



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Dalam teori ini, meliputi penjelasan tentang Pengertian Komputer, Perangkat Lunak (*Software*), dan Basis Data (*Database*), dan Sistem.

2.1.1 Pengertian Komputer

(Prawiro, 2019 dalam Harmayani *et al.*, 2021:1) menjelaskan bahwa Komputer adalah suatu perangkat elektronik yang dapat digunakan untuk mengolah data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga menghasilkan informasi bermanfaat bagi penggunanya.

Blissmer dalam Lonando & Fadillah, (2023: 54) menjelaskan bahwa Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses *input* sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan *output* dalam bentuk informasi.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa komputer adalah sebuah alat komunikasi yang dapat mengelola data dan informasi yang sangat dibutuhkan secara otomatis

2.1.2 Pengertian Perangkat Lunak (*Software*)

Menurut Harmayani *et al.*, (2021: 43) Perangkat lunak atau *software* komputer merupakan sekumpulan data elektronik, yang tersimpan dan kemudian dikendalikan oleh perangkat komputer.

Menurut Zalukhu *et al.*, (2023: 62) Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa perangkat lunak atau *software* adalah sekumpulan data elektronik yang tersimpan dan dikendalikan oleh perangkat komputer, serta program komputer yang terasosiasi dengan dokumentasi perangkat.



2.1.3 Pengertian Basis Data (*Database*)

Menurut Pahlevi dalam Khairil *et al.*, (2021: 19), Basis data adalah sekumpulan data yang saling berhubungan secara logis dan terorganisir dengan baik. Kumpulan data tersebut yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Kumpulan dokumen/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Menurut Kristy dalam Yudanto (2021: 290) Basis Data atau *Database* merupakan kumpulan data yang saling berhubungan, sehingga kita dapat memperoleh informasi data mengenai data dengan cepat. *Database* dapat berasal dari satu atau kumpulan beberapa tabel yang terhubung menjadi satu kesatuan. Tabel merupakan tempat untuk menampung data dengan susunan tertentu. Setiap kolom dalam tabel merupakan *field* yang harus berisi data dengan jenis yang sama, sedangkan setiap baris dalam tabel merupakan *record* berisi kumpulan data yang terdiri atas beberapa *field*.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Ada banyak cara untuk mengorganisasi skema atau memodelkan struktur basis data. Model yang umum digunakan sekarang adalah model relasional, yang menurut istilah layman mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom saling berhubungan di mana setiap tabel terdiri dari baris dan kolom (definisi yang sebenarnya menggunakan terminologi matematika).

Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel. Pada komputer, basis data disimpan dalam perangkat hardware penyimpan, dan dengan *software* tertentu dimanipulasi untuk kepentingan atau kegunaan tertentu. Hubungan atau relasi data biasanya ditunjukkan dengan kunci (*key*) dari tiap file yang ada. Data dalam basis data bersifat *integrated* dan *shared*:



- a. Terpadu (*integrated*), berkas-berkas data yang ada pada basis data saling terkait (terjadi dependensi data);
- b. Berbagi data (*shared*), data yang sama dapat digunakan sejumlah pengguna dalam waktu yang bersamaan atau yang disebut sebagai sistem *multiuser*.

Dalam implementasinya, untuk memudahkan dalam mengakses data, data disusun dalam suatu struktur logis yang menjelaskan bahwa:

- a. Kumpulan tabel menyusun basis data.
- b. Tabel tersusun atas sejumlah *record*.
- c. Sebuah *record* mengandung sejumlah *field*.
- d. Sebuah *field* disimpan dalam bentuk kumpulan bit.

Istilah secara umum tabel yang baik memiliki beberapa karakteristik sebagai berikut:

- a. Memiliki kolom (*field*) dan baris data (*record*).
- b. Nama kolom bersifat unik (tidak ada yang sama).
- c. Sebuah kolom (*field*) hanya memiliki satu tipe data.
- d. Antar kolom tidak terpisah kolom kosong atau tanpa nama kolom.
- e. *Header* (judul kolom) hanya terdiri dari satu baris dan ditempatkan pada baris pertama.
- f. Antar baris data tidak terpisah baris kosong ataupun baris subtotal.

Dalam pengembangan suatu sistem, basis data memiliki peran yang penting dan sangat diperlukan. Berikut merupakan alasan perlunya basis data:

1. Salah satu komponen penting dalam suatu sistem informasi adalah basis data, karena basis data merupakan dasar dalam menyediakan informasi.
2. Informasi dikatakan lebih bernilai jika memiliki manfaat yang efektif dibandingkan dengan biaya dalam mendapatkannya. Dalam hal ini, basis data akan menentukan kualitas informasi.
3. Basis data mengurangi duplikasi data (*data redundancy*).
4. Basis data dapat mengurangi pemborosan tempat simpanan luar.
5. Data akan dapat saling berelasi dengan menerapkan sistem basis data.



Dari pernyataan di atas dapat disimpulkan bahwa basis data (*database*) adalah kumpulan dari data-data terstruktur yang disimpan secara sistematis di dalam suatu sistem komputer.

