



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Abdul Kadir (2017:2), “Komputer adalah peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia.”

Menurut Hartono (2013:27), “Komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang untuk secara otomatis melakukan serangkaian urutan penghitungan (*arithmetic*) atau proses-proses yang diurutkan secara logis”.

Paramytha (2016:4), “Komputer berasal dari bahasa latin yaitu *Computer* yang berarti menghitung (*to compute* atau *to reckon*). Kata komputer itu sendiri pada awalnya dipergunakan untuk menggambarkan pekerjaan orang yang melakukan perhitungan aritmatika dengan atau tanpa alat bantu.”

Dari pendapat diatas dapat penulis simpulkan bahwa komputer adalah sebuah mesin yang dapat membantu mempermudah pekerjaan manusia untuk melakukan serangkaian urutan perhitungan atau proses-proses yang diurutkan secara logis.

2.1.2 Pengertian Internet (*Interconnected Network*)

Menurut Sujatmiko (2012:138), “Jaringan global yang menghubungkan berjuta-juta komputer diseluruh dunia melalui jalur telepon kabel maupun satelit”.

Menurut Setiawan dan Irma (2008:13), “Internet merupakan media yang dapat digunakan untuk mencari beragam informasi, mulai dari periklanan, berita, hiburan, bisnis, pemerintahan, militer, kemajuan teknologi, olah raga sampai dengan pendistribusian *software* (*downloads*)”.

Dari pendapat diatas dapat penulis simpulkan bahwa internet adalah jaringan global yang menghubungkan berjuta-juta komputer diseluruh dunia untuk mencari informasi yang diperlukan oleh penggunanya.



2.1.3 Pengertian Perangkat Lunak (*Software*)

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

Menurut Kadir (2017:2), “Perangkat Lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai. Sistem operasi seperti *Windows*, *Mac OS*, dan *Linux*, dan aplikasi seperti *Microsoft Word* dan *Microsoft Excel* adalah contoh perangkat lunak.”

Menurut Sujatmiko (2012:5), “Perangkat lunak (*software*) kumpulan-kumpulan beberapa perintah yang di eksekusi oleh mesin komputer dalam menjalankan pekerjaannya”.

Dari pendapat diatas dapat penulis simpulkan bahwa perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak dalam menjalankan pekerjaannya

2.1.4 Pengertian Perangkat Keras (*Hardware*)

Menurut Sujatmiko (2012:123), “Perangkat keras (*hardware*) perangkat komputer yang dapat dilihat (berwujud nyata) yang mendukung proses berjalannya atau beroperasinya komputer”.

Menurut Jogiyanto (2009:91), “Perangkat keras (*hardware*) sebagai sub-sistem dari komputer juga mempunyai komponen, yaitu komponen alat masukan (*input device*), komponen alat pemroses (*processing device*), komponen alat keluaran (*output device*) dan komponen alat simpanan luar (*storage*)”.

Dari pendapat diatas dapat penulis simpulkan bahwa perangkat keras (*hardware*) merupakan perangkat komputer yang berwujud nyata dan mempunyai komponen berupa alat masukan, alat pemroses, alat keluaran dan alat simpanan untuk membantu pekerjaan pemakainya.



2.1.5 Pengertian Website

Menurut Abdulloh (2018:1), “*Website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.”

Menurut Sidik (2017:1), “Situs web (*Web site*) awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hyperlink, yang memudahkan surfer (sebutan bagi pemakai komputer yang melakukan penelusuran informasi di Internet) untuk mendapatkan informasi, dengan cukup mengklik suatu link berupa teks atau gambar, maka Informasi dari teks atau gambar akan ditampilkan secara lebih rinci (*detail*).”

Dari pendapat diatas dapat penulis simpulkan bahwa *website* merupakan kumpulan halaman yang berisi informasi digital berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya dengan cara cukup mengklik suatu link, maka informasi tersebut akan ditampilkan secara lebih rinci.

2.1.6 Pengertian Basis Data

Menurut Fathansyah (2018:2), “Basis data adalah Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.”

