

## BAB 2

### LANDASAN TEORI

#### 2.1. Pengertian Sistem

Beberapa pendapat menurut para ahli yang mendukung tentang pengertian sistem antara lain adalah :

1. Menurut Hall (2001, p5), sistem adalah sekelompok dua atau lebih komponen-komponen yang saling berkaitan (*interrelated*) atau subelemen-subelemen yang bersatu untuk mencapai tujuan yang sama (*common purpose*).
2. Menurut McLeod (2001, p10), sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan.
3. Menurut Lucas (1993, p2), sistem adalah suatu himpunan komponen atau variabel yang terorganisasi, saling berinteraksi, saling bergantung satu sama lain dan terpadu.
4. Menurut Wilkinson (1993, p3), sistem adalah suatu kerangka kerja terpadu yang mempunyai satu sasaran atau lebih. Sistem ini mengkoordinasikan sumber daya yang dibutuhkan untuk mengubah masukan-masukan menjadi keluaran. Sumber daya dapat berupa manusia, bahan, mesin, maupun tenaga surya tergantung pada jenis sistem yang dibicarakan.

## 2.2 Pengertian Informasi

Menurut Mcleod (2001, p15), informasi adalah data yang telah diproses, atau data yang memiliki arti. Informasi merupakan bagian yang penting dari suatu perusahaan.

Menurut Davis (1991, p28) informasi diartikan sebagai data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang. Berdasarkan analogi tersebut, dapat dinyatakan bahwa data sebagai bahan baku dan informasi sebagai bahan jadi.

Menurut Wilkinson (1993, p3), informasi adalah data yang telah ditransformasikan dan dibuat lebih bernilai melalui pemrosesan.

Menurut Cushing (1991, p11), informasi diartikan sebagai *output* pengolahan data yang diorganisir dan berguna bagi orang yang menerimanya.

## 2.3 Pengertian Sistem Informasi Manajemen Sumber Daya Manusia

Menurut HollenBeck (2000, p28), sistem informasi sumber daya manusia adalah sebuah sistem yang memiliki fungsi mengelola, menyimpan, memanipulasi, menganalisa, memanggil kembali dan mendistribusikan informasi yang berhubungan dengan sumber daya manusia perusahaan.

Sistem informasi sumber daya manusia adalah sistem yang digunakan untuk mengumpulkan dan mengelola data yang menjelaskan sumber daya manusia, mengubah data tersebut menjadi informasi dan melaporkan informasi tersebut kepada pemakai (McLeod, 1996, p278).

Menurut Dessler (2003, p15) pada saat perusahaan berkembang, mereka mengintegrasikan sistem-sistem sumber daya manusia mereka yang terpisah-pisah menjadi apa yang disebut dengan sistem informasi sumber daya manusia.

Menurut Laudon (2003, p7), sistem informasi sumber daya manusia adalah komponen-komponen yang berhubungan dan bekerja bersama-sama untuk mengumpulkan, memproses, menyimpan, mendistribusikan informasi untuk mendukung dalam mengambil keputusan, koordinasi, kontrol, analisa, visualisasi dari sebuah aktifitas manajemen sumber daya manusia.

Menurut Riswantini (2005), sistem informasi sumber daya manusia merupakan prosedur sistematis untuk mengumpulkan, menyimpan, mengambil dan memvalidasi data oleh organisasi mengenai sumber daya manusia dan kegiatan-kegiatan personalia.

#### **2.4 Konsep Analisis dan Perancangan**

Menurut Mcleod (2001, p190) analisis sistem (*system analysis*) adalah penelitian atas sistem yang telah ada dengan tujuan untuk merancang sistem baru atau diperbarui. Di dalam tahap analisis sistem, analisis sistem terus bekerja sama dengan manajer, dan komite pengarah SIM yang terlibat dalam titik-titik yang penting. Tahapan-tahapannya antara lain adalah :

1. Mengumumkan penelitian sistem.
2. Mengorganisasikan tim proyek.
3. Mendefinisikan kebutuhan informasi .
4. Mendefinisikan kriteria kinerja sistem.