2.1.4 Pengertian Sistem

Menurut Rasefta dan Esebella (2020: 52), Sistem adalah kumpulan komponen yang saling terkait satu dengan yang lainnya kegiatan pokok untuk mencapai suatu tujuan tertentu.

Menurut Effendy *et al.*, (2023: 4346), Sistem adalah sekumpulan unsur elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa sistem adalah suatu kumpulan komponen atau unsur yang saling terkait dan saling mempengaruhi satu sama lain dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

2.2 Teori Khusus

Dalam teori ini, meliputi penjelasan tentang Pengertian *Data Flow Diagram (DFD)*, *Flowchart*, *Blockchart*, *Entity Relationship Diagram (ERD)*, Kamus Data (*Data Dictionary*), Metode Pengembangan Sistem Menggunakan *Waterfall*, dan Metode Pengujian.

2.2.1 Pengertian *Data Flow Diagram (DFD)*

Sirait dan Seabtian (2019: 3) mengatakan bahwa *Data Flow Diagram (DFD)* adalah menggambarkan atau menjelaskan langkah-langkah kegiatan yang ada di dalam sebuah sistem, aliran-aliran data dimana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanannya dari data tersebut.

Menurut Kristanto dalam Soufitri (2019: 241), *Data Flow Diagram (DFD)* disebut juga dengan Diagram Arus Data (DAD). DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data, kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang

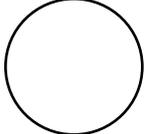
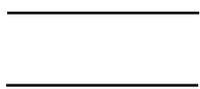


menghasilkan data tersebut, interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Data Flow Diagram* adalah suatu representasi grafis yang digunakan untuk menggambarkan aliran data dan informasi dalam sebuah sistem.

Adapun simbol-simbol yang sering digunakan dalam *Data Flow Diagram* (DFD) dapat dilihat pada tabel berikut ini.

Tabel 2.1 Simbol *Data Flow Diagram* (DFD)

No	Simbol	Nama	Keterangan
1.		Proses (<i>Process</i>)	Proses (<i>Process</i>) atau fungsi atau prosedur pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.
2.		Penyimpanan (<i>Data Store</i>)	Data store adalah file untuk menyimpan data yang digunakan untuk proses selanjutnya. Dapat dikatakan juga, sama seperti basis data (<i>database</i>).
3.		Entitas Luar (<i>External Entity</i>).	Entitas luar merupakan individu, divisi, atau perusahaan yang berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data.
4.		Aliran Data (<i>Data Flow</i>)	<i>Data flow</i> merupakan arus data yang mengalir antara entitas, proses, dan data store.

Sumber: Sukamto (2022: 91-93)



DFD dipopulerkan Oleh DeMacro & Yordan dan Gane & Sarson dengan menggunakan pendekatan Metoda Analisis Sistem Terstruktur. DFD merupakan teknik model proses untuk mengorganisasikan struktur dan alir data di dalam sistem. Dalam laporan ini akan digunakan model DeMarco/Yordan. Berikut adalah langkah-langkah dalam membuat *Data Flow Diagram*.

1. Memahami sistem yang akan dikembangkan dilakukan dengan cara wawancara kepada bagian yang menggunakan serta memproses data.
2. Tentukan batas sistem yang akan digambar, batasan meliputi data yang masuk dan informasi yang akan keluar dari sistem.
3. Setelah diketahui batasan sistem dapat dikembangkan ke dalam diagram konteks. Diagram konteks adalah diagram tingkat atas, merupakan diagram dari sebuah sistem yang menggambarkan aliran-aliran data yang masuk dan keluar dari sistem dan keluar dari agen eksternal.
4. Membuat dekomposisi fungsional untuk memudahkan kita dalam menggambarkan DFD pada level-level berikutnya.
5. Beri nomor urut pada setiap proses dan membuat diagram secara rinci.

2.2.2 Pengertian *Flowchart*

Menurut Zalukhu *et al*, (2023: 62), *Flowchart* adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut.

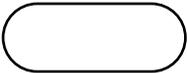
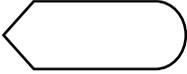
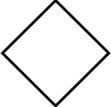
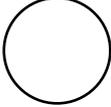
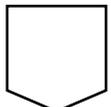
Menurut Zalukhu *et al*, (2023: 62), *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Flowchart* merupakan penggambaran langkah-langkah atau urutan prosedur dalam bentuk notasi-notasi tertentu.

Adapun simbol-simbol yang sering digunakan dalam *Flowchart* dapat dilihat pada tabel berikut ini.

