



BAB 1 LANDASAN TEORI

1.1 Penelitian Terdahulu

Fadhilah, A. dan Rizki, M. (2023), Rancang Bangun Aplikasi Pembukuan Keuangan UMKM Berbasis Website Penelitian ini berfokus pada pengembangan aplikasi berbasis web yang membantu pelaku Usaha Mikro, Kecil, dan Menengah (UMKM) dalam melakukan pembukuan keuangan secara efisien, akurat, dan terorganisir. Penulis memilih metode Waterfall, yang diawali dengan analisis kebutuhan pengguna UMKM melalui wawancara dan survei. Dari hasil analisis ditemukan bahwa banyak pelaku UMKM masih bergantung pada pencatatan manual, seperti buku tulis atau spreadsheet sederhana, yang rentan terhadap kesalahan manusia dan sulit untuk dikelola saat volume transaksi meningkat. Aplikasi ini dirancang untuk mencakup fitur-fitur seperti: Pencatatan transaksi keuangan (pemasukan dan pengeluaran). Manajemen piutang dan utang, termasuk pengingat jatuh tempo. Laporan keuangan otomatis (laporan laba rugi, arus kas, dan neraca). Dashboard interaktif, yang menyajikan data keuangan dalam bentuk grafik dan diagram. Setelah implementasi, aplikasi diuji dengan metode *Usability Testing* kepada 30 pelaku UMKM dari berbagai sektor. Hasilnya menunjukkan tingkat kepuasan sebesar 87%, dengan pengguna mengapresiasi kemudahan navigasi dan pengelolaan laporan. Penelitian ini menegaskan bahwa aplikasi berbasis web dapat meningkatkan efisiensi dan keandalan pembukuan keuangan bagi UMKM, sekaligus memberikan wawasan penting bagi pelaku usaha dalam pengambilan keputusan keuangan.

Utami, N. dan Subakti, E. (2022), Rancang Bangun Sistem Informasi Akuntansi Berbasis Website menggunakan Framework Laravel Penelitian ini bertujuan mengembangkan sistem informasi akuntansi berbasis web yang terintegrasi, dengan tujuan meningkatkan kinerja pembuatan laporan keuangan. Framework Laravel dipilih karena fleksibilitasnya



dalam pengembangan sistem yang kompleks dan skalabel. Penelitian ini menggunakan pendekatan Agile, yang memungkinkan pengembangan iteratif dengan siklus pendek, serta memberikan ruang untuk menerima feedback dari pengguna secara cepat. Fitur utama sistem meliputi: Manajemen jurnal akuntansi untuk mencatat transaksi harian. Pembuatan laporan keuangan otomatis, seperti laporan laba rugi, arus kas, dan neraca. Pelacakan arus kas secara real-time melalui integrasi dengan sistem pembayaran online. Manajemen aset tetap, yang membantu pelacakan nilai depresiasi aset. Penelitian ini melibatkan uji coba sistem pada dua perusahaan kecil menengah di bidang jasa. Hasil evaluasi menunjukkan bahwa sistem ini berhasil mempercepat pembuatan laporan keuangan hingga 50%, serta mengurangi kesalahan pencatatan akibat human error. Penulis juga menyoroti peran Agile dalam meningkatkan efisiensi pengembangan, khususnya dalam memperbaiki sistem berdasarkan masukan pengguna.

Nisa, A. dan Prasetyo, D. (2023), Rancang Bangun Sistem Informasi Keuangan pada PT. Secret Discoveries Travel and Leisure Berbasis Web Fokus penelitian ini adalah pada pengembangan sistem informasi keuangan berbasis web untuk perusahaan yang bergerak di sektor travel dan leisure. Metode penelitian yang digunakan adalah kualitatif, di mana wawancara mendalam dan observasi digunakan untuk menggali kebutuhan sistem dari sisi operasional dan manajemen perusahaan. Sistem dirancang untuk memenuhi kebutuhan spesifik, termasuk: Manajemen transaksi keuangan untuk layanan pemesanan tiket, akomodasi, dan paket wisata. Pembuatan laporan keuangan otomatis, seperti laporan pendapatan, laporan biaya, dan profitabilitas. Analisis data keuangan berbasis grafik, yang membantu manajemen dalam memahami pola pendapatan dan pengeluaran. Integrasi dengan sistem reservasi online, yang memastikan sinkronisasi data transaksi keuangan secara real-time. Setelah implementasi, sistem diuji di lingkungan kerja nyata selama tiga bulan. Hasil pengujian menunjukkan peningkatan efisiensi pengelolaan keuangan hingga 40%, serta membantu manajemen dalam mengidentifikasi peluang peningkatan profitabilitas melalui analisis data keuangan yang terintegrasi.



