



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Harmayani, dkk (2019:2), mendefinisikan bahwa "Komputer adalah alat elektronik yang terdiri dari rangkaian berbagai komponen yang saling terhubung sehingga membentuk suatu sistem kerja. Sistem di dalam komputer tersebut dapat melakukan pekerjaan secara otomatis berdasarkan program yang diperintahkan kepadanya sehingga mampu menghasilkan informasi berdasarkan data dan program yang ada".

Sedangkan menurut Yahfizham (2019:13), menjelaskan bahwa "Komputer adalah serangkaian mesin elektronik yang terdiri dari jutaan komponen yang dapat saling bekerja sama, serta membentuk sebuah sistem kerja yang rapi dan teliti. Sistem ini kemudian digunakan untuk dapat melaksanakan pekerjaan secara otomatis, berdasarkan instruksi (Program) yang diberikan kepadanya".

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa komputer adalah sebuah alat elektronik yang terdiri dari rangkaian berbagai komponen yang saling terhubung membentuk suatu sistem kerja. Komputer juga dapat melakukan pekerjaan secara otomatis berdasarkan program atau instruksi yang diberikan kepadanya.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Menurut Harmayani, dkk (2019:43), mendefinisikan bahwa "Perangkat lunak atau (*Software*) merupakan sekumpulan data elektronik, yang tersimpan dan kemudian dikendalikan oleh perangkat komputer. Data elektronik tersebut berupa instruksi atau program, yang nantinya akan menjalankan perintah khusus".

Sedangkan menurut Ahmad, dkk (2023:36), "Perangkat lunak atau *software* merupakan program komputer yang dapat menerjemahkan perintah perintah atau



input-an dari pengguna, sehingga dapat dipahami, diproses, dijalankan dan memperoleh hasil atau *output* yang sesuai dengan keinginan dari penggunanya”.

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa perangkat lunak (*Software*) adalah kumpulan data elektronik yang disimpan dan dikendalikan oleh komputer. Perangkat lunak tersebut berupa program komputer yang dapat menerjemahkan perintah atau *input* dari pengguna untuk diproses, dijalankan, dan menghasilkan *output* sesuai dengan keinginan pengguna.

2.1.3 Pengertian Basis Data (*Database*)

Menurut Fikry (2019:2), mendefinisikan bahwa “Basis Data (*Database*) adalah Kumpulan dari item data yang saling berhubungan satu dengan lainnya yang diorganisasikan berdasar sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* digunakan untuk melakukan manipulasi data (Diperbaharui, dicari, diolah dengan perhitungan-perhitungan tertentu, dan dihapus) dengan tujuan tertentu”.

Sedangkan menurut Pratiwi, dkk (2020:1), menjelaskan bahwa “Basis data atau (*Database*) adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (Program aplikasi) untuk menghasilkan informasi”.

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa basis data (*Database*) adalah kumpulan data yang tersimpan secara terstruktur di dalam komputer, yang dapat dimanipulasi menggunakan perangkat lunak atau program aplikasi untuk melakukan berbagai operasi seperti pembaruan, pencarian, pengolahan dengan perhitungan tertentu, dan penghapusan.

Setiap *database* mempunyai *Application Programming Interface* (API) tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalam sehingga dapat dimanfaatkan oleh aplikasi lainnya. Secara prinsip, dalam suatu *database* tercakup dua komponen penting, yaitu data dan informasi.

Sistem basis data adalah sistem terkomputerisasi dengan tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi



tersedia saat dibutuhkan. Sistem informasi tidak dapat dipisahkan dengan kebutuhan basis data apapun bentuknya, entah berupa file teks ataupun *Database Management System* (DBMS).

Menurut Fikry (2019:6), "DBMS adalah Sistem pengorganisasian dan pengolahan *database* pada komputer. Sistem ini dirancang untuk mampu melakukan berbagai data dengan beberapa referensi data yang sama".

Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut:

- 1) Menyediakan fasilitas untuk mengelola akses data.
- 2) Mampu menangani integritas data.
- 3) Mampu menangani akses data yang dilakukan.
- 4) Mampu menangani backup data.

Sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Dari beberapa pendapat diatas dapat disimpulkan basis data (*Database*) adalah kumpulan data atau file yang tersimpan di dalam komputer sehingga dapat diolah menjadi informasi.

2.1.4 Pengertian Sistem

Sebuah sistem mengacu pada suatu benda, dimana sangat dibutuhkan dalam membangun sebuah prosedur-prosedur serta komponen-komponen dalam tujuan tertentu. Berikut beragam definisi yang dikemukakan oleh para ahli diantaranya.

Menurut Sutejo dalam Yahfizham (2019:61), mendefinisikan bahwa "Sistem adalah kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan dalam usaha mencapai suatu tujuan".

Jogianto dalam Widarti, dkk (2024:2), "Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan suatu objek yang nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi".

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa sistem adalah kumpulan elemen-elemen yang saling terkait dan berinteraksi untuk



mencapai suatu tujuan tertentu. Sistem mencakup elemen-elemen yang dapat berupa objek nyata, seperti tempat, benda, dan orang-orang, serta kejadian-kejadian yang terjadi.

2.1.5 Pengertian Internet

Menurut Rusito (2021:1), menjelaskan bahwa “Internet merupakan sekumpulan jaringan komputer menghubungkan *website*/situs akademik, pemerintahan, bisnis, organisasi, dan individu. Internet menyediakan akses ke pelayanan telekomunikasi serta sumber informasi bagi jutaan *user* di seluruh dunia. Layanannya antara lain komunikasi langsung (*Email/surel, chating*), diskusi (*Usenet news, Email/surel, mailing list*), sumber informasi terdistribusi (*WWW, gopher*), *Login remote* dan *trafic file (telnet, FTP)* dan lainnya”.

