



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Judul

##### 2.1.1 Pengertian Aplikasi

Widianti, Fachri, dkk. (2020) mengatakan bahwa “Aplikasi adalah suatu perangkat lunak yang dibuat sebagai *front end* sebuah sistem yang dipakai untuk mengelola data sehingga menjadi suatu informasi yang bermanfaat bagi pengguna”,

Sedangkan Menurut Kadir (2008:3), menyatakan bahwa “Program aplikasi adalah program siap pakai atau program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain”.

Aplikasi juga diartikan sebagai penggunaan atau penerapan suatu konsep yang menjadi pokok pembahasan atau sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. Aplikasi *software* yang dirancang untuk penggunaan praktisi khusus, klasifikasi luas ini dapat dibagi menjadi 2 (dua) yaitu:

- a) aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu; dan
- b) aplikasi paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

Dari kedua pengertian di atas dapat disimpulkan bahwa aplikasi adalah sekumpulan perintah atau kode yang disusun secara sistematis untuk menjalankan suatu perintah yang diberikan oleh manusia melalui komponen atau *hardware* komputer yang digunakan oleh manusia dalam menjalankan program aplikasi, dengan demikian bisa membantu manusia untuk memberikan solusi dari apa yang diinginkan.



### 2.1.2 Pengertian Pengolahan Data

Pengertian pengolahan data menurut Kristanto (2018:8) “Pengolahan data merupakan waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Semakin banyak data dan kompleksnya aktivitas pengolahan data dalam suatu organisasi, baik itu organisasi besar maupun organisasi kecil, maka metode pengolahan data yang tepat sangat dibutuhkan. Salah satu metode untuk mengolah data adalah dengan media pengolahan data yang menggunakan komputer. Dengan media ini semua permasalahan yang ada dapat diselesaikan secara cepat baik itu permasalahan yang menggunakan perhitungan matematis maupun fungsi-fungsi lainnya. Selain itu dengan komputer, permasalahan yang ada dapat diselesaikan dengan baik.

### 2.1.3 Pengertian Pendistribusian

Pendistribusian atau Distribusi berakar dari bahasa Inggris *distribution* yang berarti penyaluran. Sedangkan kata dasarnya *to distribute*, berdasarkan Kamus Inggris Indonesia John M, Echols dan Hassan Shadilly dalam Damsar (2009:93) bermakna membagikan, menyalurkan, menyebarkan, mendistribusikan, dan mengageni. Sedangkan menurut Kamus Besar Bahasa Indonesia, distribusi dimaksudkan sebagai penyalur (pembagian, pengiriman) kepada beberapa orang atau beberapa tempat. Jadi berdasarkan rujukan di atas, distribusi dapat dimengerti sebagai proses penyaluran barang atau jasa kepada pihak lain. Dalam kegiatan distribusi diperlukan adanya sarana dan tujuan sehingga kegiatan distribusi dapat berjalan dan terlaksana dengan baik.

Kegiatan distribusi merupakan salah satu fungsi pemasaran yang sangat penting dilakukan dalam pemasaran yaitu untuk mengembangkan dan memperluas arus barang atau jasa mulai dari produsen sampai ketangan konsumen sesuai dengan jumlah dan waktu yang telah ditentukan.



Pemilihan proses distribusi merupakan suatu masalah yang sangat penting sebab kesalahan dalam pemilihan proses distribusi dapat memperlambat proses penyaluran barang atau jasa sampai ketangan konsumen atau pemakai.

Menurut Gugup Kismono (2001:364), “Distribusi adalah perpindahan barang dan jasa dari produsen ke pemakai industri dan konsumen”,

Sedangkan Menurut Sofyan Assauri (2004:83), menyatakan bahwa “Distribusi merupakan suatu lembaga yang memasarkan produk, yang berupa barang atau jasa dari produsen ke konsumen”.

#### **2.1.4 Pengertian Bekandam II/Sriwijaya**

Bekandam (Pembekalan Angkutan Kodam) II/Sriwijaya adalah salah satu satuan pendukung utama sebuah wilayah militer yang berkedudukan di bawah Panglima Kodam. Tugas pokok dari Bekandam II/Sriwijaya adalah mengadakan pelayanan jasa, pemeliharaan bekal materiel pembekalan angkutan dalam rangka mendukung tugas pokok Kodam.

