



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Kadir dan Triwahyuni (2014:2), “Komputer adalah mesin serbaguna yang dapat dikontrol oleh program, digunakan untuk mengolah data menjadi informasi”.

Menurut Yani (2008:1), “Komputer adalah seperangkat alat elektronik yang dihubungkan dengan listrik yang berguna untuk membantu pekerjaan manusia agar lebih mudah, cepat, dan akurat”.

Berdasarkan definisi diatas, dapat disimpulkan bahwa komputer merupakan alat elektronik serbaguna digunakan untuk mengolah data menjadi informasi yang berguna untuk membantu pekerjaan manusia.

2.1.2 Pengertian Perangkat Lunak (*Softaware*)

Menurut Utami dan Asnawati (2015:2), “Perangkat lunak (*software*) adalah program yang berisi kumpulan instruksi untuk melakukan proses pengolahan data”.

Menurut Yung (2010:51), “*Software* adalah sebuah program yang memungkinkan anda melakukan banyak hal dengan *hardware* yang biasanya dapat kita lihat namun tidak dapat kita pegang”.

Berdasarkan definisi diatas, dapat disimpulkan bahwa perangkat lunak (*software*) adalah program yang berisi kumpulan intruksi dan dapat melakukan banyak hal dengan *hardware*.

2.1.3 Pengertian Data

Menurut Aldo, dkk (2020:7), “Data dapat disebut dengan kumpulan dari fakta yang dapat berupa tulisan, angka maupun simbol dari suatu objek yang diamati”.



Pane, Sari, dan Wisacakosono (2020:7), “Data adalah kumpulan keterangan-keterangan atau deskripsi dasar dari suatu hal (objek atau kejadian) yang diperoleh dari hasil pengamatan (observasi) dan dapat diolah menjadi bentuk yang lebih kompleks, seperti; informasi, database, atau solusi untuk masalah tertentu”.

Berdasarkan definisi diatas dapat disimpulkan bahwa data adalah kumpulan dari fakta yang dapat berupa kumpulan keterangan-keterangan atau deskripsi dari suatu objek yang diamati.

2.1.4 Metode Pengembangan Sistem

Penulisan ini menggunakan metode pengembangan perangkat lunak dengan XP (*eXtreme Programming*). Menurut Suryantara (2017:23), menjelaskan bahwa “XP (*eXtreme Programming*) merupakan salah satu metodologi rekayasa perangkat lunak yang banyak digunakan untuk mengembangkan aplikasi oleh para developer”.

Adapun tahap-tahap (*fase*) dalam metode pengembangan XP menurut Suryantara (2017:24-25) adalah sebagai berikut :

1. Perencanaan (*Planning*)

Tahap ini dimulai dengan pemahaman konteks bisnis dari aplikasi, mendefinisikan keluaran (*output*), fitur yang ada pada aplikasi, fungsi dari aplikasi yang dibuat, penentuan waktu dan biaya pengembangan aplikasi, serta alur pengembangan aplikasi.

2. Perancangan (*Design*)

Tahap ini menekankan pada desain aplikasi secara sederhana. Alat untuk mendesain pada tahap ini dapat menggunakan kartu CRC (*Class Responsibility Collaborator*). CRC digunakan untuk pemetaan (membangun) kelas-kelas yang akan digunakan pada diagram *use case*, diagram kelas, dan diagram objek. CRC diperkenalkan oleh Kent Beck dan Ward Cunningham sekitar tahun 1989 sebagai kelengkapan pemrograman berorientasi objek. CRC sebagai cikal bakal yang menjadi kelas pada saat tahap analisis.



3. Pengkodean (*Coding*)

Hal utama dalam pengembangan aplikasi dengan menggunakan XP adalah *pair programming* (dalam membuat program melibatkan 2 atau lebih *programmer*).

4. Pengujian (*Testing*)

Tahap ini memfokuskan pada pengujian fitur-fitur yang ada pada aplikasi sehingga tidak ada kesalahan (*error*) dan aplikasi yang dibuat sesuai dengan proses bisnis pada klien (pelanggan).

2.2 Teori Khusus

2.2.1 Pengertian Pemrograman Berorientasi Objek

Menurut Kroenke (2015:274), “Pemrograman Berorientasi Objek (*Object-oriented programming; OOP*) adalah cara mendesain dan mengkodekan program. OOP sangat berbeda dengan pemrograman tradisional karena memerlukan cara berpikir yang baru mengenai struktur pemrograman”.

