Sutera Agung Properti menyoroti permasalahan dalam sistem penggajian manual yang mengakibatkan keterlambatan dan kesalahan dalam rekapitulasi absensi. Dengan menggunakan metode *prototyping*, mereka mengembangkan sistem penggajian berbasis web yang mampu menyelesaikan permasalahan tersebut. Sistem ini dirancang menggunakan PHP dan MySQL, serta memanfaatkan diagram UML untuk mendukung pengembangan. Hasilnya, sistem ini meningkatkan akurasi data dan mempercepat proses penggajian. Hal ini sejalan dengan penelitian Anda yang berfokus pada pengembangan sistem berbasis web untuk menyelesaikan permasalahan manual di Bless Auto Care (BAC), khususnya dalam modul keuangan [11].

Sementara itu, penelitian oleh Eka Wulansari Fridayanthie et al. di PT. Bina Angkasa mengembangkan sistem penggajian karyawan berbasis web yang dirancang untuk meningkatkan efisiensi dan akurasi proses penggajian. Dengan menggunakan metode *prototyping*, penelitian ini menunjukkan bahwa pengembangan iteratif dapat menghasilkan sistem yang disesuaikan dengan kebutuhan pengguna. Proses ini melibatkan pembuatan mockup, evaluasi oleh pengguna, dan revisi prototipe hingga menghasilkan sistem yang optimal. Penelitian ini memberikan landasan kuat bagi penelitian Anda, yang juga menggunakan pendekatan serupa untuk memastikan sistem ERP yang dirancang sesuai dengan kebutuhan operasional BAC [12].

Selain itu, penelitian oleh Supiyandi et al. tentang sistem informasi desa berbasis web di Desa Lau Gumba menyoroti pentingnya keterlibatan pengguna dalam pengembangan sistem. Dengan menggunakan metode *prototyping*, penelitian ini berhasil menciptakan sistem informasi yang meningkatkan

transparansi dan efisiensi pengelolaan data desa. Meski fokus penelitian ini berbeda, pendekatan yang digunakan relevan untuk penelitian Anda, karena sama-sama bertujuan mengatasi kendala manual dengan solusi berbasis teknologi yang interaktif dan responsif.

Penelitian oleh Fenando di Toko Denia Donuts menggunakan metode *prototyping* untuk mengembangkan sistem e-commerce berbasis web. Sistem ini mempermudah transaksi pelanggan dan pengelolaan inventaris, memberikan contoh nyata bagaimana *prototyping* dapat digunakan untuk meningkatkan efisiensi bisnis kecil. Penelitian ini relevan dengan penelitian Anda dalam hal penerapan teknologi untuk meningkatkan kepuasan pelanggan melalui sistem yang lebih efisien.

Penelitian oleh Hadi Ismatullah et al. menunjukkan pentingnya *prototyping* dalam pengembangan sistem informasi berbasis web untuk Ikatan Keluarga Alumni Santri Pondok Pesantren Walisongo (ITTMAWAS). Prototipe yang dirancang iteratif membantu pengguna memahami dan mengadopsi sistem baru secara efektif. Hal ini menunjukkan relevansi *prototyping* dalam berbagai konteks, termasuk penelitian Anda.

Terakhir, penelitian oleh Gilang Ramadhan dan Maulana Ardhiansyah tentang sistem informasi penggajian karyawan berbasis web di Minuman Tay-Tay mempertegas keunggulan metode *prototyping* dalam menghasilkan sistem yang sesuai kebutuhan spesifik. Sistem ini mengurangi kesalahan manual dan meningkatkan kecepatan pengolahan data, hal yang serupa dengan tujuan Anda dalam mengembangkan modul ERP untuk meningkatkan efisiensi operasional BAC [9].