Menurut Sukanto dan Shalahuddin (2018:43), “Basis data adalah sistem komputerisasi yang tujuan utamanya adalah memelihara data yang sudah ada yang diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.”

Dari pendapat diatas dapat penulis simpulkan bahwa basis data adalah himpunan kelompok data (arsip) yang saling berhubungan yang diolah dan membuat informasi tersedia saat dibutuhkan.

2.1.7 Metode Pengembangan Sistem

Sukanto dan Shalahudin (2018:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga



disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup terurut mulai dari analisis, desain, pengkodean, pengujian dan pemeliharaan.

a. Analisis Kebutuhan Perangkat Lunak

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

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean.

c. Pembuatan Kode Barang

Pada tahap pengkodean, 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. Dalam penelitian ini pengujian sistem akan menggunakan pengujian Black-Box. Pengujian Black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian metode ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program.

e. Pendukung (*support*) dan pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah mengirimkan ke *user*. Perubahan bisa terjadi adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.



2.2 Teori Judul

2.2.1 Pengertian Aplikasi

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

Menurut Budiharto (2013:5), “Aplikasi (*Applications*) merupakan program yang dapat berjalan di komputer tersendiri (*stand alone computer*), dari mulai program yang simple sampai dengan program besar dan rumit”.

Dari pendapat di atas dapat penulis simpulkan bahwa aplikasi adalah program yang dapat berjalan tersendiri dan juga menentukan aktivitas pemrosesan informasi untuk penyelesaian tugas-tugas khusus dari penggunanya.

2.2.2 Pengertian *Maintenance*

Menurut Ngadiyono (2010:1), “Pemeliharaan (*maintenance*) juga bermakna melakukan tindakan rutin guna menjaga perangkat (dikenal sebagai pemeliharaan terjadwal) atau mencegah timbulnya gangguan (pemeliharaan pencegahan)”.

2.2.3 Pengertian *Report*

Menurut Kamus Besar Bahasa Indonesia (2017:497), “Segala sesuatu yang dilaporkan”. Sedangkan menurut Wahana dan Riswaya (2014:31), “Laporan adalah bentuk penyajian fakta tentang suatu keadaan atau suatu kegiatan, pada dasarnya fakta yang disajikan itu berkenaan dengan tanggung jawab yang ditugaskan kepada si pelapor”. Dari pendapat di atas dapat penulis simpulkan bahwa laporan adalah bentuk penyajian fakta tentang suatu keadaan atau suatu kegiatan yang dilaporkan.

2.2.4 Pengertian Aplikasi *Maintenance Report* Teknisi Unit Kerja ICT pada PT Semen Baturaja (Persero) Tbk

Aplikasi *Maintenance Report* Teknisi Unit Kerja ICT pada PT Semen Baturaja (Persero) Tbk adalah suatu aplikasi yang menyediakan layanan pelaporan



keluhan kerusakan perangkat dan jaringan oleh unit kerja yang ada di PT Semen Baturaja kepada unit kerja ICT untuk dilakukan perbaikan terhadap perangkat dan jaringan tersebut.

2.3 Teori Khusus

2.3.1 Pengertian Kamus Data

Menurut Sukamto & Shalahuddin (2018:73), “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 cara penulisan)”.

Tabel 2.1 Simbol-simbol pada Kamus Data

No	Simbol	Keterangan
1.	=	disusun atau terdiri dari
2.	+	Dan
3.	[]	baik...atau...
4.	{n}	n kali/ bernilai banyak
5.	()	data opsional
6.	*...*	batas komentar

Sumber: Sukamto dan Shalahuddin (2018:74)

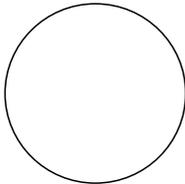
2.3.2 Pengertian Data Flow Diagram (DFD)

Sukamto dan Shalahuddin (2018:71) menjelaskan, “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 (*output*).

