



## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Teori Umum

##### 2.1.1. Pengertian Komputer

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

Asropudin (2013:23), “Komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer.”

Kesimpulannya, komputer adalah alat elektronik yang beroperasi di bawah kendali instruksi untuk pengolahan informasi sesuai prosedur yang telah dirumuskan.

##### 2.1.2. Pengertian Perangkat Lunak

Sukamto et.al, (2014:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Sujatmiko (2012:256), “Perangkat lunak adalah kumpulan beberapa perintah yang dieksekusi oleh mesin komputer dalam menjalankan pekerjaannya.”

Kesimpulannya, perangkat lunak adalah sebuah program komputer yang dibuat sedemikian rupa atau tersistem untuk melakukan tugas tertentu.

##### 2.1.3. Pengertian Aplikasi Web

Sujatmiko (2012:23), “Aplikasi web adalah suatu aplikasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti internet atau intranet.”

Kadir (2009:2), “Aplikasi web adalah jenis aplikasi yang diakses melalui *browser*, biasanya Internet Explorer dan Mozilla Firefox.”



Kesimpulannya, Aplikasi Web adalah sekumpulan halaman web yang mampu berinteraksi dengan pengunjung, dengan sesama halaman web dan dengan berbagai sumber data yang ada di server web.

#### **2.1.4. Pengertian Teknologi**

Kristanto (2008:14), “Teknologi merupakan bagian yang berfungsi untuk memasukkan input, mengolah input dan menghasilkan keluaran.”

Kamus Besar Bahasa Indonesia (2008:1422), menyatakan teknologi adalah metode ilmiah untuk mencapai tujuan praktis; ilmu pengetahuan terapan.

Kesimpulannya, teknologi adalah suatu bagian yang berfungsi untuk memasukkan input, mengolah input dan menghasilkan keluaran dengan metode-metode tertentu dalam menunjang perkembangan pada suatu daerah.

#### **2.1.5. Pengertian Data**

Kristanto (2008:7), “Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.”

Indrajani (2015:69), “Data merupakan fakta mentah tentang orang, tempat, kejadian, dan apapun yang penting bagi perusahaan, dimana data itu sendiri tidak memiliki arti.”

Kesimpulannya, data adalah suatu objek berupa angka-angka atau grafik yang mengandung arti yang dapat diolah menjadi sesuatu yang berguna.

#### **2.1.6. Pengertian Informasi**

Kristanto (2008:7), “Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima.”

Jogiyanto (2009:36), “Informasi (information) adalah data yang diolah menjadi bentuk yang berguna bagi para pemakainya.”

Kesimpulannya, informasi adalah suatu data yang bersifat penting dalam hal penggunaannya.



## 2.2. Teori Khusus

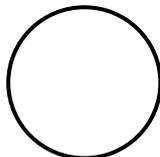
### 2.2.1. Pengertian DFD (*Data Flow Diagram*)

Indrajani (2015:27) “*Data Flow Diagram* adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut.”

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

Sukamto et.al, (2014:71), menjelaskan notasi pada DFD adalah sebagai berikut :

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

No.	Simbol	Keterangan
1		<p>Proses atau fungsi atau prosedur; pada permodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka permodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.</p> <p>catatan: nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>

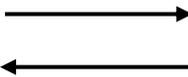


**Lanjutan Tabel 2.1** Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
2		<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada permodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka permodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus 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: nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>

*Sumber: Sukanto et.al (2014:71)*

Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
3		Entitas luar ( <i>externalentity</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> ) berupa kata benda.
4		Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan ( <i>input</i> ) atau keluaran ( <i>output</i> ).  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”.

*Sumber: Sukanto et.al (2014:71)*

Kristanto (2008:68) menjelaskan ada beberapa petunjuk yang dapat dipakai dalam proses pembuatan simbol DFD. Petunjuk-petunjuk itu adalah sebagai berikut:

1. Penamaan yang jelas
  - a. Sebaiknya menggunakan nama yang mengacu pada fungsi, yaitu gabungan antara kata kerja yang spesifik.
  - b. Jangan menggunakan nama terlalu umum.
  - c. Gunakan nama yang mudah dimengerti oleh pemakai.
2. Memberi nomor pada proses
  - a. Nomor yang diberikan pada proses tidak harus menjadi nomor urut.



