BAB II

LANDASAN TEORI

2.1 Penelitian Terkait

Tabel 2. 1 Penelitian Terkait

Penelitian Terkait		
1.	**Nama Jurnal**	Jurnal Ilmiah Teknik Mesin, Elektro Dan Komputer, 3(3), 470–489. https://doi.org/10.51903/juritek.v3i3.2304 [6]
	Tahun	2023
	Judul	Rancang Bangun Sistem Informasi Pengelolaan Inventaris Berbasis WEB pada PT BPRS Amanah Insan Cita
	Penulis	Ainun Mardiah Hasibuan, Fadhilah Ramadhani Nasution, Muhammad Ikhsan Rifki
	Permasalahan	Sistem yang dikembangkan mungkin tidak sepenuhnya memenuhi tuntutan bisnis karena persyaratan sistem tidak selalu dinyatakan atau dipahami dengan jelas. Integrasi dengan sistem yang ada saat ini menimbulkan kesulitan tambahan, terutama ketika ada perbedaan dalam format data atau teknologi. Kinerja sistem juga dapat dipengaruhi oleh keterbatasan infrastruktur teknologi, seperti server yang dapat diandalkan dan koneksi internet yang stabil.
	Hasil	Teknik *waterfall* digunakan untuk membuat sistem informasi manajemen persediaan karena memberikan informasi yang akurat dan tepat waktu mengenai ketersediaan persediaan di PT BPRS Amanah Insan Cita dan menampilkan langkah-langkah yang sesuai dengan

Penelitian Terkait		
		kondisi lapangan serta dapat mempermudah dalam mencari produk yang akan digunakan.
2.	**Nama Jurnal**	*Computer Based Information System Journal*, *11*(2), 1–8. https://doi.org/10.33884/cbis.v11i2.7385 [5].
		2023
	Judul	Rancang Bangun Sistem Informasi Manajemen Inventaris Berbasis *Website* Menggunakan Metode SDLC
	Penulis	Nadhia, A. N., & Suwarno, S
	Permasalahan	Masalah manajemen inventaris dapat diatasi dengan pemantauan stok, pemeriksaan kerusakan dan perbaikan yang terperinci, pelacakan penyimpanan yang akurat, dan alat bantu lainnya.
	Hasil	Pendekatan *System Development Life Cycle* (SDLC) digunakan dalam penelitian ini bersama dengan model prototipe. Sistem manajemen inventaris yang dapat mendokumentasikan proses pengelolaan barang inventaris secara lengkap merupakan hasil dari penelitian ini, sehingga menjadi lebih tepat, bermanfaat, dan efisien.
3.	**Nama Jurnal**	JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika) https://jurnal.stkippgritulungagung.ac.id/index.php/jipi [7]
	Tahun	2025
	Judul	Rancang Bangun Sistem Informasi Manajemen Layanan Magang Di Diskominfo Kabupaten Purwakarta Berbasis Web Dengan Menggunakan Metode *Extreme Programming*
	Penulis	Asep Yusapra Salim, Franciskus Antonius Alijoyo

Penelitian Terkait		
	Permasalahan	Berdasarkan hasil observasi yang dilakukan, manajemen pengelolaan dalam pendaftaran magang di Bidang Aplikasi Teknologi Informasi (APTIKA) Dinas Komunikasi dan Informatika (DISKOMINFO) sebagian besar masih menggunakan sistem manual.
	Hasil	Hasil penelitian menunjukkan bahwa sistem informasi manajemen layanan magang ini berhasil membantu mahasiswa dalam melakukan pendaftaran *online* dan mendapatkan informasi mengenai kesempatan magang. Teknologi ini dirancang untuk membuat proses kerja magang menjadi lebih efektif dan sistematis, serta membantu mahasiswa dan DISKOMINFO dalam hal pengelolaan program kerja magang.
4.	**Nama Jurnal**	*Journal of Information System*, vol. 8, no. 1, pp. 80–89, Jun. 2023 [8]
	Tahun	2023
	Judul	*The Development of web-based Cashier and Inventory Information Systems using Prototyping Model on Micro, Small, and Medium Enterprise (MSMEs) in Indonesia*
	Penulis	Tan Andreas Dwi Pratama Subianto & Jansen Wiratama
	Permasalahan	Ada beberapa kesulitan dalam proses bisnis saat ini, termasuk kesalahan informasi saat memverifikasi stok barang dan mencatat transaksi penjualan.
	Hasil	Penelitian ini mengembangkan sistem informasi untuk kasir dan persediaan pada UMKM Putra Anugrah Sejahtera. Sebuah sistem yang akan membantu *user* atau pengguna dalam mengelola data toko agar pekerjaan lebih efisien dalam pencarian dan penginputan stok barang, serta kemampuan untuk menghasilkan laporan keuangan harian dengan cepat.