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



Tabel 2.2 Simbol-Simbol dalam *Data Flow Diagram (DFD)*

No	Notasi	Keterangan
1.		<p>Proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
2.		<p>File basisdata atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman 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>)</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>
3.		<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakain/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan .</p> <p>Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>



Lanjutan Tabel 2.2 Simbol-Symbol dalam *Data Flow Diagram (DFD)*

No	Notasi	Keterangan
4.		<p>Aliran data: merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses masukan (<i>input</i>) atau keluaran (<i>output</i>)</p> <p>Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”</p>

Sumber: Sukamto dan Shalahuddin (2018:71)

Sukamto dan Shalahuddin (2018:72-73) menjelaskan, Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

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

DFD Level 0 (nol) menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 (nol) 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 (nol) yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-breakdown menjadi DFD Level 2. (dua) 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(satu) yang di-breakdown.

4. Membuat DFD Level 3 dan seterusnya



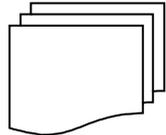
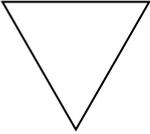
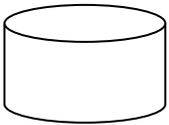
DFD Level 3,4,5 dan seterusnya merupakan breakdown dari modul pada DFD di atasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.3.3 Pengertian *Blockchart*

Menurut Kristanto (2018:75), “Block Chart berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”.

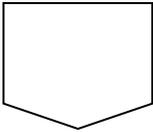
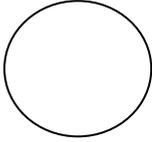
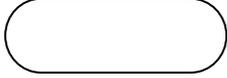
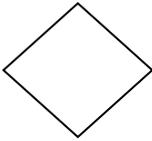
Adapun simbol-simbol yang sering digunakan dalam *BlockChart* (sig! dapat) dilihat pada tabel berikut ini:

Tabel 2.3 Simbol-simbol pada *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 (data storage)



Lanjutan Tabel 2.3 Simbol-simbol pada *BlockChart*

No.	Simbol	Keterangan
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
8.		Terminasi yang mewakili simbol tertentu untu digunakan pada aliran lain pada halaman yang lain
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>)
12.		Layar peraga (monitor)
13.		Pemasukan data secara manual

Sumber: Kristanto (2018:75-77)



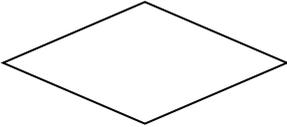
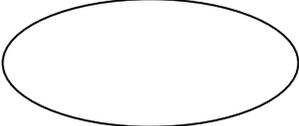
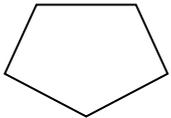
2.3.4 Pengertian *Flowchart*

Menurut Indrajani (2015:36), “Merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program”.

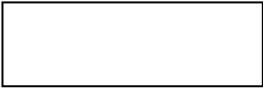
Menurut Santoso dan Nurmalina (2017:86), *Flowchart* adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek.

Dari pendapat di atas dapat penulis simpulkan bahwa *Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan suatu program untuk menyelesaikan suatu masalah.

Tabel 2.4 Simbol-simbol pada *Flowchart*

No.	Simbol	Fungsi
1.		Permulaan sub program.
2.		Perbandingan, pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.
3.		Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman.
4.		Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda.

Lanjutan Tabel 2.4 Simbol-simbol pada *Flowchart*

No.	Simbol	Fungsi
5.		Permulaan/akhir Program.
6.		Arah aliran program.
7.		Proses inisialisasi/pemberian harga awal
8.		Proses penghitung / proses pengolahan data
9.		Proses <i>input</i> / <i>output</i> data

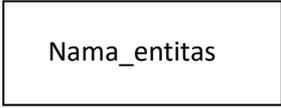
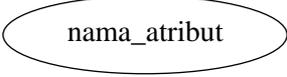
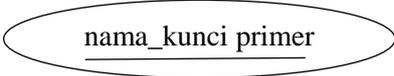
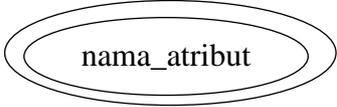
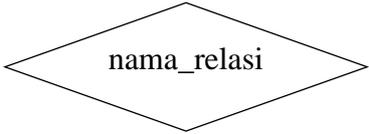
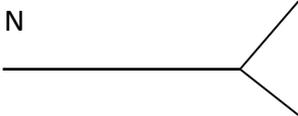
Sumber: Santoso dan Nurmalina (2017:86-87)

