

## BAB II

### LANDASAN TEORI

#### 2.1 *State Of The Art*

Dibawah merupakan beberapa penelitian yang relevan untuk dijadikan sebagai pedoman pada penelitian yang sedang dilakukan. Penelitian tersebut terdapat pada tabel berikut ini:

**Tabel 2. 1** *State Of The Art*

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
1	Alif Naufal, Chotijah Umi  Universitas Muhammadiyah Gresik, Indonesia  Jurnal Ilmiah Indonesia p- ISSN: 2541-0849 e-ISSN: 2548-1398 Vol. 9, No. 6 Juni 2022	Perancangan Sistem Informasi <i>E-Commerce</i> Untuk Usaha Mikro Kecil Dan Menengah (Umkm) Furnitur  <a href="https://jurnal.syntaxliterate.co.id/index.php/syntax-literate/article/download/15166/10140">https://jurnal.syntaxliterate.co.id/index.php/syntax-literate/article/download/15166/10140</a>	<i>Waterfall</i>	Aplikasi <i>e-commerce</i> berbasis <i>web</i> mempermudah UMKM furnitur mengakses pasar luas dan mengelola usaha lebih efisien menggunakan teknologi terkini.
2	Hersanto Fajri, Berlina Wulandari,	Aplikasi <i>E-Commerce</i> Pada Toko <i>Isc Exhaust</i> Berbasis <i>Android</i>	<i>Waterfall</i>	Aplikasi <i>Android e-commerce</i>

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Muhammad Daffan Al Farizi  Teknik Informatika, Fakultas Teknik dan Sains, Universitas Ibn Khaldun  ISSN : 2460-1861 (Print), 2615-4250 (Online) Vol (9) No (2) 2023, pp. 538-548	<a href="https://ejournal.unma.ac.id/index.php/infotech/article/download/6854/3835/31489">https://ejournal.unma.ac.id/index.php/infotech/article/download/6854/3835/31489</a>		mengintegrasikan <i>Midtrans</i> dan perhitungan ongkir otomatis. Transaksi jadi aman dan lebih terjangkau untuk konsumen otomotif.
3	Faizu Az-Zahra, Isnaeni Dhiyaa Fathiina	<i>Website E-Commerce</i> pada UMKM Omah Kopi Tarik Tuntang	<i>Waterfall</i>	<i>Website e-commerce</i> memperluas jangkauan

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Jurusan Teknik Elektro, Politeknik Negeri Semarang  JTET(Jurnal Teknik Elektro Terapan) Vol. 11 No. 2 Agustus 2022 hal 42–49 ISSN : 2503-2941 (online), 2252-4908 (print)	<a href="https://jurnal.polines.ac.id/index.php/jtet/article/download/5395/109102/118806">https://jurnal.polines.ac.id/index.php/jtet/article/download/5395/109102/118806</a>		UMKM secara online, dilengkapi ongkir otomatis (RajaOngkir) dan <i>payment gateway Midtrans.</i>
4	Ainun Mukhliza Mantu1, Emy Lenora Tatuhey, Rosiyati M.H. Thamrin  Teknik Informatika, Universitas	Rancang Bangun Platform E-commerce berbasis Website pada Media Cell  <a href="https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/download/2324/1221">https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/download/2324/1221</a>	<i>Waterfall</i>	Platform e-commerce berbasis web dilengkapi manajemen stok, pembayaran <i>Midtrans</i> , dan pelacakan

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Sepuluh Nopember Papua, Jayapura, Indonesia  Vol. 13, No. 3, Desember 2024: 1988-1997			pengiriman . Fitur berjalan sesuai spesifikasi dan meningkatkan efisiensi.
5	Fariz Reynaldo Pratama, Nurudin Santoso, Lutfi Fanani  Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  Jurnal Pengembangan Teknologi	Pengembangan Aplikasi <i>E-Commerce</i> Menggunakan <i>Payment Gateway Midtrans</i>  <a href="http://download.garuda.kemdikbud.go.id/article.php?article=1717502&amp;val=10384&amp;title=Pengembangan%20Aplikasi%20E-Commerce%20Menggunakan%20Payment%20Gateway%20Midtrans">http://download.garuda.kemdikbud.go.id/article.php?article=1717502&amp;val=10384&amp;title=Pengembangan%20Aplikasi%20E-Commerce%20Menggunakan%20Payment%20Gateway%20Midtrans</a>	<i>Waterfall</i>	Aplikasi <i>mobile e-commerce</i> untuk UMKM kerajinan memperudah pengelolaan transaksi, kategori produk, dan ekspedisi. Uji <i>usability</i> menunjukkan

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	<p>Informasi dan Ilmu Komputer e- ISSN: 2548-964X Vol. 4, No. 4, April 2020, hlm. 1133-1140</p>			<p>kepuasan pengguna 100%.</p>
6	<p>Muh Agus, Harlinda L, Muhammad Arfah Asis  Universitas Muslim Indonesia  Jurnal Literatur Informatika dan Komputer, LINIER vol 1, No 4, (2024), pp 365-378, e-ISSN: 3063-2218</p>	<p>Perancangan Sistem Informasi Penjualan Berbasis <i>E-Commerce</i> dengan <i>Metode Waterfall</i>  <a href="https://jurnal.fikom.uni.ac.id/index.php/LINIER/article/view/2536/693">https://jurnal.fikom.uni.ac.id/index.php/LINIER/article/view/2536/693</a></p>	<i>Waterfall</i>	<p>Sistem ini menyederhanakan pengelolaan informasi penjualan dan memperbaiki efisiensi dalam melayani pelanggan melalui pengujian <i>black box</i> yang memuaskan.</p>

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
7	Fachmi Maulana, Esron Rikardo Nainggolan, Mulia Rahmayu  Universitas Nusa Mandiri  Jurnal Rekayasa Perangkat Lunak Volume 5 No. 1 Mei 2024 ISSN 2747-1799	Pemanfaatan <i>API Midtrans</i> Dan RajaOngkir Dalam Sistem Penjualan <i>Online</i>  <a href="https://jurnal.bsi.ac.id/index.php/reputasi/article/download/3456/1641">https://jurnal.bsi.ac.id/index.php/reputasi/article/download/3456/1641</a>	<i>Waterfall</i>	Sistem informasi <i>e-commerce</i> mempermudah penjualan alat musik secara <i>online</i> dengan integrasi <i>Midtrans</i> dan RajaOngkir. Transaksi lebih efisien, stok dan ongkir lebih terkontrol
8	Desi Ramayanti, Yuwan Jumaryadi, Aris Sunandar, Kolidi	Implementasi Metode <i>Waterfall</i> dalam Pengembangan Sistem <i>Point of Sales</i> dan <i>e-Commerce</i>	<i>Waterfall</i>	Aplikasi <i>e-commerce</i> dan <i>POS</i> berbasis web untuk UMKM