Penelitian Terkait		
5.	**Nama Jurnal**	Vol. 1 No. 1 (2021): *Indonesian Journal of Informatic Research and Software Engineering* [9]
	Tahun	2021
	Judul	Rancang Bangun Sistem Informasi Inventaris Barang Dinas Komunikasi dan Informatika Indragiri Hulu
	Penulis	Muhammad Irzan, Depa Sutriyono
	Permasalahan	Dinas Komunikasi dan Informatika masih membuat laporan data inventaris secara manual, menggunakan buku dan *Microsoft Office Excel*. Beberapa masalah yang disebutkan di atas dapat menimbulkan kesulitan seperti kesalahan dalam pembuatan laporan, data penting yang hilang, rentan terhadap manipulasi, dan kesalahan manusia lainnya.
	Hasil	Metodologi *waterfall* digunakan untuk mengembangkan model ini secara sistematis, dimulai dari tingkat kebutuhan sistem dan berlanjut ke analisis, desain, pengkodean, pengujian, dan pemeliharaan.
6.	**Nama Jurnal**	Jurnal Teknik Informatika Unika St. Thomas (JTIUST), Volume 05 Nomor 01, Juni 2020, pISSN : 2548-1916, eISSN : 2657-1501 [10]
	Tahun	2020
	Judul	Sistem Informasi Kepegawaian Dinas Pendidikan menggunakan Metode *Waterfall*
	Penulis	Salmah Fatihatul Ula
	Permasalahan	Saat ini, sistem pelayanan yang diberikan kepada pegawai UPT Puskesmas Distrik Unter Iwes kurang optimal, dengan sistem kepegawaian yang masih manual. Data-data terkait kepegawaian diolah dengan menggunakan aplikasi perkantoran seperti Microsoft Word dan Excel, selain itu juga masih menggunakan cara manual untuk

Penelitian Terkait		
		mencari informasi seorang pegawai atau data-data yang berkaitan dengan riwayat kepangkatan dan jabatan pegawai.
	Hasil	Penggunaan sistem informasi kepegawaian Dinas Pendidikan dengan menggunakan teknik *waterfall* dapat memberikan manfaat sistem informasi terhadap proses kerja dalam hal peningkatan kinerja, kemudahan dalam pembuatan laporan, dan informasi yang lebih akurat.
7.	**Nama Jurnal**	*International Journal of Science, Technology & Management*, 4(6), 1653-1659. https://doi.org/10.46729/ijstm.v4i6.976 [11]
	Tahun	2023
	Judul	*Optimizing Educational Institutions: Web-Based Document Management*
	Penulis	Muhammad Hanif Triyana & Melissa Indah Fianty
	Permasalahan	SMAIT Asy-Syukriyyah masih menggunakan metode penyimpanan dokumen tradisional. Akibatnya, SMAIT Asy-Syukriyyah membutuhkan sebuah media penyimpanan dokumen untuk mendukung kegiatan belajar mengajar, serta pengurangan penggunaan dokumen fisik untuk membatasi bahaya kerusakan dokumen karena faktor lingkungan atau kesalahan manusia.
	Hasil	Guru dapat dengan cepat mengakses dokumen yang berhubungan dengan pembelajaran dan diharapkan dapat berkontribusi dalam proses belajar mengajar dengan menggunakan Sistem Manajemen Dokumen, yang salah satu tugasnya adalah menyimpan dokumen dalam bentuk digital. Selain itu, dengan menggunakan metode *waterfall*, dokumen pengembangan aplikasi menjadi lebih

