



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Di dalam teori umum ini, meliputi penjelasan dari Pengertian Komputer, Perangkat Lunak, Data, Basis Data (*Database*) dan Metode Pengembangan Sistem *Waterfall*.

2.1.1 Pengertian Komputer

Menurut Fuori (dalam Daliuwa dan Didipu, 2022: 137) mengatakan bahwa “Pengertian komputer adalah suatu alat yang dapat memproses data serta dapat melakukan perhitungan secara besar, cepat dan tepat termasuk menghitung secara aritmatika serta bisa mengoperasikan secara logika, dan tidak ada campur tangan manusia”.

Menurut Hanafri *et. al.*, (2019:88), “Komputer merupakan alat untuk mengolah data sesuai perintah yang sudah dirumuskan secara cepat dan tepat, serta diorganisasikan supaya secara otomatis menerima dan menyimpan data berdasarkan intruksi-intruksi yang telah tersimpan di dalam memori”.

Kesimpulan dari kedua pengertian tersebut adalah bahwa komputer adalah suatu alat yang dapat mengolah data secara cepat, tepat, dan otomatis berdasarkan instruksi-instruksi yang telah diatur sebelumnya.

2.1.2 Pengertian Perangkat Lunak

Menurut Ali (dalam Sudarso, 2022: 5) mengatakan bahwa “Perangkat lunak aplikasi merupakan program yang ditujukan untuk menyelesaikan suatu permasalahan dalam aplikasi yang tertentu yang sudah dibuat oleh pabrik pembuat perangkat lunak aplikasi. Program aplikasi dibuat dengan menggunakan perangkat lunak bahasa (*language software*)”.

Menurut Hanafri *et. al.*, (2019:88), “Perangkat lunak adalah sekumpulan data elektronik yang berupa program. Program adalah sekumpulan intruksi yang digunakan untuk mengendalikan perangkat keras komputer”.



Dari pengertian di atas dapat disimpulkan bahwa perangkat lunak aplikasi adalah program yang dibuat untuk menyelesaikan suatu permasalahan dalam aplikasi tertentu.

2.1.3 Pengertian Data

Menurut Sutarman (dalam Ma'mur, *et al.*, 2019: 378) "Data adalah fakta dari suatu pernyataan yang berasal dari kenyataan, dimana pernyataan tersebut merupakan hasil pengukuran atau pengamatan".

Menurut Ladjamudin (dalam Hasan, 2020: 47), data merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

Berdasarkan dari definisi di atas maka peneliti menyimpulkan bahwa data merupakan suatu informasi yang akurat sehingga menghasilkan informasi dan diolah atau diproses menentu suatu angka.

2.1.4 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 (*redudansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik".

Menurut Kristy (dalam Yudanto, 2021:290) "*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. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: objek yang diwakili suatu basis data, dan



hubungan di antara objek tersebut. Ada banyak cara untuk mengorganisasi skema, atau memodelkan struktur basis data: ini dikenal sebagai model basis data atau model 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 merupakan fakta atau nilai (*value*) yang tercatat atau merepresentasikan deskripsi dari suatu objek. Data yang merupakan fakta yang tercatat dan selanjutnya dilakukan pengolahan (proses) menjadi bentuk yang berguna atau bermanfaat bagi pemakainya akan membentuk apa yang disebut informasi. 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 dipakai oleh sejumlah pengguna dalam waktu yang bersamaan. Sering dinamakan 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 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.

Suatu bangunan basis data memiliki jenjang sebagai berikut:

