



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

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

Paramytha (2016:6), “Komputer adalah alat hitung dengan konstruksi elektronik, yang dapat menerima input dan mengolah data, memberikan informasi, mempunyai internal storage (untuk menyimpan program dan data yang diolah) dan bekerja secara otomatis dengan bantuan operating sistem menurut program yang disimpan di dalamnya”.

Kesimpulannya, Komputer adalah peralatan elektronik yang dapat mengolah data digital untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia secara otomatis dengan mengikuti serangkaian perintah.

2.1.2 Pengertian Internet

Sofana (2015:5), “*Internet* adalah interkoneksi jaringan komputer skala besar (mirip WAN), yang dihubungkan menggunakan protokol khusus”.

Rusman dalam Belajar dan pembelajaran Berorientasi Standar Proses Pendidikan (2017:235) menyatakan : “*Internet* atau *international networking* didefinisikan dua komputer atau lebih yang memiliki konektivitas membentuk jaringan komputer hingga meliputi jutaan komputer di dunia secara global (internasional), yang saling berinteraksi dan bertukar informasi.”

Kesimpulannya, *Internet* adalah interkoneksi jaringan global yang menghubungkan berjuta-juta komputer menggunakan protokol untuk saling bertukar informasi

2.1.3 Pengertian Perangkat Lunak

Rosa dan Shalahuddin (2014:02), “Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*).”

Sutabri (2014:06), “Perangkat lunak merupakan program-program komputer yang berguna untuk menjalankan suatu pekerjaan sesuai dengan yang dikehendaki.”

Kesimpulannya, Perangkat lunak adalah program komputer yang terasosiasi untuk menjalankan suatu pekerjaan dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan.

2.1.4 Pengertian Sistem

Pratama (2014:02), “Sistem berarti kumpulan komponen yang saling terkait dan mempunyai satu tujuan yang ingin dicapai”.

Rosa dan Shalahudin (2014:02), “Sistem berarti kumpulan komponen yang saling terkait dan mempunyai satu tujuan yang ingin dicapai”.

Kesimpulannya, Sistem adalah kumpulan komponen yang saling berhubungan berdasarkan fungsinya agar menjadi satu kesatuan untuk mencapai tujuan tertentu.

2.1.5 Metode Pengembangan Sistem

Rosa dan Shalahudin (2014:28) 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 tahap pendukung (*support*).

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.

e. Pendukung (*support*) atau 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. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

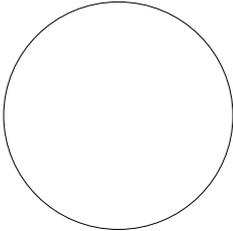
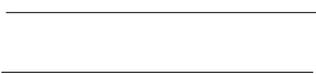
2.2 Teori Khusus

2.2.1 Pengertian DFD (*Data Flow Diagram*)

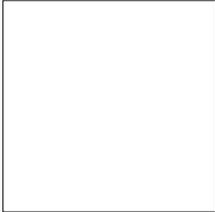
Rosa dan Shalahuddin (2014:70) menjelaskan, “*Data Flow Diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*ouput*).

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

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No.	Notasi	Keterangan
1.		<p>Proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</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 pemograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harusnya sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</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*

No.	Notasi	Keterangan
3.		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 Catatan : Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda
4.		Aliran data: merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses masukan (<i>input</i>) atau keluaran (<i>output</i>) 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 : Rosa dan Shalahuddin, 2014:71-72)

Rosa dan Shalahuddin (2014:72-73) 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 *system* 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.

2.2.2 Pengertian *Blockchart*

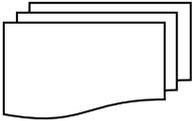
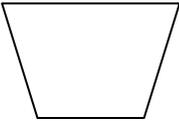
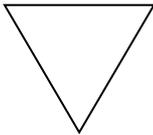
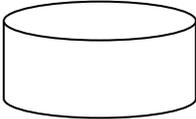
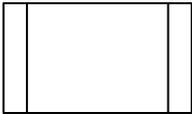
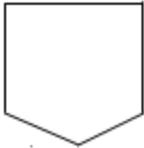
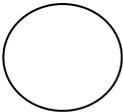
Kristanto (20018:68) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”. Pembuatan blockhart harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

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

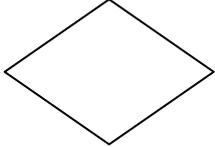
Tabel 2.2 Simbol-simbol *Block chart*

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

Lanjutan Tabel 2.2 Simbol-simbol *Block chart*

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
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

Lanjutan Tabel 2.2 Simbol-simbol *Block chart*

No.	Simbol	Arti
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>)
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukkan data secara manual.

