



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

Teori ini membahas tentang pengertian-pengertian yang berkaitan dengan judul Laporan Akhir.

2.1.1. Pengertian Komputer

Febri (2012:3), mengemukakan "Komputer adalah peralatan yang menerima data dan menyimpan kemudian diproses untuk menghasilkan data dalam bentuk lain."

Sujatmiko (2012:156), mengemukakan "Komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi komunikasi."

Asropudin (2013:19), mengemukakan "Komputer adalah alat bantu proses data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer."

Dari pengertian diatas, komputer adalah alat pemrosesan data elektronik yang membantu melakukan perintah-perintah atau proses-proses tertentu yang telah diurutkan atau diprogramkan.

2.1.2. Pengertian Perangkat Lunak

Rosa *et.al* (2014:2), mengemukakan "Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)."

Sujatmiko (2012:210), mengemukakan "Perangkat Lunak (*Software*) adalah istilah umum untuk data yang diformat dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer. Dengan kata lain, bagian sistem komputer yang tidak berwujud."



Dari pengertian diatas, Perangkat Lunak adalah program komputer yang terasosiasi untuk menyimpan data yang disimpan secara digital, termasuk program komputer.

2.1.3. Pengertian Data

Kristanto (2011:7), mengemukakan “Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.”

Al-bahra (2013:20), mengemukakan ”Data adalah komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.”

Febri (2012:3), mengemukakan ”Data adalah suatu *file* atau pun *field* yang berupa karakter atau tulisan dan gambar.”

Dari pengertian diatas, data adalah kumpulan dari angka maupun karakter yang berasal dari kenyataan yang belum berarti dan harus diolah terlebih dahulu untuk menjadi suatu informasi yang lebih berarti bagi penggunanya.

2.1.4. Pengertian Pengolahan Data

Kristanto (2011:8), mengemukakan “Pengolahan Data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data informasi yang memiliki kegunaan.”

Al-Bahra (2013:9), mengemukakan “Pengolahan Data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”

Sutarman (2012:4), mengemukakan “Pengolahan Data adalah proses perhitungan atau transformasi data *input* menjadi informasi yang mudah di mengerti ataupun sesuai dengan yang diinginkan.”

Dari pengertian diatas, pengolahan data merupakan proses pengolahan data yang *diinputkan* untuk diubah menjadi informasi yang berguna.



2.1.5. Pengertian Sistem

Hartono (2013:9), mengemukakan “Sistem adalah suatu himpunan dari berbagai bagian atau elemen, yang saling berhubungan secara terorganisasi berdasar fungsi-fungsinya, menjadi suatu kesatuan.”

Kristanto (2011:1), mengemukakan “Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.”

Dari pengertian diatas, sistem merupakan sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

2.1.6. Karakteristik Sistem

Al-Bahra (2013:3-5), menjelaskan bahwa sistem memiliki karakteristik atau sifat-sifat tertentu yang mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan ,keluaran, pengolahan dan sasaran atau tujuan. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan.

2. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan dan menunjukan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dan bersifat menguntungkan dan juga merugikan.



4. Penghubung Sistem (*Interface*)

Media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya.

5. Masukan Sistem (*Input*)

Masukan adalah Energi yang dimasukkan ke dalam sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi sub sistem yang lain.

7. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki sasaran dan tujuan yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.7. Metode Pengembangan Sistem

Rosa *et.al* (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 pemeliharaan.

