



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Berikut ini merupakan penjelasan singkat mengenai beberapa teori umum mengenai judul yang penulis tulis:

2.1.1 Pengertian Komputer

Menurut Robert H. Blissmer (dalam Harmayani *et al*, 2021) mengatakan, “Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima input, memproses input sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan output dalam bentuk informasi”.

Menurut (Harmayani *et al*, 2021), “Definisi Komputer adalah alat elektronik yang terdiri dari rangkaian berbagai komponen yang saling terhubung sehingga membentuk suatu sistem kerja”.

Dari pengertian diatas dapat disimpulkan bahwa komputer adalah alat elektronik yang terdiri dari berbagai rangkaian komponen yang saling terhubung yang dapat menerima, dan memproses, serta menyimpan input-an yang dioperasikan manusia

2.1.2 Pengertian Sistem

Menurut (Lestari *et al*, 2020) mengemukakan bahwa “Sistem adalah dua atau lebih komponen yang saling berhubungan dan berintraksi membentuk kesatuan kelompok sehingga menghasilkan satu tujuan”.

Menurut (Kristanto, 2018) pengertian ”Sistem merupakan kumpulan elemen – elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (output) yang diinginkan”.

Jadi, dapat disimpulkan bahwa sistem adalah kumpulan elemen yang saling berhubungan untuk memproses input yang ditujukan kepada sistem dan mengolah input sampai menghasilkan output yang diinginkan.



2.1.3 Pengertian Perangkat Lunak (*Software*)

Menurut (Sukamto dan Shalahuddin, 2018) “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara pengguna”.

Menurut Mulyani (dalam Fitriadi, 2020), “Perangkat Lunak (*Software*) merupakan sebuah perangkat yang tidak berbentuk secara fisik, namun dapat dioperasikan oleh user atau penggunanya dengan kumpulan program komputer yang digunakan untuk memproses data”.

Berdasarkan beberapa pengertian diatas dapat disimpulkan bahwa perangkat lunak adalah program komputer terasosiasi dengan dokumentasi yang digunakan untuk menjalankan suatu pekerjaan.

2.1.4 Pengertian Internet

Menurut (Arthalita & Prasetyo, 2020) Pengertian internet dapat disimpulkan sebagai seluruh jaringan yang saling terhubung secara fisik dan memiliki kemampuan untuk membaca dan menguraikan protocol komunikasi dengan jumlah pengguna lebih dari 200 negara.

Adapun pengertian internet menurut (Yusuf *et al*, 2019) “adalah singkatan dari *Interconnected Networking* yang apabila diartikan dalam Bahasa Indonesia berarti rangkaian komputer yang terhubung di dalam beberapa rangkaian jaringan. Internet merupakan salah satu hasil dari kecanggihan dan kemajuan ilmu pengetahuan dan teknologi buatan manusia”.

Dari pendapat di atas dapat disimpulkan bahwa internet adalah rangkaian komputer yang didalamnya berisi informasi dan dapat terjadi pertukaran informasi secara elektronik pada ruang maya.

2.1.5 Metode Pengembangan Aplikasi

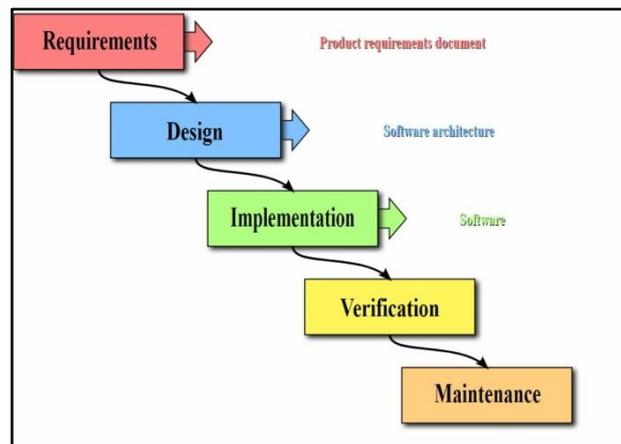
Metode *waterfall* merupakan sebuah metode dalam mengembangkan suatu perangkat lunak yang memungkinkan pembuatan sistem secara sistematis dan terstruktur (algoritma) sesuai dengan siklus dan tahapan yang berlaku. Metode ini disebut *waterfall* karena berfilosofi seperti air terjun, dimana dari tahap satu ke tahap berikutnya adalah berkesinambungan dan berkaitan erat. (Maulana, 2021)

Sedangkan menurut (Heri Ms, 2021) menyebutkan bahwa, “metode *waterfall* merupakan sebuah metode dalam mengembangkan suatu perangkat



lunak yang memungkinkan pembuatan sistem secara sistematis dan terstruktur (algoritma) sesuai dengan siklus dan tahapan yang berlaku”. Metode ini disebut *waterfall* karena berfilosofi seperti air terjun, dimana dari tahap satu ke tahap berikutnya adalah berkesinambungan dan berkaitan erat.

