



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Sujatmiko (2012:156) menyatakan, komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program.

Hartono (2013:27) berpendapat bahwa, komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang untuk secara otomatis melakukan serangkaian urutan penghitungan (*arithmetic*) atau proses–proses yang diurutkan secara logis.

2.1.2. Pengertian Internet

Sujatmiko (2012:138) menyatakan, internet atau *Interconnected Network* adalah jaringan global yang menghubungkan berjuta-juta komputer di seluruh dunia melalui jalur telepon kabel maupun satelit. Dengan adanya internet, jutaan orang di seluruh dunia berbagi segala jenis informasi.

2.1.3. Pengertian Data

Kristanto (2008:7) mendefinisikan, data sebagai penggambaran dari suatu kejadian yang sedang terjadi, dimana data tersebut akan diolah dan diterapkan dalam sistem menjadi input yang berguna dalam suatu sistem.

Sutabri (2012:1) menyatakan, data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.

Ladjamudin (2013:20) menyatakan, data merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.

2.1.4. Pengertian Pengolahan Data

Ladjamudin (2013:9) menyatakan, pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan.



Kristanto (2008:8) menyatakan, pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan.

2.1.5. Karakteristik Sistem

Sutabri (2012:20), menjelaskan tentang karakteristik atau sifat-sifat tertentu dari sistem yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut:

1. 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.

2. 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.

3. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut.



5. Masukkan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukkan 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 komputernya dan “data” adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukkan bagi subsistem yang lain seperti informasi. Informasi dapat digunakan sebagai masukkan untuk pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi subsistem lain.

7. Pengolahan Sistem (*Proses*)

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

8. Sasaran Sistem (*Objective*)

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

2.1.6. Klasifikasi Sistem

Kristanto (2008:5), menjelaskan bahwa sistem dapat diklasifikasikan menjadi beberapa bagian yaitu:

1. Sistem abstrak dan sistem fisik

Sistem abstrak merupakan sistem yang tidak bisa dilihat secara mata biasa dan biasanya sistem ini berupa pemikiran atau ide-ide. Sistem fisik merupakan sistem yang bisa dilihat secara mata biasa dan biasanya sering digunakan oleh manusia.



2. Sistem alamiah dan sistem buatan

Sistem alamiah merupakan sistem yang terjadi karena pengaruh alam. Sedangkan sistem buatan merupakan sistem yang dirancang dan dibuat oleh manusia.

3. Sistem tertutup dan sistem terbuka

Sistem tertutup merupakan sistem yang tidak berhubungan dengan bagian luar sistem dan biasanya tidak terpengaruh oleh kondisi di luar sistem. Sedangkan sistem terbuka merupakan sistem yang berhubungan dengan bagian luar sistem.

2.1.7. Tahap Pengembangan Sistem

Kristanto (2011:37), tahap-tahap pengembangan sistem adalah sebagai berikut :

1. Penyelidikan Awal

Pada tahap penyelidikan awal, analisis belajar dari pemakai mengenai apa yang diharapkan dari sebuah sistem informasi yang baru. Hal-hal yang perlu diperhatikan dalam tahapan ini:

- a. Mencoba memahami dan menjelaskan apa yang diharapkan oleh pemakai (sistem informasi bagaimana yang mereka perlukan).
- b. Menentukan ruang lingkup dari studi sistem informasi.
- c. Menentukan kelayakan dari masing-masing alternatif dengan memperkirakan keuntungan kerugian yang didapat.

