

BAB II

LANDASAN TEORI

2.1 Komponen Sistem Informasi

Sistem informasi bukan sekadar kumpulan perangkat lunak yang berdiri sendiri, melainkan sebuah ekosistem digital terintegrasi yang terdiri dari berbagai elemen yang saling mendukung untuk memproses data mentah menjadi informasi bernilai guna dalam pengambilan keputusan. Agar sistem informasi aplikasi kasir pada Toko Reihan dapat beroperasi secara optimal, handal, dan berkelanjutan, rancangan sistem ini bertumpu pada integrasi enam komponen utama. Komponen-komponen tersebut meliputi perangkat keras (*hardware*), perangkat lunak (*software*), basis data (*database*), sumber daya manusia (*brainware*), jaringan (*network*), serta prosedur operasional yang baku. Melalui keselarasan elemen-elemen ini, sistem dapat meminimalkan risiko kesalahan manusia, mempercepat proses pelayanan di meja kasir, serta menyajikan laporan keuangan toko secara akurat.

Komponen *hardware* yang digunakan mencakup satu unit komputer kasir (spesifikasi minimal prosesor Intel Core i3, RAM 4 GB, Harddisk 500 GB) yang dilengkapi dengan *barcode scanner* untuk pencarian barang cepat dan *printer thermal* untuk mencetak struk belanja. Perangkat keras tersebut digerakkan oleh perangkat lunak (*software*) berbasis *web* yang dibangun menggunakan bahasa pemrograman PHP (*server-side scripting*) serta kombinasi HTML, CSS, dan JavaScript pada sisi klien, dengan memanfaatkan paket server lokal XAMPP (Apache *web server*) di lingkungan Windows 10. Seluruh data operasional

kemudian dikelola secara terstruktur di dalam komponen *database* MySQL bernama *db_penjualan* yang mengintegrasikan tabel barang, kategori, pengguna, member, dan transaksi, serta dilengkapi mekanisme *trigger* otomatis untuk pembaruan stok barang secara langsung (*real-time*) setiap kali transaksi selesai diproses.

Keberhasilan implementasi teknologi ini selanjutnya ditentukan oleh sinergi antara sumber daya manusia (*brainware*), jaringan (*network*), dan prosedur (*procedures*). Komponen *brainware* dalam sistem ini dibagi ke dalam tiga hak akses (*privilege*) yang ketat, yaitu Admin yang mengelola seluruh master data dan konfigurasi, Kasir yang khusus mengoperasikan halaman transaksi satu layar untuk melayani pembeli, serta *Owner* (pemilik toko) yang menerima dokumen laporan berkala. Demi efisiensi sumber daya pada Toko Reihan, komponen *network* dikonfigurasi pada jaringan lokal (*localhost*) di mana arsitektur server tertanam langsung di dalam komputer kasir, namun tetap fleksibel (*scalable*) jika ingin dipindahkan ke *cloud server* di masa mendatang. Terakhir, seluruh aktivitas tersebut diatur oleh komponen prosedur berupa standar operasional (SOP) tertulis, mulai dari alur autentikasi halaman *login*, tata cara input belanjaan pada antarmuka satu layar, hingga prosedur validasi data dan penyerahan laporan harian fisik kepada pemilik toko.

2.1.1 Pengertian Sistem

Sistem adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasanya terbagi dalam sub sistem yang lebih kecil yang

mendukung sistem yang lebih besar. Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu (Gani et al., 2023).

Pengertian sistem merupakan sekelompok atau seperangkat unsur-unsur yang erat hubungannya antara unsur yang satu dengan yang lainnya dan berfungsi bersama-sama untuk mencapai suatu tujuan tertentu. Sistem memiliki ciri atau karakteristik antara lain komponen sistem (*component*), lingkungan luar sistem (*environment system*), batasan sistem (*boundry system*), penghubung sistem (*interface system*), masukan sistem (*input system*), pengolahan sistem (*processing system*), keluaran sistem (*output system*) dan sasaran sistem (*goalsystem*).

2.1.2 Informasi

Informasi adalah data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, melakukan keputusan berdasarkan informasi tersebut dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali.

Informasi merupakan hasil dari sistem informasi manajemen. Informasi berasal dari kumpulan data-data mentah, baik itu berasal dari dalam perusahaan dan juga dari luar perusahaan. Data ini kemudian diolah, diproses oleh sistem sehingga menghasilkan informasi yang berguna. Informasi yang dihasilkan harus informasi yang berkualitas, bisa digunakan dan punya manfaat bagi penggunanya. Informasi dapat diibaratkan sebagai darah yang mengalir di dalam tubuh manusia, seperti halnya informasi di dalam sebuah perusahaan yang sangat penting untuk

mendukung kelangsungan perkembangannya, sehingga terdapat alasan bahwa informasi sangat dibutuhkan bagi perusahaan.

2.1.3 Pengertian Sistem Informasi

Sistem informasi merupakan sebuah sistem dalam suatu organisasi yang menyatukan kebutuhan pengolahan transaksi harian untuk mendukung fungsi operasional organisasi yang berhubungan dengan manajemen dengan kegiatan strategis dari suatu organisasi, agar bisa menyediakan informasi yang diperlukan oleh pihak tertentu untuk pengambilan keputusan. Sistem Informasi berfungsi sebagai fondasi utama bagi setiap organisasi *modern*. Dengan memanfaatkan teknologi informasi, organisasi dapat mengelola data secara lebih efisien dan efektif, serta meraih keunggulan kompetitif.

Sistem informasi dapat terdiri dari kombinasi beberapa elemen teknologi berbasis komputer yang saling berkolaborasi berdasarkan prosedur kerja yang telah ditentukan, di mana memproses dan mengolah data menjadi sebuah bentuk informasi yang digunakan untuk mendukung keputusan. Sistem informasi merupakan suatu kerangka kerja yang mengatur sumber daya (manusia, komputer) untuk mengubah masukan (*input*) menjadi keluaran (*output*) dalam bentuk informasi guna mencapai tujuan. Sistem informasi terdiri dari komponen-komponen yang dikenal sebagai blok bangunan atau *building block*, manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai sebuah tujuan.