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	<p>Fakultas Teknik dan Informatika, Teknik Informatika, Universitas Dian Nusantara, Jakarta, Indonesia</p> <p>TIN: Terapan Informatika Nusantara Vol 4, No 2, July 2023, page 129-133 ISSN 2722-7987 (Media Online)</p>	<p><a href="https://ejurnal.seminar-id.com/index.php/tin/article/download/4208/2265/">https://ejurnal.seminar-id.com/index.php/tin/article/download/4208/2265/</a></p>		<p>Toko Momy Tika ini meningkatkan efisiensi promosi, pemasaran dan penjualan.</p>
9	<p>(Rahman &amp; Alda, 2024)</p> <p>Sains dan Teknologi, Sistem Informasi,</p>	<p>Aplikasi <i>E-Commerce</i> Ikan Teri dengan Metode <i>Waterfall</i> dan Integrasi Pembayaran <i>Midtrans</i></p>	<p><i>Waterfall</i></p>	<p>Aplikasi <i>e-commerce</i> ini berhasil meningkatkan jangkauan pemasaran</p>

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Universitas Islam Negeri Sumatera Utara, Medan  Journal of Information System Research (JOSH) Volume 6, No. 1, Oktober 2024, pp 155–163 ISSN 2686-228X (media online)	<a href="https://ejurnal.seminar-id.com/index.php/josh/article/download/5943/3058/">https://ejurnal.seminar-id.com/index.php/josh/article/download/5943/3058/</a>		ikan teri dan memberikan kemudahan transaksi melalui integrasi <i>Midtrans</i> . Sistem diuji dan berjalan sesuai harapan dengan peningkatan pengalaman pengguna.
10	(Setiawan et al., 2024)  Sistem Informasi, Universitas Pembangunan	Rancang Bangun <i>Website Store Management System Laravel</i> dengan Metode <i>Agile</i> : Studi Kasus UMKM Toko Jali	<i>Extreme Programming</i>	Integrasi <i>Midtrans</i> meningkatkan kecepatan transaksi & akurasi pencatatan

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Nasional “Veteran” Jawa Timur, Indonesia  Jurnal Pendidikan dan Teknologi Indonesia (JPTI) Vol.4, No. 11, November 2024, Hal. 301-312 p- ISSN:2775- 4227 e- ISSN:2775- 4219	<a href="https://www.journals.iid.ac.id/index.php/jpti/article/view/448">https://www.journals.iid.ac.id/index.php/jpti/article/view/448</a>		penjualan, pelanggan terbantu dalam memilih metode pembayaran, efisiensi data meningkat.
Perbedaan penelitian terdahulu dengan penelitian yang tertulis lakukan adalah:				
11	Alfin Naufal Azhali  Program Studi Teknik Informatika, Fakultas	Perancangan Sistem Informasi <i>Website E-Commerce</i> pada Toko Screamble Menggunakan Metode <i>Waterfall</i>	<i>Waterfall</i>	Sistem informasi <i>e-commerce</i> mandiri untuk UMKM

No	Nama Peneliti	Judul Penelitian	Metode Penelitian	Hasil Penelitian
	Teknik, Universitas Islam Nusantara 2025			dengan integrasi <i>Midtrans</i> dan ongkir otomatis. Fokus pada fleksibilitas bisnis, evaluasi menyeluruh terhadap <i>UX</i> dan keamanan.

## 2.2 Perancangan Sistem

Menurut (Suheri, 2020) Perancangan sistem merupakan tahap berikutnya dari analisis sistem yang bertujuan untuk menciptakan solusi baru untuk menangani masalah yang ada. Kegiatan ini mencakup identifikasi kebutuhan fungsional, perencanaan desain sistem, dan pengaturan elemen perangkat keras serta perangkat lunak agar sistem beroperasi secara efektif dan terintegrasi. Sistem itu sendiri diartikan sebagai sekumpulan komponen yang saling berhubungan dan berkolaborasi untuk mencapai tujuan tertentu dengan cara mengubah input menjadi output yang berguna. Menurut (Nery Nestary, 2020) dalam ranah sistem informasi, perancangan sistem bertujuan untuk menghasilkan desain yang tidak hanya efektif dan efisien tetapi juga mudah digunakan, sehingga dapat mendukung pengambilan keputusan dan fungsi organisasi secara optimal. Karena itu, proses perancangan sistem harus dilakukan secara komprehensif dengan memperhatikan kebutuhan pengguna serta karakteristik sistem yang akan dikembangkan, agar hasil yang diperoleh dapat memenuhi tujuan organisasi dengan tepat dan akurat.

### **2.3 E-Commerce**

*E-Commerce (Electronic Commerce)* adalah jenis transaksi bisnis yang dilakukan secara online dengan menggunakan jaringan internet sebagai media utama. *E-Commerce* saat ini digunakan untuk berbagai tujuan, termasuk penjualan barang dan layanan, konsultasi, dan manajemen pelanggan. Menurut (Alief Naufal & Chotijah Umi, 2024) *e-commerce* membantu UMKM mendapatkan pasar *online* yang lebih besar dan mendukung otomatisasi transaksi yang lebih efisien.

Pelaku bisnis dapat mengurangi biaya operasional dengan menggunakan sistem ini untuk menjangkau konsumen dari berbagai daerah tanpa harus mendirikan cabang fisik. Proses bisnis secara keseluruhan dipercepat dengan penggunaan *e-commerce*, yang memungkinkan integrasi dengan berbagai sistem lain seperti pembayaran digital dan logistik.

### **2.4 Sistem Informasi**

Sistem Informasi merupakan sekumpulan komponen yang saling berhubungan yang digunakan untuk mengumpulkan, memproses, menyimpan, dan menyebarkan informasi untuk membantu proses pengambilan keputusan perusahaan. Menurut (Maulana et al., 2024), sistem informasi berbasis *web* untuk UMKM membuat pengelolaan produk, transaksi, dan interaksi pelanggan lebih mudah.

Sistem informasi adalah dasar untuk menjalankan operasional bisnis berbasis digital. Selain meningkatkan efisiensi kerja, sistem informasi juga menyediakan data analitik yang berguna untuk strategi bisnis di masa mendatang.

### **2.5 Website**

*Website* adalah kumpulan halaman digital yang saling terhubung yang dapat diakses melalui internet. *Website* adalah alat strategi untuk mempromosikan dan berkomunikasi serta menyebarkan informasi untuk lembaga, organisasi, dan masyarakat umum. Menurut (Andik Prakasa Hadi & Faiz Abdul Rokhman, 2020), *website* membantu promosi dan penyebaran informasi pesantren kepada lebih banyak orang. Mereka juga membuat para pengelola dan masyarakat dapat berbicara satu sama lain melalui media tradisional seperti brosur atau spanduk. Karena lebih efektif dan mudah diakses, *website* dapat mengurangi biaya promosi.

Selain itu, (Ahmad et al., 2024) menyatakan bahwa *website* adalah salah satu teknologi yang membantu mempercepat penyebaran informasi dan memudahkan pengelolaan data dan komunikasi antar pengguna dan lembaga dengan masyarakat.

### **2.5.1 Fungsi Website**

*Website* berfungsi sebagai alat komunikasi terus menerus, sumber informasi, media transaksi, dan sumber hiburan. (Andik Prakasa Hadi & Faiz Abdul Rokhman, 2020) menjelaskan bahwa *website* tidak hanya berfungsi untuk menyediakan informasi teks, gambar, suara, dan video tetapi juga berfungsi sebagai alat komunikasi interaktif yang membantu pesantren lebih dekat dengan masyarakat dan meningkatkan pemahaman mereka. Dengan kontak menggunakan fitur atau ruang tanya jawab yang ada di *website*, misalnya. Menurut (Ahmad et al., 2024) *website* berfungsi sebagai media informasi yang memfasilitasi akses data, komunikasi, dan promosi, serta menyediakan layanan transaksi *online* yang praktis dan terorganisir. Di era internet saat ini, *website* dapat membantu seluruh aktivitas organisasi atau lembaga dengan menyediakan fitur seperti galeri, berita, pengumuman, dan dokumen digital.