- b. Penomoran dimaksudkan sebagai identifikasi proses dan memudahkan penurunan (level yang lebih rendah) ke proses berikutnya.
3. Penggambaran kembali
  - a. Ukuran dan bentuk lingkaran tetap sama.
  - b. Panah yang melengkung dan lurus tidak menjadi masalah.
4. Hindarilah proses yang mempunyai masukan tetapi tidak mempunyai keluaran begitu pula sebaliknya, hindarilah proses yang mempunyai keluaran tetapi tidak mempunyai masukan.
5. Hati-hati dengan aliran data dan proses yang tidak dinamai. Aliran proses yang tidak diberi nama dapat mengakibatkan elemen data yang saling tidak berhubungan akan menjadi satu.

Sukamto dan Shalahuddin (2014:72) menjelaskan tentang tahapan-tahapan perancangan dengan menggunakan Data Flow Diagram yaitu:

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 kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya



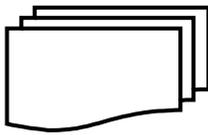
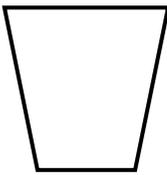
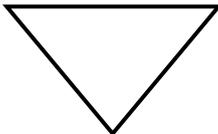
DFD Level 3,4,5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau 2.

### 2.2.2. Pengertian *Blockchart*

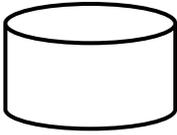
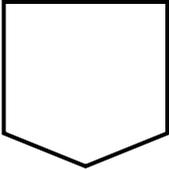
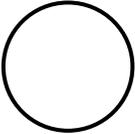
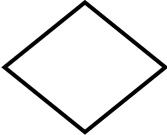
Kristanto (2008:75) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

Kristanto (2008:75) juga menjelaskan tentang 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 dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)

Lanjutan Tabel 2.2 Simbol-simbol dalam *Blockchart*

No	Simbol	Keterangan
6.		Data Penyimpanan ( <i>Storage</i> )
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
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 (2008:75)

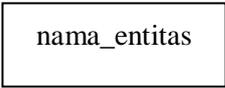
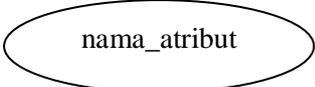
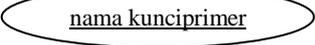
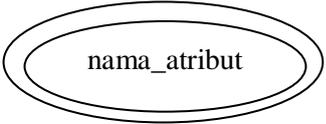


### 2.2.3. Pengertian ERD (*Entity Relationship Diagram*)

Sukamto et.al, (2014:50), “ERD adalah permodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk permodelan basis data relasional.”

Sukamto et.al, (2013:50), menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu:

**Tabel 2.3** Simbol-simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya adar dapat diakses 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.	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).
4.	Atribut multivalai / <i>multivalve</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

Lanjutan Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

No	Simbol	Keterangan
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan onetomany menghubungkan entitas A dan entitas B.

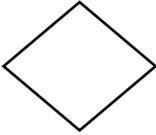
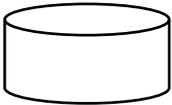
Sumber: Sukamto et.al (2014:50)

#### 2.2.4. Pengertian *Flowchart*

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

Fungsi *Flowchart* adalah untuk mengetahui alur sistem yang akan dilakukan oleh *user* dalam hal yang bertindak sebagai *user* adalah admin unit kerja, manajer departemen, admin departemen akuntansi, admin departemen anggaran, admin departemen akuntansi dan administrator sehingga kita dapat mengetahui proses yang dilakukan admin unit kerja untuk memasukkan data permintaan kode aset tetap dan proses yang dilakukan admin departemen akuntansi, admin departemen anggaran dan manajer departemen akuntansi untuk mendapatkan data permintaan kode aset tetap pada aplikasi yang telah kami buat.

Tabel 2.4 Simbol-simbol data *Flowchart*

No.	Simbol	Keterangan
1.		Terminal menyatakan awal atau akhir dari suatu algoritma.
2.		Menyatakan proses.
3.		Proses yang terdefinisi atau sub program.
4.		Menyatakan masukan dan keluaran ( <i>input/output</i> ).
5.		Menyatakan penyambung ke simbol lain dalam satu halaman.
6.		Menyatakan penyambung ke halaman lainnya.
7.		Menyatakan pencetakan (dokumen) pada kertas.
9.		Menyatakan desicion (keputusan) yang digunakan untuk penyeleksian kondisi di dalam program.
10.		Menyatakan media penyimpanan drum magnetik.