Persamaan utama dari penelitian-penelitian tersebut adalah fokus pada peningkatan efisiensi, akurasi, dan transparansi proses bisnis melalui sistem berbasis web yang dirancang secara iteratif. Pendekatan ini melibatkan pengguna dalam proses pengembangan untuk memastikan solusi yang dihasilkan benar-benar memenuhi kebutuhan mereka. Penelitian ini mengambil inspirasi dari temuan-temuan tersebut, terutama dalam hal penggunaan *prototyping* untuk

mengembangkan sistem ERP berbasis web yang bertujuan meningkatkan efisiensi operasional dan akurasi data di Bless Auto Care, sejalan dengan implementasi serupa pada sistem penggajian, e-commerce, dan pengelolaan data di berbagai konteks lainnya.

Jika dibandingkan dengan penelitian terdahulu, penelitian ini memiliki beberapa perbedaan signifikan. Pertama, penelitian ini tidak hanya berfokus pada hasil implementasi ERP, tetapi juga pada pengembangan *prototype* berbasis *website* untuk modul *Finance* dan *Sales* yang dirancang khusus untuk industri otomotif. Kedua, pendekatan *prototyping* dalam penelitian ini menggunakan alat seperti Figma, yang memberikan solusi interaktif dan fungsional bagi pengguna akhir. Dengan demikian, penelitian ini memberikan kontribusi yang lebih terfokus pada pengembangan solusi praktis dan terukur untuk sektor otomotif.

2.2 Tinjauan Teori

2.2.1 *Enterprise Resource Planning*

Enterprise Resource Planning (ERP) adalah sistem perangkat lunak yang mengintegrasikan berbagai fungsi bisnis ke dalam satu platform yang terpusat, memungkinkan perusahaan untuk mengelola sumber daya secara lebih efisien. ERP membantu perusahaan untuk mengotomatiskan proses bisnis, meningkatkan koordinasi antar departemen, dan meminimalkan kesalahan yang disebabkan oleh proses manual [15], [16].

Dengan mengintegrasikan berbagai fungsi dan departemen, ERP memungkinkan perusahaan untuk mengurangi redundansi data, meningkatkan akurasi informasi, dan merespons perubahan bisnis dengan lebih cepat. Sistem ini juga memfasilitasi otomatisasi proses bisnis, menghilangkan kegiatan manual yang rentan terhadap kesalahan dan meningkatkan efisiensi operasional [17]. Salah satu elemen terpenting dari sistem ERP adalah integrasi lintas fungsi dan lintas departemen dalam suatu perusahaan, yang berarti bahwa data dan aktivitas bisnis dari beberapa departemen seperti keuangan, distribusi, manufaktur, dan sumber daya manusia dapat langsung diintegrasikan. Hal ini mengurangi duplikasi data dan meningkatkan visibilitas di seluruh aktivitas perusahaan. Dengan ERP, data dan

informasi terkait keuangan, sumber daya manusia, rantai pasokan, manufaktur, penjualan, dan layanan pelanggan dapat diakses secara *real-time*, yang mendukung pengambilan keputusan yang lebih cepat dan lebih akurat. Sistem ERP juga bisa membuat perusahaan mengotomatisasi berbagai tugas operasional, administratif, dan manajerial dalam satu platform terpusat [18].

ERP memiliki sejumlah modul yang dapat disesuaikan dengan kebutuhan spesifik setiap perusahaan. Modul tersebut mencakup keuangan dan akuntansi (*finance*), manajemen sumber daya manusia (SDM), manufaktur, rantai pasokan (*supply chain*), penjualan (*sales*) dan manajemen (*management)* [19]. Setiap modul dirancang untuk menangani fungsi khusus dalam organisasi dan terintegrasi secara menyeluruh untuk memberikan gambaran lengkap tentang kinerja perusahaan. Implementasi ERP tidak hanya berkaitan dengan teknologi, tetapi juga melibatkan perubahan proses bisnis dan perubahan organisasi.

Keuntungan yang diperoleh dari implementasi ERP mencakup peningkatan efisiensi operasional, pengambilan keputusan yang lebih baik berdasarkan data yang akurat, pengurangan biaya administratif, dan peningkatan kepuasan pelanggan. Dengan menyediakan visibilitas menyeluruh terhadap rantai pasokan, perusahaan juga dapat merespons lebih cepat terhadap perubahan pasar dan mengoptimalkan persediaan. Namun, implementasi ERP juga melibatkan sejumlah tantangan [20]. Biaya implementasi dan waktu yang diperlukan untuk mengadaptasi organisasi menjadi fokus utama. Selain itu, perubahan budaya perusahaan dan resistensi dari karyawan terhadap perubahan seringkali menjadi hambatan yang perlu diatasi.

