



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Data

Sutabri (2012:3) mengemukakan, data merupakan bentuk mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut.

Mulyanto (2009:15) mengemukakan, data merupakan representasi dunia nyata yang mewakili suatu objek seperti manusia, hewan, peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi atau kombinasinya. Dengan kata lain, data merupakan kenyataan yang menggambarkan suatu kejadian dan kesatuan yang nyata.

Ladjamudin (2013:8) mengemukakan, data adalah deskripsi dari suatu dan kejadian yang kita hadapi (*the description of things and events that we face*).

Jadi data adalah sesuatu atau bahan baku yang masih belum bisa bermanfaat bagi para penggunanya.

2.1.2. Pengertian Sistem

Sutabri (2012:3) mengemukakan, sistem adalah suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu.

Gordon B, Davis (2012:6) mengemukakan, sistem bisa berupa abstrak atau fisik. Sistem yang abstrak adalah susunan gagasan atau konsepsi yang teratur yang saling bergantung.

Yakub (2012:2) mengemukakan, sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu”.

Jadi, dapat disimpulkan bahwa sistem adalah suatu kumpulan dari berbagai elemen yang saling berhubungan satu sama lain.



2.1.3. Klasifikasi Sistem

Sutabri (2012:15) menjelaskan bahwa, sistem dapat diklasifikasikan dari beberapa sudut pandangan, yaitu:

a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sistem fisik adalah sistem yang ada secara fisik, misalnya sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sistem buatan manusia adalah sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*, misalnya sistem informasi komputer.

c. Sistem deterministik dan sistem probabilistik

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

d. Sistem terbuka dan sistem tertutup

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya.



2.1.4. Karakteristik Sistem

Sutabri (2012:13) menjelaskan bahwa, karakteristik atau sifat-sifat tertentu dari sistem yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut:

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem adalah bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut.

d. Penghubung Sistem (*Interface*)

Penghubung sistem atau *interface* adalah media yang menghubungkan sistem dengan subsistem yang lain.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, di mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan *input* bagi subsistem lain.



g. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran 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.

2.1.5. Klasifikasi Sistem

Sutabri (2012:15) menjelaskan bahwa sistem dapat diklasifikasikan dari beberapa sudut pandangan, yaitu:

e. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sistem fisik adalah sistem yang ada secara fisik, misalnya sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

f. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sistem buatan manusia adalah sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*, misalnya sistem informasi komputer.

g. Sistem deterministik dan sistem probabilistik

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



h. Sistem terbuka dan sistem tertutup

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya.

2.1.6. Metode Pengembangan Sistem

Sukanto dan Shalahuddin (2013:28) 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 perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

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 pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengodean

Pada tahap pengodean, desain harus ditranlasikan 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. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

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.

2.1.12. Metode Pengujian Sistem

Yakub (2012:150) mengemukakan, “Metode pengujian sistem adalah suatu cara untuk menguji perangkat lunak dan data kemungkinan terjadi kesalahan”. Pengujian sistem dapat menggunakan metode *black box testing*. *Black box testing* berfokus pada pengujian persyaratan fungsional perangkat lunak yang dibuat untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program dan mampu mengungkapkan kesalahan yang lebih luas.

2.1.6. Pengertian Informasi

Mulyanto (2009:17) mengemukakan, informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut.

Menurut Barry E, informasi merupakan sesuatu yang menunjukkan hasil pengolahan data organisasi dan berguna kepada orang yang menerimanya.

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

Jadi informasi adalah data yang telah diproses dan diolah untuk digunakan dalam proses pengambilan keputusan.



2.2. Teori Judul

2.2.1. Pengertian Sistem Informasi

Sujatmiko (2012:251) mengemukakan, Sistem Informasi adalah sistem, pengiriman data melalui fasilitas telekomunikasi dari satu lokasi ke pusat pengolahan data tetapi data yang dikirimkan tidak langsung diproses oleh CPU.

Sutabri (2012:38) mengemukakan, sistem informasi adalah suatu system didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu.

Ladjamudin (2013:13) mengemukakan, Sistem Informasi dapat didefinisikan sebagai berikut:

1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan organisasi.

Jadi Sistem Informasi adalah sebuah system yang dibuat oleh manusia untuk mempermudah pekerjaannya.