Penelitian Terkait		
		terorganisir, karena setiap langkah harus diselesaikan secara keseluruhan sebelum beralih ke langkah berikutnya.
8.	**Nama Jurnal**	Kumpulan jurnal Ilmu Komputer (KLIK) Volume 09, No 1 Februari 2022 ISSN: 2406-7857 DOI: http://dx.doi.org/10.20527/klik.v9i1.448 [12]
	Tahun	2022
	Judul	Perancangan Sistem Informasi Inventaris Berbasis *Website* Menggunakan Metode *Waterfall*
	Penulis	Darmansah, Sisilia Rahma Widiasari, Raswini, Marcel Agustine Bacsafra
	Permasalahan	Dinas Perhubungan Kabupaten Banyumas merupakan salah satu instansi pemerintah yang telah menggunakan sistem informasi, meskipun pada bagian pendataan inventaris masih menggunakan sistem tradisional yaitu Microsoft Excel.
	Hasil	*Website* Inventaris Dinas Perhubungan Kabupaten Banyumas dirancang dengan menggunakan pendekatan *Waterfall*. Teknik *Waterfall* melibatkan pengembangan sistem perangkat lunak dalam beberapa tahapan, termasuk perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan. Pembuatan sistem ini dimaksudkan untuk memudahkan staf dalam menangani data dan memantau barang inventaris dengan lebih cepat dan efisien.
9.	**Nama Jurnal**	Jurnal Sistem Informasi Dan E-Bisnis, 4(2), 112–118. https://doi.org/10.54650/jusibi.v4i2.453 [13]
	Tahun	2022
	Judul	Metode SDLC *Waterfall* Pada Rancang Bangun Sistem Informasi Sekolah SMP Negeri 10 Kaur

Penelitian Terkait		
	Penulis	Kirman & Erdi Epta Saputra
	Permasalahan	Kesulitan yang dihadapi dunia pendidikan, khususnya sekolah dasar, sekolah menengah pertama, dan sekolah menengah atas, adalah memodifikasi sistem pembelajaran yang berdampak pada sistem promosi sekolah.
	Hasil	Pengembangan prototipe sistem informasi ini berhasil, dengan menggunakan pengembangan metode SDLC *Waterfall*, dibuktikan dengan hasil pengujian *black box* yaitu keseluruhan sistem yang dikembangkan tidak terdapat *bug/error* pada sistem.

Berdasarkan penelitian terkait yang ditunjukkan pada tabel 2.1, tinjauan terhadap penelitian sebelumnya menunjukkan konsensus bahwa transformasi dari sistem manual ke sistem informasi terotomatisasi secara konsisten menghasilkan peningkatan efisiensi dan akurasi data. Berbagai penelitian terdahulu menyoroti bagaimana sistem berbasis web berhasil mengatasi risiko kesalahan manusia, kehilangan data, dan kelambatan pelaporan yang melekat pada proses manual berbasis *spreadsheet* atau pencatatan fisik [9][12]. Dari sisi metodologi pengembangan, model *Waterfall* sering kali menjadi pilihan utama dalam konteks instansi formal karena pendekatannya yang terstruktur [6][10], sementara pendekatan interaktif seperti *Protoyping* dipilih dalam lingkungan dalam lingkungan menuntut fleksibilitas dan umpan balik pengguna yang cepat, seperti pada Usaha Kecil dan Menengah atau UKM [8]. Meskipun demikian, analisis yang lebih mendalam terhadap kumpulan penelitian tersebut mengungkapkan adanya celah penelitian (*research gap*) yang signifikan. Pertama, mayoritas studi kasus yang ada cenderung berfokus pada manajemen inventaris untuk aset atau barang fisik. Akibatnya, pembahasan mengenai tantangan unik yang dihadapi dalam pengelolaan produk digital masih sangat terbatas. Kedua, terdapat keterbatasan studi kasus yang secara detail mendokumentasikan penerapan metode *Waterfall* yang terstruktur di dalam lingkungan perusahaan yang dinamis, terutama yang bergerak di sektor digital. Banyak penelitian yang ada cenderung hanya melaporkan

hasil akhir tanpa menjabarkan proses analitis di baliknya, seperti bagaimana kebutuhan unik yang diterjemahkan menjadi desain sistem yang efektif. Oleh karena itu, penelitian ini diposisikan secara strategis untuk mengisi kombinasi celah tersebut. Penelitian ini tidak hanya bertujuan membangun sistem inventaris yang fungsional, tetapi secara spesifik mengambil studi kasus pada sebuah perusahaan di sektor produk digital dan menganalisis secara detail penerapan metode SDLC *Waterfall* dari awal hingga akhir. Dengan demikian, kontribusi utama dari penelitian ini adalah menyediakan sebuah studi kasus yang terdokumentasi dengan baik, yang dapat menjadi referensi aplikatif mengenai bagaimana pendekatan pengembangan terstruktur dapat diadaptasi secara efektif untuk menjawab kebutuhan nyata pada segmen perusahaan digital yang masih kurang terwakili dalam literatur.