- a. Basis data/*Database*, merupakan kumpulan dari bermacam-macam tipe record yang memiliki hubungan antar *record*.
- b. Berkas/*File*, merupakan sekumpulan data rekaman yang berkaitan dengan suatu objek.
- c. *Record*, merupakan sekumpulan *field*/atribut/*data item* yang saling berhubungan dengan obyek tertentu dengan panjang record yang tetap, semua *field* dalam *record* memiliki ukuran yang tetap. Variabel panjang *record*, *field-field* dalam *record* dapat memiliki ukuran yang berbeda (metode penandaan yang digunakan adalah: penanda akhir *record*, indikator panjang, dan tabel posisi *record*).
- d. *Field*, menyatakan data terkecil yang memiliki makna. *Field* merupakan implementasi dari suatu atribut data. *Field* merupakan unit terkecil dari data yang berarti (*meaningful data*) yang disimpan dalam suatu *file* atau basis data lain untuk *field* yaitu elemen data, kolom item, dan atribut. Contoh *field* yaitu nama, alamat, telepon, dan jenis kelamin.
- e. *Byte*, adalah bagian terkecil yang dialamatkan dalam memori. *Byte* merupakan kumpulan bit yang secara konvensi terdiri atas kombinasi delapan bit yang menyatakan sebuah karakter dalam memori (1 *byte*= 1 karakter) .
- f. *Bit*, adalah sistem biner yang terdiri atas dua macam nilai, yaitu 0 dan 1. sistem biner merupakan dasar yang dapat digunakan untuk komunikasi antara manusia dan mesin, yang merupakan rangkaian komponen elektronik dan hanya dapat membedakan 2 macam keadaan, yaitu ada tegangan dan tidak ada

tegangan yang masuk ke rangkaian tersebut.

Dalam pengembangan suatu sistem informasi, 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. Data akan dapat saling berelasi dengan mengimplementasikan basis data.
4. Basis data mengurangi duplikasi data (*data redundancy*).
5. Basis data dapat mengurangi pemborosan tempat penyimpanan luar.

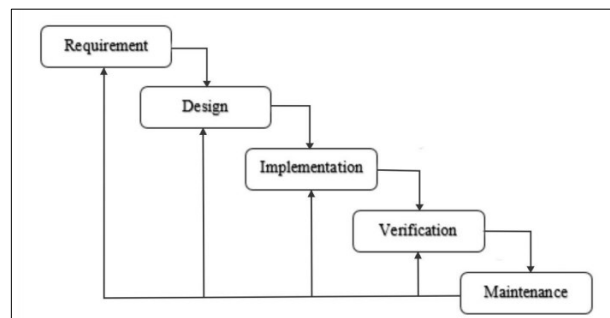
Dari pernyataan di atas dapat disimpulkan bahwa basis data adalah kumpulan data terstruktur yang disimpan secara sistematis di dalam suatu sistem komputer.

2.1.5 Metode Pengembangan Sistem Waterfall

Menurut Pressman (dalam Wahid, 2020: 2) “Metode air terjun atau yang sering disebut metode *waterfall* seing dinamakan siklus hidup klasik (*classic life cycle*), nama model ini sebenarnya adalah “*Linear Sequential Model*” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modelling*), konstruksi (*contruction*), serta penyerahan sistem ke para pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan”.

2.1.5.1. Tahapan Metode Waterfall

Tahapan dari metode *waterfall* dapat dilihat pada gambar di bawah ini:



Sumber: Pressman (dalam Wahid, 2020: 2)

Gambar 2.1 Tahapan dalam Model Waterfall

Tahapan-tahapan dalam model ini menurut Pressman (dalam Wahid, 2020: 2) adalah sebagai berikut:

1. *Requirement*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2. *Design*

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit *testing*.

4. *Verification*

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau



nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

5. *Maintenance*

Ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.2 Teori Judul

Dalam teori ini meliputi penjelasan mengenai aplikasi, manajemen, pembayaran, cicilan, dan uang kuliah tunggal.

2.2.1 Pengertian Aplikasi

Menurut Aziz (2018: 72) “Aplikasi berasal dari kata *application* yang artinya penerapan, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju”.

Menurut Sutarman (dalam Nurjaman dan Yasin, 2020: 145) “Aplikasi merupakan program-program yang dibuat oleh suatu perusahaan komputer untuk para pemakai yang beroperasi dalam bidang umum seperti pada pertokoan, komunikasi, penerbangan, perdagangan dan sebagainya”.

Dari kedua pengertian di atas dapat disimpulkan bahwa aplikasi adalah program siap pakai yang dibuat untuk melaksanakan suatu fungsi bagi pengguna.

2.2.2 Pengertian Laporan

Menurut Hapsari *et al.*, dalam Prasetyo *et al.*, (2020: 36) “Laporan keuangan merupakan produk akhir dari serangkaian pencatatan dan pengikhtisaran data transaksi bisnis”.

Menurut Wikipedia (2022) “Laporan adalah suatu bentuk penyampaian berita, keterangan, pemberitahuan ataupun pertanggungjawaban baik secara lisan maupun secara tertulis”.