2.2.2 *Modul Sales*

Modul *sales* pada sistem ERP digunakan untuk mengelola aktivitas-aktivitas yang mencakup penjualan, seperti manajemen pemesanan, penanganan pelanggan, hingga pengelolaan proses transaksi penjualan. Modul ini memberi kesempatan untuk sebuah perusahaan melacak penjualan secara *real-time*, mengola data pelanggan, serta menyediakan analisis penjualan yang berguna untuk pengambilan

keputusan bisnis [21]. Modul ini juga akan membantu perusahaan dalam mempercepat proses penjualan dan meningkatkan kepuasan pelanggan.

Dalam implementasi ERP pada BAC, modul *sales* akan membantu mengotomatisasi proses pemesanan layanan pelanggan. Pelanggan dapat memesan layanan *detailing* mobil melalui platform berbasis web, melihat ketersediaan jadwal, serta melacak status pengerjaan secara *real-time*. Modul ini juga akan membantu perusahaan dalam mengelola antrian pelanggan, mengurangi waktu tunggu, dan memastikan layanan yang lebih cepat dan efisien, sehingga meningkatkan kepuasan pelanggan secara keseluruhan.

2.2.3 *Modul Finance*

Modul *finance* dalam *Enterprise Resource Planning* (ERP) berperan sangat penting karena modul ini mempunyai tanggung jawab dalam mengelola seluruh aspek keuangan suatu organisasi [22]. Modul ini mencakup berbagai fungsi, seperti proses penggajian, mengurangi kesalahan manual dalam perhitungan gaji dan memastikan pembayaran tepat waktu. Lalu Meningkatkan efisiensi kerja tim keuangan dan mengurangi risiko *over budget*. Dengan mengotomatisasi proses keuangan, modul *Finance* membantu perusahaan dalam meminimalkan kesalahan manual, mempercepat proses administrasi, dan meningkatkan akurasi pelaporan keuangan.

BAC memakai modul *finance* untuk memudahkan Perusahaan dalam mengelola penggajian karyawan yang sebelumnya masih dilakukan secara manual. BAC mempunyai sistem bonus gajian mereka sendiri yang dimana terkadang mereka mengalami kesalahan menghitung dalam perhitungan bonus dan gaji karyawan. Hadirnya modul *finance* ini dapat membuat BAC terhindar dari kesalahan tersebut, karena sistem ERP akan mengotomatisasi perhitungan gaji, bonus dan potongan, sehingga dapat memastikan pembayaran yang tepat waktu dan akurat, sekaligus memberikan laporan keuangan yang lebih transparan dan mudah diakses oleh manajemen.

2.3 Teori tentang *Framework* yang Digunakan

2.3.1 *Prototype*

Prototype adalah versi percobaan dari suatu sistem yang dirancang untuk menguji konsep dan fungsionalitas sebelum dilakukannya implementasi. Dalam metode ini, *prototype* sistem dibuat sebagai perantara untuk memungkinkan pengembang dan pengguna berinteraksi selama proses pengembangan sistem informasi. *Prototyping* merupakan metode yang efektif untuk mengevaluasi kegunaan dan fungsionalitas suatu sistem, karena memungkinkan pengembang dan pengguna untuk berinteraksi langsung dengan sistem dalam fase awal pengembangan [23]. *Prototype* ini akan disesuaikan dengan rencana dan analisis yang dilakukan oleh pengembang sampai proses eksperimen selesai bersamaan dengan pengembangan sistem. *Prototype* mempunyai 4 manfaat utama, yaitu:

a. *Understanding*

Membantu mengidentifikasi pendekatan optimal untuk menyelesaikan permasalahan atau menghasilkan solusi yang sesuai dengan kebutuhan pengguna.

b. *Communicate*

Memungkinkan pertukaran gagasan yang efektif antara anggota tim pengembang, dan klien. Lalu bisa membantu penentuan desain final dan interaksi yang akan diimplementasikan.

c. *To test*

Menyempurnakan semua ide berdasarkan masukan dari pengguna. Lalu penyempurnaan desain juga bisa dilakukan dari saran dan masukan yang diterima.

d. *Advocate*

Menunjukkan bahwa pilihan desain didasarkan pada penelitian pengguna dan sejalan dengan kepentingan bisnis.