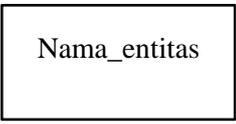
(Sumber : Kristanto, 2018:68-70)

2.2.3 Pengertian ERD (*Entity Relational Diagram*)

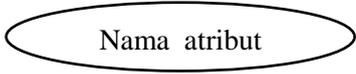
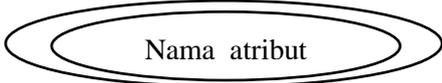
Rosa dan Shalahuddin (2013:50) menjelaskan, “*Entity Relational Diagram* (ERD) digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain.

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

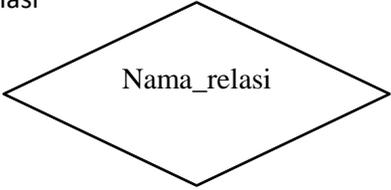
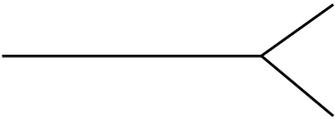
Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

No.	Simbol	Deskripsi
1.	Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki

Lanjutan Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

No.	Simbol	Deskripsi
		data dan harus disimpan datanya agar 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 multivali / multivalue 	<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 Relational Diagram*

No.	Simbol	Deskripsi
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6.	Asosiasi / <i>association</i> 	Penghubung antar 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 <i>one to many</i> menghubungkan entitas A dan entitas B maka

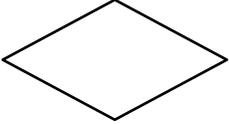
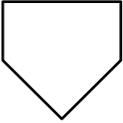
(Sumber: Rosa dan Shalahuddin , 2013:50-51)

2.2.4 Pengertian *Flowchart*

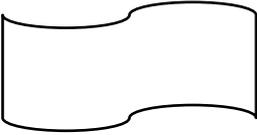
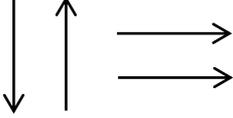
Lamhot (2015:14) menjelaskan, “*Flowchart* adalah langkah-langkah menyelesaikan masalah yang dituliskan dalam simbol-simbol tertentu.”

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.
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.

Lanjutan Tabel 2.4 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
8.		<i>Fancher Card</i>	Menyatakan input berasal dari kartu atau output data ke kartu.
9.		<i>Punch Tape</i>	
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

Rosa dan Shalahuddin (2014:73) 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

No.	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik...atau...
4.	{ }n	N kali diulang/bernilai banyak

Lanjutan Tabel 2.5 Simbol-simbol Kamus Data

No.	Simbol	Keterangan
5.	()	Data opsional
6.	*...*	Batas komentar

(Sumber: Rosa dan Shalahuddin, 2013:74)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Chan (2017:4), “Aplikasi adalah koleksi windows dan objek-objek yang menyediakan fungsi untuk aktivitas user, seperti pemasukan data, proses, dan pelaporan”.

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

Kesimpulannya, Aplikasi adalah suatu program komputer yang menyediakan fungsi untuk aktivitas user dan penyelesaian tugas-tugas tertentu.

2.3.2 Sekilas Tentang *E-commerce*

2.3.2.1 Pengertian *E-Commerce*

Pratama (2015:02), “Istilah *E-Commerce* mulai muncul di tahun 1990-an melalui adanya internet untuk mengubah paradigma transaksi jual beli dan pembayaran dari cara konvensional ke dalam bentuk digital elektronik berbasis komputer dan jaringan internet”.