Model *waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari *Requirments*, *Design*, *Implementation*, *Verification*, *Maintenance*.



Sumber : Maulana (2021)

Gambar 2.1 Metode *Waterfall*

1. *Requirement* (Analisis Kebutuhan)

Requirement adalah proses analisa atau pengumpulan data-data yang berkaitan dengan sistem yang akan dibuat. Sebelum melakukan pengembangan perangkat lunak, seorang pengembang harus mengetahui dan memahami bagaimana informasi kebutuhan pengguna terhadap sebuah perangkat lunak.

2. *Design System* (Desain Sistem)

Proses ini akan berfokus pada pembangunan struktur data, arsitekur perangkat lunak, perancangan *interface*, perancangan fungsi internal dan eksternal serta detail dari setiap algoritma prosedural. Tahapan design akan menghasilkan dokumen bernama “*Software Requirement*” yang nantinya menjadi landasan para programmer dalam membuat code-code aplikasi.

3. *Implementation* (Pengerjaan)

Tahap ini adalah tahapan pembuatan aplikasi oleh para programmer dengan menggunakan kode-kode bahasa pemrograman tertentu. Proses penulisan



sinkode (coding) aplikasi mengacu pada dokumen-dokumen yang telah dibuat sebelumnya.

4. *Verification* (Verifikasi)

Tahapan verifikasi meliputi pengintegrasian sistem dan juga melakukan testing terhadap aplikasi yang telah dibuat. Sistem akan diverifikasi untuk diuji sejauh mana kelayakannya. Dalam tahapan ini semua modul yang dikerjakan oleh programmer berbeda akan digabungkan kemudian diuji apakah telah sesuai dengan spesifikasi yang ditetapkan atau terdapat kesalahan/error dalam sistem sebelum kemudian diperbaiki ulang.

5. *Maintenance* (Pemeliharaan)

Tahapan ini umumnya meliputi tahapan penginstalasian perangkat lunak dan pengujian aplikasi. *Maintenance* juga adalah bentuk tanggung jawab tim pengembang untuk memastikan aplikasi dapat berjalan lancar setelah diserahkan pada klien dalam periode waktu tertentu.

2.1.6 Metode Pengujian

Menurut Jaya (dalam Ramdani *et al*, 2023) *Black Box Testing*, merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. *Black Box Testing* memungkinkan pengembang *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program.

Menurut (Vikasari, 2018) Salah satu metode pengujian yang berfokus pada dari perangkat lunak disebut *Black Box Testing*. Pengujian ini memberikan gambaran atas sekumpulan kondisi masukan dan melakukan pengujian pada uraian fungsional program.

2.2 Teori Khusus

Berikut ini merupakan penjelasan singkat mengenai beberapa teori khusus terkait dengan judul laporan yang penulis tulis :

2.2.1 Pengertian *Unified Modeling Language* (UML)

Menurut (Sukamto dan Shalahuddin, 2018) “UML (*Unified Modelling Language*) 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”.



Menurut Sa'ad (dalam Pressman 2020), "UML adalah sebuah patokan bahasa untuk menulis kerangka kerja terperinci dari sebuah perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan sebuah sistem perangkat lunak"

UML (*Unified Modelling Language*) juga dapat diartikan sebagai bahasa visual untuk menggambarkan definisi tentang requirement, membuat analisis dan desain serta menggambar arsitektur dalam pemrograman berorientasi objek dengan menggunakan teks-teks pendukung.

2.2.2 Jenis-Jenis Diagram *Unified Modeling Language* (UML)

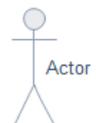
2.2.2.1 Pengertian *Use Case Diagram*

Adapun menurut (Sukanto dan Shalahuddin, 2018), "*Use Case* atau diagram *Use Case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat".

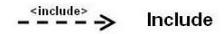
Menurut Sa'ad (dalam Pressman, 2020), "*Use Case Diagram* membantu menentukan fungsi dan fitur dari perangkat lunak". Dalam diagram ini, gambar yang menyerupai boneka kayu mewakili aktor yang berhubungan dengan kategori dari pengguna dan terhubung oleh garis ke *Use Case* yang mereka kerjakan.

