

BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Kadir (2017:2) menyatakan “Komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia.”

Krisbiantoro (2016:2) menyatakan “Komputer merupakan alat yang dipakai untuk mengolah memproses data menurut perintah yang telah dirumuskan.”

Kesimpulannya, Komputer adalah alat elektronik yang dapat menerima dan mengelola data sehingga menghasilkan informasi yang berguna bagi banyak orang.

2.1.2 Pengertian Data

Nofriansyah dan Nurcahyo (2015:109) menyatakan “Data adalah kumpulan fakta yang terekam atau entitas yang tidak memiliki arti dan selama ini terabaikan.”

Reksoatmodjo (2018:2) menyatakan “Data diidentikan dengan fakta tentang objek dan kejadian yang dapat direkam atau di simpan pada media computer.”

Kesimpulannya, Data adalah catatan atas kumpulan fakta untuk dijadikan keterangan.

2.1.3 Pengertian Sistem

Indrajani (2015:69) menyatakan “Sistem adalah sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan bersama dengan menerima input serta menghasilkan output dalam proses transformasi yang teratur.”

Kristanto (2018:3) menyatakan “Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.”

Kesimpulannya, Sistem adalah sebuah komponen yang saling berkaitan untuk mencapai proses dari awal sampai ke tahap akhir.

2.1.4 Pengertian Database

Enterprise (2017:2) menyatakan “Database merupakan kumpulan tabel-tabel yang berisi data-data yang saling berkaitan.”

Setiawan (2015:25) menyatakan “Database adalah susunan record data operasional lengkap dari suatu organisasi atau perusahaan yang diorganisir dan disimpan secara integrasi dengan menggunakan metode tertentu sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna.”

Kesimpulannya, Database adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis dan saling berhubungan satu dengan yang lain yang tujuan utamanya adalah memelihara data yang sudah ada yang diolah.

2.1.5 Pengertian Perangkat Lunak (*Software*)

Kadir (2017:2) menyatakan “Perangkat Lunak adalah intruksi-intruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai.”

Rosa dan Shalahuddin (2016:2) menyatakan “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Kesimpulannya, Perangkat lunak adalah suatu program komputer yang tersusun atas perintah-perintah atau fungsi-fungsi untuk menjalankan suatu tugas tertentu.

2.1.6 Pengertian Internet

Hidayatullah dan Khairul (2017:1) menyatakan “Internet adalah jaringan global yang menghubungkan komputer-komputer di seluruh dunia.”

Pratama (2014:37) menyatakan “Internet (*international networking*) secara umum didefinisikan sebagai jaringan terbesar di dunia yang menghubungkan semua jaringan komputer yang ada (*Intranet, WAN, MAN, LAN, dan lain-lain*)

beserta dengan semua komputer, perangkat terhubung (*Smartphone, tablet, switch, router* dan perangkat penghubung lainnya), serta pengguna komputer itu sendiri, ke dalam satu wadah jaringan komputer dunia.”

Kesimpulannya, Internet adalah interkoneksi jaringan komputer skala besar hingga meliputi jutaan komputer di dunia secara global yang dihubungkan dengan protokol khusus.

2.1.7 Model Pengembangan Perangkat Lunak

Rosa dan Shalahuddin (2018:28) menjelaskan bahwa model *Waterfall* sering disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.



d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau pemeliharaan (*maintance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2 Teori Khusus

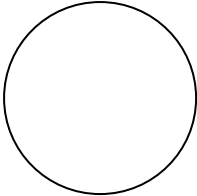
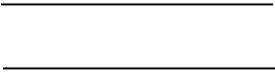

2.2.1 Pengertian *Data Flow Diagram (DFD)*

Indrajani (2015:30) menyatakan “DFD adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut.”


Rosa dan Shalahuddin (2016:70) menyatakan “*Data Flow Diagram (DFD)* adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*ouput*).”

Adapun notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
1.		Proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program Catatan : Nama yang diberikan pada sebuah proses biasanya berupa kata kerja
2.		File basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harusnya sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram (ERD)</i> , <i>Conceptual Data Model (CDM)</i> , <i>Physical Data Model (PDM)</i>) Catatan : Nama yang diberikan pada sebuah penyimpanan biasanya kata benda
3.		Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan Catatan : Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda

Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
4.		Aliran data: merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses masukan (<i>input</i>) atau keluaran (<i>output</i>)

(sumber: Rosa dan Shalahuddin, 2016:71-72)

Rosa dan Shalahuddin (2016:72-73) menjelaskan, Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-breakdown menjadi DFD Level Modul mana saja yang harus di-breakdown lebih lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-breakdown lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul ada DFD Level 1 yang di-breakdown.