Sedangkan Menurut Rosika dan Pratama (2021:9), “Internet adalah suatu jaringan komunikasi yang mampu menghubungkan satu media elektronik dengan media lain”.

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa internet adalah sebuah jaringan komputer yang menghubungkan berbagai situs, organisasi, bisnis, dan individu di seluruh dunia. Internet juga menyediakan akses ke layanan telekomunikasi serta sumber informasi bagi jutaan pengguna di berbagai belahan dunia.

2.1.6 Metode Pengembangan Sistem

Menurut Vindua (2024:15), menjelaskan bahwa “Metode Pengembangan Sistem adalah sebuah metodologi yang digunakan dalam pengembangan perangkat lunak untuk memastikan bahwa proyek tersebut berjalan dengan baik, sesuai dengan kebutuhan pengguna, dalam batas waktu dan anggaran yang telah ditentukan”. Metode pengembangan sistem yang penulis gunakan adalah metode *waterfall*

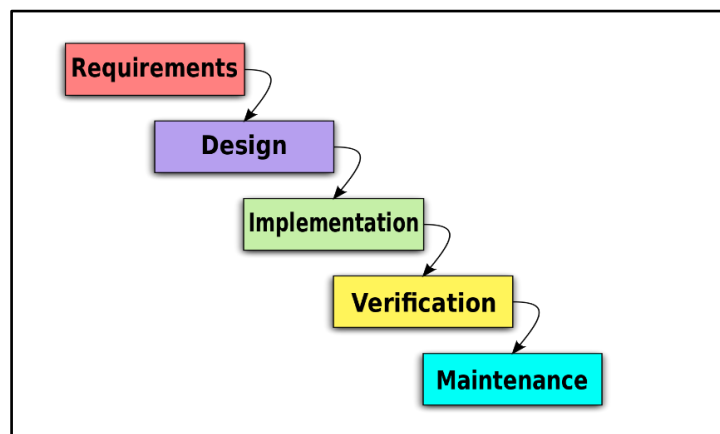
Menurut Hartono (2024:68), menjelaskan bahwa “Model *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara *linear*. Jadi jika



langkah satu belum dikerjakan maka tidak akan bisa melakukan pengerjaan langkah 2, 3 dan seterusnya. Secara otomatis tahapan ke-3 akan bisa dilakukan jika tahap ke-1 dan ke-2 sudah dilakukan”.

Sedangkan menurut Hasanah dan Untari (2020:21), ”Model air terjun (*Waterfall model*) adalah pendekatan klasik dalam pengembangan perangkat lunak yang menggambarkan metode pengembangan linier dan berurutan. Ini terdiri dari lima hingga tujuh *fase*, setiap *fase* didefinisikan oleh tugas dan tujuan yang berbeda, di mana keseluruhan *fase* menggambarkan siklus hidup perangkat lunak hingga pengirimannya. Setelah *fase* selesai, langkah pengembangan selanjutnya mengikuti dan hasil dari *fase* sebelumnya mengalir ke *fase* berikutnya”.

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa model *waterfall* adalah model dalam pengembangan sistem atau perangkat lunak yang dilakukan secara *linear* dan berurutan. Proses pengerjaannya terdiri dari beberapa *fase* yang harus diselesaikan satu per satu, di mana setiap *fase* memiliki tugas dan tujuan yang berbeda. *Fase* berikutnya tidak dapat dimulai sebelum *fase* sebelumnya selesai, sehingga hasil dari setiap *fase* mengalir ke *fase* berikutnya.



Gambar 2.1 Metode *Waterfall*

Sumber: Hartono (2024:68)

Dalam pengembangannya model *waterfall* memiliki tahapan yang berurut. Adapun penjelasan dari tahapan yang ada pada model *waterfall*.



1) Analisa Kebutuhan (*Requirement Analysis*)

Analisa Kebutuhan adalah proses analisa atau pengumpulan data-data yang berkaitan dengan sistem yang akan dibuat. Pada tahap ini, peneliti mengumpulkan data kebutuhan sistem agar sesuai dengan apa yang diperlukan.

2) Desain (*Design*)

Tahap desain ini bertujuan untuk memberikan gambaran lengkap mengenai apa yang harus peneliti kerjakan dalam melakukan tahap desain, bagaimana nantinya tampilan dari sebuah sistem yang akan dibuat. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3) Implementasi (*Implementation*)

Implementasi akan dilakukan penulisan *code*, yaitu membuat *software* yang dilakukan dengan cara memecah modul-modul menjadi kecil yang nantinya digabungkan dalam tahap selanjutnya.

4) Uji Coba (*Testing*)

Pada tahap uji coba akan dilakukan penggabungan modul yang sudah dibuat sebelumnya, setelah itu dilakukannya pengujian yang bertujuan untuk mengetahui apakah *software* sudah berjalan dengan baik atau belum.

5) Pemeliharaan (*Maintenance*)

Tahap ini merupakan tahap terakhir pada *waterfall* model dalam pengembangan sistem. Tahap ini *software* yang telah dibuat atau dijalankan oleh *user*, dimana dilakukan pemeliharaan agar sistem berjalan dengan baik. Tahapan 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 Teori Judul

2.2.1 Pengertian Aplikasi

Menurut (Susanty et al., 2019), mendefinisikan bahwa “Aplikasi adalah satu unit perangkat lunak yang sengaja dibuat untuk memenuhi kebutuhan akan berbagai aktivitas ataupun pekerjaan, seperti aktivitas perniagaan, periklanan, pelayanan masyarakat, *game* dan berbagai aktivitas lainnya yang dilakukan oleh manusia”.