Dari pengertian di atas dapat disimpulkan bahwa laporan menyajikan informasi tentang aspek data suatu entitas bisnis.



2.2.3 Pengertian Pembayaran

Menurut Kamus Besar Bahasa Indonesia (dalam Rusman, *et al.*, 2022: 201) “Sistem pembayaran atau pembayaran yaitu sistem yang mencakup seperangkat aturan, lembaga dan proses yang digunakan untuk melakukan pengalihan dana untuk melengkapi suatu keharusan yang timbul dari kegiatan ekonomi”.

Menurut Evy (dalam Astriyani, *et al.*, 2020: 107) “Ada dua pengertian pembayaran yaitu pengertian secara sempit dan yuridis teknis. Pengertian pembayaran dalam arti sempit, adalah pelunasan utang oleh debitur kepada kreditur. Pembayaran seperti ini dilakukan dalam bentuk uang atau barang. Namun, pengertian pembayaran dalam arti yuridis tidak hanya dalam bentuk uang atau barang, tetapi juga dalam bentuk jasa, seperti jasa dokter dan lain-lain”.

Dari pernyataan diatas, dapat diambil kesimpulan bahwa sistem pembayaran atau pembayaran merupakan suatu sistem yang melibatkan aturan, lembaga, dan proses untuk melakukan pengalihan dana guna memenuhi kewajiban yang muncul dari kegiatan ekonomi.

2.2.4 Pengertian Cicilan

Menurut Kamus Besar Bahasa Indonesia (dalam Rizky, 2021: 2), Angsuran yaitu uang yang dipakai untuk mengangsur atau cicilan. Angsuran adalah uang yang dipakai untuk diserahkan sedikit demi sedikit atau tidak sekaligus, seperti untuk pembayaran hutang, pajak dan sebagainya.

Menurut Wikipedia bahasa Indonesia, ensiklopedia bebas (2023), Cicilan atau Angsuran adalah besar nominal yang harus dibayar oleh debitur atau peminjam ke kreditur atau pemberi pinjaman dalam jangka waktu yang telah disepakati untuk melunasi suatu pembayaran, pembelian, atau utang atas barang atau jasa. Besar angsuran mencakup sisa pokok utang dan bunga pinjaman.

Berdasarkan pernyataan di atas dapat disimpulkan bahwa angsuran atau cicilan adalah pembayaran uang secara berkala dalam jangka waktu tertentu.

2.2.5 Pengertian Uang Kuliah Tunggal

Menurut Permendikbud (dalam Asni, *et al.*, 2019: 52) “Uang Kuliah Tunggal (UKT) merupakan sebagian biaya kuliah yang ditanggung setiap mahasiswa berdasarkan kemampuan ekonominya”.

Menurut Karim *et al.*, dalam Ginting (2023:5), “Uang kuliah tunggal adalah biaya keseluruhan mahasiswa dalam operasional per semester pada salah satu program studi di perguruan tinggi negeri.”

Berdasarkan pernyataan di atas dapat disimpulkan bahwa Uang Kuliah Tunggal (UKT) adalah biaya keseluruhan mahasiswa yang ditanggung berdasarkan kemampuan ekonomi setiap mahasiswa.

2.3 Teori Khusus

Dalam teori ini meliputi penjelasan diagram konteks, *data flow diagram* (DFD), *flowchart*, *blockchart*, *entity relationship diagram* (ERD), dan kamus data.

2.3.1 Diagram Konteks

Menurut Jogiyanto (dalam Fatimah & Samsudin, 2019: 35) mengatakan bahwa “*Context Diagram* adalah gambaran umum tentang suatu sistem yang terdapat didalam suatu organisasi yang memperlihatkan batasan (*boundary*) sistem, adanya interaksi antara eksternal entity dengan suatu sistem dan informasi secara umum mengalir diantara entity dan sistem”.

Menurut Santoso dan Nurmalina (2017: 88) mengatakan bahwa “Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari *DFD* yang menggambarkan seluruh *input* ke sistem atau output dari sistem yang memberikan gambaran tentang keseluruhan sistem”.

Dari pernyataan di atas dapat disimpulkan bahwa diagram konteks adalah gambaran umum tentang suatu sistem yang memperlihatkan batasan sistem, interaksi dengan entitas eksternal, dan aliran informasi secara umum antara entitas dan sistem.