2.1.4 Pengertian Komponen Sistem Informasi

Komponen yaitu bagian-bagian yang membentuk suatu sistem atau produk. Komponen-komponen tersebut bisa berupa perangkat keras atau sarana intraksi yang bekerja sama akan membentuk sistem yang utuh (Effendi et al., 2023). Komponen sistem informasi yaitu berupa sistem digital yang terdiri dari perangkat keras dan perangkat lunak, antamuka untuk entri data dan akses informasi, peran orang-orang yang terlibat, dan proses yang mengatur operasinya secara jelas (Soliha & Kusumaningtyas, 2024). Komponen-komponen dalam sistem informasi mencakup :

1. Perangkat keras (*Hardware*) merupakan perangkat fisik yang digunakan untuk memasukkan, memproses, dan mengeluarkan data. Contohnya adalah komputer, *printer*, dan *scanner*.
2. Perangkat lunak (*Software*) merujuk pada program yang berjalan di dalam perangkat keras yang memungkinkan sistem ini berjalan dan digunakan oleh manusia. Contohnya adalah sistem operasi (OS), *software* aplikasi, dan sistem manajemen basis data.
3. Jaringan telekomunikasi merupakan infrastruktur sistem penghubung yang digunakan untuk mengkoneksikan sumber atau resources yang digunakan dan diakses secara bersama-sama serta diakses oleh beberapa pengguna.
4. Basis data merupakan kumpulan data yang terorganisir yang dapat diakses, dikelola, dan diperbarui.

5. Sumber daya manusia manusia atau orang turut menjadi komponen dalam sistem informasi sebagai pengguna yang memasukkan data, menjalankan program, dan menginterpretasikan *output*.
6. Prosedur sekumpulan aturan yang dipakai untuk menjalankan sebuah proses data dan juga untuk menghasilkan output sesuai dengan yang dikehendaki.



Gambar 2. 1 Komponen sistem informasi

Sumber : (Soliha & Kusumaningtyas, 2024)

2.2 Perancangan Sistem Informasi Berbasis *Web*

Perancangan sistem informasi berbasis *web* untuk Toko Reihan bertujuan untuk menggantikan sistem manual yang selama ini digunakan dalam pengelolaan data penjualan dan pembelian. Sistem yang dirancang diharapkan dapat mengatasi masalah yang sering terjadi pada sistem manual, seperti ketidakteraturan dalam pencatatan stok, kesalahan dalam perhitungan transaksi, dan keterlambatan dalam pembuatan laporan. Dengan sistem berbasis *web*, data dapat diakses secara *real-*

time, mempercepat pencarian informasi, serta menghasilkan laporan yang lebih akurat dan tepat waktu.

2.2.1 Pengertian Perancangan

Perancangan merupakan tahapan bagaimana mencapai sasaran yang ditetapkan melibatkan pembentukan perangkat lunak dan komponen perangkat keras sistem, yang dimana setelah pemasangan sistem akan memenuhi spesifikasi yang dibuat pada tahap akhir.

Perancangan adalah sekumpulan aktivitas yang menggambarkan secara rinci bagaimana sistem akan berjalan. Hal itu bertujuan untuk menghasilkan produk perangkat lunak yang sesuai dengan kebutuhan pengguna. Dalam tahap perancangan, tim kerja harus merancang dalam berbagai kertas kerja mengenai spesifikasi yang dimaksud sesuai kebutuhan pengguna akhir (*end user*) melalui alat perancangan yang terstandarisasi.

2.3 Metode Yang Digunakan

Untuk menyelesaikan sebuah *project* perancangan *website* diperlukan metode yang tepat untuk melakukannya, metode yang digunakan adalah *Waterfall*.

2.3.1 Metode *Waterfall*

Metode *Waterfall* merupakan salah satu model pengembangan aplikasi yang termasuk dalam kategori siklus hidup klasik, yang menekankan pada langkah-langkah yang dilakukan secara berurutan dan terstruktur. Konsep dasarnya mirip dengan air terjun, di mana setiap tahapan proses dilakukan secara berturut-turut dari atas ke bawah, dengan langkah-langkah yang jelas dan terdefinisi dengan baik.

Metode air terjun biasanya terdiri dari lima tahapan utama. Dengan adanya tahapan-tahapan yang terstruktur ini, metode *Waterfall* membantu dalam menjaga keteraturan dan kejelasan dalam proses pengembangan aplikasi. Berikut adalah tahapan utama dalam model *Waterfall*.



Gambar 2. 2 Metode Perancangan Waterfall

Sumber : (Putri & Taufik, 2024)

1. Analisis Kebutuhan

Sebelum mengembangkan perangkat lunak, pengembang perlu memahami kebutuhan pengguna melalui diskusi, observasi, survei, atau wawancara. Data yang terkumpul kemudian dianalisis untuk memastikan spesifikasi kebutuhan terdokumentasi dengan baik sebelum pengembangan dimulai. Langkah ini membantu memastikan bahwa solusi yang dibangun benar-benar sesuai dengan masalah dan tujuan pengguna.

2. Desain Sistem dan Perangkat Lunak

Setelah spesifikasi kebutuhan teridentifikasi, langkah selanjutnya adalah menerjemahkannya ke dalam desain sistem yang konkret. Desain ini juga

membantu pengembang menyiapkan infrastruktur perangkat keras yang mendukung arsitektur perangkat lunak.

3. Implementasi dan Pengujian Unit

Pada tahap implementasi, pengembang memprogram perangkat lunak dalam modul-modul kecil yang diuji dan diverifikasi fungsionalnya. Pengujian ini memastikan setiap modul memenuhi standar dan spesifikasi yang telah ditetapkan. Setelah itu, modul-modul tersebut akan digabungkan untuk membentuk sistem yang utuh.

4. Integrasi dan Pengujian Sistem

Setelah semua modul atau unit telah selesai dikembangkan dan diuji secara individual, langkah selanjutnya adalah mengintegrasikan mereka ke dalam sistem secara keseluruhan. Proses integrasi ini kemudian diikuti dengan pengujian menyeluruh terhadap seluruh sistem untuk mengidentifikasi potensi kesalahan atau kegagalan sistem yang mungkin terjadi.

5. Operasi dan Pemeliharaan

Pada tahap terakhir metode *Waterfall*, perangkat lunak dioperasikan dan dipelihara oleh pengguna. Pemeliharaan ini memungkinkan perbaikan kesalahan, peningkatan, dan penyesuaian sistem sesuai kebutuhan nyata. Dokumentasi teknis dan pengguna diperbarui untuk memastikan keberlanjutan pengelolaan sistem.

2.4 Alat Bantu Perancangan

Untuk merancang sebuah *website*, diperlukan alat bantu perancangan, dan kali ini saya menggunakan *Unified Modeling Language (UML)*

2.4.1 Unified Modeling Language (UML)

Unified Modelling Language (UML) merupakan sebuah bahasa yang divisualisasikan dalam bentuk gambar atau grafik yang berfungsi untuk memberikan gambaran dan spesifikasi dalam pembangunan dan dokumentasi dari sebuah pengembangan sistem berorientasi objek.

