



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

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

Menurut Wahyudin dan Munir (2018:1), “Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses *input* sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan *output* dalam bentuk informasi.”

Berdasarkan definisi di atas, dapat disimpulkan bahwa komputer adalah suatu alat elektronik yang berguna dalam melaksanakan berbagai pekerjaan manusia. Mampu melakukan tugas seperti mengolah dan memproses *input* sesuai instruksi lalu menghasilkan informasi yang dibutuhkan.

##### 2.1.2 Pengertian Perangkat Lunak

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 Sukanto et al. (2018:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Berdasarkan definisi di atas, dapat disimpulkan bahwa perangkat lunak adalah program komputer berupa instruksi-instruksi yang diarahkan pada komputer sehingga dapat melaksanakan tugas sesuai kebutuhan pemakai sistem. program komputer yang juga terasosiasi dengan berbagai dokumen perangkat lunak.



### 2.1.3 Pengertian Internet

Menurut Sudarsono (2021:69), “*Interconnected network* atau yang lebih populer dengan sebutan internet adalah sebuah sistem komunikasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia.”

Menurut Sidik (2019:1), “Internet adalah jaringan dari jaringan komputer seluruh dunia yang saling terhubung internet. Internet merupakan jaringan yang menghubungkan seluruh komputer di dunia melalui banyak media komunikasi, ada yang terhubung ke dalam jaringan komputer dunia melalui kabel, saluran telepon kabel, telepon seluler, radio dan atau satelit.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa internet adalah jaringan komputer yang luas sehingga dapat menghubungkan komputer di seluruh dunia dengan berbagai media komunikasi yang terhubung dan di dalamnya terdapat sumber daya informasi statis, dinamis dan interaktif.

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

Menurut Sutanta (2011:32), “Basis data (*database*) adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data, disimpan dengan cara-cara tertentu sehingga mudah digunakan atau ditampilkan kembali.”

Menurut Firly (2019:110), “Basis Data (*database*) adalah sebuah tempat menyimpan yang memproses suatu informasi secara terstruktur dalam bentuk elektronik.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa basis data (*database*) adalah suatu tempat untuk menyimpan data serta dapat mengelola kumpulan data yang saling berhubungan satu dengan yang lainnya menjadi informasi yang tersimpan secara sistematis di komputer sehingga dapat digunakan kembali saat dibutuhkan.



### 2.1.5 Metode Pengembangan Sistem

Menurut Sukamto dan Shalahuddin (2018:28-30), “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*). Berikut penjelasannya:

1. Analisis

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

2. 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 pengodean.

3. 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.

4. 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.

5. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

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.



## **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 Marjito dan Tesaria (2016:42), “Secara umum pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju.”

Berdasarkan definisi di atas, dapat disimpulkan bahwa aplikasi adalah suatu program yang siap digunakan dalam melaksanakan tugas-tugas yang dibutuhkan pengguna.

### **2.2.2 Pengertian Pengajuan**

Pengajuan adalah proses, cara, perbuatan mengajukan dari tindakan pengusulan atau menyerahkan sesuatu dari satu pihak kepada suatu pihak lain.

### **2.2.3 Pengertian Pembuatan**

Pembuatan adalah proses dalam merancang dan menciptakan sesuatu sehingga dapat menghasilkan sesuatu yang diinginkan.

### **2.2.4 Kartu Pegawai (KARPEG)**

Kartu Pegawai (KARPEG) merupakan kartu identitas Aparatur Sipil Negara (ASN) yang berguna sebagai kelengkapan administrasi kepegawaian. Salah satunya adalah pengajuan kenaikan pangkat dan pensiun. Kartu ini hanya berlaku ketika yang bersangkutan menjadi ASN.

### **2.2.5 Pengertian Kartu Istri (KARIS) dan Kartu Suami (KARSU)**

Kartu Istri (KARIS) dan Kartu Suami (KARSU) merupakan kartu identitas istri atau suami Aparatur Sipil Negara (ASN) yang berguna sebagai kelengkapan administrasi kepegawaian dan hanya berlaku ketika yang bersangkutan menjadi istri atau suami sah ASN.