Bekandam II/Sriwijaya memiliki tanggung jawab fungsi utama berupa pembekalan meliputi perencanaan atau pemenuhan kebutuhan pengadaan, penerimaan, penimbunan atau penyimpann, pengeluaran atau pendistribusian, pengembalian atau pengantar bekal materiel, penyampaian berit administrasi dan pertanggung jawaban bekal atau materiel sebagai implementasi fungsi pembekalan Kodam.

#### **2.1.5 Pengertian Website**

Menurut Yeni Susilowati (2019), “*Website* adalah sejumlah halaman web yang memiliki topik saling terkait antara halaman yang satu dengan halaman lain, yang biasanya ditempatkan pada sebuah server web yang dapat diakses melalui jaringan interet ataupun jaringan wilayah lokal (LAN)”,

Sedangkan menurut Abdulloh (2018:1) menyatakan bahwa, “*Website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang



disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia”.

## **2.2 Teori Umum**

### **2.2.1 Perangkat Lunak**

Menurut Kadir (2017:2) “Perangkat lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai”,

Sedangkan Menurut Sukamto dan Shalahuddin (2018:2), Menyatakan Bahwa, “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*)”.

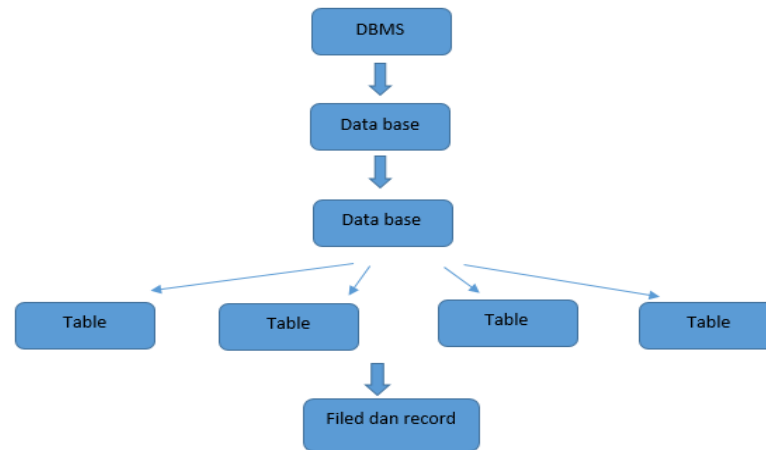
Dari pendapat di atas dapat disimpulkan bahwa Perangkat lunak adalah perintah-perintah yang diarahkan kepada komputer sehingga dapat menjalankan tugas dengan benar sesuai keinginan *user*.

### **2.2.2 Basis Data (*Database*)**

Fathansyah (2018:2) “Basis data (*Database*) adalah himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.”

Sedangkan Menurut Sukamto dan Shalahuddin (2018:43), Menyatakan Bahwa, “*Database* adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat”.

Jadi, basis data adalah kumpulan data yang saling berhubungan dan diorganisasikan dalam media penyimpanan sehingga dapat diakses dengan mudah dan cepat. Berikut struktur basis data :



**Gambar 2.1** Struktur *Database*

### 2.2.3 Metode Pengembangan Sistem

Menurut Prabowo dalam (Supriyatna, 2018) *Extreme Programming (XP)* merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan-perubahan *requirement* yang sangat cepat,

Sedangkan Menurut Pratama (2017:93), “*Extreme Programming (XP)* adalah sebuah pendekatan atau model pengembangan perangkat lunak yang mencoba menyederhanakan berbagai tahapan dalam proses pengembangan tersebut sehingga menjadi lebih *adaptif* dan *fleksibel*”.