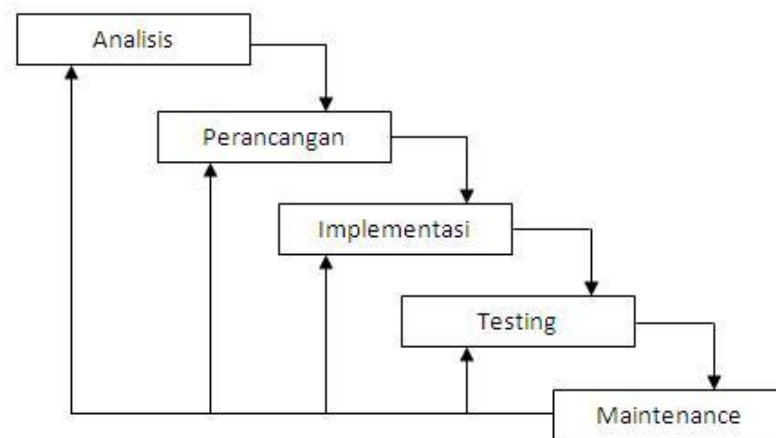
### **2.6 Metode Waterfall**

Metode *Waterfall* adalah pendekatan linier untuk pengembangan perangkat lunak yang terdiri dari beberapa langkah yang dilakukan secara bertahap. Setiap tahap dimulai setelah tahap sebelumnya selesai, tidak ada dua tahap yang dilakukan bersamaan. Hal ini memungkinkan pengembangan dilakukan secara terorganisir dan sistematis, dengan penekanan pada pencapaian tujuan di setiap tahap. (Aceng Abdul Wahid, 2020) menyatakan bahwa metode ini ideal untuk proyek yang memiliki persyaratan yang jelas dan tidak berubah selama proses pengembangan. Model ini terdiri dari beberapa tahap, yaitu:

1. Analisis kebutuhan, mengidentifikasi, dan mendokumentasikan kebutuhan sistem yang akan dikembangkan.
2. Perancangan Sistem, Membuat rancangan teknis dari sistem berdasarkan kebutuhan yang telah dianalisis.
3. Implementasi, Mengembangkan perangkat lunak berdasarkan rancangan yang telah dibuat.

4. Pengujian , Melakukan pengujian terhadap sistem untuk memastikan bahwa sistem berjalan sesuai dengan spesifikasi.
5. Pemeliharaan, Memastikan sistem tetap berjalan dengan optimal setelah diterapkan.

Dalam penelitian ini, metode *Waterfall* digunakan untuk mengembangkan sistem *e-commerce* Toko Screamble agar proses pengembangannya lebih terstruktur dan sistematis.



**Gambar 2. 1** Tahapan Metode *Waterfall*

**Sumber :** <https://adoc.pub/download/sistem-informasi-pendaftaran-wisuda-berbasis-website-pada-un.html>

### 2.6.1 Perbedaan Metode *Waterfall* , *Agile* , dan *Extreme Programming*

Berikut ini adalah penjelasan paragraf lengkap tentang perbedaan metode *Waterfall*, *Agile*, dan *Extreme Programming* :

1. Menurut (Qolbi et al., 2024) metode *Waterfall* adalah model pengembangan perangkat lunak tradisional yang diimplementasikan secara berurutan dan sistematis. Analisis, desain, implementasi, pengujian, dan pemeliharaan adalah beberapa tahap yang dilakukan secara bertahap dan harus diselesaikan satu per satu sebelum melanjutkan ke tahap berikutnya. Struktur yang jelas, dokumentasi lengkap, dan kemudahan pengelolaan proyek adalah keuntungan metode ini. Kekurangannya, bagaimanapun,

adalah sulit untuk menyesuaikan perubahan di tengah proses dan ada kemungkinan besar kesalahan pada tahap awal. Oleh karena itu, metode *Waterfall* ideal untuk proyek yang memiliki persyaratan yang jelas dan tidak berubah.

2. Menurut (Saputra et al., 2024) metode *Agile* adalah teknik pengembangan perangkat lunak yang iteratif dan inkremental yang menggunakan sprint, atau siklus pengembangan yang singkat. *Agile* mengutamakan kerja sama tim pengembang dan pengguna yang kuat dan respons cepat terhadap perubahan. *Agile* memberi nilai bisnis lebih awal daripada *Waterfall* yang lebih terstruktur dan dokumentasi berat karena memprioritaskan komunikasi intensif, pengujian berkelanjutan, dan pengiriman produk yang cepat. Metode *Agile* sangat cocok untuk proyek dinamis dan kompleks yang membutuhkan fleksibilitas tinggi.
3. Menurut (Kurniawan et al., 2023) *Extreme Programming (XP)* merupakan turunan dari metode *Agile*, mengutamakan metode pengembangan seperti *pair programming*, integrasi berkelanjutan, keterlibatan langsung pengguna dalam pengembangan, dan pengembangan berbasis uji. Dengan pengujian otomatis yang konsisten dan kolaborasi intensif, *XP* bertujuan untuk meningkatkan kualitas perangkat lunak dan kemampuan adaptasi perubahan. Namun, metode *XP* membutuhkan komitmen penuh dari tim dan pengguna serta disiplin tinggi. Itu cocok untuk proyek kecil dan menengah dengan kebutuhan yang terus berubah..

## **2.7 Black Box Testing**

Teknik pengujian *Black Box* adalah metode pengujian perangkat lunak yang melihat fungsi sistem hanya dari sisi *input dan output* tanpa melihat struktur kode di dalamnya. (Handayanto & Nuryasin, 2024) mengatakan bahwa metode ini sangat bermanfaat untuk memverifikasi apakah setiap fitur eksternal memenuhi spesifikasi. Ini karena pengujian berfokus pada antarmuka dan perilaku yang langsung dirasakan pengguna, sehingga pemangku kepentingan non-teknis dapat memahaminya dengan mudah. Meskipun demikian, (Handayanto & Nuryasin, 2024) menekankan bahwa berbagai metode dalam pengujian kotak hitam, termasuk

pembagian *equivalence*, analisis nilai batas, tabel keputusan, dan transisi pemerintahan, memungkinkan identifikasi cacat fungsional secara sistematis dan cepat, khususnya untuk aplikasi yang terus berkembang.

Selain itu, kedua penelitian mengingatkan bahwa pengujian dapat mengabaikan kesalahan di tingkat struktur kode atau jalur eksekusi tersembunyi karena mereka tidak melihat logika internal. Oleh karena itu, pengujian *Black Box* idealnya dipadukan dengan pengujian *White Box* untuk mendapatkan cakupan pengujian yang luas.

