

BAB II

LANDASAN TEORI

A. Teori-Teori Umum

1. Data

Menurut Jayanti dan Sumiari (2018:1): “Data merupakan catatan atas kumpulan fakta yang mewakili suatu objek”. Sedangkan menurut Ningsih, et al (2022:14):

“Data merupakan fakta yang mewakili suatu objek seperti manusia, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang dapat dicatat dan mempunyai arti yang implisit. Data dicatat atau direkam dalam bentuk angka, huruf, simbol, gambar, bunyi, atau kombinasinya”.

Dari pengertian di atas dapat disimpulkan bahwa data adalah kumpulan fakta yang berbentuk dalam angka, huruf, kata, gambar, simbol, bunyi, dan sebagainya.

2. Informasi

Menurut Negara, et al (2021:13): “Informasi sendiri merupakan sekumpulan data yang sudah diolah hingga memberikan pengetahuan kepada yang membaca atau melihat informasi tersebut”.

Menurut Romindo, et al (2020:30): “Informasi adalah suatu data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat untuk mengambil keputusan pada saat ini atau yang akan datang”.

Dari pendapat para ahli di atas dapat disimpulkan bahwa informasi adalah data yang telah diolah dan menjadi berguna bagi yang menerimanya.

3. Sistem

Menurut Santi (2020:1): “Sistem, adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk menyelesaikan suatu sasaran tertentu”.

Menurut Sudirman, et al (2020:14):

“Sistem didefinisikan sebagai seperangkat komponen yang saling terkait, dengan batas yang jelas, bekerja bersama untuk mencapai serangkaian tujuan bersama dengan menerima input dan menghasilkan output dalam proses transformasi yang terorganisir”.

Jadi, berdasarkan definisi-definisi di atas dapat disimpulkan bahwa sistem adalah sekumpulan elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu.

4. Sistem Informasi

Definisi sistem informasi menurut Suryadharma dan Triyani (2019:5): “Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan”.

Adapun juga definisi lain sistem informasi menurut Prehanto (2020:21): “Sistem informasi merupakan proses pengumpulan, penyimpanan, analisis sebuah informasi dengan tujuan tertentu”.

Jadi, sistem informasi adalah gabungan dari banyak kombinasi dari teknologi informasi untuk mencapai tujuan atau sasaran tertentu

5. Analisis Sistem

Menurut Yendrianof, et al (2022:30): “Analisis sistem adalah berupa bagian dari siklus hidup pembangunan sistem”. Sedangkan analisis sistem menurut Negara (2021:76):

“Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya”.

Dari pengertian di atas dapat disimpulkan bahwa analisis sistem adalah penguraian sistem ke dalam beberapa bagian untuk pengidentifikasian masalah dan penyusunan alternatif penyelesaian masalah yang terjadi.

6. Perancangan Sistem

Menurut Arifin, et al (2021:4): “Perancangan sistem merupakan pelengkap dari analisa sistem yang dituangkan ke dalam sebuah sistem yang utuh dengan tujuan mendapatkan sistem yang lebih baik”.

Menurut Santi (2020:71) :

“Perancangan sistem artinya adalah:

- a. Merupakan tahap lanjutan setelah tahap analisis sistem dalam daur hidup pengembangan sistem.
- b. Mendefinisikan setiap kebutuhan-kebutuhan fungsional.
- c. Mempersiapkan rancangan implementasi sistem yang baru/usulan.
- d. Menggambarkan sistem baru/usulan yang akan dikembangkan
- e. Mengatur dan merencanakan elemen-elemen yang terpisah serta mengkonfigurasi perangkat lunak dan keras”.

Jadi, perancangan sistem adalah tahap lanjutan setelah menganalisis sistem dalam daur hidup pengembangan sistem oleh seorang analis sistem.

7. Perancangan Input

Menurut Yendrianof, et al. (2020:115): “Perancangan *input* merupakan *form* untuk meng-*input* data yang berkaitan dengan sistem informasi yang akan dibuat”. Sedangkan menurut Pujiastuti, Wulandika, dan Solikhun (2020:22): “Perancangan *input* merupakan desain yang dirancang untuk menerima masukan dari pengguna sistem”.

Berdasarkan menurut para ahli di atas, maka dapat disimpulkan bahwa perancangan masukan (*input*) adalah penginputan data yang bertujuan untuk memberikan penjelasan atau suatu informasi, baik dari bentuk maupun dari masukan masukan yang akan diinputkan.