Menurut Prabawati (2010:88), “Pemrograman berorientasi objek (*Object-oriented programming- OOP*) adalah suatu cara mengorganisasi program dengan memodelkan objek-objek dunia nyata (seperti benda, sifat, sistem, dan lainnya) ke dalam sebuah bahasa pemrograman.

Berdasarkan pengertian diatas dapat disimpulkan bahwa Pemrograman Berorientasi Objek (*Object-oriented programming; OOP*) adalah cara mendesain dan mengkodekan program dengan memodelkan objek-objek dunia nyata seperti benda, sifat, sistem, dan lainnya) ke dalam sebuah bahasa pemrograman.

2.2.2 Pengertian *Unified Modeling Language* (UML)

Menurut Aziz (2005:116), “UML adalah sekumpulan simbol dan diagram untuk memodelkan *software*. Dengan menggunakan UML, desain *software* dapat diwujudkan dalam bentuk simbol dan diagram, kemudian dapat diterjemahkan menjadi kode program”. Jadi, UML adalah sekumpulan simbol dan diagram yang dapat diterjemahkan menjadi kode program.



2.2.3 Jenis-jenis Diagram UML

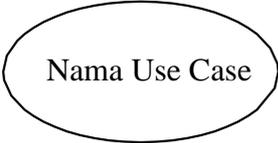
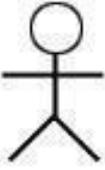
2.2.3.1 Pengertian *Use Case*

Hendini (2016:108), menjelaskan tentang *use case* adalah sebagai berikut:

“*Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut”.

Berikut adalah simbol-simbol yang ada pada diagram *use case* :

Tabel 2.1 Simbol-simbol *Use Case Diagram*

No	Simbol	Keterangan
1		Use case menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan anatar unit atau actor; biasanya dinyatakan dengan menggunakan kata kerja.
2		<i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i> tetapi tidak memiliki kontrol terhadap <i>Use Case</i> .
3		Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.
4		Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.



5		<i>Include</i> , merupakan didalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program
6		<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

Sumber: Hendini (2016:108-109)

2.2.3.2 Pengertian *Class Diagram*

Hendini (2016:111), “*Class diagram* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.”

Ayu dan Permatasari (2018:16), “*Class diagram* dibuat setelah diagram *use case* dibuat terlebih dahulu. Pada diagram ini harus menjelaskan hubungan apa saja yang terjadi antara suatu objek dengan objek lainnya sehingga terbentuklah suatu sistem aplikasi”.

Berikut adalah simbol-simbol yang ada pada *class diagram* :

Tabel 2.2 Simbol-simbol *Class Diagram*

No	Simbol	Keterangan			
1	Kelas <table border="1" style="margin-left: 20px;"> <tr><td>nama_kelas</td></tr> <tr><td>+atribut</td></tr> <tr><td>+operasi()</td></tr> </table>	nama_kelas	+atribut	+operasi()	Kelas pada struktur sistem.
nama_kelas					
+atribut					
+operasi()					
2	Antarmuka/ <i>Interface</i> <div style="text-align: center;"> nama_interface </div>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.			
3	Asosiasi/ <i>association</i> <div style="text-align: center;"> </div>	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .			
4	Asosiasi berarah/ <i>directed association</i> <div style="text-align: center;"> </div>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang			



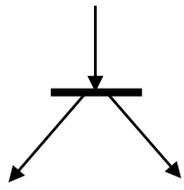
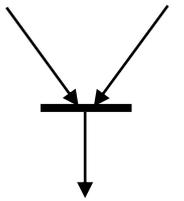
		lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi/ <i>generalitation</i> 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus).
6	Kebergantungan/ <i>dependency</i> 	Kebergantungan antar kelas.
7	Agresasi/ <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole part</i>).