Lanjutan Tabel 2.4 Simbol-simbol data *Flowchart*

No	Simbol	Keterangan
11.		Menyatakan <i>input/output</i> menggunakan disket.
12.		Menyatakan operasi yang dilakukan secara manual.
13.		Menyatakan arah aliran pekerjaan (proses).
14.		Multidocument (banyak dokumen).
15.		Delay (penundaan atau kelambatan).

Sumber: Indrajani (2015:37)

### 2.2.5. Pengertian Kamus Data

Sukamto et.al, (2014:73), "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)."

Sukamto et.al, (2014:74), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

Tabel 2.5 Simbol-simbol dalam Kamus Data

No.	Simbol	Arti
1.	=	disusun atau terdiri atas
2.	+	Dan
3.	[[]]	baik ...atau...



4.	{ } <sup>n</sup>	n kali diulang/ bernilai banyak
5.	( )	data operasional
6.	*...*	batas komentar

Sumber: Sukanto et.al (2014:74)

## 2.3. Teori Judul

### 2.3.1. Pengertian Aplikasi

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

Jogiyanto, (2009:149), “Aplikasi merupakan program yang ditujukan untuk menyelesaikan suatu permasalahan dalam aplikasi yang tertentu yang sudah dibuat oleh pabrik pembuat perangkat lunak aplikasi.”

Kesimpulannya, aplikasi adalah suatu program yang digunakan untuk melakukan aktifitas dalam mengelola data informasi.

### 2.3.2. Pengertian Pengolahan

Kamus Besar Bahasa Indonesia (2008:979), menyatakan pengolahan adalah proses, cara perbuatan mengolah.

### 2.3.3. Pengertian Dana

Kamus Besar Bahasa Indonesia (2008:291), menyatakan dana adalah uang yang disediakan untuk suatu keperluan.

### 2.3.4. Pengertian Santunan

Kamus Besar Bahasa Indonesia (2008:1224), menyatakan santunan adalah uang yang diberikan sebagai pengganti kerugian karena kecelakaan, kematian, dan sebagainya.



### **2.3.5. Pengertian Anggota**

Kamus Besar Bahasa Indonesia (2008:64), menyatakan anggota adalah orang (badan) yang menjadi bagian atau masuk dalam suatu golongan (perserikatan, dewan, panitia, dan sebagainya).

### **2.3.6. Pengertian KORPRI**

Korps Pegawai Republik Indonesia, atau disingkat Korpri, adalah organisasi di Indonesia yang anggotanya terdiri dari Pegawai Negeri Sipil, pegawai BUMN, BUMD serta anak perusahaan, dan perangkat Pemerintah Desa. ([https://id.wikipedia.org/wiki/Korps\\_Pegawai\\_Republik\\_Indonesia](https://id.wikipedia.org/wiki/Korps_Pegawai_Republik_Indonesia) )

### **2.3.7. Pengertian Badan Kepegawaian dan Pengembangan Sumber Daya Manusia kota Palembang**

Badan Kepegawaian dan Pengembangan Sumber Daya Manusia Kota Palembang adalah salah satu instansi pemerintah kota Palembang yang memiliki tugas pokok melaksanakan penyusunan dan pelaksanaan kebijakan daerah di bidang kepegawaian.

### **2.3.8. Pengertian Aplikasi Pengolahan Dana Santunan bagi Anggota KORPRI pada Badan Kepegawaian dan Pengembangan Sumber Daya Manusia kota Palembang.**

Aplikasi Pengolahan Dana Santunan bagi Anggota KORPRI pada Badan Kepegawaian dan Pengembangan Sumber Daya Manusia kota Palembang adalah suatu aplikasi yang digunakan oleh bendahara KORPRI untuk mengelola data pegawai dan data pengajuan dana santunan sehingga mempermudah dalam melakukan pengolahan dana santunan.

## **2.4. Teori Program**

### **2.4.1. Pengertian Basis Data (*Database*)**

Raharjo (2011:3), “*database* didefinisikan sebagai kumpulan data yang



terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat.”

Pamungkas et.al (2013:90), “Basis data adalah kumpulan data yang tersimpan dalam tabel-tabel.”

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

Kesimpulannya, Basis Data adalah suatu media penyimpanan data yang saling berhubungan.

#### **2.4.2. Pengertian MySQL**

Raharjo (2011:21), “*MySQL* merupakan *software* RDBMS (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 berbarengan (*multi-threaded*).”

Winarno (2014:102), “MySQL merupakan tipe data rasional yang artinya MySQL menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan.”

Kesimpulannya, MySQL adalah suatu bahasa yang digunakan untuk melakukan manipulasi data yang saling terkait satu sama lain.