2.3.2 Metode *SDLC Prototyping*

SDLC (*System Development Life Cycle*) adalah pendekatan sistematis dalam pengembangan perangkat lunak atau sistem informasi, yang terdiri dari beberapa tahap berurutan untuk memastikan bahwa sistem yang dibangun dapat memenuhi kebutuhan pengguna secara tepat. Tahapan dalam SDLC biasanya mencakup perencanaan, analisis kebutuhan, perancangan, pengembangan, pengujian, implementasi, dan pemeliharaan. Dengan struktur yang jelas ini, SDLC membantu organisasi meminimalkan risiko, menghindari kesalahan dalam pengembangan, dan memastikan bahwa sistem yang dihasilkan dapat berfungsi dengan baik dalam mendukung operasional bisnis [24].

Pada tahap perencanaan dan analisis kebutuhan, tim pengembang bekerja sama dengan pemangku kepentingan untuk mengidentifikasi tujuan bisnis dan kebutuhan spesifik yang harus dipenuhi oleh sistem. Hasil dari fase ini menjadi dasar untuk desain sistem dalam tahap perancangan, di mana alur kerja, antarmuka pengguna, dan arsitektur teknis sistem dirumuskan. Setelah itu, sistem masuk ke fase pengembangan, di mana programmer dan teknisi membangun sistem berdasarkan spesifikasi desain yang telah ditetapkan [24]. Fase pengujian kemudian dilakukan untuk memastikan bahwa semua fungsi berjalan sesuai rencana dan tidak terdapat bug atau kesalahan dalam sistem.

SDLC juga menekankan implementasi dan pemeliharaan sebagai bagian penting dari siklus hidup sistem. Setelah sistem diuji dan disetujui, tahap implementasi dilakukan, di mana sistem diterapkan secara penuh dan diintegrasikan dengan proses bisnis yang ada. Pemeliharaan menjadi aspek krusial karena sistem harus terus dipantau dan ditingkatkan agar tetap relevan dengan kebutuhan bisnis yang dinamis. Metode SDLC memastikan bahwa setiap tahap pengembangan dilakukan secara terstruktur dan terkendali [24], sehingga meminimalkan risiko kegagalan dan memaksimalkan nilai bagi organisasi.

2.3.3 *Blackbox Testing*

Blackbox testing adalah metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas sistem tanpa mengacu pada kode sumber program [25]. Dalam pendekatan ini, penguji tidak perlu mengetahui struktur internal atau implementasi teknis dari aplikasi. Blackbox testing bertujuan untuk memeriksa apakah perangkat lunak dapat berfungsi sesuai dengan spesifikasi yang telah ditentukan [25]. Proses pengujian dilakukan dengan memberikan berbagai jenis input ke dalam sistem dan kemudian mengevaluasi output yang dihasilkan, guna memastikan bahwa aplikasi berjalan dengan benar sesuai dengan harapan.

Metode pengujian ini sangat berguna untuk mengidentifikasi cacat atau kekurangan pada aspek fungsional aplikasi, terutama dalam hal validasi data input, respon sistem terhadap masukan, dan integritas hasil keluaran [25]. Pendekatan ini juga membantu mengidentifikasi potensi bug pada tahap awal pengembangan, sehingga perbaikan dapat dilakukan dengan lebih cepat dan efisien.