2.2.2. Pengertian Administrasi

Soewarno Handyaningrat mengemukakan, administrasi ialah sebuah kegiatan ketatausahaan yang didalamnya diwarnai kegiatan yang mampu menyediakan informasi dan mempermudah seseorang buat mendapatkan sebuah informasi yang diperlukan.

The Liang Gie mengemukakan, administrasi diartikan sebagai sebuah rangkaian aktivitas nan dilakukan oleh sekelompok orang pada sebuah kerjasama guna mendapatkan sebuah tujuan.

Yuniar (2015:12) mengemukakan, administrasi adalah usaha dan kegiatan yang berkenaan dengan penyelenggaraan kebijaksanaan untuk mencapai tujuan.



Jadi Administrasi adalah seluruh rangkaian kerjasama antara dua orang atau lebih buat meraih sebuah tujuan dengan menggunakan fasilitas eksklusif yang digunakan serta mampu mencapai sasaran yang sudah ditetapkan.

2.2.5. Pengertian Sistem Informasi Administrasi Pada Jurusan Manajemen Informatika Politeknik Negeri Sriwijaya Berbasis Web

Pengertian Sistem Informasi Administrasi Pada Jurusan Manajemen Informatika Politeknik Negeri Sriwijaya Berbasis Web adalah suatu sistem informasi yang berguna untuk mempermudah masyarakat kampus seperti dosen dan mahasiswa dalam mencari informasi administrasi pada jurusan Manajemen Informatika secara *online* dan tanpa harus datang ke bagian administrasi untuk melihat informasi administrasi. Informasi administrasi tersebut adalah :

1. Administrasi kegiatan belajar dikelas maupun di laboratorium.
2. Administrasi tentang Laporan Magang.
3. Administrasi tentang Laporan Akhir.
4. Administrasi tentang jumlah Dosen, dan Jabatan Akademiknya.
5. Admnistrasi tentang pencatatan SP 1 (Surat Peringatan 1) sampai DO (Drop - Out).
6. Administrasi pencatatan data Alumni, tempat mereka bekerja dan judul Laporan Akhir yang mereka buat.

2.3. Teori Khusus

2.3.1. Data Flow Diagram (DFD)

Shalahuddin (2013:70) mengemukakan, *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*).

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD *level 0* atau sering disebut juga Diagram Konteks



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* (turunan) DFD *level 0* yang sebelumnya sudah dibuat.

3. Membuat DFD *level 2*

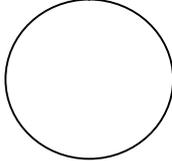
Modul-modul pada DFD *level 1* dapat di-*breakdown* lebih *detail* tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup *detail* dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD *level 2* sama dengan jumlah modul pada DFD *level 3* dan seterusnya.

4. Membuat DFD *level 3* dan seterusnya

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

Adapun simbol-simbol *Data Flow Diagram* adalah sebagai berikut:

Tabel 2.1. Simbol *Data Flow Diagram* (DFD)

No.	Simbol	Keterangan
1.	Proses (<i>Process</i>) 	Pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program

Lanjutan Tabel 2.1. Simbol *Data Flow Diagram* (DFD)

No.	Simbol	Keterangan
-----	--------	------------



2.	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 harus sesuai dengan perancangan tabel-tabel pada basis data Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda
3.	Entitas Luar <i>(External Entity)</i> 	Orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan Catatan: Nama yang digunakan pada entitas luar biasanya 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>)

(Sumber: Shalahuddin, 2013:71-72)

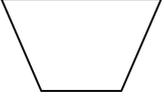
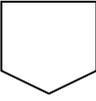
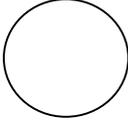
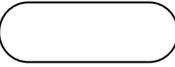
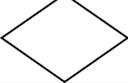
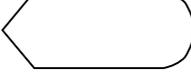
2.3.2. Block Chart

Kristanto (2008:75) mengemukakan, *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.

Adapun simbol-simbol *Block Chart* adalah sebagai berikut:

Tabel 2.2. Simbol-Simbol *Block Chart*



No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku atau bendel atau 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 (<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
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>)
12.		Layar peraga (<i>monitor</i>)

Lanjutan Tabel 2.2. Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
13.		Pemasukan data secara manual

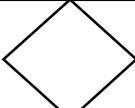
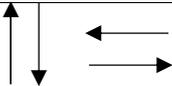
(Sumber: Kristanto, 2008:64)

2.3.3. Flowchart

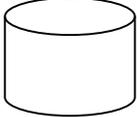
Ewolf community (2012:16), *flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya.