### 2.2.6 Pengertian *Framework Codeigniter*

Menurut Asyikin (2019:16), “*Codeigniter* merupakan suatu *Web Application Framework* (WAF) yang dirancang khusus untuk mempermudah *developer web* dalam mengembangkan aplikasi berbasis *web*. *Codeigniter* berisi kumpulan kode berupa pustaka (*library*) dan alat (*tools*) yang dipadukan sedemikian rupa menjadi suatu kerangka kerja (*framework*).”

Menurut Sulistiono (2018:7), “*Codeigniter* adalah sebuah aplikasi *open source* yang berupa kerangka kerja atau *framework* untuk membangun *website* menggunakan bahasa pemrograman PHP.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa *framework codeigniter* merupakan sebuah aplikasi *open source* berupa kerangka kerja (*framework*) untuk mempermudah *developer web* dalam membangun dan mengembangkan aplikasi berbasis *website* dengan menggunakan bahasa pemrograman PHP.

### 2.2.7 Pengertian Judul Secara Keseluruhan

Aplikasi Pengajuan Pembuatan Kartu Pegawai, Kartu Istri dan Kartu Suami pada Badan Kepegawaian dan Pengembangan Sumber Daya Manusia Kota Palembang dengan *Framework Codeigniter* adalah sebuah aplikasi yang dapat mempermudah Aparatur Sipil Negara (ASN) di lingkungan Pemerintah Kota Palembang dalam mengajukan pembuatan kartu pegawai, kartu istri dan kartu suami secara *online* tanpa harus datang langsung ke Badan Kepegawaian dan Pengembangan Sumber Daya Manusia Kota Palembang.

## 2.3 Teori Khusus

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

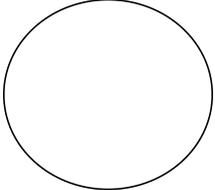
Menurut Sukamto dan Shalahuddin (2018:69-73), “*Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

Adapun beberapa simbol atau notasi-notasi yang sering digunakan di



dalam perancangan DFD seperti pada tabel berikut:

**Tabel 2.1** Simbol–simbol *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: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>
2.		<p><i>File</i> atau basis data 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 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>

Lanjutan Tabel 2.1 Simbol–simbol *Data Flow Diagram* (DFD)

No.	Notasi	Keterangan
3.		<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau <i>system</i> 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>
4.		<p>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>).</p> <p>Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata dan misalnya “siswa”.</p>

(Sumber: Sukamto dan Shalahuddin, 2018:70-72)

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membentuk 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 pada DFD *Level 1* dan seterusnya.

### 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 *Level 2*.

## 2.3.2 Pengertian *Blockchart*

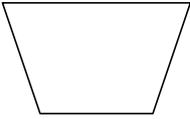
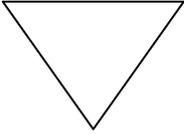
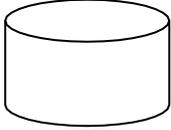
Kristanto (2018:75) menjelaskan, “*Block chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Block chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

Kristanto (2018:75) menjelaskan, “Simbol-simbol yang sering digunakan dalam *block chart* dapat dilihat pada tabel berikut ini:

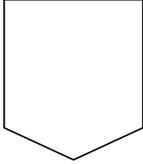
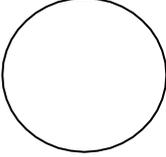
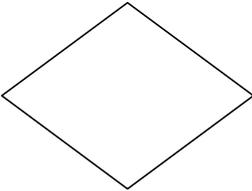
**Tabel 2.2** Simbol-simbol *Blockchart*

No.	Simbol	Arti
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku, berkas atau cetakan.

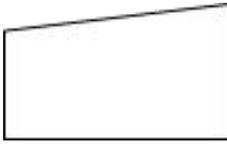
Lanjutan **Tabel 2.2** Simbol-simbol *Blockchart*

No.	Simbol	Arti
2.		Multi dokumen.
3.		Proses manual.
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual).
6.		Data penyimpanan ( <i>data storage</i> ).
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.

