



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Teori Umum**

##### **2.1.1 Pengertian Aplikasi**

Aplikasi merupakan program yang berisikan perintah-perintah untuk melakukan pengolahan data. Jadi aplikasi secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal. Jogiyanto (2004:4)

Perangkat lunak/aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak system yang mengintegrasikan berbagai kemampuan computer

##### **2.1.2 Pengertian Pemesanan**

Proses pembelian yang dilakukan pembeli individu ataupun perusahaan, maka dibutuhkan perencanaan dan pelaksanaan kegiatan pemesanan yang matang agar hasil yang dicapai maksimal. Sistem informasi pemesanan merupakan subsystem informasi bisnis yang mencakup kumpulan prosedur yang melaksanakan, mencatat, mengkalkulasi, membuat dokumen dan informasi pemesanan untuk keperluan manajemen dan bagian lain yang berkepentingan, mulai dari diterimanya order pesanan sampai mencatat laporan penjualan. (Eka, 2010:3).

##### **2.1.2 Pengertian Catering**

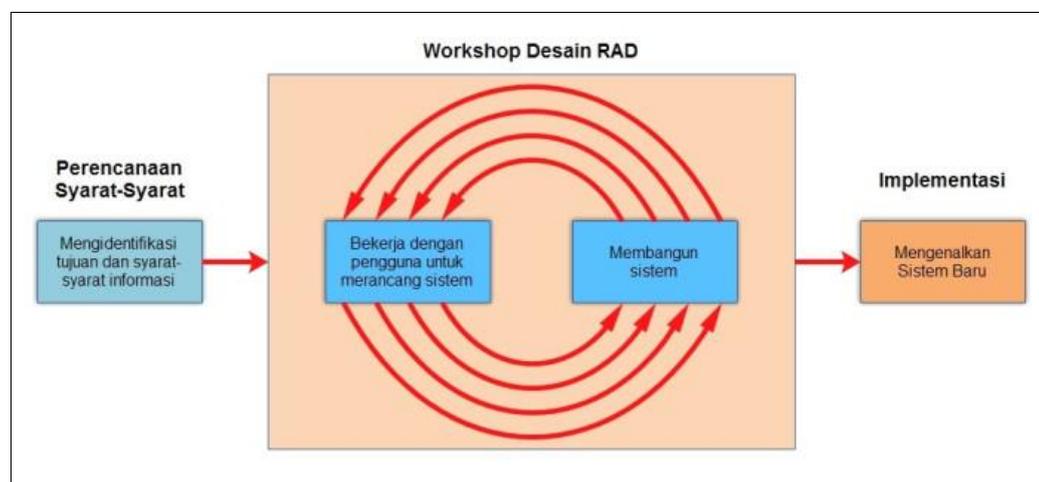
menurut Sjahmien Moehyi (1992 : 5) menyatakan bahwa katering adalah jenis penyelegaraan makanan yang tempat memasak makanan berbeda dengan tempat menghidangkan makanan. Makanan jadi diangkut ke tempat lain untuk dihidangkan, misalnya ke tempat penyelenggaraan pesta, rapat, pertemuan, kantin atau kafeteria industri. Makanan yang disajikan dapat berupa makanan kecil dan besar.

## 2.2 Teori Khusus

### 2.2.1 Pengertian Rapid Application Development

Menurut Kendall (2010), RAD adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. RAD bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi. Pada akhirnya, RAD sama-sama berusaha memenuhi syarat-syarat bisnis yang berubah secara cepat.

Menurut Kendall (2010), terdapat tiga fase dalam RAD yang melibatkan penganalisis dan pengguna dalam tahap penilaian, perancangan, dan penerapan. Adapun ketiga fase tersebut adalah *requirements planning* (perencanaan syarat-syarat), *RAD design workshop* (*workshop* desain RAD), dan implementasi. Sesuai dengan metodologi RAD menurut Kendall (2010), berikut ini adalah tahap-tahap pengembangan aplikasi dari tiap-tiap fase pengembangan aplikasi.



Sumber : (Kosasi, Ayu 2015).