Adapun simbol-simbol *Flowchart* adalah sebagai berikut:

Tabel 2.3. Simbol-Simbol *Flowchart*

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

Lanjutan Tabel 2.3. Simbol-Simbol *Flowchart*

No.	Simbol	Arti
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 sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau sebuah pita magnetic
13.		Simbol database atau basis data

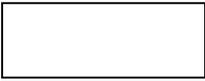
(Sumber: Ewolf Community, 2012:17)

2.3.4. *Entity Relationship Diagram (ERD)*

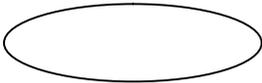
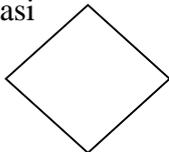
Shalahuddin (2013:50) mengemukakan, ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi *Chen* (dikembangkan oleh *Peter Chen*), *Barker* (dikembangkan oleh *Richard Barker*, *Ian Palmer*, *Harry Ellis*), notasi *Crow's Foot*, dan beberapa notasi lain.

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

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

No.	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	<i>Entitas</i> 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.
2.	Atribut 	<i>Atribut</i> merupakan <i>field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Relasi 	Relasi yang menghubungkan antar <i>entitas</i> , biasanya diawali dengan kata kerja.
4.	Asosiasi N 	Penghubung antara relasi dan <i>entitas</i> di mana di kedua ujungnya memiliki <i>multiplicity</i> .

(Sumber:Shalahuddin, 2013:50)

2.3.5. Kamus Data (*Data Dictionary*)

Shalahuddin (2013:73) mengemukakan, 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). Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur.

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

Tabel 2.5. Simbol-Simbol Kamus Data

No.	Simbol	Arti
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik.....atau.....
4.	{ } ⁿ	N kali diulang atau bernilai banyak
5.	()	Data opsional
6.	*....*	Batas komentar

(Sumber: Shalahuddin,2013:74)



2.3.6. Daftar Kejadian (*Event List*)

Kristanto (2008:70) mengemukakan, daftar kejadian digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Suatu kejadian mewakili satu aliran data atau proses dalam diagram konteks serta deskripsi penyimpanan yang digunakan untuk memodelkan data harus diperhatikan dalam kaitannya dengan daftar kejadian.

Adapun cara-cara mendeskripsikan daftar kejadian adalah sebagai berikut:

1. Pelaku adalah entiti luar, jadi bukan sistem.
2. Menguji setiap entiti luar dan mencoba mengevaluasi setiap entiti 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 entiti luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respondari 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

2.4.1. *PHP Hypertext Preprocessor*

2.4.1.1. Pengertian *PHP*

Winarno (2014:49) mengemukakan, *PHP (Hypertext Preprocessor)* adalah sebuah bahasa pemrograman *web* berbasis *server (server-side)* yang mampu memarsing kode *PHP* dari kode *web* dengan ekstensi *.php*, sehingga menghasilkan tampilan *website* yang dinamis di sisi *client (browser)*.

Jadi, *PHP* adalah bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis *web*.

2.4.1.2. Memasukkan Kode *PHP*

Winarno (2014:64), mengemukakan, ada empat jenis tag yang bisa digunakan untuk memasukkan kode *PHP*, yaitu :

Tabel 2.6. Tag Memasukkan Kode *PHP*

No.	Jenis Tag	Tag Pembuka	Tag Penutup
1.	Tag standar	<?php	?>
2.	Tag pendek	<?	?>
3.	Tag ASP	<%	%>
4.	Tag <i>script</i>	<script language="php">	</script>

(Sumber: Winarno, 2014:63)

Di dalam tag kode *PHP* juga terdapat komentar, ada 3 cara untuk membuat komentar di dalam tag *PHP* yaitu:

1. Tag // digunakan untuk komentar dalam satu baris,
2. Tag # juga digunakan untuk komentar dalam satu baris, dan
3. Tag /*.....*/ digunakan apabila jumlah keterangan lebih dari 1 baris.

Nugroho (2013:157) mengemukakan, *PHP* memiliki perintah *echo* dan *print* untuk menampilkan apapun ke layar. Lebih baik gunakan perintah *echo* pada semua program yang dibuat supaya program bisa dijalankan pada semua paket aplikasi *PHP*. Dalam menggunakan perintah *echo* setiap objek yang berbentuk teks harus kita pait dengan **petik ganda** (“), sedangkan untuk objek yang berbentuk angka bisa diabaikan. Setiap satu baris perintah diakhiri dengan tanda **titik koma** (;), itu wajib jadi selama belum ada tanda **titik koma**, maka pembacaan kode akan dilanjutkan pada baris berikutnya.