2.4 *Tools/Software* yang Digunakan

2.4.1 *MySQL*

MySQL adalah sistem manajemen basis data yang bersifat relasional, artinya data yang disimpan dalam MySQL diatur dalam bentuk tabel-tabel yang saling terhubung melalui relasi tertentu [26]. MySQL berperan dalam pengolahan basis data dengan menggunakan bahasa *Structured Query Language* (SQL). SQL merupakan bahasa standar yang digunakan untuk mengakses, memanipulasi, dan mengelola data dalam basis data relasional [26]. Keunggulan utama MySQL adalah kemampuannya untuk menangani berbagai operasi basis data secara efisien, seperti penyimpanan, pembaruan, penghapusan, dan pengambilan data.

Kemampuan MySQL untuk mengelola data dalam jumlah besar dan melakukan replikasi basis data secara otomatis menjadikannya pilihan unggul bagi perusahaan yang membutuhkan pengelolaan basis data besar dan

kompleks. Selain itu, MySQL mendukung fitur *clustering*, yang memungkinkan beberapa server MySQL bekerja secara simultan untuk meningkatkan keandalan dan ketersediaan basis data, bahkan saat menghadapi beban kerja yang tinggi. MySQL juga memungkinkan pengolahan data menjadi lebih terorganisir dan transparan. Selain itu, MySQL dikenal sebagai platform yang andal dan *scalable*, yang cocok untuk mendukung pengembangan sistem dengan kebutuhan data yang terus berkembang di masa depan.

2.4.2 *PHP*

PHP adalah bahasa pemrograman yang banyak digunakan untuk pengembangan web, khususnya dalam menghasilkan konten dinamis pada situs [27]. Diperkenalkan oleh Rasmus Lerdorf pada tahun 1994, PHP awalnya dikembangkan untuk memantau lalu lintas pengunjung situs pribadinya. Kini, PHP menjadi bahasa scripting server-side yang andal untuk mengelola konten dinamis, basis data, dan aplikasi web yang kompleks. Kemampuannya untuk terintegrasi dengan berbagai basis data seperti MySQL, PostgreSQL, dan SQLite, serta kompatibilitas tinggi dengan HTML, CSS, dan JavaScript, menjadikan PHP fleksibel dan efisien dalam pengembangan web [27].

Sebagai bahasa open-source, PHP dapat digunakan dan dimodifikasi secara bebas, dengan dukungan komunitas luas dan dokumentasi yang kaya. Meskipun persaingan dari bahasa pemrograman lain seperti Python dan Node.js semakin meningkat, PHP tetap populer berkat keunggulannya dalam solusi cepat dan skalabilitas. Selain itu, dukungan berbagai framework seperti Laravel dan CodeIgniter semakin mempermudah pengembang dalam menciptakan aplikasi web yang aman dan terstruktur.

2.4.3 *Visual Studio Code*

Visual Studio Code adalah editor kode yang dikembangkan oleh Microsoft dan dapat digunakan di berbagai perangkat desktop [28]. Dengan fitur yang lengkap dan berbagai ekstensi, *Visual Studio Code* menjadi pilihan utama banyak pengembang. Aplikasi ini mendukung berbagai sistem operasi seperti Windows, Mac OS, Linux, dan lainnya.

Efisiensi *Visual Studio Code* semakin maksimal berkat integrasi Git, yang memungkinkan pengelolaan kode sumber langsung dari editor, sehingga mendukung alur kerja kolaboratif dan manajemen proyek secara efektif [28]. Selain itu, dengan semakin populernya pengembangan berbasis cloud, *Visual Studio Code* menyediakan dukungan untuk pengembangan jarak jauh melalui fitur seperti *Visual Studio Codespaces*. Hal ini memungkinkan pengembang untuk menulis, menjalankan, dan debugging kode langsung pada server jarak jauh tanpa perlu meninggalkan *editor*, menjadikannya alat yang relevan untuk pengembangan perangkat lunak modern.