Jadi *Use Case* diagram adalah model diagram UML yang digunakan untuk menggambarkan requirement fungsional yang diharapkan dari sebuah sistem. Berikut ini adalah simbol-simbol dari *Use Case* diagram :

Tabel 2.1 Simbol-simbol *Use Case Diagram*

| No | Simbol | Deskripsi |
|----|---|--|
| 1 |  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal-awal frase nama <i>use case</i> |
| 2 |  | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang tapi aktor belum tentu merupakan orang. |



| | | |
|---|---|---|
| 3 | Asosiasi/ <i>Association</i>  | Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki ingeraksi dengan actor. |
| 4 | Ekstensi/ <i>Extend</i>  | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu. |
| 5 | <i>Include</i>  | <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, dan juga melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan. |

Sumber : Haviluddin (2011)

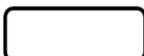
2.2.2.2 Pengertian *Activity Diagram*

Pengertian *Activity Diagram* menurut (Sukamto dan Shalahuddin, 2018) “adalah diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (*software*)”.

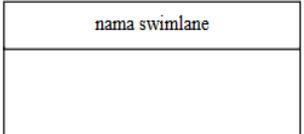
Menurut (Munawar, 2018), “*Activity Diagram* adalah bagian penting dari UML yang menggambarkan aspek dinamis dari system”.

Dapat disimpulkan bahwa *Activity Diagram* merupakan diagram yang menggambarkan aktifitas-aktifitas sistem dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan. Berikut merupakan simbol-simbol dari *Activity Diagram* :

Tabel 2.2 Simbol-simbol *Activity Diagram*

| No | Simbol | Deskripsi |
|----|--|--|
| 1 | Status Awal  | Status awal aktivitas sistem sebuah diagram aktivitas memiliki sebuah status awal. |
| 2 | Aktivitas  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3 | Percabangan/ <i>decision</i> | Asosiasi percabangan dimana jika ada pilihan |



| | | |
|---|--|---|
| |  | aktivitas lebih dari satu. |
| 4 | Penggabungan/ <i>join</i>  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5 | Status Akhir  | Status akhir dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 6 | <i>Swimlane</i>  | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

Sumber : Haviluddin (2011)

2.2.2.3 Pengertian *Class Diagram*

Menurut (Munawar, 2018), “*Class diagram* adalah salah satu jenis diagram pada UML yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. *Class diagram* menggambarkan atribut, operation dan juga constraint yang terjadi pada system”. Sedangkan menurut Sa’ad (dalam Pressman, 2020), “Unsur-unsur utama dari diagram kelas adalah kotak, yang merupakan ikon yang digunakan untuk mewakili kelas dan *interface*”. Setiap kotak dibagi menjadi bagian-bagian horizontal. Bagian atas berisi nama kelas. Bagian tengah berisi daftar atribut kelas dan bagian tengah bawah merupakan operation dari.

Jadi dapat disimpulkan pengertian *Class diagram* adalah salah satu jenis diagram pada UML yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci.

Tabel 2.3 Simbol-simbol *Class Diagram*

| No | Simbol | Deskripsi |
|----|--|---|
| 1 | <i>Generalization</i>  | Relasi antarkelas dengan makna generalisasi spesialisasi (umum khusus). |
| 2 | <i>Nary Association</i>  | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek |



| | | |
|---|------------------------|--|
| 3 | <i>Class</i> | Kelas pada struktur item. |
| 4 | <i>Interface</i> | Sama dengan konsep interface dalam pemrograman berorientasi objek. |
| 5 | <i>Association</i> | Relasi antarclass dengan arti umum, asosiasi biasanya juga disertai dengan <i>Multiplicity</i> . |
| 6 | <i>Dependency</i> | Relasi antarkelas dengan makna kebergantungan antar kelas. |
| 7 | <i>Aggregation</i> | Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>) . |

Sumber : Haviluddin (2011)

2.2.2.4 Pengertian *Sequence Diagram*

Menurut (Munawar, 2018), “*Sequence diagram* adalah salah satu jenis diagram pada UML yang menjelaskan interaksi obyek yang berdasarkan urutan waktu. Diagram ini menunjukkan sejumlah contoh objek dan *message* (pesan) yang diletakkan diantara objek-objek ini di dalam *use case*”.