Unified Modeling Language (UML) merupakan suatu standar bahasa yang dipakai untuk melakukan pemodelan, dokumentasi, dan perancangan sistem perangkat lunak yang mengadopsi pendekatan berorientasi objek. *UML* menawarkan serangkaian simbol grafis dan pedoman untuk membuat diagram yang menjelaskan beragam aspek sistem. *UML* berguna dalam upaya memahami, merencanakan, dan mengomunikasikan desain perangkat lunak. Di kalangan profesional, *UML* telah menjadi alat yang sangat berguna dalam industri perangkat lunak dan digunakan secara meluas untuk merancang, mengomunikasikan, dan mengelola sistem yang berbasis pada pendekatan berorientasi objek.

2.4.2 Use Case Diagram

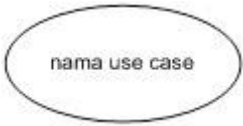
Use case atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan

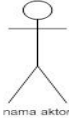
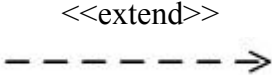
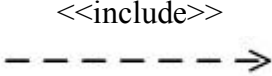
sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

Use case diagram merupakan *tools* untuk mengembangkan sistem dengan menjelaskan hubungan antara aktor yang terlibat dalam sistem. Diagram ini biasanya digunakan untuk mengkomunikasikan fungsi *high-level* dari sistem dan ruang lingkup sistem yang menggambarkan beberapa ekspektasi (harapan) dari sistem oleh praktisi diluar sistem. Diagram ini berguna untuk orang diluar sistem yang biasanya digunakan untuk menentukan perilaku sistem dari sudut pandang pengguna.

Sebuah *use case* menjelaskan sebuah fungsi yang disediakan oleh sistem yang menghasilkan *output* yang terlihat untuk seorang aktor. Sedangkan seorang aktor menjelaskan setiap entitas yang berinteraksi dengan sistem. Identifikasi aktor dan *use case* menghasilkan definisi batas sistem (*system boundary*), yaitu membedakan tugas/kegiatan yang diselesaikan oleh sistem dan tugas/kegiatan yang diselesaikan oleh lingkungannya. Aktor berada diluar *system boundary*, sedangkan *use case* berada di dalam *system boundary*. Berikut adalah simbol-simbol yang ada pada diagram *use case*.

Tabel 2. 1 Simbol Use Case Diagram

Simbol	Deskripsi
<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>

Aktor/ <i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri
Asosiasi/ <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi/ <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu
Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya
Menggunakan/ <i>include</i> 	<i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan

Sumber : (Syahrizal, 2025)

2.4.3 Activity Diagram

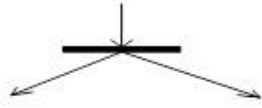
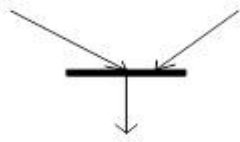
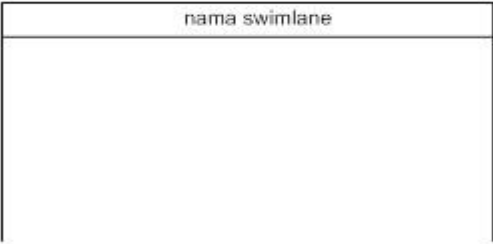
Diagram aktivitas, atau yang dikenal sebagai *activity diagram*, memvisualisasikan alur kerja atau aktivitas dari suatu sistem, proses bisnis, atau opsi pada perangkat lunak. Yang penting untuk diingat adalah bahwa diagram ini merepresentasikan aktivitas yang dilakukan oleh sistem, bukan oleh aktor; jadi, semua tindakan diambil oleh sistem. *Activity diagram* sering digunakan untuk menjelaskan hal-hal berikut.

1. Rancangan proses bisnis di mana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Pengelompokan tampilan dari sistem di mana setiap aktivitas dianggap memiliki rancangan antar muka tampilan.
3. Rancangan pengujian di mana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya (Amanda & Nopriani, 2024).

Berikutnya, terdapat simbol-simbol yang digunakan dalam *activity diagram* untuk merepresentasikan proses tersebut.

Tabel 2. 2 Simbol Activity Diagram

Simbol	Deskripsi
<i>Initial State</i> 	Status awal aktivitas sistem
<i>Activity</i> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
<i>Control Flow</i> 	Urutan perpindahan suatu aktivitas
<i>Decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
<i>Final State</i> 	Status akhir yang dilakukan sistem

<i>Fork</i> 	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel
<i>Join</i> 	Digunakan untuk menunjukkan kegiatan yang digabungkan
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : (Amanda & Nopriani, 2024)

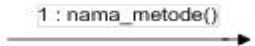
2.4.4 *Sequence Diagram*

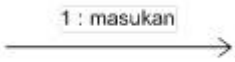
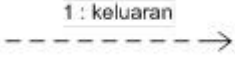
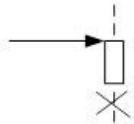
Untuk menggambar *Sequence Diagram*, perancang perlu memahami objek dan metode yang terlibat dalam sebuah *Use Case*. *Sequence Diagram* menggambarkan interaksi antar objek dalam suatu urutan waktu, yang menunjukkan bagaimana objek saling berkomunikasi untuk menyelesaikan suatu proses atau fungsi tertentu. Diagram ini membantu dalam memvisualisasikan alur eksekusi sistem secara rinci.

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *Sequence Diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Membuat *Sequence Diagram* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya *Sequence Diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan inter-aksi jalannya dalam *Sequence Diagram* sehingga semakin banyak *use case* yang didefinisikan maka *Sequence Diagram* yang harus dibuat juga semakin banyak.

Tabel 2. 3 Simbol Sequence Diagram

Simbol	Deskripsi
Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
Garis Hidup/<i>Lifeline</i> 	Menyatakan kehidupan suatu objek
Objek 	Menyatakan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya
Pesan Tipe <i>Create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan Tipe <i>Call</i> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri

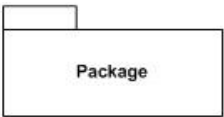
Pesan Tipe <i>Send</i> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
Pesan Tipe <i>Return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah megarah pada objek yang menerima
Pesan Tipe <i>Destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destory</i>

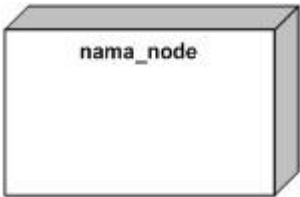
Sumber : (Khasani & Subrata, 2025)

2.4.5 Deployment Diagram

Secara umum, jika kita menafsirkan *deployment diagram*, diagram ini merupakan salah satu bentuk dari *Unified Modeling Language (UML)*. Tujuannya adalah untuk menggambarkan, memvisualisasikan, merinci, serta mendokumentasikan suatu proses yang berlangsung dalam sebuah sistem yang berbasis pada *Object Oriented Programming (OOP)* yang akan dikembangkan. Pendapat lainnya menyatakan bahwa diagram *deployment* adalah representasi dari proses-proses dalam sebuah sistem yang beroperasi, serta menjelaskan bagaimana hubungan di dalam sistem tersebut.