5. Menyiapkan usulan rancangan.
6. Menyetujui atau menolak rancangan proyek.

Menurut Jogianto (1990, p129) analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian – bagian dan komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Langkah-langkah analisis sistem yang diungkapkan oleh Jogianto (1990, p130) antara lain adalah sebagai berikut :

1. Mengidentifikasi masalah.
2. Memahami kerja dari sistem yang ada.
3. Menganalisa sistem.
4. Membuat laporan hasil analisis.

## **2.5 Manajemen Sumber Daya Manusia**

Menurut Cushing (1991, p6) manajemen SDM didefinisikan sebagai rangkaian, strategi, proses dan aktivitas yang didesain untuk menunjang tujuan perusahaan dengan cara mengintegrasikan kebutuhan perusahaan dan individu.

Menurut Noe (2000, p4) manajemen sumber daya manusia adalah segala kebijakan, kegiatan dan sistem yang mempengaruhi perilaku, sikap, dan kinerja karyawan.

Menurut Hasibuan (1997, p9) manajemen sumber daya manusia mempelajari hubungan dan peranan manusia dalam organisasi perusahaan. Unsur manajemen sumber daya manusia yang merupakan tenaga kerja pada perusahaan. Dengan demikian fokus yang dipelajari hanyalah masalah yang berhubungan dengan tenaga kerja manusia saja.

Menurut Dessler (2003, p2) sumber daya manusia adalah kebijakan dan praktik yang dibutuhkan seseorang untuk menjalankan aspek “orang” atau sumber daya manusia dari posisi seorang manajemen meliputi perekrutan, penyaringan, pelatihan, pengimbalan, dan penilaian.

### **2.5.1 Kegiatan-Kegiatan Dalam Manajemen Sumber Daya Manusia**

Manajemen SDM menurut Cushing meliputi hal-hal berikut :

1. Memberikan saran kepada manajemen tentang kebijaksanaan SDM guna memastikan organisasi memiliki tenaga kerja yang bermotivasi dan berkinerja tinggi, serta dilengkapi dengan sarana untuk menghadapi perubahan dan dapat memenuhi kebutuhan pekerjanya.
2. Melaksanakan dan memelihara semua kebijakan dan prosedur SDM yang diperlukan untuk memastikan pencapaian tujuan organisasi.
3. Membantu perkembangan arah dalam strategi organisasi secara keseluruhan, terutama dengan memperhatikan segi-segi SDM.
4. Meredakan krisis dalam situasi sulit didalam hubungan antar karyawan untuk memastikan tidak adanya gangguan dalam pencapaian tujuan organisasi.

5. Menyediakan sarana komunikasi antara karyawan dengan manajemen organisasi.
6. Menjamin standar dan nilai organisasi dalam pengelolaan SDM.

Dessler (1997, p28) juga mengungkapkan bahwa manajemen sumber daya manusia meliputi hal-hal di bawah ini:

1. Perekrutan dan Penempatan :

- Melakukan analisis jabatan (menetapkan sifat dari pekerjaan masing-masing karyawan) : Bagaimana menganalisa sebuah jabatan, menentukan tuntutan sumber daya manusia atas jabatan, juga tugas dan tanggung jawab spesifiknya.
- Merencanakan kebutuhan tenaga kerja dan merekrut para calon pekerja : Menentukan orang macam apakah yang perlu dipekerjakan dan kemudian merekrut mereka.
- Testing dan seleksi para calon pekerja : Teknik-teknik seperti testing harus memastikan bahwa perusahaan mempekerjakan orang yang tepat.
- Wawancara calon karyawan : Bagaimana mewawancara calon membantu memastikan bahwa perusahaan mempekerjakan orang yang benar pada jabatan yang tepat.

2. Pelatihan dan Pengembangan

- Memberikan orientasi dan pelatihan bagi karyawan baru : Memberikan pelatihan yang perlu untuk memastikan bahwa karyawan baru memiliki pengetahuan dan keterampilan yang dibutuhkan untuk menyelesaikan tugas-tugas mereka.