a. Analisis

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

**b. Desain**

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengkodean

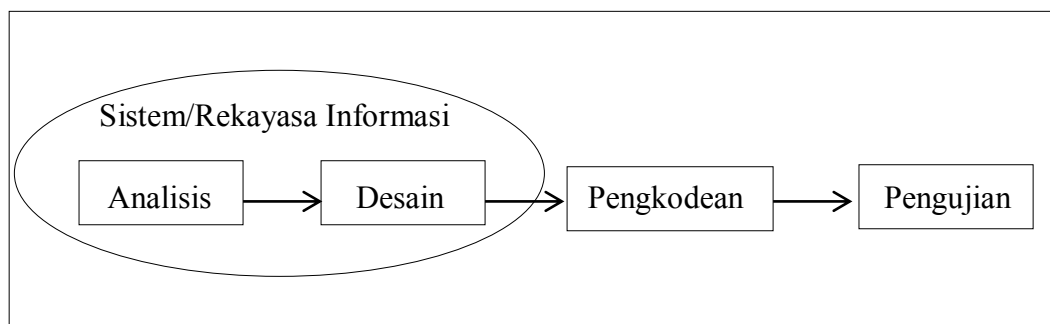
Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika 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. Pemeliharaan

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.



Gambar 2.1. Ilustrasi Model *Waterfall*



2.2. Pengertian Judul

2.2.1. Pengertian Informasi

Sutabri (2012:22), mengemukakan “Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.”

Sutarman (2012:4), mengemukakan “Informasi adalah hasil kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian.”

Kristanto (2011:6), mengemukakan “Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima”.

Dari pengertian diatas, Informasi adalah keluaran yang dihasilkan dari kegiatan pengolahan data menjadi lebih berguna dan lebih berarti bagi penerima.

2.2.2. Pengertian Layanan

Kamus Besar Bahasa Indonesia (368), “Layanan adalah cara membantu yang dibutuhkan pihak lain”

2.2.3. Pengertian Perizinan

Kamus Besar Bahasa Indonesia, “ Perizinan adalah hal pemberian izin”.
(<http://kbbi.web.id/izin>)

2.2.4. Pengertian Badan Penanaman Modal dan Pelayanan Terpadu Satu Pintu

Badan Penanaman Modal dan Pelayanan Terpadu Satu Pintu adalah badan yang mendorong pertumbuhan ekonomi melalui peningkatan investasi, dengan memberikan perhatian lebih besar pada peran usaha mikro, kecil dan menengah.



2.2.5. Pengertian Sistem Informasi Layanan Perizinan pada Badan Penanaman Modal dan Pelayanan Terpadu Satu Pintu Prabumulih

Dari uraian di atas dapat disimpulkan bahwa pengertian Sistem Informasi Layanan Perizinan pada Badan Penanaman Modal dan Pelayanan Terpadu Satu Pintu Prabumulih merupakan suatu sistem informasi yang berfungsi untuk mengolah data pemohon yang membuat surat izin, melakukan perpanjangan dan daftar ulang sehingga dapat mempermudah dalam pembuatan laporan surat izin pada Badan Penanaman Modal Dan Pelayanan Terpadu Satu Pintu.

2.3. Teori Khusus

Teori ini akan membahas mengenai desain sistem yang merupakan suatu fase yang memerlukan suatu keahlian perencanaan untuk elemen-elemen komputer yang akan menggunakan sistem baru.

2.3.1. Pengertian DFD (*Data Flow Diagram*)

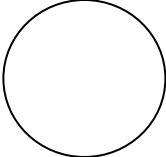
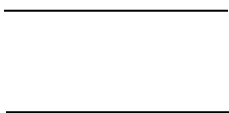
Kristanto (2011:55), mengemukakan “*Data Flow Diagram (DFD)* adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.”

Sutabri (2012:117), mengemukakan “*Data Flow Diagram* adalah suatu *network* yang menggambarkan suatu sistem automat atau komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya.”

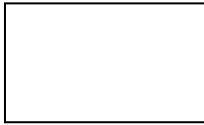
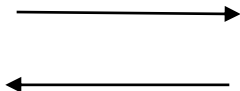
Al-Bahra (2013:64), mengemukakan “*Data Flow Diagram* adalah model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil.”

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

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

No	Simbol	Keterangan
1		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 catatan: nama yang diberikan pada sebuah proses biasanya berupa kata kerja
2		<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>)

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

No	Simbol	Keterangan
3.		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, catatan :nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) berupa kata benda.
4.		Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>), catatan :nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”

(Sumber : Rosa et.al, 2014:71)

Menurut Rosa et.al (2014:72), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram* :

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

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. *DFD* Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat *DFD* Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. *DFD* Level 1 merupakan hasil *breakdown* *DFD* Level 0 yang sebelumnya sudah dibuat.



