# BAB II
# TINJAUAN PUSTAKA

## 2.1. Teori Umum

### 2.1.1. Data

Sutabri (2012:25), “Data merupakan bentuk mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model agar menghasilkan informasi”.

Ladjamudin (2013:8), “Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata.”

Dari kedua definisi diatas, dapat disimpulkan bahwa data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih membutuhkan adanya suatu pengolahan.

### 2.1.2. Aplikasi

Sutabri (2012:147), “Aplikasi adalah alat terapan/bantu yang difungsikan secara khusus dan terpadu sesuai dengan kemampuan yang dimilikinya”.

Sugiar (2014:83), “Aplikasi adalah program yang dibuat untuk melaksanakan tugas tertentu yang dibutuhkan oleh pengguna komputer (user).”

Dari kedua definisi diatas, dapat disimpulkan bahwa aplikasi merupakan alat yang difungsikan secara khusus sesuai dengan kemampuan yang dimilikinya seperti penerapan, penggunaan, dan penambahan data.

### 2.1.3. Sistem

Fatta (2007:3), “Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain”.

### 2.1.4. Karakteristik Sistem

Fatta (2007:5), Karakteristik Sistem terdiri dari:

a. Batasan (*boundary*)

Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.

b. Lingkungan (*environment*)

Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.

c. Masukan (*input*)

Sumber daya (data, bahan baku, perlatan, energy) dari lingkungan yang di konsumsi dan di manipulasi oleh suatu sistem.

d. Keluaran (*output*)

Sumber daya atau produk (informasi, laporan, dokumen, tampilan layer komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.

e. Komponen (*component*)

Kegiatan_kegiatan atau proses dalam suatu sistem yang mentranspormasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.

f. Penghubung (*interface*)

Tempat di mana komponen atau sistem dan lingkungannya bertemu dan berinteraksi.

g. Penyimpanan (*storage*)

Area yang di kuasai dan di gunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

### 2.1.5. Klasifikasi Sistem

Sutabri (2012:15), Sistem dapat diklasifikasikan menjadi beberapa bagian yaitu:

a. Sistem Abstrak dan Sistem Fisik

Sistem abstrak *(abstract system)* adalah sistem sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sitem yang berupa pemikiran tentang hubungan antara manusia dengan tuhan. Sistem Fisik *(physical system)* merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, dan lain sebagainya.

b. Sistem Alamiah dan Sistem Buatan

Sistem alamiah *(natural system)* adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam, dan pergantian musim. Sistem buatan *(human made system)* merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

d. Sistem Tertutup dan Sistem Terbuka

Sistem tertutup *(closed system)* merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sistem terbuka *(open system)* merupakan sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsitem lainnya.

### 2.1.6. Metode Pengembangan Sistem

Sukamto dan Shalahudin (2014:28), menjelaskan tentang metode pengembangan sistem model *waterfall*, sebagai berikut:

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 dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

1. Analisis kebutuhan perangkat lunak

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.

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. Tahap ini mentransaksi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

   Desain harus ditranslasikan ke 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 dan fungsional dan memastikan bahwa bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*)

   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. 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.

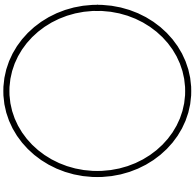
## 2.2. Teori Khusus

### 2.2.1. Data Flow Diagram

Sutabri (2012:117), "DFD adalah suatu *network* yang menggambarkan suatu sistem komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya di susun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya".

Jadi, dapat disimpulkan bahwa DFD (*Data Flow Diagram*) merupakan gambaran dari suatu sistem yang di susun dalam bentuk kumpulan komponen yang saling berhubungan dan digunakan untuk merancang maupun untuk merencanakan sistem yang baru.

