



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Aplikasi

Sujatmiko (2012:23) menyatakan “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas–tugas tertentu, misalnya MS-Word, MS-Excel.”

Asropudin (2013:7) menyatakan, “Aplikasi adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Word dan Microsoft Excel*.”

Puspitosari (2013:11) menyatakan, “Program aplikasi atau biasa disebut dengan aplikasi, merupakan program yang berjalan ada sistem operasi dan dibuat untuk membantu user mengerjakan sesuatu untuk meningkatkan produktivitasnya.”

##### 2.1.2 Pengertian Penghitungan

Menurut Kamus Besar Bahasa Indonesia “Penghitungan adalah proses, cara, perbuatan menghitung.” <https://kbbi.web.id/hitung>

Penghitungan adalah kegiatan melakukan pencatatan, perincian, dan keterangan mengenai keluar masuknya uang, dalam hal ini bisa pendapatan, potongan, dan tunjangan karyawan. Tujuan penghitungan ini untuk mendapatkan total gaji yang diperoleh karyawan setiap bulannya, selain itu untuk membuat slip gaji karyawan dan laporan gaji kepada pimpinan.

##### 2.1.3 Pengertian Gaji

Hanggraeni (2012: 213), “Upah (*wage*) atau gaji (*salary*) merupakan imbal balik yang diterima oleh individu sebagai balas jasa atas kontribusinya terhadap organisasi.”

Mulyadi (2013: 373), “Gaji merupakan pembayaran atas penyerahan jasa yang dilakukan oleh karyawan yang mempunyai jenjang jabatan manajer,



umumnya gaji dibayarkan secara tetap perbulan. Sedangkan upah adalah pembayaran atas penyerahan jasa yang dilakukan oleh karyawan pelaksana (buruh), upah dibayarkan berdasarkan hari kerja, jam kerja atau jumlah satuan produk yang dihasilkan oleh karyawan.”

Jadi, dapat disimpulkan bahwa gaji adalah imbalan atau timbal balik yang diterima oleh karyawan atas kontribusinya atau jasa yang telah diberikan kepada perusahaan tempatnya bekerja.

#### **2.1.4 Pengertian Insentif**

Hanggraeni (2012:153), “Secara sederhana insentif bisa diartikan sebagai salah satu bentuk kompensasi perusahaan kepada pekerjanya didasarkan pada kinerja yang ditunjukkannya.”

Hasibuan (2013:118) mengemukakan “Insentif adalah tambahan balas jasa yang diberikan kepada karyawan tertentu yang prestasinya di atas prestasi standar. Insentif ini merupakan alat yang dipergunakan pendukung prinsip adil dalam pemberian kompensasi.”

Dari dua pengertian di atas dapat disimpulkan bahwa insentif adalah kompensasi atau balas jasa dari perusahaan untuk karyawannya yang didasarkan pada prestasinya dalam kinerja bekerjanya yang di atas standar.

#### **2.1.5 Pengertian Potongan**

Potongan adalah pengurangan gaji karyawan oleh perusahaan dikarenakan hal-hal tertentu, seperti potongan dikarenakan adanya selisih antara data barang yang dijual dan data barang yang ada di gudang serta jika adanya kelebihan barang dalam proses penjualan. Berikut ini beberapa jenis potongan yang dilakukan perusahaan, antara lain:

##### **a. Potongan Iuran Kesehatan**

Potongan iuran kesehatan adalah potongan yang dikenakan oleh pemberi kerja terhadap penghasilan yang diterima oleh pekerja untuk membayar pihak asuransi kesehatan. Iuran kesehatan biasanya dilakukan karena perusahaan mengikuti asuransi kesehatan atau keselamatan kerja para pegawai atau pekerja.