## **2.8 Teknologi yang Digunakan**

Dalam pengembangan sistem informasi *e-commerce* pada Toko Screamble, digunakan berbagai teknologi untuk mendukung proses perancangan, implementasi, dan pengujian sistem, diantaranya:

### **2.8.1 *React.js***

Menurut (Murti et al., 2021) *ReactJS* adalah pustaka *JavaScript* yang sangat populer yang digunakan untuk membuat komponen antarmuka pengguna (*UI*) yang dapat digunakan kembali. Dengan menggunakan pustaka ini, pengembang dapat membuat aplikasi *web* yang dinamis dan responsif dengan cara yang lebih terstruktur dan efisien. *ReactJS* memungkinkan pengembangan elemen antarmuka yang modular dan mudah dipelihara berkat konsep komponen yang dapat digunakan kembali. Keunggulan utama *ReactJS* adalah kemampuannya untuk menghasilkan pembaruan antarmuka pengguna dengan sangat cepat menggunakan *Virtual DOM*, yang memungkinkan pengembang untuk meningkatkan kinerja aplikasi web secara signifikan. Karena proses pengembangan menjadi lebih cepat dan lebih fleksibel, *ReactJS* menjadi pilihan utama bagi banyak pengembang yang menciptakan aplikasi web interaktif dan kontemporer.

Fungsi dari *React.js* adalah sebagai berikut ini:

1. Membantu dalam pembuatan komponen *UI* yang dapat digunakan kembali.
2. Mempercepat proses render halaman melalui virtual *DOM*.
3. Memungkinkan pengembangan aplikasi *SPA (Single Page Application)* yang responsif.

Adapun kelebihan dari *React.js* adalah:

1. Komponen bersifat modular dan reusable.
2. Dukungan ekosistem dan komunitas yang besar.
3. Mudah diintegrasikan dengan berbagai *library* atau *backend API*.

### **2.8.2 Tailwind CSS**

*Tailwind CSS* adalah *framework CSS* yang dirancang untuk memenuhi persyaratan dasar untuk membangun komponen tampilan *website*. Menurut (Arhandi et al., 2022), *Tailwind* memungkinkan pengembang mengatur berbagai elemen desain seperti margin, ukuran, posisi, warna, dan elemen lainnya tanpa terbatas pada gaya desain *framework* lainnya. Dengan hanya menggunakan kelas-kelas utilitas yang sudah tersedia, *Tailwind* memungkinkan pembuatan komponen web yang fleksibel dan responsif melalui pendekatan *utilitas-first*.

Keunggulan utama *Tailwind CSS* adalah fleksibilitasnya, yang memungkinkan pengembang untuk menyesuaikan desain mereka sesuai keinginan mereka tanpa khawatir tentang ketergantungan pada gaya desain tertentu. Pengembang dapat dengan mudah mengubah elemen desain hanya dengan mengubah kelas yang ada, tanpa perlu menulis *CSS* tambahan secara manual. Hal ini membuat *Tailwind CSS* menjadi alat yang bagus untuk pengembangan *frontend*, terutama untuk membuat antarmuka pengguna yang cepat dan responsif.

Selain itu, *Tailwind CSS* sangat cocok digunakan untuk proyek pengembangan web yang membutuhkan desain yang *fleksibel* dan mudah dikustomisasi karena mengurangi jumlah kode *CSS* yang diperlukan dan meningkatkan pemeliharaan dan keterbacaan kode.

### **2.8.3 Laragon**

Menurut (Padang et al., 2024), Laragon adalah aplikasi yang bisa mengubah sistem pada komputer menjadi *server* lokal atau lingkungan pengembangan web yang lengkap. Aplikasi ini berjalan di sistem operasi *Windows* dan memudahkan pengembang dalam mengatur versi perangkat lunak, mengelola *database*, serta membuat proyek aplikasi dengan tampilan antarmuka yang mudah digunakan. Kecepatan, kemudahan, dan portabilitas Laragon membuat aplikasi ini populer di kalangan pengembang *web* untuk membuat lingkungan pengembangan yang efisien

tanpa perlu melakukan konfigurasi yang rumit. Selain itu, *Laragon* juga memungkinkan pengujian aplikasi pada berbagai versi *PHP* dan *MySQL*, sehingga mendukung pengembangan aplikasi yang responsif dan fleksibel.

#### **2.8.4 MySQL**

*MySQL* adalah Sistem manajemen basis data relasional (*RDBMS*) yang populer dalam pengembangan aplikasi digital, khususnya aplikasi berbasis web dan sistem informasi. Menurut (Hartati, 2022) keunggulan utama *MySQL* yaitu bersifat *open source MySQL*, kinerja yang cepat, dan kemudahan instalasi dan penggunaan pada berbagai sistem operasi seperti *Windows*, *Linux*, dan *macOS*. (Kalsum Siregar et al., 2024) mengatakan bahwa sistem ini sangat fleksibel dan dapat terintegrasi dengan berbagai bahasa pemrograman, seperti *PHP*, *Java*, dan *Python*, sehingga mempermudah proses pembuatan aplikasi dan sistem informasi. *MySQL* mendukung perintah dasar *SQL* untuk pengelolaan data dan menyediakan fitur multithreaded dan multi-*user*, yang menjadikannya solusi pilihan baik untuk bisnis kecil maupun besar.

*MySQL* juga memiliki fitur keamanan data yang luas melalui pengaturan hak akses pengguna dan prosedur backup-recovery yang efisien, selain keunggulan dari sisi kinerja dan kehidupan. Menurut (Eyni Alfia & Waseso, 2020) untuk menjaga efisiensi dan kinerja sistem secara keseluruhan, normalisasi tabel, optimasi *query*, dan penggunaan indeks sangat penting. *MySQL* banyak digunakan di sekolah, bisnis, dan industri karena basis pengguna dan komunitasnya yang besar untuk memenuhi kebutuhan pengelolaan data yang kuat, efektif, dan terorganisir.

#### **2.8.5 Prisma ORM**

*Prisma ORM* merupakan salah satu teknologi Mapping Objek-Relasi (*ORM*) kontemporer, *Prisma ORM* digunakan dalam pengembangan aplikasi web, terutama di lingkungan *Node.js* dan *TypeScript*. Menurut (Widodo et al., 2024) *Prisma ORM* dipilih untuk membangun sistem pemesanan berbasis web untuk layanan fotografi bayi baru lahir di Potret Kecilmu karena kemampuannya untuk menjaga tipe data aman, melakukan migrasi *database* dengan cepat, dan menyediakan pembuat pertanyaan yang mudah digunakan. Penjadwalan, pemesanan, dan pencatatan keuangan secara otomatis dalam satu sistem terintegrasi

menjadi lebih mudah dengan *Prisma ORM*. Implementasi ini meningkatkan kinerja operasional dan kenyamanan pengguna dengan membatasi proses pemesanan jumlah langkah pemesanan berkurang dari 13 menjadi 7 tahapan, yang setara dengan efisiensi sebesar 46,15%.

Selain itu, *Prisma ORM* memungkinkan skema migrasi dan manajemen data menjadi lebih mudah. Ini juga membuat sistem yang dibangun lebih interaktif dan efektif, dan mengurangi kemungkinan kesalahan pencatatan data. Penelitian ini menunjukkan bahwa fitur-fitur *Prisma ORM*, seperti pelengkapan otomatis, keamanan jenis , dan dokumentasi yang kuat, membantu mencapai tujuan utama digitalisasi proses bisnis: meningkatkan pengelolaan pemesanan, mengurangi kesalahan operasional, dan mempercepat pengolahan data pelanggan dalam industri jasa fotografi.