Gambar 1. *Workshop* Desain RAD

### Fase dan Tahapan Pengembangan Aplikasi

#### A. *Requirements Planning* (Perencanaan Syarat-Syarat)



Dalam fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut. Orientasi dalam fase ini adalah menyelesaikan masalah-masalah perusahaan. Meskipun teknologi informasi dan sistem bisa mengarahkan sebagian dari sistem yang diajukan, fokusnya akan selalu tetap pada upaya pencapaian tujuan-tujuan perusahaan (Kendall, 2010).

**B. *RAD Design Workshop (Workshop Desain RAD)***

Fase ini adalah fase untuk merancang dan memperbaiki yang bisa digambarkan sebagai *workshop*. Penganalisis dan pemrogram dapat bekerja membangun dan menunjukkan representasi visual desain dan pola kerja kepada pengguna. *Workshop* desain ini dapat dilakukan selama beberapa hari tergantung dari ukuran aplikasi yang akan dikembangkan. Selama *workshop* desain RAD, pengguna merespon prototipe yang ada dan penganalisis memperbaiki modul-modul yang dirancang berdasarkan respon pengguna. Apabila seorang pengembangnya merupakan pengembang atau pengguna yang berpengalaman, Kendall menilai bahwa usaha kreatif ini dapat mendorong pengembangan sampai pada tingkat terakselerasi (Kendall, 2010).

**C. *Implementation (Implementasi)***

Pada fase implementasi ini, penganalisis bekerja dengan para pengguna secara intens selama *workshop* dan merancang aspek-aspek bisnis dan nonteknis perusahaan. Segera setelah aspek-aspek ini disetujui dan sistem-sistem dibangun dan disaring, sistem-sistem baru atau bagian dari sistem diujicoba dan kemudian diperkenalkan kepada organisasi (Kendall, 2010).

**2.2.2. Pengertian Data Flow Diagram (DFD)**

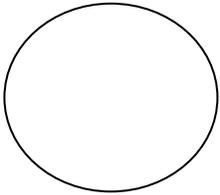


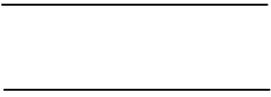
Kristanto (2008:61) menyatakan, “*Data flow diagram / (DFD)* adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.

Sukanto dan Shalahuddin (2014:70) menyatakan, “*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*).”

Sukanto dan Shalahuddin (2014:71), menjelaskan notasi-notasi pada DFD adalah sebagai berikut :

**Tabel 2.1. Notasi-notasi *Data Flow Diagram***

No	Notasi	Keterangan
1		Proses atau fungsi atau prosedur pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnta menjadi fungsi atau prosedur di dalam kode program.  Catatan : nama yang diberikan pada sebuah proses biasanya berupa kata kerja

2		<p>File 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 table-tabel basis data yang dibutuhkan, table-tabel ini juga harus sesuai dengan perancangan table-tabel pada basis data (Entity Relationship Diagram (ERD), Conceptual Data Model (CDM), Physical Data Model (PDM)).</p>
3		<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 : Nama yang digunakan pada masukan (<i>input</i>) dan keluaran (<i>output</i>) biasanya berupa kata benda.</p>

(*Sumber : Sukamto dan Shalahuddin (2014:72)*)

Sukamto dan Shalahuddin (2014:72) menjelaskan tentang 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-sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya 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 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 Level 2.

### 2.2.2. Pengertian *Block Chart*

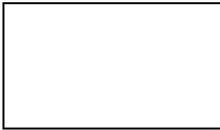
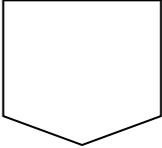
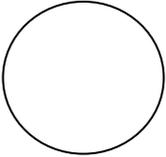
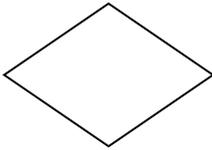
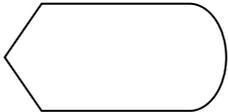
Kristanto (2011:68), “*Block Chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *block chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Kristanto (2011:68) menjelaskan tentang simbol-simbol yang sering digunakan dalam *block chart* adalah sebagai berikut :

**Tabel 2.2. Simbol-simbol *Block Chart***

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



2		Multi dokumen.
3		Proses manual.
4		Proses yang dilakukan oleh komputer.
7		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
8		Terminasi yang mewakili symbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9		Terminasi yang mewakili simbol tertentu untuk digunakan pada giliran lain pada halaman yang sama.
10		Terminasi yang menandakan awal dan akhir dari suatu aliran.
11		Pengambilan keputusan (decision).
12		Layar peraga (monitor).