Tabel 2. 4 Simbol Deployment Diagram

Simbol	Deskripsi
<i>Package</i> 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i>

<i>Node</i> 	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak (<i>software</i>) yang tidak dibuat sendiri, jika didalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
Kebergantungan/ <i>dependency</i> 	Kebergantungan antara <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
<i>Link</i> 	Relasi antar <i>node</i>

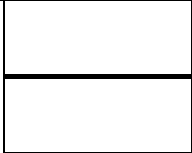
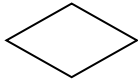
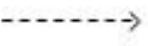
Sumber : (Gani & Awaludin, 2024)

2.4.6 Class Diagram

Diagram kelas menggambarkan interaksi antara kelas-kelas serta memberikan penjelasan mendalam mengenai setiap kelas dalam rancang bangun sebuah sistem. Ini juga menunjukkan aturan dan tanggung jawab dari entitas yang membentuk perilaku sistem. Dengan demikian, dapat disimpulkan bahwa *Diagram* kelas adalah representasi visual dari struktur sistem perangkat lunak yang dihasilkan. Diagram ini menampilkan kelas, atribut, metode, serta hubungan antar kelas dalam sistem. Oleh karena itu, diagram kelas menjadi alat bantu bagi pengembang untuk memahami dan merencanakan arsitektur perangkat lunak dengan lebih sistematis. *Diagram* kelas berfungsi sebagai alur *database* yang akan dibangun. *Diagram* kelas juga dapat disebut sebagai kumpulan beberapa kelas beserta hubungannya. Kelas itu sendiri identik dengan entitas yang direpresentasikan dalam bentuk persegi, di mana nama kelas dituliskan di bagian

atas, diikuti dengan atribut kelas dibawahnya, dan dibawahnya lagi terdapat metode yang ada dalam kelas.

Tabel 2. 5 Simbol Class Diagram

Simbol	Keterangan
 <i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
 <i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
 <i>Assosiasi</i>	Hubungan statis antar <i>class</i> yang menggambarkan <i>class</i> yang memiliki atribut berupa <i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain
 <i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
 <i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independet</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak

	mandiri (<i>independent</i>).
--	---------------------------------

Sumber : (Ramdany et al., n.d.)

2.5 Sekilas Tentang *Web*

Web menyediakan informasi dan layanan yang dapat diakses pengguna lewat internet, di mana *browser* digunakan sebagai sarana untuk membuka serta menampilkan halaman-halaman tersebut.

2.5.1 *Website*

Website atau *web* adalah ruang informasi di internet yang memungkinkan pengguna komputer yang terhubung ke internet untuk mengakses berbagai layanan, termasuk fasilitas *hypertext* yang memungkinkan penampilan data seperti teks, gambar, suara, animasi, dan jenis data multimedia lainnya. Pemakai dapat menemukan informasi dengan mengikuti tautan yang ditemukan dalam dokumen yang ditampilkan di internet. Web ini menyediakan informasi bagi pemakai komputer yang terhubung ke ininternet. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman.

Website adalah kumpulan dari halaman *web* yang saling terhubung dalam sebuah domain atau subdomain yang berada didalam *Wordwide web* (*WWW*) diinternet. Aplikasi berbasis web pada umumnya dibangun dengan bantuan dari struktur *HTML* (*Hypertext Markup Language*), serta dengan

kombinasi dari beberapa bahasa pemrograman lain, seperti *PHP* ataupun *javascript*. *Website* juga dapat dipercantik tampilanya dengan bantuan *CSS* (*Cascading style Sheets*). Mengenai *database* atau media penyimpanan, cukup banyak yang dapat digunakan, salah satunya adalah *MySQL*.

2.5.2 Web Browser

Browser merupakan perangkat lunak untuk menjalankan program atau *scriptweb*. Untuk mengakses sebuah halaman *web* maka kita perlu menggunakan sebuah *web browser*. Contoh *browser* adalah *internet explorer*, *Opera*, *Mozilla FireFox* dan lain-lainnya.

Web browser adalah alat yang digunakan untuk melihat halaman *web*, diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video dan gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman.

Google Chrome merupakan salah satu jenis aplikasi *web browser*. Aplikasi ini berfungsi sebagai media dalam melakukan *browsing*, atau berselancar di dunia maya. Aplikasi ini merupakan aplikasi yang dibutuhkan untuk menampilkan struktur dari halaman *website*. Selain halaman *web*, *Google Chrome* juga mampu menampilkan beberapa jenis *file* seperti *PDF*, ataupun *file* berbentuk gambar. *Google Chrome* cukup banyak digunakan, karena kemudahan dalam menggunakannya serta aplikasi ini dapat berjalan pada berbagai macam *platform*.

2.5.3 *World Wide Web (WWW)*

Ketika memasuki peralihan dari abad ke-20 ke abad ke-21, perkembangan media komunikasi mengalami transformasi teknologi yang membuatnya lebih interaktif (dua arah) dan melahirkan inovasi dalam cara berekspresi. Teknologi informasi dan komunikasi (TIK) memungkinkan media tidak hanya digunakan secara sepihak tetapi juga mendukung komunikasi interaktif dua arah. Media TIK juga memberi peluang untuk berkomunikasi dalam berbagai representasi seperti gambar, suara, teks, dan format lainnya melalui layanan internet World Wide Web (WWW). Layanan WWW dirancang untuk menyebarkan informasi, yang pada awalnya digunakan sebagai alat untuk mendistribusikan hasil riset di CERN (lembaga penelitian fisika di Eropa). Teknologi ini diperkenalkan oleh Tim Berners-Lee, seorang peneliti di CERN. World Wide Web (WWW) adalah sebuah sistem informasi yang terdiri dari beragam dokumen dan sumber daya yang terhubung, yang dapat diakses melalui internet. Ini merupakan salah satu layanan yang tersedia di dunia maya. Web memfasilitasi pengguna untuk menjelajahi berbagai halaman web yang berisi teks, gambar, audio, video, dan sejumlah jenis konten lainnya. Sistem ini memungkinkan akses dan penyebaran informasi di internet melalui halaman-halaman yang berhubungan satu sama lain (Kurniasari et al., 2025). Web telah diintegrasikan oleh perusahaan sebagai bagian dari strategi teknologi informasi mereka, karena berbagai alasan. Akses informasi mudah.

1. Akses informasi mudah.
2. Setup server lebih mudah.
3. Informasi mudah distribusikan.

4. Bebas *platform*.

Informasi dapat disajikan oleh *browser web* pada sistem operasi mana saja karena adanya standar dokumen berbagai tipe data dapat disajikan (Widiyanto, 2022).

2.6 Alat Bantu Pemrograman

Dalam proses pengerjaan, saya memanfaatkan berbagai *software* dan bahasa pemrograman untuk membantu merancang program.

