
BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Santosa *et al.* (2020:2), “komputer adalah alat elektronik yang dapat menerima masukan (*input*) berupa data dan instruksi, memproses data tersebut sesuai dengan program yang diberikan, dan menghasilkan keluaran (*output*) berupa informasi”.

Menurut Sihombing (2022:3), mendefinisikan bahwa “komputer adalah mesin elektronik yang mendefinisikan operasi aritmatika dan logika, menyimpan data dalam jumlah besar, dan memproses data dengan cepat”.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa komputer adalah alat yang sangat serbaguna yang dapat digunakan untuk berbagai macam tugas, mulai dari yang sederhana seperti mengetik dokumen hingga yang kompleks seperti merancang pesawat terbang.

2.1.2 Pengertian Perangkat Lunak

Menurut Suryadinata (2021:12), perangkat lunak didefinisikan sebagai kumpulan instruksi yang ditulis dalam bahasa yang dapat dimengerti oleh komputer. Perangkat lunak ini menentukan apa yang dapat dilakukan komputer dan bagaimana cara kerjanya.

Menurut Nugroho *et al.* (2022:115), perangkat lunak adalah program komputer yang berisi instruksi-instruksi yang ditulis dalam bahasa pemrograman tertentu. Program-program ini disimpan dalam memori komputer dan dijalankan oleh prosesor untuk menyelesaikan tugas tertentu.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa perangkat lunak adalah "otak" komputer, sedangkan perangkat keras adalah "tubuh" komputer.

2.1.3 Pengertian Data

Menurut Mardiah & Syahroni (2020:10) Data merupakan kumpulan nilai atau fakta yang belum diolah dan tidak memiliki makna tertentu. Data dapat berupa angka, teks, gambar, suara, atau video.

Menurut Setyowati (2020:4) Data adalah kumpulan fakta atau nilai yang dikumpulkan dan diolah untuk menghasilkan informasi yang bermanfaat bagi pengambilan keputusan.

Berdasarkan beberapa definisi di atas dapat disimpulkan bahwa data adalah kumpulan fakta atau nilai yang belum atau sudah diolah untuk menghasilkan informasi yang bermanfaat.

2.1.4 Pengertian Basis Data

Basis Data menurut Putri (2019:11) adalah suatu sistem, yang mengorganisasikan, menyimpan, dan mengelola data secara sistematis dan terstruktur dengan baik. Basis Data memungkinkan pengguna untuk mengakses, memperbarui, dan menghapus data dengan mudah dan efisien.

Menurut Raharjo (2021:6), basis data adalah sistem yang mengorganisasikan, menyimpan, dan mengelola data secara fleksibel dan terstruktur. Sistem ini memungkinkan pengguna untuk mengakses, memperbarui, dan menghapus data dengan mudah dan aman.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa basis data merupakan sistem yang membantu pengguna untuk mengelola data secara efektif dan efisien.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Pratama, *et al.* (2022:115) menjelaskan bahwa aplikasi merupakan program komputer yang dirancang untuk menjalankan tugas atau fungsi tertentu guna membantu pengguna menyelesaikan pekerjaan atau mencapai tujuan tertentu.

Menurut Sutanta, *et al.* (2019:16) menjelaskan bahwa aplikasi adalah perangkat lunak yang dibuat untuk memenuhi kebutuhan dan menyelesaikan masalah tertentu bagi penggunanya. Aplikasi dapat diklasifikasikan menjadi beberapa jenis, seperti aplikasi sistem, aplikasi bisnis, aplikasi pendidikan, aplikasi hiburan, dan lain sebagainya.

Irawan, *et al.* (2020:124) menjelaskan bahwa aplikasi merupakan program komputer yang dirancang untuk menyediakan layanan atau fungsi tertentu bagi penggunanya. Aplikasi dapat dijalankan pada berbagai gawai, seperti komputer pribadi, ponsel pintar, tablet, dan perangkat elektronik lainnya.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa aplikasi adalah suatu bagian dari perangkat lunak yang dapat melakukan proses-proses yang digunakan dalam komputer oleh pengguna untuk melakukan koneksi ke suatu *database* sepanjang aplikasi itu dijalankan.