Lanjutan Tabel 2.2 Simbol-simbol *Blockchart*

No.	Simbol	Arti
8.		Terminasi yang mewakili simbol tertentu untuk 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> ).

Lanjutan Tabel 2.2 Simbol-simbol *Blockchart*

No.	Simbol	Arti
12.		Layar peraga ( <i>monitor</i> ).
13.		Pemasukkan data secara manual.

Sumber: Kristanto (2018:75-77)

### 2.3.3 Pengertian Bagan Alir (*Flowchart*)

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

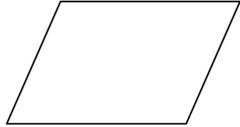
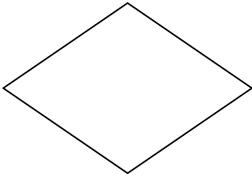
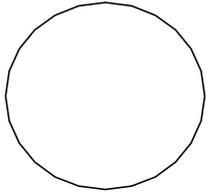
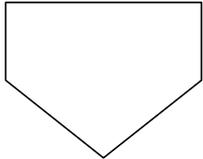
Menurut Ladjamudin (2013:263), “*Flowchart* adalah bagan bagan yang mempunyai arus yang menggambarkan langkah langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma.”

Adapun simbol-simbol dalam *flow chart* adalah sebagai berikut:

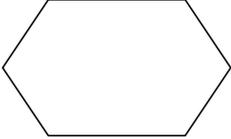
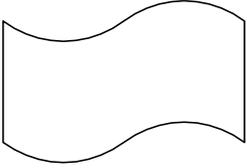
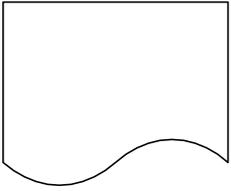
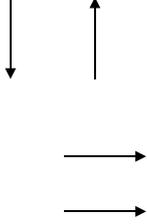
Tabel 2.3 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1.		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program.

Lanjutan Tabel 2.3 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
2.		<i>Input / output</i>	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya.
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.

Lanjutan Tabel 2.3 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
7.		<i>Predefined process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
8.		<i>Fanned Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> data ke kartu.
9.		<i>Punch Tape</i>	Berfungsi untuk <i>input</i> atau <i>ouput</i> yang menggunakan pita kertas berlubang.
10.		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i> ).
11.		<i>Flow</i>	Menyatakan jalannya arus suatu proses.

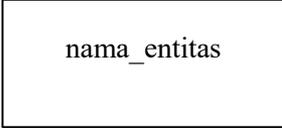
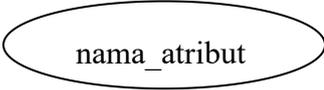
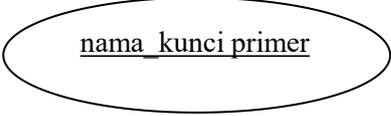
(Sumber: Lamhot, 2015:14-15)



### 2.3.4 Pengertian *Entity Relationship Diagram* (ERD)

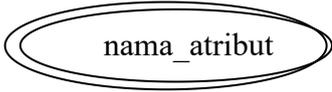
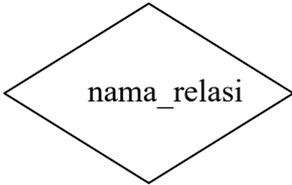
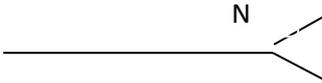
Menurut Sukamto dan Shalahuddin (2018:50-51), “*Entity Relationship Diagram* (ERD) merupakan permodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional.” Adapun simbol-simbol yang digunakan dalam *Entity Relationship Diagram* (ERD), yaitu:

**Tabel 2.4** Simbol-simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Deskripsi
1.	Entitas / <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas.
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.