2.6.1 *HyperText Markup Language (HTML)*

Hypertext Markup Language (HTML) merupakan bahasa pemrograman yang terstruktur digunakan untuk pembuatan laman *website* dapat diproses menggunakan *web browser* (perambanan *web*). *HTML* diciptakan oleh Tim *Berners Lee* pada tahun 1989 dan dikembangkan *Worlds Wide Web Consortium (W3C)* dan terbentuk *Web Hypertext Application Technology Working Group (WHATG)* pada tahun 2004, hingga kini *WHATG* mengembangkan *HTML* sampai terbentuk *HTML 5* yang dapat mendukung gambar, teks, video, audio, menu interaktif dan lain-lain (Rahayu et al., 2023).

HTML adalah singkatan dari *Hypertext Markup Language*, bahasa *web* standar yang dikelola oleh *W3C (World Wide Web Consortium)* dalam bentuk tag yang menyusun setiap elemen situs *web*. *HTML* bertindak sebagai pembangun struktur halaman situs web Anda dan menempatkan setiap elemen tata letak situs *web* yang Anda inginkan. Pengembangan *HTML* dimulai dengan *HTML v1.0*. *HTML v1.0* menandai lahirnya pengembangan *HTML*. *HTML v2.0* menyediakan cara untuk menghubungkan dokumen dan menyajikan

informasi dalam berbagai format dan gaya. Anda memiliki opsi untuk menampilkan data dan formulir dalam dokumen, memasukkan alamat, nama, saran dan kritik. *HTML* v3.0 menambahkan fitur baru seperti angka. Diagram merupakan evolusi gambar yang digunakan untuk menyusun foto dan tabel. *HTML* v4.0 menyertakan banyak perubahan dari versi sebelumnya. Perubahan ini memengaruhi perintah *HTML* seperti gambar, tabel, teks, tautan, dan formulir. *HTML* v5.0 adalah sintaksis yang disederhanakan yang meminimalkan dampak kesalahan (Hidayat et al., 2025).

2.6.2 *Cascading Style Sheets (CSS)*

Cascading Style Sheets (CSS) adalah bahasa pemrograman yang digunakan untuk mengendalikan tampilan *web* secara terpisah dari kontennya. *CSS* memberikan *fleksibilitas* dalam pengaturan desain dan tata letak, yang memudahkan pengembangan dan pemeliharaan situs *web* (Masikhoh et al., 2025).

Cascading Style Sheet (CSS) merupakan aturan untuk mengatur beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam. *CSS* bukan merupakan bahasa pemrograman. Sama halnya *styles* dalam aplikasi pengolahan kata seperti *Microsoft Word* yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas (*file*). Pada umumnya *CSS* dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa *HTML* dan *XHTML*. *CSS* dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran border, warna border, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas,

bawah, dan parameter lainnya. *CSS* adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya *CSS* memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda (Kumalasari et al., 2023).

2.6.3 Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) merupakan skrip yang bersifat *server site* dimana proses pengerjaan skripnya berlangsung di *server*. Dengan menggunakan *PHP* maka perawatan suatu situs *web* akan menjadi lebih mudah. *PHP* pertama kali ditemukan oleh *Rasmus Lerdoff*. Penulisan skrip *PHP* tersebut dengan cara disisipkan pada *HTML*. *PHP* merupakan bahasa pemrograman yang digunakan untuk membangun aplikasi-aplikasi berbasis *web* khususnya aplikasi *web* yang bersifat dinamis .

Hypertext preprocessor atau yang biasanya kita kenal dengan sebutan *PHP* ialah satu diantara banyaknya jenis bahasa pemrograman yang didalamnya memiliki karakteristik *open source* dan biasanya dikhususkan dalam pengembangan dan pembuatan *web*, serta biasanya diletakkan dalam suatu deskripsi *HTML*. Dalam pengertian lain *PHP* adalah sebagai bahasa *script server-side* yang digunakan dalam pengembangan *web* yang disematkan dalam dokumen *HTML* dan merupakan perangkat lunak yang memiliki karakteristik *open source*. Penggambaran dari bahasa pemrograman *PHP* memiliki sifat yang hampir mirip dengan bahasa-bahasa pemrograman lain diantaranya, serta cukup mudah dalam memahaminya (Arribe & Ryandi, 2023).

2.6.4 *Bootstrap*

Bootstrap adalah *front-end framework* yang mengutamakan tampilan pada *mobile* atau *website*. *Bootstrap* diimplementasikan menggunakan *Less*, teknologi CSS yang sederhana untuk dijalankan. Dengan adanya *Less* maka pembuatan tampilan *design* bisa mudah untuk dirubah (Angelo & Ridho, 2022). *Bootstrap* diciptakan untuk mempermudah proses *design website* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Dengan adanya *bootstrap* tersebut tentu saja membuat halaman *website* bisa menyesuaikan dengan ukuran *monitor device*. Baik jika diakses lewat ponsel, tablet ataupun *desktop*. Awal mulanya, *bootstrap* sendiri bernama *Twitter Blueprint*. Sebagai pengguna anda hanya perlu memanggil setiap kelas yang digunakan, contohnya seperti navigasi, tabel, *grind*, tombol atau sebagainya. Banyak fungsi *bootstrap* yang bisa dipakai untuk sebuah *website* (Nurfaizal et al., 2025).

Bootstrap dikembangkan oleh Mark Otto dan Jacob Thornton dari *twitter*. *Framework* ini diluncurkan sebagai produk *open source* pada Agustus 2011 di *GitHub*. *Bootstrap* menampilkan komponen antarmuka yang baik seperti Tipografi, Formulir, Tombol, Tabel, Navigasi, *Dropdown*, Lansiran, Modal, Tab, Akordeon, Korsel, dan sebagainya. Dengan menggunakan *bootstrap*, dapat dengan mudah membuat tata letak situs yang responsif. Salah satu kekuatan *bootstrap* adalah berisi kumpulan alat gratis untuk membuat tata letak *web* yang fleksibel dan responsif (Saputra, 2023).

2.6.5 Javascript

JavaScript adalah bahasa pemrograman *web* yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti *Google Chrome*, *Mozilla Firefox*, *Opera Mini* dan sebagainya (Hardianto et al., 2023).

JavaScript adalah bahasa skrip utama untuk *browser web* dan digunakan dalam pengembangan aplikasi yang kompleks. *JavaScript* dapat ditulis langsung dalam tag `<script>` pada *file HTML* atau disimpan dalam *file* terpisah dengan format *.js* yang dihubungkan ke *HTML*. Untuk membangun *website* yang lebih kompleks dan efisien, penggunaan *framework* menjadi solusi yang mempermudah pengembangan dan pengelolaan kode. *JavaScript* pertama kali diperkenalkan oleh *Netscape* pada tahun 1995 sebagai bahasa *scripting* untuk meningkatkan interaktivitas halaman web, *JavaScript* telah berevolusi menjadi bahasa utama dalam pengembangan aplikasi *web* dinamis dan kompleks (Fatikasari & Fargazzi, 2025).