Sumber: Ayu dan Permatasari (2018:16)

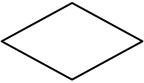
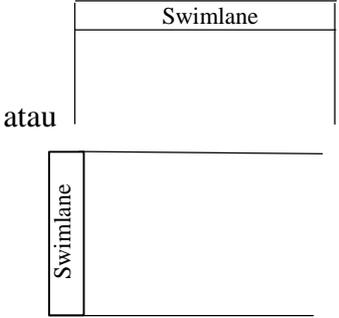
2.2.3.3 Pengertian Activity Diagram

Menurut Hendini (2016:109), “*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis”. Berikut adalah simbol-simbol *activity diagram* :

Tabel 2.3 Simbol-simbol *Activity Diagram*

No	Simbol	Keterangan
1		<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
2		<i>End point</i> , akhir aktivitas.
3		<i>Activites</i> , menggambarkan suatu proses kegiatan/bisnis.
4		<i>Fork/percabangan</i> , digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
5		<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.



6		<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan <i>true</i> atau <i>false</i> .
7		<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

Sumber : Hendini (2016 : 109)

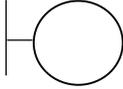
2.2.3.4 Pengertian *Sequence Diagram*

Menurut Hendini (2016:110), “*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek”.

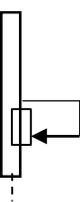
Nurdam (2014:22), “*Sequence Diagram* adalah *tool* yang sangat populer dalam pengembangan sistem infotmasi secara *object-oriented* untuk menampilkan interaksi antar objek”. Selain itu *Sequence Diagram* dapat digunakan sebagai perkakas dalam perancangan antarmuka pemakai.

Berikut adalah simbol-simbol yang ada pada *sequence diagram* :

Tabel 2.4 Simbol-simbol *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
2		<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan form entry dan form cetak.



3		<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
4		<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
5		<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
6		<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
7		<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat <i>activation</i> .

Sumber : Hendini (2016:110)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Pramana (2006:1), “Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game, pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia”.

Menurut Santoso (2005:9), “Aplikasi adalah suatu kelompok *file (form, class, report)* yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait”.



Berdasarkan pengertian diatas dapat disimpulkan bahwa Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas tertentu yang saling terkait.

2.3.2 Konsep Pemesanan

2.3.2.1 Pengertian Pemesanan

Menurut Ramdani, Ginting dan Fajri (2019:57), “Pemesanan adalah suatu aktifitas yang dilakukan oleh konsumen sebelum membeli”.

Menurut Muharram, dkk (2020:7), “Pemesanan merupakan proses atau cara memesan baik berupa barang maupun jasa yang dilakukan antara dua pihak atau lebih”.

Berdasarkan penjelasan diatas, dapat disimpulkan bahwan Pemesanan adalah suatu aktifitas atau proses memesan barang atau jasa yang dilakukan dua pihak atau lebih.

2.3.2.2 Jenis-Jenis Pemesanan

Menurut Surtana (1987:2), Kata *reservation* atau pemesanan dalam dunia pariwisata disebut juga *booking*. Berikut beberapa jenis reservasi atau pemesanan :

1. *New Reservation*, adalah reservasi baru dan dilakukan untuk pertama kali.
2. *Amanded Reservation*, adalah reservasi yang telah dirubah, perubahan biasanya terjadi untuk tanggal kedatangan atau keberangkatan.
3. *Cancelled Resevation*, adalah reservasi/pemesanan yang dibatalkan.
4. *Confirm Reservation* adalah reservasi yang sudah pasti dan dikonfirmasi.
5. *Individual Reservation*, adalah reservasi untuk tamu perorangan.
6. *Group Reservation*, adalah reservasi tamu untuk rombongan.
7. *Confrence Reservation*, adalah reservasi untuk konfrensi atau seminar.

2.3.2.3 Sistem Pemesanan

Menurut Abwandi dan Sudaryatno (2016:9), “Sistem pemesanan jasa mencakup sistem dalam suatu perusahaan jasa yang pada kegiatannya menawarkan sesuatu barang yang belum jadi atau barang yang tidak berwujud. Setelah melalui



proses produksi barang yang dipesan akan baru akan terlihat hasilnya, hasil dari proses produksi sesuai dengan proses pemesanan yang dilakukan oleh konsumen”.

2.3.3 Konsep Paket Wisata

2.3.3.1 Pengertian Paket Wisata

Menurut Utama (dalam Lumanaw 2014:37), “Paket wisata merupakan suatu perjalanan wisata satu atau beberapa tujuan kunjungan yang disusun dari berbagai fasilitas perjalanan tertentu dalam suatu acara perjalanan yang tetap, serta dijual sebagai harga tunggal yang menyangkut seluruh komponen dari perjalanan”.