- Manajemen pengembangan : Konsep dan teknik untuk mengembangkan lebih banyak karyawan, manajer, dan organisasi yang berkemampuan.
- Menata-olah mutu dan produktifitas : Teknik-teknik seperti program perbaikan mutu dan pembangunan tim yang digunakan perusahaan untuk membantu menata-olah mutu dan produktifitas.
- Menilai kinerja : Teknik-teknik untuk menilai kinerja.
- Menata-olah karir : Teknik-teknik seperti perencanaan dan promosi karir dari dalam yang digunakan perusahaan untuk membantu memastikan bahwa karyawan dapat mencapai potensi mereka.

### 3. Kompensasi

- Menetapkan rencana pembayaran : Bagaimana mengembangkan rencana pembayaran yang adil untuk karyawan perusahaan.
- Membayar kinerja dan insentif keuangan : Rencana pembayaran kinerja seperti insentif keuangan, membayar jasa dan insentif yang membantu mengikat kinerja untuk dibayar.
- Kesejahteraan dan jasa : Memberikan kesejahteraan yang memperjelas bahwa perusahaan melihat karyawannya sebagai investasi jangka panjang dan memperhatikan kesejahteraan mereka seperti rencana pemilikan saham, pensiun, dan rencana kesehatan di atas rata-rata.

### 4. Keamanan dan Keselamatan Karyawan

- Hubungan tenaga kerja dan tawar menawar kolektif : Konsep dan teknik menyangkut hubungan antara serikat kerja dan manajemen, termasuk kampanye pengorganisasian-serikat kerja, menegosiasikan dan

menyetujui kesepakatan tawar menawar kolektif antara serikat kerja dan manajemen, dan selanjutnya menata-olah kesepakatan lewat proses keluhan.

- Terjaminnya perlakuan yang adil : Menjamin aliran komunikasi dua arah dalam organisasi, juga menjamin perlakuan adil seperti berhubungan dengan disiplin, pemecatan, dan keamanan kerja karyawan.
- Keselamatan dan kesehatan karyawan : Sebab-sebab kecelakaan, bagaimana membuat tempat kerja aman, dan hukum-hukum yang mengatur tanggung jawab perusahaan sehubungan dengan keselamatan dan kesehatan karyawan.

#### 5. Manajemen Sumber Daya Manusia Internasional

- Menata-olah sumber daya manusia dalam arena internasional : Semakin pentingnya bisnis internasional, dan peran SDM dalam menata olah sisi personil dari operasi multinasional.

Peranan kegiatan manajemen sumber daya manusia dalam dunia bisnis yang kompetitif dapat dilihat dari Gambar 2.1.



**Gambar 2.1 Peranan Manajemen Sumber Daya Manusia dalam  
Membangun Perusahaan Kompetitif**

### 2.5.2 Pembagian Tanggung Jawab SDM Antara Lini dan Staf

Menurut Dessler (1997, p5) dalam organisasi-organisasi kecil, manajer lini (manajer yang diberi wewenang untuk mengarahkan pekerjaan bawahan dan bertanggung jawab untuk mencapai tujuan-tujuan organisasi) bisa menjalankan tugas-tugas personal tanpa bantuan. Namun, ketika organisasi bertumbuh, dibutuhkan asisten, pengetahuan khusus, dan nasihat dari sebuah staf sumber daya manusia sendiri. Dalam melakukan hal tersebut, manajemen SDM menjalankan tiga fungsi yang berbeda :

1. Fungsi Lini : Mengarahkan kegiatan dari orang-orang di dalam departemennya sendiri dan bidang-bidang jasa (seperti kafeteria pabrik).
2. Fungsi Koordinatif : Mengkoordinasi kegiatan personil SDM.
3. Fungsi (jasa) Staf : Melayani dan membantu manajer lini.

## 2.6 e-HRM

Menurut Biesalski (2003) e-HRM adalah singkatan dari *Electronic Human Resource Management*. e-HRM adalah alat bantu berbasis web untuk mengotomatisasikan dan mendukung proses manajemen sumber daya manusia. Implementasi dari e-HRM adalah sebuah kesempatan untuk memudahkan pemasukan data oleh karyawan. e-HRM memberikan fasilitas untuk pemasaran sumber daya manusia (*e-Recruitment*) dan menawarkan fitur kepada karyawan untuk melayani diri sendiri dalam bekerja (*selfservice*). (Anonymous, 2002)

e-HRM juga dapat diidentifikasi sebagai data, transaksi, dan alat bantu yang digunakan perusahaan untuk membantu mengatur dan memberikan layanan kepada pelamar, karyawan, dan mantan karyawan.