8. Perancangan Output

Menurut Santi (2020:77): “Perancangan *output* merupakan keluaran dari hasil proses suatu sistem informasi dikenal dengan istilah *output*, *output* yang dimaksud di sini dapat berbentuk kertas atau tampilan layar monitor”.

Menurut Yendrianof, et al (2022:144): “Perancangan *output* merupakan tahapan awal dari perancangan sistem informasi karena ketika meminta data penelitian maka yang diberikan adalah data berupa laporan”.

Berdasarkan menurut para ahli di atas, maka dapat disimpulkan bahwa perancangan keluaran (*output*) adalah bagian dari perancangan sistem yang merupakan hasil dari *input*-an yang telah di olah sistem menjadi sebuah laporan.

9. Perancangan Database

Menurut Henderi (2020:25): “Perancangan basis data utamanya melibatkan desain skema basis data. Rancangan lingkungan aplikasi basis data yang lengkap harus memenuhi kebutuhan perusahaan”.

Menurut Ginantra (2020:23):

“Rancangan database yang baik memiliki ciri-ciri yaitu dapat mencegah adanya anomali, tidak konsistennya data (inkonsistensi), memperkecil peluang munculnya duplikasi data atau tersimpannya data yang sama secara berulang, dan tidak banyak membutuhkan ruang penyimpanan”.

Jadi, perancangan *database* merupakan proses mendesain sebuah *database* untuk menentukan isi dan pengaturan data yang diperlukan dalam mendesain sistem yang sesuai dengan kebutuhan perusahaan.

B. Teori-Teori Khusus

1. Gereja

Menurut Hutahaean (2021:1): “Kata Gereja berasal dari kata dalam bahasa Portugis “*igreja*”, yang berasal dari kata Yunani “*ekklesia*” yang berarti: mereka yang dipanggil”.

Menurut Siregar (2022:5):

“Gereja berasal dari bahasa Portugis: *igreja*, yang berasal dari bahasa Yunani: *ἐκκλησία* (*ekklesía*) yang berarti dipanggil keluar (*ek* = keluar; *klesía* dari kata *kaleo* = memanggil); kumpulan orang yang dipanggil ke luar dari dunia, yaitu orang-orang yang dipanggil keluar dari kegelapan pada terangNya yang ajaib”.

Jadi, gereja merupakan tempat ibadah dari perkumpulan orang-orang yang menganut serta mempercayai ajaran iman kekristenan.

2. Administrasi

Menurut Taufiqurokhman, et al (2021:1):

“Administrasi sederhananya adalah kegiatan penyusunan dan pencatatan data serta informasi secara sistematis dengan tujuan untuk menyediakan keterangan serta memudahkan memperolehnya kembali secara keseluruhan dan dalam satu hubungan satu sama lain”.

Menurut Lubis dan Haidir (2019:1): “Administrasi dapat diartikan sebagai usaha untuk membantu, usaha untuk menolong, usaha untuk mengarahkan semua kegiatan dalam mencapai suatu tujuan”.

Berdasarkan definisi administrisasi di atas, dapat disimpulkan administrasi adalah seluruh proses penyusunan dan pencatatan data serta informasi untuk mengarahkan semua kegiatan dalam mencapai suatu tujuan.

3. Kamus Data

Menurut Rianto (2021:133): “Kamus data adalah kumpulan informasi terpusat terkait data. Ia menyimpan arti dan asal data, hubungannya dengan data lain, format data untuk penggunaan, dan sebagainya.”

Menurut Sukamto dan Shalahuddin (2018:73-74):

“Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum. Kamus data memiliki beberapa simbol tipe data untuk menjelaskan informasi tambahan sebagai berikut:

TABEL 2.1
NOTASI ATAU SIMBOL TIPE DATA

Simbol	Keterangan
X	Untuk setiap karakter
9	Angka <i>numeric</i>
A	Karakter alfabet
Z	Angka nol yang ditampilkan dalam spasi kosong
.	Pemisah ribuan
,	Pemisah pecahan
-	Tanda penghubung
/	Pembagi

Sumber: Sukamto dan Shalahuddin (2018:75)

Menurut Weli (2019:72-73):

“Kamus data merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data terdiri dari informasi-informasi data yang berisi informasi tentang atribut data lengkap mengenai data yang mengalir dalam sistem dan kelompok data dalam sebuah sistem. Untuk menunjukkan informasi-informasi tambahan di kamus data dipergunakan notasi sebagai berikut:

TABEL 2.2
NOTASI STRUKTUR KAMUS DATA

Simbol	Deskripsi
=	Terdiri dari/mendefinisika/diuraikan menjadi/artinya
+	Dan
()	Bersifat opsional
{ }	Pengulangan
[]	Seleksi (pilih salah satu dari sejumlah alternatif
*	Keterangan/komentar
@	Identifikasi atribut kunci
	Tanda pemisah pada sejumlah alternatif pilihan antara simbol []

Sumber: Weli (2019:73-74)

Jadi, berdasarkan definisi para ahli di atas, kamus data merupakan sekumpulan data yang mengalir dalam sistem perangkat lunak dan tersimpan di dalam *database*.