2.3.2 Data Flow Diagram (DFD)


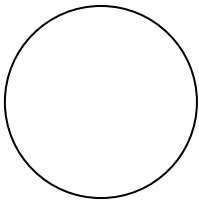
Menurut Sirait dan Seabtian (2019: 3) mengatakan bahwa “*Data Flow Diagram (DFD)* adalah menggambarkan atau menjelaskan langkah-langkah kegiatan yang ada di dalam sebuah sistem, aliran-aliran data dimana komponen-komponen tersebut, dan asal, tujuan dan penyimpanannya dari data tersebut”.

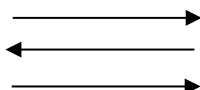
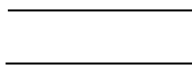
Menurut Kristanto (dalam Soufitri, 2019:241), “*Data Flow Diagram (DFD)* disebut juga dengan Diagram Arus Data (DAD). *DFD* adalah suatu model logika data atau proses yang dibuat untuk menggambarkan: darimana asal data, dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa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

Dari pernyataan di atas dapat disimpulkan bahwa *Data Flow Diagram (DFD)* adalah representasi grafis dari aliran data di dalam suatu sistem.

DFD dipopulerkan Oleh DeMacro & Yordan dan Gane & Sarson dengan menggunakan pendekatan Metoda Analisis Sistem Terstruktur. *DFD* ini merupakan model proses. Model proses merupakan teknik untuk mengorganisasikan dan mendokumentasikan struktur dan alir data di dalam sistem. Dalam laporan ini akan digunakan model DeMarco/Yordan.

Tabel 2.1 Simbol-simbol dalam *Data Flow Diagram (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: Kristanto (dalam Soufitri, 2019:241)

2.3.2.1 Langkah-langkah Pembuatan DFD

1. Pahami Sistem

Memahami sistem yang akan dikembangkan dilakukan dengan cara wawancara kepada bagian yang menggunakan serta memproses data.

2. Menetapkan Batasan Sistem

Tentukan batas sistem yang akan digambar, batasan meliputi data yang masuk dan informasi yang akan keluar dari sistem.

3. Kembangkan Diagram Konteks

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

Diagram dekomposisi digambarkan untuk mempartisi sistem menjadi subsistem. Hal ini dibuat untuk memudahkan kita dalam menggambarkan DFD pada level-level berikutnya.

5. Beri nomor urut pada setiap proses dan membuat diagram secara rinci.


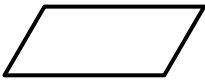


2.3.3 Flowchart

Menurut Indrajani (dalam Budiman et al., 2021: 2186) mengatakan bahwa “*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut. Flowchart dapat digunakan untuk menyajikan kegiatan manual, kegiatan pemrosesan ataupun keduanya”.

Menurut Santoso dan Nurmalina (2017: 86) mengatakan bahwa “*Flowchart* adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek”.

Dari pernyataan di atas dapat disimpulkan bahwa *Flowchart* merupakan representasi grafis dari langkah-langkah dan urutan prosedur suatu program atau algoritma untuk menyelesaikan masalah.

Tabel 2.2 Simbol-simbol *Flowchart*

No	Simbol	Nama	Keterangan
1.		Dokumen	Dokumen atau laporan, dapat berupa hasil tulisan tangan atau cetakan komputer.
2.		<i>Input/Output;</i>	Proses <i>input</i> atau <i>output</i> data, representasi jurnal atau buku besar.
3.		<i>Predefined Process</i>	Untuk menyatakan penyediaan tempat penyimpanan untuk memberi harga awal.
4.		Proses	Pelaksanaan pemrosesan yang dilakukan oleh komputer.