Lanjutan **Tabel 2.4** Simbol-Simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Deskripsi
4.	Atribut Multi nilai / <i>Multivalued</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 / <i>Association</i>  	Penghubung antara relasi dan entitas di mana di kedua ujungnya punya <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut

(Sumber: Sukanto dan Shalahuddin, 2018:50-51)

### 2.3.5 Pengertian Kamus Data

Kristanto (2018:72), “Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem.”

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

No.	Notasi	Arti
1.	=	Terdiri atas

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

No.	Notasi	Arti
2.	+	<i>Dan</i>
3.	()	Opsional
4.	[ ]	Memilih salah satu alternatif
5.	**	Komentar
6.	@	Identifikasi atribut kunci
7.		Pemisah alternatif simbol [ ]

(Sumber: Kristanto, 2018:72)

## 2.4 Teori Program

### 2.4.1 Pengertian XAMPP

Menurut Madcoms (2016:186), “XAMPP adalah sebuah paket kumpulan software yang terdiri dari Apache, MySQL, Php MyAdmin, PHP, Perl, Filezilla dan lain -lain.”

Menurut Februariyanti dan Zuliarso (2012:129), “XAMPP adalah sebuah *software web server* apache yang didalamnya sudah tersedia *database server* MySQL dan dapat mendukung pemrograman PHP. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa XAMPP adalah sebuah *software web server* yang didalamnya terdiri dari Apache, MySQL, Php MyAdmin, PHP dan lain - lain dan dapat mendukung bahasa pemrograman PHP.

### 2.4.2 Pengertian MySQL

Menurut Mudzir MF (2018:217), “MySQL adalah *system* manajemen *database* SQL yang sifatnya *open source* (terbuka) dan paling banyak digunakan saat ini.”

Menurut Rusmawan (2019:97), “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS



yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa MySQL adalah *system* manajemen *database* SQL yang bersifat *open source* (terbuka). Sehingga paling banyak digunakan saat ini, karena siapa saja boleh menggunakannya.

### 2.4.3 Pengertian PHP

Menurut Enterprise (2018:1) menjelaskan bahwa “PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif.”

Menurut Mudzir MF (2018:3), “PHP berasal dari kata "*Hypertext Preprocessor*", yaitu bahasa pemrograman universal untuk penanganan pembuatan dan pengembangan sebuah situs *web* dan biasa digunakan bersamaan dengan HTML.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa PHP merupakan bahasa pemrograman universal yang digunakan untuk membuat dan mengembangkan sebuah situs *website* agar dinamis dan interaktif.

#### 2.4.3.1 Sintaks Dasar PHP

Yuana (2015:2), menjelaskan kode-kode PHP dituliskan diantara tanda berikut ini:

```
<?php
```

```
...
```

```
...
```

```
...
```

```
?>
```

Atau

```
<?
```

```
...
```

```
...
```



...

?>

Apabila membuat kode PHP dan berencana akan mendistribusikan ke pihak atau orang lain, maka usahakan menggunakan sintaks `<?php ... ?>`. Hal ini dikarenakan untuk penggunaan kode yang menggunakan `<? ... ?>` terkadang tidak bisa dijalankan dalam *server* tertentu.

### 2.4.3.2 Tipe Data PHP

Tipe data merupakan jenis dari suatu data yang akan diproses oleh bahasa pemrograman. 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 -2.147.4833.647 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 variabel yang di set 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 MySQL Front

Menurut Supriyatno (2010:72), “MySQL Front merupakan sebuah *software* yang digunakan untuk mengakses dan mengelola *database* MySQL dengan tampilan GUI sehingga akan lebih memudahkan dan mempercepat pengelolaan *database* MySQL.”

Menurut Jovan (2007:9), “*Software* MySQL Front adalah salah satu dari sekian *software* yang mampu membuat dan mengolah *file* atau *script database* dengan tidak mengetikkan bahasa pemrogramannya.”

Berdasarkan pendapat di atas, dapat disimpulkan bahwa MySQL Front merupakan sebuah *software* yang digunakan dalam mengakses, membuat dan mengolah *database* MySQL dengan tampilan *Graphical user interface* (GUI).