2.2 Teori tentang Topik Skripsi

2.2.1 Rancang Bangun

Rancang bangun sistem adalah suatu proses yang sistematis dan terstruktur dalam merancang, mengembangkan, dan mengimplementasikan sistem informasi yang efektif dan efisien. Rancang bangun dapat diartikan sebagai proses yang bertujuan untuk menciptakan dan mengembangkan sebuah aplikasi atau sistem yang sebelumnya belum ada di suatu instansi atau objek tertentu [14]. Proses ini mencakup beberapa tahapan, mulai dari analisis kebutuhan, perancangan arsitektur sistem, pengembangan perangkat lunak, pengujian, hingga implementasi dan pemeliharaan sistem [15]. Dalam setiap tahapan, penting untuk mempertimbangkan kebutuhan pengguna dan tujuan organisasi. Misalnya, dalam analisis kebutuhan, pengembang dapat melakukan wawancara dengan pemangku kepentingan untuk mendokumentasikan kebutuhan fungsional, seperti fitur untuk mencatat barang masuk/keluar dan laporan stok. Rancang bangun sistem merupakan langkah awal yang krusial dalam menciptakan sistem yang dapat memenuhi kebutuhan informasi yang dinamis dan kompleks.

2.2.2 *Website*

Website adalah sekumpulan halaman yang saling terhubung dan dapat diakses melalui internet, yang berfungsi untuk menyajikan informasi, layanan, atau produk kepada pengguna. Menurut Elgamar, *website* dapat didefinisikan sebagai sebuah media yang terdiri dari sejumlah halaman yang saling terhubung satu sama lain [16]. Halaman-halaman ini berfungsi untuk menyajikan berbagai jenis informasi, baik berupa teks, gambar, maupun multimedia, yang dapat diakses oleh pengguna melalui internet. *Website* berperan penting dalam menyediakan platform untuk komunikasi, interaksi, dan penyebaran informasi, baik untuk individu, organisasi, maupun perusahaan. Dengan struktur yang terorganisir, *website* memungkinkan pengguna untuk menjelajahi konten dengan mudah dan menemukan informasi yang mereka butuhkah. Selain itu, *website* juga dapat berfungsi sebagai alat pemasaran dan promosi, memberikan kesempatan bagi pemiliknya untuk menjangkau audiensi yang lebih luas dan meningkatkan visibilitas di dunia digital.

Website terdiri dari dua komponen utama, yaitu *frontend* dan *backend*. *Frontend* adalah bagian dari *website* yang berinteraksi langsung dengan pengguna, mencakup elemen visual seperti desain, *layout*, dan antarmuka pengguna yang dibangun menggunakan teknologi seperti *HTML, CSS*, dan *JavaScript*. Sementara itu, *backend* adalah bagian yang tidak terlihat oleh pengguna, bertanggung jawab untuk mengelola data, logika bisnis, dan komunikasi dengan server. *Backend* biasanya dibangun menggunakan bahasa pemrograman seperti *PHP, Python, atau Node.js*, serta melibatkan penggunaan *database* untuk menyimpan informasi. Dalam konteks *website*, terdapat dua jenis utama, yaitu *website* statis dan dinamis. *Website* statis adalah halaman yang kontennya tetap dan tidak

"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

berubah kecuali dilakukan pengeditan manual, sehingga lebih sederhana dan cepat diakses. Di sisi lain, *website* dinamis dapat menampilkan konten yang berubah secara otomatis berdasarkan interaksi pengguna atau data yang diambil dari *database*, memberikan pengalaman yang lebih interaktif dan personal. Salah satu manfaat signifikan dari sistem berbasis web adalah kemampuannya dalam manajemen inventaris. Dengan menggunakan sistem manajemen inventaris berbasis web, organisasi dapat memantau dan mengelola stok barang secara *real-time*, meningkatkan efisiensi operasional, mengurangi kesalahan manusia, serta memudahkan akses informasi dari berbagai lokasi. Hal ini memungkinkan pengambilan keputusan yang lebih cepat dan akurat, serta meningkatkan transparansi dalam pengelolaan inventaris.