13		Pemasukan data secara manual
----	---	------------------------------

(Sumber : Kristanto (2011:68))

### 2.2.3. Pengertian ERD

Sukanto dan Shalahuddin (2014:50), “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional”.

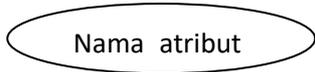
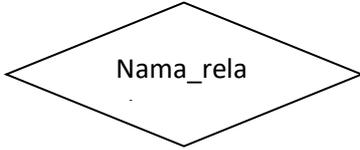
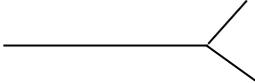
Sukanto dan Shalahuddin (2014:50) menjelaskan simbol-simbol yang digunakan pada ERD dengan notasi Chen :

**Tabel 2.3. Simbol-simbol**

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

No	Simbol	Deskripsi
1	Entitas / <i>entity</i>  	Entitas merupakan data inti yang akan disimpan; bakal table pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.

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

No	Simbol	Deskripsi
2	Atribut  Nama atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3	Atribut kunci primer  Nama kunci prime	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record 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>multivalue</i>  Nama_rela	Relasi yang menghubungkan antar entitas biasanya diawali dengan kata kerja.
5	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas dimana 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 dengan entitas B.

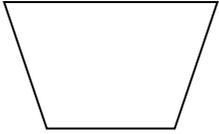
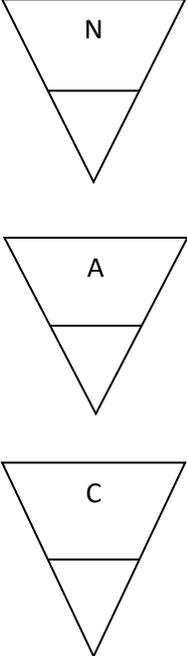
(Sumber : Sukamto dan Shalahuddin (2014:50))

#### 2.2.4. Pengertian Flowchart

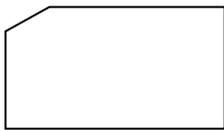
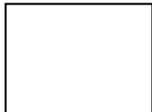
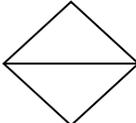
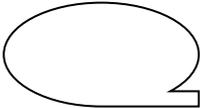
Supardi (2013:51), “*Flowchart* merupakan diagram alur yang sering digunakan sistem analis dalam membuat atau menggambarkan lgika program. Namun, Flowchart juga dapat menggambarkan jalannya sistem”.

Supardi (2013:52) menjelaskan tentang simbol-simbol dalam flowchart adalah sebagai berikut :

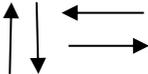
**Tabel 2.4. Simbol-simbol dalam Flow Chart**

No	Simbol	Keterangan
1	Simbol dokumen 	Menunjukkan dokumen input dan output baik untuk proses manual, mekanik atau komputer.
2	Simbol kegiatan manual 	Menunjukkan pekerjaan manual.
3	Simbol simpanan <i>Offline</i> 	<p>file non-komputer yang diarsip urut angka (<i>numerical</i>).</p> <p>File non-komputer yang diarsip urut huruf (<i>alphabetical</i>).</p> <p>File non-komputer yang siarsip urut tanggal (<i>chronological</i>).</p>

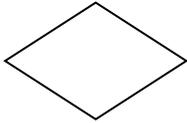
**Lanjutan Tabel 2.4. Simbol-simbol  
dalam Flow Chart**

No	Simbol	Keterangan
4	Simbol kartu plong 	Menunjukkan <i>input/output</i> yang menggunakan kartu plong ( <i>punched card</i> ).
5	Simbol proses 	Menunjukkan kegiatan proses dari operasi program komputer.
6	Simbol operasi luar 	Menunjukkan operasi yang dilakukan diluar proses operasi komputer.
7	Simbol pengurutan <i>offline</i> 	Menunjukkan proses pengurutan data diluar proses komputer.
8	Simbol pita magnetic 	Menunjukkan <i>input / output</i> menggunakan pita magnetik.
9	Simbol disket 	Menunjukkan <i>input / output</i> menggunakan disket.