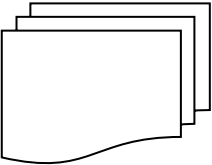
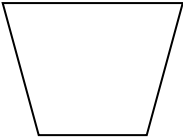

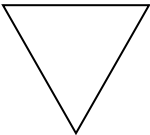
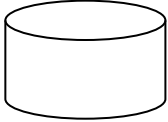
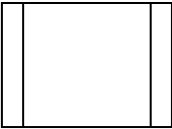
5.		Proses Manual	Pelaksanaan pemrosesan yang dilakukan secara manual.
6.		<i>Disk Storage</i>	Data disimpan secara <i>online</i> melalui media yang dapat diakses secara langsung.
7.		<i>Off-line Storage</i>	File dokumen yang disimpan secara manual.
8.		Arus dokumen / arus pemrosesan	Arus pemrosesan atau arus dokumen, arus yang normal berada dibawah dan mengarah
9.		<i>Communication link</i>	Pengiriman data dari satu lokasi ke lokasi lainnya melalui jalur komunikasi.
10.		<i>Off-page Connector</i>	Menggambarkan keluar atau masuk proses dalam lembar atau halaman yang lain.
11.		<i>On-page connector</i>	Menghubungkan arus pemrosesan di halaman yang sama.
12.		<i>Decision</i>	Menyatakan kondisi yang menghasilkan beberapa kemungkinan atau aksi.
13.		Terminal	Simbol yang menyatakan permulaan atau akhir suatu proses atau program.

Sumber : Budiman, et al (2021: 2186)

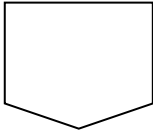
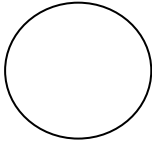

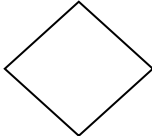

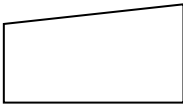
2.3.4 Blockchart

Menurut Kristanto (dalam Noor *et al*, 2018: 22) “*Block Chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *block chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Tabel 2.3 Simbol-simbol *Block Chart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel, berkas atau cetakan.
2.		Multi Dokumen
3.		Proses Manual
4.		Proses yang dilakukan oleh computer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (data storage)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik

Lanjutan Tabel 2.1 Simbol-simbol *Block Chart*

8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
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 (decision)
12.		Layar peraga (monitor)
13.		Pemasukan data secara manual

Sumber: (Noor et al., (2018:22))

2.3.5 Entity Relationship Diagram (ERD)


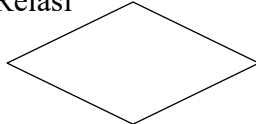
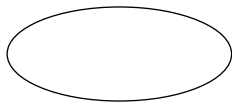

Menurut Sutanta (dalam Putra et al., 2019: 139), “Entity Relationship Diagram (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek”.

Menurut Toledo dan Cushman (dalam Solikin et al., 2018: 143) mengatakan bahwa “Entity Relationship Diagram (ERD) merupakan representasi grafis dari logika database dengan menyertakan deskripsi detail mengenai seluruh entitas (*entity*), hubungan (*relationship*), dan batasan (*constraint*)”.

Dari pernyataan 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 (*entity*), hubungan (*relationship*), dan batasan (*constraint*) dalam suatu sistem *database*. Dengan menggunakan *ERD*, pengguna dapat memvisualisasikan dan merencanakan struktur data serta relasi antar entitas dalam suatu sistem secara jelas dan terorganisir.

Solikin, *et al* (2018: 143) menjelaskan simbol-simbol yang terdapat dalam *ERD*:

Tabel 2.4 Simbol-Simbol pada ERD

No.	Simbol	Deskripsi
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 	<i>Atribut</i> , berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)
4.	Garis 	Garis, sebagai penghubung antara relasi dengan entitas relasi dan entitas dengan atribut

Sumber : Solikin, *et al* (2018: 143)

2.3.6 Kamus Data

Menurut Prasetija dan Syafitri (2022: 23) mengatakan bahwa “Kamus data adalah dokumen yang berisi gambaran terperinci tentang data yang digunakan dalam basis data dan menggambarkan karakteristik semua item dalam basis data, menjelaskan bagian dari metadata dan nilai dari setiap atribut, mendeskripsikan aliran data, mencakup informasi cara catatan diatur, dan diperlukan untuk membantu programmer dalam membuat kode untuk mengakses data”.

Menurut Sukamto dan Shalahuddin (dalam Sirait dan Seabtian, 2019: 3-4) mengatakan bahwa “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Tabel 2.5 Simbol-simbol dalam Kamus Data

No	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	()	Opsional
4.	[]	Memilih salah satu alternatif
5.	**	Komentar
6.	{ } ⁿ	n kali diulang/ bernilai banyak

Sumber: Sukamto dan Shalahuddin (dalam Sirait dan Seabtian, 2019: 3-4)

2.4 Teori Program

Dalam teori ini meliputi penjelasan tentang *visual studio code*, *html*, *css*, *bootstrap*, *xampp*, *mysql*, *php*, *Laravel*, dan *javascript*.