Dwi, S. dan Amelia, R. (2023), Sistem Informasi Keuangan Berbasis Web Menggunakan Pendekatan Agile Penelitian ini menggunakan pendekatan Agile dalam pengembangan sistem informasi keuangan berbasis web. Agile dipilih untuk memastikan bahwa sistem dapat terus disesuaikan dengan kebutuhan pengguna yang berubah-ubah, serta untuk memfasilitasi kolaborasi tim pengembang dan pengguna akhir. Sistem yang dikembangkan mencakup fitur: Manajemen anggaran, yang membantu perencanaan keuangan organisasi. Pelaporan keuangan otomatis, termasuk laporan anggaran, laporan pengeluaran, dan laporan keuangan tahunan. Monitoring transaksi secara real-time, dengan notifikasi otomatis untuk transaksi penting. Penelitian ini melibatkan studi kasus pada sebuah lembaga pendidikan yang membutuhkan transparansi tinggi dalam pengelolaan dana. Implementasi Agile memastikan bahwa fitur yang dikembangkan langsung disesuaikan berdasarkan feedback pengguna. Hasil evaluasi menunjukkan bahwa sistem ini berhasil meningkatkan akurasi laporan keuangan hingga 30% dan mempersingkat waktu proses pelaporan sebesar 40%.

Siti, R. dan Budi, W. (2023), Rancang Bangun Sistem Informasi Laporan Keuangan pada SMP Nasional Berbasis Web Penelitian ini bertujuan untuk mengembangkan sistem informasi laporan keuangan berbasis web guna meningkatkan efisiensi dan transparansi pengelolaan keuangan sekolah. Metode Rapid Application Development (RAD) digunakan untuk mempercepat proses pengembangan, dengan melibatkan pengguna aktif dalam setiap tahap, mulai dari perancangan hingga pengujian. Fitur sistem meliputi: Pencatatan transaksi harian, seperti pembayaran siswa dan pengeluaran operasional sekolah. Pembuatan laporan keuangan otomatis, yang dapat diakses oleh manajemen dan komite sekolah. Dashboard keuangan, yang menyajikan data dalam bentuk grafik untuk mempermudah analisis. Notifikasi otomatis, untuk mengingatkan manajemen terkait tenggat pembayaran atau kebutuhan alokasi dana. Penelitian ini menggarisbawahi pentingnya partisipasi pengguna dalam proses pengembangan untuk memastikan sistem yang dihasilkan sesuai dengan kebutuhan nyata. Hasil evaluasi menunjukkan bahwa sistem ini berhasil mengurangi



kesalahan pencatatan hingga 50%, serta memberikan transparansi yang lebih baik kepada semua pemangku kepentingan.



Tabel 1.1 daftar penelitian-penelitian terdahulu yang terkait

No.	Nama Penelitian	Metode	Persamaan	Perbedaan	Hasil
1	Fadhilah dan Rizki (2023) - Rancangan Aplikasi Pembukuan Keuangan Berbasis Website	Metode Waterfall: Metode ini mengikuti tahapan yang berurutan, mulai dari analisis kebutuhan, desain, pengembangan, pengujian, hingga implementasi.	Semua penelitian fokus pada sistem informasi berbasis web untuk keuangan.	Penelitian ini khusus untuk pengelolaan keuangan UMKM dengan fitur pembukuan dan laporan.	Aplikasi berbasis web yang memudahkan UMKM dalam melakukan pembukuan keuangan secara efektif dan efisien.
2	Utami Subakti (2022) - Rancangan Sistem Informasi Akuntansi Berbasis Website menggunakan Framework Laravel	Metode Agile: Metode ini menekankan kolaborasi tim dan adaptasi cepat terhadap perubahan, dengan pengembangan dilakukan secara iteratif dan	Menggunakan teknologi web untuk sistem informasi akuntansi yang memudahkan pengguna dalam akses data informasi akuntansi.	Menggunakan framework Laravel, yang memungkinkan pengembangan yang lebih cepat dan fleksibel.	Sistem informasi akuntansi berbasis web yang terintegrasi dan responsif terhadap kebutuhan pengguna.web yang terintegrasi



No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil
3	Nisa dan Prasetyo (2023) - Rancangan Sistem Informasi Keuangan PT. Discoveri Travel and Leisure Web	Metode Kualitatif dan Deskriptif: Menggunakan wawancara dan observasi untuk mengumpulkan data, kemudian dianalisis secara deskriptif untuk menggambarkan kondisi saat ini.	Fokus pada kebutuhan sektor keuangan dan pengelolaan informasi yang akurat untuk pengambilan keputusan.	Dikhususkan untuk sektor industri travel dan leisure, dengan fitur yang relevan untuk bisnis tersebut.	Sistem informasi keuangan yang efektif dan efisien untuk pengelolaan laporan keuangan di industri travel dan leisure.
4	Dwi dan Amelia (2023) - Informasi Keuangan Berbasis Web Menggunakan Pendekatan Agile	Metode Agile: Pendekatan ini melibatkan sprint pendek untuk pengembangan, di mana fitur dikembangkan dan diuji secara bersamaan, serta	Menggunakan pendekatan modern dalam pengembangan yang menekankan kolaborasi dan adaptasi.	Menekankan proses pengembangan yang cepat dan iteratif, berbeda dari metode tradisional.	Aplikasi sistem informasi keuangan yang fleksibel, memungkinkan update dan perubahan yang cepat sesuai kebutuhan