#### **2.8.6 Vite**

*Vite* adalah alat *build* modern yang mendukung berbagai *framework* seperti *React* dan *Vue* dan fokus pada keringkasan dan ekstensibilitas. Tujuan utama *Vite* adalah untuk membahas pola pengembangan aplikasi *web* yang paling umum secara *out-of-the-box*. Salah satu tujuan utamanya adalah untuk mempercepat waktu kompilasi proyek melalui pendekatan *on-demand bundling* yang menggunakan modul *ES* asli pada *browser* kontemporer. Menurut (Alfa Rizzi et al., 2025) metode ini menyajikan kode langsung ke *browser* selama proses pengembangan tanpa memerlukan waktu untuk *pre-bundling*, yang secara signifikan mempercepat proses pengembangan dan meningkatkan pengalaman pengguna. Selain itu, fitur *Hot Module Replacement (HMR)* yang dimana prosesnya hampir instan yang ditawarkan *Vite* memungkinkan pengembang melihat perubahan kode di *browser* secara langsung tanpa rekompilasi penuh atau *refresh* halaman, sehingga mengurangi interupsi dan mengurangi waktu awal dingin.

#### **2.8.7 Axios**

*Axios* merupakan sebuah *library JavaScript* populer yang berfungsi sebagai klien *HTTP* berbasis *Promise* untuk *browser* dan *Node.js*. Menurut (Azizbek Ravshan, 2024) tujuan dari *library* ini adalah untuk mempercepat proses pembuatan permintaan *HTTP* asinkron ke *API* (Rajabov, 2024). *Axios* menawarkan antarmuka

yang mudah digunakan untuk mengirim berbagai jenis permintaan *HTTP*, seperti *GET*, *POST*, *PUT*, dan *DELETE*, dan juga bekerja dengan baik dengan *browser* kontemporer. Peningkatan keterbacaan dan pemeliharaan kode dipengaruhi oleh kemampuan untuk pertukaran data dengan server dan menyediakan sintaks yang bersih dan mudah digunakan.

Kemudahan dalam menangani permintaan *HTTP* asinkron, dukungan penuh terhadap *API Promise*, dan kemampuan untuk menambahkan *header* dan parameter pada permintaan adalah keunggulan utama *Axios*. Penulisan kode asinkron dengan *Axios* dengan sintaks *async* atau *await* menjadi lebih mudah. Hal tersebut membuat alur kontrol lebih mudah dipahami dan dikelola. Selain itu, menurut (Azizbek Ravshan, 2024) *Axios* memiliki fitur seperti *interceptor* untuk permintaan dan *respons*, yang memungkinkan penanganan internasional terhadap kesalahan atau perubahan data sebelum atau sesudah permintaan dikirim. Fitur-fitur ini membuat *Axios* pilihan yang bagus untuk mengelola permintaan *HTTP* dalam proyek *JavaScript* dan aplikasi *web* kontemporer.

### 2.8.8 RajaOngkir

RajaOngkir yaitu salah satu layanan *web service* yang menampilkan informasi ongkos kirim dari berbagai macam kurir di Indonesia, seperti POS Indonesia, JNE, dan TIKI (Nugroho et al., 2021). Untuk memfasilitasi proses cek ongkos kirim secara otomatis, layanan ini menyediakan *Application Programming Interface (API)*, yang dapat diintegrasikan ke dalam aplikasi atau sistem lainnya. Menyediakan data pengiriman yang akurat dan *real-time* adalah tujuan utama RajaOngkir. Menurut (Setyo Nugroho et al., 2021) hal tersebut sangat penting untuk pengembangan aplikasi yang memerlukan fitur perhitungan biaya pengiriman, seperti sistem *e-commerce* atau aplikasi logistik.

Karena menggunakan arsitektur *REST* dengan format balasan *JSON*, yang didukung oleh hampir semua bahasa pemrograman, kelebihan utama RajaOngkir adalah kemudahan integrasinya (Nugroho et al., 2021). RajaOngkir memungkinkan pengembang menambahkan fitur cek ongkos kirim ke dalam aplikasi mereka dengan cepat dan efisien tanpa perlu mengelola tarif data dari setiap kurir secara manual. Hal ini memungkinkan RajaOngkir memenuhi kebutuhan perusahaan

untuk pengiriman data yang digunakan dalam pengembangan aplikasi, meningkatkan efisiensi operasional, dan meningkatkan pengalaman pengguna dengan informasi ongkos kirim yang jelas.

### **2.8.9 Google Chrome**

*Google Chrome* adalah salah satu *browser* web paling populer yang digunakan secara luas di banyak perangkat komputer dan ponsel, dengan banyak fitur yang meningkatkan keamanan, kecepatan, dan kenangan pengalaman penggunaan internet pengguna. Menurut (Ramadhan & Fauzan, 2023), salah satu keunggulan utama *google chrome* adalah kemampuan untuk mendukung ekstensi *browser*, yaitu aplikasi perangkat lunak kecil yang dapat menambahkan berbagai fitur dan fungsi ke *browser web* secara instan. Dengan mendukung ekstensi *browser*, pengguna dapat memperoleh fitur tambahan seperti pemblokiran iklan, peningkatan privasi, dan bahkan otomatisasi tugas tertentu yang tidak tersedia dalam fitur *default browser*.

Studi tersebut juga menyoroti pemanfaatan ekstensi di *google chrome* sebagai solusi pembatasan akses internet yang fleksibel. Berbeda dengan metode pemblokiran berbasis *firewall* yang hanya berjalan pada lingkup jaringan tertentu, ekstensi pada *chrome* dapat diterapkan secara spesifik pada *browser* tanpa terikat jenis jaringan. Salah satu contoh implementasinya adalah ekstensi pembatasan akses pada konten atau domain tertentu yang dapat diatur berikut kategorinya sesuai kebutuhan pengguna. Dengan ekosistem ekstensi *browser* yang berkembang pesat dan *API* terbuka, *Google Chrome* terus berinovasi memperluas layanan dan keamanan bagi pengguna, serta mendukung berbagai kebutuhan mulai dari peningkatan produktivitas, perlindungan data, hingga kontrol akses internet pada lingkungan pendidikan maupun bisnis

## **2.9 Arsitektur Client Server**

Arsitektur *client-server* merupakan suatu konsep perangkat lunak yang membagi peran antara komputer *client* yang melakukan permintaan layanan dan komputer *server* yang menyuplai layanan tersebut dalam sebuah jaringan. Dalam struktur ini, komputer klien bertugas untuk mengirimkan permintaan dan menerima tanggapan, sementara server melaksanakan proses pengolahan dan memberikan

data sesuai permintaan (Yonatan Koentjoro et al., 2023). Pemahaman arsitektur ini sangat penting di zaman *Society 5.0* sebagai landasan untuk memahami inovasi dalam teknologi informasi modern yang mendukung solusi untuk berbagai masalah sosial. Topologi yang sering digunakan adalah topologi bintang, di mana *client* terhubung dengan *server* pusat yang mengelola akses serta komunikasi. Arsitektur *client-server* juga memiliki variasi seperti *1-tier*, *2-tier*, dan *3-tier* yang masing-masing memiliki tingkat kompleksitas serta manfaat dalam aspek keamanan dan efisiensi operasional (Yonatan Koentjoro et al., 2023). Penerapan arsitektur ini memungkinkan sumber daya dikelola secara terpusat sehingga mempermudah dalam pengawasan, pemeliharaan, serta meningkatkan keamanan dan kemampuan skala sistem informasi..