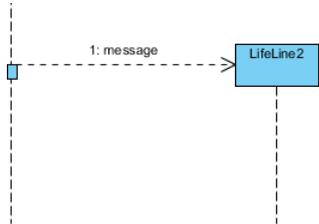
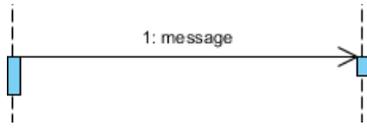
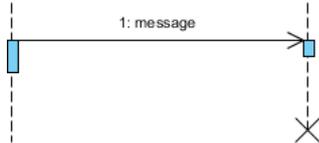
Menurut (Sukanto dan Shalahuddin, 2018) mendefinisikan bahwa “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek”.

Dari pendapat diatas, dapat disimpulkan bahwa *Sequence diagram* adalah salah satu diagram pada UML yang menggambarkan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirim dan diterima antar objek. Berikut ini merupakan simbol-simbol dari *sequence diagram* :

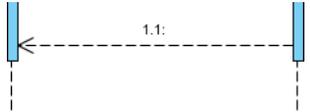
Tabel 2.4 Simbol-simbol *Sequence Diagram*

| No | Simbol | Deskripsi |
|----|------------------|--|
| 1 | Aktor/nama actor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walalupun aktor |



| | | |
|---|---|---|
| |  | dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 2 | <i>LifeLine</i> ----- | Menyatakan kehidupan suatu objek. |
| 3 | Objek <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;">nama objek : nama kelas</div> | Menyatakan objek yang berinteraksi pesan. |
| 4 | Waktu aktif  | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya. |
| 5 | Pesan tipe <i>create</i>  | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| 6 | Pesan tipe call 1 : nama_metode()  | Menyatakan suatu objek memanggil atau operasi atau metode yang ada pada objek lain atau dirinya sendiri. |
| 7 | Pesan tipe <i>send</i>  | Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim. |
| 8 | Pesan tipe <i>destroy</i>  | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri. |



| | | |
|---|---|---|
| 9 | Pesan tipe <i>return</i>  | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |
|---|---|---|

Sumber : Haviluddin (2011)

2.3 Teori Judul

Berikut dibawah ini merupakan beberapa pengertian yang diambil dari judul yang penulis tulis :

2.3.1 Pengertian Aplikasi

Menurut Syani *et al*, (dalam Suryana dan Yulianti, 2019) “aplikasi adalah sebuah perangkat lunak yang berisi sebuah *coding* atau perintah yang dimana bisa diubah sesuai dengan keinginan”.

Menurut Abdurrahman dan Riswaya dalam (Riyowati & Fadlilah, 2019:7) ”Aplikasi merupakan program yang siap dipakai dan digunakan untuk 38 menjalankan perintah *user* dengan tujuan untuk mendapatkan hasil yang tepat dan akurat sesuai dengan tujuan dari pembuatan aplikasi tersebut”. Aplikasi merupakan salah satu proses pemecahan salah satu masalah yang menggunakan teknik komputasi yang diinginkan dalam pemrosesan data.

Berdasarkan pengertian di atas, dapat disimpulkan bahwa pengertian aplikasi adalah alat terapan yang difungsikan secara khusus dan sistematis untuk mengolah data yang menggunakan bahasa pemrograman tertentu untuk melakukan tugas yang diinginkan pengguna.

2.3.2 Pengertian *Internship*

Menurut Fajri (dalam Rosa, 2018), “*Internship* atau yang biasa disebut dengan magang adalah kegiatan dan program yang diadakan secara individu maupun lembaga program magang yang digunakan sebagai sarana dalam memberikan gambaran real dunia kerja”.

Menurut (Setyawan dan Munari, 2020), “Magang merupakan suatu proses pembelajaran yang mengandung unsur belajar sambil bekerja. Biasanya warga belajar sebagai pemegang akan membiasakan diri mengikuti proses pekerjaan yang dilakukan oleh pendidik atau orang yang bekerja disitu”.



Jadi dapat disimpulkan bahwa magang merupakan aktivitas pembelajaran yang memberikan wawasan dan pengalaman praktis kepada mahasiswa mengenai kegiatan riil di suatu lembaga sehingga mahasiswa memiliki kompetensi yang memadai dalam melaksanakan tugas sesuai dengan bidang keahliannya.

2.3.3 Pengertian *Framework*

Pengertian *framework* menurut (Setyawan *et al*, 2019), “kumpulan fungsi-fungsi dasar atau perintah yang biasa digunakan dalam mengembangkan suatu *software*, dengan harapan agar *software* yang dibangun menjadi lebih cepat dan terstruktur”.