Dari kedua pendapat tersebut dapat disimpulkan bahwa metode *Extreme Programming (XP)* merupakan metode pengembangan perangkat lunak yang ringan dan termasuk salah satu metode *agile*. *Extreme Programming (XP)* lebih cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium. Sehingga metode *extreme programming (XP)* lebih mengedepankan proses pengembangan yang



---

lebih *responsive* terhadap kebutuhan. Adapun tahap-tahap (*fase*) dalam metode pengembangan metode *extreme programming* (XP) adalah sebagai berikut :

1) *planning* (perencanaan)

Tahap ini merupakan Langkah awal dalam pembangunan sistem dimana dalam tahapan ini dilakukan beberapa kegiatan perencanaan yaitu, identifikasi permasalahan, menganalisa kebutuhan sampai dengan penetapan jadwal pelaksanaan pembangunan sistem;

2) *design* (perancangan)

Tahap ini adalah perancangan, dimana pada tahapan ini dilakukan kegiatan pemodelan yang dimulai dari pemodelan sistem, pemodelan arsitektur sampai dengan pemodelan basis data;

3) *coding* (pengkodean)

Tahap ini merupakan kegiatan penerapan pemodelan yang sudah dibuat ke dalam bentuk *user interface* dengan menggunakan Bahasa pemrograman; dan

4) *testing* (pengujian)

Setelah tahapan pengkodean selesai, kemudian dilakukan tahapan pengujian sistem untuk mengetahui kesalahan apa saja yang timbul saat aplikasi sedang berjalan serta mengetahui apakah sistem yang dibangun sudah sesuai dengan kebutuhan pengguna.

## 2.3 Teori Khusus

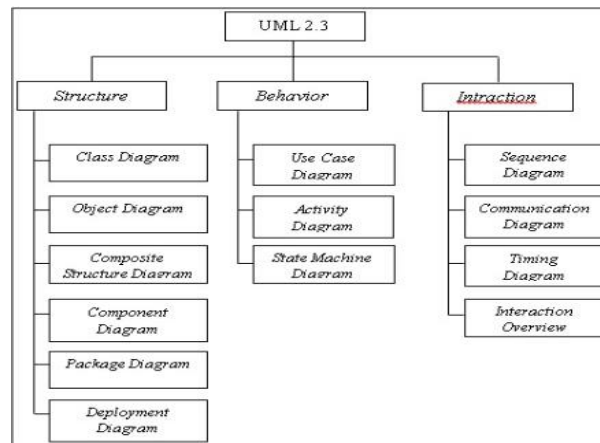
### 2.3.1 *Unified Modelling Language* (UML)

Menurut Rosa Irmayani & Susyati (2017) “UML adalah salah satu standar Bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”,

Sedangkan Menurut Sukanto dan Shalahuddin (2018:137), menjelaskan bahwa “*Unified Modeling Language* (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks – teks pendukung”.



Pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam – macam diagram tersebut yaitu ada pada halaman berikutnya.



(Sumber: Sukamto dan Shalahuddin (2018:140))

**Gambar 2.2 Diagram UML**

Berikut penjelasan singkat dari pembagian kategori pada diagram UML:

- 1) *structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan;
- 2) *behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem; dan
- 3) *interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

## 2.3.2 Jenis-Jenis Diagram UML

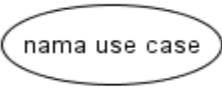
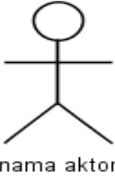

### 2.3.2.1 Pengertian *Use case Diagram*

Sukamto dan Shalahuddin (2018:155), menjelaskan tentang *use case* diagram sebagai berikut:



“*Use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem.” Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut:

**Tabel 2.1** Simbol-simbol *Use case* Diagram

No	Simbol	Deskripsi
1.	<p><i>Use case</i></p> 	Menggambarkan bagaimana seseorang menggunakan sistem
2.	<p>Aktor / Actor</p> 	Seseorang atau apa saja yang berhubungan dengan sistem
3.	<p>Asosiasi / Association</p> 	Relasi yang dipakai untuk menunjukkan hubungan antara aktor dan <i>use case</i>
4.	<p>Relasi Extend</p> <p>&lt;&lt;extend&gt;&gt;</p>	Memungkinkan suatu <i>use case</i> secara optional menggunakan fungsionalitas yang disediakan oleh <i>use case</i> lainnya
5.	<p>Relasi include / uses</p> <p>&lt;&lt;include&gt;&gt;</p>	Memungkinkan satu <i>use case</i> menggunakan fungsionalitas yang disediakan oleh <i>use case</i> lainnya