3. Membuat *DFD* Level 2

Modul-modul pada *DFD* Level 1 dapat di-*breakdown* menjadi *DFD* Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-*breakdown* lagi. Untuk sebuah sistem, jumlah *DFD* Level 2 sama dengan jumlah modul pada *DFD* Level 1 yang di-*breakdown*.

4. Membuat *DFD* Level 3 dan seterusnya

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

2.3.2. Pengertian Blockchart

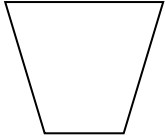
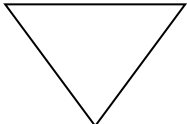
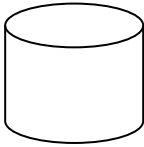
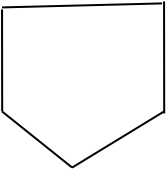
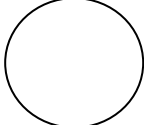
Kristanto (2011:68), mengemukakan “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

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

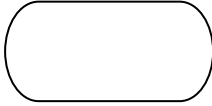
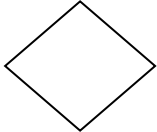
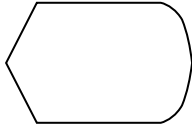
Tabel 2.2. Simbol-simbol dalam *Blockchart*

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

Lanjutan Tabel 2.2. Simbol-simbol dalam *Blockchart*

No	Simbol	Keterangan
3.		Proses Manual
4.		Proses dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>Storage</i>)
7.		Proses apa saja yang tidak <i>terDefinisi</i> termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.

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

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

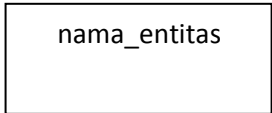
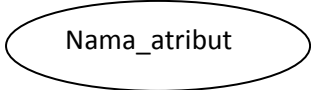
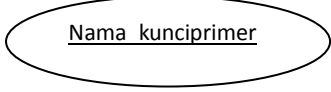
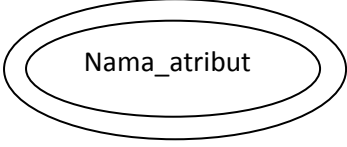
(Sumber : Kristanto, 2011:68)

2.3.3. Pengertian ERD (*Entity Relational Diagram*)

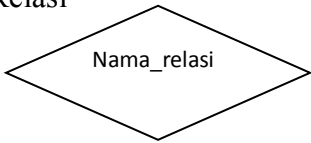
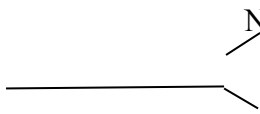
Rosa *et.al* (2014:53), Mengemukakan “*Entity Relationship Diagram (ERD)* merupakan bentuk paling awal dalam melakukan perncangan basis data relasional.”

Adapun simbol-simbol yang digunakan dalam *ERD*, yaitu:

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

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya 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 multinilai/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 Relationship Diagram (ERD)*

No	Simbol	Keterangan
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

(Sumber : Rosa et.al, 2014:53)

2.3.4. Pengertian Flowchart

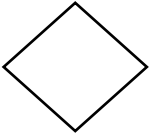
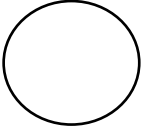
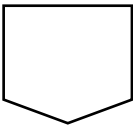
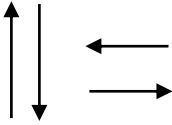
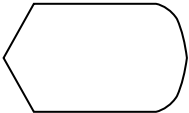
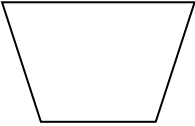
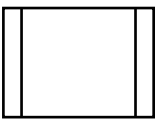
Ewolf Community (2011:16), Mengemukakan “*Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *Flowchart* melambangkan pekerjaan dan instruksinya”.