Menurut (Raharjo, 2018), “*Framework* adalah suatu kumpulan kode berupa pustaka (*library*) dan alat (*tool*) yang dipadukan sedemikian rupa menjadi satu kerangka kerja (*framework*) guna memudahkan dan mempercepat proses pengembangan aplikasi web”.

2.3.4 Pengertian *CodeIgniter*

Menurut (Supardi, 2018), “*CodeIgniter* merupakan kerangka kerja PHP (*framework* PHP) sehingga pembuatan web dengan PHP menjadi lebih mudah”.

Sedangkan menurut (Sulistiono, 2018), “*CodeIgniter* adalah sebuah aplikasi *open source* yang berupa kerangka kerja atau *framework* untuk membangun *website* menggunakan bahasa pemrograman PHP”.

Dari dua pengertian di atas dapat disimpulkan, bahwa *CodeIgniter* adalah aplikasi *open source* berupa *framework* untuk membangun *website* dengan bahasa pemrograman PHP.

2.3.5 Pengertian *Website*

Menurut (Arkhiansyah & Rasikun, 2018) “*website* adalah kumpulan halaman-halaman”. Yang digunakan untuk menampilkan informasi teks gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Menurut (Sar *et al*, 2019) “*Website* merupakan kumpulan halaman digital yang berisi informasi berupa teks, animasi, gambar, suara dan video atau



gabungan dari semuanya yang terkoneksi oleh internet, sehingga dapat dilihat oleh seluruh siapapun yang terkoneksi jaringan internet.”

Jadi dapat disimpulkan bahwa *website* merupakan halaman informasi yang terkoneksi dengan internet sehingga dapat dilihat oleh di seluruh dunia.

2.3.6 Pengertian Aplikasi Pendaftaran *Internship* dan Mahasiswa Kerja Praktek Berbasis *Website* Menggunakan *Framework Codeigniter* di PT Kilang Pertamina Internasional Refinery Unit III

Pengertian Aplikasi Pendaftaran *Internship* dan Mahasiswa Kerja Praktek Berbasis *Website* Menggunakan *Framework Codeigniter* di PT Kilang Pertamina Internasional Refinery Unit III adalah sebuah aplikasi berbasis *website* untuk PT Kilang Pertamina Internasional Refinery Unit III dalam menjalankan program pendaftaran magang atau Kerja Praktik agar lebih efektif dan efisien dalam melakukan proses rekrutmen calon peserta.

2.4 Teori Program

Berikut dibawah ini merupakan beberapa pengertian dari teori program yang diambil terkait judul yang penulis tulis :

2.4.1 Pengertian HTML

Menurut (Rerung, 2018) “*Hypertext Markup Language (HTML)* merupakan bahasa dasar pembuatan web”. HTML menggunakan tanda (mark), untuk menandai bagian-bagian dari text. HTML disebut sebagai bahasa dasar, karena dalam membuat web, jika hanya menggunakan HTML maka tampilan web terasa hambar.

Sedangkan Menurut (Abdulloh, 2018) “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*”.

Jadi, dapat disimpulkan bahwa HTML merupakan sebuah bahasa pemrograman terstruktur dengan tag-tag tertentu yang digunakan untuk mendesain sebuah halaman website dan ditampilkan menggunakan web browser.

2.4.2 Penulisan HTML

Penulisan HTML sebagai berikut :

<HTML>



```
<head>
<title>Selamat Datang</title>
</head>
<body>
</body>
</HTML>
```

2.4.3 Pengertian CSS

Menurut (Rerung, 2018). Memberikan pengertian bahwa “CSS adalah singkatan dari *Cascading Style Sheets*. Biasanya sering dikaitkan dengan HTML, karena keduanya saling melengkapi dimana HTML ditunjukkan untuk membuat struktur, atau konten dari halaman web. Sedangkan CSS digunakan untuk tampilan dari halaman *website* tersebut.” Menurut (Pahlevi *et al*, 2018), “CSS kepanjangan dari *Cascading Style Sheet* adalah bahasa-bahasa yang merepresentasikan halaman web. Seperti warna, *layout*, dan *font*”

Dapat disimpulkan bahwa pengertian dari CSS adalah sebuah bahasa pemrograman website yang digunakan untuk mengendalikan tampilan halaman web agar lebih seragam dan berstruktur.