**Tabel 2.1 Simbol DFD menurut teknik Yurdon/De Marco**

| Gambar | Keterangan |
|---|---|
| | Entity: digambarkan dengan symbol bujursangkar dengan bayangan kedua sisinya. |
| | Aliran data: berfungsi untuk menggambarkan aliran data dari satu proses ke proses lainnya. |
|  | Proses atau fungsi yang mentransformasikan data secara umum digambarkan dengan segi empat tumpul. |
| | Berkas atau tempat penyimpanan : untuk menyimpan data atau file. |

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

   DFD Level 0 menggambarkan system yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun system lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara system 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 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 system, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di- *breakdown*.

4. Membuat DFD Level 3 dan seterusnya

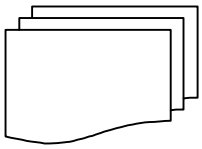
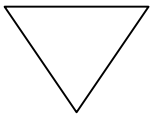
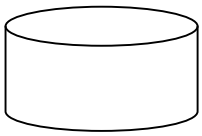
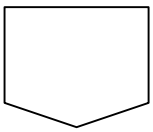
   DFD Level 3, 4, 5 dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2. Pada satu diagram DFD sebaiknya jumlah modul tidak boleh lebih dari 20 buah.

   Jika lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk dibaca sehingga menyebabkan system yang dikembangkan juga menjadi rumit.

### 2.2.2. Blockchart

Kristanto (2008:75), *"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.

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

| No | Simbol | Arti |
|---|---|---|
| 1 | | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan. |
| 2 |  | Multi dokumen. |
| 3 | | Proses manual. |
| 4 | | Proses yang dilakukan oleh komputer. |
| 5 |  | Menandakan dokumen yang diarsipkan (arsip manual). |
| 6 |  | Data penyimpanan (data storage). |
| 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. |

*Sumber: Buku Karangan Kristanto Tahun 2008 Halaman 75*

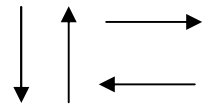
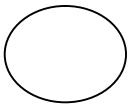
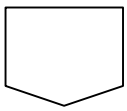
### 2.2.3. Flowchart

Agus Saputra (2014:14), “Bagan Alir (*Flowchart*) merupakan suatu diagram yang menggambarkan alur kerja dari suatu sistem”.

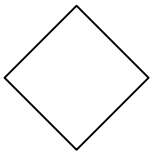
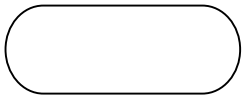
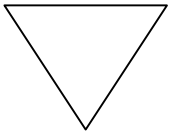
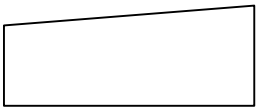
eWolf (2011:16), “Flowchart adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap symbol flowchart melambangkan pekerjaan dan instruksinya”.

Jadi, dapat disimpulkan bahwa *Flowchart* merupakan diagram yang tersusun dari simbol-simbol dan sebagai alat untuk menjelaskan sistem informasi yang ada.

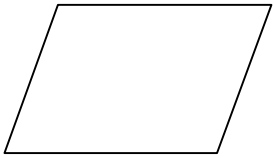
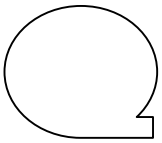
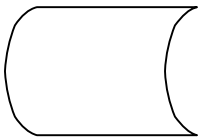
**Tabel 2.3 Simbol Bagan Alir**

| No | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 1 |  | **Garis Alir** (***Flow Line***) | Menyatakan jalannya arus suatu proses. |
| 2 | | ***Communication Link*** | Digunakan untuk memberikan nilai awal pada suatu variabel atau counter. |
| 3 |  | ***Connector*** | Digunakan untuk menunjukkan hubungan arus proses yang terputus masih dalam halaman yang sama. |
| 4 |  | ***Offline Connector*** | Digunakan untuk menunjuk-kan hubungan arus proses yang terputus masih dalam halaman yang sama. |

**Tabel 2.4 Simbol-simbol *Processing Symbols***

| No | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 1 | | ***Process*** | Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer. |
| 2 | | ***Manual*** | Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer. |
| 3 |  | ***Decision*** | Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya/tidak. |
| 4 | | ***Predefined Process*** | Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal. |
| 5 |  | ***Terminal*** | Menyatakan permulaan atau akhir suatu program. |
| 6 | | ***Keying Operation*** | Menyatakan segal jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard. |
| 7 |  | **Offline Storage** | Menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu. |
| 8 |  | ***Manual Input*** | Memasukkan data secara manual dengan menggunakan online keyboard. |