2.4.1.3. Variabel *PHP*

Winarno (2014:64) mengemukakan, variabel merupakan sebuah istilah yang menyatakan sebuah tempat yang digunakan untuk menampung nilai-nilai tertentu dimana nilai di dalamnya bisa diubah-ubah. Di *PHP* variabel dikenal dengan



adanya tanda (\$) yang kemudian bisa diikuti dengan angka, huruf, dan *underscore*. Namun, variabel tidak bisa mengandung spasi.

2.4.2. Responsive Web Design

Riyanto (2014:12) mengemukakan, *Responsive Web Design* adalah *client-side* yang berarti halaman akan dikirim ke browser perangkat dan browser kemudian memodifikasi tampilan halaman yang berkaitan dengan ukuran jendela browser.

2.4.3. CSS(Cascading Style Sheet)

2.4.3.1. Pengertian CSS (Cascading Style Sheet)

Koesheryatin (2014:101) mengemukakan, CSS (*Cascading Style Sheet*) 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 (*Hypertext Markup Language*).

2.4.3.2. Cara Menuliskan CSS

Koesheryatin (2014:101) mengemukakan, ada dua cara yang bisa diterapkan untuk menggunakan CSS langsung di dalam satu file HTML (*internal/inline style sheet*). Cara yang kedua dengan memanggil CSS tersebut dari file CSS tersendiri (*external style sheet*).

2.4.4. Pengertian JS (Java Script)

Winarno (2014:129) mengemukakan, *Java Script* adalah bahasa scripting yang banyak digunakan oleh programmer *web* untuk memberikan efek pemrograman di halaman.

2.4.5. Pengertian XAMPP

Nugroho (2013:1) mengemukakan, XAMPP adalah paket program *web* lengkap yang dapat Anda pakai untuk belajar pemrograman *web*, khususnya *PHP*



dan *MySQL*, paket ini dapat didownload secara gratis dan legal.

2.4.6. Pengertian Basis Data (*Database*)

Kadir (2010:2) mengemukakan, *database* (basis data) dapat diungkapkan sebagai suatu pengorganisasian data dengan bantuan komputer yang memungkinkan data dapat diakses dengan mudah dan cepat. Dalam hal ini, pengertian akses dapat mencakup pemerolehan data maupun manipulasi data, seperti menambah dan menghapus data.

Jadi, basis data adalah suatu data yang diletakkan di dalam sebuah tempat sehingga memudahkan kita untuk mengaksesnya.

2.4.7. *phpMyAdmin*

2.4.7.1. Pengertian *phpMyAdmin*

Nugroho (2013:71) mengemukakan, *phpmyadmin* adalah *tools MySQL Client* berlisensi *freeware*, anda dapat mengunduhnya secara gratis pada situs resminya, *phpmyadmin* harus dijalankan di sisi *server web* dan pada komputer harus tersedia *PHP*, karena berbasis *web*.

2.4.7.2. Menjalankan *phpMyAdmin*

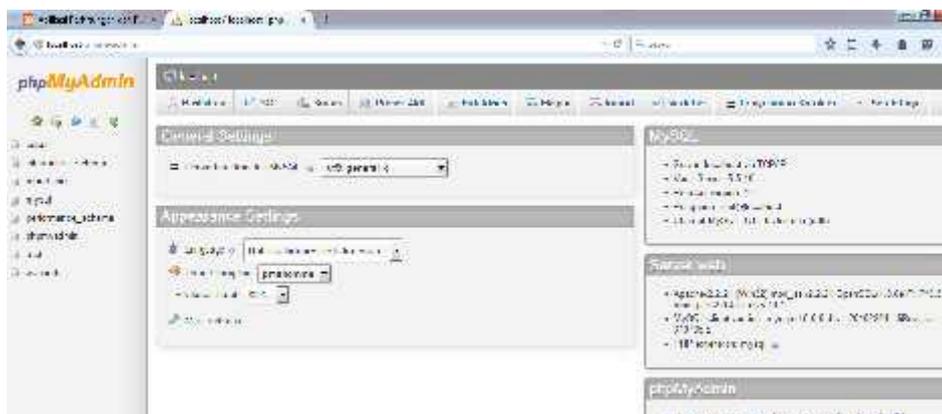
Nugroho (2013:72) mengemukakan, cara menjalankan *phpmyadmin* adalah sebagai berikut:

1. Buka **XAMPP Control Panel** lalu jalankan modul servis *Apacheweb server*.
2. Jalankan juga *MySQL server*nya, sehingga sekarang **Apache** dan **MySQL** statusnya *running*.