2.2.3 *Inventory Management*

Inventory management adalah proses pengelolaan persediaan barang yang mencakup pencatatan, pemantauan, dan pengendalian stok. Sistem informasi inventaris dirancang untuk mengurangi kesalahan manusia dalam pencatatan data, meningkatkan akurasi data inventaris, dan mendukung pengambilan keputusan berdasarkan data stok yang selalu diperbarui secara *real-time*. Dengan penerapan sistem berbasis web, perusahaan dapat mengakses informasi persediaan secara langsung, yang memungkinkan manajer untuk melakukan analisis yang lebih baik dan pengambilan keputusan yang lebih cepat. Hal ini sangat penting dalam menjaga ketersediaan produk dan meminimalkan biaya yang terkait dengan penyimpanan dan pengelolaan barang. Sebuah studi yang dipublikasikan dalam jurnal "*International Journal of Production Economics*" menunjukkan bahwa perusahaan yang mengadopsi sistem manajemen persediaan berbasis web mengalami peningkatan signifikan dalam akurasi data persediaan dan pengurangan biaya operasional [17]. Dengan demikian, pengelolaan persediaan yang efektif tidak hanya

berkontribusi pada efisiensi operasional, tetapi juga pada kepuasan pelanggan melalui ketersediaan produk yang lebih baik.

2.3 Teori tentang Framework/Algoritma yang digunakan

2.3.1 Metode *Waterfall*

Metode *Waterfall* adalah metode pengembangan perangkat lunak yang dikenal juga *System Development Life Cycle* (SDLC) yaitu kerangka kerja yang digunakan untuk merencanakan, mengembangkan, menguji, dan memelihara sistem informasi. Tujuannya adalah memastikan bahwa sistem yang dikembangkan memenuhi kebutuhan pengguna dan berfungsi secara efektif[18]. SDLC bertujuan untuk menghasilkan *output* sistem berkualitas tinggi yang memenuhi atau melebihi harapan pelanggan, berdasarkan kebutuhan pelanggan dengan memberikan sistem yang bergerak melalui setiap tahap yang jelas, dalam jangka waktu yang dijadwalkan dan menggunakan perkiraan biaya bila ada [19]. Tahapan metode *waterfall* umumnya meliputi perencanaan, analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Keunggulan utama dari metode *waterfall* adalah kemampuannya untuk memberikan struktur yang jelas dan terorganisir dalam proses pengembangan, sehingga memudahkan tim pengembang untuk fokus pada satu tahap pada satu waktu [6]. Hal ini juga membantu dalam mengidentifikasi dan mengatasi masalah lebih awal dalam proses pengembangan, yang dapat mengurangi risiko kesalahan dan meningkatkan kualitas sistem yang dihasilkan.

2.3.2 Metode *Agile*

Metode Agile pada intinya bukanlah sebuah metodologi yang kaku, melainkan sebuah pola pikir (mindset) yang berlandaskan pada empat nilai dan dua belas prinsip dalam Manifesto Agile [20]. Pendekatan ini

dirancang untuk dapat berhasil dalam lingkungan proyek yang kompleks dan penuh ketidakpastian dengan menggunakan siklus hidup yang bersifat adaptif, yang secara spesifik menggabungkan karakteristik iteratif dan inkremental. Artinya, pengembangan dilakukan dalam siklus-siklus pendek berdurasi tetap (iterasi atau sprint) yang di setiap akhirnya menghasilkan tambahan fungsionalitas produk yang berfungsi dan berpotensi untuk dirilis (*increment*). Tujuan utama dari pendekatan ini adalah untuk memaksimalkan pengiriman nilai (*value*) kepada pelanggan dan merespons perubahan secara efektif, berbeda dengan pendekatan prediktif (seperti *Waterfall*) yang mengandalkan rencana tetap di awal. Untuk mencapai hal ini, *Agile* sangat menekankan pentingnya kolaborasi erat dengan pemangku kepentingan, umpan balik berkelanjutan, dan pemberdayaan tim untuk mengatur pekerjaannya sendiri dalam mencapai tujuan.

2.3.3 Metode *Prototyping*

Metode *Prototyping* adalah sebuah paradigma pengembangan perangkat lunak yang idealnya digunakan ketika kebutuhan awal suatu proyek tidak terdefinisi dengan baik [21]. Proses ini berjalan secara *iteratif*, diawali dengan komunikasi untuk mengumpulkan kebutuhan dasar, diikuti oleh perancangan cepat yang berfokus pada aspek antarmuka pengguna. Berdasarkan rancangan tersebut, sebuah model kerja (prototipe) dibangun dan diserahkan kepada pengguna untuk dievaluasi. Umpan balik yang diterima kemudian digunakan untuk memperbaiki dan menyempurnakan prototipe dalam siklus berulang hingga kebutuhan menjadi jelas. *Pressman dan Maxim* [21]membedakan dua pendekatan utama: prototipe sekali pakai (*throwaway*), yang dibuat untuk tujuan pembelajaran lalu dibuang, dan prototipe evolusioner (*evolutionary*), yang secara bertahap dikembangkan menjadi produk akhir. Meskipun metode ini sangat efektif untuk mengurangi risiko kesalahpahaman kebutuhan dan meningkatkan keterlibatan pengguna, ia memiliki