Sedangkan menurut (Wahyuni, 2020), ”Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut”.

Dari beberapa definisi diatas, maka penulis menyimpulkan bahwa aplikasi adalah sebuah program atau perangkat lunak yang dirancang atau dibuat untuk tujuan tertentu dengan melakukan aktifitas tertentu melalui proses dan prosedur aliran data dalam infrastruktur teknologi informasi yang sesuai dengan jenjang dan kebutuhan.

2.2.2 Pengertian Pengolahan Data

Menurut Rusmawan (2019:34), menjelaskan bahwa “Data adalah catatan atas kumpulan fakta. Data merupakan jamak dari datum, berasal dari bahasa latin yang berarti sesuatu yang diberikan”.

Sedangkan menurut Kristanto (2019:8), “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Dari definisi diatas maka, penulis menyimpulkan bahwa pengolahan data adalah sebuah rangkaian pengolahan untuk menghasilkan informasi atau pengetahuan dari sebuah data.



2.2.3 Pengertian Produksi

Menurut Anggraini, dkk (2022:51), mendefinisikan bahwa “Produksi adalah suatu perusahaan yang mengorganisir sumber daya yang dimilikinya, dan mengubah bahan baku (*Input*) menjadi produk (*Output*)”.

Sedangkan menurut Zahra dan Cep (2020:97), “Produksi adalah usaha untuk menciptakan, meningkatkan manfaat barang dengan mengkombinasikan faktor-faktor produksi untuk memenuhi kebutuhan. Produksi merupakan aktivitas ekonomi yang menyediakan barang/jasa sampai ke konsumen. Serta produksi merupakan proses perubahan *input* menjadi *output*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa, produksi merupakan kegiatan yang menghasilkan suatu barang atau jasa.

2.2.4 Pengertian Pengiriman

Menurut (Rahmatuloh & Revanda, 2022), mendefinisikan bahwa “Pengiriman adalah kegiatan pemasaran untuk memudahkan dalam Penyampaian produk dari produsen kepada konsumen”.

Sedangkan menurut (Suryanto dalam Dewantoro et al., 2020), “pengiriman barang atau distribusi merupakan sarana perpindahan barang dari produsen melalui jalur perantara sehingga dapat sampai ke konsumen sebagai pemakai akhir.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa, pengiriman merupakan kegiatan mendistribusikan produk barang dan jasa produsen ke konsumen.

2.2.5 Pengertian Distributor

Menurut Tegar (2019:11), mendefinisikan bahwa “Distributor adalah suatu perantara yang menyalurkan produk dari pabrik (manufacturer) ke pengecer (*Retailer*).

Sedangkan menurut Agustina, dkk (2024:33), “Distributor atau lembaga-lembaga penyalur yang mempunyai kegiatan untuk menyalurkan barang-barang atau jasa-jasa dari produsen ke konsumen”.



Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa distributor adalah orang yang melakukan kegiatan menyalurkan barang/jasa dari produsen (Penjualan) ke konsumen (pembeli).

2.2.6 Pengertian Website

Menurut Anamisa dan Mufarroha (2020:1), mendefinisikan bahwa ” *Website* adalah kumpulan dari halaman-halaman situs, yang terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* (WWW) di dalam internet”.

Sedangkan menurut Sari, Abdilah, dan Sunarti (2019:1), “*Website* merupakan kumpulan halaman digital yang berisi informasi berupa teks, animasi, gambar, suara dan video atau gabungan dari semuanya yang terkoneksi oleh internet, sehingga dapat dilihat oleh seluruh siapapun yang terkoneksi jaringan internet”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Website* adalah kumpulan dari keseluruhan halaman-halaman *web* yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

2.3 Teori Khusus

2.3.1 Pengertian Data Flow Diagram (DFD)

Menurut Sumirat (2023:53), menjelaskan bahwa “*Data Flow Diagram* (DFD) biasa disebut dengan diagram aliran data (DAD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *bubble chart*, *bubble diagram*, model proses, diagram alur kerja, atau model fungsi”.

Sedangkan menurut Hasanah dan Untari (2020:49), ”*Data Flow Diagram*

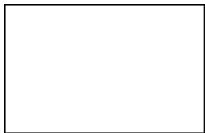
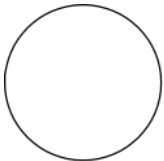
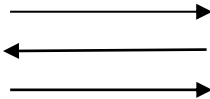
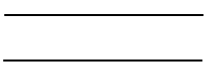


(DFD) merupakan sebuah teknik analisis yang digunakan untuk menggambarkan aliran *input* dalam sebuah sistem yang diolah oleh proses dan menghasilkan suatu keluaran (*Output*)”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Data Flow Diagram* (DFD) adalah representasi grafis dari aliran data di dalam suatu sistem.

Adapun beberapa simbol yang biasanya digunakan dalam perancangan DFD, diantaranya seperti pada tabel dibawah ini:

Tabel 2.1 Simbol-simbol DFD

No	Simbol	Nama Simbol	Keterangan
1.		Entitas Luar (<i>External Entity</i>)	Entitas luar dapat berupa orang unit terkait yang berinteraksi dengan sistem tetapi diluar sistem.
2.		Proses	Orang atau unit yang mempergunakan/melakukan tranformasi data. Komponen fisik tidak diidentifikasi.
3.		Aliran Data	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>).
4.		File atau basis data	Penyimpanan data atau tempat data dilihat oleh proses.