2. Studi Kelayakan

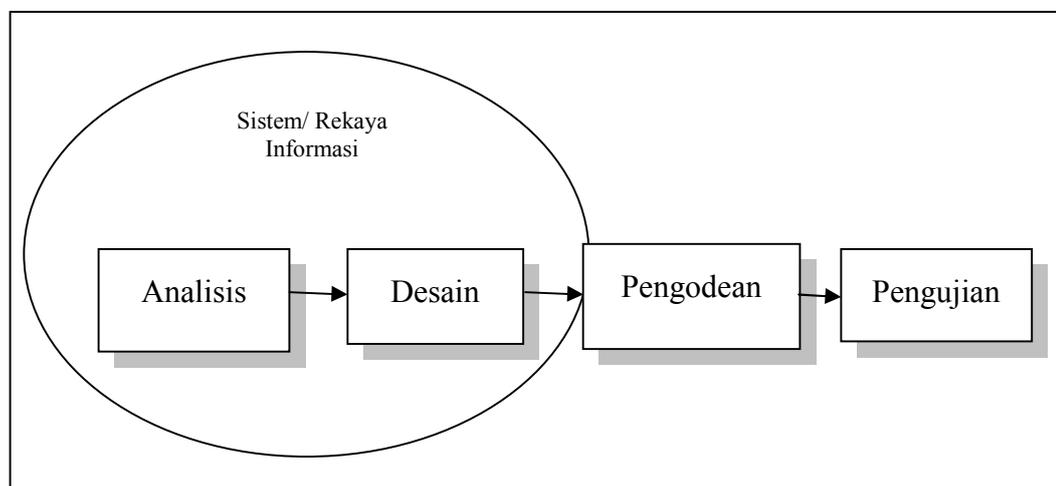
Studi kelayakan merupakan tahap yang paling penting, karena didalamnya menyangkut berbagai aspek sistem baru yang diusulkan. Laporan mengenai studi kelayakan harus disampaikan kepada manajemen, yang pada gilirannya akan memberikan beberapa perubahan, menyarankan untuk diadakan penelitian lebih mendalam atau memutuskan untuk segera dilaksanakan.

Adapun tujuan dilaksanakannya studi kelayakan adalah sebagai berikut:

- a. Memperhitungkan sifat penyusunan sistem dengan memperhitungkan keberadaan masalah dan sifat masalah.
- b. Memperhitungkan jangkauan masalah.
- c. Mengajukan aksi-aksi yang dapat menyelesaikan masalah.
- d. Memperhitungkan kelayakan penyusunan sistem yang diajukan.
- e. Menyusun rencana detail untuk langkah analisis sistem.
- f. Menyusun rencana ringkasan untuk seluruh proyek penyusunan sistem.

2.1.8. 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 dimulai dari analisis desain, pengodean, pengujian, dan tahap pendukung (*support*). Berikut ini adalah gambar model air terjun:



Gambar 2.1. Model *Waterfall*

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat



lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk 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 mentranslasi 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 dalam 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 logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintance*)

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 Judul

2.2.1. Pengertian Sistem

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

Sutabri (2012:10) menyatakan, sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.

Kristanto (2008:1) menyatakan, sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

2.2.2. Pengertian Informasi

Hartono (2013:15) menyatakan, informasi pada dasarnya adalah sehimpunan data yang telah diolah menjadi sesuatu yang memiliki arti dan kegunaan lebih luas.

Sujatmiko (2012:135) menyatakan, informasi merupakan serangkaian simbol yang jika disusun menurut aturan tertentu dapat menyampaikan suatu pesan.

Kristanto (2008:7) menjelaskan, informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima.

2.2.3. Pengertian Sistem Informasi

Sutabri (2012:46) menyatakan, sistem informasi adalah suatu sistem di dalam 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 kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

Ladjamudin (2013:13) menyatakan, sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.



2.2.4. Pengertian Basis Data (*Database*)

Sujatmiko (2012:40) menyatakan, basis data (*database*) merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.

Ladjamudin (2013:21) menyatakan, *database* merupakan kumpulan file yang saling terintegrasi, namun *database* tidak akan dapat diakses oleh siapapun tanpa adanya *software* aplikasi.

Selain itu, Sukanto dan Shalahuddin (2014:43) menyatakan, basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.

2.2.5. Pengertian Alumni

Kamus Besar Bahasa Indonesia (2008:45), alumni adalah orang-orang yang telah mengikuti atau tamat dari suatu sekolah atau perguruan tinggi.

2.2.6. Pengertian Per-

Kamus Besar Bahasa Indonesia (2008:1049), per adalah tiap-tiap; bagi (dalam angka pecahan, memakai, menggunakan); menjadikan atau membuat menjadi; membagi menjadi; melakukan; memanggil atau menganggap.

2.2.7. Pengertian Jurusan

Kamus Besar Bahasa Indonesia (2008:595), jurusan adalah arah; bagian (pengkajian ilmu); bagian dari suatu fakultas atau sekolah tinggi yang bertanggung jawab untuk mengelola dan mengembangkan suatu bidang studi, misal jurusan akuntansi, jurusan manajemen.