kelemahan, seperti potensi pengembang mengambil jalan pintas dalam kualitas teknis dan timbulnya ekspektasi yang tidak realistis dari pengguna.

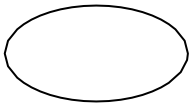
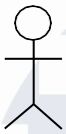
2.3.4 *Unified Modeling Language* (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang digunakan untuk melakukan spesifikasi, visualisasi, konstruksi, dan dokumentasi dari artefak-artefak pada sebuah sistem perangkat lunak. Menurut para penggagasnya Roger S. Pressman, Bruce R. Maxim [21] , tujuan utama UML adalah untuk menyediakan sebuah bahasa yang umum dan konsisten bagi semua pemangku kepentingan—mulai dari analis bisnis, perancang sistem, hingga pengembangan—untuk memodelkan sistem yang kompleks dari berbagai sudut pandang yang berbeda namun tetap saling terhubung. Dalam konteks penelitian ini, UML akan digunakan sebagai alat bantu utama dalam tahapan analisis dan perancangan sistem yang diatur oleh metodologi SDLC *Waterfall*. UML menyediakan serangkaian diagram yang dapat dikelompokkan ke dalam dua kategori utama: Diagram Perilaku (*Behavior Diagrams*) yang menggambarkan aspek dinamis sistem, dan Diagram Struktur (*Structure Diagrams*) yang menggambarkan arsitektur statis sistem. Dengan menggunakan diagram-diagram terpilih dari kedua kategori ini, perancangan sistem informasi inventaris dapat dimodelkan secara komprehensif.

2.3.4.1 *Use Case Diagram*

Use Case Diagram termasuk dalam kategori Diagram Perilaku. Diagram ini berfungsi untuk menggambarkan fungsionalitas apa saja yang disediakan oleh sistem serta siapa saja yang berinteraksi dengan fungsionalitas tersebut. Fokus utamanya adalah pada kebutuhan sistem dari perspektif pengguna eksternal, tanpa merinci bagaimana cara kerja internal sistem [21].

Tabel 2. 2 *Use Case Diagram*

No.	Gambar	Nama	Keterangan
1.		*Use Case*	Digambarkan sebagai elips, merepresentasikan sebuah fungsionalitas spesifik yang memberikan nilai tambah bagi *actor*.
2.		*Actor*	Mewakili peran pengguna atau sistem eksternal lain yang berinteraksi dengan sistem. Meskipun sering digambarkan dengan ikon manusia (*stick figure*), *actor* tidak selalu manusia; bisa juga berupa sistem lain yang membutuhkan atau memberikan data.
3.	<<*Include*>> ----------►	*Include*	Panah putus-putus dengan label <<*include*>> yang mengarah dari *use case* dasar ke *use case* yang di-inklusi. Ini menandakan bahwa sebuah *use case* akan selalu menjalankan fungsionalitas dari *use case* lain sebagai bagian dari alurnya.
4.	<<*Extend*>> ----------►	*Extend*	Panah putus-putus dengan label <<*extend*>> yang mengarah dari *use case* tambahan ke *use case* dasar.

"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

No.	Gambar	Nama	Keterangan
			Ini menandakan fungsionalitas opsional yang hanya berjalan jika kondisi tertentu terpenuhi.
5.		Generalisasi atau *Generalization*	Panah dengan ujung segitiga kosong yang menghubungkan dua *actor* atau dua *use case*. Ini menunjukkan hubungan umum-khusus (induk-anak), di mana elemen anak mewarisi karakteristik dari elemen induk namun memiliki perilaku yang lebih spesifik.