2.4.4 *Figma*

Figma merupakan *platform* desain kolaboratif berbasis web yang memungkinkan pembuatan *prototype* interaktif dan desain antarmuka pengguna (UI) secara *real-time* [29]. *Platform* ini dimanfaatkan oleh para desainer untuk mengembangkan UI/UX aplikasi dan situs web, dengan fitur-fitur yang mendukung kerjasama tim baik secara langsung maupun jarak jauh. Keunggulan utama Figma terletak pada aksesibilitasnya melalui *browser* tanpa memerlukan instalasi perangkat lunak lokal, serta kemampuan kolaborasi langsungnya yang memungkinkan berbagai pihak terkait untuk berpartisipasi secara simultan dalam proses desain.

Dalam konteks penelitian ini, Figma akan diaplikasikan untuk merancang *prototype* modul *Sales* dan *Finance* serta *landing page* berbasis web pada ERP *Systems* Bless Auto Care. Penggunaan Figma memungkinkan proses desain antarmuka pengguna yang lebih interaktif, di mana desainer dapat berkolaborasi erat dengan pemangku kepentingan untuk memastikan bahwa *prototype* yang dihasilkan sesuai dengan kebutuhan bisnis dan memiliki tingkat kegunaan yang tinggi bagi pengguna akhir. Kemampuan Figma dalam memfasilitasi pengujian dan penyempurnaan *prototype* sebelum implementasi sistem dapat secara signifikan mengurangi risiko kesalahan desain yang berpotensi memengaruhi fungsionalitas sistem ERP.

2.5 *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah bahasa standar yang digunakan untuk memvisualisasikan, menspesifikasi, membuat, dan mendokumentasikan sistem perangkat lunak. UML dikembangkan untuk membantu pengembang dalam merancang dan mengkomunikasikan struktur dan perilaku sistem yang kompleks dengan menggunakan serangkaian diagram. Melalui UML, tim pengembang dapat memiliki pandangan yang lebih terstruktur tentang bagaimana sistem dibangun dan beroperasi, sehingga mengurangi risiko kesalahpahaman dalam pengembangan sistem [30].

UML sendiri menyediakan berbagai jenis diagram yang terbagi dalam dua kategori utama: diagram struktur dan diagram perilaku. Diagram struktur, seperti *class* diagram, *component* diagram, dan *object* diagram, berfungsi untuk menggambarkan struktur statis dari sistem, seperti hubungan antar kelas, atribut, dan metode yang dimiliki. Sementara itu, diagram perilaku, *seperti use case* diagram, *sequence* diagram, dan *activity* diagram, menggambarkan bagaimana sistem berinteraksi dengan pengguna dan menunjukkan urutan alur kerja atau proses yang terjadi dalam sistem. Dengan adanya berbagai diagram ini, UML memungkinkan pengembang untuk memahami berbagai aspek sistem dari sudut pandang yang berbeda.

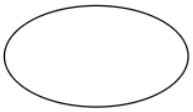
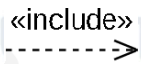
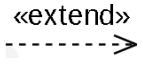
Penggunaan UML memiliki beberapa keuntungan dalam pengembangan perangkat lunak. Pertama, UML membantu tim untuk berkomunikasi lebih efektif dan memastikan setiap anggota memiliki pemahaman yang sama tentang desain sistem [30]. Kedua, UML membantu mempercepat proses desain dengan memberikan kerangka kerja yang jelas untuk menggambarkan sistem. Ketiga, UML memungkinkan dokumentasi yang lebih baik, sehingga ketika sistem perlu diperbarui atau diperbaiki di masa depan, pengembang dapat merujuk pada dokumentasi ini untuk memahami desain asli dari sistem. Dalam praktiknya, UML dapat digunakan dalam berbagai metode pengembangan perangkat lunak, seperti *waterfall*, *agile*, dan *iterative development*. Fleksibilitas ini membuat UML menjadi alat yang esensial bagi para

pengembang perangkat lunak, analis, dan desainer sistem yang ingin menciptakan solusi yang efisien dan terstruktur.