2.7 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, dan merupakan kompilasi dari beberapa program (Kurniasari et al., 2025). *XAMPP* sendiri merupakan singkatan dari *Apache*, *MySQL*, *PHP* dan juga *Perl*. Secara luas defenisi *XAMPP* merupakan web lengkap untuk melakukan pemrograman *web*, *XAMPP* memudahkan *web developer* untuk mengembangkan *website* di *local computer*, sehingga pembuatan *website* lebih aman dan cepat.

XAMPP sering digunakan untuk menjalankan sebuah *website* dengan basis *PHP* dengan menggunakan *MySQL* sebagai pengolah data utama di *local computer* (Lim & Silalahi, 2023). *XAMPP* merupakan perangkat lunak server yang digunakan untuk menampilkan perintah *PHP* yang telah dirancang pemrogram dan data yang telah disimpan dalam *MySQL* di penyimpanan lokal (Lim & Silalahi, 2023). Dalam *XAMPP* berisi berbagai perangkat lunak seperti, *PHP*, *MySQL*, *Apache*.



Gambar 2. 3 Logo XAMPP

Sumber : *Wikipedia.org*

Melalui *XAMPP* ini aplikasi *website* atau *CMS* Bisa di jalankan, termasuk *Joomla*, *Drupal*, *Wordpress*, dan lainnya. *XAMPP* adalah *software web server apache* yang di dalamnya tertanam server *MySQL* yang didukung dengan bahasa pemrograman *PHP* untuk membuat *website* yang dinamis. *XAMPP* sendiri mendukung dua sistem operasi yaitu *windows* dan *linux*. Untuk *linux* dalam proses penginstalannya menggunakan *command line* sedangkan untuk *windows* dalam proses penginstalannya menggunakan *interface grafis* sehingga lebih mudah. Didalam *XAMPP* ada 3 komponen utama yang di tanam di dalamnya yaitu *web server Apache*, *PHP*, dan *MySQL* (Suhendri, 2025).

2.7.1 *MySQL*

MySQL adalah suatu perangkat lunak database relasi atau *Relational Database management system (RDBMS)* yang didistribusikan gratis di bawah *lisensi GPL (General Public License)*. Dimana setiap orang bebas menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang dijadikan *closed source* atau komersial (Ningsih et al., 2022).

Perangkat lunak *MySQL* terdiri dari server *MySQL*, beberapa program utilitas yang membantu dalam administrasi *database MySQL*, dan beberapa perangkat lunak pendukung yang dibutuhkan server *MySQL* (tetapi kita tidak perlu mengetahuinya). Jantung dari sistem adalah server *MySQL*. Server *MySQL* adalah pengelola sistem *database*. Ini menangani semua instruksi *database*. Misalnya, jika kita ingin membuat *database* baru, kita mengirim pesan ke server *MySQL* yang mengatakan, misalnya, buat *database* baru dan beri nama data baru. Server *MySQL* kemudian membuat subdirektori datanya, menamai subdirektori baru dengan data baru, dan menempatkan *file* yang diperlukan dengan format yang diperlukan ke dalam subdirektori data baru (Aziz et al., 2024).

MySQL merupakan *database* yang menghubungkan *script PHP* dengan perintah *query* dan *escape character* yang sama dengan *PHP*. *MySQL* bersifat *open source* dimana anda tidak perlu membayar untuk menggunakannya pada berbagai *platform* kecuali jenis-jenis *enterprise* dikarenakan bersifat komersial. *MySQL* ini masuk kedalam jenis *Relation Database Management System (RDMS)* sehingga istilah seperti baris, kolom dan tabel digunakan didalam *MySQL* (Solihin et al., 2024).

2.7.2 *PhpMyAdmin*

PhpMyAdmin adalah aplikasi perangkat lunak bebas (*open source*) yang ditulis dalam bahasa pemrograman *PHP* dan digunakan untuk mengelola administrasi *database MySQL* melalui jaringan lokal maupun internet. *PhpMyAdmin* mendukung berbagai operasi pada *MySQL*, termasuk mengelola basis data, tabel, kolom (*fields*), relasi (*relations*), indeks, pengguna (*users*), perizinan (*permissions*), dan lainnya (Tandi et al., 2024).

PhpMyAdmin sendiri mulai dikembangkan sejak tahun 1998 oleh *Tobias Ratschiller* seorang konsultan *IT*. *Ratschiller* mengerjakan sebuah program bernama *MySQL-Webadmin* pada awalnya, yang merupakan produk dari *Petrus Kuppelwieser*, yang telah berhenti mengembangkannya pada saat itu. *Ratschiller* kemudian menuliskan kode baru untuk *PhpMyAdmin*, dan ditingkatkan dengan dasar konsep dari proyek *Kuppelwieser*. Kemudian *Ratschiller* meninggalkan proyek *PhpMyAdmin* pada tahun 2001. Sekarang, Sebuah tim dari pengembang yang dipimpin oleh *Oliver Muller* meneruskan pengembangan *PhpMyAdmin* di *SourceForge.net* (Hartati, 2022).

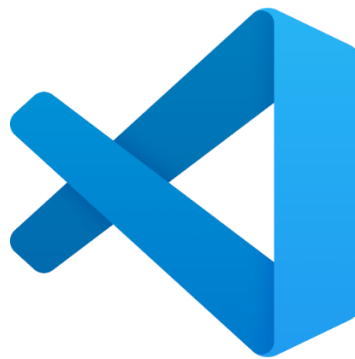
2.8 Tools Yang Digunakan

Berikut adalah beberapa tools yang saya gunakan untuk merancang aplikasi kasir ini yaitu.

2.8.1 *Visual Studi Code*

Visual Studio Code adalah kode editor sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux* dan *mac OS*. *Visual Code* memudahkan dalam penulisan *code* yang mendukung beberapa jenis bahasa pemrograman yang

digunakan dan memberi variasi warna sesuai dengan fungsi dalam rangkaian *code* tersebut. *Visual Studio Code* bersifat *open source*, yaitu aplikasi dengan *source code* yang dapat dilihat oleh siapapun untuk berkontribusi pada pengembangan aplikasi tersebut. *Code* juga dapat dilihat melalui link *github*, menjadikan aplikasi *Visual Studio Code* memiliki banyak penggemar dalam mengembangkan aplikasi kedepannya (Firnando et al., 2023).