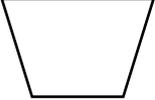


Tabel 2.2 Simbol *Flowchart*

No	Simbol	Nama	Keterangan
1.		Terminator	Terminator berfungsi sebagai awal atau akhir dari sebuah sistem.
2.		Proses (<i>Process</i>)	Simbol yang menyatakan suatu proses yang dilakukan oleh komputer.
3.		<i>Input / Output.</i>	Simbol yang menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung peralatan.
4.		<i>Display</i>	Simbol yang menyatakan layar peraga (monitor).
5.		Aliran (<i>Flow</i>)	Simbol yang menyatakan aliran / alur dari sebuah sistem.
6.		Keputusan (<i>Decision</i>)	Simbol yang menunjukkan kondisi tertentu dan menghasilkan dua kemungkinan jawaban.
7.		<i>On Page Connector</i>	Simbol yang berfungsi menghubungkan bagian <i>flowchart</i> yang terdapat pada satu halaman.
8.		<i>Off Page Connector</i>	Simbol yang berfungsi menghubungkan bagian <i>flowchart</i> yang terdapat pada halaman berbeda.
9.		Dokumen (<i>Document</i>)	Simbol untuk mencetak keluaran dokumen berupa kertas.



Lanjutan **Tabel 2.2** Simbol *Flowchart*

10.		<i>Predefined Process</i>	Simbol yang menyatakan proses permulaan penjalanan sub program.
11.		<i>Manual Operation</i>	Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.

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

2.2.3 Pengertian *Blockchart*

Menurut Zahara *et al.*, (2023: 330), *Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Menurut Lestari (2021: 39), Diagram blok (*Blockchart Diagram*) berfungsi memodelkan masukan, keluaran, referensi, master, proses maupun transaksi dalam simbol-simbol tertentu.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Blockchart* berfungsi untuk menjelaskan alur masukan, keluaran, dan proses dari sebuah sistem dengan simbol-simbol tertentu. *Blockchart* juga dapat memudahkan pengguna dalam memahami alur dari sebuah sistem.

Adapun simbol-simbol yang sering digunakan dalam *Blockchart* dapat dilihat pada tabel berikut ini.

Tabel 2.3 Simbol *Blockchart*

No	Simbol	Nama	Keterangan
1.		Terminator	Terminator berfungsi sebagai awal atau akhir dari sebuah sistem.



Lanjutan **Tabel 2.3** Simbol *Blockchart*

2.		Proses (<i>Process</i>)	Simbol yang menyatakan suatu proses yang dilakukan oleh komputer.
3.		<i>Input / Output.</i>	Simbol yang menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung peralatan.
4.		<i>Display</i>	Simbol yang menyatakan layar peraga (monitor).
5.		Aliran (<i>Flow</i>)	Simbol yang menyatakan aliran / alur dari sebuah sistem.
6.		Keputusan (<i>Decision</i>)	Simbol yang menunjukkan kondisi tertentu dan menghasilkan dua kemungkinan jawaban.
7.		Dokumen (<i>Document</i>)	Simbol untuk mencetak keluaran dokumen berupa kertas.
8.		<i>Multi Document</i>	Simbol untuk mencetak lebih dari 1 keluaran dokumen berupa kertas.
9.		<i>Manual Input</i>	Simbol untuk pemasukan data secara manual melalui keyboard.

Sumber: Kristanto (2018: 76)



2.2.4 Pengertian *Entity Relationship Diagram* (ERD)

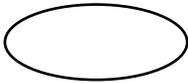
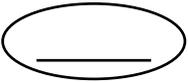
Menurut Pane *et al*, (2022: 91), *Entity Relationship Diagram* (ERD) adalah suatu bentuk diagram yang menjelaskan hubungan antar objek - objek data yang mempunyai hubungan antar relasi.

Menurut Widjanarko *et al*, (2022: 138) menjelaskan bahwa *Entity Relationship Diagram* (ERD) adalah sebuah bentuk diagram yang digunakan untuk menjelaskan hubungan antara objek-objek data dalam suatu basis data. ERD merepresentasikan entitas sebagai objek-objek dasar data yang memiliki atribut dan menjelaskan hubungan antar entitas melalui relasi.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *Entity Relationship Diagram* (ERD) merupakan model data yang berbasis objek dan representasi grafis dari logika *database*. ERD mencakup deskripsi detail tentang entitas, hubungan dan batasan dalam suatu sistem *database*.

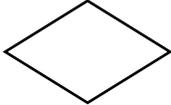
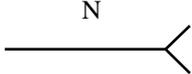
Adapun simbol-simbol yang sering digunakan dalam *Entity Relationship Diagram* (ERD) dapat dilihat pada tabel berikut ini.

Tabel 2.4 Simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Nama	Keterangan
1.		Entitas (<i>Entity</i>)	Entitas merupakan benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh sistem atau aplikasi komputer.
2.		Atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.		Atribut Kunci Primer	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan. Kunci Primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut bersifat unik.