2.3.2.2 Faktor-Faktor Pendukung *E-Commerce*

Muncul dan berkembangnya aplikasi dan layanan *E-Commerce* tidak dapat lepas dari adanya beberapa hal berikut ini :

1. Perkembangan teknologi informasi, terutama akses internet, yang dapat mudah dijangkau oleh setiap pengguna komputer di seluruh dunia.

2. Perkembangan perangkat keras komputer yang memungkinkan komputer dekstop, komputer jinjing (Notebook, Netbook), perangkat mobile (Smartphone, Handphone), Tablet, PDA, untuk mudah terhubung ke dalam jaringan internet secara otomatis.
3. Perkembangan perangkat lunak komputer dalam bentuk aplikasi dan sistem operasi, yang memungkinkan pengguna untuk semakin mudah dalam mengoperasikan komputer dan menggunakan semua layanan internet, termasuk juga *E-Commerce*. Bagi para pengembang dalam pembuat aplikasi (*Developer*), kemajuan di bidang perangkat lunak sangat membantu di dalam pembuatan aplikasi dan layanan *E-Commerce*, misalkan yang berbasis *website*.
4. Perkembangan jaringan komputer yang makin meluas (*Scalable*) dan handal (*Reliable*), termasuk juga keamanan di dalam jaringan komputer (*Network Security*).
5. Gaya hidup masyarakat, terutama masyarakat perkotaan besar, yang makin mudah dan ingin dimudahkan. Termasuk juga di dalam belanja secara *online*.
6. Dukungan dari pihak lain, misalkan Bank memberikan fasilitas pembayaran secara *online*, kartu kredit dan lain-lain.

2.3.2.3 Dua Jenis Pembagian E-Commerce

Dilihat dari interaksi di dalamnya yang melibatkan pihak-pihak tertentu beserta elemen lainnya, maka secara garis besar, *E-Commerce* dibagi menjadi dua bagian utama. Kedua bagian tersebut meliputi *Business to Business* (B2B) dan *Business to Consumer* (B2C) atau Retail. Pembahasan keduanya akan disajikan pada setiap subbab berikut ini .

2.3.2.3.1 *Business to Business* (B2B)

Business to business yaitu bentuk interaksi *E-Commerce* secara *online* yang terjadi antara produsen (perusahaan, industri rumah tangga, penyedia barang dan jasa) dengan distributor dan pengecer, untuk kemudian di salurkan ke konsumen masing-masing. Bentuk interaksi ini bersifat umum dan tidak langsung berinteraksi ke konsumen akhir yang memerlukan barang dan jasa tersebut.

2.3.2.3.2 Business to Consumer (B2C)

Business to Consumer (B2C) merupakan bagian dari *E-Commerce* yang menekankan kepada proses pemesanan, pembelian, dan penjualan produk atau jasa melalui akses internet. Hal itu berarti bahwa penjual dan pembeli dapat bertemu dan bertransaksi secara elektronik dan *online*, memanfaatkan fitur-fitur yang disediakan. Misalkan keranjang belanja virtual dan pembayaran secara elektronik memanfaatkan kartu kredit.

2.3.3 Pengertian Bahan Bangunan

Kamus Besar Bahasa Indonesia menjelaskan Bahan Bangunan adalah materia, segala barang yang digunakan membangun rumah atau gedung.

2.3.4 Pengertian Android

Yudha dan Wijayanto (2017:1), “Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet”.

2.3.5 Pengertian Aplikasi *E-commerce* Bahan Bangunan pada CV Dwi Putri Berdikari Berbasis *Android*

Aplikasi *E-commerce* Bahan Bangunan pada CV Dwi Putri Berdikari Berbasis *Android* adalah suatu aplikasi yang berisi informasi pembelian, pemasaran material bahan bangunan untuk menunjang kegiatan perdagangan yang dataya terkoneksi secara langsung dengan *internet* sehingga memudahkan informasi perdagangan secara merata kepada konsumen dan dapat di akses melalui telepon pintar dan komputer tablet untuk melakukan pemesanan dan pembelian.

2.4 Teori Program

2.4.1 Pengertian Basis Data

Kadir (2017:103), “Basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi”.

Kristanto (2018:79), “Basis data adalah kumpulan data, yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi”.