2.2.8. Pengertian Politeknik Negeri Sriwijaya

Politeknik Negeri Sriwijaya adalah salah satu perguruan tinggi negeri yang berada di Palembang. Politeknik Negeri Sriwijaya dahulunya bernama Politeknik Universitas Sriwijaya yang resmi dibuka pada tanggal 20 September 1982.

2.2.9. Pengertian Sistem Informasi Database Alumni Per-Jurusan pada Politeknik Negeri Sriwijaya Berbasis Web

Sistem Informasi *Database* Alumni Per-Jurusan pada Politeknik Negeri Sriwijaya *Berbasis Web* merupakan suatu sistem informasi tentang alumni Politeknik Negeri Sriwijaya dari tiap jurusan yang dibuat secara *online* agar dapat diakses kapanpun dan dimanapun dengan menggunakan bahasa pemrograman *PHP* dan *database MySQL*.

2.3. Teori Khusus

2.3.1. Data Flow Diagram (DFD)

Ladjamudin (2013:64) menyatakan, *Data Flow Diagram* (DFD) atau diagram aliran data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Diagram ini memudahkan pemakai atau *user* yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan. Ada beberapa tahapan perancangan dengan menggunakan *Data Flow Diagram*, yaitu sebagai berikut:

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari *Data Flow Diagram* yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem.

2. Diagram Nol/Zero (*Overview Diagram*)

Diagram nol adalah diagram yang menggambarkan proses dari *Data Flow Diagram*. Diagram nol memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal *entity*.



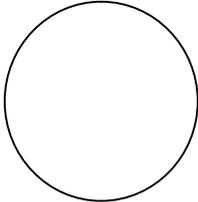
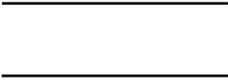
3. Diagram Rinci (*Level Diagram*)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram zero atau diagram level di atasnya.

Selain itu, Sukamto dan Shalahuddin (2013:70) menyatakan, *Data Flow Diagram* atau dalam bahasa Indonesia menjadi diagram alir data adalah representatif grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

Berikut ini merupakan simbol–simbol atau notasi–notasi dengan menggunakan teknik Edward Yourdon dan Tom DeMarco yang menggambarkan *Data Flow Diagram* (DFD), sebagai berikut:

Tabel 2.1. Simbol–simbol *Data Flow Diagram* (DFD) menurut Edward Yourdon dan Tom DeMarco

Notasi	Keterangan
	Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.
	File atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel–tabel basis data yang dibutuhkan, tabel–tabel ini juga harus sesuai dengan perancangan tabel–tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM),



Notasi	Keterangan
	<p><i>Physical Data Model (PDM)</i>.</p> <p>Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>
	<p>Entitas luar (<i>External Entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.</p>
	<p>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>).</p> <p>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”.</p>

(Sumber: Sukanto dan Shalahuddin, 2014:71-72)

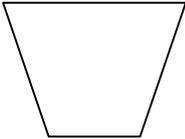
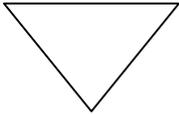
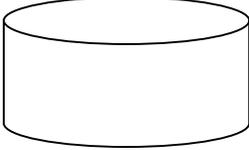
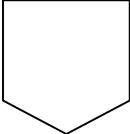
2.3.2. Pengertian *Block Chart*

Kristanto (2008:68), *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.

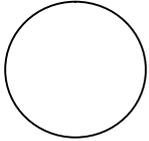
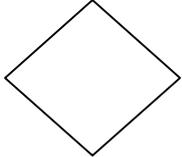


Adapun simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.2. Simbol-simbol dalam *Blockchart*

Simbol	Keterangan
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
	Multi dokumen
	Proses manual
	Proses yang dilakukan oleh komputer
	Menandakan dokumen yang diarsipkan (arsip manual)
	Data penyimpanan (data storage)
	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain



Simbol	Keterangan
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran
	Pengambilan keputusan (<i>decision</i>)
	Layar peraga (<i>monitor</i>)
	Pemasukkan data secara manual

(Sumber: Kristanto, 2008:68-70)

2.3.3. Pengertian *Flow Chart*

Ladjamudin (2013:263) menyatakan, *flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.

2.3.3.1. Simbol-simbol *Flowchart*

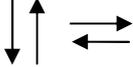
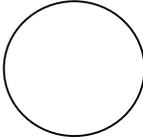
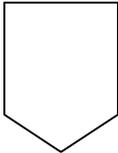
Flowchart disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses didalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut.