Sehingga tanggungan kesehatan dan keselamatan para pekerja dibebankan pada perusahaan tersebut.

b. Potongan Iuran Tunjangan Hari Tua

Potongan iuran kesehatan adalah potongan yang dikenakan oleh pemberi kerja terhadap penghasilan yang diterima oleh pegawai atau pekerja untuk membayar pihak perusahaan yang menanggung dana pekerja seperti PT Jamsostek.

### **2.1.6 Pengertian Karyawan**

Rachman (2016:4) menyatakan bahwa, “Karyawan adalah kekayaan utama suatu perusahaan, karena tanpa keikutsertaan mereka, aktivitas perusahaan tidak akan terjadi. Karyawan berperan aktif dalam menetapkan rencana, system, proses, dan tujuan yang ingin dicapai.”

Menurut Kamus Besar Bahasa Indonesia, “Karyawan adalah orang yang bekerja pada suatu lembaga (kantor, perusahaan, dan sebagainya) dengan mendapatkan gaji (upah).” <https://kbbi.web.id/karyawan>

Jadi, karyawan adalah sumber daya manusia yang bekerja untuk mencapai tujuan suatu perusahaan dan mendapatkan gaji sebagai imbalannya.”

### **2.1.7 Pengertian CV**

Silondae dan Ilyas (2011:38), menyatakan “Persekutuan komanditer (CV) adalah persekutuan firma yang mempunyai satu atau beberapa orang sekutu komanditer.”

Nayla (2014:39), “Persekutuan komanditer atau biasa disebut CV adalah badan usaha yang didirikan dan dimiliki oleh dua orang atau lebih dengan tingkatan keterlibatan yang berbeda di antara para pemiliknya.”

Ramansyah (2014:39), “Persekutuan komanditer atau biasa disebut CV adalah badan usaha yang didirikan oleh dua orang atau lebih dengan tingkat keterlibatan yang berbeda di antara para pemiliknya dan dijalankan oleh pemilik aktif.”



### **2.1.8 Pengertian Aplikasi Penghitungan Gaji, Insentif, dan Potongan Karyawan pada CV Sinar Mas Sejahtera**

Aplikasi Penghitungan Gaji, Insentif, dan Potongan Karyawan pada CV Sinae Mas Sejahtera adalah sebuah aplikasi yang dibangun untuk menghitung total gaji karyawan setiap bulannya. Total gaji ini didapatkan dari gaji pokok, insentif, dan potongan-potongan. Selain itu, aplikasi ini juga ditujukan untuk membuat slip gaji karyawan dan laporan kepada pimpinan.

## **2.2 Teori Khusus**

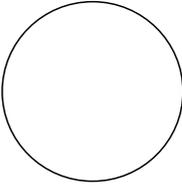
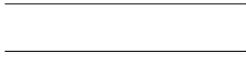
### **2.2.1 Pengertian DFD (*Data Flow Diagram*)**

S. Rosa A dan Shalahuddin (2016:70) menjelaskan, “*Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Aliran Data (DAD) 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*

Notasi	Keterangan
	<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>
	<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</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>

Lanjutan **Tabel 2.1** Simbol-simbol *Data Flow Diagram*

Notasi	Keterangan
	<p>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</p> <p>Catatan :</p> <p>Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>
	<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:</p> <p>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 : S. Rosa A dan Shalahuddin, 2016:71-72)



S. Rosa A 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 2. Modul mana saja yang harus di-breakdown 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.

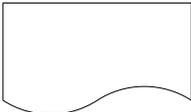
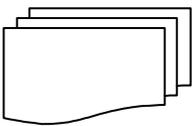
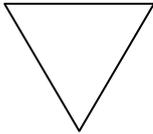
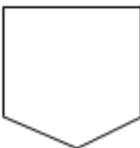
### 2.2.2 Pengertian *Block Chart*

Kristanto (2008:75) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”. Pembuatan blockhart harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

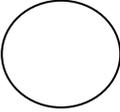
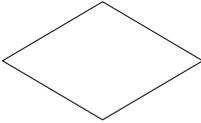