Bidang-bidang yang dapat dikaji dalam e-HRM umum adalah :

- Perekrutan.
- Seleksi.
- Manajemen kinerja.
- Kompensasi.
- Keuntungan.
- Pelatihan dan pengembangan.
- Survei dan pencarian sumber daya manusia.
- Layanan diri sendiri (*self service*) karyawan dan manajer.

Menurut Mayer (2003) dalam praktiknya e-HRM mempunyai beberapa keuntungan strategik, yaitu :

- Meningkatkan efisiensi layanan sumber daya manusia bagi pelanggan internal (karyawan) dan mengurangi biaya rutin administrasi.
- Spesialis sumber daya manusia mempunyai waktu yang cukup untuk memfokuskan diri pada permasalahan-permasalahan dan layanan sumber daya manusia yang strategis.
- Karyawan akan dianggap sebagai manusia dewasa dan akan berinteraksi dengan perusahaan dan rekan kerjanya secara lebih dekat dan mudah.
- Menghasilkan lingkungan kerja yang sangat memperhatikan kinerja.
- Menambah nilai perusahaan melalui proses standar yang efisien.

## **2.7 Aplikasi Jaringan**

### **2.7.1 Internet**

Menurut Hahn (1996, p2) internet adalah jaringan global yang berisikan berbagai macam informasi yang menyatukan jaringan-jaringan komputer untuk universitas, perusahaan, instansi-instansi pemerintah, intuisi-intuisi swasta dan sebagainya.

Menurut Septanto (1998, p1) internet secara umum adalah sebuah jaringan *super network* yang terdiri dari kumpulan jaringan yang saling berhubungan satu sama lain dengan menggunakan protokol TCP/IP, dimana jaringan tersebut dapat dengan mudah diakses dari jarak jauh hanya dengan menggunakan saluran telepon lokal.

Menurut Ellsworth (1995, p3) internet (*Interconnection Networking*) adalah suatu jaringan yang luas yang terdiri dari jaringan komputer yang saling berhubungan diseluruh penjuru dunia. Internet terdiri dari ribuan sampai jutaan jaringan komputer yang tersebar diseluruh dunia yang terhubung lewat media seperti: satelit, jalur telepon dan sistem komunikasi lainnya.

Menurut Nugroho (2004,p1) internet adalah suatu media informasi komputer global yang dapat dikatakan sebagai teknologi tercanggih abad ini.