Gambar 2. 4 Logo Visual Studio Code

Sumber : *code.visualstudio.com*

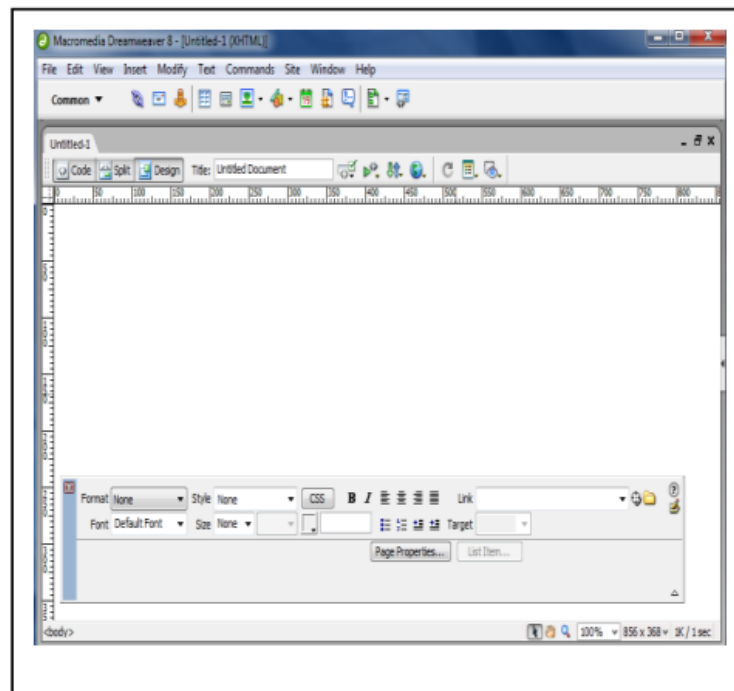
Teks editor *Visual Studio Code* dibuat oleh *Microsoft*. Teks editor ini menawarkan berbagai macam kemampuan, termasuk *code completion*, *syntax highlighting*, *debugging*, dan *git control*. teks editor ini kompatibel dengan semua sistem operasi, termasuk *Windows*, *Linux*, dan *Macintosh*. *Typescript*, *HTML*, *CSS*, *Node.js*, dan *JavaScript* semuanya didukung oleh *Visual Studio Code*. Salah satu *Integrated Development Environments (IDE)* yang paling disukai dan efisien bagi para pengembang adalah *Visual Studio Code*. IDE ini sangat kaya akan fitur dan memfasilitasi pengembang aplikasi *mobile* bagi para pengembang (Abidah et al., 2025).

2.8.2 *Macromedia Dreamweaver 8*

Adobe Dreamwaver adalah suatu produk *Web Developer* yang dikembangkan oleh *Adobe System Inc.* Sebelumnya produk *Dreamweaver* dikembangkan oleh *Macromedia Inc.* Yang kemudian sampai saat ini pengembangannya diteruskan oleh *Adobe System Inc.* Setelah diambil oleh *Adobe Systems Inc*, *dreamweaver* dikembangkan dan dirilis dengan kode nama *Creative Suit (CS)*. *Adobe dreamweaver* merupakan *web* editor yaitu program aplikasi yang berfungsi untuk mengetikkan perintah-perintah dokumen *web* baik *client side scripting* maupun *server side scripting*. Saat ini banyak tersedia *web* editor mulai dari yang paling sederhana hingga yang lebih *smart*. Mulai dari *web* editor yang berbayar hingga yang gratis (Nababan & Sinambela, 2025). Dalam aplikasi *Macromedia Dreamweaver 8* terdapat banyak *icon* yang bisa digunakan untuk membantu dalam proses . Berikut dapat dijelaskan satu persatu fungsinya yaitu:

- 1 *Document windows* adalah tempat atau daerah kerja pembuatan dan *editing* dokumen, dimana semua aktifitas dalam proses desain *website* dilakukan disini.
- 2 Objek Panel berisikan *icon-icon* yang dapat diklik untuk memasukkan objek yang dipilih kedalam dokumen.
- 3 *Launcher Bar* berisikan *icon* untuk membuka dan menutup *windows* lainnya, seperti *inspector* dan panel.
- 4 *Property Inspector* untuk menampilkan *property* objek dan teks dan untuk memodifikasi objek atau teks sesuai dengan kebutuhan.

- 5 *Toolbar* berisikan *icon* yang umumnya dipakai dalam pembuatan *website*.



Gambar 2. 5 Layar tampilan *Macromedia Dreamweaver 8*

Sumber : *Macromedia Dreamweaver 8*

2.8.3 *Wondershare EdrawMax*

Edraw Max sebuah perangkat lunak diagram berbasis *vector*, yang biasanya digunakan untuk mempermudah dalam membuat suatu rancangan. Salah satunya adalah perancangan sistem aplikasi dapat menggunakan aplikasi ini (Tundo & Saidah, 2023). *EdrawMax* cocok digunakan oleh berbagai kalangan, mulai dari pelajar hingga profesional di berbagai industri.

Kombinasi antarmuka yang ramah pengguna dan fitur canggih menjadikannya pilihan tepat untuk menciptakan visualisasi berkualitas tinggi. Tidak hanya itu, *EdrawMax* juga mendukung kolaborasi tim dan kompatibilitas lintas *platform*, sehingga mempermudah pengguna untuk berbagi hasil kerja

mereka. Dengan ribuan *template* dan ikon yang tersedia, *EdrawMax* memungkinkan siapa saja untuk memulai proyek visual dengan cepat tanpa perlu keahlian desain yang mendalam. Perangkat lunak ini dirancang untuk membantu pengguna menyampaikan ide dan informasi dengan cara yang lebih menarik dan efektif.



Gambar 2. 6 Logo Wondershare EdrawMax

Sumber : *webcatalog.io*

2.8.4 Microsoft Visio 2007

Microsoft Visio adalah *software* yang memudahkan untuk membuat diagram, *flowchart*, dan lainnya. Selain kemampuan menggambar diagram, juga dapat menggunakan *Microsoft Visio* untuk memvisualisasikan berita yang diperoleh dari gambar. *Microsoft Visio* terutama digunakan untuk membuat *flowchart*, jaringan komputer, jaringan komunikasi, grafik, dll, tetapi juga dapat menggunakan *Microsoft Visio* untuk memvisualisasikan data informasi, sehingga gambar yang dihasilkan bukan hanya gambar, dan dapat menjadi

laporan. *Microsoft Visio* juga sebagai perangkat lunak yang digunakan untuk mengekspresikan ide dan konsep dalam bentuk diagram alur, diagram, dan grafik lainnya. Aplikasi *Visio* memungkinkan untuk menggunakan fungsi dan template khusus sesuai kebutuhan, sehingga memudahkan untuk merancang berbagai jenis diagram (Supriadi & Haslinda, 2022).