(Sumber: Rusmawan, (2019:72-73))





Ada dua hal utama pada *use case* yaitu:

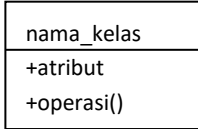
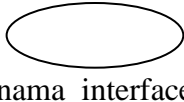
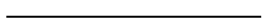
- 1) aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri. Jadi, walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; dan
- 2) *use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

### 2.3.2.2 Pengertian *Class Diagram*

“*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.” (Sukanto dan Shalahuddin (2018:141)),

Adapun simbol-simbol yang digunakan dalam *class diagram* adalah sebagai berikut:

**Tabel 2.2** Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.	kelas 	Kelas pada struktur sistem
2	antarmuka / interface 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3.	asosiasi / association 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>

Lanjutan Tabel 2.2 Simbol-simbol *Class Diagram*

4.	asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus)
6.	kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas
7.	agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> )

(Sumber: Sukanto dan Shalahuddin (2018:144-147))

### 2.3.2.3 Pengertian *Activity Diagram*



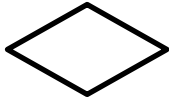

Sukanto dan Shalahuddin (2016:161), “*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.”

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:

Tabel 2.3 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	<i>Start State/ Status Awal</i> 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.

Lanjutan Tabel 2.3 Simbol-simbol *Activity Diagram*

2.	<i>End State/ Status Akhir</i> 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
3.	<i>Activity/ Aktivitas</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
4.	<i>Decision/ Percabangan</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
5.	<i>Interaction/ Interaksi</i> 	Alur.

(Sumber: Rusmawan (2019:80))

#### 2.2.3.4 Pengertian *Sequence Diagram*

“Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah beserta metode-metode yang dimiliki kelas yang diinstansikan menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*” (Sukamto dan Shalahuddin, 2016:165).


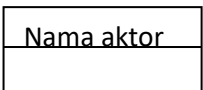

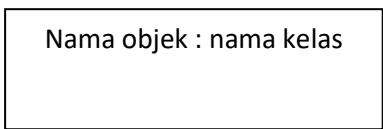


“Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence* diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence* diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan *output* tertentu” (Safaat, 2015:33-34).



Dapat penulis simpulkan bahwa *Sequence* diagram adalah penggambaran skenario dari sebuah objek yang ada pada *Use Case* yang meliputi rangkaian langkah-langkah aktivitas dari objek berdasarkan waktu hidup objek dan pesan-pesan yang diterima maupun yang dikirimkan objek kepada objek lainnya.

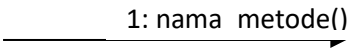
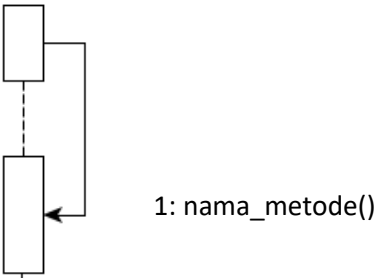
Berikut simbol-simbol pada *Sequence* Diagram :

**Tabel 2.4** Simbol-simbol pada *Sequence* Diagram

No	Simbol	Deskripsi
1.	<p>Actor</p>  <p>atau</p>  <p>tanpa waktu aktif</p>	orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama actor
2.	<p>Garis hidup / <i>lifeline</i></p> 	menyatakan kehidupan suatu objek
3.		menyatakan objek yang berinteraksi pesan
4.	<p>Waktu aktif</p> 	menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.	<p>Pesan tipe create</p> 	menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat



**Lanjutan Tabel 2.4** Simbol-simbol pada *Sequence* Diagram

6.	Pesan tipe call 	menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,  arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
----	--	--

## 2.4 Teori Program

### 2.4.1 Pengertian PHP

Enterprise (2018:1) menegaskan, “PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif. Dinamis artinya, *website* tersebut bisa berubah tampilan dan kontennya sesuai kondisi tertentu”,

Sedangkan Abdullah (2018:127) menegaskan bahwa, “PHP merupakan bahasa pemrograman web yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server”.