Adapun menjelaskan tentang simbol-simbol yang sering digunakan dalam *Flowchart* dapat dilihat pada tabel berikut ini:

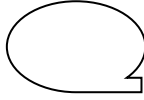
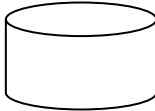
Tabel 2.4. Simbol-simbol dalam *Flowchart*

No	Simbol	Arti
1.		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses

**Lanjutan Tabel 2.4.** Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu.
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar <i>symbol</i>
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll.
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual.
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen.
11.		Simbol yang menyatakan bagian dari program (subprogram)

**Lanjutan Tabel 2.4.** Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
12.		Simbol masukan atau keluaran dari atau ke sebuah pita <i>magnetic</i>
13.		Simbol <i>database</i> atau basis data

(Sumber : Ewolf Community, 2011:16)

2.3.5. Pengertian Kamus Data

Al-Bahra (2013 : 70), mengemukakan ”Kamus data adalah katalog fakta tentang data dan kebutuhan.

Kristanto (2011:66), mengemukakan “Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem.”

Rosa *et.al* (2014:73-74), mengemukakan “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum.

Adapun simbol-simbol yang di gunakan dalam kamus data, yaitu :

Tabel 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	Disusun atau terdiri atas
2	+	Dan
3	[]	Memilih salah satu alternative
4	{ ⁿ }	N kali diulang / bernilai banyak
5	()	Data operasional
6	* ... *	Batas komentar



2.3.6. Pengertian Daftar Kejadian (*Event List*)

Kristanto (2011:64), menjelaskan bahwa daftar kejadian (*event list*) digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Adapun cara mendeskripsikan daftar kejadian adalah sebagai berikut :

1. Pelaku adalah *entity* luar, jadi bukan sistem.
2. Menguji setiap *entity* luar dan mencoba mengevaluasi setiap *entity* luar yang terjadi pada sistem.
3. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
4. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan *entity* luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respon dari kejadian.
6. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
7. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

2.4. Teori Program

Teori program adalah teori yang menjelaskan bahasa pemrograman yang di aplikasikan penulis dalam pembuatan Laporan Akhir.

2.4.1. Pengertian Basis Data (*Database*)

Kadir (2013: 15), mengemukakan “*Database* adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data.”



Yakub (2012:51), mengemukakan “Basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi). Relasi biasanya ditunjukkan dengan kunci (*key*) dari tiap *file* yang ada.”

Dari pengertian diatas, Basis data adalah kumpulan data yang terorganisasi dan saling berhubungan untuk memudahkan dalam penyimpanan dan pengaksesan data.

2.4.2. Pengertian *HTML*

Asporudin (2013:44), mengemukakan “*Hyper Text Markup Language (HTML)* adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*.”

Anhar (2010:40), mengemukakan “*HTML* adalah sekumpulan simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah *file* yang digunakan untuk menampilkan halaman *web* pada *web browser*.”

Dari pengertian diatas, *HTML* adalah bahasa komputer berupa simbol atau *tag* yang digunakan untuk membuat sebuah *web*.

2.4.3. Pengertian *CSS*

Madcoms (2010:141), mengemukakan “*CSS* adalah suatu kumpulan kode-kode untuk memformat, yang mengendalikan tampilan isi dalam suatu halaman *web*”.

2.4.4. Pengertian *PHP*

Anhar (2010:3), mengemukakan “*PHP* adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru. Semua *script PHP* dieksekusi pada server dimana *script* tersebut dijalankan.”

Winarno *et.al* (2014:49), mengemukakan “*PHP* adalah sebuah bahasa pemrograman *web* berbasis server (*server-side*) yang mampu mempasing kode



PHP dari kode *web* dengan ekstensi *.php*, sehingga menghasilkan tampilan *website* yang dinamis di sisi *client (browser)*.”

Hakim (2014:2), mengemukakan “ *PHP* adalah bahasa pemrograman yang digunakan untuk membuat aplikasi *web*.”

Dari pengertian diatas, *PHP* adalah bahasa pemrograman atau *script* yang digunakan untuk membuat sebuah halaman *web* dengan ekstensi *.php*.