**Tabel 2.5 Simbol-simbol *Input / Output Symbols***

| No | Simbol | Nama Simbol | Keterangan |
|---|---|---|---|
| 1 |  | ***Input/Output*** | Menyatakan proses input atau output tanpa tergantung jenis peralatannya. |
| 2 | | ***Punched Card*** | Menyatakan input berasal dari kartu atau output ditulis ke kartu. |
| 3 |  | ***Magnetic Tape*** | Menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis. |
| 4 |  | ***Disk Storage*** | Menyatakan input berasal dari dari disk atau output disimpan ke disk. |
| 5 | | ***Document*** | Digunakan untuk mencetak keluaran dalam bentuk dokumen (melalui printer). |
| 6 | | ***Display*** | Digunakan untuk mencetak keluaran dalam layar monitor. |

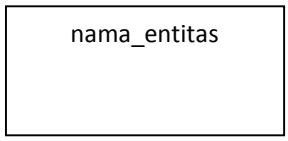
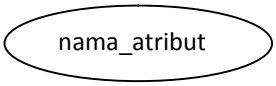
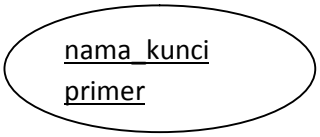
*Sumber: Buku Karangan eWolf Community Tahun 2011 Halaman 17*

### 2.2.4. Entity Relationship Diagram (ERD)

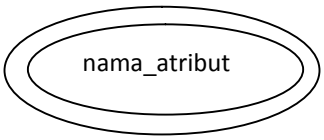
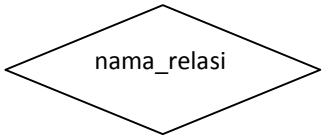
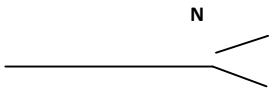
Sukamto dan Shalahuddin (2014:50) menjelaskan, “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen. “

Berikut adalah symbol-simbol yang digunakan pada ERD dengan notasi Chen:

**Tabel 2.6 Simbol-simbol ERD dengan notasi Chen**

| **Simbol** | **Deskripsi** |
|---|---|
| Entitas / *entity*<br>  | Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel. |
| Atribut<br>  | Field atau kolom data yang butuh disimpan dalam suatu entitas |
| Atribut kunci primer<br>  | Field 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 tanpa ada yang sama). |

Lanjutan Tabel 2.6 Simbol-simbol ERD dengan notasi Chen

| Simbol | Deskripsi |
|---|---|
| Atribut multinilai   | Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu. |
| Relasi   | Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja. |
| Asosiasi / association   | Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. |

ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau N-*ary* (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak mengizinkan hubungan *ternary* atau N-*ary*.

### 2.2.5. Kamus data

Sukamto dan Shalahuddin (2014:73) menjelaskan, “Kamus data *(data dictionary)* dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada system perangkat lunak sehingga masukan *(input)* dan keluaran *(output)* dapat dipahami secara umum (memiliki standar cara penulisan).”

Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

a. Nama – nama dari data
b. Digunakan pada – merupakan proses-proses yang terkait data
c. Deskripsi – merupakan deskripsi data
d. Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

Kamus data memiliki beberapa symbol untuk menjelaskan informasi tambahan sebagai berikut:

**Tabel 2.7 Simbol-simbol dalam Kamus Data**

| **Simbol** | **Keterangan** |
|---|---|
| = | disusun atau terdiri dari |
| + | Dan |
| [\|] | baik ...atau... |
| $\{\}^n$ | n kali diulang/ bernilai banyak |
| ( ) | data opsional |
| *...* | batas komentar |

Kamus data pada DFD nanti harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai, sehingga harus ada yang diperbaiki baik perancangan basis datanya, perancangan DFD-nya, atau keduanya.