Lanjutan Tabel 2.4. Simbol-simbol dalam Flow Chart

No	Simbol	Keterangan
10	Simbol drum magnetik 	Menunjukkan <i>input</i> / <i>output</i> menggunakan drum magnetik.
11	simbol pita kertas berlubang 	Menunjukkan <i>input</i> / <i>output</i> menggunakan pita kertas berlubang.
12	Simbol <i>keyboard</i> 	Menunjukkan <i>input</i> / <i>output</i> menggunakan <i>on-line keyboard</i> .
13	Simbol pita control 	Menunjukkan penggunaan pita control ( <i>control tape</i> ) dalam <i>batch control</i> total untuk pencocokan diproses <i>batch processing</i> .
14	Simbol garis alir 	Menunjukkan arus dari proses.
15	Simbol <i>input</i> / <i>output</i> 	Simbol <i>input</i> / <i>output</i> ( <i>input</i> / <i>output symbol</i> ) digunakan untuk mewakili data <i>input</i> / <i>output</i> .

**Lanjutan Tabel 2.4. Simbol-simbol dalam Flow Chart**

No	Simbol	Keterangan
16	Simbol penghubung 	Simbol penghubung ( <i>connector symbol</i> ) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman lainnya.
17	Simbol keputusan 	Simbol keputusan ( <i>decision symbol</i> ) digunakan untuk suatu penyelesaian kondisi didalam program.
18	Simbol proses terdefinisi 	Simbol proses terdefinisi ( <i>predefined process symbol</i> ) digunakan untuk menunjukkan suatu proses yang rinciannya ditunjukkan di tempat lain.
19	Simbol persiapan 	Simbol perisapan ( <i>preparation symbol</i> ) digunakan untuk member nilai awal suatu besaran.
20	Simbol titik terminal 	Simbol titik terminal ( <i>terminal point symbol</i> ) digunakan untuk menunjukkan awal dan akhir dari suatu proses.

(Sumber : Supardi (2013:52))

### 2.2.5. Pengertian Kamus Data

Sukamto dan Shalahuddin (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 dan Shalahuddin (2014:43) menjelaskan kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut :

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

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

(*Sumber* : Sukamto dan Shalahuddin (2014:43))



## **2.3 Teori Program**

### **2.3.1. Pengertian HTML**

Suryana dan koesheryatin (2014:29), “*Hyper Text Markup Language* (HTML) adalah sebuah bahasa yang digunakan untuk menulis halaman web dan merupakan pengembangan dari standar pemformatan dokumen teks, yaitu Standard Generalized Markup Language (SGML). HTML pada dasarnya merupakan dokumen ASCII atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem tertentu.”

### **2.3.2. Pengertian CSS**

Suryana dan koesheryatin (2014:179), “*Cascading Style Sheet* (CSS) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu website, baik tata letaknya, jenis huruf, warna dan semua yang berhubungan dengan tampilan, Pada umumnya CSS digunakan untuk memformat halaman web yang ditulis dengan HTML atau XHTML.”

Sibero (2013:112), “*Cascading Style Sheet* (CSS) memiliki arti Gaya menata halaman bertingkat, yang berarti setiap satu elemen yang telah diformat dan memiliki anak dan telah diformat, maka anak dari elemen tersebut secara otomatis mengikuti format elemen induknya.”

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

Sukamto dan Shalahuddin (2014:43), “Basis Data adalah sistem yang terkomputerisasi yang tujuan utamanya adalah memelihara data yang suda diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.”

### **2.3.4. Pengertian MySQL**

Nugroho (2004:29), “MySQL (My Structure Query Language) atau yang biasa dibaca “mai-se-kuel” adalah sebuah program pembuat database yang bersifat open source, artinya siapa saja boleh menggunakannya.”