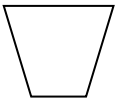

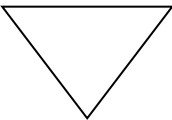
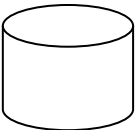

4. Membuat DFD Level 3 dan seterusnya

DFD Level 3,4,5 dan seterusnya merupakan breakdown dari modul pada DFD di atasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

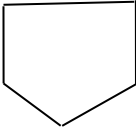
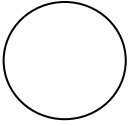

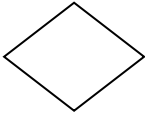
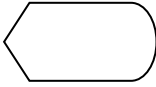

2.2.2 Blockchart

Kristanto (2018:75) menyatakan “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”. Adapun simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-Simbol dalam *Blockchart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/ bendel/ berkas atau cetakan.
2.		Multi dokumen
3.		Proses Manual
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>Storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.

Lanjutan Tabel 2.2 Simbol-Simbol dalam *Blockchart*

No	Simbol	Keterangan
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (<i>Decision</i>).
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukkan data secara manual.




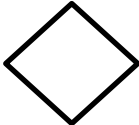


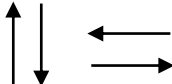

(Sumber: Kristanto 2018:75-77)

2.2.3 Flowchart

Mundzir (2018:843) menyatakan “*Flowchart* adalah sebuah bagan atau aliran dari sesuatu, dan sesuatu itu dapat juga berupa aliran proses.”

Indrajani (2015:30) menyatakan “*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program.”

Tabel 2.3 Simbol-Simbol *Flowchart*

No.	Simbol	Arti
1.		Terminal menyatakan awal atau akhir dari suatu algoritma
2.		Menyatakan proses
3.		Menyatakan masukan dan keluaran.
4.		Menyatakan decision (keputusan) yang digunakan untuk penyeleksian kondisi di dalam program.
5.		Menyatakan penyambung kehalaman lainnya.
6.		Menyatakan penyambung kehalaman lainnya.
7.		Menyatakan arah aliran pekerjaan (proses).
8.		Menyatakan input/output menggunakan disket.

Lanjutan Tabel 2.3 Simbol-Simbol *Flowchart*

No.	Simbol	Arti
9.		Menyatakan operasi yang dilakukan secara manual.
10.		Menyatakan percetakan (dokumen) pada kertas.
11.		Proses yang terdefinisi atau sub program.
12.		Menyatakan media penyimpanan drum magnetik.
13.		Persiapan yang digunakan untuk memberi nilai awal suatu besaran.
14.		Menyatakan input/output dari kartu plong.
15.		Multidocument (banyak dokumen).
16.		Delay (penundaan atau kelambatan).

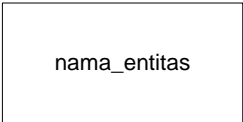
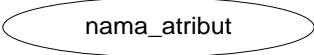

(Sumber: Indrajani, 2015:31-32)

2.2.4 Entity Relationship Diagram (ERD)

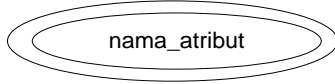
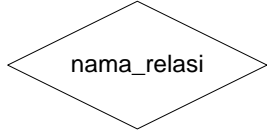
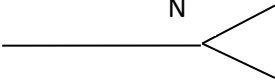
Rosa dan Shalahuddin (2016:50) menyatakan “*Entity Relational Diagram* (ERD) digunakan untuk pemodelan basis data relasional”. ERD memiliki beberapa aliran notasi seperti notasi *Chen* (dikembangkan oleh Peter Chen), *Barker* (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi *Crow’s Foot*, dan beberapa notasi lain.

Adapun simbol-simbol yang digunakan pada ERD dengan notasi *Chen* adalah sebagai berikut:

Tabel 2.4 Simbol-simbol *Entity Relational Diagram*

No	Simbol	Deskripsi
1	Entitas 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimoan datanya agar dapat diaks es oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
2	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3	Kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)

Lanjutan Tabel 2.3 Simbol-Simbol *Flowchart*

No	Simbol	Deskripsi
4	Multinilai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6	Asosiasi / Association 	Penghubung antar relasi dan entitas di mana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaianKemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B maka

(Sumber: Rosa dan Shalahuddin, 2016:50-51)

2.2.5 Pengertian Kamus Data

Rosa dan Shalahuddin (2016:73) menyatakan “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar penulisan).”

Indrajani (2015:30) menyatakan “Kamus data adalah katalog akta tentang dan kebutuhan informasi. Kamus data terdapat pada tahapan analisis dan perancangan.”