## 2.10 Evaluasi Sistem

Evaluasi sistem dilakukan untuk memastikan bahwa sistem yang dikembangkan sesuai dengan kebutuhan pengguna dan berfungsi dengan baik.

Metode Evaluasi:

1. Pengujian Fungsionalitas , Memeriksa apakah setiap fitur bekerja sesuai dengan spesifikasi.
2. Pengujian kinerja, Mengukur seberapa cepat dan responsif sistem dalam menangani permintaan pengguna.
3. Evaluasi pengalaman pengguna (*UX*), Menganalisis kepuasan pengguna terhadap tampilan dan kemudahan penggunaan sistem.
4. Evaluasi ini akan diterapkan pada sistem *e-commerce* Toko Screamble untuk memastikan kualitas dan fungsionalitasnya.

## 2.11 *Unified Modeling Language (UML)*

*Unified Modeling Language (UML)* merupakan bahasa pemodelan standar yang digunakan secara luas dalam pengembangan perangkat lunak, terutama untuk sistem yang berfokus pada objek. Melalui berbagai jenis diagram, seperti *use case diagram*, *class diagram*, *sequence diagram*, dan *activity diagram*, *UML* membantu proses analisis, desain, dan dokumentasi sistem. (Nistrina & Sahidah, 2022) menyatakan bahwa *UML* memudahkan waktu pengembang untuk menjelaskan secara terstruktur struktur dan perilaku sistem karena merupakan produk dari

koneksi metode pemodelan objek sebelumnya dan telah menjadi bahasa standar yang dapat digunakan di seluruh metodologi rekayasa perangkat lunak. (Nurshadrina & Voutama, 2022) menyatakan pendapat serupa, *UML* menawarkan notasi lengkap untuk visualisasi semua aspek sistem perangkat lunak, dari kebutuhan hingga interaksi antar komponen. Dengan demikian, *UML* dapat mengurangi keraguan dalam komunikasi di antara anggota tim pengembang.

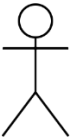
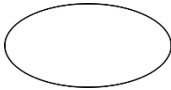
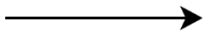
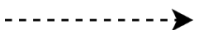
Selain itu, terbukti bahwa penerapan *UML* membantu dalam meningkatkan kualitas dan efisiensi hasil pengembangan sistem informasi, baik dalam proyek pelayanan publik maupun sektor bisnis kecil. Studi (Puspita & Anggita, 2020) menemukan bahwa penerapan *UML* dalam perancangan proses sistem informasi pada UKM di Kota Pagaralam dapat mempercepat proses analisis kebutuhan, membuat proses pembuatan desain sistem yang terstruktur lebih mudah, dan mengurangi kemungkinan kesalahan desain sebelum implementasi. Secara keseluruhan, *UML* sangat direkomendasikan karena membantu proses dokumentasi sistem, memudahkan proses waktu komunikasi, dan mengurangi risiko kesalahan dan miskomunikasi dalam pembangunan perangkat lunak.

### **2.11.1 Use Case Diagram**

Salah satu komponen *Unified Modeling Language (UML)* adalah *use case* diagram. Diagram ini menunjukkan kebutuhan dan fungsi utama yang diharapkan dari sebuah perangkat lunak serta menggambarkan interaksi antara aktor dan sistem. *Use case* diagram adalah diagram yang dapat menggambarkan sistem secara global berdasarkan fungsionalitas, yaitu layanan apa saja yang dapat diberikan sistem kepada aktor yang berinteraksi dengan sistem tersebut (Nistrina & Sahidah, 2022). Menurut (Puspita & Anggita, 2020) *use case* diagram membantu menganalisis dan menentukan kebutuhan sistem dengan jelas karena mereka dapat menunjukkan semua orang yang berinteraksi dan proses apa yang terjadi dalam sistem informasi. Menurut (Nurshadrina & Voutama, 2022), diagram ini sangat penting sebagai langkah awal untuk mengidentifikasi kebutuhan fungsional selama proses pengembangan sistem informasi.

Pada *use case diagram* terdapat beberapa simbol yang ada sebagai berikut:

**Tabel 2. 2** Simbol pada *Use Case Diagram*

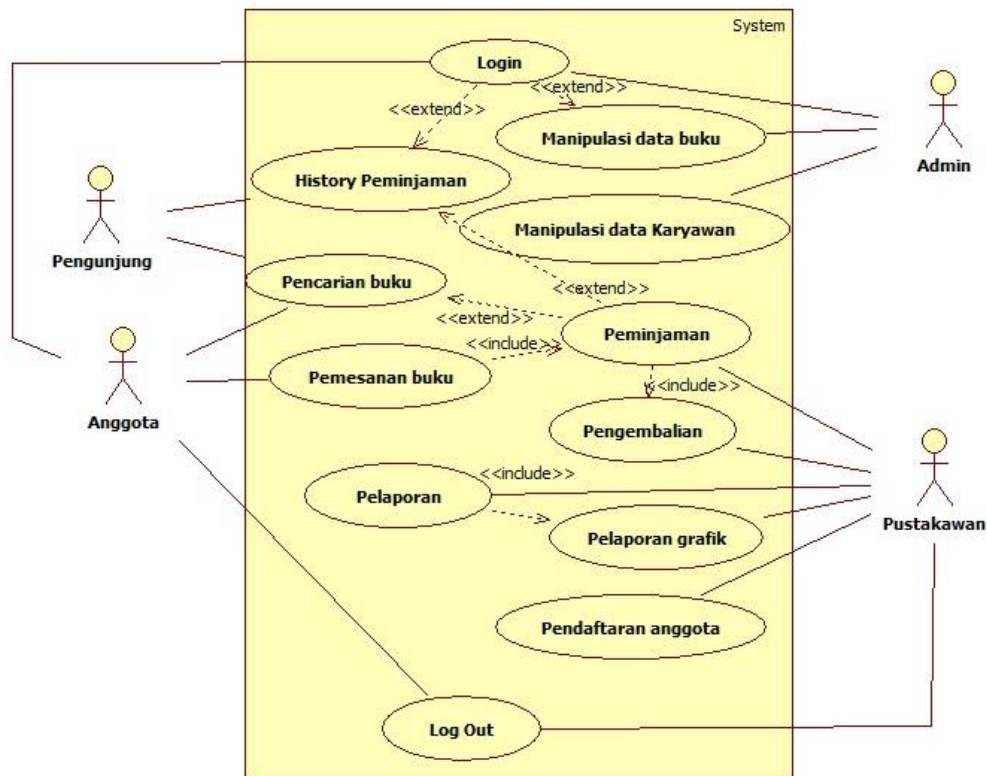
No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menggambarkan peran orang, sistem lain, atau alat untuk berkomunikasi dengan <i>use case</i> .
2		<i>Use Case</i>	Merepresentasikan fungsi atau tugas yang dilakukan oleh aktor dalam sistem.
3		<i>Association</i>	Garis penghubung yang menghubungkan antara aktor dan <i>use case</i> .
4		Generalisasi	Menunjukkan hubungan pewarisan antara dua aktor atau dua <i>use case</i> yang lebih spesifik mewarisi atribut dan perilaku dari yang lebih umum.