No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil
		umpan balik pengguna diperoleh secara terus-menerus.			pengguna.
5	Siti, R. dkk (2019) - Rancangan Sistem Informasi Laporan Keuangan pada SMP Berbasis Web	Metode Rapid Application Development (RAD): Metode ini berfokus pada pengembangan yang cepat melalui prototyping, dengan partisipasi pengguna yang aktif dalam setiap tahap.	Fokus pada pengembangan sistem informasi berbasis web untuk laporan keuangan yang mudah diakses.	Dikhususkan untuk institusi pendidikan (SMP) dengan fitur pelaporan yang sesuai untuk siswa.	Sistem informasi laporan keuangan yang efisien dan mudah digunakan oleh pihak sekolah untuk pengelolaan laporan keuangan.



1.2 Kajian Pustaka

Dalam bab ini, kajian pustaka akan menguraikan konsep-konsep penting yang menjadi landasan dalam perancangan dan pembangunan sistem informasi keuangan berbasis web di Puskesmas Jogoloyo. Kajian pustaka ini mencakup pengertian sistem informasi keuangan, teknologi informasi dalam pengelolaan keuangan, serta pemrograman berbasis *PHP* dan *MySQL* yang digunakan dalam pengembangan aplikasi ini. Setiap sub-bagian dalam kajian pustaka ini disusun untuk mendukung perancangan sistem yang dapat meningkatkan efisiensi pengelolaan keuangan di Puskesmas.

1.2.1 Sistem Informasi Keuangan

Sistem Informasi Keuangan adalah sistem yang mengintegrasikan teknologi informasi dengan manajemen keuangan untuk mengumpulkan, menyimpan, mengelola, dan melaporkan informasi keuangan secara otomatis. Menurut *Romney & Steinbart (2020)*, sistem ini biasanya terdiri dari beberapa komponen utama, yaitu perangkat lunak, perangkat keras, basis data, dan prosedur yang memungkinkan pengelolaan data keuangan yang lebih efektif.

Sistem informasi keuangan yang berbasis web memberikan sejumlah manfaat dibandingkan sistem konvensional, di antaranya:

- a. **Akses Real-Time:** Data keuangan dapat diakses secara cepat dan akurat dari berbagai perangkat yang terhubung dengan internet, memudahkan manajemen dalam mengambil keputusan.
- b. **Otomatisasi Proses:** Dengan fitur otomatisasi, tugas-tugas seperti pencatatan transaksi, penghitungan laporan keuangan, dan audit dapat dilakukan dengan lebih cepat dan minim kesalahan.



- c. **Transparansi dan Akuntabilitas:** Sistem ini memungkinkan pelacakan penggunaan dana secara lebih transparan dan memberikan akses kepada auditor atau pihak terkait untuk memantau alur keuangan.

Di Puskesmas Jogoloyo, sistem ini dirancang dengan menggunakan *PHP* sebagai bahasa pemrograman untuk logika aplikasi, serta *MySQL* sebagai sistem manajemen basis data. Penggunaan *PHP* dan *MySQL* dipilih karena kedua teknologi ini mudah diimplementasikan, memiliki fleksibilitas tinggi, serta mendukung pengelolaan data yang aman dan terstruktur.

1.2.2 Perancangan

Perancangan merujuk pada proses identifikasi proses dan data yang diperlukan oleh sistem yang baru dikembangkan. Tahap perancangan sistem memberikan gambaran lengkap tentang struktur yang direncanakan, berfungsi sebagai panduan bagi para pengembang aplikasi. Sesuai dengan elemen-elemen yang terkomputerisasi dalam sistem, aspek-aspek yang harus didesain pada tahap ini meliputi perangkat keras (*hardware*), perangkat lunak (*software*), basis data, dan aplikasi (Mardiyah, Saputra, & Safii, 2023). Menurut Berto Nadeak perancangan diidentifikasi sebagai tahap awal dalam proses pembangunan rekayasa transaksi atau sistem. Perancangan menggambarkan proses penerapan berbagai teknik dan prinsip dengan tujuan mendefinisikan secara detail peralatan, proses, atau sistem tertentu, sehingga memungkinkan untuk dilakukannya realisasi fisik (Sitorus & Sakban, 2021).

1.2.3 Proses Bisnis

Pemodelan proses bisnis merupakan sebuah diagram umum yang digambarkan secara implisit dan saling berkaitan satu sama lain antar sistem ataupun manusia baik di



dalam organisasi maupun di luar organisasi dengan tujuan untuk melakukan peacatatan ataupun pertimbangan serta melakukan improvisasi pada proses bisnis di yang akan datang. Pada permodean proses bisnis dibutuhkan sebuah tindakan ataupun kegiatan kerja sumber daya (*Resource*) yang dimiliki yang memiliki peran sebagai *input* dan akan menghasilkan sebuah *output* dengan sebuah nilai yang berguna bagi perusahaan maupun pelanggan (Faruq, Lubis, & Fajrillah, 2023).