**Tabel 2.5** Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	Terdiri atas, mendefinisikan, diuraikan menjadi, artinya.
2	+	Dan
3	()	<i>Optional</i> (boleh ada atau boleh tidak)
4	{ }	Pengulangan
5	[]	Memilih salah satu dari sejumlah alternative, seleksi.
6	**	Komentar
7	@	Identifikasi atribut kunci
8	!	Pemisah sejumlah alternatif pilihan antar simbol []

(Sumber: Indrajani, 2015:31)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Dhanta dalam Supardi (2015:4) menyatakan “Aplikasi adalah software yang dibuat oleh suatu perusahaan *computer* untuk mengerjakan tugas-tugas tertentu seperti *Microsoft word*, *Microsoft excel*”.

Indrajani (2018:3) menyatakan “Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk penyelesaian tugas-tugas khusus dari pemakai komputer.”

Kesimpulan, Aplikasi merupakan suatu program komputer yang digunakan untuk membantu pengguna dalam membuat suatu tugas tertentu.

2.3.2 Pengertian Pengolahan Data

Reksoatmodjo (2018:8) menyatakan “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”

Hutahaean (2014:8) menyatakan “Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan.”



Kesimpulan, Pengolahan data adalah manipulasi data kebentuk yang lebih informatif berupa informasi yang dapat memiliki kegunaan.

2.3.3 Pengertian Surat Tugas

Rahaju (2019:36) menyatakan “Surat Tugas adalah naskah dinas dari atasan atau pejabat yang berwenang yang ditujukan kepada bawahan atau pegawai lainnya yang berisi penugasan untuk melaksanakan pekerjaan sesuai dengan tugas dan fungsi.”

Sovia (2017:68) menyatakan “Surat Tugas merupakan naskah dinas berisi mengenai kebijakan pokok atau kebijaksanaan pelaksanaan yang harus dipedomani dan dilaksanakan dalam penyelenggaraan tugas dan kegiatan setiap instansi pemerintahan yang berupa produk hukum yang bersifat pengaturan.”

Kesimpulannya, Surat Tugas adalah penjualan adalah aktivitas atau bisnis menjual produk atau jasa.

2.3.4 Pengertian Pegawai

Juli (2017:117) menyatakan “Pegawai adalah orang pribadi yang bekerja pada pemberi kerja, baik bekerja sebagai pegawai tetap/tenaga kerja lepas berdasarkan perjanjian atau kesepakatan kerja baik secara tertulis maupun tidak tertulis, untuk melaksanakan suatu pekerjaan dalam jabatan atau kegiatan tertentu dengan memperoleh imbalan yang dibayarkan berdasarkan periode tertentu, penyelesaian pekerjaan, atau ketentuan lain yang ditetapkan pemberi kerja, termasuk orang pribadi yang melakukan pekerjaan dalam jabatan negeri atau badan usaha milik negara atau badan usaha milik daerah.”

Kesimpulannya, Pegawai adalah seseorang yang melakukan penghidupannya dengan bekerja dalam kesatuan organisasi, baik kesatuan kerja pemerintah maupun kesatuan kerja swasta.

2.3.5 Pengertian Website

Iqbal (2014:4) menyatakan “*Website* merupakan halaman yang menampilkan informasi data teks, gambar, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis.”



Hikmah, dkk (2015:1) menyatakan “*Website* merupakan kumpulan *hyperlink* yang menuju alamat satu ke alamat lainnya dengan bahasa HTML.”

Dari beberapa definisi diatas penulis menyimpulkan bahwa, web adalah sekumpulan halaman yang berisikan informasi yang disediakan melalui koneksi internet.

2.3.6 Pengertian Aplikasi Pengolahan Data Surat Tugas Pegawai pada Kantor Wilayah Direktorat Jenderal Pajak Sumatera Selatan dan Kepulauan Bangka Belitung Berbasis Web

Aplikasi Pengolahan Data Surat Tugas Pegawai adalah aplikasi yang dibuat untuk membantu sub bagian umum dan kepegawaian dalam melakukan pengelolaan data surat tugas pegawai yang dimana pengelolaan tersebut terdiri dari data pegawai yang akan melaksanakan perjalanan dinas yang telah ditentukan di Kantor Wilayah Direktorat Jenderal Pajak Sumatera Selatan dan Kepulauan Bangka Belitung.

2.4 Teori Program

2.4.1 Pengertian Basis Data

Setiawan (2015:38) menyatakan “Basis data adalah kumpulan data yang terdiri dari atribut, entity dan relationship dari informasi sutau instansi atau perusahaan yang dirancang untuk memenuhi kebutuhan informasi dari sebuah organisasi.”

Indrajani (2015:30) menyatakan “Basis data adalah sebuah kumpulan data yang saling secara logis, dan merupakan sebuah penjelasandari data tersebut, yang didesain utnuk menemukan data yang dibutuhkan oleh sebuah organisasi.”

Kesimpulannya, Basis data adalah kumpulan data yang disimpan dikomputer secara sistematis untuk memperoleh informasi dari organisasi yang berelasi.