## 2.3. Pengertian Judul

### 2.3.1. Data

Sutabri (2012:25), “Data merupakan bentuk mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model agar menghasilkan informasi”.

Ladjamudin (2013:8), “Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata.”

Dari kedua definisi diatas, dapat disimpulkan bahwa data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih membutuhkan adanya suatu pengolahan.

### 2.3.2. Aplikasi

Sutabri (2012:147), “Aplikasi adalah alat terapan/bantu yang difungsikan secara khusus dan terpadu sesuai dengan kemampuan yang dimilikinya”.

Sugiar (2014:83), “Aplikasi adalah program yang dibuat untuk melaksanakan tugas tertentu yang dibutuhkan oleh pengguna computer (user).”

Dari kedua definisi diatas, dapat disimpulkan bahwa aplikasi merupakan alat yang difungsikan secara khusus sesuai dengan kemampuan yang dimilikinya seperti penerapan, penggunaan, dan penambahan data.

### 2.3.3. Pengertian Pengolahan

Kristanto (2008:8), “Pengolahan berasal dari kata olah yang berarti mengerjakan, mengusahakan supaya menjadi barang lain atau menjadi lebih sempurna. Pengolahan berarti proses, cara, perbuatan mengolah.”

### 2.3.4. Pengertian Pengolahan Data

Ladjamuddin (2013:9), “Pengolahan data adalah massa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”

Ashkaf (2014), “Pengolahan data adalah pengubahan atau transformasi simbol-simbol seperti nomor dan huruf untuk tujuan peningkatan kegunaannya.”

Dari kedua definisi diatas, dapat disimpulkan bahwa pengolahan data adalah manipulasi data kedalam bentuk yang lebih berarti berupa informasi.

### 2.3.5. Pengertian Kinerja

Jelantik (2015:137) menjelaskan, “Kinerja atau *performance* adalah usaha yang dilakukan dari hasil kerja yang dapat dicapai seseorang atau sekelompok orang dalam suatu organisasi sesuai dengan tugas dan tanggung jawab.”

Nurlaila (2010:71) menjelaskan, “*Perfomance* atau kinerja merupakan hasil atau keluaran dari suatu proses.”

Dari kedua definisi diatas, kinerja merupakan hasil usaha yang dilakukan dari hasil kerja yang merupakan hasil atau keluaran dari suatu proses.

### 2.3.6. Pengertian Penilaian Kinerja

Widodo (2015:130) menjelaskan, “Penilaian kinerja (*performance appraisal*) adalah suatu evaluasi terhadap tingkat kinerja seseorang dibandingkan dengan Standar kinerja yang sudah ditentukan, guna bahan pertimbangan dalam menentukan promosi, kompensasi, perlunya pelatihan atau pengembangan, maupun untuk pemberhentian seseorang.”

### 2.3.7. Pengertian Karyawan/Pegawai

Prastowo dkk (2011:132) menjelaskan, “Pegawai adalah pribadi yang bekerja pada pemberi kerja baik sebagai pegawai tetap atau peagawai tidak tetap atau tenaga kerja lepas berdasarkan perjanjian atau kesepakatan baik secara tertulis maupun tidak tertulis.”

Muljono (2010:113) menjelaskan, “Pegawai adalah setiap orang pribadi yang melakukan pekerjaan berdasarkan perjanjian atau kesepakatan kerja, baik tertulis maupun tidak tertulis, termasuk yang melakukan pekerjaan dalam jabatan negeri atau badan usaha milik negara atau badan usaha milik daerah.”

Dari kedua definisi diatas, pegawai adalah setiap orang pribadi yang melakukan pekerjaan bedasarkan perjanjian di dalam jabatan negeri atau badan usaha milik negara .

## 2.4. Teori Program

### 2.4.1. Pengertian Basis Data (Database)

Indrajani (2015:70), “Basis data adalah sekumpulan elemen data terintegegrasi yang secara logika saling berhubungan.”

Sukamto dan Shalahudin (2014:43), “Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah 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.”