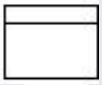
2.3.4.2 *Activity Diagram*

Activity Diagram, atau diagram aktivitas, adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan aliran kerja atau aktivitas dalam suatu sistem atau proses bisnis. Diagram ini menyajikan serangkaian kegiatan, tindakan, dan keputusan yang terjadi sepanjang waktu, sehingga memudahkan pemahaman mengenai proses yang berlangsung dalam sistem tersebut [21]. Secara umum, *Activity Diagram* menggambarkan bagaimana aliran-aliran aktivitas di dalam sistem yang sedang dirancang, termasuk bagaimana aliran-aliran tersebut berawal, keputusan yang mungkin terjadi, dan bagaimana aliran-aliran tersebut berakhir. Hal ini membantu dalam memvisualisasikan proses bisnis atau alur kerja secara lebih jelas dan terstruktur.

Tabel 2. 3 *Activity Diagram*

No.	Gambar	Nama	Keterangan
1.		Status Awal	Merupakan titik awal dari aliran aktivitas dalam sebuah diagram. Biasanya digambarkan dengan sebuah lingkaran hitam kecil.
2.		Aktivitas	Menunjukkan langkah atau aktivitas yang sedang berlangsung dalam proses. Aktivitas ini digambarkan sebagai persegi panjang dengan sudut melengkung, dan sering kali berisi deskripsi kegiatan yang dilakukan.
3.		Penggabungan atau *Join*	Titik di mana beberapa aliran aktivitas yang paralel digabung menjadi satu aliran. Biasanya digambarkan dengan sebuah garis horizontal atau vertikal yang menghubungkan beberapa aliran aktivitas yang masuk.
4.		Percabangan atau *Decision*	Titik di mana aliran aktivitas bercabang berdasarkan suatu kondisi tertentu. Biasanya digambarkan sebagai bentuk berlian.

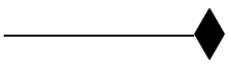
"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

No.	Gambar	Nama	Keterangan
5.		Status Akhir	Titik di mana aliran aktivitas bercabang berdasarkan suatu kondisi tertentu. Biasanya digambarkan sebagai bentuk berlian.
6.		*Swimlane*	*Swimlane* adalah pembagian dalam diagram aktivitas yang mengelompokkan aktivitas berdasarkan aktor atau bagian dari sistem. Setiap *swimlane* biasanya diberi label untuk menunjukkan siapa atau apa yang bertanggung jawab untuk aktivitas dalam *lane* tersebut.

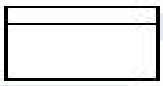
2.3.4.3 *Class Diagram*

Class Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan struktur statis dari sistem dengan menunjukkan kelas-kelas dalam sistem dan hubungan antar kelas tersebut [21]. *Class Diagram* sering digunakan dalam tahap perancangan sistem untuk memodelkan data dan relasi antar data.

Tabel 2. 4 *Class Diagram*

No.	Gambar	Nama	Keterangan
1.		*Composition*	*Composition* adalah jenis hubungan antar kelas yang lebih kuat daripada agregasi. Ini

"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

No.	Gambar	Nama	Keterangan
			menggambarkan hubungan "bagian-dari" di mana objek dari kelas yang satu tidak dapat ada tanpa kelas lainnya. Dengan kata lain, jika objek dari kelas "*whole*" dihapus, maka objek dari kelas "*part*" juga akan ikut terhapus.
2.		*Class*	Kelas adalah *blueprint* atau cetak biru yang mendefinisikan atribut dan metode untuk objek. Dalam *class* diagram, kelas digambarkan sebagai kotak yang terbagi menjadi tiga bagian: bagian atas untuk nama kelas, bagian tengah untuk atribut, dan bagian bawah untuk metode (operasi).
3.		*Aggregation*	*Aggregation* adalah hubungan antara dua kelas yang lebih lemah daripada komposisi, namun masih menggambarkan hubungan "bagian-dari".
4.		*Dependency*	*Dependency* adalah hubungan antara dua kelas di mana satu kelas (klien)

"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

No.	Gambar	Nama	Keterangan
			bergantung pada kelas lain (penyedia).
5.		*Association*	*Association* adalah hubungan yang menggambarkan hubungan antara dua kelas yang lebih umum dan dapat bersifat satu-ke-satu, satu-ke-banyak, atau banyak-ke-banyak.