Lanjutan **Tabel 2.4** Simbol *Entity Relationship Diagram* (ERD)

4.		Atribut Multi Nilai	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.		Relasi (<i>Relation</i>)	Simbol ini digunakan untuk menunjukkan hubungan yang ada antara entitas yang satu dan entitas yang lainnya. Biasanya diawali dengan kata kerja.
6.		Asosiasi	Penghubung antara relasi dan entitas di mana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

Sumber: Rosa A.S & M. Shalahuddin (2018: 50-51)

2.2.5 Pengertian Kamus Data (*Data Dictionary*)

Menurut Yunita *et al.*, (2023: 173), Kamus data merupakan suatu penjelasan tertulis secara lengkap dan jelas tentang suatu data dalam *database* sesuai dengan sistem yang dibuat. Kamus data dibuat untuk mengurangi duplikasi data.

Menurut Premana (2019: 56), Kamus data adalah penjabaran dari aliran-aliran data yang ada di dalam sebuah data flow diagram. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa kamus data merupakan suatu dokumen yang penting dalam pengelolaan data dan pengembangan sistem informasi karena memberikan panduan yang jelas tentang struktur data serta aliran-aliran data yang terjadi dalam sistem.

Adapun simbol-simbol yang sering digunakan dalam kamus data dapat dilihat pada table berikut.



Tabel 2.5 Simbol Kamus Data

No	Simbol	Keterangan
1.	=	Disusun atau terdiri atas
2.	+	Dan
3.	()	Opsional
4.	[]	Memilih salah satu alternatif
5.	**	Komentar
6.	@	Identifikasi atribut kunci
7.		Pemisah alternatif simbol

Sumber: Rosa A.S & M. Shalahuddin (2018: 74)

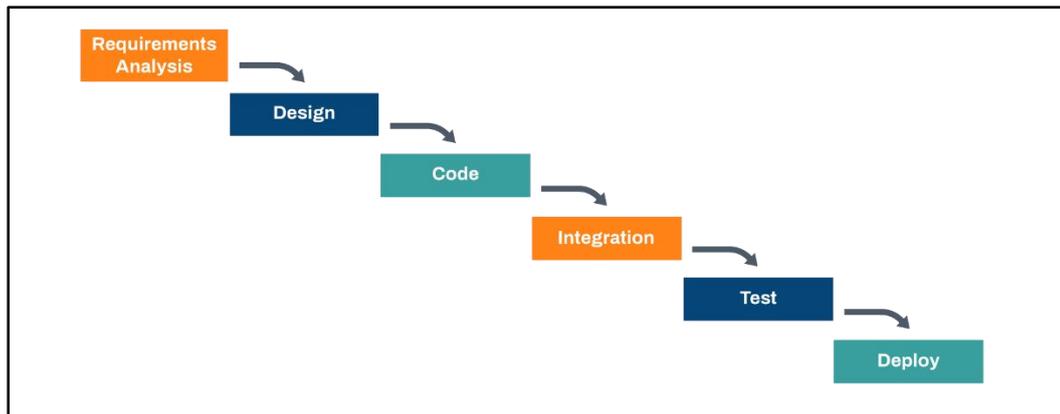
2.2.6 Metode Pengembangan Sistem

Metode pengembangan sistem yang penulis gunakan adalah metode *waterfall*. Menurut Lonando dan Fadillah (2023: 54), Metode *waterfall* adalah rangkaian pada saat proses pengembangan software secara berurutan, dimana kemajuan akan tetap terus mengalir kebawah melalui proses perencanaan, pemodelan, implementasi dan pengujian.

Kinaswara *et al*, 2019 dalam Lonando & Fadillah, (2023: 54) menjelaskan pengertian dari metode *waterfall* juga rangkaian pada saat proses pengembangan *software* secara berurutan, dimana kemajuan akan tetap terus mengalir kebawah melalui proses perencanaan, pemodelan, implementasi dan pengujian.



Pengembangan sistem dengan menggunakan metode *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari Analisis, Desain, Pengodean, Pengujian dan Tahap Pendukung (*support*). Berikut ini adalah gambar ilustrasi model *waterfall*.



Sumber: Sukamto & Shalahuddin (2016)

Gambar 2.1 Metode *Waterfall*

1) Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan.

2) Desain perangkat lunak

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.



3) Pembuatan kode program

Desain harus dibuat kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4) Pengujian Aplikasi

Pengujian fokus pada perangkat lunak dari segi logik dan fungsional serta memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5) Pendukung

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2.7 Metode Pengujian

Pengujian sistem merupakan proses pengeksekusian sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut cocok dengan spesifikasi sistem dan berjalan dengan lingkungan yang diinginkan. Pengujian sistem sering diasosiasikan dengan pencarian *bug*, ketidaksempurnaan program, kesalahan pada baris program yang menyebabkan kegagalan pada eksekusi sistem perangkat lunak. Pengujian yang dilakukan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian *White-Box* dan *Black-Box*. Menurut Wahyudi dan Jakara (2023: 25) *Black-Box* adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak.