2.2.2 Pengertian E-Faktur

Menurut Yuliansyah, *et al.* (2019:12), e-faktur adalah faktur elektronik yang diterbitkan dan disimpan dalam bentuk elektronik. E-faktur dibuat dan ditandatangani secara elektronik oleh penjual dan pembeli, serta disimpan dalam sistem elektronik yang disediakan oleh Direktorat Jenderal Pajak (DJP).

Handayani, *et al.* (2021:117) menjelaskan bahwa e-faktur adalah faktur elektronik yang digunakan untuk menggantikan faktur kertas. E-faktur memiliki beberapa keunggulan dibandingkan faktur kertas, seperti lebih efisien, lebih aman, dan lebih mudah disimpan.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa e-faktur adalah faktur elektronik yang menggantikan faktur kertas dan menawarkan beberapa keunggulan, seperti lebih efisien, lebih aman, dan lebih mudah disimpan

2.2.3 Pengertian Pajak

Menurut Mustofa, *et al.* (2019:4), pajak adalah iuran wajib dari rakyat kepada negara untuk digunakan untuk keperluan negara.

Menurut Hamzah, et al. (2022:114), pajak merupakan kewajiban bagi setiap orang pribadi atau badan usaha untuk membayar kepada negara sejumlah uang tertentu dengan cara yang ditentukan dalam undang-undang, sebagai kontribusi atas kesanggupan untuk menikmati “jasa negara”.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa pajak adalah kewajiban wajib bagi rakyat dan badan usaha untuk membayar kepada negara sejumlah uang tertentu.

2.2.4 Pengertian Perusahaan

Menurut Simamora, et al. (2019:3) menjelaskan bahwa “perusahaan adalah organisasi yang didirikan dan dioperasikan untuk menghasilkan keuntungan atau mencapai tujuan tertentu.” Perusahaan dapat berbentuk badan usaha perorangan, persekutuan, atau perseroan.

Setiawan, et al. (2021:83) menjelaskan bahwa “perusahaan merupakan organisasi yang didirikan dan dioperasikan untuk mencapai tujuan bersama dengan menggunakan sumber daya yang dimiliki. Perusahaan dapat berbentuk badan usaha milik negara (BUMN), badan usaha milik daerah (BUMD), atau perusahaan swasta”.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa Perusahaan adalah organisasi yang didirikan dan dioperasikan untuk mencapai tujuan, baik itu keuntungan maupun bersama, dengan menggunakan sumber daya yang dimiliki. Bentuk badan usaha perusahaan beragam, termasuk perorangan, persekutuan, perseroan, BUMN, BUMD, dan swasta.

2.2.5 Pengertian Website

Menurut Setiawan (2022:1), “*website* adalah kumpulan halaman web yang saling terhubung dan dapat diakses melalui internet. *Website* dapat berisi berbagai informasi, seperti teks, gambar, video, dan audio.”

Menurut Fauzi et al. (2022:118), “*website* adalah kumpulan halaman web yang terhubung dan dapat diakses melalui internet.”

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *website* adalah kumpulan halaman web yang terhubung dan dapat diakses melalui internet.

2.2.6 Pengertian Aplikasi E- Faktur Pajak Perusahaan Berbasis *Website* Pada PT Sayovi Karyatama

Aplikasi yang dibuat untuk membantu dalam mengelola dan melaporkan faktur pajak secara elektronik kepada Direktorat Jenderal Pajak (DJP). Aplikasi ini berbasis *website*, sehingga dapat diakses melalui internet dari mana saja dan kapan saja.

2.3 Teori Khusus

2.3.1 Pengertian Kamus Data

Menurut Setyowati (2020:1) mengatakan bahwa “Kamus data adalah suatu kumpulan informasi yang terorganisir tentang elemen data dalam suatu sistem informasi. Kamus data mendeskripsikan elemen data, seperti nama, tipe data, panjang data, format data, nilai valid, aturan bisnis, dan hubungan antar elemen data.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa kamus data adalah kumpulan daftar elemen data yang tertulis secara lengkap dari data yang diisikan ke dalam database sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum.