1. *Flow Direction Symbols* (Simbol penghubung/alur)
2. *Processing Symbols* (Simbol proses).
3. *Input-output Symbols* (Simbol *input-output*)

2.3.3.1.1. Flow Direction Symbols

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut.

Tabel 2.3. *Flow Direction Symbols*

No.	Simbol	Arti
1.		Simbol arus / flow Untuk menyatakan jalannya arus suatu proses.
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya.
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.
4.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.

(Sumber: Ladjamudin, 2013:266)

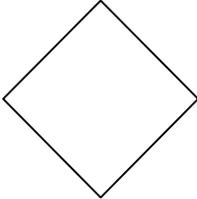
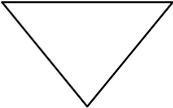
2.3.3.1.2. Processing Symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses/prosedur. Simbol-simbol tersebut adalah sebagai berikut.

Tabel 2.4. *Processing Symbols*

No.	Simbol	Arti
1.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.



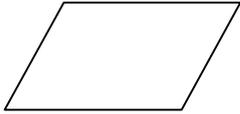
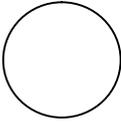
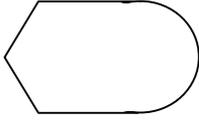
No.	Simbol	Arti
2.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
3.		Simbol Decision/logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/ tidak.
4.		Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program.
6.		Simbol Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		Simbol Offline Storage Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.		Simbol Manual Input Untuk memasukkan data secara manual dengan menggunakan online keyboard.

(Sumber: Ladjamudin, 2013:267)

2.3.3.1.3. Input-output Symbols

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol-simbol tersebut adalah sebagai berikut.

Tabel 2.5. Input-output Symbols

No.	Simbol	Arti
1.		Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
2.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output dituliskan ke kartu.
3.		Simbol Magnetic-tape Unit Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic.
4.		Simbol Disk Storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
5.		Simbol Document Untuk mencetak laporan ke printer.
6.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

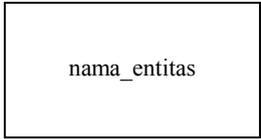
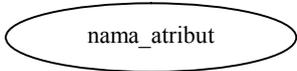
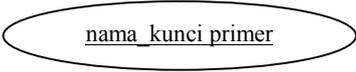
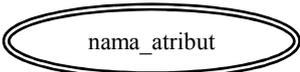
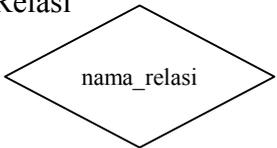
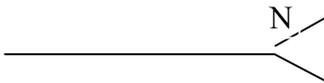
(Sumber: Ladjamudin, 2013:268)

2.3.4. Pengertian *Entity Relationship Diagram* (ERD)

Ladjamudin (2013:142) menyatakan, *Entity Relationship Diagram* (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.

Adapun simbol-simbol yang digunakan dalam *Entity Relationship Diagram* (ERD), yaitu:

**Tabel 2.6.** Simbol–Simbol *Entity Relationship Diagram* (ERD)

Simbol	Deskripsi
Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; 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.
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
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).
Atribut Multinilai/ <i>Multivalue</i> 	<i>Field</i> 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/ <i>Association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya punya <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

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



2.3.5. Pengertian Kamus Data (*Data Dictionary*)

Sukanto dan Shalahuddin (2013:73) menyatakan, 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 biasanya berisi:

1. Nama, nama dari data
2. Digunakan pada, merupakan proses–proses yang terkait data
3. Deskripsi, merupakan deskripsi data
4. Informasi tambahan, seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

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

Tabel 2.7. Simbol-simbol Kamus Data (*Data Dictionary*)

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

(Sumber : Sukanto dan Shalahuddin, 2013:74)

2.4. Teori Program

2.4.1. Sekilas Tentang *PHP Hypertext Preprocessor* (PHP)

2.4.1.1. Pengertian PHP

Sidik (2012:5), *PHP* adalah kependekan dari *PHP:Hypertext Preprocessor* (rekursif, mengikut gaya penamaan di *nix), merupakan bahasa utama *script*



server-side yang disisipkan pada HTML yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi desktop.

Kadir (2008:2) menyatakan, menurut dokumen resmi PHP, PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya yang dikirimkan ke klien tempat pemakai menggunakan *browser*.