1.4.1 Pengertian *Visual Studio Code*

Menurut Salendah *et al.*, (2022:84), “*Visual studio code* yang biasa disingkat dengan *VSCode* adalah perangkat lunak yang digunakan untuk menuliskan kode-kode atau coding yang dapat diakses pada berbagai sistem operasi seperti *Windows*, *Linux*, maupun *macOS*”.

Menurut Pangestu, *et al* (2021: 1820) mengatakan bahwa “*Visual Studio Code (VS Code)* adalah *editor* teks yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, banyak fitur yang disediakan oleh kode *Visual Studio*, termasuk Intektivitas, Intertasi *Git*, *Debug* dan Perpanjangan fitur. Fitur-fitur ini akan terus tumbuh dengan peningkatan kode studio visual”.

Dari pengertian di atas dapat disimpulkan bahwa *Visual Studio Code (VS Code)* adalah perangkat lunak atau *editor* teks yang digunakan untuk menulis kode atau *coding*.

1.4.2 Pengertian *HTML*

Menurut Abdulloh (dalam Hidayat *et al.*, 2022: 42) mengatakan bahwa “*HTML* singkatan dari *Hyper Text Markup Language*, yaitu skrip yang berupa *tag-tag* untuk membuat dan mengatur struktur *website*”.

Menurut Solihin (dalam Kostaman & Sumaryana, 2018: 124) mengatakan bahwa “*HTML* merupakan singkatan dari *Hyper Text Markup Language* yang merupakan bahasa pemrograman *web* yang memberitahukan peramban *web (web browser)* bagaimana menyusun dan menyajikan konten di halaman *web*”.

Dari pengertian di atas dapat disimpulkan bahwa *HTML* adalah singkatan dari *Hyper Text Markup Language*, yang merupakan bahasa pemrograman *web* berbasis tag untuk membuat dan mengatur struktur serta menyajikan konten di halaman *website*.

1.4.3 Pengertian *CSS*

Menurut Sulistyawan, *et al* (dalam Kostaman dan Sumaryana, 2018: 124), mengatakan bahwa “*CSS* adalah sebuah dokumen yang berdiri sendiri dan dapat dimasukkan dalam kode *HTML* atau sekedar menjadi rujukan oleh *HTML* dalam pendefinisian *style*”.



Menurut Sengkey (dalam Fatimah dan Samsudin, 2019: 36) mengatakan bahwa “*Cascading Style Sheet (CSS)* merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS bukan merupakan bahasa pemrograman. Pada umumnya *CSS* dipakai untuk memformat tampilan halaman web yang dibuat dengan bahasa *HTML* dan *XHTML*”.

Dari pengertian di atas dapat disimpulkan bahwa *css* adalah skrip yang digunakan untuk mengatur tampilan dan gaya dari halaman web yang ditulis dengan menggunakan *HTML*.

1.4.4 Pengertian *Bootstrap*

Menurut Abdulloh (dalam Hidayat et al., 2022: 46) mengatakan bahwa “*Bootstrap* merupakan *framework* *CSS* yang sangat populer di kalangan pecinta pemrograman *Website*. Dengan menggunakan *Bootstrap*, proses desain *Website* tidak dibuat dari nol, sehingga proses desain *Website* lebih cepat dan mudah. Bahkan tanpa kita membuat skrip *CSS* sedikit pun, kita sudah dapat membuat desain *Website* yang bagus”.

Menurut Noviantoro et al., (2022:91), “*Bootstrap* adalah sebuah platform *CSS (Cascading Style Sheet)* yang digunakan untuk merancang *website*. *Bootstrap* merupakan *tool* yang sangat baik untuk digunakan programmer saat membuat tampil sebuah *website*. *CSS*, misalnya, dalam *Bootstrap* menyediakan jenis, tombol, navigasi, dan komponen lainnya, serta *JavaScript*, yang membuat antarmuka perkembangan menjadi lebih mudah dan stabil”.