1.3 UML (Unified Modeling Language)

Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (*OOP*), serta bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah metodologi pengembangan sistem berorientasi objek (*OOP*) dan sekelompok perangkat untuk mendukung pengembangan sistem tersebut. *UML* pertama kali diperkenalkan oleh Object Management Group pada tahun 1980-an dan sekarang banyak digunakan oleh para praktisi *OOP*.

UML juga merupakan dasar bagi perangkat desain berorientasi objek (*OOP*) dari IBM dan juga merupakan standar visualisasi. (Ramadhanti, 2023) Untuk mengetahui pendapat para ahli mengenai pengertian *UML*, berikut pemaparannya.

- a. **Menurut Booch**, *UML* merupakan bahasa standar yang digunakan dalam perancangan sebuah sistem.
- b. **Menurut Nugroho**, *UML* merupakan metode kolaborasi antara metode-metode booch, *OMT* (*Object Modeling Technique*), serta *OOSE* (*Object Oriented Software Engineering*) yang digunakan untuk menganalisa perancangan sistem.
- c. **Menurut Herlawati**, *UML* merupakan kumpulan



diagram yang terdiri dari sembilan diagram atau delapan diagram.

Berikut ini adalah beberapa fungsi *UML* yang perlu diketahui. Diantaranya adalah:

1. Memodelkan Sistem Perangkat Lunak

Unified Modeling Language menggambarkan berbagai aspek sistem perangkat lunak, seperti struktur, perilaku, dan interaksi antara objek.

2. Meningkatkan Komunikasi dan Kolaborasi

Unified Modeling Language membantu pengembang berkomunikasi dengan baik dan menyederhanakan pemodelan sistem yang kompleks dengan memberikan notasi grafis yang dapat digunakan untuk menampilkan ide dan konsep yang rumit secara visual.

3. Meningkatkan Kualitas Perangkat Lunak

Unified Modeling Language membantu pengembang perangkat lunak mengurangi kesalahan dan meningkatkan kualitas transaksi akhir. Dengan memodelkan sistem secara visual, pengembang dapat menemukan masalah dan kesalahan sebelum transaksi akhir dibuat.

4. Meningkatkan Efisiensi Pengembangan Perangkat Lunak

Unified Modeling Language memudahkan pengembang untuk mengembangkan perangkat lunak dengan cepat dan efisien karena meningkatkan pemahaman sistem secara keseluruhan dan mengurangi kompleksitas.

5. Memastikan Kepuasan Klien

Dalam pengembangan perangkat lunak, membantu memastikan bahwa transaksi akhir sesuai dengan kebutuhan dan persyaratan klien. Para pengembang dapat memperjelas dan menyederhanakan pemodelan sistem dengan menggunakan *Unified Modeling Language*. (Faulina, 2023)



Ada beberapa jenis dan juga bagian dari UML, di antaranya yaitu :

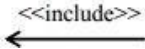
1. Use Case Diagram

Untuk menggambarkan model proses bisnis dari sudut pandang pengguna sistem, diagram use case digunakan. Diagram ini terdiri dari *aktor* dan *use case*. *Aktor* mewakili orang yang akan mengoperasikan sistem atau orang yang berinteraksi dengan sistem aplikasi yang dibuat, sedangkan *use case* mewakili operasi yang dilakukan oleh aktor.

Tabel 2. 2 Use Case Diagram

Simbol	Nama	Keterangan
	<i>Actor</i>	Mewakili peran orang, sistem yang lain, atau alat komunikasi dengan use case
	<i>Use case</i>	Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek yang lainnya



Simbol	Nama	Keterangan
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
	<i>generalization</i>	Hubungan dimana objek anak (dependent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
	<i>Include</i>	Menunjukkan bahwa suatu use case



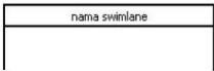
Simbol	Nama	Keterangan
		seluruhnya merupakan fungsionalitas dari use case lainnya
	<i>Extend</i>	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi Terpenuhi
	<i>Sistem</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2. Activity Diagram

Activity Diagram dibuat untuk menjelaskan aktivitas komputer dan alur aktivitas organisasi. Ini adalah bentuk visual dari alir kerja yang berisi tindakan dan aktivitas user dalam sebuah sistem aplikasi. Diagram ini menunjukkan alur sistem aplikasi secara garis besar.



Tabel 2. 3 Activity Diagram

Simbol	Nama	Deskripsi
	<i>Status Awal</i>	Status awal aktifitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
	<i>Aktivitas</i>	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata Kerja
	<i>Percabangan</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
	<i>Status Akhir</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
	<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara Paralel



Simbol	Nama	Deskripsi
	<i>Join</i>	Digunakan untuk menunjukkan kegiatan yang Digabungkan

3. Sequence Diagram

Diagram ini adalah kombinasi *diagram class* dan *object* yang menunjukkan model statis, tetapi ada juga yang dinamis. Diagram ini menunjukkan proses operasi, pesan yang akan dikirim, dan kapan operasi dilakukan. Diagram yang diatur berdasarkan waktu ini memiliki objek yang terkait dengan proses berlangsungnya operasi, yang diurutkan dari kiri ke kanan.