Adapun simbol-simbol yang sering digunakan dalam block chart dapat dilihat pada tabel berikut ini:

**Tabel 2.2** Simbol-simbol *Block chart*

Simbol	Arti
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/berkas atau cetakan.
	Multi Dokumen
	Proses Manual
	Proses yang dilakukan oleh komputer
	Menandakan dokumen yang diarsipkan (arsip manual)
	Data penyimpanan ( <i>data storage</i> )
	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain

Lanjutan **Tabel 2.2** Simbol-simbol *Block chart*

Simbol	Arti
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran
	Pengambilan keputusan ( <i>decision</i> )
	Layar peraga ( <i>monitor</i> ).
	Pemasukkan data secara manual.

(Sumber : Kristanto, 2008:75-77)

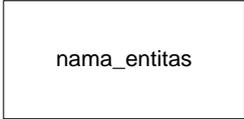
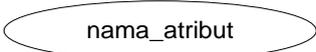
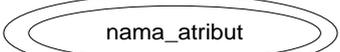
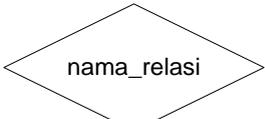
### 2.2.3 Pengertian ERD (*Entity Relational Diagram*)

S. Rosa A dan Shalahuddin (2016:50) 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.

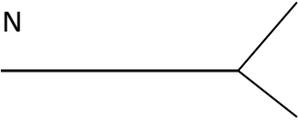


Adapun simbol-simbol yang digunakan pada ERD dengan notasi Chen adalah sebagai berikut:

**Tabel 2.3** Simbol-simbol *Entity Relational Diagram*

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimoan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Atribut 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)
Atribut multivalai / <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja

Lanjutan Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

Simbol	Deskripsi
<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: S. Rosa A dan Shalahuddin , 2016:50-51)

#### 2.2.4 Pengertian *Flowchart*

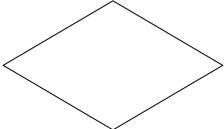
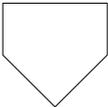
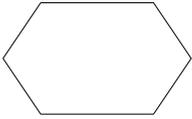
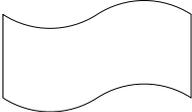
Lamhot (2015:14) menjelaskan, “*Flowchart* adalah langkah-langkah menyelesaikan masalah yang dituliskan dalam simbol-simbol tertentu.”

Menurut Lamhot (2015:14-16), Gambaran simbol-simbol *Flowchart* sebagai berikut:

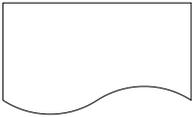
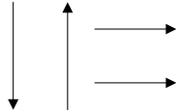
Tabel 2.4 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program.
2		<i>Input/output</i>	Menyatakan proses input atau output tanpa tergantung jenis peralatannya.

Lanjutan **Tabel 2.4** Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya atau tidak.
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
7		<i>Predefined process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
8		<i>Fanned Card</i>	Menyatakan input berasal dari kartu atau output data ke kartu.
9		<i>Punch Tape</i>	

Lanjutan **Tabel 2.4** Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
10		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui Printer)
11		<i>Flow</i>	Menyatakan jalannya arus suatu proses

(Sumber: Lamhot, 2015:14-15)

### 2.2.5 Pengertian Kamus Data

S. Rosa A dan Shalahuddin (2016:73) menjelaskan, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (ouput) dapat dipahami secara umum (memiliki standar cara penulisan).”