Dari pernyataan di atas, dapat disimpulkan bahwa *Bootstrap* adalah sebuah *framework CSS (Cascading Style Sheet)* yang digunakan untuk merancang tampilan *website* dan menyediakan *class* serta komponen yang siap digunakan, termasuk jenis, tombol, navigasi, dan komponen lainnya.

1.4.5 Pengertian *XAMPP*

Menurut Yudhanto dan Prasetyo (2019:17), *XAMPP* adalah kompilasi program aplikasi gratis terfavorit di kalangan developer/programmer yang berguna untuk pengembangan *website* berbasis *PHP* dan *MySQL*.



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

Dari pengertian di atas dapat disimpulkan bahwa *XAMPP* adalah sebuah paket perangkat lunak sumber terbuka yang berfungsi sebagai *server* lokal untuk mengembangkan dan menguji situs web secara *offline*.

2.4.5.1. Mengetahui Fungsi *XAMPP*

Menurut Wardhani (2022:95), secara umum, *XAMPP* berfungsi sebagai *web server* yang bisa diakses melalui *localhost* komputer. Aplikasi web yang belum sepenuhnya jadi, bisa dikembangkan dalam server lokal ini. *Developer* bisa menguji seluruh fungsi aplikasi web, menguji coba desain web, menyusun struktur konten, dan hal-hal lain sebelum resmi diluncurkan ke internet. Dengan *XAMPP*, aplikasi web sudah bisa dipresentasikan secara real. Antarmuka, struktur konten, bahkan fungsi yang berjalan di *back-end* cukup identik dengan aplikasi *web* yang asli.

Setelah pengujian di *XAMPP* selesai dilakukan, aplikasi *web* bisa di *publish* secara *online* dengan mengunggahnya di *web hosting*. Fitur-fitur yang ada dalam *XAMPP* sangat memudahkan proses pengujian aplikasi *web*. Selain *developer profesional*, *XAMPP* juga cukup lazim digunakan oleh pelajar atau mahasiswa untuk belajar mengembangkan aplikasi *web*.

2.4.5.2 Bagian Penting dalam *XAMPP* dan Fitur Penunjangnya

a. *Htdocs*

Hasil instalasi *XAMPP* di komputer memiliki sejumlah folder di dalamnya. *Htdocs* merupakan salah satu folder tersebut. Folder *Htdocs* sendiri berfungsi untuk menyimpan seluruh file dan dokumen yang akan menyusun aplikasi *web*. Karena *XAMPP* merupakan server lokal, kapasitas penyimpanan dalam *htdocs* akan menyesuaikan dengan kapasitas hardisk pada PC. Untuk membuka *Htdocs*, hanya perlu membuka folder instalasi *XAMPP* dan foldernya akan terdapat di situ.



b. *PhpMyAdmin*

PhpMyAdmin merupakan program untuk mengelola sistem database pada aplikasi web. *PhpMyAdmin* tidak hanya digunakan pada *XAMPP*, *web hosting* profesional biasanya juga menyediakan program ini. Cara membuka *PhpMyAdmin* di *XAMPP* cukup sederhana, hanya perlu menuliskan URL *http://localhost/phpMyAdmin* di *web browser*.

c. *Control Panel*

Secara umum, *Control Panel* pada *XAMPP* berfungsi untuk mengatur, menjalankan, dan mengelola fitur secara umum. Di *Control Panel* terdapat sejumlah fitur yang bisa diatur, diantaranya:

- a. *Apache Web server* yang digunakan dalam *XAMPP*.
- b. *MySQL* : Database yang digunakan dalam *XAMPP*.
- c. *FileZilla* : *FTP Server* untuk memindahkan file antar komputer pada jaringan lokal.
- d. *Mercury* : Mail Server berfungsi untuk menerima dan mengirim email pada suatu jaringan lokal.
- e. *Tomcat* : Fitur ini berfungsi untuk mempercepat akses data pada halaman aplikasi *website* dengan mendapatkan *Java Server Pages (JSP)* Setiap fitur tersebut terbagi dalam suatu *module*.