4. Normalisasi

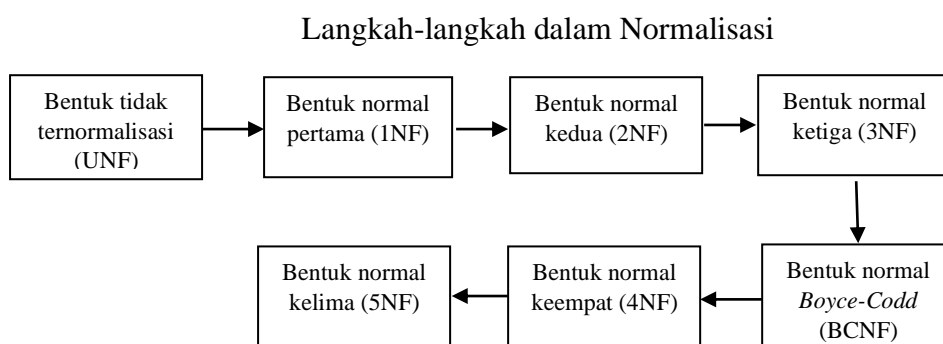
Menurut Ginantara, et al (2020:31):

“Normalisasi merupakan salah satu cara pendekatan atau teknik yang digunakan dalam membangun desain logic database relation dengan menerapkan sejumlah aturan dan kriteria standar, sehingga menghasilkan struktur tabel yang normal dan berstruktur baik”.

Menurut Haryanti (2022:25): “Normalisasi merupakan proses pengorganisasian atribut dengan pengaturan pengelompokan tertentu, agar entitas yang telah melalui proses normalisasi menjadi tidak redundant, tetap stabil dan dapat digunakan dengan fleksibel”.

Berdasarkan definisi para ahli di atas normalisasi adalah suatu proses pengelompokan atribut-atribut, agar entitas yang telah melalui proses normalisasi menjadi tidak berulang, tetap stabil dan dapat digunakan dengan fleksibel.

GAMBAR 2.1
LANGKAH-LANGKA DALAM NORMALISASI



Sumber: Ginantara, et al (2020:31)

5. *Unified Modelling Language (UML)*

Menurut Mesran, Nasution, dan Waruwu (2019:8): “UML merupakan bahasa visual untuk memodelkan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

Menurut Sukamto dan Shalahuddin (2018:133):

“UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”.

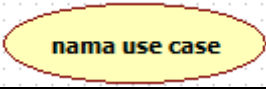


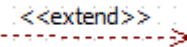

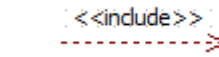
Dari pernyataan para ahli di atas dapat disimpulkan bahwa *Unified Modelling Language (UML)* adalah sebuah standar bahasa yang banyak digunakan oleh industri untuk memvisualisasi, merancang dan mendokumentasikan sistem perangkat lunak dengan menggunakan diagram dan teks-teks tertentu.

Menurut Sukamto dan Shalahuddin (2018:140-171) *Unified Modelling Language (UML)* ada beberapa diagram yang disediakan dan dapat digunakan, yaitu:

a. Use Case Diagram

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Berikut adalah simbol-simbol yang ada pada diagram *use case*:

TABEL 2.3
NOTASI ATAU SIMBOL USE CASE DIAGRAM


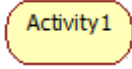



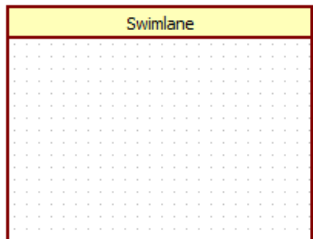
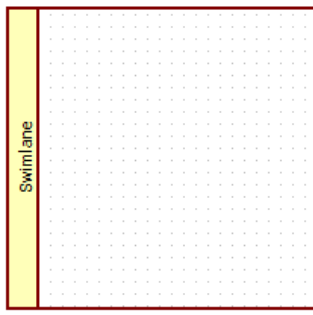
Simbol	Package
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
Aktor / <i>actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
Asosiasi / <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
Ekstensi / <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> .
Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus).
Menggunakan / <i>include / uses</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