#### **2.4.3. Pengertian HTML**

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

Sujatmiko (2012:128), “ HTML (*Hypertext Markup Language*) adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *web page* atau program yang digunakan untuk menulis (membuat) halaman web di internet.”

Winarno (2014:1), “*HTML* adalah sebuah bahasa untuk menampilkan konten di Web.”



Kesimpulannya, *HTML* adalah bahasa pemrograman untuk membuat halaman web yang berfungsi memberi perintah ke browser untuk melakukan manipulasi tampilan melalui *tag-tag* yang ditulis dalam *HTML*.

#### 2.4.4. Pengertian CSS (*Cascading Style Sheets*)

Pamungkas et.al (2013:39), “CSS (*Cascading Style Sheets*) adalah kumpulan aturan-aturan pemformatan yang mengontrol tampilan konten dalam sebuah halaman web.”

Suryana et.al (2014:101), “CSS (*Cascading Style Sheets*) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu *website*, baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan.”

Kesimpulannya, CSS adalah bahasa pemrograman yang fungsi utamanya mendesain tampilan halaman web menjadi lebih cantik.

#### 2.4.5. Pengertian PHP (*Hypertext Preprocessor*)

Pamungkas et.al (2013:57), “PHP (*Hypertext Preprocessing*) merupakan bahasa pemrograman yang biasa digunakan untuk membuat halaman HTML.”

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

Kesimpulannya, PHP adalah bahasa pemrograman yang mudah digunakan dalam pengimplementasiannya.

##### 2.4.5.1. Tipe Data PHP

Pamungkas et.al (2013:60), PHP mempunyai tipe data dasar yaitu:

##### 1. *Integer*

*Integer* merupakan tipe data yang mencakup semua bilangan bulat. Range bilangan *integer* adalah antara -2.147.483.647 sampai dengan 2.147.483.647.

Contoh *integer* :

\$biaya = 1000;

\$a = 1234;

\$a = -123;



## 2. *Floating Point*

Floating point merupakan tipe data yang mencakup semua bilangan desimal (bilangan yang memiliki angka dibelakang koma). Range bilangan floating point antara  $1e-308$  sampai dengan  $1e308$  ( $10^{-308}$ s.d.  $10^{308}$ ).

Contoh floating point :

```
$a = 1.234;
```

```
$a = 1.2e3;
```

## 3. *Character*

*Character* merupakan tipe yang digunakan untuk menyimpan data-data yang berupa karakter (1 huruf). Penulisan nya biasanya diapitnya dengannya tandanya kutipnya satunya ('...').

Contoh *character* :

```
$abjad = 'a';
```

```
$abjad = 'b';
```

Jika karakter-karakter berkumpul membentuk suatu kata, kalimat atau paragraf maka tipe tadi tidak bisa lagi disebut sebagai *character* tapi bertipe *string*. *String* merupakan tipe data tersendiri dan tidak dapat dikelompokkan menjadi tipe data dasar. Penulisan nya biasa diapit dengan tanda kutip 2 ("...").

Contoh *string* :

```
$nama = "gus";
```

```
$salamat = "Jalan Tanjung";
```

## 4. *Boolean*

*Boolean* menyimpan data yang bernilai logika (benar atau salah, 0 atau 1, ya atau tidak).

### 2.4.5.2. Skrip PHP

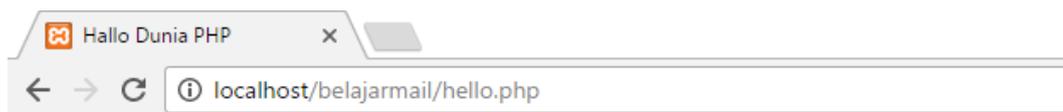
Winarno (2014:49), "PHP adalah bahasa pemrograman web berbasis server (server-side) yang mampu memarsing kode PHP dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser)."



*PHP* dapat dimasukkan pada kode *HTML* dalam satu *file*. Berikut ini contoh sintaks dasar penggunaan *PHP* :

```
<!DOCTYPE html>
<html>
<head>
  <title>Hallo Dunia PHP</title>
</head>
<body>
<h1>Hallo PHP</h1>
<?php
echo "<h2>Hallo, dunia!</h2>";
phpInfo();
?>
</body>
</html>
```

Bila dijalankan melalui browser, kode tersebut membentuk tampilan seperti berikut:



# Hallo PHP

## Hallo, dunia!

**Gambar 2.1** Tampilan *PHP* sederhana



#### 2.4.6. Hubungan *HTML*, *CSS*, *PHP*, dan *MySQL*

Ketika *user* akan mengakses data perusahaan maka *PHP* akan berusaha untuk mengambil data dari *MySQL* menggunakan salah satu fungsi yang ada di *PHP* yaitu *mysqli\_query()*. Setelah data diambil, maka *PHP* akan mengirimkan hasilnya ke *webserver* yang berbasis *HTML* dengan menggunakan salah satu fungsi *PHP* yaitu *echo()*. Selanjutnya *HTML* bertugas untuk menampilkan halaman web ke browser sesuai kebutuhan menggunakan tag-tag *HTML* dan dipercantik oleh *CSS*.