Kamus data memiliki beberapa simbol sebagai berikut :

**Tabel 2.5** Simbol-simbol Kamus Data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau...
{ }n	N kali diulang/bernilai banyak
()	Data opsional
*...*	Batas komentar

(Sumber: S. Rosa A dan Shalahuddin, 2016:74)



### 2.2.6 Metode Pengembangan Sistem

S. Rossa A dan Shalahuddin (2013:28) 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 perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis kebutuhan perangkat lunak

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pembuatan kode program

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika 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 (*maintenance*)

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses



pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

### **2.2.7 Black-Box Testing (Pengujian Kotak Hitam)**

S. Rosa A dan Shalahuddin (2016:275), *black-box testing*, yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

## **2.3 Teori Program**

### **2.3.1 Pengertian HTML**

Raharjo (2016:449), “HTML merupakan file teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke user melalui suatu aplikasi web browser.”

Asropudin (2013:44), “*Hyper Text Markup Language*, adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*.”

Setiawan (2015:33), “Hypertext Markup Language (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser internet.”

### **2.3.2 Pengertian Basis Data (Database)**

Raharjo (2016:521), “ Secara umum, database dapat didefinisikan sebagai kumpulan tabel (walaupun lebih tepat dikatakan kumpulan objek karena yang terkandung di dalam database sebenarnya bukan hanya tabel, melainkan indeks, *view*, *constraint*, *trigger*, dan sebagainya.”

Hidayat (2011:13), “Database adalah aplikasi yang berorientasi pada pengolahan data.”

Setiawan (2015:25), “Database (basis data) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa



menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.”

### 2.3.3 Pengertian MySQL

Faizal dan Irnawati (2015:4), “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 (enam) juta instalasi di seluruh dunia.

Sujatmiko (2012:76), “Basis data adalah representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.”

Sebagai sebuah program penghasil database, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain(*interface*). MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open source* seperti PHP maupun yang tidak, yang ada pada platform windows seperti Visual Basic, Delphi, dan lainnya.

### 2.3.4 Pengertian PHP (*Hypertext Preprocessor*)

Raharjo (2016:38) menyatakan, “PHP adalah salah satu bahasa pemrograman skrip yang dirancang untuk membangun aplikasi web.”

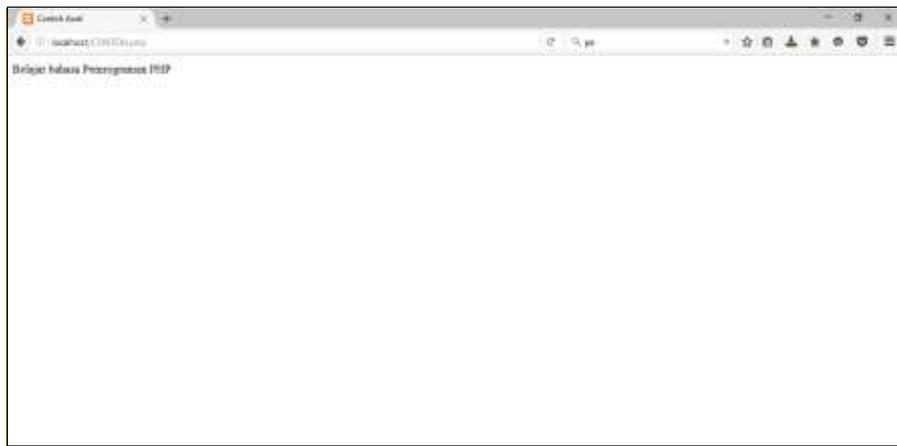
Sujatmiko (2012: 213), “PHP adalah bahasa pemrograman yang bisa kita gunakan untuk membuat aplikasi web.”

Wardana (2016:4), “PHP adalah bahasa scripting yang menyatu dengan HTML (kode dasar *website*) dan dijalankan pada *server side*.”

### 2.3.5 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 2.1** Tampilan di Browser

### 2.3.6 Pengertian Sublime Text

Supono dan Virdiandry (2016:14), “Sublime Text merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi. Sublime Text mempunyai fitur *plugin* tambahan yang memudahkan *prigrammer*. Selain itu, Sublime Text juga memiliki desain yang simple dan keren menjadikan Sublime Text berkesan elegan untuk sebuah *syntax editor*.”