Sumber: Sukanto dan Shalahuddin (2018:156-159)

b. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

TABEL 2.4
NOTASI ATAU SIMBOL ACTIVITY DIAGRAM


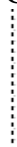
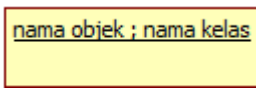

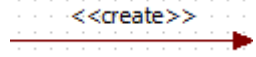
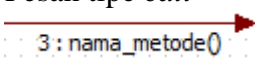
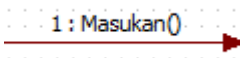
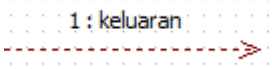
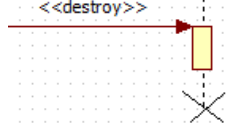
Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: Sukamto dan Shalahuddin (2018:162-163)

c. Sequence diagram

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Berikut adalah simbol-simbol yang ada pada diagram sekuen:

TABEL 2.5
NOTASI ATAU SIMBOL SEQUENCE DIAGRAM

Simbol	Deskripsi
<p>Aktor</p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.
<p>Objek</p> 	Menyatakan objek yang berinteraksi pesan
<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
<p>Pesan tipe <i>create</i></p> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
<p>Pesan tipe <i>call</i></p> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
<p>Pesan tipe <i>send</i></p> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
<p>Pesan tipe <i>return</i></p> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
<p>Pesan tipe <i>destroy</i></p> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri.

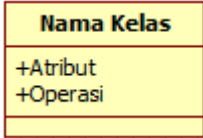



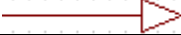


Sumber: Sukanto dan Shalahuddin (2018:165-167)

d. Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat

untuk membangun sistem. Kelas memiliki apa yang disebut dan atribut dan metode atau operasi.

TABEL 2.6
NOTASI ATAU SIMBOL CLASS DIAGRAM

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka / <i>Interface</i></p> 	Sama dengan konsep interface dalam pemrograman berorientasi objek
<p>Asosiasi / <i>Association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Asosiasi berarah / <i>directed association</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p>Generalisasi / <i>generalization</i></p> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
<p>Kebergantungan / <i>dependency</i></p> 	Relasi antarkelas dengan makna kebergantungan antarkelas
<p>Agregasi / <i>aggregation</i></p> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

Sumber: Sukanto dan Shalahuddin (2018: 146-147)

e. Object Diagram

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem.

f. Component Diagram

Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem

g. Composite Structure Diagram

Composite structure diagram digunakan untuk menggambar struktur dari bagian-bagian yang saling terhubung maupun mendeskripsikan struktur pada saat berjalan (*runtime*) dari *instance* yang saling terhubung.

h. Package Diagram

Package diagram menyediakan cara mengumpulkan elemen-elemen yang saling terkait dalam diagram UML.

i. Deployment Diagram

Diagram pengembangan atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi.

j. State Machine Diagram

State machine diagram atau diagram mesin status digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek.

k. Communication Diagram

Communication diagram atau diagram komunikasi menggambarkan interaksi antar objek/bagian dalam bentuk urutan pengiriman pesan. Diagram ini memrepresentasikan informasi yang diperoleh dari Diagram Kelas, Diagram sekuen, dan Diagram *Use Case* untuk mendeskripsikan gabungan antata struktur statis dan tingkah lau dinamis dari suatu sistem.

l. Timing Diagram

Timing diagram merupakan diagram yang fokus pada penggambarannya terkait batasan waktu. Timing diagram digunakan untuk menggambarkan tingkah laku sistem dalam periode waktu tertentu.

m. Interaction Overview Diagram

Interaction overview diagram berfungsi untuk menggambarkan sekumpulan urutan aktivitas. *Interaction overview* diagram adalah bentuk aktivitas diagram yang setiap titik merepresentasikan diagram interaksi.

6. Visual Studio Code

Menurut Salamah (2021:1): “Visual Studio Code (VS 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”.

Menurut Habibi, Rahman, dan Dwiifanka (2020:3): “Visual Studio Code adalah versi ringan dari lingkungan pengembangan Microsoft resmi yang secara eksklusif berfokus pada editor kode”.

Jadi, Visual Studio Code adalah sebuah teks editor yang dibuat oleh Microsoft yang bersifat *multiplatform*.