2.4.5. Script PHP

Madcoms (2010:350), mengemukakan “*Script PHP* tidak harus berdiri sendiri namun dapat disisipkan di antara kode *HTML*. *Script PHP* harus diawali dengan `<? Atau <?php` dan diakhiri dengan `?>`”.

Hakim (2014:8), berikut adalah cara menyisipkan *script PHP* ke dalam *HTML* ada 2 cara, yaitu:

- **Embedded Script**

```
<p align =”center”>
<font size=”6” color=”blue” face=”arial”>
<?php
Echo date(“d-m-y”);
?>
</font>
</p>
```

- **Non-Embedded Script**

```
<?php
Echo “<p align =\”center\”>”;
Echo “<font size=\”6\” color=\”blue\” face=\”arial\”>”;
Echo date(“d-M-Y”);
Echo “</font>”;
Echo “</p>”; ?>
```



2.4.6. Pengertian *MySQL*

Kadir (2013 :15), mengemukakan ”*MySQL* adalah nama *databases* server. *Database* server adalah server yang berfungsi untuk menangani *database*, dengan menggunakan *MySQL* kita bisa menyimpan data dan kemudian data bisa di akses dengan cara mudah dan cepat.”

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

Dari pengertian diatas, *MySQL* adalah sebuah *database* yang dapat menyimpan data dalam bentuk tabel serta dapat diakses dengan mudah dan cepat.

2.4.7. Pengertian *Adobe Dreamweaver 8*

Madcoms (2010: 1), mengemukakan “*Dreamweaver* adalah sebuah *HTML* editor profesional untuk mendesain *web* secara visual dan mengelola situs atau halaman *web*”.

Macromedia Dreamweaver adalah program untuk membuat dan mengedit dokumen *HTML* secara visual dan mengelola halaman sebuah situs. *Dreamweaver* menyediakan banyak perangkat yang berkaitan dengan pengkodean dan fitur seperti *HTML*, *CSS*, *JavaScript*, *PHP*, *ASP*, *ColdFusion*, dan *XML*.



Gambar 2.2. Tampilan Logo *Macromedia Dreamweaver 8*



2.4.7.1. Halaman Awal Macromedia Dreamweaver 8



Gambar 2.3. Tampilan Awal Macromedia Dreamweaver 8

Pada halaman awal *Dreamweaver 8* terdapat beberapa menu yang dapat dipilih :

1) *Open a Recent Item*

Pada menu ini akan ditampilkan beberapa *file* yang sebelumnya pernah kita buka dengan menggunakan *Dreamweaver 8*. Atau di paling bawah ada *Open* yang dapat digunakan untuk membuka *file* yang lain.

2) *Create New*

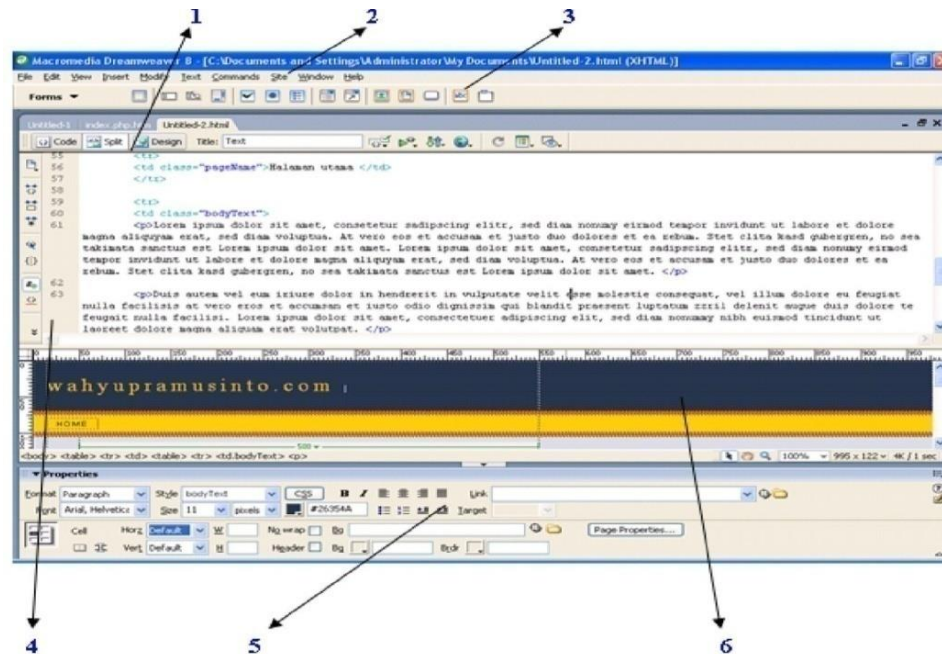
Pada menu ini kita dapat memilih dokumen baru apa yang akan kita buat dengan menggunakan *Dreamweaver 8*. Ada banyak pilihan, diantaranya *HTML*, *ColdFusion*, *PHP*, *ASP*, *JavaScript*, *CSS*.