2.3.5 HTML (*HyperText Markup Language)*

HTML adalah bahasa *markup* yang digunakan untuk membuat struktur dan konten halaman web. Hal ini dapat digunakan untuk menentukan elemen seperti teks, gambar, tautan, tabel, dan formulir pada halaman web [22]. HTML menggunakan *tag* untuk menentukan elemen-elemen seperti teks, grafik, formulir *input*, tombol, dan sebagainya. HTML bersifat statis, tetapi dapat ditingkatkan dengan CSS dan *JavaScript* untuk membuat halaman lebih menarik secara visual dan interaktif. HTML adalah komponen penting dalam pengembangan web yang berfungsi sebagai fondasi untuk integrasi teknologi web lainnya seperti CSS dan *JavaScript*. HTML mencakup elemen semantik seperti <*section*>, <*article*>, dan <*nav*> yang membantu menciptakan situs yang terstruktur dan ramah mesin pencari [23].

2.3.6 *PHP (Preprocessor Hypertext)*

PHP (*Preprocessor Hypertext*) adalah bahasa pemrograman yang dapat digunakan untuk membuat situs web dan aplikasi web yang dinamis [24]. PHP dapat menangani permintaan dari *form*, melakukan validasi di sisi server, dan menjalankan proses CRUD ke *database MySQL*. Dengan meningkatnya kebutuhan akan aplikasi web yang interaktif dan

responsif, PHP menawarkan solusi yang efisien dan fleksibel. PHP dapat digunakan untuk mengelola konten, mengakses *database*, dan menghasilkan halaman web secara dinamis. Selain itu, berbagai *database*, termasuk *MySQL, PostgreSQL*, dan *SQLite*, didukung oleh PHP, sehingga memungkinkan pemilihan solusi yang paling sesuai dengan kebutuhan pengembang.

2.3.7 *MySQL*

MySQL adalah perangkat lunak sistem manajemen basis data relasional yang bersifat *open source* dan berbasis *Structured Query Language* (SQL) [25]. Secara umum, *MySQL* digunakan untuk membuat dan mengelola basis data yang diperlukan dalam pengembangan aplikasi atau *website* pada sebuah server. Dengan dukungan *multi-user*, *MySQL* sangat cocok untuk pengelolaan basis data secara kolaboratif dalam sebuah tim. Cara kerja *MySQL* melibatkan penerimaan, penyimpanan, modifikasi, dan pengiriman data. Proses ini dimulai ketika *client* mengirimkan permintaan melalui aplikasi dengan instruksi tertentu, lalu server menerima dan memproses permintaan tersebut untuk memberikan informasi yang diminta [26].

2.4 Teori tentang *tools/software* yang digunakan

2.4.1 Visual Studio Code (VSC)

Visual Studio Code adalah editor kode milik Microsoft yang banyak digunakan oleh pengembang karena fitur lengkap dan persyaratan sistem yang ringan [27]. Selain menawarkan berbagai fitur bawaan, Visual Studio Code juga menyediakan *marketplace* ekstensi yang memungkinkan pengguna untuk menambahkan fitur tambahan sesuai kebutuhan mereka dalam pengembangan perangkat lunak. *Software* ini mendukung banyak bahasa pemrograman seperti *PHP, Python, Java, C#,* dan lainnya. Visual Studio Code termasuk *software* yang ringan dengan performa cepat dan dapat berjalan di berbagai platform. Untuk para

"Rancang Bangun Sistem", Alycia Angel Natalie, Universitas Multimedia Nusantara

pengembang, Visual Studio Code juga menawarkan fitur bawaan seperti integrasi *GitHub* dan *IntelliSense*, yang sangat membantu dalam pemrograman dengan bahasa seperti *HTML, CSS, JavaScript, dan TypeScript* [27].

2.4.2 *XAMPP*

XAMPP adalah perangkat lunak *open-source* dengan akronim dari "*Cross-Platform, Apache, MariaDB, PHP, dan Perl*." Dikembangkan oleh *Apache Friends*, XAMPP dapat diunduh gratis dan berjalan di *Windows, Linux, Solaris, dan Mac*. Perangkat lunak ini menyertakan *PHP, Apache, MySQL, dan PHPMyAdmin*, memungkinkan instalasi mudah dengan semua komponen yang diperlukan pada *hard disk* [28].

2.4.3 *Figma*

Figma merupakan aplikasi dalam mendesain dan *prototyping* dengan basis *cloud* untuk proyek digital. Dirancang untuk mendukung kolaborasi dan kerja tim, *Figma* memungkinkan pengguna bekerja bersama secara efektif dan fleksibel dari berbagai lokasi. Aplikasi ini umumnya digunakan oleh desainer antarmuka pengguna (UI) atau pengalaman pengguna (UX) untuk menciptakan tampilan antarmuka *website* atau aplikasi seluler.