7. XAMPP

Menurut Roza, Fauzan, dan Rahayu (2020:82): “XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program”.

Menurut Sembiring, et al (2020:11): “XAMPP adalah perangkat yang menggabungkan tiga aplikasi kedalam satu paket yaitu Apache, MySQL, dan phpMyAdmin”.

Jadi, XAMPP adalah perangkat lunak sumber terbuka yang berisi kompilasi dari beberapa program ke dalam satu paket aplikasi.

8. *Database Management System* (DBMS)

Menurut Ginantara, et al (2020:4): “*Database Management System* (DBMS) adalah sebuah perangkat lunak yang dirancang untuk melakukan pemrosesan suatu basis data dan menjalankan operasi yang diminta oleh pengguna”.

Menurut Fatkhurrochman, Muin, dan Priyoatmoko (2021:1): “*Database Management System* (DBMS) merupakan *software* yang digunakan untuk mengelola, penyimpanan dan pengambilan data yang telah tersimpan di dalam *database*”.

Jadi, *Database Management System* (DBMS) adalah perangkat lunak yang dirancang dan digunakan untuk mengelola, menyimpan dan mengambil data yang telah disimpan di *database*.

9. Web

Menurut Widia dan Asriningtias (2021:3): “Website adalah kumpulan dokumen berupa halaman web yang berisi teks dalam format *Hyper Text Markup Language* (HTML)”.

Menurut Limbong dan Sriadhi (2021:4): “Web merupakan fasilitas *hypertext* untuk menampilkan data berupa teks, gambar, suara, animasi, dan data multimedia lainnya”.

Dari pendapat para ahli di atas dapat disimpulkan, web atau *website* merupakan fasilitas yang dapat menampilkan data multimedia hingga bertukar informasi di internet melalui web *browser*.

10. HTML

Menurut Arifin (2019:19): “HTML merupakan bahasa utama yang digunakan untuk membangun halaman web dan merupakan sumber kode dari hampir semua halaman web”.

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

Berdasarkan definisi-definisi di atas, HTML merupakan bahasa pemrograman standar untuk membangun sebuah *website* dari *tag-tag* untuk membuat sebuah dokumen yang dapat dibaca oleh web *browser*.

11. Bootstrap

Menurut Styawantoro dan Komarudin (2021:20): “*Bootstrap* merupakan *library framework* CSS yang dibuat khusus untuk

pengembangan *front-end website*”. Sedangkan, menurut Sulistiono (2018:17): “Bootstrap adalah sebuah pustaka *open source* yang merupakan *framework* CSS dan Javascript untuk membuat website yang responsif”.

Berdasarkan definisi-definisi *bootstrap* di atas dapat disimpulkan, *Bootstrap* adalah *framework library* yang berisi kumpulan fungsi-fungsi CSS yang membantu programmer dalam pengembangan sisi *front-end* sebuah *website*.

12. JavaScript

Menurut Limbong dan Sriadhi (2021:69): “*JavaScript* merupakan salah satu bahasa pemrograman tingkat tinggi dan dinamis dalam membuat sebuah *website*. Selain HTML dan CSS, *JavaScript* merupakan salah satu teknologi inti dalam pembuatan Web”.

Menurut Supriyadi (2021:1): “*JavaScript* merupakan bahasa pemrograman tingkat tinggi (*High Level Language*) dan bersifat dinamis yang berjalan disisi klien (*Client Side Script Programming*)”.

Jadi, *JavaScript* merupakan salah satu bahasa pemrograman tingkat tinggi yang berjalan pada sisi klien guna membuat sebuah web menjadi dinamis.

13. PHP

Menurut Krisbiantoro dan Abda’u (2021:4): “PHP dikenal sebagai sebuah bahasa *scripting* yang menyatu dengan tag-tag HTML yang dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis”.

Menurut Widia dan Asriningtias (2021:31): “PHP merupakan singkatan dari *Hypertext Preprocessor* yang adalah bahasa berbentuk *script* yang ditempatkan di dalam server dan diproses di dalam server dan hasilnya akan dikirimkan ke user dalam bentuk halaman web yang diakses menggunakan browser”.

Jadi, PHP merupakan salah satu bahasa pemrograman yang mendukung dalam pembuatan web pada sisi server.

14. MySQL

Menurut Fitri (2020:2): “MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data”.

Menurut Indrawan (2021:3): “MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya sangat cepat, *multiuser* serta menggunakan perintah dasar SQL (*Structured Query Language*)”.