Gambar 2.1. XAMPP Control Panel

3. Sekarang buka *web browser*, misalnya **Internet Explore** atau **Firefox**.
4. Pada kotak alamat, ketikkan; <http://localhost/phpmyadmin/> tekan enter.



Gambar 2.2. *phpmyadmin*

2.4.8. MySQL

2.4.8.1. Pengertian MySQL

Kadir (2013:15) mengemukakan, *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. *MySQL* tergolong sebagai *database* relasional. Pada model ini, data dinyatakan dalam bentuk dua dimensi yang secara khusus dinamakan tabel. Tabel tersusun atas baris dan kolom.



2.4.8.2. Fungsi-Fungsi MySQL

eWolf Community (2014:206) mengemukakan, fungsi yang berawalan **mysql_** yang digunakan untuk mengakses *database serverMySQL* yang dibahas di bawah ini :

Tabel 2.7. Fungsi-Fungsi MySQL

No.	Nama Fungsi	Kegunaan
1.	mysql_connect()	Digunakan untuk menghubungkan dokumen dengan <i>databaseMySQL</i>
2.	mysql_close()	Digunakan untuk memutus hubungan dengan <i>databaseMySQL</i>
3.	mysql_affected_rows()	Digunakan untuk mendapatkan jumlah baris (<i>record</i>) yang dihasilkan dari operasi <i>MySQL</i> sebelumnya
4.	mysql_fetch_array() mysql_fetch_assoc()	Digunakan untuk mendapatkan <i>array</i> dari suatu <i>recordset</i>
5.	mysql_fetch_field()	Digunakan untuk mendapatkan sebuah objek yang mengandung informasi <i>field</i> dari sebuah <i>recordset</i>
6.	mysql_fetch_lengths()	Digunakan untuk mendapatkan panjang baris pada setiap isi <i>field</i>
7.	mysql_fetch_object()	Digunakan untuk menghasilkan baris dari <i>recordset</i> sebagai sebuah objek
8.	mysql_fetch_row()	Digunakan untuk menghasilkan <i>array</i> numerik dari baris pada sebuah <i>recordset</i>
9.	mysql_field_len()	Digunakan untuk mendapatkan informasi panjang maksimum <i>field</i> dalam sebuah <i>recordset</i>
10.	mysql_field_name()	Digunakan untuk mendapatkan nama <i>field</i> dalam sebuah <i>recordset</i>
11.	mysql_field_seek()	Digunakan untuk melompat ke <i>field</i> tertentu



		dalam sebuah <i>recordset</i>
12.	mysql_get_client_info() mysql_get_host_info()	Digunakan untuk mendapatkan informasi tentang <i>MySQL client</i>
13.	mysql_get_server_info()	Digunakan untuk mendapatkan informasi tentang <i>MySQL server</i>
14.	mysql_info()	Digunakan untuk mendapatkan informasi tentang <i>query</i> terakhir
15.	mysql_num_fields()	Digunakan untuk mendapatkan informasi jumlah <i>field</i> pada suatu <i>recordset</i>
16.	mysql_num_rows()	Digunakan untuk mendapatkan informasi jumlah baris pada <i>recordset</i>
17.	mysql_ping()	Digunakan untuk memeriksa koneksi <i>server</i> dan akan mencoba untuk melakukan koneksi ulang jika koneksi terputus
18.	mysql_query()	Digunakan untuk menjalankan <i>query</i> pada <i>databaseMySQL</i>
19.	mysql_result()	Digunakan untuk mendapatkan nilai dari <i>field</i> dalam sebuah <i>recordset</i>
20.	mysql_select_db()	Digunakan untuk menentukan (set) <i>database MySQL</i> yang aktif
21.	mysql_stat()	Digunakan untuk menghasilkan informasi status <i>server</i> terkini

(Sumber: eWolf Community, 2014:206)

2.4.9. Adobe Dreamweaver CS3

2.4.9.1. Pengertian Adobe Dreamweaver CS3

Madcoms (2008:1) mengemukakan, *Dreamweaver* adalah sebuah HTML (Hyper Text Markup Language) editor profesional untuk mendesain *web* secara visual dan mengelola situs atau halaman *web*.