Tabel 2. 4 Sequence Diagram

Simbol	Nama	Keterangan
	<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem
	<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
	<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem
	<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan table
	<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
	<i>A message</i>	Menggambarkan pengiriman pesan



4. Class Diagram

Diagram kelas (*class diagram*) adalah salah satu jenis diagram struktur dalam Unified Modeling Language (UML) yang menggambarkan struktur statis dari suatu sistem. Diagram ini menampilkan kelas-kelas dalam sistem beserta atribut, metode, dan hubungan antar kelas tersebut. Sebagai representasi statis, diagram kelas memodelkan aspek-aspek sistem yang tidak berubah seiring waktu. Menurut penelitian yang dipublikasikan dalam *Jurnal Ilmu Komputer dan Bisnis*, diagram kelas digunakan untuk memodelkan kelas-kelas dalam sistem, termasuk atribut dan metode mereka, serta hubungan antar kelas.

Penerapan diagram kelas dalam perancangan sistem informasi perpustakaan, misalnya, membantu dalam pengembangan sistem informasi yang berkualitas. Selain itu, studi lain yang dipublikasikan dalam *Jurnal Informatika Terpadu* menyebutkan bahwa diagram kelas merupakan diagram yang digunakan untuk merepresentasikan kelas, komponen-komponen kelas, dan hubungan masing-masing kelas, serta mendeskripsikan jenis-jenis objek dalam sistem. Dengan demikian, diagram kelas memberikan representasi visual yang esensial dari struktur statis sistem, mencakup kelas-kelas yang terlibat, atribut dan metode mereka, serta hubungan di antara kelas-kelas tersebut. Penggunaan diagram kelas dalam pengembangan perangkat lunak berorientasi objek membantu dalam perencanaan, komunikasi antar tim, dan dokumentasi sistem secara efektif.

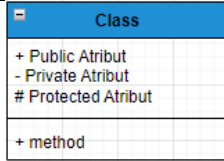
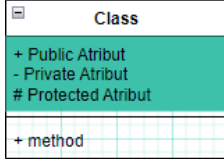
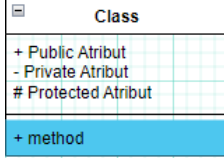
Diagram kelas ini melakukan banyak hal, tetapi fungsi utamanya adalah menunjukkan struktur sistem. Ada fungsi tambahan berikut:

- a. Menunjukkan struktur sistem dengan jelas;
- b. Meningkatkan pemahaman tentang gambaran umum atau skema program;
- c. Dapat digunakan untuk analisis bisnis dan membuat model sistem dari sisi bisnis; dan
- d. Memberikan gambaran tentang sistem atau perangkat lunak dan hubungannya.



Setiap komponen berada di bagian yang berbeda. Class (*upper section*), atribut (*middle section*), dan metode (*bottom section*). Selain itu, masing-masing komponen dijelaskan di sini. *class diagram*:

Tabel 2. 5 Class Diagram

Simbol	Nama	Keterangan
	<i>Class</i>	Nama dari class tersebut
	<i>Atribut</i>	Atribut berisikan karakteristik objek dan berfungsi menjelaskan kualitas kelas. Atribut dituliskan pada bagian atau kolom kedua, dan hanya digunakan untuk mendeksripsikan <i>instance</i> dari suatu kelas.
	<i>Metode</i>	Metode atau operasi (<i>operation</i>) dituliskan pada bagian atau kolom ketiga. Metode menggambarkan perilaku kelas, seperti bagaimana interaksi kelas dengan data dan apa yang bisa dilakukan oleh kelas.
	<i>Asosiasi</i>	Asosiasi dapat diartikan sebagai hubungan antara dua <i>class</i> yang bersifat statis. Biasanya asosiasi menjelaskan <i>class</i> yang memiliki atribut tambahan seperti class lain.



Simbol	Nama	Keterangan
	<i>Agregasi</i>	Agregasi adalah hubungan antara dua <i>class</i> di mana salah satu <i>class</i> merupakan bagian dari <i>class</i> lain, tetapi dua <i>class</i> ini dapat berdiri masing-masing.
	<i>Pewarisan</i>	Pewarisan atau <i>inheritance</i> dapat disebut juga <i>generalization</i> dalam <i>class diagram</i> adalah suatu kemampuan untuk mewarisi seluruh atribut dan metode dari <i>class</i> asalnya (<i>superclass</i>) ke <i>class</i> lain (<i>subclass</i>).