Dapat disimpulkan bahwa PHP adalah bahasa pemrograman universal untuk penangan pembuatan dan pengembangan sebuah situs *web* dan bisa digunakan bersamaan dengan HTML.

### 2.4.4 Pengertian CSS

Abdullah (2018:45) menegaskan, “CSS adalah 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”,

Sedangkan Mulyani (2020:77) menegaskan bahwa, “CSS adalah kumpulan perintah yang digunakan untuk menjelaskan sebuah halaman situs *web* dalam *mark-up language*”.

Dapat disimpulkan bahwa, CSS adalah salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan/*layout* halaman web supaya lebih elegan dan menarik.

#### 2.4.2 Pengertian MySQL

Menurut Kadir (dikutip Pranata dkk. 2015:25), “*MySQL* merupakan *software* yang berbasis *structure query language* (SQL) tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. *MySQL* adalah sebuah implementasi dari sistem manajemen *database* relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial”,

Sedangkan menurut Nugroho (2014:31), “*MySQL* atau program aplikasi *database*, yaitu *software* yang dapat kita pakai untuk menyimpan data berupa informasi *text* dan juga angka”.

Dapat disimpulkan bahwa *MySQL* merupakan *software* DBMS yang bersifat *Open Source* dan digunakan untuk menyimpan data atau server *database* yang mendukung bahasa *database* pencarian SQL.

#### 2.4.3 Pengertian HTML (*Hypertext Markup Language*)

Menurut Madcoms (dikutip Pranata dkk. 2015:26), “HTML (*HyperText Markup Language*) dikenal sebagai bahasa kode berbasis teks untuk membuat sebuah halaman *web*, keberadaannya dikenal dengan adanya ekstensi \*.htm atau



\*.html.”

Sedangkan menurut Supriyanto (dikutip Pranata dkk. 2015:26), “HTML merupakan suatu bahasa dari *website* (*www*) yang dipergunakan untuk menyusun dan membentuk dokumen agar dapat ditampilkan pada program *browser*.”

Abdulloh (2018:127) menyatakan bahwa, “HTML merupakan singkatan dari *Hypertext Markup Language* yaitu bahasa standar *web* yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website*”.

#### 2.4.5 Pengertian JavaScript

Menurut Abdulloh (2018:193), JavaScript merupakan bahasa pemrograman *web* yang pemrosesannya dilakukan di sisi klien. Karena berjalan di sisi klien, JavaScript dapat dijalankan hanya dengan menggunakan *browser*. Karena berjalan di sisi klien, JavaScript dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan PHP yang bekerja di sisi *server*, untuk menjalankan skrip JavaScript tidak memerlukan refresh pada *browser*.

Sedangkan menurut Sunyoto (2007:17), menyatakan bahwa “JavaScript adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar browser populer seperti *Internet Explorer* (IE), *Mozilla Firefox*, *Netscape* dan *Opera*”.

Kode Javascript dapat disisipkan dalam halaman web menggunakan tag SCRIPT. Beberapa hal tentang Javascript:

- 1) javascript didesain untuk menambah interaktif suatu web;
- 2) javascript merupakan sebuah bahasa *scripting*;
- 3) bahasa *scripting* merupakan bahasa pemrograman yang ringan;
- 4) javascript berisi baris kode yang dijalankan di komputer (web browser);
- 5) javascript biasanya disisipkan (*embedded*) dalam halaman HTML; dan
- 6) javascript adalah bahasa *interpreter* (yang berarti skrip dieksekusi tanpa proses kompilasi).



---

### 2.4.5.1 Cara Penulisan Kode Javascript pada HTML

#### 1. Tag <script>

Cara yang umum yang dipakai adalah menuliskanya dalam tag <script>. Tag <script> bisa dibuat di dalam tag <head>, mauapun di dalam tag <body>. Contoh:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Penulisan Javascript</title>
  <script>
    console.log("Hi, ini kode Javascript");
  </script>
</head>
<body>

<script>
document.write("Javascript itu keren!");
</script>
</body>
</html>
```

(Sumber: <https://www.petanikode.com/javascript-dalam-html/>).