2.3 Teori Judul

Dalam teori ini, meliputi penjelasan tentang Pengertian Aplikasi, Pengolahan Data, Proyek Konstruksi, *Website*, *CV (Commanditaire Venootschap)*, serta Aplikasi Pengolahan Data Proyek Konstruksi Berbasis *Website* Pada CV. Mansyur Bersaudara Palembang

2.3.1 Pengertian Aplikasi

Menurut Habibi dan Karnovi (2020:14) menjelaskan bahwa, “Aplikasi adalah sebuah program siap pakai yang bisa dipakai untuk menjalankan sejumlah perintah dari pengguna aplikasi itu sendiri. Aplikasi juga memiliki pengertian sebagai pemecah masalah yang memakai salah satu Teknik pemrosesan data aplikasi yang mengacu pada sebuah komputerisasi atau *smarthphone* yang diinginkan atau diharapkan”.

Menurut Huda dan Priyatna (2019:82) menjelaskan bahwa, “Aplikasi adalah suatu program berbentuk perangkat lunak yang berjalan pada suatu system tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia”.

Menurut Rachmad Hakim S (2018:12), menjelaskan bahwa “Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur windows, permainan (*game*), dan sebagainya”.

Berdasarkan beberapa definisi diatas dapat disimpulkan bahwa aplikasi adalah perangkat lunak yang berfungsi untuk menjalankan perintah sesuai dengan kemampuan yang dimilikinya.

2.3.2 Pengertian Pengolahan Data

Menurut Hutahaean (2020:8) menjelaskan bahwa, “Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan”.



Menurut Sutarman (2020:4) menjelaskan bahwa, “Pengolahan data adalah proses perhitungan atau transformasi data input menjadi informasi yang mudah dimengerti ataupun sesuai dengan yang diinginkan”.

Menurut Rohmah dan Sa’adah (2021:71) menjelaskan bahwa, ”Pengolahan Data adalah proses yang memberikan pengawasan pada semua hal yang terlibat dalam pelaksanaan kebijaksanaan tujuan”.

Berdasarkan beberapa definisi diatas dapat disimpulkan bahwa pengolahan data melibatkan serangkaian operasi atau proses untuk mengubah data mentah menjadi informasi yang bermakna dan berguna sesuai dengan tujuan atau kebutuhan yang diinginkan.

2.3.3 Pengertian Proyek Konstruksi

Menurut Ervianto (2018:9) menjelaskan bahwa, “Proyek konstruksi ialah suatu rangkaian kegiatan yang dilaksanakan sekali dan berjangka waktu pendek”.

Menurut Dimiyati dan Nurjaman (2019:6) menjelaskan bahwa, “Ada sebagian pihak yang terlibat dalam proyek konstruksi yaitu pemilik proyek (*owner*), konsultan, pelaksana proyek (kontraktor), dan konsultan perencana”.

Menurut Fardinal, *et al.*, (2022:374) menjelaskan bahwa, “Proyek Kontruksi adalah suatu kegiatan yang di lakukan untuk membangun sarana dan prasarana, suatu konstruksi juga dikenal sebagai sebuah bangunan pada sebuah area”.

Berdasarkan beberapa definisi diatas dapat disimpulkan bahwa proyek konstruksi merupakan suatu rangkaian kegiatan yang saling berkaitan untuk mencapai tujuan tertentu (bangunan/konstruksi) dalam batasan waktu, biaya dan mutu tertentu.

2.3.4 Pengertian Website

Menurut Yuhefizar dalam Sa’ad (2020:3) menjelaskan bahwa, “*Website* merupakan keseluruhan halaman-halaman web yang terdapat dalam sebuah domain



yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman web yang saling berhubungan”.

Menurut Salamah, dkk (2020:3) menjelaskan bahwa, “*Website* adalah halaman informasi yang disediakan jalur internet sehingga bisa diakses dimana saja, selama terkoneksi dengan jaringan”.

Menurut Herlina (2020:9), menjelaskan bahwa, “Pengolahan data adalah bagian dari rangkaian kegiatan penelitian setelah pengumpulan data”.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa *website* adalah bagian dari internet yang terdiri dari halaman-halaman yang dapat diakses oleh *browser website*.

2.3.5 Pengertian CV (*Commanditaire Venootschap*)

Menurut Partyani, (2020:1) menjelaskan bahwa, “Perseroan Komanditer (CV) atau seringkali disebut dengan *Commanditaire Venootschap* (untuk selanjutnya disebut CV) adalah suatu perseroan terbatas untuk menjalankan suatu perusahaan yang dibentuk antara satu orang atau beberapa orang persero yang secara tanggung jawab memikul tanggung jawab untuk seluruhnya”.

Menurut Partyani, (2020:71) menjelaskan bahwa, “*Commanditaire Venootschap* (CV) adalah suatu persekutuan yang didirikan oleh seorang atau beberapa orang yang mempercayakan uang atau barang kepada seorang atau beberapa orang yang menjalankan perusahaan dan bertindak sebagai pemimpin”.