Selain diagram-diagram di atas masih ada beberapalainnya, yaitu :

- *Collaboration Diagram*,
- *Object Diagram*,
- *Component Diagram*, dan
- *Deployment Diagram*. (Guntoro, 2023)

1.4 HTML (*HyperText Markup Language*)

HTML (HyperText Markup Language) adalah bahasa markup standar yang digunakan untuk merancang struktur dan tampilan halaman web. Dikembangkan oleh Tim Berners-Lee pada tahun 1991, *HTML* telah menjadi elemen penting dalam dunia pengembangan web modern. Bahasa ini berfungsi untuk menyusun teks, gambar, video, dan elemen lainnya sehingga dapat ditampilkan dengan baik oleh browser. Meski sering disalahartikan sebagai bahasa pemrograman, *HTML* sebenarnya adalah bahasa markup yang mengatur bagaimana konten disusun dan ditampilkan. *HTML* terdiri dari elemen-elemen berbasis tag, seperti `<p>` untuk membuat paragraf dan `<a>` untuk tautan. Setiap elemen memiliki tag



pembuka dan penutup yang memberikan instruksi kepada browser tentang cara menampilkan konten tersebut. Selain itu, *HTML* mendukung integrasi dengan teknologi web lainnya, seperti *CSS (Cascading Style Sheets)* untuk memberikan gaya visual dan *JavaScript* untuk menambahkan interaktivitas, sehingga menghasilkan halaman web yang dinamis dan menarik.

Keunggulan utama *HTML* terletak pada kemudahannya untuk dipelajari dan digunakan. Dengan struktur yang sederhana, *HTML* sangat ramah bagi pemula yang baru memulai perjalanan di dunia pengembangan web. Bahasa ini memungkinkan pengembang untuk menentukan hierarki dan struktur konten dengan mudah, seperti pengaturan judul, paragraf, tabel, atau daftar. Selain itu, *HTML* diakui secara universal dan kompatibel dengan semua browser modern, memastikan bahwa halaman web dapat diakses dengan baik di berbagai perangkat dan platform. Sebagai teknologi open-source, *HTML* tidak memerlukan biaya lisensi dan dapat digunakan oleh siapa saja. Versi terbaru, *HTML5*, membawa banyak pembaruan penting, termasuk dukungan bawaan untuk video, audio, dan grafik tanpa memerlukan plugin tambahan seperti Flash. Kemampuan ini memungkinkan pengembang web untuk menciptakan pengalaman multimedia yang kaya secara lebih efisien. *HTML5* juga memperkenalkan elemen-elemen baru yang mendukung pengembangan aplikasi web dengan fitur-fitur modern. Selain itu, *HTML* sangat berperan dalam optimasi mesin pencari (*SEO*). Dengan memanfaatkan struktur heading dan tag meta yang tepat, *HTML* membantu meningkatkan visibilitas konten di mesin pencari seperti Google. *HTML* juga mendukung aksesibilitas, memungkinkan situs web yang dirancang untuk pengguna dengan disabilitas melalui penggunaan tag dan atribut khusus yang kompatibel dengan pembaca layar. Dengan berbagai keunggulannya, *HTML* menjadi tulang punggung pengembangan web. Kombinasinya dengan teknologi pendukung seperti *CSS* dan *JavaScript* membuka peluang untuk menciptakan situs web yang responsif, interaktif, dan modern, memenuhi kebutuhan pengguna di era digital.



1.1 PHP (Hypertext Preprocessor)

PHP (*Hypertext Preprocessor*), yang sebelumnya dikenal sebagai *Personal Home Pages*, merupakan bahasa pemrograman yang banyak digunakan dalam pengembangan web. Bahasa ini pertama kali dikembangkan oleh Rasmus Lerdorf, seorang programmer asal Denmark-Kanada, pada tahun 1994. Saat ini, PHP dikelola oleh *PHP Group*, yang bertanggung jawab atas pengembangan dan implementasi bahasa ini. Meskipun awalnya PHP adalah singkatan dari *Personal Home Page*, kini istilah tersebut menjadi akronim rekursif dari *PHP: Hypertext Preprocessor* (Labdajiwa, 2021). PHP dirancang untuk bekerja secara optimal di sisi server. Sebagian besar kode PHP dieksekusi oleh interpreter PHP di server web, yang dapat berjalan sebagai *daemon*, modul, atau Common Gateway Interface (*CGI*). Hasil dari eksekusi kode *PHP* biasanya berupa data dalam berbagai format, seperti gambar biner atau dokumen *HTML*, yang digunakan untuk menghasilkan respons *HTTP*. Pengembang juga dapat memanfaatkan *framework web*, *sistem manajemen konten web*, atau *sistem template web* untuk menyederhanakan pembuatan respons ini. Selain pengembangan web, PHP memiliki kemampuan luas, seperti pembuatan aplikasi grafis mandiri, kontrol robotik, hingga eksekusi langsung melalui *command line interface* (*CLI*). Popularitas *PHP* di kalangan programmer didukung oleh berbagai keunggulan. Bahasa ini terkenal mudah dipelajari karena sintaksnya yang sederhana dan intuitif, membuatnya ideal bahkan bagi pemula. Kemampuannya untuk bekerja secara harmonis dengan *HTML* memungkinkan pengembang mengintegrasikan kedua teknologi ini tanpa konflik. Sebagai bahasa pemrograman *open-source*, *PHP* tersedia secara gratis dan dapat diakses oleh siapa saja, memberikan fleksibilitas tinggi dalam pengembangan.