Di *Control Panel* juga terdapat sejumlah tombol yang menjalankan fungsi penunjang lain:

- a. *Config* : Tombol ini menuju halaman konfigurasi dasar pada *XAMPP*.
- b. *Netstat* : Untuk mengecek aplikasi yang berjalan di *port default XAMPP*.
- c. *Shell* : Membuka *CMD* bisa menggunakannya untuk melakukan konfigurasi pada *web server* tentu saja dengan *script CMD*.
- d. *Explorer*: Sebuah *shortcut* ke folder instalasi *XAMPP* di *windows explorer*.
- e. *Services* : Untuk menampilkan semua *service* yang pada saat itu sedang berjalan di *background*.
- f. *Help* : Menu bantuan yang mengarahkan pada forum komunitas *XAMPP*.
- g. *Quit* : Tombol untuk menutup *software XAMPP*.



1.4.6 Pengertian *MySQL*

Menurut Enterprise (dalam Hidayat et al., 2022: 45) mengatakan bahwa “*MySQL* adalah *Relational Database Management System (RDBMS)* yang cepat dan mudah digunakan, serta banyak digunakan berbagai kebutuhan. *MySQL* dikembangkan oleh *MySQL AB Swedia*”.

Menurut Zefriyenni dan Santoso (dalam Sukrianto, 2017: 20) mengatakan bahwa “*MySQL* merupakan *software RDBMS (server database)* yang dapat mengelola *database* dengan cepat, dapat menampung data dengan jumlah yang besar, dapat diakses banyak user (*multi-user*) dan dapat melakukan suatu proses sikron atau bersamaan (*multi-threaded*)”.

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

- a. *Relational Database System*.
- b. *Arsitektur Client – Server*.
- c. Mengetahui perintah *Structured Query Language (SQL)* standar.
- d. Mendukung *Sub Select*.
- e. Mendukung *View, Stored Prosedured (SP), Triggers, Replication, Transaction, dan Foreign Key (FK)*.
- f. Tersedia fungsi GIS, bebas diunduh (*download*), stabil, tangguh, fleksibel, dan memiliki keamanan (*security*) yang baik

Dari pengertian di atas 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.

1.4.7 Pengertian *PHP*

Menurut Solichin (dalam Hidayat et al., 2022: 43) mengatakan bahwa “*PHP* merupakan salah satu bahasa pemrograman berbasis *web* yang ditulis oleh dan untuk pengembang *web*”.

Menurut Prahasti et al., (2022:155), “*PHP (PHP: hypertext preprocessor)* adalah suatu bahasa pemrograman yang digunakan untuk menterjemahkan basis



kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang ditambahkan ke *HTML*”.

Dari pernyataan di atas, dapat disimpulkan bahwa *PHP* adalah sebuah bahasa pemrograman dari kode dengan ekstensi yang dapat dimengerti oleh komputer.

1.4.8 Pengertian *Laravel*

Menurut Yusup et al., (2019:217) “*Laravel* merupakan *framework PHP open source* yang dikembangkan oleh Tylor Otwell yang berada di bawah lisensi MIT yang bertujuan untuk mempermudah para developer untuk membuat web dengan sintaks yang sederhana, elegan, ekspresif dan juga menyenangkan”.

Menurut Hermanto (2019:19) “*Laravel* adalah sebuah *framework PHP* yang dirilis di bawah lisensi MIT, dibangun dengan konsep *MVC (model view controller)*. *Laravel* adalah pengembangan *website* berbasis *MVC* yang ditulis dalam *PHP* yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu”.

Dari pernyataan di atas, dapat disimpulkan bahwa *Laravel* merupakan *framework PHP* dengan konsep *MVC* untuk mempermudah para *developer* untuk membuat *web*.

1.4.9 Pengertian *Javascript*

Menurut Sibero (dalam Noviantoro et al., 2022: 91) mengatakan bahwa “*JavaScript* adalah suatu bahasa pemrograman yang dikembangkan untuk dapat berjalan pada *web browser*. Pada awalnya *JavaScript* dikembangkan pada *web browser Netscape* oleh Brenden Eich dengan nama Mocha, kemudian berubah menjadi *Live-Script* dan yang akhirnya menjadi *JavaScript*”.

Menurut Sunyoto (dalam Kostaman dan Sumaryana, 2018: 124) mengatakan bahwa “*Javascript* adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian browser populer seperti *internet explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*”.



Dari pernyataan diatas, dapat diambil kesimpulan bahwa Javascript adalah suatu bahasa pemrograman yang dikembangkan untuk berjalan pada *web browser* dan dapat bekerja di sebagian browser populer seperti *internet explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*.