Sutanta (2011:35), “Basis data adalah sekumpulan dari berbagai macam-macam tipe *record* yang memiliki hubungan antar-*record* dan rincian data terhadap obyek tertentu.”

Jadi berdasarkan pengertian diatas, dapat disimpulkan bahwa basis data adalah kumpulan informasi yang disimpan didalam computer secaara sistematik untuk memperoleh informasi dari basis data tersebut.

### 2.4.2. Pengertian XAMPP

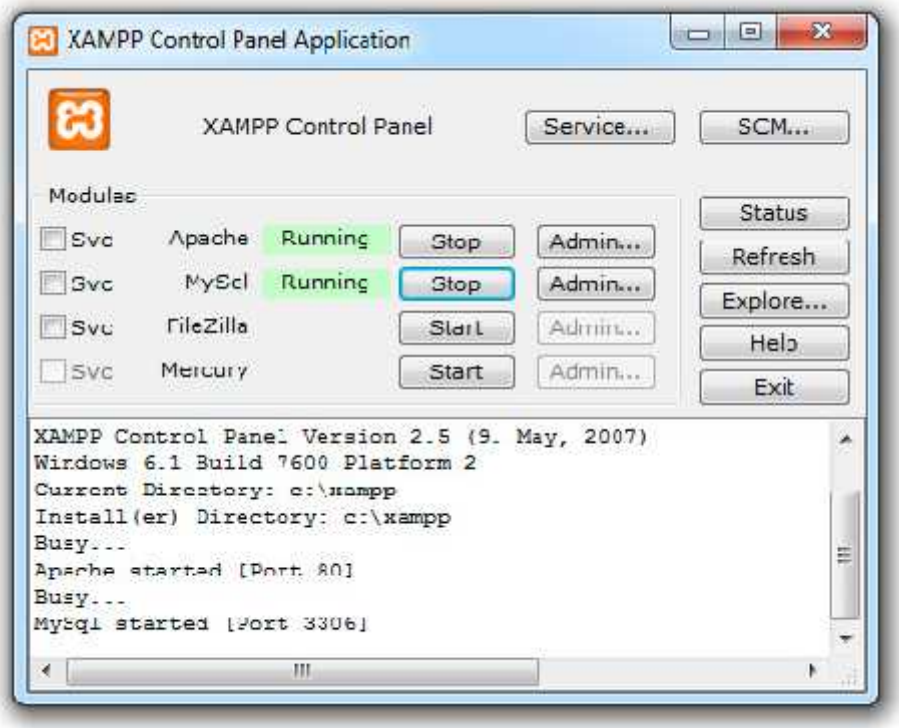
Mulhim (2013:4), “XAMPP adalah paket instalasi program yang terdiri atas program Apache HTTP Server, MySQL database, dan Perl”.

Dibawah folder xampp, terdapat beberapa folder penting yang perlu diketahui. Penjelasan fungsinya sebagai berikut:

1. ***apache*** : Folder utama dari Apache Webserver
2. ***htdocs*** : Folder utama untuk menyimpan data-data latihan *web*, baik *PHP* maupun *HTML* biasa. Pada folder ini, anda dapat membuat subfolder sendiri untuk mengelompokkan file latihannya. Semua folder dan file program di htdocs bisa diakses dengan mengetikan alamat http://localhost/ di browser.
3. **manual** : Berisi subfolder yang di dalam terdapat manual program dan *database*, termasuk manual *PHP* dan *MYSQL*.
4. **mysql** : Folder utama untuk *database MYSQL* server. Didalamnya

terdapat subfolder data (lengkapnya: ***C:\xampp\mysql\data***) untuk merekam semua nama *database*, serta subfolder bin yang berisi *tools* klien dan server *MYSQL*.

5. **php** : Folder utama untuk program *PHP*.




### 2.4.3. Pengertian HTML

Jubile Enterprise (2015:135), "*Hypertext Markup Language* (HTML) adalah *script* pemrograman yang mengatur bagaimana menyajikan informasi di dunia internet dan bagaimana informasi itu membawa kita dari satu tempat ke tempat lainnya."