3) *Create From Samples*

Pada menu ini kita dapat membuat *file* berdasarkan contoh yang sudah diberikan oleh *Dreamweaver*.



2.4.7.2. Halaman Utama Macromedia Dreamweaver 8



Gambar 2.4. Tampilan Utama Macromedia Dreamweaver 8

Keterangan:

1. Toolbar Dokumen

Toolbar dokumen digunakan untuk mengubah tampilan dan mengakses fungsi-fungsi penting secara cepat dan mudah. Pada toolbar dokumen terdapat menu untuk berpindah antar dokumen kerja *window* dan mengatur tampilan area kerja. Untuk mengatur tampilan kita bisa memilih *Code*, *Split* dan *Design*.

2. Menu Utama

Menu Utama berisi semua perintah yang dapat digunakan untuk bekerja pada *Dreamweaver*.

3. Insert Bar



Gambar 2.5. Tampilan Insert Bar Macromedia Dreamweaver 8

Insert bar merupakan tempat semua perangkat kerja (tombol) yang digunakan untuk membuat halaman *web*. *Insert bar* mempunyai dua jenis tampilan, yaitu



tampilan sebagai menu dan tampilan sebagai *tab*.

a) **Tab Common**

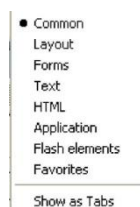
Berisi semua tombol yang sering atau umum digunakan untuk membuat halaman *web*. Tombol yang ada di *tab common* antara lain *Hyperlink*, *Email Link*, *Named Anchor*, *Table*, *Images*.

b) **Tab Layout**

Tab layout digunakan untuk membuat *layout* halaman *web*. Terdapat tiga jenis *layout* yang dapat dipilih, yaitu *Standard*, *Expanded* dan *Layout*.

- 1) Untuk *Standard view* tampilan dokumen seperti biasa (berupa garis-garis tebal)
- 2) Untuk *Expanded view* menampilkan border tabel yang direnggangkan sehingga semua rancangan tabel dapat dilihat dengan jelas baik itu baris dan kolomnya.
- 3) Untuk *Layout view* rancangan tabel ditampilkan sebagai kotak-kotak yang dapat di-*drag*, dan di atur ulang ukurannya dengan mudah.

Tampilan *Insert bar* sebagai menu



Gambar 2.6. Tampilan *Insert Bar* Sebagai Menu

c) **Tab Form**

Tab form digunakan untuk membuat elemen dalam *form*, misalnya saja *textarea*, *textfield*, *radio button*, *checkbox*.

d) **Tab Text**

Tab text digunakan untuk membuat pengaturan *text*. Misalnya saja membuat *text italic*, *strong*, *underline*, *h1*.

e) **Tab HTML**

Tab HTML digunakan untuk membuat garis *horizontal*, menambahkan meta



tag dalam *tag <head>*, dan *frame*.

f) ***Tab Application***

Tab application digunakan jika aplikasi kita sudah berhubungan dengan suatu bahasa pemrograman dan sebuah *database*.

g) ***Tab Flash elements***

Tab flash elements digunakan untuk memasukkan elemen *flash* dalam dokumen yang kita buat.