Menurut Pasal 19 KUHD menjelaskan bahwa, “CV adalah sebagai persekutuan dengan peminjaman uang (*Geldscheiter*) atau disebut juga persekutuan komanditer, diadakan antara seorang sekutu atau lebih yang bertanggung jawab secara pribadi dan untuk seluruhnya dengan seorang atau lebih sebagai sekutu yang meminjamkan uang”.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa CV merupakan persekutuan di mana pemilik modal mempercayakan modalnya kepada pengelola perusahaan yang bertanggung jawab atas jalannya usaha, sedangkan *komanditer* hanya bertanggung jawab sebatas besarnya modal yang disetorkannya.



2.3.6 Pengertian Aplikasi Pengolahan Data Proyek Konstruksi Berbasis Website Pada CV Mansyur Bersaudara Palembang

Aplikasi pengolahan data proyek konstruksi berbasis website pada CV Mansyur Bersaudara Palembang adalah suatu sistem perangkat lunak yang dirancang dan dikembangkan untuk memfasilitasi pengelolaan data terkait proyek konstruksi mereka melalui *platform website*. Aplikasi ini bertujuan untuk menyederhanakan dan meningkatkan efisiensi dalam proses pengelolaan proyek konstruksi, mulai dari tahap perencanaan, pelaksanaan, hingga pemantauan dan evaluasi.

2.4 Teori Program

Dalam teori ini, meliputi penjelasan tentang Pengertian *Visual Studio Code*, *XAMPP*, *MySQL*, *Bootstrap*, *PHP*, *HTML*, *CSS*, dan *Javascript*.

2.4.1 Pengertian Visual Studio Code

Umyy Gusti Salamah dalam Ningsih *et al.*, (2022: 95) menjelaskan bahwa *Visual Studio Code* adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung berbagai bahasa pemrograman dengan bantuan *plugin* yang dapat dipasang via *marketplace Visual Studio Code* seperti: *C++*, *C#*, *Python*, *Go*, *Java*, *PHP*, dst.

Visual Studio Code merupakan pilihan utama untuk pengembangan sistem karena ekstensibilitasnya yang luas, dukungan yang komprehensif untuk berbagai bahasa pemrograman, serta kinerja yang ringan dan responsif di berbagai platform. Dengan ribuan ekstensi yang tersedia, pengguna dapat menyesuaikan lingkungan pengembangan sesuai kebutuhan, sementara dukungan untuk banyak bahasa pemrograman memungkinkan fleksibilitas dalam pengembangan sistem.

2.4.2 Pengertian XAMPP

Menurut Ningsih *et al.*, (2021: 2), *XAMPP* merupakan aplikasi atau perangkat lunak bebas atau yang dapat berfungsi sebagai server yang berdiri sendiri



(*localhost*) dan dapat mendukung banyak sistem operasi, XAMPP terdiri dari beberapa program yaitu, *Apache* HTTP Server, *MySQL database*, dan bahasa yang ditulis dapat diterjemahkan dengan bahasa pemrograman PHP dan *Perl*.

Enterprise dalam Hidayat *et al.*, (2022: 45) mengatakan bahwa XAMPP merupakan server yang paling banyak digunakan. Fiturnya lengkap, mudah digunakan programmer PHP pemula karena yang perlu digunakan hanyalah menjalankan salah satu module bernama *Apache* yang dapat memproses PHP.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa XAMPP adalah aplikasi atau perangkat lunak bebas yang berfungsi sebagai *server* lokal yang berdiri sendiri (*localhost*) dan mendukung berbagai sistem operasi untuk mengembangkan dan menguji situs *website* secara *offline*.

XAMPP merupakan software yang dikembangkan oleh sekelompok tim *Apache Friend* pada 2002 dan bisa didapatkan secara gratis dengan label *General Public License* (GNU). Sebagai *software open source* berbasis *web server*, XAMPP ini memiliki berbagai program dan mendukung berbagai sistem operasi yang umum digunakan, seperti Linux, Windows, MacOS, dan Solaris. Aplikasi ini berfungsi sebagai server lokal yang sudah mencakup program *Apache*, *MySQL*, dan PHP.

XAMPP disebut sebagai *standalone server* atau server yang dapat berdiri sendiri sehingga memudahkan pengguna saat menjalankan proses pengeditan, desain, dan pengembangan sistem atau aplikasi. Penggunaan XAMPP dapat dikatakan mampu menghemat anggaran karena dapat menggantikan peran *web hosting* dengan cara menyimpan file *website* ke dalam *localhost* agar bisa dihubungkan melalui browser. Kepanjangan XAMPP merupakan program yang tersedia pada *software* ini.

1. X (*Cross Platform*)

Kode ini merupakan kode penanda untuk *software cross platform* atau yang bisa berjalan di banyak sistem operasi, seperti Windows, Linux, MacOS, dan Solaris.



2. A (*Apache*)

Berarti *web server* gratis (*opensource*) yang bisa dikembangkan oleh banyak orang untuk menciptakan halaman *web*. Dengan menggunakan *web server*, pengguna dapat menjalankan file yang berisi bahasa pemrograman PHP pada *localhost*.