Tabel 2.1 Simbol-simbol pada kamus data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau...
{ }n	n kali diulang/bernilai banyak
()	Data opsional
...	Batas komentar

Sumber : Jurnal Ilmiah Komputer dan Informatika (dalam Susanti et al,2020: 4)

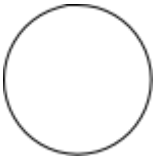
2.3.2 Pengertian DFD (*Data Flow Diagram*)

Menurut Siau, *et al.* (2022: 90), *Data Flow Diagram (DFD)* atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

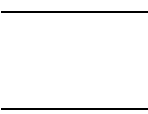


Menurut Sari & Brata (2022:4), "*Data Flow Diagram (DFD)* adalah alat bantu visual yang digunakan untuk menggambarkan aliran data dalam suatu sistem informasi. DFD menunjukkan bagaimana data diubah dan dialirkan dari satu proses ke proses lain dalam sistem".

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Data Flow Diagram (DFD)* adalah alat yang penting dalam pengembangan sistem informasi. Dengan memahami DFD, kita dapat lebih mudah memahami cara kerja sistem informasi, mengidentifikasi potensi masalah, dan merancang sistem baru yang lebih baik.

Tabel 2.2 Simbol-simbol *Data Flow Diagram (DFD)*

Notasi	Keterangan
	<p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.</p> <p>Catatan:</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>

Lanjutan Tabel 2.2 Simbol-simbol *Data Flow Diagram (DFD)*

	File atau basis data atau penyimpanan(<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perencanaan tabel-tabel pada basis data (<i>Entity Relationship Diagram (ERD)</i> , <i>Conceptual Data Model (CDM)</i> , <i>Physical Data Model (PDM)</i>). Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.
	Entitas luar (<i>External entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau <i>system</i> lain yang terkait dengan aliran data dari <i>system</i> yang dimodelkan. Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.
	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.

2.3.3 Pengertian *Blockchart*

Menurut Fitriana *et al.* (2020:44) "*Blockchart* adalah diagram alir yang menggambarkan alur proses secara umum, tanpa menunjukkan detail langkah-langkahnya."

Menurut Br Peranginangin *et al.* (2020:62) "*Blockchart* adalah diagram alir yang menunjukkan alur proses secara umum, tanpa menunjukkan detail kondisi atau percabangan."

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Blockchart* adalah diagram alir yang menggambarkan alur proses secara umum, tanpa menunjukkan detail langkah-langkah atau kondisi percabangan.

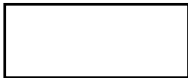
2.3.4 Pengertian *Entity Relationship Diagram (ERD)*

Menurut Sutarman & Kristiawan (2021:130), *Entity Relationship Diagram* (ERD) adalah diagram yang menggambarkan entitas (tabel) dalam suatu sistem informasi, atributnya (kolom), dan hubungan antar entitas. ERD bermanfaat untuk memahami struktur data dalam sistem, mengidentifikasi potensi redundansi data, dan merancang basis data yang efisien.


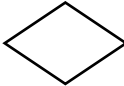





Menurut Sari & Brata (2022:6), *Entity Relationship Diagram* (ERD) adalah model visual yang menunjukkan entitas (tabel) dalam suatu sistem informasi, atributnya (kolom), dan hubungan antar entitas. ERD bermanfaat untuk mengkomunikasikan struktur data sistem kepada pemangku kepentingan dan merancang basis data yang efektif.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Entity Relationship Diagram* adalah diagram yang menggambarkan entitas (tabel) dalam suatu sistem informasi, atributnya (kolom), dan hubungan antar entitas.

Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

No	Simbol-Simbol	Keterangan
1.		<i>Entity</i>

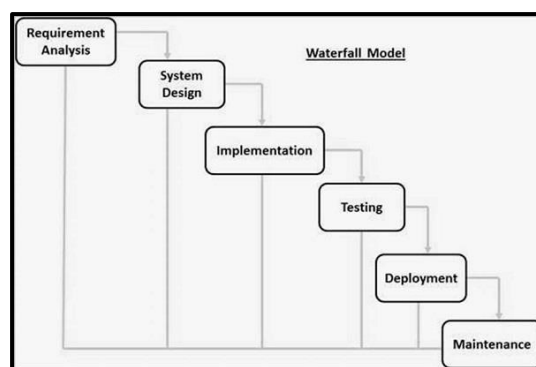
Lanjutan Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

2.		<i>Weak Entity</i>
3.		<i>Relationship</i>
4.		<i>Identifying Relationship</i>
5.		<i>Attribute</i>
6.		<i>Key Attribute</i>
7.		<i>Multivalued Attribute</i>
8.		<i>Derivative Attribute</i>

Sumber : Nugroho (dalam Agarina et al, 2019: 53)

2.3.5 Metode Pengembangan Sistem

Menurut Putri & Hariadi (2021:12), Model *Waterfall* adalah model pengembangan perangkat lunak yang bersifat sekuensial dan linier. Model ini terdiri dari beberapa tahap yang harus dilalui secara berurutan, dan setiap tahap tidak dapat kembali ke tahap sebelumnya. Berikut adalah gambar model air terjun:

**Gambar 2.1** Ilustrasi model *waterfall*

Menurut Sommerville (2021:152), metode *Waterfall* memiliki beberapa tahapan sebagai berikut:

- 1) Persyaratan dan Analisis: Tahap ini bertujuan untuk memahami kebutuhan pengguna dan sistem, serta menganalisis kelayakan proyek.
- 2) Desain: Tahap ini bertujuan untuk merancang sistem secara detail, termasuk arsitektur sistem, database, dan antarmuka pengguna.
- 3) Implementasi: Tahap ini bertujuan untuk menerjemahkan desain ke dalam kode program.
- 4) Pengujian: Tahap ini bertujuan untuk menguji sistem untuk memastikan bahwa sistem berfungsi dengan baik dan sesuai dengan kebutuhan.
- 5) Penerapan: Tahap ini bertujuan untuk menerapkan sistem ke pengguna dan melatih mereka untuk menggunakan sistem.
- 6) Pemeliharaan: Tahap ini bertujuan untuk memelihara sistem setelah sistem diimplementasikan, termasuk memperbaiki bug dan menambahkan fitur baru.

2.3.6 Pengertian *Flowchart*



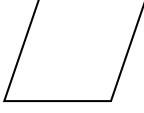
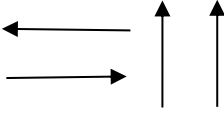
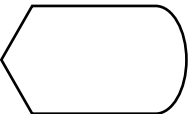

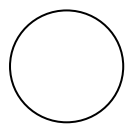

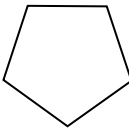
Menurut Sofwan (2020:228), *flowchart* adalah diagram alir yang menggambarkan urutan langkah dan keputusan dalam suatu proses atau sistem secara visual dengan menggunakan simbol-simbol standar.

Menurut Indrajani (2021:102), *flowchart* merupakan representasi grafis dari suatu proses yang menggambarkan urutan langkah-langkah dan keputusan yang terlibat dalam proses tersebut, dengan menggunakan simbol-simbol standar yang terhubung oleh garis atau panah.



Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *flowchart* adalah alat yang bermanfaat untuk memvisualisasikan, memahami, dan menganalisis suatu proses. *Flowchart* dapat digunakan dalam berbagai bidang, seperti bisnis, ilmu komputer, dan teknik.