2.4.4 Penulisan CSS

2.4.4.1 Penulisan dengan *Inline* CSS

```
<h1 style="color:red;">
H1 dengan CSS
</h1>
```

2.4.4.2 Penulisan dengan *Embedded* CSS

```
<head>
<style
type="text/CSS">
h1{color:red;}
</style>
</head>
```



2.4.4.3 Penulisan dengan *External CSS*

Buat sebuah file CSS, misal `style.CSS`, lalu tulis kode berikut :

```
p{font-family:arial; font-size:small;}h1{color:red;}
```

2.4.5 Pengertian PHP

Menurut (Harani dan Sunandhar, 2020), “PHP adalah bahasa pemrograman *script* sisi server yang didesain untuk pengembangan web”. Bahasa pemrograman PHP sering disebut juga sebagai bahasa pemrograman *server-side*, karena berbeda dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses di web browser (*client*). Menurut (Supono dan Putratama, 2018) mengemukakan bahwa “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”.

Jadi, PHP merupakan bahasa pemrograman *server side scripting* yang bersifat *open source* dan berfungsi untuk membuat aplikasi berbasis web yang berjalan secara dinamis, sehingga dapat terintegrasi dengan database.

2.4.6 Penulisan Program PHP

Cara penulisan skrip PHP ada dua macam, yaitu *Embedded Script* dan *Non Embedded Script*, contoh :

a. *Embedded Script*

```
<HTML>
<head>
</head>
<body>
<?PHP
Echo "Hallo Dunia";
?>
</body>
</HTML>
```

b. *Non Embedded Script*

```
<?PHP
Echo "<HTML>";
Echo "<head>";
```



```

Echo "<title>”Mengenal PHP</title>;
Echo”</head>”;
Echo”<body>”;
Echo”<p> PHP cukup menyenangkan</p>”;
Echo”</body>”;
Echo”</HTML>”;
?>

```

2.4.7 Pengertian JavaScript

Menurut Prasetio (dalam Daniel Rudjiona dan Heru Saputro, 2020) mengatakan bahwa “*JavaScript* adalah sebuah bahasa pemrograman yang digunakan untuk membuat website lebih dinamis dan interaktif”.

Sedangkan menurut (Abdulloh, 2018) “*JavaScript* merupakan bahasa pemrograman web yang pemrosesnya dilakukan disisi *client*.”

Jadi dapat disimpulkan bahwa *JavaScript* secara umum adalah sebuah bahasa pemrograman yang menempel dan bekerja pada halaman HTML.

2.4.8 Penulisan JavaScript

Untuk penulisan kode Javascript adalah sebagai berikut :

```

<html>
<head>
<title>Mencoba JavaScript</title>
<script>
console.log(“Halo, ini Jagoan Hosting”);
</script>
</head>
<body>
<script>
document.write(“Ayo belajar JavaScript!”);
</script>
</body>
</html>

```



2.4.9 Pengertian Sublime Text

Menurut (Pahlevi *et al*, 2018) “Sublime Text merupakan salah satu text editor yang sangat powerful yang dapat meningkatkan produktivitas dan mengembangkan kualitas kode yang tinggi”.

Sedangkan menurut (Syifani dan Dores, 2018) ”Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform *operating system* dengan menggunakan teknologi Phyton API”.

Dari kedua pendapat diatas, penulis menyimpulkan bahwa sublime text merupakan sebuah aplikasi editor yang dapat digunakan untuk mengedit kode dan teks.

2.4.10 Pengertian MySQL

Menurut (Nugroho, 2019) “MySQL merupakan *database* yang paling digemari dikalangan Programmer Web, dengan alasan bahwa program ini merupakan *database* yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data”.

Sedangkan menurut (Rusmawan, 2019) “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS yang *multithread*, *multi-user* dengan sekitar 6 juta instalasi di seluruh dunia”.

Jadi dapat disimpulkan bahwa *Structured Query Language* atau SQL adalah bahasa *query* yang dirancang untuk mempermudah mengambil dan mengelola informasi dalam *database*.

2.4.11 Pengertian XAMPP

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

Menurut (Wicaksono, 2017) adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MySQL di komputer lokal. XAMPP sangat populer di kalangan pengembang web karenamudah digunakan, berfungsi sebagai *server* pengembangan lokal yang stabil, dan mendukung sebagian besar fitur yang dibutuhkan untuk pengembangan web.



Dari pengertian diatas dapat disimpulkan bahwa XAMPP adalah media atau web *server localhost* yang bisa digunakan secara offline untuk pengembangan *website* berbasis PHP dan MySQL.