3. M (*MySQL/MariaDB*)

MySQL adalah salah satu aplikasi *database server* dengan Bahasa pemrograman *structured query language* (SQL) yang berfungsi untuk mengelola data secara terstruktur dan sistematis. Misalnya, mengolah, mengedit, dan menghapus daftar melalui *database*. *MySQL* juga dapat digunakan di *localhost* tanpa memerlukan sambungan internet. Sedangkan, *MariaDB* merupakan sistem manajemen *database* yang merupakan bagian dari pengembangan mandiri *MySQL*.

4. P (*PHP*)

Hypertext Preprocessor (PHP) merupakan bahasa pemrograman khusus berbasis *web* untuk kebutuhan pada sisi server atau *back end* sehingga dapat digunakan untuk membuat halaman *web* menjadi lebih dinamis dengan menerapkan *server-side scripting*. Selain itu, PHP juga mendukung manajemen sistem pada *Oracle*, *PostgreSQL*, *Microsoft Access*, dan lain sebagainya.

5. P (*Perl*)

Perl adalah bahasa pemrograman yang dapat digunakan untuk segala kebutuhan dan dapat berjalan di berbagai sistem operasi sehingga sangat fleksibel dan banyak digunakan. Bahasa pemrograman ini juga berfungsi sebagai penunjuk eksistensi dari PHP. *Perl* banyak digunakan untuk keperluan pengembangan aplikasi, *web server*, hingga sistem berbasis *Content Management System* (CMS), seperti *WordPress*.

Adapun komponen-komponen penting yang terdapat pada XAMPP adalah sebagai berikut.



1. **Control Panel**, merupakan layanan yang digunakan untuk mengelola XAMPP.
2. **HTDocs**, merupakan komponen XAMPP yang berbentuk folder dan berfungsi sebagai tempat penyimpanan file *website*.
3. **Config**, merupakan menu yang digunakan untuk melakukan konfigurasi dasar pada XAMPP, seperti mengatur aplikasi editor teks dan browser yang akan digunakan secara default oleh XAMPP.
4. **PhpMyAdmin**, merupakan komponen XAMPP yang berfungsi untuk mengelola *database* melalui browser.
5. **NetStat**, merupakan menu untuk memastikan *port* yang digunakan oleh XAMPP telah digunakan oleh aplikasi lain atau belum. Bila *port* standar XAMPP sudah digunakan oleh aplikasi lain, maka XAMPP tidak dapat berfungsi dengan baik. Dengan demikian, pengguna harus mengubah *port* tersebut dengan yang lainnya yang masih belum digunakan.

XAMPP diperlukan dalam pembuatan sistem berbasis *website* ini karena menyediakan lingkungan pengembangan lokal yang lengkap, mudah digunakan, dan fleksibel untuk mengembangkan, menguji, dan memelihara situs *web*.

2.4.3 Pengertian MySQL

Menurut Wahyudi dan Jakara (2023: 24), *MySQL (My Structure Query Language)* adalah sebuah program berbasis DOS yang bersifat *open source*. MySQL adalah produk yang berjalan pada platform baik windows maupun Linux. Selain itu, *MySQL* merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk *multiuser* (banyak pengguna).

Menurut Pane *et al.*, (2022: 94), *MySQL* adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah *database server* yang gratis dengan lisensi GNU *General Public*



License (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada.

Menurut Wardhani (2022: 95), dalam MySQL terdapat fitur-fitur yang sangat melekat pada perangkat lunak ini, yaitu:

1. *Relational Database System*.
2. Arsitektur *Client – Server*.
3. Mendukung *Sub Select*.
4. Mendukung *View, Stored Prosedured (SP), Triggers, Replication, Transaction, dan Foreign Key (FK)*.
5. Tersedia fungsi GIS, bebas diunduh (*download*), stabil, tangguh, fleksibel, dan memiliki keamanan (*security*) yang baik.
6. Mengenal perintah *Structured Query Language (SQL)* standar.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa *MySQL* adalah sebuah sistem untuk menyimpan, mengelola, dan mengakses data secara efisien dalam aplikasi *web* dan berbagai sistem yang memerlukan penyimpanan data yang bersifat jaringan serta memungkinkan pengguna *multiuser*, dan dapat digunakan pengguna tanpa biaya lisensi.

MySQL digunakan dalam pembuatan sistem ini karena *MySQL* memiliki kelebihan sebagai berikut.

1. Berlisensi GPL dan dapat dijalankan di berbagai sistem operasi.
2. Dapat diintegrasikan dengan beberapa bahasa pemrograman.
3. Mendukung ODBC untuk sistem operasi windows.
4. Bisa dijalankan pada spesifikasi *hardware* yang rendah karena lebih hemat *resource memory* (dibandingkan *database* lain) sehingga mudah digunakan untuk bahan pembelajaran.
5. *MySQL* dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari 20 bahasa meskipun bahasa indonesia belum termasuk didalamnya.



2.4.4 Pengertian *Bootstrap*

Menurut Firmansyah dan Herman (2023: 364), *Bootstrap* adalah salah satu kerangka kerja dari CSS yang digunakan untuk membantu *developer* dalam mempermudah, mempercepat dan memperindah tampilan website tersebut.