2.3.5 Pengertian *Entity Relationship Diagram* (ERD)

Sukamto dan Shalahuddin (2018:50) menjelaskan, “*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 2.5 Simbol-simbol pada *Entity Relationship Diagram* (ERD)

No.	Simbol	Deskripsi
1.	<p>Entitas / <i>Entity</i></p> 	Entitas merupakan data inti yang akan disimpan; bakal
2.	<p>Atribut</p> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.		<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)
4.	<p>Atribut multivalui / <i>multivalued</i></p> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5.	<p>Relasi</p> 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6	<p>Asosiasi / <i>association</i></p> 	<p>Penghubung antar relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian</p> <p>Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B maka</p>

Sumber: Sukamto dan Shalahuddin (2018:50-51)



2.4 Teori Program

2.4.1 Pengertian XAMPP

Aryanto (2016:4), “*Xampp* merupakan sebuah aplikasi perangkat lunak pemrograman dan *database* yang di dalamnya terdapat berbagai macam aplikasi pemrograman seperti: *Apache*, *HTTP*, *MySQL*, *database*, bahasa pemrograman *PHP* dan *Perl*.”

Madcoms (2014:186), “*Xampp* adalah sebuah paket kumpulan software yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *Perl*, *Filezilla* dan lain-lain.”

Dari pendapat di atas dapat penulis simpulkan bahwa *XAMPP* adalah sebuah aplikasi perangkat lunak pemrograman dan *database* yang di dalamnya terdapat berbagai macam aplikasi pemrograman yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *Perl*, *Filezilla* dan lain-lain.

2.4.2 Pengertian MySQL

Sukamto dan Shalahuddin (2018:46), “*SQL (Structured Query Language)* adalah bahasa yang digunakan untuk mengelola data pada RDBMS. *SQL* awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”

Raharjo (2015:16), “*MySQL* adalah *software* RDMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user (multi-user)*, dan dapat melakukan suatu proses secara sinkron atau bebarengan (*multi-threaded*).”

Dari pendapat di atas dapat penulis simpulkan bahwa *MySQL* adalah bahasa yang digunakan untuk mengelola data *SQL* pada (*database management system*) DBMS yang dapat diakses oleh banyak *user* dan dapat melakukan proses secara sinkron.

2.4.3 Pengertian PHP

Murya (2014:65), “*PHP Hypertext Preprocessor* atau sering disebut *PHP* merupakan bahasa pemrograman berbasis *server-side* yang dapat melakukan parsing *script php* menjadi *script web* sehingga dari sisi client menghasilkan suatu tampilan yang menarik.”



Madcoms (2016:2), “PHP (*Hypertext preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML.”

Dari pendapat di atas dapat penulis simpulkan bahwa PHP adalah bahasa pemrograman berbasis *server-side* yang dapat melakukan parsing *script* php menjadi *script* web dan dapat disisipkan pada HTML sehingga menghasilkan tampilan yang menarik.