Simbol-simbol baku dari bagan arus data menurut Iswandy (dalam Indrianto *et al*, 2019: 102) dapat dilihat pada tabel berikut:

Tabel 2.4 Simbol-simbol *Flowchart*

No	Simbol	Nama	Keterangan
1.		Terminator	Sebagai awal dan akhir program.
2.		Proses	Proses pengolahan data.
3.		<i>Input/output</i> data	Proses input dan output data, parameter dan informasi.
4.		Garis Alur	Merupakan arah aliran dari sebuah program.
5.		<i>Display</i>	Menyatakan peralatan output yang digunakan.
5.		<i>Display</i>	Menyatakan peralatan output yang digunakan.
6.		<i>On Page Conector</i>	Menghubungkan bagian <i>flowchart</i> yang terdapat pada satu halaman.
7.		<i>Predefined Process</i> (sub program)	Permulaan sub program dan proses permulaan penjalanan sub program.
8.		<i>Off Page Connector</i>	Menghubungkan bagian <i>flowchart</i> yang berada pada halaman berbeda.

Lanjutan Tabel 2.4 *Flowchart*

9.		Simbol Dokumen	Untuk mencetak keluaran dokumen berupa kertas.
10.		Storage Data	Menunjukkan input / output menggunakan disket.

Sumber : Iswandy (dalam Indrianto et al, 2019: 102)

2.3.7 Pengertian *Black Box Testing*

Menurut Anggraini & Sari (2020:123), *Black box testing* adalah metode pengujian perangkat lunak yang berfokus pada fungsionalitas eksternal dari perangkat lunak tanpa mempertimbangkan struktur internal atau kode programnya. Pengujian ini dilakukan dengan memberikan input kepada perangkat lunak dan mengamati *outputnya* untuk memastikan bahwa perangkat lunak berperilaku sesuai dengan spesifikasi yang telah ditentukan.

Menurut Putri & Sari (2021:78), *Black box testing*, juga dikenal sebagai pengujian fungsional, adalah metode pengujian perangkat lunak yang memvalidasi fungsionalitas eksternal suatu perangkat lunak tanpa mempertimbangkan struktur internal atau kode programnya. Pengujian ini dilakukan dengan memberikan *input* kepada perangkat lunak dan mengamati *outputnya* untuk memastikan bahwa perangkat lunak berperilaku sesuai dengan spesifikasi yang telah ditentukan.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Black box testing* adalah metode pengujian perangkat lunak yang penting untuk memastikan fungsionalitas dan kualitas perangkat lunak.

2.3.8 Pengertian *User Acceptance Testing* (UAT)

Menurut Handayani & Maulidina (2020:256), UAT (*User Acceptance Testing*) adalah proses pengujian perangkat lunak yang dilakukan oleh pengguna akhir untuk memastikan bahwa perangkat lunak tersebut memenuhi kebutuhan dan ekspektasi mereka.

Menurut Rahmawati & Yuwono (2022:118), UAT (*User Acceptance Testing*) adalah proses pengujian perangkat lunak yang dilakukan oleh pengguna akhir untuk memastikan bahwa perangkat lunak tersebut memenuhi kebutuhan dan ekspektasi mereka. Pengujian ini dilakukan dengan memberikan pengguna akhir akses ke perangkat lunak dan meminta mereka untuk melakukan tugas-tugas tertentu.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *User acceptance testing* adalah tahap penting dalam pengembangan perangkat lunak untuk memastikan bahwa perangkat lunak yang dihasilkan benar-benar memenuhi kebutuhan dan ekspektasi pengguna akhir.

2.3.9 Pengertian *RESTful API*

Menurut Ramadhani & Amin (2022:23), *RESTful API (Representational State Transfer Application Programming Interface)* adalah arsitektur gaya pemrograman antarmuka yang mengikuti prinsip-prinsip *REST (Representational State Transfer)*, yaitu:

- 1) *Client-Server* : Klien dan *server* adalah entitas yang terpisah dan berkomunikasi melalui pesan yang tidak berstatus.
- 2) *Stateless* : *Server* tidak menyimpan status percakapan antara klien dan *server*.
- 3) *Cacheable* : Respon dari *server* dapat dicache oleh klien.
- 4) *Uniform Interface* : Antarmuka yang digunakan untuk mengakses sumber daya bersifat seragam dan konsisten.
- 5) *Layered System* : Arsitektur *RESTful API* tersusun dari beberapa lapisan, di mana setiap lapisan memiliki tanggung jawab yang berbeda.
- 6) *Man-Readable* : Respon dari *server* mudah dipahami oleh manusia.
- 7) *Code on Demand* : *Server* dapat mengirimkan kode kepada klien untuk dieksekusi.