TABEL 2.7
TIPE DATA MYSQL UNTUK BILANGAN

Tipe Data	Keterangan
<i>TINYINT</i>	Ukuran 1 <i>byte</i> . Bilangan bulat terkecil, dengan jangkauan untuk bilangan bertanda: -128 s/d 127 dan jangkauan <i>unsigned</i> : 0 s/d 255.
<i>SMALLINT</i>	Ukuran 2 <i>byte</i> . Integer yang kecil jangkuan nilainya - 32768 s/d 32767 dan jangkauan <i>unsigned</i> 0 s/d 65535.
<i>MEDIUMINT</i>	Ukuran 3 <i>byte</i> . Bilangan bulat dengan jangkauan - 8388608 s/d 8388607 dan jangkauan <i>unsigned</i> 0 s/d 16777215.
<i>INT</i>	Ukuran 4 <i>byte</i> . Integer yang berukuran normal. Jangkauan nilainya -2147483648 s/d 2147483647.

Sumber: Indrawan (2021:4-5)

TABEL 2.8
TIPE DATA MYSQL UNTUK BILANGAN LANJUTAN

Tipe Data	Keterangan
<i>BIGINT</i>	Ukuran 8 <i>byte</i> . Integer dengan jangkauan nilai -9223372036854775808 s/d 9223372036854775807. Jangkauan <i>unsigned</i> 0 s/d 18446744073709551615.
<i>FLOAT</i>	Ukuran 4 <i>byte</i> . Bilangan pecahan.
<i>DOUBLE</i>	Ukuran 8 <i>byte</i> . Bilangan pecahan
<i>DECIMAL</i> (M,D)	Ukuran M <i>byte</i> . Bilangan pecahan, misalnya DECIMAL (5,2) dapat digunakan untuk menyimpan bilangan -99,99 s/d 99,99.

Sumber: Indrawan (2021:4-5)

TABEL 2.9
TIPE DATA MYSQL UNTUK TANGGAL DAN JAM

Tipe Data	Keterangan
<i>DATETIME</i>	Ukuran 8 <i>byte</i> . Kombinasi tanggal dan jam dengan jangkauan '1000-01-01 00:00:00' s/d '9999-12-31 23:59:59'.
<i>DATE</i>	Ukuran 3 <i>byte</i> . Tanggal dengan jangkauan '1000-01-01' s/d '9999-12-31'.
<i>TIMESTAMP</i>	Ukuran 4 <i>byte</i> . Kombinasi tanggal dan jam dengan jangkauan '1970-01-01 00:00:00' s/d '2037'.
<i>TIME</i>	Ukuran 3 <i>byte</i> . Tipe data waktu dengan jangkauannya adalah '839:59:59' s/d '838:59:59'.
<i>YEAR</i>	Ukuran 1 <i>byte</i> . Tahun antara 1900 s/d 2155.

Sumber: Indrawan (2021:5)

TABEL 2.10
TIPE DATA MYSQL UNTUK KARAKTER DAN LAIN-LAIN

Tipe Data	Keterangan
<i>CHAR</i>	Menangani data hingga 255 karakter. Tipe data ini mengharuskan untuk memasukkan data yang telah ditentukan oleh kita
<i>VARCHAR</i>	Menangani data hingga 255 karakter. Tipe data ini tidak mengharuskan untuk memasukkan data yang telah ditentukan oleh kita
<i>TINYBLOB</i> , <i>TINYTEXT</i>	Mampu menangani data hingga 255 karakter.
<i>BLOB</i> , <i>TEXT</i>	Lebar maksimum 65535 karakter.
<i>MEDIUMBLOB</i> , <i>MEDIUMTEXT</i>	Lebar maksimum 16777215 karakter.
<i>LOBLOB</i> , <i>LONGTEXT</i>	Lebar maksimum 4294967295 karakter.

Sumber: Indrawan (2021:5-6)

TABEL 2.11
TIPE DATA MYSQL UNTUK KARAKTER DAN LAIN-LAIN LANJUTAN

Tipe Data	Keterangan
<i>ENUM</i> ('nilai1', 'nilai2', '....', 'nilai N')	Ukuran 1 atau 2 <i>byte</i> . Tergantung jumlah nilai enumerasinya (maksimum 65535 nilai).
<i>SET</i> ('nilai1', 'nilai2', '....', 'nilai N')	1,2,3,4 atau 8 <i>byte</i> . Tergantung jumlah anggota himpunan (maksimum 64 anggota).

Sumber: Indrawan (2021:5-6)