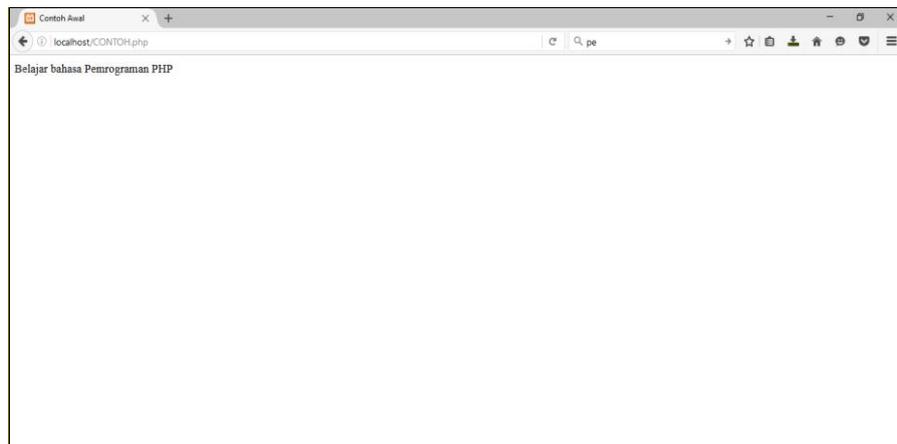


Gambar 2.1. Tampilan Logo PHP

2.4.3.1 Sintaks Dasar PHP

Kode (Script) PHP yang sering disebut dengan istilah *embedded script* yaitu *script* PHP yang disisipkan di antara *script* HTML. Jadi dapat dikatakan *script* PHP hanya ditulis atau disisipkan ketika dibutuhkan saja, seperti menampilkan data dari database meng-upload file, delete data, edit data dan lain sebagainya. Contoh script:

```
<HTML>
  <HEAD>
    <TITLE>Contoh Awal</TITLE>
  </HEAD>
<BODY>
  <?php
    echo “Belajar bahasa Pemrograman PHP”;
  ?>
</BODY>
</HTML>
```



Gambar 2.2. Contoh Script PHP

2.4.3.2 Tipe Data PHP

Tipe data merupakan jenis dari suatu data yang akan diproses oleh bahasa pemrograman. Yosef Murya (2014:26), menjelaskan beberapa tipe data dalam PHP, sebagai berikut:

1. **Integer** merupakan tipe data yang berguna untuk menyimpan bilangan bulat. Range bilangan integer adalah antara -2.147.483.647 sampai dengan 2.147.483.647
2. **Double Floating** adalah tipe data yang berguna untuk menyimpan bilangan desimal. Range bilangan floating point antara 1e308 sampai dengan 1e308.
3. **Boolean** adalah tipe data yang paling sederhana, hanya berupa **TRUE** dan **FALSE**.
4. **String** adalah tipe data yang terdiri dari kata, bias berupa kata tunggal maupun kalimat. Penulisan string harus diapit dengan tanda petik, baik berupa petik tunggal ('...') maupun petik ganda ("...").
5. **Objek** adalah tipe data dibuat dengan tujuan agar para programmer terbiasa dengan OOP. Tipe data ini bias berupa bilangan.
6. **Array** merupakan **Tipe Compound Primitif**, terdapat pada bahasa pemrograman lain.
7. **Null** adalah tipe data yang tidak memuat apapun. Setiap variable yang diset menjadi tipe data Null, ini akan menjadikan variabel tersebut kosong.
8. **Resources** tipe data spesial yang satu ini dikhususkan untuk menyimpan *resources*, sumber atau alamat.



2.4.4 Pengertian JavaScript

Abdulloh (2018:193), “*Javascript* merupakan Bahasa pemrograman web yang pemrosesannya dilakukan di sisi *client*. Karena berjalan di sisi *client*. *Javascript* dapat dijalankan hanya dengan menggunakan browser.”

Anggie Bratadinata (2013:5), “*JavaScript* adalah bahasa pemrograman yang dibuat mengikuti spesifikasi standar yang disebut *ECMAScript* dan saat ini versi termmodern dari *ECMAScript* adalah versi 5.”

Dari pendapat di atas dapat penulis simpulkan bahwa *javascript* adalah Bahasa pemrograman yang pemrosesannya dilakukan di sisi *client* yang dibuat mengikuti spesifikasi standar yang disebut *ECMAScript* dan saat ini versi termmodern nya adalah versi 5.

2.4.4.1 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 wanda 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 :

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <script>
      var a = “ Selamat Datang! ” ;
      function selamat () {
        var b = “Selamat Pagi!” ;
        document.write (a);
      }
      selamat ();
```



```
document.write (b) ;  
</script>  
</body>  
</html>
```