Suprayogi dan Rahmanesa dalam Suli dan Nirsal, (2023: 27), *Bootstrap* adalah *front-end framework* yang mengedepankan tampilan untuk *mobile device* guna mempercepat dan mempermudah pengembangan *website*.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa *Bootstrap* adalah sebuah kerangka kerja (*framework*) *front-end* CSS yang dirancang untuk memudahkan dan mempercepat pengembangan tampilan *website*. *Bootstrap* membantu pengembang dalam menciptakan desain yang menarik dan responsif secara efisien.

2.4.5 Pengertian PHP

Menurut Ningsih *et al.*, (2022: 95), *Hypertext Preprocessor* (PHP) merupakan *script* untuk pemrograman *script web server-side*, *script* yang membuat dokumen HTML secara *on the fly*, maksudnya dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML.

Menurut Firmansyah dan Herman (2023: 364), PHP atau singkatan dari *Hypertext Preprocessor* merupakan salah satu bahasa pemrograman *open source* yang digunakan untuk komunikasi ke sisi server yang kemudian transfer hasilnya ke *client* yang melakukan permintaan.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa PHP adalah sebuah bahasa pemrograman skrip yang digunakan secara server-side untuk menghasilkan dokumen HTML secara dinamis.

Penggunaan PHP memungkinkan *web* dapat dibuat dinamis sehingga *maintenance web* menjadi lebih mudah dan efisien. PHP ditulis menggunakan bahasa C. PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa pemrograman lainnya. PHP difokuskan pada pembuatan *script server-side* yang



bisa melakukan apa saja yang dilakukan oleh CGI, seperti mengumpulkandata dari form, menghasilkan isi halaman web dinamis, dan kemampuan mengirim serta menerima *cookies*, bahkan lebih daripada kemampuan CGI.

PHP tidak terbatas pada hasil keluaran HTML, namun PHP juga memiliki kemampuan untuk mengolah gambar, file PDF, dan *movie flash*. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya. Salah satu fitur yang dapat diandalkan oleh PHP adalah dukungan terhadap *database*, salah satunya adalah *MySQL*.

2.4.6 Pengertian HTML

Menurut Kaban dan Sembiring (2021: 10), HTML (*HyperText Markup Language*) adalah bahasa *markup* yang digunakan *website* browser untuk menafsirkan dan menulis teks, gambar dan konten lainnya ke dalam halaman *website* secara visual maupun suara.

Menurut Firmansyah dan Herman (2023: 364), HTML atau singkatan dari (*HyperText Markup Language*) merupakan salah satu bahasa *markup* standar yang digunakan untuk membuat sebuah halaman *website*.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa HTML merupakan fondasi dari struktur dan tampilan halaman *website*, memungkinkan interaksi antara pengguna dan konten *website* melalui berbagai media.

2.4.7 Pengertian CSS

Menurut Firmansyah dan Herman (2023: 364), CSS atau singkatan dari (*Cascading Style Sheet*) merupakan salah satu bahasa *stylesheet* yang digunakan untuk memperindah atau mempercantik sebuah tampilan *website*. Biasanya CSS membantu HTML dalam proses memperindah sebuah *website*.

Menurut Arisantoso *et al.*, (2023: 85), CSS adalah seperangkat alat untuk membuat dan memodifikasi jenis huruf, warna, bentuk, dan elemen terkait gaya lainnya yang tidak dapat dilakukan menggunakan elemen HTML. CSS merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan



lebih terstruktur. CSS bukan sebuah bahasa pemrograman, melainkan *styles* seperti dalam aplikasi Microsoft Word yang dapat mengatur beberapa *style*.

Berdasarkan beberapa defisini diatas, dapat disimpulkan bahwa CSS adalah bahasa *stylesheet* yang digunakan untuk meningkatkan estetika dan tata letak sebuah halaman *website*.

2.4.8 Pengertian *Javascript*

Menurut Hardianto *et al.* (2023: 116), (*Javascript* adalah bahasa pemrograman yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada browser seperti Google Chrome, Mozilla Firefox, Opera Mini dan sebagainya.

Menurut Arisantoso *et al.*, (2023: 59), *Javascript* adalah bahasa pemrograman tingkat tinggi yang pertama kali dibuat untuk membuat *website* lebih hidup. *Javascript*, bersama dengan HTML dan CSS, adalah bahasa pemrograman yang paling umum digunakan untuk membuat aplikasi berbasis *website*. Bahasa ini dapat membuat *website* lebih interaktif.

Berdasarkan beberapa definisi diatas, dapat disimpulkan bahwa *Javascript* adalah sebuah bahasa pemrograman yang berperan penting dalam pengembangan aplikasi *website*. Dalam konteks pengembangan *website*, *Javascript* bekerja bersama dengan HTML dan CSS untuk menciptakan interaktivitas dan dinamika yang meningkatkan pengalaman pengguna secara keseluruhan.