Menurut Putri & Sari (2021:32), *RESTful API (Representational State Transfer Application Programming Interface)* adalah arsitektur gaya pemrograman

antarmuka aplikasi yang mengikuti prinsip-prinsip *REST* (*Representational State Transfer*). *RESTful API* memungkinkan aplikasi untuk bertukar data dengan cara yang terdistribusi, *stateless*, dan *cacheable*.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Restful API* adalah arsitektur gaya pemrograman aplikasi web yang didasarkan pada prinsip-prinsip *REST*. Prinsip-prinsip ini memungkinkan aplikasi web untuk berkomunikasi satu sama lain dengan cara yang terstandarisasi dan terstruktur.

2.4 Teori Program

2.4.1 Pengertian Visual Studio Code

Menurut Mishra & Sinha (2020:5), *Visual Studio Code* adalah editor kode sumber lintas platform yang ringan dan kuat yang dikembangkan oleh *Microsoft*. Ini adalah salah satu editor kode sumber paling populer yang digunakan oleh pengembang perangkat lunak di seluruh dunia.

Visual Studio Code adalah editor kode sumber *open-source* yang *powerful* dan *user-friendly*. *Visual Studio Code* menyediakan berbagai fitur yang lengkap untuk membantu programmer menulis, mengedit, dan *men-debug* kode dengan mudah dan efisien. (Susanti *et al.*, 2021:4)

Berdasarkan definisi diatas, dapat disimpulkan bahwa *Visual studio code* adalah editor kode sumber yang *powerful*, *user-friendly*, dan kaya fitur yang menjadikannya pilihan populer bagi pengembang perangkat lunak di seluruh dunia.

2.4.2 Pengertian HTML

Menurut Singh & Kumar (2020:4), HTML adalah bahasa *markup* standar untuk membuat halaman web. Ini adalah bahasa yang digunakan untuk mendefinisikan struktur dan konten halaman web. HTML terdiri dari *tag*, yang merupakan elemen yang digunakan untuk memformat teks, gambar, dan elemen lainnya di halaman web.

Menurut Jones & Smith (2022:3), HTML didefinisikan sebagai bahasa *markup* standar yang digunakan untuk membuat halaman web. Laporan ini

membahas sejarah HTML, dari awal mulanya hingga versi terbaru saat ini. Laporan ini juga membahas dampak HTML pada pengembangan web dan masa depan bahasa.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Hypertext Markup Language* (HTML) adalah bahasa fundamental yang penting untuk dipahami bagi siapa pun yang ingin membuat halaman web atau mempelajari pengembangan web.

2.4.3 Pengertian PHP

Wijaya (2023:14) menjelaskan bahwa PHP adalah bahasa pemrograman yang mudah dipelajari dan digunakan. PHP memiliki *sintaks* yang sederhana dan banyak sumber daya belajar yang tersedia *online*. Hal ini membuat PHP menjadi pilihan yang tepat bagi pemula yang ingin belajar bahasa pemrograman web.

Suryadinata (2021:15) mendefinisikan PHP yang besar dan aktif yang menyediakan banyak dukungan dan sumber daya *online*.

Berdasarkan beberapa definisi di atas, dapat bahwa *Hypertext Preprocessor* (PHP) adalah bahasa pemrograman yang tepat untuk membangun situs web dan aplikasi web dinamis karena kemudahan penggunaan, komunitas yang besar, sifat *open-source*, dan performanya yang tinggi.