#### 2.4.7. *Javascript*

Suryana et.al (2014:101), “*Java Script* adalah bahasa *script* berdasarkan pada objek yang memperolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen *HTML*.”

Sidik (2011:1), “*Javascript* adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen *HTML* yang ditampilkan dalam browser menjadi lebih interaktif, tidak sekedar indah saja.”

*Javascript* merupakan bahasa pemrograman *clientside*. Kode *javascript* dituliskan dalam bentuk fungsi yang ditaruh di tak `<head>` yang dibuka dengan tag `<script type="text/javascript" src="alamat.js">` dan diakhir dengan tag `</script>`.

##### 2.4.7.1. Tipe Data, Variabel dan Operator *JavaScript*

Sidik (2011:36) menjelaskan variabel dan operator yang biasa digunakan *javascript*:

##### 1. Tipe Data

Tipe data yang dapat diproses dalam *Javascript*, dibedakan menjadi :

##### a. Numerik

Data numerik adalah data berupa bilangan atau angka yang dapat dikenai operasi matematika, seperti penjumlahan, pengurangan, perkalian, atau pembagian.

##### b. *String*



Data *string* adalah semua data berupa huruf, tanda baca, angka, dan rangkaian huruf dan angka.

c. *Boolean*

Data *boolean* adalah data yang terdiri atau dua nilai, benar atau salah, *true* atau *false*.

d. *Date*

Data *date* adalah data khusus yang menyatakan tentang waktu – waktu tanggal dan jam.

e. *Objek*

Tipe data objek adalah tipe data yang mendefinisikan objek.

f. *RegExp*

Tipe data *RegExp* adalah tipe data yang digunakan untuk mendefinisikan pola atau *regularexpression (regex)*.

g. *Null* dan *undefine*

Tipe data *null* dan *undefine* adalah tipe data khusus. *Null* digunakan untuk mendefinisikan data kosong (*null*), sedangkan *undefined* didefinisikan untuk data yang belum didefinisikan untuk data yang belum didefinisikan, bernilai *null*.

## 2. Variabel

Variabel adalah tempat untuk menyimpan data atau informasi yang akan diproses, berada dalam memori komputer.

a. *Variabel Tunggal*

Variabel tunggal adalah variabel yang digunakan untuk menyimpan satu data.

b. *Array*

*Array* adalah variabel jamak, yaitu satu nama variabel yang dapat memiliki banyak data; beberapa variabel yang diacu dengan satu nama.

c. *Lingkup dan Umur Variabel*

Tempat mendeklarasi suatu variabel akan menentukan lingkup atau ruang variabel itu dikenali dalam program.

## 3. Operator



Operator adalah alat yang digunakan untuk melakukan pemrosesan data.

a. Operator Berdasar Jumlah *Operand*.

Operator menurut jumlah *operand* yang dimanipulasinya :

- i. *Unary* adalah operator yang beroperasi ada *operand* tunggal.
- ii. *Binaryoperand* yang beroperasi untuk dua *operand*.

b. Operator Berdasar Tipe Data

1. Operator *Aritmatika*

Operator *aritmatika* digunakan untuk melakukan pemrosesan data numerik hasilnya adalah data numerik.

2. Operator *Assignment*

Operator *assignment* adalah operator yang digunakan untuk menyatakan suatu pernyataan perintah *assignment* dan ekspresi *aritmatika* dalam bentuk yang disingkat.

3. Operator Bit (*Bitwise Operator*)

4. Operator *String*

Operator *string* adalah operator yang digunakan untuk melakukan pemrosesan data *string* hasilnya berupa data *string*.

5. Operator Perbandingan

Operator perbandingan digunakan untuk melakukan perbandingan dua data, hasilnya berupa nilai *boolean* benar atau salah.

6. Operator Logika

Operator logika (*boolean*) digunakan untuk melakukan pemrosesan data *boolean* hasilnya adalah data *boolean*.