Komponen dalam *Use Case Diagram* :

1. Aktor, yaitu entitas yang berinteraksi dengan sistem.
2. *Use Case*, yaitu fungsi atau layanan yang disediakan oleh sistem.
3. Relasi, yaitu hubungan antara aktor dan *use case* yang menggambarkan interaksi.

Berdasarkan simbol pada *use case diagram* diatas, maka dapat digambarkan

sebagai contoh berikut:



**Gambar 2. 2** Contoh *Use Case Diagram*



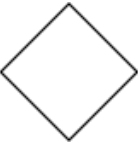


Sumber : <https://badoystudio.com/contoh-use-case/>

### 2.11.2 *Activity Diagram*

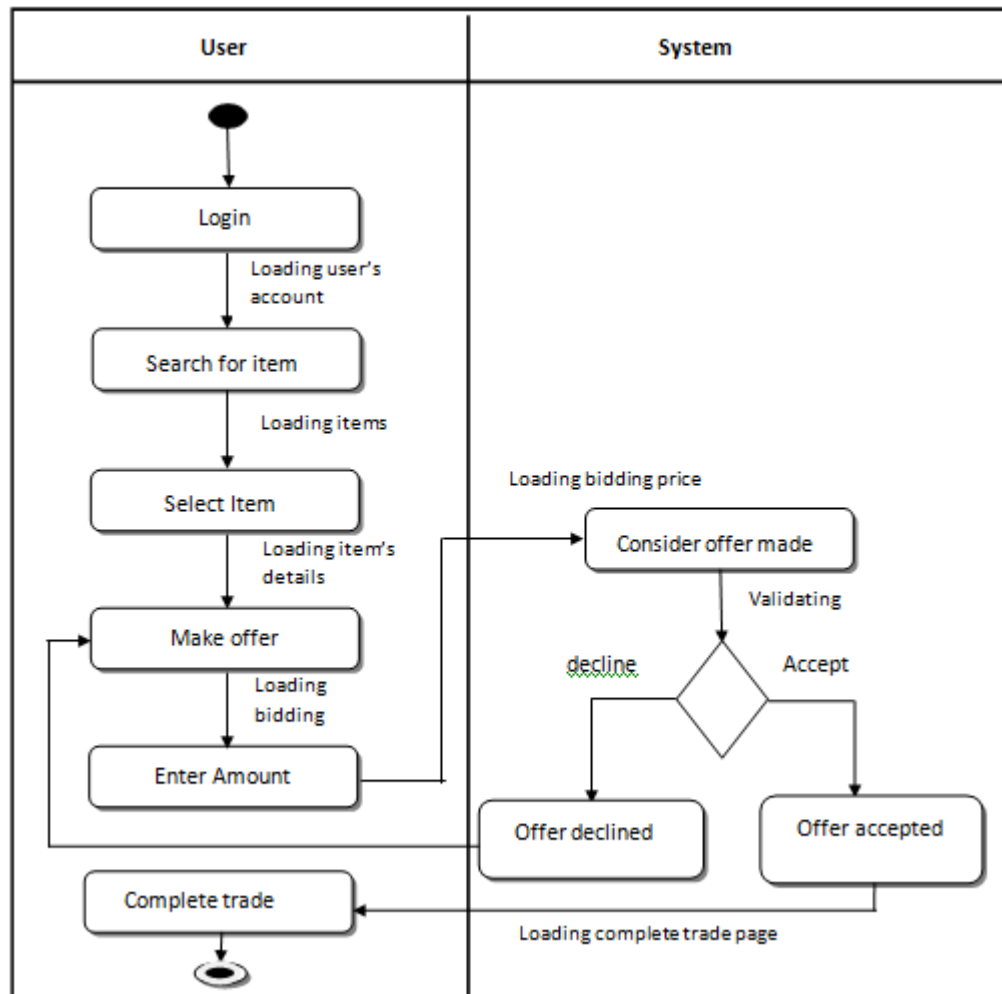
Salah satu jenis diagram dalam *Unified Modeling Language (UML)* adalah *Activity Diagram*, yang digunakan untuk memodelkan alur aktivitas atau alur kerja dalam suatu sistem perangkat lunak. Dalam penelitian yang dilakukan oleh (Fu'adi et al., 2022), *Activity Diagram* menunjukkan urutan proses bisnis utama di Sistem Informasi Akademik Akademi Komunitas Negeri Pacitan (AKNP). Proses ini dimulai dari aktivitas awal hingga aktivitas terakhir, seperti kegiatan perkuliahan. Diagram aktivitas membantu pengembang dan analis sistem memahami urutan langkah-langkah dan percabangan yang terjadi. Ini membuat gambaran yang jelas dan runtut serta memudahkan analisis kebutuhan dan tahap implementasi sistem informasi akademik secara visual dan terstruktur. Pada *activity diagram* terdapat

beberapa simbol yang ada sebagai berikut:

**Tabel 2. 3** Simbol pada *Activity Diagram*

No	Simbol	Nama	Kegiatan
1		<i>Initial Node</i>	Menjelaskan aktivitas status awal/ mulai.
2		<i>Activity</i>	Memperlihatkan bagaimana setiap kelas antarmuka saling berinteraksi satu sama lain.
3		<i>Decision</i>	Menjelaskan aktivitas lebih dari satu untuk menentukan suatu keputusan.
4		<i>Activity Final node</i>	Menjelaskan aktivitas akhir sistem.
5		<i>Fork Node</i>	Pada tahap tertentu satu aliran akan berubah menjadi beberapa aliran

*Activity Diagram* dapat dibuat dengan menggunakan simbol-simbol berikut:



**Gambar 2. 3** Contoh *Activity Diagram*

**Sumber :** [https://www.researchgate.net/figure/Activity-Diagram-for-Buy-Item\\_fig4\\_323535611](https://www.researchgate.net/figure/Activity-Diagram-for-Buy-Item_fig4_323535611)

Komponen dalam *Activity Diagram*:


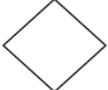
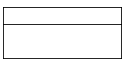

1. *Initial Node* : Awal proses.
2. *Activity* : Tugas atau aksi yang dilakukan.
3. *Decision Node* : Poin pengambilan keputusan dengan jalur alternatif.
4. *Final Node* : Akhir proses.