2.4.1.2. Skrip PHP

Skrip PHP berkedudukan sebagai tag dalam bahasa HTML. Sebagaimana diketahui, HTML (*HyperText Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. Sebagai contoh, berikut adalah kode HTML (disimpan dengan ekstensi *.htm* atau *.html*):

```
<html>
  <head>
    <title>Latihan Pertama</title>
  </head>
  <body>
    Selamat Belajar PHP
  </body>
</html>
```

Bila dijalankan melalui *browser*, kode tersebut membentuk tampilan seperti berikut:



Gambar 2.2. Tampilan Program HTML pada *browser*



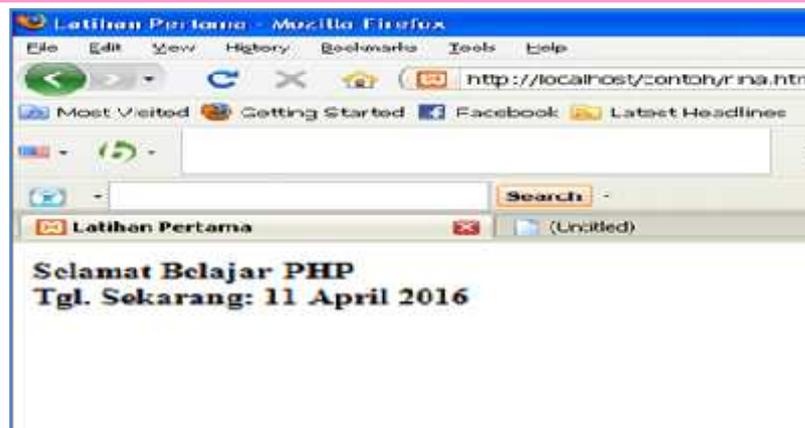
Adapun kode berikut adalah contoh kode PHP yang berada di dalam kode HTML:

```
<html>
  <head>
    <title>Latihan Pertama</title>
  </head>
  <body>
    Selamat Belajar PHP <br>
    <?php
      Printf("Tgl. Sekarang: %s", date (" d F Y"));
    ?>
  </body>
</html>
```

Kode di atas disimpan dengan ekstensi .php. Perhatikan baris-baris berikut:

```
<?php
  Printf("Tgl. Sekarang: %s ", Date ("d F Y"));
?>
```

Kode inilah yang merupakan kode PHP. Kode PHP diawali dengan <?php dan diakhiri dengan ?>. Pasangan kedua kode inilah yang berfungsi sebagai tag kode PHP. Berdasarkan tag inilah, pihak *server* dapat memahami kode PHP dan kemudian memprosesnya. Hasilnya dikirim ke *browser*. Contohnya diperhatikan pada gambar berikut:



Gambar 2.3. Tampilan kode PHP pada browser

2.4.2. Sekilas Tentang MySQL

2.4.2.1. Pengertian MySQL

Sidik (2012:313) menyatakan, SQL adalah bahasa standar yang digunakan untuk melakukan akses dan manipulasi *database*, dalam hal ini database relasional.

Kadir (2008:348) menyatakan, MySQL adalah salah satu *database server* yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*-nya. Selain itu, ia bersifat *Open Source* (Anda tidak perlu membayar untuk menggunakannya) pada berbagai platform (kecuali untuk jenis Enterprise, yang bersifat komersial).

2.4.2.2. Fitur-fitur MySQL

Sebagai *software* DBMS, MySQL memiliki sejumlah fitur seperti dibawah ini:

1. Multiplatform

MySQL tersedia pada beberapa platform (Windows, Linux, Unix, dan lain-lain).

2. Andal, cepat, dan mudah digunakan

MySQL tergolong sebagai *database server* (server yang melayani permintaan terhadap database) yang andal, dapat menangani database yang besar dengan



kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses *database*, dan sekaligus mudah untuk digunakan.

3. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan.

4. Dukungan SQL

MySQL mendukung perintal SQL (*Structured Query Language*). Sebagaimana diketahui, SQL merupakan standar dalam pengaksesan database relasional.

2.4.2.3. Tipe Data

MySQL mendukung banyak tipe data yang terbagi menjadi beberapa kategori, antara lain tipe *numeric*, tipe date dan time, dan tipe string. Berikut ini akan dibahas tentang kategori tipe data masing–masing tersebut.