MySQL juga merupakan program pengakses database yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak pengguna). Kelebihan dari MySQL adalah ia menggunakan bahasa Query standar yang dimiliki SQL (*Structure Query Language*).

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.

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

Sibero (2013:49), “PHP adalah pemrograman interpreter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti computer secara langsung pada saat baris kode dijalankan. PHP disebut sebagai pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada server.”

Kadir (2008:358) menyatakan, “*PHP* merupakan bahasa pemrograman skrip yang diletakkan dalam server yang biasa digunakan untuk membuat aplikasi *web* yang bersifat dinamis.”

Hidayatullah dan kawistara (2017:223) menyatakan, “*PHP Hypertext Preprocessor* atau disingkat dengan *PHP* ini adalah suatu bahasa scripting khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan *PHP* harus menggunakan *web server*.”

### 2.3.6. Sintaks Dasar PHP

Hidayatullah dan kawistara (2017:231) menyatakan sebelum kita menggunakan *PHP* kita akan mempelajari sintaks dasar *PHP* itu sendiri, ada beberapa aturan sintaks yang harus kita penuhi ketika kita membuat *file* program php.

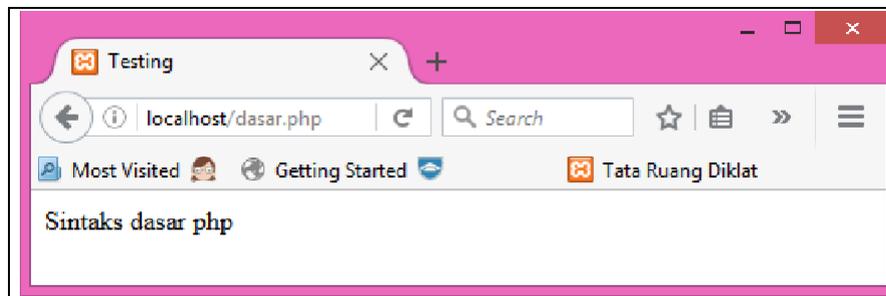
Berikut ini contoh sintaks dasar penggunaan *PHP* :

```
<!doctype html>
<HTML>
```

```

<HEAD>
    <META charset="utf-8">
    <TITLE>Testing</TITLE>
</HEAD>
<?php
Echo "Sintaks dasar php";
?>
<BODY>
<BODY>
</HTML>
  
```

Bila di jalankan di Browser maka sintaks dasar tersebut akan menjadi seperti gambar di bawah ini:



**Gambar 2.1** Tampilan di Browser

### 2.3.7. Pengertian *PhpMyAdmin*

Nugroho (2004:161), "*PhpMyAdmin* adalah suatu program open source yang berbasis web yang dibuat menggunakan aplikasi PHP. Program ini digunakan untuk mengakses database MySQL dan untuk mempermudah dan mempersingkat kerja kita. Dengan kelebihanannya, para pengguna awam tidak harus paham syntax-syntax SQL dalam pembuatan database dan tabel."



Hidayatullah dan kawistara (2017:181) menyatakan, “*phpMyAdmin* adalah *tool open source* yang ditulis dalam bahasa *PHP* untuk menangani administrasi *MySQL*, berbasis *World Wide Web*.”

### **2.3.8. Pengertian Notepad++**

Nugroho (2004:6), “Notepad++ adalah sebuah software bawaan windows sebagai editor dasar.”

### **2.3.9. Pengertian XAMPP**

Wahana Komputer (2014:72) menjelaskan, “XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Xampp adalah tool yang menyediakan paket perangkat lunak dalam satu buah paket. Dalam paket XAMPP sudah terdapat Apache (*web server*), MySQL (*database*), PHP (*server side scripting*), Perl, FTP Server, PhpMyAdmin dan berbagai pustaka bantu lainnya.”

### **2.3.10. Pengertian Java Script**

Suryana dan Koesheryatin (2014:181), “Java Script adalah bahasa script berdasar pada objek yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen HTML. Di mana objek tersebut dapat berupa suatu window, frame, URL, dokumen, form, button, atau item yang lain. Yang semuanya itu mempunyai properti yang saling berhubungan dengannya dan masing-masing memiliki nama, lokasi, warna nilai dan atribut lain.”