Sumber: Sumirat, dkk (2023:54-55)



Lanjutan Tabel 2.1 Simbol-simbol DFD

5.		<i>Jump</i>	<i>Jump</i> disebut lompatan garis, juga dikenal sebagai lompatan tepi atau jembatan, digunakan untuk menunjukkan bahwa garis-garis saling bersilangan, tetapi tidak terhubung. Mereka dapat digunakan dalam diagram kelistrikan atau diagram apa pun yang memiliki panah penghubung yang saling bersilangan, untuk membuat diagram agar tidak ambigu dan mudah 'dibaca'.
----	---	-------------	---

Sumber: Kymberly Fergusson (2017)

Dinyatakan oleh Rosa dan Shalahuddin (2018:72-7), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*: DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
2. Membuat DFD Level 1 DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah di buat.
3. Membuat DFD Level 2 Modul-Modul Pada DFD Level 1 (satu) dapat di *breakdown* menjadi DFD Level 2 (dua). Modul mana saja yang harus di *breakdown* lebih detail tergantung pada tingkat ke detailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di *breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di



breakdown.

4. Membuat DFD Level 3 dan seterusnya DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.3.2 Pengertian *Flowchart*

Menurut Arief dkk, dalam Putri, dkk (2022:9), menjelaskan bahwa “*Flowchart* merupakan diagram alir yang disajikan secara sistematis dengan tampilan grafis yang menggambarkan suatu proses dan logika dari kegiatan penanganan informasi yang memuat urutan-urutan atau langkah-langkah prosedur pada suatu program yang digunakan dalam penyelesaian masalah untuk dipelajari dan di evaluasi lebih lanjut”.

Sedangkan menurut Dianta (2021:10), “*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program”.

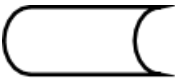
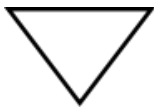
Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Flowchart* merupakan diagram alir yang menggambarkan secara grafik langkah-langkah atau urutan-urutan prosedur dari suatu program.

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

Tabel 2.2 Simbol-Simbol *Flowchart*

No	Simbol	Nama Simbol	Keterangan
1.		Dokumen	Dokumen atau laporan, dapat berupa hasil tulisan tangan atau cetakan komputer.
2.		Input /Output	Proses <i>input</i> atau <i>output</i> data, representasi jurnal atau buku besar.

Lanjutan **Tabel 2.2** Simbol-Simbol *Flowchart*

3.		<i>Predefined Process</i>	Untuk menyatakan penyediaan tempat penyimpanan untuk memberi harga awal.
4.		<i>Proses</i>	Pelaksanaan pemrosesan yang dilakukan oleh komputer.
5.		Proses Manual	Data disimpan secara <i>online</i> melalui media yang dapat diakses secara langsung.
6.		<i>Disk Storage</i>	Data disimpan secara <i>online</i> melalui media yang dapat diakses secara langsung.
7.		<i>Off-line Storage</i>	File dokumen yang disimpan secara manual.
8.		Arus dokumen / arus pemrosesan	Arus pemrosesan atau arus dokumen, arus yang normal berada dibawah dan mengarah.

Sumber: Dianta, (2021:11-12)

2.3.3 Pengertian *Blockchart*

Menurut (Kristanto dalam Hidayah et al., 2021), menjelaskan bahwa, “*Blockchart* adalah pemodelan untuk proses, masukan, keluaran, dan transaksi dengan menggunakan simbol tertentu. *Blockchart* berfungsi memudahkan pengguna untuk memahami alur sistem yang ada”.

Berdasarkan definisi diatas, maka penulis menyimpulkan bahwa *Blockchart* merupakan diagram pemodelan yang berfungsi untuk proses, masukan, keluaran, ataupun sebuah transaksi dengan menggunakan simbol-simbol yang telah ditentukan.

Adapun simbol-simbol yang sering digunakan dalam *blockchart* dapat

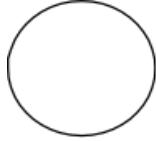
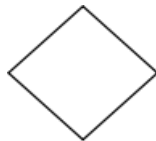


dilihat pada tabel berikut ini

Tabel 2.3 Simbol-simbol *Blockchart*

No	Simbol	Keterangan
1.		Dokumen atau laporan, dapat berupa hasil tulisan tangan atau cetakan komputer.
2.		Proses Manual.
3.		Multi Dokumen.
4.		Proses yang dilakukan oleh <i>computer</i> .
5.		Menandakan dokumen yang diarsipkan (arsip manual).
6.		Data penyimpanan (data storage).
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.

Lanjutan Tabel 2.3 Simbol-simbol *Blockchart*

9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (<i>Decision</i>).
12.		Layar peraga (monitor).
13.		Pemasukan data secara manual.

Sumber: (Kristanto dalam Hidayah et al., 2021)

2.3.4 Pengertian *Entity Relationship Diagram* (ERD)

Menurut Pratiwi, dkk (2020:9), menjelaskan bahwa “*Entity Relationship Diagram* (ERD) adalah suatu diagram yang menggambarkan data dalam konteks entitas dan hubungannya. ERD digunakan untuk membuat suatu model *database*”.

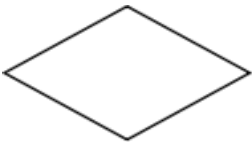
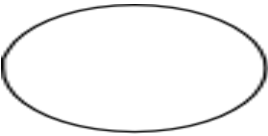
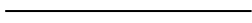
Sedangkan menurut (Simarmata dan Paryudi dalam Rifai dan Yuniar, 2019), “*Entity Relationship Diagram* (ERD) merupakan alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Entity Relationship Diagram* (ERD) merupakan diagram yang digunakan untuk pemodelan data dalam konteks entitas dan hubungannya dalam suatu *database*.