Menurut Camilleri (dalam Lumanaw 2019:18), “Paket wisata sebagai wisata yang memasukkan semua layanan seperti, pengaturan layanan transportasi dan akomodasi yang dipesan oleh wisatawan dalam satu harga”.

Jadi, dapat disimpulkan bahwa Paket Wisata merupakan suatu perjalanan wisata atau beberapa tujuan kunjungan yang memiliki pengaturan layanan transportasi dan akomodasi yang dipesan oleh wisatawan dalam satu harga.

2.3.3.2 Perencanaan Paket Wisata

Menurut Deskey (2001:8), terdapat beberapa pertimbangan yang harus ditentukan dalam pembuatan rencana sebuah paket wisata antara lain :

1. Pemilihan daerah tujuan wisata.
2. Fasilitas di daerah tujuan wisata.
3. Keunggulan daerah tujuan wisata.
4. Akses ke daerah tujuan wisata.
5. Musim di daerah tujuan wisata.
6. Situasi politik dan keamanan di daerah tujuan wisata.
7. Sistem bea cukai dan keimigrasian di daerah tujuan wisata.
8. Kebijakan harga di daerah tujuan wisata.
9. Jarak tempuh daerah tujuan wisata.



Perencanaan wisata memerlukan tahapan-tahapan. Seluruh tahapan tersebut berkaitan erat dengan aspek-aspek dalam perencanaan. Tahapan-tahapan tersebut juga memerlukan instrument dalam observasi. Perencanaan yang baik haruslah didasarkan akan pertimbangan-pertimbangan rasional dan data-data yang akurat.

2.3.3.3 Jenis Paket Wisata

Ada bermacam-macam jenis paket wisata yang ditawarkan kepada wisatawan. Menurut Suyitno (2001:7), ditinjau dari penyusunannya, paket wisata dapat dibagi menjadi dua jenis, antara lain :

1. *Ready Made Tour*, yaitu paket wisata yang disusun tanpa menunggu permintaan dari calon peserta wisatawan dan disusun oleh tour operator.
2. *Tailored Made Tour*, yaitu paket wisata yang disusun setelah adanya permintaan dari calon peserta wisata yang disusun setelah adanya permintaan dari calon peserta wisata.

Sedangkan menurut Edu Tourism (2010:7) paket wisata dapat dibagi kedalam beberapa jenis paket wisata. Jenis paket wisata tersebut antara lain :

1. *Pleasure Tourism*, yaitu paket wisata disusun untuk tujuan mengisi liburan guna menghilangkan kepenatan sehari-hari.
2. *Recreation Tourism*, yaitu paket wisata yang disusun untuk tujuan memanfaatkan liburan guna pemulihan kesegaran jasmani maupun rohani.
3. *Cultural Tourism*, yaitu paket wisata yang diselenggarakan untuk mengetahui tujuan adat istiadat, gaya hidup dan seni budaya suatu bangsa.
4. *Adventure Tourism*, yaitu paket wisata yang diselenggarakan untuk melatih keberanian dan ketangkasan dengan mengambil resiko yang dapat membahayakan keselamatan jiwa dan dipandu oleh seorang ahli yang berpengalaman.
5. *Sport Tourism*, yaitu paket wisata yang bertujuan untuk kegiatan olahraga.
6. *Business Tourism*, yaitu paket wisata yang bertujuan untuk bisnis atau dagang.
7. *Convention Tourism*, yaitu paket wisata yang bertujuan untuk menghadiri acara rapat, pameran, seminar, konferensi, dan acara lainnya



2.3.4 Pengertian *Website*

Menurut Yuhefizar, Mooduto, dan Hidayat (2009:2), “*Website* adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi”.

Menurut Ginanjar (2014:5), “*Website* adalah rangkaian atau sejumlah halaman diinternet yang memiliki topik saling terkait untuk mempresentasikan suatu informasi”.

Jadi, dapat disimpulkan bahwa *Website* merupakan rangkaian halaman diinternet yang memiliki topik yang terkait dan terdapat sebuah domain yang mengandung suatu informasi.

2.3.5 Pengertian Aplikasi Pemesanan Paket Wisata Pada CV. Pesona Indah Wisata Berbasis *Website*

Berdasarkan pengertian yang telah dijelaskan diatas dapat disimpulkan bahwa aplikasi pemesanan paket wisata berbasis *website* merupakan suatu perangkat lunak yang dibuat untuk melayani kebutuhan atau aktivitas khususnya dalam proses pemesanan perjalanan wisata atau beberapa tujuan kunjungan yang memiliki pengaturan layanan transportasi dan akomodasi yang dipesan oleh wisatawan dalam satu harga.

2.4 Teori Program

2.4.1 Pengertian HTML

Kuswayanto (2006:47), “HTML adalah kependekan dari *Hyper Text Markup Language*. HTML merupakan halaman yang berada pada suatu situs Internet atau *Web*. Jadi, suatu situs terdiri atas beberapa halaman HTML atau *Web page*”.

Enterprise (2016:16), HTML adalah *Hypertext Markup Language* yang artinya adalah sebuah teks berbentuk link dan mungkin juga foto atau gambar yang saat diklik, akan membawa si pengakses internet dari suatu dokumen ke dokumen lainnya”.



Berdasarkan pengertian diatas dapat disimpulkan bahwa HTML merupakan halaman yang berada pada suatu situs internet yang berbentuk link dan mungkin dan mungkin juga foto atau gambar yang saat diklik, membawa pengakses dari suatu dokumen ke dokumen lainnya.

2.4.2 Penulisan HTML

Penulisan HTML adalah sebagai berikut :

```
<html>
<head>
  <title>Selamat Datang</title>
</head>
<body>...
</body>
</html>
```

2.4.3 Pengertian CSS

Juju (2007:10), “CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS, memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda”.

Sutisna (2007:35), ”CSS adalah untuk mempermudah format dan *layout* pada halaman website. Dengan CSS tampilan *website* akan lebih cantik dan konsisten”.

Jadi dapat disimpulkan bahwa CSS merupakan bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen dan membuat *website* akan lebih cantik dan konsisten.

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 Eksternal CSS

Buat sebuah file css, misal style.css, lalu tulis kode sebagai berikut :

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

2.4.5 Pengertian PHP

Menurut Anhar (2010:3), “PHP adalah script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client.”

Kadir dan Triwahyuni (2013:193), “PHP merupakan bahasa pemrograman yang populer untuk membuat aplikasi web. Namun, sebenarnya bahasa pemrograman ini bisa dipakai untuk pembuatan aplikasi-non web. Kepanjangan awalnya adalah *Personal Home Page*. Namun, PHP lebih dikenal sebagai kependekan dari PHP: Hypertext Preprocessor, suatu akronim yang bersifat rekursif”.

Dari pengertian diatas dapat disimpulkan bahwa PHP yaitu *Hypertext Preprocessor* merupakan bahasa pemrograman yang digunakan untuk membuat halaman website yang dinamis dan bisa dipakai untuk pembuatan aplikasi- non web.

2.4.6 Penulisan Program PHP

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

Embedded Script

```

<html>
    <head>
    </head>
    <body>
<?php
    Echo “Hello dunia”;
?>

```



```
</body>
```

```
</html>
```

Non Embedded Script

```
<?php
```

```
    Echo"<html>";
```

```
    Echo"<head>";
```

```
    Echo "<title>Menenal PHP</titile>"
```

```
    Echo"<head>";
```

```
    Echo"<body>";
```

```
    Echo "<p> PHP cukup menyenangkan </p>";
```

```
    Echo "</body>";
```

```
    Echo"</html>";
```

```
?>
```

2.4.7 Pengertian Framework

Menurut Wardana (2010:3), “Framework adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi website, kita harus mengikuti aturan dari framework tersebut”.

Menurut Yudhanto dan Prasetyo (2018:8), “Framework juga dapat diartikan sebagai kumpulan script (terutama class dan function) yang dapat membantu developer/programmer dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke datavase, pemanggilan variable, file, dan lain-lain sehingga pekerjaan developer lebih focus dan lebih cepat dalam membangun aplikasi.”

Dapat disimpulkan dari pengertian diatas bahwa Framework adalah kumpulan perintah atau fungsi yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain yang dapat membantu developer/programmer dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke datavase, pemanggilan variable, file, dan lain-lain sehingga pekerjaan developer lebih focus dan lebih cepat dalam membangun aplikasi.



2.4.8 Pengertian Laravel

Menurut Yudhanto dan Prasetyo (2019:22), “Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT dan dibangun dengan konsep MVC (model view controller). Laravel adalah pengembangan website berbasis MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu”.

2.4.9 Keuntungan Menggunakan Laravel

Habibi, Fakhri, dan Damayanti (2019:5-6) :

Laravel adalah PHP open source framework yang dibangun dengan *model view controller* dan dibekali dengan berbagai macam sintaks.

Beberapa keunggulan yang ditawarkan oleh Laravel adalah sebagai berikut :

1. *Website* menjadi lebih *scalable* (mudah dikembangkan).
2. Terdapat namespace dan tampilan yang dapat membantu anda mengorganisir dan mengatur sumber daya *website*.
3. Proses pengembangan menjadi lebih cepat sehingga menghemat waktu karena Laravel dapat dikombinasikan dengan beberapa komponen dari framework lain untuk mengembangkan *website*.

2.4.10 Pengertian Bootstrap

Pranaya dan Hendra (2019:2), “Bootstrap adalah suatu framework yang memungkinkan developer dapat mengembangkan website dengan mudah dan cepat, developer hanya perlu memanggil class tertentu untuk membuat tombol, panel, table, pesan peringatan, dan lain sebagainya”.

Enterprise (2016:1), “Bootstrap adalah framework front-end yang intuitif dan powerful untuk pengembangan aplikasi web yang lebih cepat dan mudah”.



2.4.11 Pengertian MySQL

Huda (2010:181), “MySQL adalah suatu perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (database management system), database ini multithread, multi-user”.

Sakur (2005:57), “MySQL merupakan salah satu sistem database yang sangat handal karena menggunakan sistem SQL. Pada awalnya, SQL berfungsi sebagai bahasa penghubung antara program database dengan bahasa pemrograman yang kita gunakan. Dengan adanya SQL maka para pemrogram jaringan dan aplikasi tidak mengalami kesulitan sama sekali didalam menghubungkan aplikasi yang mereka buat”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa, MySQL adalah perangkat lunak sistem manajemen basis data yang menggunakan sistem SQL yang awalnya berfungsi sebagai bahasa penghubung antara program database dengan bahasa pemrograman yang kita gunakan. Database ini multithread dan multi-user.

2.4.12 Pengertian XAMPP

Menurut Saputra dan Aprilian (2009:113), “XAMPP adalah perangkat lunak sumber terbuka yang dikembangkan oleh teman-teman Apache. Paket perangkat lunak XAMPP berisi distribusi Apache untuk server Apache, MariaDB; PHP, Dus Peet. Dan itu juga pada dasarnya adalah tuan rumah lokal atau server lokal”.

Wicaksono (2008:7), “XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolahan data MySQL di computer local. XAMPP berperan sebagai server web pada computer anda. XAMPP juga dapat disebut sebuah CPanel server virtual, yang dapat membantu anda melakukan preview sehingga dapat memodifikasi website tanpa harus online atau terakses dengan internet”.

Dari pengertian diatas dapat disimpulkan bahwa XAMPP merupakan software untuk menjalankan website berbasis PHP yang berisi distribusi Apache dan pada dasarnya merupakan server lokal.



2.4.13 Pengertian PhpMyAdmin

Menurut Zaki (2008:97), “PhpMyAdmin adalah MySQL client yang berupa aplikasi web dan umumnya tersedia di server PHP seperti XAMPP maupun server komersial lainnya”.

Rahman (2013:21), “PHPMyAdmin adalah sebuah software berbasis pemrograman PHP yang dipergunakan sebagai administrator MySQL melalui browser (web) yang digunakan untuk manajemen database. PhpMyAdmin mendukung berbagai aktivitas MySQL seperti pengolahan data, table, relasi antar table, dan lain sebagainya”.

Dari pengertian diatas dapat disimpulkan bahwa PhpMyAdmin adalah software berbasis PPHP yang mendukung aktivitas MySQL seperti pengolahan data, relasi table, dan lain sebagainya.

2.4.14 Pengertian Sublime Text

Faris (2020:4), “Sublime adalah text editor yang sangat bagus, dilihat dari fungsi dan fitur yang mendukung beberapa bahasa pemrograman seperti PHP, Phyton, Java, HTML, Ruby, SQL, C, C++, dll”.

Setyawan dan Pratiwi (2019:88), “Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API”.

Dari pengertian diatas dapat disimpulkan bahwa Sublime text merupakan text editor untuk kode atau teks dan memiliki fitur yang mendukung bahasa pemrograman/dapat berjalan di berbagai platform operating sistem.