4. Code View

Code View digunakan untuk melihat kode *HTML* dari halaman *web* yang sedang kita buat.

5. Panel Properties

Panel properties merupakan panel yang digunakan untuk melihat dan mengubah *property* dari semua objek yang ada di area kerja. Masing-masing objek mempunyai *property* yang berbeda. Untuk melihat *property* dari objek yang diinginkan, seleksi dulu objek tersebut.

6. Design View

Design View digunakan untuk melihat tampilan *web* dari kode *HTML* yang kita buat.

(<http://repository.stmiktasikmalaya.ac.id/sesepuh/data/Macromedia%20Dreamweaver%208.pdf>)

2.4.8. Sekilas tentang XAMPP

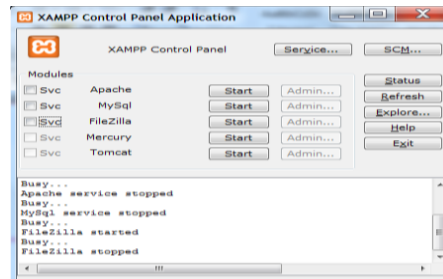
Nugroho (2013:1), mengemukakan ”*XAMPP* adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemrogram *web* khususnya *PHP* dan *MySQL*.

Adapun langkah-langkah pemasangan *XAMPP* dari *file zip* yang telah di *download* adalah sebagai berikut :

1. Unzip *file XAMPP* yang telah di *download*, ke dalam *file* direktori *c:/xampp*
2. Masuk ke dalam direktori *c:/xampp*



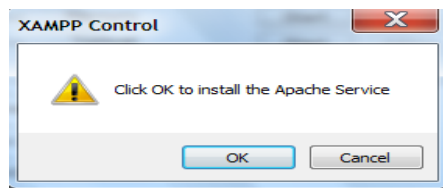
3. Pilih *file* xampp-control.exe, kemudian jalankan dengan menekan <enter> atau klik dua kali, tampilan berikut akan ditampilkan :



(Sumber : Nugroho, 2013:1)

Gambar 2.7. Tampilan *XAMPP Control Panel*

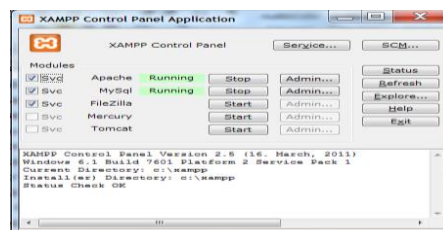
Dari tampilan di atas, beri tanda cek pada baris *Svc Apache*, dialog konfirmasi unjuk menjalankan *Apache* menjadi *Service* ditampilkan seperti berikut :



(Sumber : Nugroho, 2013:1)

Gambar 2.8. Tampilan Dialog Konfirmasi Menjalankan *Apache*

4. Klik tombol **OK**.
5. Setelah itu klik tombol start pada baris yang sama, menjalankan *Apache*. Maka tampilan menjadi seperti berikut :



(Sumber : Nugroho, 2013:1)

Gambar 2.9. Tampilan *Svc Apache Running*



2.4.9. JavaScript

Asporudin (2013:52), mengemukakan “ *Javascript* adalah bahasa *script* interpreter untuk menciptakan isi yang aktif dalam halaman *web*.”

Winarno *et.al* (2014:130-134), mengemukakan “ *Javascript* adalah kode pemrograman yang bisa dimasukkan atau disisipkan ke halaman *HTML*.”

JavaScript biasanya dituliskan pada bagian *<body>* dalam *HTML*. Contohnya:

```
<!Doctype html>
<html>
<body>
<p>
Menuliskan JavaScript Langsung ke Body:
</p>
<script>
document.write("<h1>Ini adalah Heading </h1>");
document.write("<p>Ini adalah paragraf.</p>");
</script>
</body>
</html>
```