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

Tabel 2.4 Simbol-Simbol ERD

No	Simbol	Keterangan
1.	Entitas 	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
2.	Relasi 	Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda.
3	Atribut 	Atribut berfungsi mendeskripsikan karakter entitas (Atribut yang berfungsi sebagai <i>key</i> diberi garis bawah).
4	Garis 	Proses yang dilakukan oleh <i>computer</i> .

Sumber: Putri (2022:82)

2.3.5 Pengertian Kamus Data

Menurut Raharjo (2021:2), menjelaskan bahwa “Kamus data (*Data dictionary*) adalah sistem manajemen *database* mini yang mengelola meta data. Ini adalah gudang informasi tentang *database* yang mendokumentasikan elemen data dari *database*. Kamus data merupakan bagian integral dari sistem manajemen *database* dan menyimpan metadata atau informasi tentang *database*, nama atribut dan definisi untuk setiap tabel dalam *database*”.



Sedangkan menurut (Antares et al., 2020), “Kamus data (*Data dictionary*) adalah sebuah daftar yang terorganisasi dari elemen data yang berhubungan dengan sistem, dengan definisi yang tepat dan teliti sehingga pemakai dan analis sistem akan memiliki pemahaman yang umum mengenai *input*, *output*, komponen penyimpanan, dan bahkan kalkulasi *intermediate*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa Kamus data (*Data dictionary*) merupakan sebuah sistem atau daftar yang mengelola meta data dari sebuah *database*. Kamus data ini berisi informasi tentang elemen-elemen data dalam *database*, seperti nama atribut dan definisi untuk setiap *table*. Dengan adanya kamus data, *user* (Pemakai) dapat dengan mudah memahami struktur dan konten dari *database* yang digunakan.

Kamus data biasanya berisi hal-hal sebagai berikut:

- a) Nama-nama dari data
- b) Digunakan pada proses-proses yang terkait data
- c) Deskripsi data
- d) Informasi tambahan seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut.

Tabel 2.5 Simbol Kamus Data

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

Sumber: Raharjo (2021:3)



2.4 Teori Program

2.4.1 Pengertian *Visual Studio Code*



Gambar 2.2 Logo *Visual Studio Code*

(Sumber: www.deviantart.com)

Menurut (Hendri & Asep, 2021), “*Visual Studio Code* adalah *editor source code* atau *text editor* yang dikembangkan oleh *microsoft* untuk *windows*, *linux* dan *macos* yang mendukung program bahasa *PHP*”.

Sedangkan menurut Salamah dalam Ningsih et al., (2022), 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 bahasa pemrograman *PHP*, *Javascript*, *Typescript*, dan *Node.js*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Visual Studio Code* adalah sebuah teks editor yang ringan dan handal yang dikembangkan oleh *microsoft* yang tersedia untuk berbagai sistem operasi, termasuk *linux*, *macOS*, dan *Windows*. Dan mendukung secara langsung berbagai bahasa pemrograman seperti *PHP*, *Javascript*, *Typescript*, dan *Node.js*.

2.4.2 Pengertian *Hypertext Markup Language (HTML)*

Menurut Abdulloh (2023:7), menjelaskan bahwa “*HTML* merupakan singkatan dari *Hyper Text Markup Language*, yaitu bahasa standar *web* yang dikelola penggunaanya oleh *W3C (World Wide Web Consortium)* berupa tag-tag



yang menyusun setiap elemen dari *Website*. HTML berperan sebagai penyusun struktur halaman *website* yang menempatkan setiap elemen *website* sesuai *layout* yang diinginkan”.

Sedangkan menurut Muthohir (2021:6), “HTML atau *Hyper-Text Markup Language* adalah bahasa utama dalam membuat *website*. Lazimnya semua halaman *web* ditulis menggunakan variasi HTML. Dengan HTML, pengembang *website* dapat memastikan bahwa *text*, gambar dan multimedia dapat menyatu dengan elemen lainnya ketika dijalankan di *browser*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa HTML adalah bahasa standar yang digunakan untuk membuat struktur halaman *website* menggunakan tag-tag dari setiap elemen *website* sesuai dengan *layout* yang diinginkan. HTML memungkinkan pengembang untuk menyatukan teks, gambar, dan multimedia dengan elemen lainnya sehingga dapat ditampilkan dengan baik di *browser*.

Tag adalah kode yang digunakan untuk memoles (*mark-up*) teks menjadi file HTML. Setiap tag diapit dengan tanda kurung runcing. Ada tag pembuka dan tag penutup, yang membedakan tag penutup ditandai dengan tanda garis miring (*Slash*) di depan awal tulisannya. Dokumen HTML merupakan dokumen yang disajikan dalam *web browser* dan biasanya hanya untuk menampilkan informasi maupun *interface*.



Gambar 2.3 Logo HTML

(Sumber: www.flaticon.com)



2.4.3 Pengertian *Cascading Style Sheets* (CSS)

Menurut Sa'ad (2023:2), mendefinisikan bahwa “*Cascading Style Sheets* (CSS) merupakan bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga lebih terstruktur dan seragam. CSS dikembangkan oleh *Word Wide Web Consortium* (W3C) dan menjadi bahasa standar dalam pembuatan *web*”.

Sedangkan menurut Abdulloh (2023:50), “CSS merupakan singkatan dari *Cascading Style Sheet*, yaitu dokumen *web* yang berfungsi mengatur elemen HTML dengan berbagai *property* yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa CSS adalah bahasa pemrograman *web* yang digunakan untuk mengendalikan tampilan dan gaya dari beberapa komponen dalam sebuah halaman *website*. CSS memungkinkan pengembang *website* untuk membuat halaman *website* lebih terstruktur dan seragam dengan mengatur elemen-elemen HTML menggunakan berbagai properti yang tersedia.