2.4.2.3.1. Tipe Numerik

MySQL mendukung hampir semua tipe dara *numeric*/angka, asalkan sesuai dengan standar SQL. Termasuk di dalamnya adalah tipe data untuk bilangan bulat (misalnya, Integer, SmallInt, Decimal, dan Numeric) serta tipe data yang digunakan untuk bilangan desimal atau pecahan (misalnya, Float, Real, dan Double Precision). Kata kunci INT merupakan persamaan untuk Integer dan kata kunci DEC merupakan persamaan untuk Decimal.

Untuk menyesuaikan dengan standar SQL, MySQL juga mendukung tipe *integer* lain, seperti TinyInt, MediumInt, dan BigInt. Berikut ini tabel yang menunjukkan jenis tipe data *numeric*/angka.

Tabel 2.8. Tipe Data Numerik

Type	Byte	Signed	UnSigned
TinyInt	1	-128 s/d 127	0 s/d 255
SmallInt	2	-32768 s/d 32767	0 s/d 65535
MediumInt	3	-8388608 s/d 8388607	0 s/d 16777215



Type	Byte	Signed	UnSigned
Int, Integer	4	-2147483648 s/d 2147483647	0 s/d 4294967295
BigInt	8	-922337203685475808 s/d 9223372036854775807	0 s/d 1844674407370955 1615
Float	4	-3.402823466E+38 s/d -1.175494351E-38 1.175494351E-38 s/d 3.402823466E+38	
Double	8	-1.7976931348623157E+308 s/d 2.2250738585072014 E-308	

2.4.2.3.2. Tipe Date dan Time

MySQL memiliki tipe data yang berhubungan dengan penanganan tanggal dan pewaktuan, antara lain *Datetime*, *Date*, *Timestamp*, *Time*, dan *Year*. Berikut ini jenis-jenis tipe data yang berhubungan dengan tipe date dan time.

Tabel 2.9. Tipe Date dan Time

Type	Keterangan
Datetime	Ukuran 8 byte. Kombinasi tanggal dan jam dengan jangkauan mulai '1000 - 01 - 01 00:00:00' s/d '9999 - 12 - 31 23:59:59'.
Date	Ukuran 3 byte. Tanggal dengan jangkauan mulai '1000 - 01 - 01' s/d '9999 - 12 - 31'.
Timestamp	Ukuran 4 byte. Kombinasi tanggal dan jam dengan jangkauan mulai dari '1970 - 01 - 01 00:00:00 s/d tahun 2037.
Time	Ukuran 3 byte. Waktu dengan jangkauan mulai dari '- 838:59:59' s/d '838:59:59'.
Year	Ukuran 1 byte. Data tahun antara 1901 s/d 2155.

**Tabel 2.10.** Nilai Kosong pada Tipe Date dan Time

Type	Jika bernilai kosong
Datetime	'0000 - 00 - 00 00:00:00'
Date	'0000 - 00 - 00'
Timestamp	'0000 - 00 - 00 00:00:00'
Time	'00:00:00'
Year	0000

2.4.2.3.3. Tipe Character

Tipe character adalah tipe yang berhubungan dengan string. Adapun jenis tipe datanya antara lain sebagai berikut.

Tabel 2.11. Tipe Data Character

Type	Jika bernilai kosong
binary, char	Ukuran M byte, $1 \leq M \leq 255$. Data string dengan panjang yang tetap.
varbinary, varchar	Ukuran bersifat fleksibel dan maksimal M byte.
Tinyblob, tinytext	Tipe yang mampu mengendalikan data besar, tergantung seberapa besar kapasitas penyimpanannya.
Blob, text	
Mediumblob, mediumtext	
Longblob, longtext	

Tabel 2.12. Perbedaan Tipe Character Char dan Varchar

Nilai	Char(4)	Ukuran	Varchar(4)	Ukuran
' '	' '	4 byte	' '	1 byte
'ab'	'ab '	4 byte	'ab '	3 byte
'abcd'	'abcd '	4 byte	'abcd '	5 byte
'abcdefgh'	'abcd '	4 byte	'abcd '	5 byte



2.4.2.4. Perintah–perintah SQL Dasar

1. *Select*

Select digunakan untuk membaca data dari suatu tabel atau *view*.

Sintaks:

```
Select namafield1, namafield2 ... from namatabel1 where kriteria
```

Atau

```
Select * from namatabel1 where kriteria
```

2. *Insert*

Digunakan untuk menambahkan data ke dalam tabel.