2.4.9.2. Ruang Kerja Adobe Dreamweaver CS3

Madcoms (2008:2) menjelaskan tentang ruang kerja dasar *Dreamweaver CS3* adalah sebagai berikut:

1. *Insert Bar*

Berisi tombol-tombol untuk menyisipkan berbagai macam objek seperti: *image*, tabel dan *layer* ke dalam dokumen. *Insert Bar* terdiri dari beberapa kategori sesuai dengan jenis yang disisipkan.



Gambar 2.3. Tampilan *Insert Bar*

2. *Document Toolbar*

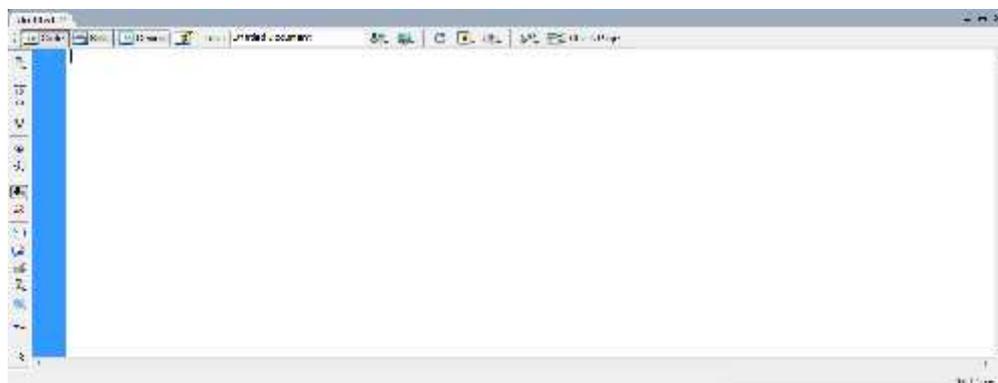
Berisi tombol-tombol dan menu *pop-up* yang menyediakan tampilan berbeda dari jendela dokumen. Anda bisa mengatur tampilan jendela dokumen dalam bentuk *design view* atau *code view* atau kedua-duanya. Anda juga dapat memberikan judul jendela dokumen pada bagian *title*.



Gambar 2.4. Tampilan *Document Toolbar*

3. *Coding Window*

Berisi kode-kode HTML dan tempat untuk menuliskan kode-kode pemrograman, misalnya *PHP* atau *ASP*.

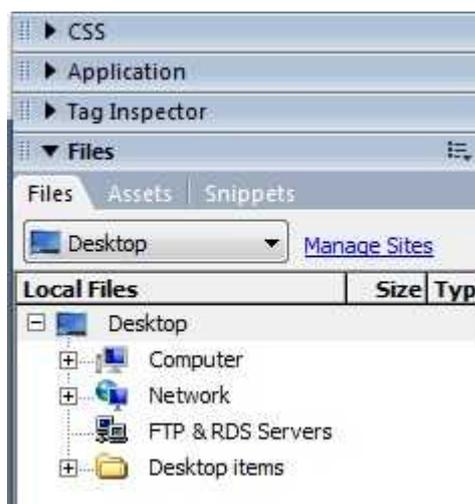


Gambar 2.5. Tampilan *Coding Window*



4. *Panel Group*

Adalah kumpulan *panel* yang saling berkaitan satu sama lainnya yang dikelompokkan di bawah satu judul.



Gambar 2.6. Tampilan *Panel Group*

5. *Property Inspector*

Digunakan untuk melihat dan mengubah berbagai properti objek atau teks.



Gambar 2.7. Tampilan *Property Inspector*

6. *Jendela Dokumen*

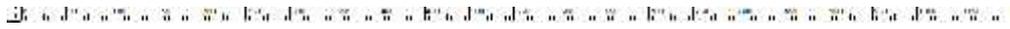
Berfungsi untuk menampilkan dokumen di mana Anda sekarang bekerja.



Gambar 2.8. Tampilan *Jendela Dokumen*

7. Ruler

Digunakan untuk memudahkan pengukuran saat Anda mendesain suatu halaman *web*.



Gambar 2.9. Tampilan *Ruler*

8. Site Panel

Digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *web* Anda.



Gambar 2.10. Tampilan *Site Panel*