Kesimpulannya, Basis data merupakan kumpulan data yang yang disimpan secara sistematis untuk memperoleh informasi dari sebuah organisasi.

2.4.2 Pengertian XAMPP

Riyanto (2014:11), “XAMPP merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP.”

Wahana (2014:72), “XAMPP adalah *tool* yang menyediakan paket perangkat lunak dalam satu buah paket”.

Kesimpulannya, XAMPP merupakan paket PHP dan MySQL berbasis *open source* oleh sebuah komunitas *open source* yang digunakan sebagai *tool* pembantu dalam pengembangan aplikasi berbasis PHP.

2.4.3 Pengertian MySQL

Ardhana (2014:46), “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”.

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

Kesimpulannya, MySQL merupakan perangkat lunak database relasional yang *multithread*, *multi-user* yang dapat mengelola database dengan cepat.

2.4.4 Pengertian PHP

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

Subagia (2018:1), “PHP merupakan sebuah bahasa pemrograman yang berjalan dalam sebuah web server (*server side*)”.

Kesimpulannya, PHP merupakan bahasa pemrograman yang disisipkan HTML untuk membuat website dinamis dan interaktif dalam sebuah web server..



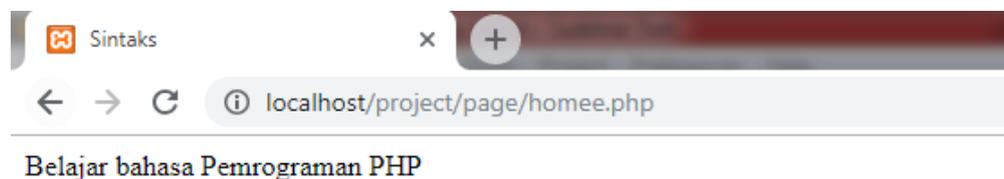
Gambar 2.1. Tampilan Logo PHP

2.4.4.1 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>Sintaks</TITLE>
  </HEAD>
<BODY>
  <?php
    echo "Belajar bahasa Pemrograman PHP";
  ?>
</BODY>
</HTML>
```



Gambar 2.2 Contoh Script PHP

2.4.4.2 Tipe Data PHP

Tipe data PHP digunakan untuk menentukan jenis data yang akan disimpan dalam suatu variabel. Raharjo (2015:37-40), menjelaskan PHP mempunyai empat tipe data dasar sebagai berikut :

1. **Integer (Bilangan Bulat)** adalah tipe data yang merepresentasikan bilangan utuh atau bilangan bulat. Rentang nilai yang diperbolehkan pada tipe data integer adalah dari -2,147,483,648 sampai +2,147,483,647.
2. **Floating-point (Bilangan Riil)** adalah tipe data yang merepresentasikan nilai-nilai numerik dalam bentuk pecahan atau mengandung angka desimal di belakang koma. Tipe *floating-point* berada dalam rentang $1.7E-308$ dan $1.7E+308$ dengan 15 digit akurasi.
3. **Karakter** merupakan tipe data yang digunakan untuk merepresentasikan data dengan nilai karakter tunggal, misal 'A', 'z', 'o', dan sebagainya. Karakter dalam PHP diapit dengan menggunakan tanda petik tunggal ('...').
4. **String** adalah tipe data yang merepresentasikan data yang berupa teks (kumpulan karakter). Penulisan tipe *string* diapit oleh petik ganda ("...") dan tunggal ('...').
5. **Boolean** merupakan tipe data yang digunakan untuk menyatakan nilai true (benar) atau false (salah) dan digunakan dalam proses pemilihan atau pengulangan statement, di bagian kondisi.
6. **Array** adalah variabel yang menyimpan sekelompok nilai, yang dapat diidentifikasi atau diakses berdasarkan posisinya atau dengan nama yang telah didefinisikan sebelumnya. Dalam PHP, indeks *array* dimulai dari 0 bukan 1.

2.4.5 Pengertian JavaScript

Sidik (2018:1), "JavaScript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak hanya sekedar indah saja."



2.4.5.1 Dasar JavaScript

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

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

```
<script>
```

Kode JavaScript ditempatkan di sini.

```
</script>
```