Winarno (2014:1), "*Hypertext Markup Language* (HTML) adalah sebuah bahasa untuk menampilkan konten di web."

Jadi berdasarkan pengertian diatas, dapat disimpulkan bahwa *Hypertext Markup Language* (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web.

### 2.4.4. Pengertian CSS

Nugroho (2013:155), "CSS *(Cascading Style Sheets)* merupakan sintaks yang berguna sebagai kode pemanis web, juga bisa jadi pengganti HTML.

### 2.4.5. Pengertian Java Script

Betha dan Husni (2009:267) menjelaskan, "JavaScript merupakan modifikasi dari bahasa c++ dengan pola penulisan yang lebih sederhana. Interpreter bahasa ini sudah disediakan asp ataupun internet explorer." Secara khusus beberapa hal yang penting dalam JavaScript adalah:

a. Menggunakan blok awal "{" dan blok akhir "}"
b. Automatic conversion dalam pengoperasian tipe data yang berbeda
c. Sensitive case, sehingga programmer java harus ekstra hati-hati dalam menggunakan nama variable, fungsi dan lain-lain
d. Extension umumnya menggunakan "*.js"
e. Setiap statement dapat diakhiri dengan ";" sebagaimana halnya c++ tetapi dapat juga tidak
f. Jika tidak didukung oleh browser versi lama, scriptnya dapat disembunyikan di antara tag "<!- -" dan "- ->"
g. Jika program dalam satu baris terlalu panjang dapat disambung ke baris berikut dengan karakter

Betha dan Husni (2009 : 272) menjelaskan, "Tipe data pada JavaScript umumnya sama pada setiap mesin, hal ini diakibatkan desain awal java memang dikhususkan untuk bisa berjalan pada semua mesin."

**Tabel 2.8. Tipe data JavaScript**

| Primitive Type | Size | Minimum | Maximum | Version Type |
|---|---|---|---|---|
| Boolean | 1 bit | - | - | |
| Char | 16 bit | Unicode 0 | Unicode 2E16-1 | |

| | | | | |
|---|---|---|---|---|
| Byte | 8 bit | -128 | +127 | Versi 1.1 keatas |
| Short | 16 bit | -2E-15 | +2E15-1 | Versi1.1 keatas |
| Int | 32 bit | -2E-31 | +2E31-1 | |
| Long | 64 bit | -2E-63 | +2E63-1 | |
| Float | 32 bit | IEEE754 | IEEE754 | |
| Double | 64 bit | IEEE754 | IEEE754 | |
| Void | - | - | - | Versi 1.1 keatas |

### 2.4.6. Pengertian PHP

Nugroho (2013:153) menjelaskan, "PHP kepanjangan dari *Hypertext Preprocessor)* itu bahasa pemrograman berbasis web. PHP itu adalah bahasa program yang digunakan untuk membuat aplikasi berbasis web (website, blog, atau aplikasi web)."

Kadir (2013:120) menjelaskan, "PHP merupakan bahasa pemrograman yang ditujukan untuk membuat aplikasi web. Ditinjau dari pemrosesannya, PHP tergolong berbasis *server side.* Artinya, pemrosesan dilakukan di *server."*

Sidik (2014:4) menjelaskan, "PHP merupakan secara umum dikenal sebagai bahasa pemrograman script script yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML."

### 2.4.7. Pengertian MySQL

Winarno (2014:101), "*MySQL* adalah sebuah *software database*. *MySQL* merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan."

Kadir (2013:15), "*MySQL* adalah nama *database server*. *Database server* adalah *server* yang berfungsi untuk menangani *database*. *Database* adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data. Dengan menggunakan *MySQL*, kita bisa menyimpan data dan kemudian data bisa diakses dengan cara yang mudah dan cepat."

Saputra dkk (2012:7), "*MySQL* merupakan *database* yang bekerja menggunakan SQL *Language* (*Structure Query Language*)."

Jadi berdasarkan pengertian diatas, dapat disimpulkan bahwa *MySQL* adalah sistem manajemen database SQL yang bersifat *open source* dan paling populer saat ini.