2.4.2 Pengertian XAMPP

Riyanto (2015:1)menyatakan “XAMPP merupakan paket *PHP* dan *MySQL* berbasis *open source* yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis *PHP*.”

Haqi dan Setiawan (2019:9) menyatakan “XAMPP adalah perangkat lunak



bebas (*free software*) yang mendukung banyak sistem operasi, merupakan komplikasi dari beberapa program.”

Kesimpulannya, *XAMPP* adalah suatu program *server* yang terdiri dari *Apache (server)*, *MySQL (database)*, dan *PHP* digunakan untuk mengakses fungsi yang ada dalam halaman website tersebut agar bisa diakses oleh *user*.

2.4.3 Pengertian *MySQL*

Winarno, dkk (2014:149) menyatakan “*MySQL* merupakan tipe data relasional yang artinya *MySQL*. Menyimpan datanya dalam bentuk tabel yang saling berhubungan.”

Enterprise (2014:3) menyatakan “*MySQL* adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak dipakai untuk berbagai kebutuhan.”

Kesimpulannya, *MySQL* adalah sebuah *software* pembuat *database* yang bersifat *open source*, sehingga siapa saja bisa menggunakannya dan tidak dicekal.

2.4.4 Pengertian *PHP*

Winarno, dkk (2014:49) menyatakan “*PHP (Hypertext Prerocessor)* adalah sebuah bahasa pemrograman web berbasis *server (server-side)* yang mampu parsing kode *PHP* dari kode *web* dengan ekstensi *php*.”

Setiawan (2015:33) menyatakan “*PHP* merupakan bahasa pemograman script yang paling banyak dipakai saat ini.”

Kesimpulannya, *PHP* adalah bahasa pemograman berbasis *server-side* yang bisa kita gunakan untuk membuat aplikasi *web* yang disisipkan pada *HTML*, yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi *dekstop*.



Gambar 2.1 Tampilan Logo *PHP*



2.4.4.1 Sintaks Program PHP

PHP adalah bahasa yang dirancang untuk mudah diletakkan didalam kode *HTML*. Banyak dijumpai kode *PHP* yang menyatu dengan kode *HTML*, Kode *PHP* diawali dengan *tag* `<?php` dan diakhiri dengan *tag* `?>`. Apabila kita melakukan konfigurasi terhadap *file php* ini untuk mengizinkan penggunaan *tag* pendek (*short tag*) dengan mengubah nilai *short_open_tag* menjadi *On*, maka *tag* tersebut dapat diganti dengan `<?` dan `?>`. Berikut ini contoh kode *PHP* yang sangat sederhana.

```
<?php
    echo "Hello Word!";
?>
```

Perintah *echo* didalam *PHP* berguna Untuk mencetak nilai, baik teks maupun numerik, ke layar *web browser*. Selain *echo*, kita juga dapat menggunakan perintah *print*, sehingga kode diatas dapt juga ditulis sebagai berikut.

```
<?php
    Print "Hello Word!";
?>
```

Perintah *echo* dan *print* juga memiliki bentuk penulisan yang lain, seperti layaknya pemanggilan fungsi. Berikut ini bentuk lain tersebut.

```
<?php
    echo ("Hello Word!"); // mencetak teks "Hello Word!"
    echo (1);             // mencetak angka 1
    print ("Hello Word!"); // mencetak teks "Hello Word!"
    print (100);          // mencetak angka 100
?>
```

Setiap perintah atau statemen di dalam kode *PHP* harus diakhiri dengan tanda titik koma atau *semicolon*(:).



2.4.5 Pengertian *JavaScript*

Hidayatullah dan Khairul (2017:395) menyatakan “*JavaScript* ialah suatu bahasa *scripting* yang digunakan sebagai fungsionalitas dalam membuat suatu *web*.”

Setiawan (2015:22) menyatakan “*JavaScript* adalah salah satu platform yang menggunakan teknik *client side* yang cukup dikenal dan mampu berjalan di banyak platform sistem operasi, termasuk sistem operasi *mobile*.”

Kesimpulannya, *JavaScript* adalah bahasa pemrograman yang bisa disisipkan ke *HTML* seperti halnya *PHP* akan tetapi *javascript* berjalan di sisi *Client*.

2.4.5.1 Dasar *JavaScript*

Cara menggunakan *JavaScript* adalah dengan dimasukkan di antara kode *HTML* menggunakan tag `<script>` dan `</script>`. *JavaScript* bisa diletakkan di tag `<body>` ataupun tag `<head>` dari kode *HTML*. Untuk memasukkan *javascript* anda harus menggunakan tag `<script>`, tag `<script>` dan `</script>` menentukan dimana *javascript* harus dimulai dan diakhiri.

Baris diantara tag `<script>` dan `</script>` ini berisi data *JavaScript* contohnya seperti berikut:

```
<script>  
    Alert (“kode javascript pertama”);  
</script>
```