Gambar 2. 7 Logo *Microsoft Visio* 2007

Sumber : *logos.fandom.com*

2.9 Penelitian Terdahulu

Penelitian terdahulu yang menjadi pembanding dalam perancangan sistem informasi kasir berbasis web ini antara lain sebagai berikut.

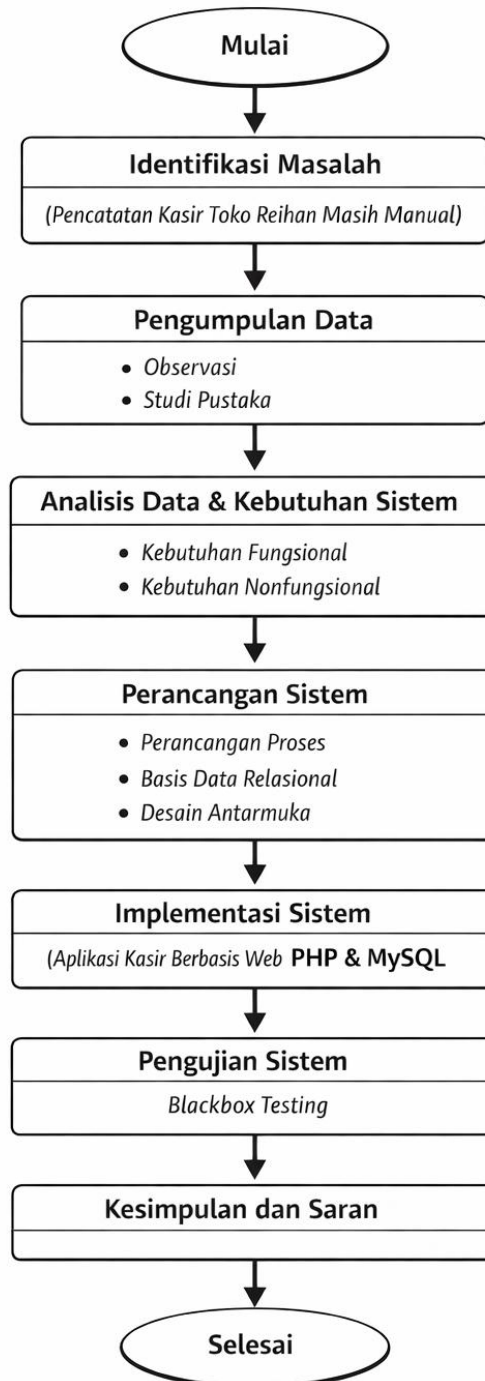
Tabel 2. 6 Penelitian Terdahulu

No	Penulis dan Tahun	Judul Penelitian	Hasil Penelitian
1	Sopia Dia Pangestu (2024)	Rancangan aplikasi kasir toko kelontong berbasis website menggunakan metode waterfall.	Aplikasi kasir berbasis web menggunakan PHP & MySQL melalui tahapan analisis, desain, implementasi, pengujian, evaluasi, hingga perbaikan untuk mempermudah transaksi dan manajemen stok.
2	Lolanda Hamim Annisa, Pangestu Salman Farizi, Ismail Hasan Mustofa, Kharisma Khoerunnisa (2023)	Perancangan dan Pembuatan Sistem Informasi Kasir Berbasis Web pada UMKM Toko Ritel.	Sistem kasir berbasis web yang mendukung pengelolaan transaksi, manajemen produk, serta pelaporan penjualan dengan teknologi PHP & MySQL.
3	Rahmasari (2022)	Perancangan Point Of Sales Pada Toko Bangunan Mulya	sistem POS web untuk mempercepat proses transaksi dan

		Ampuan Lumpo Berbasis Web Menggunakan PHP dan MySQL.	mempermudah pengelolaan data toko bangunan.
4	El Vionna Laellyn Nurul Fatic, Asni Tafrikhatin, Ariesta Tegar Juangera Reformadyananda (2023)	Perancangan Sistem Informasi Kasir Toko Retail Berbasis Web Menggunakan Metode POS (Point Of Sales).	Sistem POS berbasis web dengan fitur transaksi penjualan, manajemen stok, dan laporan transaksi untuk toko retail.
5	Ali Givahri Windyaning Ustyannie (2024)	Rancang bangun sistem point of sale (POS) berbasis website dengan metode waterfall pada Toko Nans Beauty Toili	Sistem POS berbasis website yang memperbaiki proses transaksi, pengelolaan inventori, dan efisiensi operasional di toko Nans Beauty Toili.
6	Leni Darsima Simatupang (2025)	Perancangan Sistem Informasi Aplikasi Kasir Menggunakan Bahasa Pemrograman PHP dan MySQL Pada Toko Reihan	

Penelitian sebelumnya menjadi dasar penting dalam pembuatan penelitian ini karena memberikan wawasan mengenai metode, pendekatan, dan hasil yang diperoleh oleh peneliti terdahulu. Berbagai kajian yang berhubungan dengan pengembangan sistem informasi kasir serta Point of Sales (POS) berbasis web menunjukkan bahwa penggunaan teknologi informasi dapat meningkatkan efisiensi, akurasi pencatatan transaksi, dan pengelolaan data penjualan di sektor ritel dan UMKM. Berbagai penulis sebelumnya menerapkan metode pengembangan sistem seperti model waterfall dan menggunakan teknologi PHP serta MySQL untuk pengolahan data. Temuan dari penelitian-penelitian tersebut menunjukkan bahwa sistem kasir yang berbasis web bisa membantu para pelaku usaha dalam mempercepat transaksi, mengurangi kesalahan pencatatan, dan menghasilkan laporan penjualan yang lebih tepat. Dengan demikian, penelitian ini disusun dengan mengacu pada penelitian terdahulu sebagai referensi dan pembandingan, serta untuk mengembangkan aplikasi sistem informasi kasir yang lebih sesuai dengan kebutuhan operasional Toko Reihan.

2.10 Flowchart Kerangka Penelitian



Gambar 2. 8 Flowchart Kerangka Penelitian

Kerangka penelitian dimulai dari identifikasi masalah operasional kasir di toko reihan yang masih menggunakan pencatatan manual. Selanjutnya dilakukan pengumpulan data melalui observasi dan studi pustaka untuk memahami kebutuhan sistem. Data yang diperoleh dianalisis untuk merumuskan kebutuhan fungsional dan nonfungsional sistem. Berdasarkan hasil analisis, dilakukan perancangan sistem meliputi perancangan proses, basis data relasional, dan antarmuka pengguna. Tahap berikutnya adalah implementasi aplikasi kasir berbasis web menggunakan PHP dan MySQL. Setelah sistem diimplementasikan, dilakukan pengujian fungsional menggunakan metode blackbox untuk memastikan sistem berjalan sesuai kebutuhan. Penelitian diakhiri dengan penarikan kesimpulan dan pemberian saran pengembangan.

\