Fleksibilitas *PHP* juga terlihat dari kemampuannya untuk diintegrasikan dengan berbagai bahasa pemrograman lainnya. Hal ini memudahkan pengembang dalam menyesuaikan sistem sesuai kebutuhan, termasuk untuk proyek yang lebih kompleks. Selain itu, meskipun telah digunakan selama hampir tiga dekade, *PHP* tetap mendukung



layanan *cloud* secara efisien, menjadikannya relevan dalam pengembangan teknologi modern. Dukungan yang luas dari komunitas global menjadi salah satu kekuatan utama PHP. Tersedia banyak dokumentasi, forum, dan panduan yang dapat membantu pengembang, baik pemula maupun profesional, dalam memahami dan menyelesaikan berbagai tantangan terkait penggunaan *PHP*. Bahasa ini juga mendukung berbagai jenis basis data, mulai dari *MySQL* hingga basis data non-relasional, memberikan fleksibilitas dalam manajemen data. Kecepatan adalah salah satu keunggulan utama *PHP*. Dibandingkan dengan beberapa bahasa pemrograman lainnya, *PHP* mampu memberikan waktu respons yang lebih cepat. Selain itu, *PHP* adalah *multi-platform*, sehingga dapat berjalan di berbagai sistem operasi, termasuk Linux, Windows, dan *MacOS*. Meskipun telah lama digunakan, *PHP* terus diperbarui secara berkala untuk memenuhi kebutuhan pengembangan modern, dengan versi terbarunya saat ini adalah *PHP 8.2.12*. Dengan kemampuan yang luas, fleksibilitas tinggi, dan dukungan komunitas yang solid, *PHP* tetap menjadi pilihan utama untuk pengembangan web dan berbagai aplikasi lainnya. Sebagai bahasa yang terus berkembang, *PHP* memberikan solusi yang andal dan relevan dalam menghadapi tantangan teknologi masa kini.

1.2 CSS (Cascading Style Sheets)

CSS (Cascading Style Sheets) adalah bahasa *stylesheet* yang digunakan untuk mengatur tampilan dan format elemen-elemen *HTML* pada halaman web. Bahasa ini pertama kali dikembangkan oleh Håkon Wium Lie dan secara resmi direkomendasikan oleh *W3C (World Wide Web Consortium)* pada tahun 1996. Dengan *CSS*, pengembang web dapat mengontrol berbagai aspek visual, seperti tata letak, warna, font, dan elemen desain lainnya secara konsisten di seluruh halaman situs web (Wardani & Sukma, 2021). Keunggulan utama *CSS* adalah kemampuannya untuk memisahkan konten dan presentasi dalam sebuah halaman web. Jika *HTML* berperan menyusun struktur dan konten, maka *CSS* bertanggung jawab mengatur penampilan visualnya. Dengan pemisahan ini, pengembang dapat membuat perubahan pada desain tanpa perlu mengubah kode



HTML secara langsung, sehingga meningkatkan efisiensi dan konsistensi dalam pengembangan web (Saputra, 2020). *CSS* bekerja melalui penggunaan selektor, yaitu instruksi yang mengidentifikasi elemen-elemen *HTML* tertentu untuk diterapkan aturan gaya tertentu. Misalnya, selektor *h1* digunakan untuk mengatur tampilan elemen *heading* tingkat satu, sementara selektor *.class-name* digunakan untuk elemen dengan kelas tertentu. *CSS* juga mendukung konsep hierarki atau *cascading*, yang memungkinkan penerapan aturan gaya berdasarkan urutan tertentu dan tingkat prioritas yang ditentukan (Nurhayati & Febrianto, 2020).

Sebagai komponen penting dalam pengembangan web modern, *CSS* menawarkan sejumlah keunggulan yang membuatnya tak tergantikan. Salah satu manfaatnya adalah kemampuan untuk memisahkan konten *HTML* dari desain visual. Hal ini memungkinkan perubahan tampilan dilakukan dengan hanya memodifikasi file *CSS*, tanpa memengaruhi struktur konten. Keunggulan lain adalah konsistensi desain yang dapat diterapkan di seluruh halaman situs, sehingga menghemat waktu dan upaya pengembangan. Dengan *CSS*, kode *HTML* menjadi lebih bersih dan efisien karena elemen gaya tidak perlu diulang dalam setiap elemen *HTML*. Kemampuan responsivitas yang dimiliki *CSS* juga menjadi fitur unggulan. Dengan menggunakan *media queries*, tata letak halaman dapat menyesuaikan diri dengan berbagai ukuran perangkat, mulai dari ponsel hingga desktop. Kustomisasi desain juga menjadi lebih mudah berkat struktur fleksibel *CSS*. Pengembang dapat mengubah tampilan situs dengan hanya memodifikasi file *CSS*, tanpa harus menyentuh setiap halaman *HTML*. Selain itu, *CSS* memiliki kompatibilitas yang luas dan didukung oleh semua browser modern, memastikan tampilan konsisten di berbagai perangkat dan platform. Dari sisi kinerja, *CSS* juga memberikan manfaat signifikan. Dengan memisahkan file *CSS* dari *HTML*, beban jaringan menjadi lebih ringan karena file *CSS* dapat disimpan dalam *cache* browser. Hal ini meningkatkan kecepatan pemuatan halaman web, yang menjadi aspek penting dalam memberikan pengalaman pengguna yang optimal. Sebagai standar industri, *CSS* sering digunakan bersama *HTML* dan



JavaScript untuk menciptakan situs web yang interaktif dan responsif (Yunita & Sari, 2020).