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna *table*, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse-over*, spasi antar paragraf, spasi antar teks, *margin* kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan *format* yang berbeda.



Gambar 2.4 Logo CSS

(Sumber: www.flaticon.com)



Casading Style Sheet (CSS) digunakan untuk membuat *web* menjadi lebih rapi dan terstruktur. Adapun contoh skrip dari *CSS* adalah sebagai berikut.

```
<div style= "font-weight:bold">
    <p> Contoh Paragraf dalam DIV dengan style font weight : bold </p>
    <span> Contoh SPAN dalam DIV dengan style font-weight : bold </span>
</div>
<div>
    <p> Contoh Paragraf dalam DIV tanpa style </p>
```

2.4.4 Pengertian *Hypertext Preprocessor (PHP)*

Bahasa pemrograman berupa *script* yang bersifat *open source* yang digunakan untuk membuat halaman *website* untuk menghasilkan isi *web* yang sesuai dengan permintaan *client*.

Menurut Abdulloh (2023:160), mendefinisikan bahwa “*PHP* merupakan singkatan dari *Hypertext Preprocessor*, yaitu bahasa pemrograman *web* yang dapat disisipkan dalam skrip *HTML* dan bekerja di sisi *server*. Tujuan dari bahasa ini adalah membantu para pengembangan *web* untuk membuat *web* dinamis dengan cepat”.



Gambar 2.5 Logo *PHP*

(Sumber: www.flaticon.com)



Sedangkan menurut Hidayatullah (2021:257), “*PHP* merupakan singkatan dari *Hypertext Preprocessor* adalah *scripting* yang bersifat *open source*, sangat cocok digunakan untuk pengembangan *web* dan dapat disematkan ke dalam *HTML*”

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *PHP* merupakan bahasa *scripting open source* yang sangat cocok digunakan dalam pengembangan *web*. Dengan menggunakan *PHP*, pengembang dapat membuat halaman *web* yang responsif dan berinteraksi dengan pengguna secara dinamis.

2.4.4.1 Sintaks Dasar *Hypertext Preprocessor (PHP)*

Abdulloh (2023:161), menjelaskan bahwa skrip *PHP* dituliskan di antara tanda `<?PHP` dan `?>` yang memisahkan skrip *PHP* dengan skrip lainnya. Satu file *PHP* dapat berisi full skrip *PHP* atau dapat disisipkan diantara skrip lain seperti *HTML*, *CSS* maupun *Javascript*. Namun, file yang berisi skrip *PHP* wajib disimpan dengan ekstensi `*.PHP` dan disimpan di *server* (Folder *htdocs* atau *www*).

Berikut contoh penulisan skrip *PHP* di dalam skrip *HTML*.

```
<!DOCTYPE HTML>
<html>
<head>
    <title> Belajar PHP </title>
</head>
<body>
<?PHP
    Echo “Belajar PHP untuk Pemula”;
?>
</body>
</html>
```



2.4.4.2 Tipe Data *Hypertext Preprocessor (PHP)*

Tipe data merupakan jenis dari suatu data yang akan di proses oleh bahasa pemrograman.

Huda (2020:34), menjelaskan beberapa tipe data dalam *PHP*, yaitu:

- 1) **Integer** merupakan tipe data yang berguna untuk menyimpan bilangan bulat. *Range* bilangan *integer* adalah antara -2.147.483.647 sampai dengan 2.147.483.647.
- 2) **Double Floating** adalah tipe data yang berguna untuk menyimpan bilangan desimal. *Range* bilangan *floating point* antara 1e308 sampai dengan 1e308.
- 3) **Boolean** adalah tipe data yang paling sederhana, berupa *TRUE* dan *FALSE*.
- 4) **String** adalah tipe data yang terdiri dari kata, bias berupa kata tunggal maupun kalimat. Penulisan *string* harus diapit dengan tanda petik, baik berupa petik tunggal (‘ ... ’) maupun petik ganda (“ ... ”).
- 5) **Objek** adalah tipe data dibuat dengan tujuan agar para *programmer* terbiasa dengan OOP. Tipe data ini bisa berupa bilangan.
- 6) **Array** merupakan **Tipe Compound Primitif**, terdapat pada bahasa pemrograman lain.
- 7) **Null** adalah tipe data yang tidak memuat apapun. Setiap *variable* yang diset menjadi tipe data *null*, ini akan menjadikan variabel tersebut kosong.
- 8) **Resources** tipe data spesial yang satu ini dikhususkan untuk menyimpan *resources*, sumber atau alamat.

2.4.5 Pengertian *My Structured Query Language (MySQL)*

Menurut Abdulloh (2023:130), menjelaskan bahwa “*MySQL* merupakan sistem manajemen basis data *relasional* (RDBMS) yang banyak digunakan untuk penyimpanan data pada aplikasi berbasis *web*. Bersama *PHP*, *MySQL* pernah sangat populer, sehingga hampir semua aplikasi *web* dibuat dengan menggunakan *PHP* sebagai bahasa pemrogramannya dan *SQL* sebagai *Database*-nya”.



Gambar 2.6 Logo MySQL

(Sumber: www.logo.wine)

Sedangkan Menurut Arief dalam Sa'ad (2023:3), “MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa MySQL adalah sistem manajemen basis data relasional (RDBMS) yang digunakan untuk penyimpanan dan pengelolaan data dalam aplikasi berbasis *web*. Bersama dengan *PHP*, MySQL telah menjadi kombinasi yang umum digunakan dalam pengembangan aplikasi *web*, di mana *PHP* digunakan sebagai bahasa pemrograman dan MySQL sebagai *database server*.