### 2.11.3 *Class Diagram*

Menurut (Ramdany et al., 2024), *class diagram* adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk

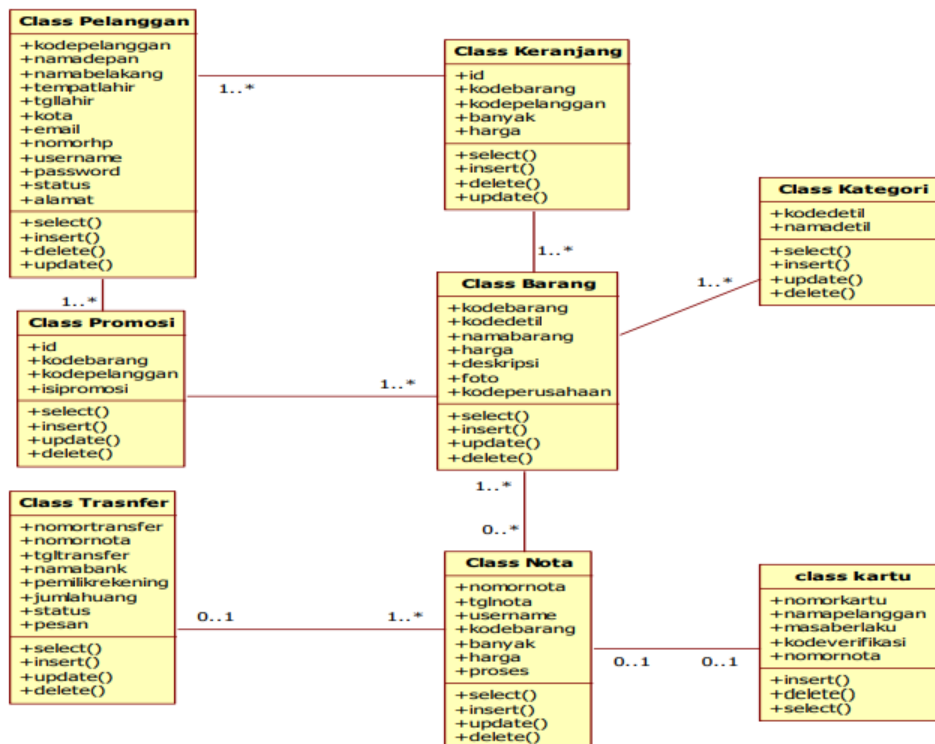
menggambarkan struktur statis dari suatu sistem perangkat lunak. Diagram ini menampilkan kelas-kelas, sifat-sifatnya, metode-metode yang dimilikinya, serta hubungan antar kelas. *Class diagram* sangat penting dalam pengembangan sistem berbasis objek karena membantu menyusun spesifikasi kelas yang nanti akan dijadikan objek. Selain itu, diagram ini juga menjelaskan berbagai hubungan antar kelas, seperti pewarisan, asosiasi, dan agregasi, yang memudahkan dalam merancang struktur data dan interaksi antar entitas dalam sistem. *Class diagram* juga digunakan secara efektif dalam merancang *database* serta hubungan antar tabel. Sebagai contohnya, dalam aplikasi sistem simpan pinjam koperasi, diagram ini membantu pengembang dalam menganalisis dan merancang sistem secara terstruktur serta menyeluruh.. Ini memungkinkan pengembang untuk memenuhi kebutuhan sistem informasi koperasi. Terdapat simbol yang ada pada class diagram yaitu sebagai berikut :

**Tabel 2. 4** Simbol Simbol pada *Class Diagram*

No	Simbol	Nama	Kegiatan
1		<i>Generalization</i>	Hubungan antara objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.

No	Simbol	Nama	Kegiatan
5	←-----	<i>Realization</i>	Operasi yang benar benar dilakukan oleh suatu objek.
6	----->	<i>Dependency</i>	Hubungan perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7	_____	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.

Berdasarkan simbol pada gambar diatas, dapat digambarkan melalui *class diagram* sebagai berikut.



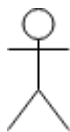
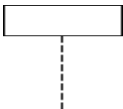

**Gambar 2. 4** Contoh *Class Diagram*

Sumber: D. Risan Q, STMIK Pontianak

#### 2.11.4 Sequence Diagram

Menurut (Sastra, 2021) *sequence diagram* merupakan salah satu bagian dari UML yang sangat penting dalam perancangan sistem informasi. Diagram ini mampu menggambarkan interaksi antara objek secara berurutan, mengikuti jalur use case yang ada, dan membantu dalam memahami proses yang berlangsung dalam sistem. Dalam penelitian yang berkaitan dengan desain sistem informasi untuk penggajian, penerapan Diagram Urutan terbukti efektif untuk menggambarkan alur interaksi antara aktor dan sistem, yang pada gilirannya memperjelas proses bisnis dalam sistem yang sedang dikembangkan.





**Tabel 2. 5** Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	Aktor	Merupakan entitas luar yang berinteraksi langsung dengan sistem.
	<i>Life Line</i>	Garis vertikal putus-putus untuk menunjukkan keberadaan objek atau aktor pada saat proses berlangsung.
	Message	Panah yang menunjukkan aliran pesan atau aktivitas antar aktor dan objek

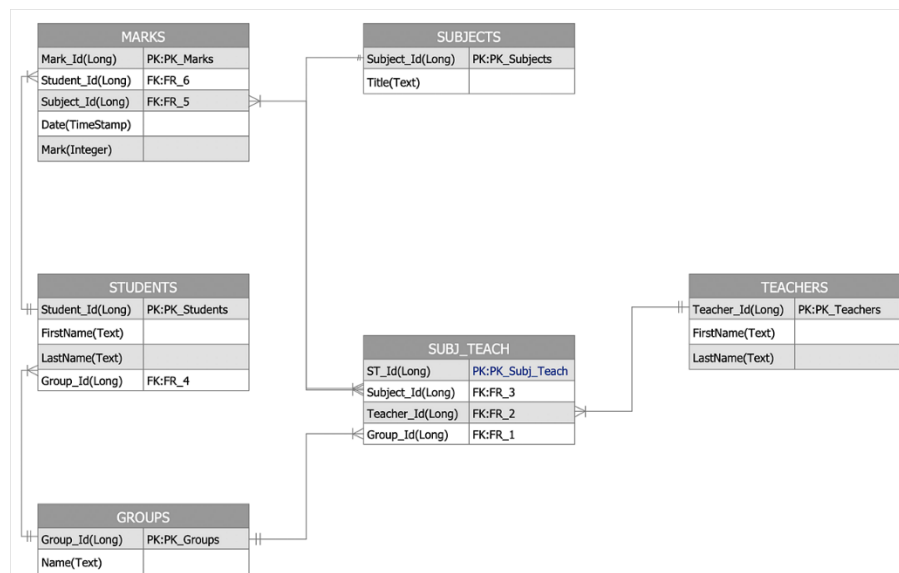
#### 2.11.5 Entity Relationship Diagram (ERD)

Menurut (Pulungan et al., 2022), *entity relationship diagram (ERD)* adalah instrumen pemodelan data yang sangat penting dalam merancang basis data, karena menunjukkan entitas, atribut, dan hubungan antar entitas dengan cara yang teratur sebagai fondasi sebelum pelaksanaan. *ERD* berfungsi sebagai langkah awal dalam desain database, memungkinkan pengenalan kebutuhan data dan struktur hubungan dengan akurat, serta menurunkan kemungkinan terjadinya kesalahan konseptual, prosedural, atau teknis dalam perancangan basis data.

**Tabel 2. 6** Simbol *Entity Relationship Diagram (ERD)*

Gambar	Nama	Keterangan
	<i>One</i>	Satu entitas yang hanya bisa berhubungan dengan satu entitas lain.
	<i>Many</i>	Satu entitas bisa berhubungan dengan banyak entitas lain
	<i>One (and only one)</i>	Relasi harus ada tepat satu.
	<i>One or many</i>	Relasi bisa satu atau bisa banyak entitas.

Berdasarkan simbol pada gambar diatas, dapat digambarkan melalui *entity relationship diagram* sebagai berikut.



**Gambar 2. 5** Contoh *ERD*

**Sumber :** <https://www.conceptdraw.com/helpdesk/how-to-make-crows-foot-erd>