2.4.4 Pengertian Tailwind CSS

Menurut Nugroho *et al.* (2022:142), *Tailwind CSS* adalah kerangka kerja utilitas CSS yang menyediakan kelas-kelas CSS yang telah ditulis sebelumnya untuk membangun antarmuka pengguna (UI) web.

Suryadinata (2021:16) mendefinisikan *Tailwind CSS* sebagai kerangka kerja CSS yang fleksibel dan dapat disesuaikan dengan kebutuhan pengembang. *Tailwind CSS* menyediakan berbagai kelas CSS yang dapat dikombinasikan untuk membuat tata letak yang kompleks.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Tailwind* CSS merupakan *framework* yang tepat untuk pengembang yang ingin membangun *UI* yang indah, responsif, dan mudah dipelihara dengan cepat dan mudah.

2.4.5 Pengertian *JavaScript*

Menurut Hartono, *et al.* (2019:10), *JavaScript* memungkinkan halaman web menjadi lebih dinamis dan interaktif. Hal ini membuat pengalaman pengguna menjadi lebih baik.

Wijaya (2023:16) menjelaskan bahwa *JavaScript* adalah salah satu bahasa pemrograman web paling populer saat ini. *JavaScript* digunakan oleh hampir semua situs web modern dan merupakan bahasa pemrograman yang penting.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Javascript* adalah bahasa pemrograman yang sangat penting untuk pengembangan web modern

2.4.6 Pengertian *MySQL*

MySQL merupakan salah satu sistem manajemen basis data relasional (RDBMS) yang populer dan banyak digunakan. *MySQL* bersifat open-source dan gratis, sehingga mudah diakses dan dimodifikasi. *MySQL* terkenal dengan kemudahan penggunaan, skalabilitas, dan performanya yang baik. (Fitriani, 2020:36)

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang bersifat *open-source* dan gratis. *MySQL* banyak digunakan untuk membangun aplikasi web karena kemudahan penggunaan, skalabilitas, dan performanya yang baik. (Handayani, *et al.*, 2019:118)

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa *MYSQL* adalah sistem manajemen basis data relasional (RDBMS) yang populer dan banyak digunakan karena beberapa keunggulannya, antara lain: *open source*, mudah digunakan, skalabilitas, performa, cocok untuk aplikasi web.

2.4.7 Pengertian *Laravel*

"*Laravel* merupakan framework PHP *open-source* yang *powerful* dan *user-friendly*. *Laravel* terkenal dengan arsitekturnya yang modular, *sintaksnya* yang elegan, dan banyaknya fitur yang mendukung pengembangan aplikasi web modern." (Sari, 2019:37)

"*Laravel* adalah framework PHP *open-source* yang populer digunakan untuk membangun aplikasi web. *Laravel* terkenal dengan arsitekturnya yang modular, *sintaksnya* yang elegan, dan banyaknya fitur yang mendukung pengembangan aplikasi web modern." (Raharjo, et al., 2020:123)

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa *Laravel* adalah sebuah *framework PHP open-source* yang populer dan banyak digunakan untuk membangun aplikasi web modern karena beberapa keunggulannya.

2.4.8 Pengertian *Laragon*

Menurut Sitorus, et al. (2024:1) *Laragon* adalah aplikasi untuk membuat program atau aplikasi di server *local* (komputer). Salah satu kelebihan *laragon* adalah fitur yang lebih lengkap dibanding aplikasi lain dan kemudahan penggunaannya, karena menggunakan tampilan *GUI*.

Menurut Shahib, et al. (2024:382) *Laragon* adalah platform pengembangan web yang dirancang untuk mempermudah pengembangan dan pengujian aplikasi web. Ini dirancang untuk menyediakan lingkungan pengembangan lokal, atau *localhost*, yang dapat diinstal pada sistem operasi *Windows*.

Berdasarkan beberapa definisi di atas, maka dapat disimpulkan bahwa *Laragon* dilengkapi dengan berbagai fitur yang lebih lengkap dibandingkan aplikasi lain, sehingga memudahkan proses pengembangan.