2.5.1 *Use Case Diagram*

Use Case dalam konteks rekayasa perangkat lunak adalah elemen dalam pemodelan Unified Modeling Language (UML) yang mewakili sekumpulan interaksi atau transaksi yang dilakukan oleh sistem dalam rangka menyediakan suatu layanan atau fungsi yang terlihat oleh pengguna eksternal (aktor). Penggunaan *Use Case* bertujuan untuk mendefinisikan serangkaian perilaku tanpa harus memperlihatkan struktur internal sistem yang mendasarinya. *Use Case* menggambarkan semua variasi dalam interaksi antara sistem dan aktor, termasuk perilaku normal, variasi, serta kondisi-kondisi pengecualian atau error yang mungkin terjadi [30]. Berikut merupakan table mengenai *Use Case Diagram*.

Tabel 2.2 *Use Case Diagram*

Nama	Simbol	Keterangan
Actor		Merujuk pada individu atau entitas yang berinteraksi dengan sistem untuk menjalankan sebuah proses.
Use Case		Merupakan bentuk dari fungsi pada sebuah sistem
Include		Merupakan simbol dari keseluruhan fungsionalitas use case lain
Extend		Merupakan petunjuk suatu use case yang merupakan tambahan fungsionalitas dari use case lainnya apabila kondisi tertentu

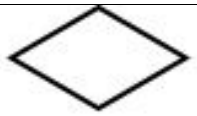
Nama	Simbol	Keterangan
		terpenuhi
Association		Merupakan penghubung anrara Actor dengan Use Case "Association"
Generalization		Mengkoneksikan 2 actor & case. "Generalization"

2.5.2 *Activity Diagram*

Activity diagram adalah representasi *visual* yang menunjukkan serangkaian aktivitas yang terjadi dalam sebuah sistem. Dengan menampilkan setiap aktivitas secara berurutan, diagram ini membantu dalam memahami bagaimana suatu sistem bekerja dan bagaimana setiap elemen dalam sistem berinteraksi. Selain itu, activity diagram juga berguna untuk mengidentifikasi potensi hambatan atau ketidakefisienan dalam alur kerja. Dengan analisis yang lebih mendalam, diagram ini dapat membantu tim pengembang atau manajer proyek untuk merancang proses yang lebih efisien, memastikan bahwa semua langkah dilakukan dengan cara yang optimal, serta meningkatkan koordinasi antara berbagai elemen dalam sistem [30]. Berikut merupakan table mengenai *Activity Diagram*:

Tabel 2.3 *Activity Diagram*

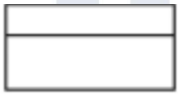
Nama	Simbol	Keterangan
Initial State/Status Awal		Merupakan awalan dari aliran *activity* diagram
Activity/Aktivitas		Aktivitas yang dilakukan dalam aliran kerja

Nama	Simbol	Keterangan
Decision		Merupakan titik keputusan dimana terdapat lebih dari satu kondisi
Final State/Status Akhir		Merupakan bagian akhir dari suatu proses.

2.5.3 *Class Diagram*

Class Diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan struktur statis dari suatu sistem. Diagram ini menggambarkan berbagai class, yaitu representasi dari objek atau entitas dalam sistem, beserta atribut dan metode yang dimilikinya. Selain itu, hubungan antara *class*, seperti asosiasi, generalisasi, dan agregasi, juga direpresentasikan dalam diagram ini. *Class Diagram* berfungsi untuk memberikan gambaran lengkap mengenai struktur internal sistem, sehingga membantu pengembang memahami bagaimana berbagai komponen sistem saling berinteraksi [30]. Berikut merupakan table mengenai *Class Diagram*:

Tabel 2.4 *Class Diagram*

Nama	Simbol	Keterangan
Class		Mewakili sekelompok objek yang memiliki karakteristik, fungsi, dan properti yang serupa.
Nary Association		Digunakan untuk mencegah asosiasi yang melibatkan hanya dua objek, memberikan fleksibilitas dalam menghubungkan lebih banyak objek.
Collaboration		Menentukan urutan tindakan dalam sistem yang dirancang untuk menghasilkan output yang terukur dan bermanfaat bagi aktor.

Nama	Simbol	Keterangan
Generalization		Menunjukkan hubungan antara objek anak (descendent) dengan objek induk (ancestor), di mana objek anak mewarisi perilaku dan struktur dari objek induk.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA