



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Web

Menurut Sibero (2014:11), “*Web* adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia, dan lainnya pada jaringan internet.”

Menurut Yuhefizer et.al (2008:2) *Web* adalah “sebuah sistem penyebaran informasi melalui internet.”

Menurut Kadir (2003:460), “*Web* merupakan sumber daya internet yang sangat populer dan dapat digunakan untuk memperoleh informasi atau bahkan melakukan transaksi pembelian barang. *Web* menggunakan protokol yang disebut HTTP (*Hypertext Transfer Protocol*) yang berjalan pada TCP/IP (*Transfer Control Protocol/Internet Protocol*).”

Dari definisi diatas maka dapat disimpulkan bahwa, *Web* adalah sumberdaya *Internet* yang digunakan untuk memperoleh informasi dalam bentuk teks, gambar multimedia pada jaringan Internet.

2.1.2 Pengertian Internet (Interconnected Network)

Menurut Sibero (2014:10), “*Internet (Interconnected Network)* adalah jaringan komputer yang menghubungkan antar jaringan secara global, *internet* dapat juga disebut jaringan dalam suatu jaringan yang luas.”

Yuhefizar et.al (2008:36), “*Internet (Interconnected Network)* merupakan suatu sistem yang menghubungkan jaringan dari jaringan-jaringan komputer yang ada di dunia.”

Dari definisi diatas maka dapat disimpulkan bahwa, *Internet* adalah jaringan yang menghubungkan komputer dalam suatu jaringan- jaringan luas mencakup seluruh dunia.



2.1.3 Pengertian Perangkat Lunak

Menurut Rosa dan Shalahuddin (2014:2), “Perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan.”

Pressman (2012:1), menjelaskan tentang perangkat lunak sebagai berikut:

Perangkat Lunak merupakan produk yang dibuat oleh para profesional yang kemudian mendukungnya dalam jangka waktu panjang. Perangkat lunak yang dimaksud mencakup didalamnya program-program komputer yang berjalan didalam suatu komputer dari berbagai ukuran dan berbagai arsitektur, terdapat isi yang akan diperlihatkan ketika program-program komputer berjalan, selain itu terdapat informasi-informasi yang bersifat deskriptif dalam bentuk salinan tercetak dalam bentuk berkas salinan secara maya yang meliputi segala bentuk media elektronik.

Dari definisi diatas dapat disimpulkan bahwa, perangkat lunak (*software*) adalah kumpulan beberapa perintah atau program-program untuk melakukan pekerjaan yang dieksekusi oleh mesin komputer yang memiliki informasi bersifat deskriptif seperti dokumentasi kebutuhan, model desain, dan cara penggunaan.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Menurut Sutabri (2012:147), “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya.”

Menurut Sugiar (2014:83), “Aplikasi adalah program yang dibuat untuk melaksanakan tugas tertentu yang dibutuhkan oleh pengguna komputer (*user*)”.

Dari kedua definisi diatas, dapat disimpulkan bahwa aplikasi adalah Program yang difungsikan secara khusus dan terpadu yang dibuat untuk melaksanakan tugas tertentu yang dibutuhkan oleh pengguna komputer.

2.2.2 Aplikasi Web

Menurut Simarmata (2010:56), “aplikasi web adalah sebuah sistem informasi yang mendukung interaksi pengguna melalui antarmuka berbasis web”.



Menurut Pressman dan lowe (2009:4), “Aplikasi web adalah Perangkat lunak komputer yang memiliki kumpulan instruksi dan data eksekusi yang menyediakan kedua informasi tersebut dan yang memiliki fungsi untuk pengguna Akhir.”

Dari pengertian diatas dapat disimpulkan bahwa, aplikasi *web* adalah suatu aplikasi yang sejak awal dirancang untuk dieksekusi didalam lingkungan berbasis *web*. Perkembangan aplikasi *web* berkembang menjadi lingkungan komputasi canggih yang tidak hanya menyediakan fitur *stand alone* (berdiri sendiri), fungsi komputasi, dan *content* kepada pengguna akhir, tetapi juga terintegrasi dengan *database*.

2.2.3 Pelayanan Operasional

Dalam Kamus Besar Bahasa Indonesia (2008:797), “Pelayanan adalah perihal atau cara melayani, usaha melayani kebutuhan orang lain dengan memperoleh imbalan, kemudahan yang diberikan kepada pihak lain sehubungan dengan jual beli barang atau jasa.”

Dalam Kamus Besar Bahasa Indonesia (2008:394), “Operasional adalah bersifat operasi, yang berhubungan dengan pelaksanaan program kerja”.

Dari pengertian diatas, dapat disimpulkan bahwa Pelayanan operasional adalah usaha melayani kebutuhan orang lain untuk memperoleh imbalan, yang bersifat operasi dan berhubungan dengan pelaksanaan kerja.

2.2.4 Tagihan Listrik

Dalam Kamus Besar Bahasa Indonesia (2008:1374), “Tagihan adalah uang dan sebagainya yang harus ditagih .”

Dalam Kamus Besar Bahasa Indonesia (2008:836), “Listrik adalah daya atau kekuatan yang ditimbulkan oleh adanya gesekan atau melalui proses kimia, dapat digunakan untuk menghasilkan panas, cahaya, atau untuk menjalankan mesin”.



Dari pengertian diatas dapat disimpulkan bahwa, Tagihan Listrik adalah uang yang harus ditagih berdasarkan daya listrik yang digunakan untuk menghasilkan panas, cahaya maupun untuk menjalankan mesin.

2.2.5 Mikro Hidro

Menurut Dwiyanto (2016:408), “Mikro Hidro atau yang dimaksud dengan Pembangkit Listrik Tenaga Mikro Hidro (PLTMH), adalah suatu pembangkit listrik skala kecil yang menggunakan air sebagai tenaga pegeraknya seperti saluran irigasi, sungai atau air terjun alam dengan cara memanfaatkan tinggi terjunan (head) dan jumlah debit air.”

2.2.6 Istilah-Istilah Dalam Kelistrikan

Berikut merupakan Istilah istilah yang digunakan dalam Kelistrikan yang digunakan dalam pembuatan aplikasi pelayanan operasional pembayaran tagihan listrik berbasis *web* yang bersumber dari PLN :

1. KW (kilowatt) adalah satuan daya nyata dan kVA (kilovolt ampere) adalah satuan daya semu (atau daya nyata ditambah *active power*). Singkatnya, kVA adalah perkalian antara Voltase dengan Amper.
2. Biaya Beban (BB) merupakan komponen biaya dalam rekening listrik yang besarnya tetap, dihitung berdasarkan daya tersambung.
3. Biaya Keterlambatan (BK) adalah biaya yang dibebankan pada pelanggan karena tidak memenuhi kewajiban membayar tagihan PLN tepat pada waktunya.
4. Biaya Penyambungan (BP) adalah biaya yang harus dibayar kepada PLN oleh calon pelanggan atau pelanggan untuk memperoleh penyambungan baru atau tambah daya.
5. Daya Terpasang adalah besar daya yang disepakati oleh PLN dan pelanggan dalam Perjanjian Jual Beli Tenaga Listrik yang menjadi dasar perhitungan biaya beban.



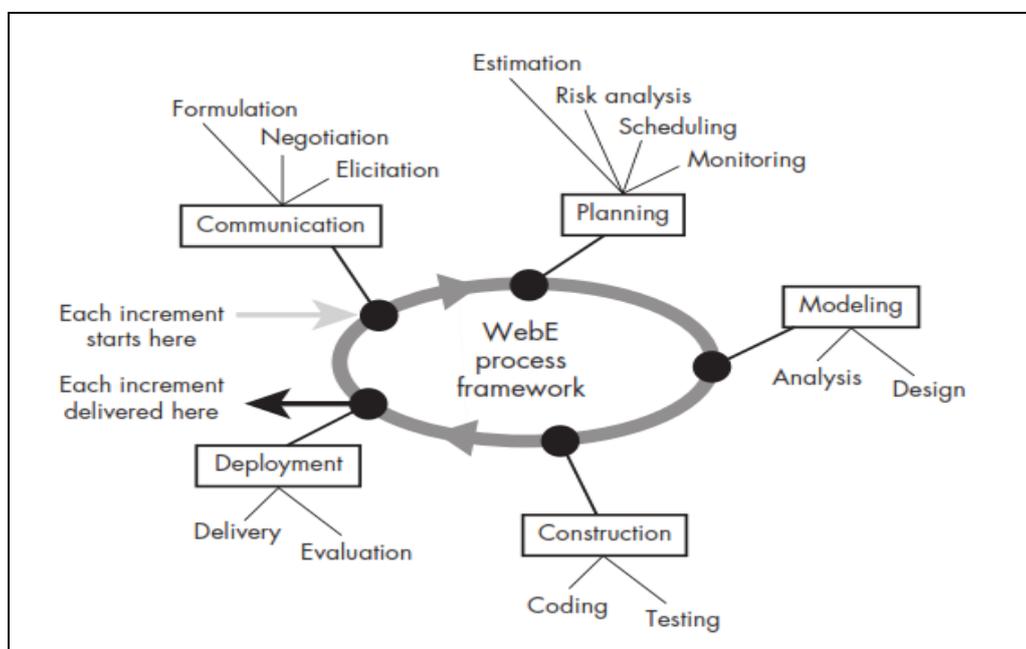
2.2.7 Web Engineering

Menurut Mustakim et.al (2016:2), “*Web Engineering* merupakan gabungan antara *web publishing* (suatu konsep yang berasal dari *printed publishing*) dan aktifitas rekayasa perangkat lunak karena desain sebuah aplikasi *web* menekankan pada desain grafis, desain informasi, teori *hypertext*, desain sistem dan pemrograman.

Simarmata (2010:1), menjelaskan tentang *Web Engineering* sebagai berikut :

Sebuah aplikasi yang menggunakan pendekatan sistematis, disiplin, dan terukur untuk pengembangan operasi pemeliharaan aplikasi berbasis *web* (*Web-based application*). *Web Engineering* atau sering dikenal dengan *Rekayasa Web* yang mengadaptasi *Rekayasa Perangkat Lunak* (*Software Engineering*) dalam hal konsep dasar yang menekankan pada aktifitas teknis dan manajemen dengan perubahan dan penyesuaian.

Berikut merupakan tahapan kerangka proses *Web Engineering* yang dapat dilihat pada Gambar 2.1



(Sumber: Pressman dan Lowe, 2009:26)

Gambar 2.1 Tahapan-Tahapan Kerangka Proses Web Engineering



Pressman dan Lowe (2009:26), menjelaskan tentang tahapan-tahapan kerangka proses *Web Engineering* sebagai berikut:

1. *Comunication*

Dalam proses *Web Engineering* komunikasi karakter terwujud dengan tiga tindakan yaitu *formulation*, *elisation*, dan *negosiation*. Formulasi mendefinisikan bisnis dan organisasi konteks untuk aplikasi *web*. Selain itu, para *stakeholder* yang diidentifikasi adalah perubahan potensial dalam lingkungan bisnis atau persyaratan diperkirakan, integrasi antara aplikasi *web* dan aplikasi bisnis lainnya, *database*, dan fungsi yang didefinisikan. *Elisitasi* merupakan kegiatan persyaratan pengumpulan melibatkan semua *stakeholder*. Jalur maksud adalah untuk menggambarkan masalah yang aplikasi *web* adalah untuk memecahkan (bersama dengan persyaratan dasar untuk aplikasi *web*) menggunakan informasi terbaik yang tersedia. Selain itu, dilakukan usaha untuk mengidentifikasi area ketidak pastian dan dimana potensi perubahan akan terjadi. *Negosiasi* diperlukan untuk mendamaikan perbedaan antara berbagai pemangku kepentingan untuk proyek tersebut.

2. *Planing*

Keseluruhan aplikasi *web* bertahap yaitu diidentifikasi dan perencanaan proyek singkat. untuk aplikasi *web* berikutnya yang akan dikerahkan untuk dibuat. Sumberdaya diestimasikan untuk pengembangan, mempertimbangkan risiko, tugas yang dipilih dan dijadwalkan, dan pelacakan dan pemantauan proyek dimulai.

3. *Modeling*

Analisis rekayasa dan desain perangkat lunak disesuaikan dengan perkembangan tugas aplikasi *web* konvensional, bergabung, dan kemudian menyatu ke dalam *Web Engineering* kegiatan modeling. Tujuannya adalah untuk mengembangkan analisis dan desain model tangkas bahwa persyaratan mendefinisikan dan pada saat yang sama adalah aplikasi *web* yang akan memuaskan mereka.

4. *Construction*

Alat *Web Engineering* dan teknologi yang diterapkan untuk membangun aplikasi *web* yang telah dimodelkan. Setelah pengembangan aplikasi *web* telah dibangun, serangkaian tes cepat yang dilakukan untuk memastikan bahwa kesalahan dalam desain (misalnya, kesalahan dalam konten, arsitektur, antarmuka, dan navigasi) yang terungkap. pengujian tambahan alamat karakteristik aplikasi lainnya.



5. Deployment

Aplikasi *web* dikonfigurasi untuk lingkungan operasional. Hal ini kemudian disampaikan kepada pengguna akhir dan periode evaluasi dimulai. umpan balik evaluasi disajikan kepada tim *Web Engineering*, dan pengembangan yang dimodifikasi seperti yang diperlukan.

2.3 Teori Khusus

2.3.1 Unified Model Language (UML)

Menurut Mustakim et.al (2016:4), “UML adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.”

Menurut Nugroho (2005:16), “UML adalah notasi-notasi yang digunakan sebagai pemodelan visual untuk sistem informasi atau perangkat lunak. Pemodelan visual adalah proses penggambaran informasi-informasi secara grafis dengan notasi notasi baku yang telah disepakati.”

Beberapa diagram dalam UML yang akan digunakan dalam pembuatan Laporan Akhir ini antara lain sebagai berikut:

2.3.1.1 Diagram Usecase (*Usecase Diagram*)

Menurut Indrajani (2015:45), “*Usecase diagram* adalah suatu diagram yang berisi *usecase*, *actor* serta *relationship*, yang merupakan titik awal dalam menganalisis kebutuhan sistem pada saat perancangan.”

Menurut Nugroho (2005:19), “*Usecase diagram* bersifat statis yang memperlihatkan himpunan *usecase* dan *actor* untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan dan diharapkan pengguna.”

Diagram Usecase memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*, sedangkan Usecase merupakan konstruksi untuk



mendesripsikan bagaimana sistem akan terlihat di mata pengguna dan aktor mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Usecase diagram dapat digunakan untuk menentukan kebutuhan apa saja yang diperlukan suatu sistem, jadi dapat digambarkan dengan detail bagaimana suatu sistem memproses atau melakukan sesuatu.

Berikut Merupakan Simbol-simbol Diagram Usecase yang dapat dilihat pada Tabel 2.1

Tabel 2.1 Simbol-Simbol Diagram Usecase

| No | Gambar | Nama | Keterangan |
|----|--------|--------------------------------|--|
| 1 | | Aktor/ actor | Aktor menggambarkan pengguna dari sistem, dapat berupa manusia atau sistem terotomatisasi lain yang berinteraksi dengan sistem lain untuk berbagi, mengirim dan menerima informasi |
| 2 | | Usecase | Simbol ini menggambarkan interaksi antara aktor dengan software aplikasi tersebut |
| 3 | | <i>System Boundary</i> | Menggambarkan batasan antara sistem dengan aktor |
| 4 | | Asosiasi/ <i>Asosiation</i> | Menggambarkan hubungan antara aktor dan usecase |

(Sumber: Indrajani 2015:46)

2.3.1.2 Diagram Aktivitas (*Activity Diagram*)

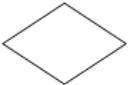
Menurut Nugroho (2005:61), "*Activity Diagram* adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu Usecase. *Activity diagram* digunakan untuk memodelkan aspek dinamis dari sistem. *Activity*



diagram secara *esensial* mirip diagram alir (*flowchart*), memperlihatkan aliran kendali dari suatu aktivitas ke aktivitas lainnya.”

Menurut Rosa dan Salahudin (2014:161), “Diagram aktivitas (*Diagram Activity*) menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.” Berikut merupakan simbol-simbol yang disunukan dalam Diagram Aktivitas yang dapat dilihat pada tabel 2.3

Tabel 2.2 Simbol-simbol Diagram Aktivitas

| No | Simbol | Nama | Keterangan |
|----|--|--------------|---|
| 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 | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| 4. |  | Penggabungan | Asosiasi penggabungan, lebih dari satu aktivitas digabung menjadi satu |
| 5. |  | Status akhir | Status akhir yang dilakuka sistem. |
| 6. |  Atau  | Swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

(Sumber: Rosa dan Salahudin, 2014:161)



2.3.1.3 Diagram Sekuensial (*Sequence Diagram*)

Diagram Sekuensial adalah interaction diagram yang memperlihatkan *event-event* yang berurutan sepanjang berjalannya waktu, dibaca dari atas kebawah, Diagram Sekuen akan menggambarkan aliran-aliran pada suatu usecase.

Indrajani (2015:50), menjelaskan tentang *sequence diagram* sebagai berikut:

Sequence Diagram merupakan suatu diagram yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi dan pesan yang ditukar dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem waktu atau urutan gunakan *Sequence Diagram*.

Berikut simbol yang digunakan dalam Diagram Sekuensial pada yang dapat dilihat pada Tabel 2.2

Tabel 2.3 Simbol-Simbol Diagram Sekuensial

| No | Simbol | Nama | Keterangan |
|----|--------|-------------------------|---|
| 1. | | <i>LifeLine</i> | Objek <i>entity</i> , yang saling berinteraksi |
| 2. | | Aktor | Menggambarkan aktor yg terlibat |
| 3. | | <i>Message (call)</i> | Menggambarkan alur <i>message</i> yang merupakan kejadian dari objek pengiriman <i>lifeline</i> ke objek penerima <i>lifeline</i> |
| 4. | | <i>Message (return)</i> | Menggambarkan alur pengembalian message ke objek pemanggil dan tanda bahwa objek penerima telah menyelesaikan prosesnya |
| 5. | | <i>Activation</i> | <i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada <i>lifeline</i> . Mengindikasikan sebuah obyek yang melakukan sebuah aksi. |

(Sumber: Indrajani 2015:51)



2.3.1.4 Diagram Kelas (*Class Diagram*)

Menurut Nugroho (2005:110) “*Class diagram* adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/ perangkat lunak yang sedang dikembangkan, dalam *Class Diagram* memberi gambaran diagram statis tentang sistem/ Perangkat lunak dengan relasi-relasi yang ada didalamnya.”

Menurut Indrajani (2014:49) “*Class diagram* adalah diagram yang digunakan untuk menggambarkan perbedaan antara *class-class* hubungan antar class dan dimana sub sistem *class*, pada *class diagram* terdapat nama *class*, *attribut* dan *operation*.”

Berikut merupakan simbol-simbol pada Diagram Kelas yang digunakan dalam menggambarkan Aplikasi yang akan dibuat pada tabel 2.5

Tabel 2.4 Simbol-simbol Diagram Kelas

| No | SIMBOL | NAMA | KETERANGAN |
|----|--------|-----------------------------|---|
| 1. | | Kelas | Kelas pada struktur sistem |
| 2. | | Antarmuka /Interface | Sama dengan konsep interface dalam pemrograman berorientasi obyek |
| 3. | | Asosiasi/ association | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |
| 4. | | Asosiasi berarah/ directed | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |
| 6. | | Generalisasi | Relasi antar kelas dengan makna generalisasi spesialisasi (umum – Khusus) |
| 7. | | Kebergantungan / dependency | Kebergantungan antarkelas |
| 8. | | Kebergantungan / dependency | Relasi antarkelas dengan makna semua bagian (whole-part) |

(Sumber: Rosa dan Salahudin, 2014:141)



2.4 Teori Program

2.4.1 Basis Data (*Database*)

Menurut Indrajani (2015:70), “Basisdata adalah sekumpulan data terintegrasi yang secara logika saling berhubungan.”

Menurut Sutanta (2011:35), “Basisdata adalah sekumpulan dari berbagai macam-macam tipe record yang memiliki hubungan antar-record dan rincian data terhadap obyek tertentu.”

Menurut Madcoms (2010:13), “*Database* adalah wadah atau tempat yang digunakan untuk mengumpulkan atau megorganisasi data atau informasi.”

Dari ketiga definisi diatas, dapat disimpulkan bahwa basis data adalah kumpulan data yang disimpan didalam wadah tertentu secara sistematis untuk memperoleh informasi dari basis data tersebut.

2.4.2 *Hypertext Markup Language* (HTML)

Menurut Jubile Enterprise (2015:135), “*Hypertext Markup Language* (HTML) adalah *script* pemrograman yang mengatur bagaimana menyajikan informasi di dunia *internet* dan bagaimana informasi itu membawa kita dari satu tempat ke tempat lainnya.”

Menurut Saputra et.al (2012:1), “*Hypertext Markup Language* (HTML) adalah suatu bahasa pemograman *hypertext* yang memiliki fungsi untuk membangun kerangka ataupun format *web* berbasis HTML.”

Dari kedua definisi diatas, dapat disimpulkan bahwa *Hypertext Markup Language* (HTML) adalah suatu bahasa pemograman *hypertext* yang memiliki fungsi untuk membangun kerangka ataupun format *web* berbasis HTML yang mengatur bagaimana menyajikan informasi di dunia internet.

2.4.3 MySQL

Menurut Saputra et.al (2012:7), “MySQL merupakan database yang bekerja menggunakan SQL Language (*Structure Query Language*).”



Menurut Sidik (2014:333), “MySQL merupakan *software* database yang termasuk paling populer di lingkungan *linux*, kepopuleran ini karena ditunjang performasi *query* dari databasenya yang saat itu bisa dikatakan paling cepat, dan jarang bermasalah.”

Dari kedua definisi diatas dapat disimpulkan bahwa, MySQL sistem manajemen *database* SQL yang bersifat open source dan paling populer saat ini.

2.4.4 PHP Hypertext Preprocessor (PHP)

2.4.4.1 Pengertian PHP Hypertext Preprocessor (PHP)

Menurut Sidik (2014:4), “PHP secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server web*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan *editor* teks atau *editor* HTML, dikenal juga sebagai bahasa pemrograman *server side*.”

Menurut Raharjo (2015:315) “PHP adalah bahasa pemrograman yang dibuat oleh Rasmus Lerdorfdan dirilis pada tahun 1995, yang umumnya digunakan untuk mengembangkan aplikasi berbasis Web.”

Dari definisi diatas dapat disimpulkan bahwa, PHP *Hypertext Preprocessor* dikenal sebagai bahasa pemrograman *server side* yang membuat dokumen HTML secara *on the fly* yang dieksekusi dari *server web* yang difungsikan untuk membangun suatu *web* dinamis atau aplikasi berbasis *web*.

2.4.4.2 Script Hypertext Preprocessor (PHP)

Kadir, Abdul (2008:3), skrip *Hypertext Preprocessor* berkedudukan sebagai tag dalam Bahasa HTML (*Hypertext Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. Adapun kode berikut contoh kode PHP (*Hypertext Preprocessor*) yang berada dalam kode HTML.

```
<HTML>
```

```
<HEAD>
```



```

<TITLE>Latihan Pertama</TITLE>
</HEAD>
<BODY>
Selamat belajar PHP</BR>
<?php
Printf (“Tgl. Sekarang: %s”, Date (“d F Y”));
?>
</BODY>
</HTML>

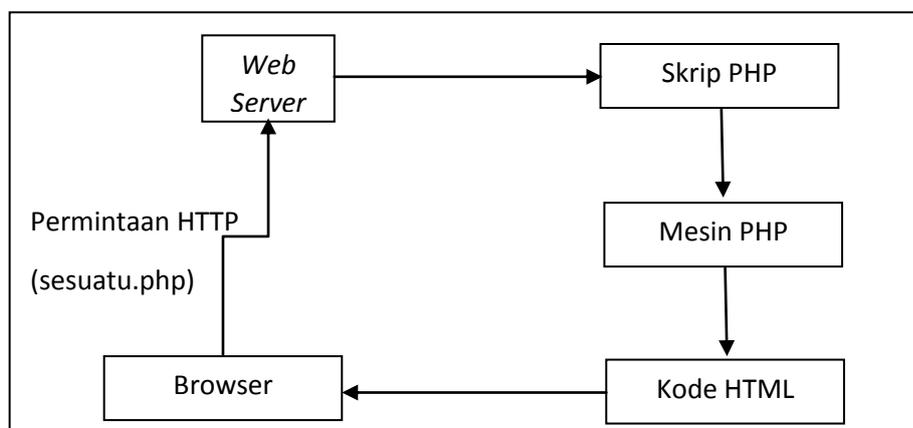
```

Kode *Hypertext Preprocessor* (PHP) diawali dengan `<?PHP` dan diakhiri dengan `?>`. Pasangan kedua kode inilah yang berfungsi sebagai tag kode PHP. Berdasarkan tag inilah, pihak server dapat memahami kode PHP dan kemudian memprosesnya.

2.4.4.3 Konsep kerja PHP

Menurut Kadir (2008:4) menjelaskan tentang Skema kerja PHP sebagai berikut:

Model kerja HTML (*Hypertext Markup Language*) diawali dengan permintaan suatu halaman web oleh *browser*. Berdasarkan URL (*Uniform Resources Localator*) atau dikenal dengan sebutan alamat internet, *browser* merupakan alamat dari web server, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*. Selanjutnya, *web server* akan mencarikan file yang diminta dan memberikan isinya ke *web browser*. *Browser* yang mendapatkan isinya segera melakukan proses penerjemahan kode HTML (*Hypertext Markup Language*) dan menampilkan ke layar pemakai. Sedangkan model kerja PHP (*Hypertext Preprocessor*) pada prinsipnya serupa dengan kode HTML (*Hypertext Markup Language*). Hanya saja, ketika berkas PHP (*Hypertext Preprocessor*) yang diminta didapatkan oleh web server, isinya segera dikirimkan ke mesin PHP (*Hypertext Preprocessor*) dan mesin inilah yang memproses dan memberikan hasilnya ke *web server*. Selanjutnya, *web server* menyampaikan ke klien.



(Sumber: Kadir, 2008:6)

Gambar 2.2 Skema PHP (*Hypertext Preprocessor*)

2.4.5 Cascading Style Sheet (CSS)

Koesheryatin (2014:101), “*Cascading Style Sheet* adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu *website*, baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan. Pada umumnya CSS digunakan untuk memformat halaman *web* yang ditulis dengan HTML.”

Madcoms (2012:88, “*Cascading Style Sheet (CSS)* adalah suatu kumpulan kode-kode untuk memformat dan mengendalikan tampilan isi dalam suatu halaman *web*.”

Dari kedua definisi diatas dapat disimpulkan bahwa CSS, kode-kode untuk mengatur tampilan web dan memformat halaman Web yang ditulis dengan HTML baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan.

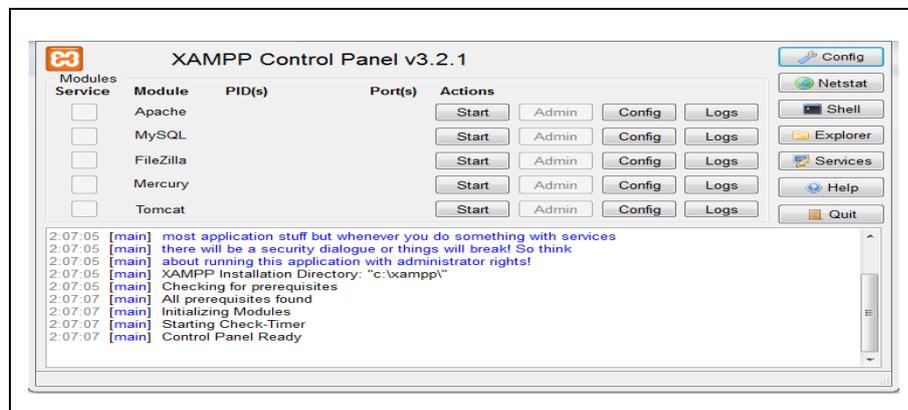
2.4.6 XAMPP

Raharjo (2015:25), menjelaskan tentang XAMPP sebagai berikut:

XAMPP adalah kompilasi *software* yang membungkus *Apache HTTP Server*, *MySQL*, *PHP* dan *Pearl*. Dengan menggunakan instalasi paket



XAMPP, instalasi paket *software* yang dibutuhkan untuk proses pengembangan *Web* (Apache HTTP Server, MYSQL, PHP) dapat dilakukan dengan sangat mudah tanpa harus dilakukan secara terpisah.



(Sumber: Raharjo, 2015:29)

Gambar 2.3 Control Panel XAMPP

2.5 Black Box testing

Testing atau pengujian bertujuan untuk menemukan kesalahan, sehingga dapat diketahui kesalahan- kesalahan yang terjadi dan kemudian dapat dilakukan perbaikan sehingga aplikasi yang dikembangkan sesuai dengan yang diharapkan

Menurut mustaqbal et.al (34:2015) menjelaskan tentang *Black Box Testing* sebagai berikut:

Berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program. *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*.

Black Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.



2.6 Referensi Jurnal

Beberapa referensi jurnal yang digunakan dalam tugas akhir ini adalah sebagai berikut:

1. Aplikasi Web Logbook Bimbingan Tugas Akhir Mahasiswa Stmik Jenderal Achmad Yani Yogyakarta

Melakukan aktifitas bimbingan Tugas Akhir mahasiswa maupun dosen pembimbing membutuhkan sebuah catatan kontak (*Logbook*) namun Pelaksanaan prosedur standar seperti pencatatan formulir bimbingan tidak terorganisasi dengan baik atas faktor ketidak sengajaan sehingga dapat menurunkan kualitas bimbingan. Hasilnya Perancangan sistem *Logbook* tugas akhir dengan menggunakan metode *web engineering* dan diimplementasikan menggunakan bahasa pemrograman PHP dengan Framework CodeIgniter dan pattern HMVC (Hierarchical Mode View Controller), merupakan solusi untuk membangun sebuah sistem yang bisa dilakukan secara cepat dan memungkinkan untuk merubah kembali perangkat lunak agar sesuai dengan kebutuhan sistem *Logbook* bimbingan yang ada di STMIK Jend. A. Yani.

2. Sistem Informasi Budi Daya Jamur Menggunakan Metode Web Engineering

Informasi Belum adanya sistem informasi proses budi daya jamur ya dapat diakses petani. Sistem ini diperlukan petani untuk membuat daya tarik harga jual yang tinggi karena jamur merupakan sumber pangan yang bergizi. Metode yang digunakan dalam penelitian atau pengembangan web ini adalah Web engineering merupakan rekayasa web yang mengadaptasi rekayasa perangkat lunak dalam hal konsep dasar yang menekankan pada aktifitas teknis dan manajemen. Hasilnya Hubungan antara perangkat lunak yangdirancang dengan lingkungan eksternalnya sangat membantu memutuskan bagaimana seharusnya menyediakan fungsionalitas system yang diminta dan bagaimana



menstruktur system sehingga dapat berkomunikasi secara efektif dengan lingkungannya khususnya pembudiyayaan jamur.

3. Membangun Web Engineering Puzzle Research Data Mining Menggunakan Model Navigational Development Technique

Masalahny adalah adanya sebuah perkumpulan mahasiswa yang mendalami riset pada bidang data mining yang disebut dengan *Puzzle Research Data Mining* . Penelitian terus dilakukan dibidang Data Mining dengan target-target yang sangat tinggi sebagai salah satu capaian keberhasilan mata kuliah tersebut serta proyeksi akreditasi yang lebih baik. Akan tetapi bagian Program studi dan fakultas belum memiliki ruang publik guna menyimpan data penelitian tersebut. Hasilnya bahwa *Web Engineering* dengan konsep *Modified Waterfal* dan teknik NDT mampu memberikan kemudahan dari beberapa proses yang berkaitan dengan requirement, kegunaan dan implementasi aplikasi.

4. Implementasi Metode Web Engineering Dalam Perancangan Sistem Informasi Penerimaan Mahasiswa Baru dan Tes Online.

Masalahnya dalam aktivitas pendaftaran mahasiswa dapat dilakukan secara online, namun belum ada fasilitas bagi calon mahasiswa yang mendaftar dari luar daerah untuk melakukan tes secara *online* dan bisa langsung melakukan tes dari tempat dimana mereka berada melalui jaringan internet. Berdasarkan hasil penelitian yang telah dilakukan, didapatkan sebuah sistem informasi yang dapat berjalan pada jaringan *internet*. Sistem informasi Penerimaan Mahasiswa Baru (PMB) berbasis *web* ini berguna bagi calon mahasiswa untuk dapat mendaftar di Universitas melakukan tes ujian masuk secara *online*. Penelitian ini menggunakan metode web engginering sehingga mendapatkan hasil yang memiliki kualitas Tinggi yang dapat mengefisienkan pekerjaan Staf penerimaan mahasiswa baru, mempermudah mahasiswa luar daerah melakukan test online.



5. Rekyasa E-Market Untuk Kelompok Usaha Pemuda Binaan Dinas Pemuda Dan Olahraga Provinsi Jawa Tengah Sebagai Upaya Peningkatan Pemasaran Dan Penjualan Produk UMKM

Masalahnya Adalah makin tingginya faktor persaingan dan mahalnya proses marketing secara manual, maka dibutuhkan penawaran penjualan secara online. Hasilnya E-market yang dibangun secara online meggunakan metode *web Engginering* memberikan kemudahan kepada pelanggan untuk melakukan pembelian barang secara online sehingga pelanggan tidak perlu datang ke lokasi dan memberikan kemudahan bagi kelompok pemuda untuk memasarkan produk yang dihasilkan sehingga dapat meningkatkan hasil penjualan.