Berikut beberapa keistimewaan yang dimiliki MySQL antara lain:

- a. Portabilitas yaitu MySQL dapat berjalan secara stabil pada berbagai macam sistem operasi.
- b. *Open Source* yaitu MySQL di distribusikan secara gratis.
- c. *Multi User* yaitu MySQL dapat digunakan dengan banyak *User* tanpa memiliki masalah.
- d. *Performance Turning* yaitu MySQL memiliki tipe kolom yang sangat kompleks.
- e. Perintah dan fungsi yaitu MySQL operator dan fungsi yang secara penuh perintah *select* dan *where* dalam *query*.
- f. Keamanan yaitu MySQL memiliki beberapa lapisan keamanan dengan



sistem perizinan yang detail serta sandi yang terenkripsi.

- g. Skalabilitas dan pembatasan yaitu *MySQL* mampu menangani *database* berskala besar dengan jumlah *record* lebih dari 50 juta, *table* lebih dari 60 ribu dan baris lebih dari 5 milyar.
- h. Konektivitas yaitu *MySQL* dapat melakukan koneksi dengan klien dengan menggunakan *TCP/IP*, *UNIX* dan *Named Pipes*.
- i. Lokalisasi yaitu *MySQL* mendeteksi kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa meskipun bahasa Indonesia belum masuk didalamnya.
- j. Antarmuka yaitu *MySQL* memiliki antarmuka yang mendukung berbagai bahasapemrograman yang menggunakan fungsi *API*.
- k. Klien dan peralatan yaitu *MySQL* dilengkapi dengan berbagai *tool* yang dapat digunakan secara *online*.
- l. Struktur *table* yaitu *MySQL* memiliki struktur *table* yang *fleksibel* dalam menangani *alter table* jika dibandingkan dengan bahasa pemrograman lainnya

2.4.6 Pengertian *MySQL Front*



Gambar 2.7 Logo *MySQL Front*

(Sumber: www.logo.wine)



Menurut (Istiyanto & Novianti, 2019) menjelaskan bahwa “*MySQL Front* merupakan *software* yang digunakan untuk memudahkan dalam *me-manage database* yang dibuat, baik dalam penambahan tabel, *record*, dan *field* maupun menghapus dan mengedit *database* yang ada”.

Sedangkan menurut (Sunarti & Suryadi, 2018) “*MySQL-Front* merupakan salah satu *software* yang dibuat untuk antarmuka *database MySQL*. *MySQL-Front* memungkinkan untuk mengelola *database MySQL* dengan mudah melalui antar muka *windows*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *MySQL Front* merupakan perangkat lunak yang dirancang untuk mempermudah manajemen basis data *MySQL* melalui antarmuka grafis yang *user-friendly*. Perangkat lunak ini memungkinkan pengguna untuk menambah, mengedit, dan menghapus tabel, *record*, dan *field*, serta mengelola *database MySQL* secara keseluruhan melalui antarmuka *windows*.

2.4.7 Pengertian XAMPP

Menurut (Hendri & Asep, 2021), menjelaskan bahwa ”Nama XAMPP sendiri merupakan singkatan dari XAMPP (Empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. XAMPP adalah paket program *web* lengkap yang dapat dipakai untuk pemrograman *web*, khususnya *PHP* dan *MySQL*. Fungsi XAMPP sendiri adalah sebagai *server* yang berdiri sendiri (*Local host*)”.



Gambar 2.8 Logo XAMPP

(Sumber: www.pngwing.com)



Sedangkan menurut (Anggraini et al., 2020), “XAMPP adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemrograman *web*, khususnya *PHP* dan *MySQL*. Fungsinya adalah sebagai *server* yang berdiri sendiri (*Localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL Database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa XAMPP adalah paket program *web* lengkap yang digunakan untuk belajar dan mengembangkan pemrograman *web*, terutama *PHP* dan *MySQL*. Nama "XAMPP" sendiri merupakan singkatan dari X (Empat sistem operasi), *Apache*, *MySQL*, *PHP*, dan *Perl*. Fungsi utama XAMPP adalah sebagai *server* yang berdiri sendiri (*Local host*), yang terdiri dari *Apache HTTP server*, *MySQL Database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP*.

2.4.8 Pengertian *Framework CodeIgniter*

2.4.8.1 *Framework*

Menurut Hidayatullah (2021:343), menjelaskan bahwa “*Framework* merupakan kerangka kerja berupa sekumpulan perintah/program dasar di mana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa harus membuat program dari awal”.

Sedangkan menurut (Sallaby & Kanedi, 2020), “*Framework* adalah kumpulan intruksi-intruksi yang dikumpulkan dalam *framework* dan *function-function* dengan fungsi masing-masing untuk memudahkan *developer* dalam memanggilnya tanpa harus menuliskan *syntax* program yang sama berulang-ulang serta dapat menghemat waktu”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *framework* adalah kerangka kerja yang terdiri dari sekumpulan perintah atau program dasar yang disusun dalam *framework* dan *function*. Perintah dasar tersebut dapat digunakan ulang untuk menyelesaikan masalah yang lebih kompleks, sehingga memudahkan pengembang dalam pembuatan aplikasi baru atau aplikasi



yang lebih kompleks tanpa perlu menulis program dari awal.

2.4.8.2 CodeIgniter



Gambar 2.9 Logo CodeIgniter

(Sumber: www.pngwing.com)

Menurut Hidayatullah (2021:349), Menjelaskan bahwa “CodeIgniter adalah *framework* berbasis *PHP* oleh karena itu wajib untuk menginstal terlebih dahulu *web server* misalnya *Apache*, *Tomcat*, *Microsoft Windows Server*, dan *web server* lainnya”.