Sintaks:

```
Insert into namatabel (field1, ...) values ('nilai1', ...)
```

3. *Update*

Digunakan untuk memperbaiki data suatu field dalam suatu tabel.

Sintaks:

```
Update namatabel set field1=nilai, field2=nilai2... where kriteria
```

4. *Delete*

Digunakan untuk menghapus data dari suatu tabel.

Sintaks:

```
Delete from namatabel where kriteria
```

2.4.3. Sekilas Tentang Adobe Dreamweaver CS6

Sulistya (2013:1) menyatakan, *dreamweaver* adalah sebuah editor profesional yang menggunakan HTML untuk mendesain web secara visual dan mengelola situs atau halaman web.

Saat ini terdapat banyak *software* Adobe yang digunakan untuk mendesain suatu web. Versi Adobe Dreamweaver CS6 bukan hanya *software* untuk desain web, tetapi juga memiliki beberapa kemampuan untuk menyunting kode serta pembuatan aplikasi web antara lain JSP, PHP, ASP, XML, dan ColdFusion.



1. Menu Utama

Sistem menu yang terdapat pada Dreamweaver CS6 sangat sederhana dan mudah untuk dipahami karena perintah-perintah yang terdapat pada hampir sebagian besar fungsi menu terdapat juga dalam panel. Dengan demikian, Anda tidak akan menghabiskan banyak waktu dengan melakukan perintah yang terdapat pada menu, tetapi juga banyak bekerja dengan mengundang dan menggunakan perintah-perintah yang terdapat pada panel.

2. Insert Bar

Insert Bar terdiri dari tombol-tombol untuk membuat dan menyiapkan objek seperti tabel, layer, dan gambar. Berikut ini adalah tampilan dari Insert Bar.



Gambar 2.5. Tampilan Insert Bar Dreamweaver CS6

Insert Bar dikategorikan dan digunakan untuk mengatur beberapa pilihan sebagai berikut:

- a. Kategori Comman, digunakan untuk membuat dan menyisipkan objek seperti gambar dan tabel.
- b. Kategori Layout, digunakan untuk menyisipkan tabel, div tag, layer, dan freame.
- c. Kategori Form, berisi tombol-tombol untuk membuat form dan menyisipkan elemen form.
- d. Kategori Text, digunakan untuk menyisipkan beberapa teks dan daftar format tag seperti b, em, p, h1, dan u1.



- e. Kategori Favorites, berguna untuk mengelompokkan tombol-tombol baris Insert yang digunakan dalam satu wadah.

3. Document Toolbar

Document Toolbar berisikan tombol-tombol dan menu pop-up yang menyediakan tampilan berbeda dari jendela dokumen.

Berikut ini adalah tampilan dari Document Toolbar pada Dreamweaver CS6:



Gambar 2.6. Tampilan Document Toolbar

4. Jendela Dokumen

Jendela dokumen merupakan bagian yang digunakan untuk mendesign halaman situs web. Jendela dokumen dapat menyisipkan text, image, serta objek lain yang mendukung pembuatan situs web atau tempat untuk membuat desain web.

5. Panel Group

Panel Group adalah kumpulan panel yang dikelompokkan bersama di bawah satu judul. Kumpulan panel tersebut antara lain Design, Code, Application, dan Files.

6. Tag Selector

Tag Selector terdapat pada bagian bawah jendela dokumen atau pada bagian status bar. Fungsi dari Tag Selector adalah untuk menampilkan hirerarki tag pada Design View yang aktif.

7. Property Inspector

Property Inspector digunakan untuk melihat dan mengubah berbagai property objek dan teks yang terpilih.

2.4.4. Sekilas Tentang XAMPP

Sidik (2012:72) menyatakan, XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket *serverweb* PHP dan databae MySQL yang paling



populer di kalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai databasenya.

Sebelum menjalankan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) yang harus dilakukan adalah memastikan bahwa Apache dan MySQL terkoneksi, dengan cara mengubah tombol *stop* pada action menjadi *start*. Berikut adalah tampilan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) control panel untuk menjalankan Apache dan MySQL.



Gambar 2.7. Tampilan XAMPP Control Panel Application

Berikut ini adalah tampilan awal dari XAMPP:



Gambar 2.8. Tampilan Awal XAMPP