Sebagai fondasi desain visual dalam pengembangan web, CSS tidak hanya menyediakan alat untuk menciptakan estetika yang menarik, tetapi juga mendukung efisiensi, kompatibilitas, dan kinerja yang optimal. Dengan kombinasi fitur dan fleksibilitasnya, CSS tetap menjadi teknologi penting yang mendukung inovasi dalam dunia pengembangan web modern.

1.3 Database

Database adalah kumpulan file atau data yang disusun secara terstruktur dan disimpan bersama-sama untuk dapat diakses, dikelola, dan digunakan oleh berbagai aplikasi. Database berfungsi untuk menghindari terjadinya duplikasi data serta memfasilitasi pengelolaan data yang lebih efisien. Konsep ini memungkinkan berbagai program aplikasi untuk menggunakan kumpulan data yang sama tanpa memerlukan pengulangan data, sehingga menjaga integritas dan konsistensi data dalam sistem. Hal ini disebabkan oleh pemisahan antara data yang disimpan dalam database dan aplikasi yang mengaksesnya, membuat data lebih terorganisir dan dapat diakses secara lebih efisien (*Sabbrina, Sufa, Ritonga, & Siregar, 2023*).

Penggunaan database memiliki berbagai keunggulan, salah satunya adalah kemampuannya untuk mendukung pengelolaan data dalam skala besar dan berbagai jenis aplikasi. Dengan menggunakan sistem manajemen basis data (*Database Management System* atau *DBMS*), data dapat dikelola secara lebih terpusat dan aman. *DBMS* memungkinkan pengguna untuk menyimpan, mengedit, mengambil, serta memelihara data dengan cara yang terstruktur, sehingga data dapat dengan mudah diproses dan diakses oleh berbagai aplikasi tanpa menimbulkan masalah seperti ketidakkonsistenan atau duplikasi. Selain itu, *DBMS* juga menyediakan fitur keamanan seperti kontrol akses dan



manajemen hak pengguna untuk menjaga data dari akses yang tidak sah (*Sabbrina et al., 2023*).

a) **MySQL**

MySQL merupakan salah satu sistem manajemen basis data relasional (*Relational Database Management System* atau *RDBMS*) yang sangat populer dan digunakan secara luas untuk berbagai keperluan pengelolaan data. MySQL memungkinkan pengguna untuk menyimpan, mengelola, dan mengambil data menggunakan bahasa *SQL* (*Structured Query Language*), yang menjadi standar dalam interaksi dengan basis data relasional. Sebagai perangkat lunak *open-source*, MySQL tidak hanya gratis untuk digunakan, tetapi juga didukung oleh komunitas besar yang terus mengembangkan dan meningkatkan fungsionalitasnya.

MySQL dikenal dengan kecepatan dan efisiensinya dalam menangani volume data besar, sehingga menjadi pilihan utama bagi banyak aplikasi *web* termasuk situs dinamis seperti *blog*, *e-commerce*, dan platform sosial media (*Nurhayati & Nasution, 2023*). Selain itu, MySQL mendukung berbagai jenis operasi database yang diperlukan dalam pengembangan aplikasi, mulai dari penyimpanan dan pemrosesan data hingga pembuatan laporan dan analisis. Salah satu keunggulan MySQL adalah kemampuannya untuk bekerja dengan berbagai sistem operasi, seperti *Linux*, *Windows*, dan *macOS*. Dengan dukungan *multi-user* dan *multi-threading*, MySQL memungkinkan beberapa pengguna atau proses aplikasi untuk mengakses dan mengelola data secara bersamaan tanpa mengorbankan kinerja atau stabilitas sistem. Kelebihan lainnya termasuk dukungan terhadap berbagai tipe data, kemampuan untuk menangani transaksi, serta fleksibilitas dalam hal keamanan dan perizinan akses (*Sabbrina et al., 2023*).

1.4 Profil Puskesmas Jogoloyo

Puskesmas Jogoloyo, berlokasi di Jl. Raya Jogoloyo, Kecamatan Sumobito, Kabupaten Jombang, merupakan fasilitas kesehatan tingkat pertama di bawah Kementerian



Kesehatan Republik Indonesia. Puskesmas ini bertujuan untuk meningkatkan kualitas kesehatan masyarakat melalui berbagai layanan, seperti imunisasi, pengobatan, pemeriksaan kesehatan, dan penyuluhan. Setiap harinya, Puskesmas ini didukung oleh 46 tenaga kesehatan yang terdiri dari dokter, perawat, dan staf pendukung lainnya. Tantangan Pengelolaan Keuangan.