Sedangkan menurut (Sallaby & Kanedi, 2020), “CodeIgniter adalah Sebuah *Framework PHP* yang bersifat *open source* dan menggunakan metode MVC (*Model*, *View*, *Controller*) untuk memudahkan *developer* atau *programmer* dalam membangun sebuah aplikasi berbasis *web* tanpa harus membuatnya dari awal”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *codeigniter* adalah sebuah *framework PHP* yang bersifat *open source* dan menggunakan metode MVC (*Model*, *View*, *Controller*) dalam pengembangan aplikasi *web*. Dalam penggunaannya, *codeigniter* membutuhkan instalasi *web server* seperti *Apache*, *Tomcat*, *Microsoft Windows Server*, atau *web server* lainnya karena *framework* ini berjalan di atas lingkungan *server*.

CodeIgniter pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc. (www.ellislab.com), sebuah perusahaan yang memproduksi *CMS* (*Content Management System*) yang cukup handal, yaitu *Expression Engine*



(www.expressionengine.com). Saat ini, *codeigniter* dikembangkan dan dimaintain oleh *Expression Engine Development Team*.

Adapun beberapa keuntungan menggunakan *CodeIgniter*, diantaranya:

a. Gratis

CodeIgniter berlisensi di bawah *Apache BSD open source*.

b. Ditulis menggunakan *PHP 4*

Meskipun *codeigniter* dapat berjalan di *PHP 5*, namun sampai saat ini kode program *codeigniter* masih dibuat dengan menggunakan *PHP 4*.

c. Berukuran kecil

Ukuran *codeigniter* yang kecil merupakan keunggulan tersendiri dibandingkan dengan *framework* lain yang berukuran besar.

d. Menggunakan konsep *MVC27*

CodeIgniter menggunakan konsep *MVC* yang memungkinkan pemisahan *layer application-logic* dan *presentation*.

e. *URL* yang sederhana

Secara default, *URL* yang dihasilkan *CodeIgniter* sangat bersih dan *Search Engine Friendly (SEF)*.

f. Memiliki paket *library* yang lengkap

CodeIgniter mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis *web*, misalnya mengakses *database*, mengirim *email*, memvalidasi *form*, menangani *session* dan sebagainya.

g. *Extensible*

Sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper*, atau dengan menggunakan *hooks*.

h. Tidak memerlukan *template engine*

Meskipun *codeigniter* dilengkapi dengan *template parser* sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan kita untuk menggunakannya.

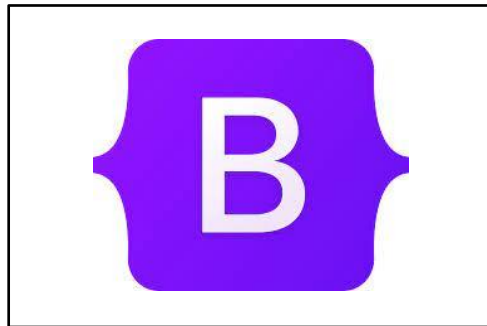
i. Dokumentasi lengkap dan jelas



Dari sekian banyak *framework*, *CodeIgniter* adalah satu-satunya *Framework* dengan dokumentasi yang lengkap dan jelas.

- j. Komunitas-komunitas *codeigniter* saat ini berkembang pesat. Salah satu komunitasnya bisa dilihat di (<https://forum.CodeIgniter.com/>).

2.4.9 Pengertian *Bootstrap*



Gambar 2.10 Logo *Bootstrap*

(Sumber: <https://icons.getBootstrap.com/>)

Menurut Huda (2020:10), menjelaskan bahwa “*Bootstrap* merupakan *Framework HTML, CSS, Javascript* populer untuk membangun situs *web* yang *responsive*. Dengan *Bootstrap*, *developer website* dapat membuat *responsive Website* dan dapat berjalan sempurna pada *browser-browser* populer seperti *Chrome, Firefox, Opera*, dan *Internet Explorer* serta pada perangkat-perangkat *handphone* maupun *PC desktop*”.

Sedangkan menurut (Cristian dan Hesinto dalam Sari et al., 2021), “*Bootstrap* adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *Bootstrap* adalah *template desain web* dengan fitur *plus*. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Bootstrap* adalah sebuah *framework* yang populer dalam pengembangan situs *web*. *Bootstrap* menyediakan kumpulan alat dalam bentuk *HTML, CSS*, dan *Javascript* yang memungkinkan pengembang *website* untuk membangun situs *web* yang



responsif dan dapat berjalan dengan baik di berbagai jenis perangkat dan *browser*, termasuk perangkat *mobile* dan *desktop*.

2.4.10 Pengertian Javascript



Gambar 2.11 Logo Javascript

(Sumber: www.flaticon.com)

Menurut Muthohir (2021:68), menjelaskan bahwa “*Javascript* merupakan bahasa pemrograman *web* yang berjalan disisi *client/browser*. *Javascript* biasa digunakan untuk memanipulasi *element-element HTML* dan menambahkan *style* secara otomatis atau lebih sederhananya membuat dokumen *HTML* menjadi lebih interaktif”.

Sedangkan menurut (Ratnawati dalam Risti, 2022), “*Javascript* adalah bahasa *script* yang biasa diletakkan bersama kode *HTML* untuk menentukan suatu aksi. *Javascript* digunakan pada pemrograman *web* untuk meningkatkan atau menambah kemudahan pengguna ketika mengakses halaman *web*”.

Berdasarkan beberapa definisi diatas, maka penulis menyimpulkan bahwa *Javascript* adalah sebuah bahasa pemrograman *web* yang berjalan di sisi klien atau *browser*. *Javascript* digunakan untuk memanipulasi elemen-elemen *HTML* dan menambahkan gaya secara otomatis, yang pada akhirnya membuat dokumen *HTML* menjadi lebih interaktif.