### **2.7.2 Intranet**

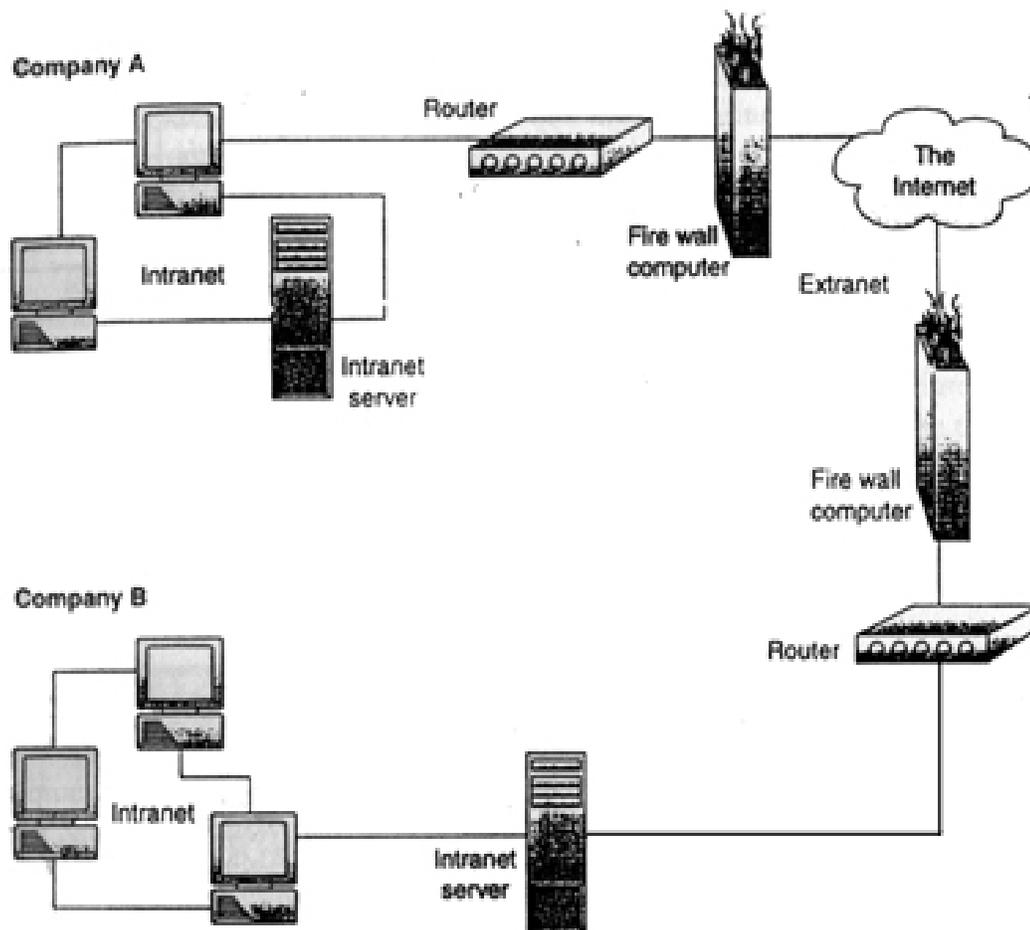
Menurut Basri intranet adalah jejaring internal perusahaan yang menggunakan arsitektur, protocol dan aplikasi yang sama seperti internet. Kesimpulannya bahwa intranet datang setelah internet. Jejaring (informasi)

perusahaan sudah ada beberapa saat lamanya, dan intranet kemudian muncul begitu saja dan banyak dipakai oleh personil dan staf dari perusahaan. Kekuatan dari standar protokol serta teknologi HTML (bahasa *hypertext* yang ditingkatkan) memancing beberapa pakar berkreasi untuk memperkenalkan intranet : yang pada dasarnya mengarah kepada peningkatan pemanfaatan jejaring perusahaan dengan memadukannya bersama internet ke dalam bentuk khusus baru intranet. Standard khusus tersebut adalah : *Internet Protocol Suit* (IPS) pada tataran infrastruktur, *web server* dan teknologi *browser* (pencarian) bagi komunikasi, *retrival* (penggapaian) dan penyajian data, serta HTML (*Hypertext Markup Language*) bagi pembentukan isi (content). Karena semua ini tercakup di dalam daerah bisnis internet, maka *Internet Service Provider* (ISP) adalah yang paling berkepentingan dalam menginvestasikan pengetahuan dan ketrampilan ke dalamnya.

Jelaslah bahwa teknologi yang saling berkait dalam mensukseskan standard tersebut diatas menuntut tingkat disiplin sejenis yang umumnya merupakan bagian dari aktivitas pembinaan jejaring tertentu, maka ISP muncul lagi sebagai penyedia jasa yang tepat. Teknologinya yang mendukung semua ini adalah : *Common Gateway Interface* (CGI) *Scripting*, Java atau bahasa sejenis bagi jaringan ( contoh : Microsoft ActiveX), GIF dan JPEG *image file formats*, format suara dan Video (paket seperti Real Audio dan Macromedia).

Menurut Martin (1999 , p6) perbedaan utama internet dengan intranet adalah sifat informasi yang dikandungnya. Informasi pada intranet tidak bisa ditemukan secara umum seperti informasi pada internet karena informasi tersebut dikembangkan oleh suatu perusahaan atau organisasi untuk kebutuhan internal

saja. Karyawan menerima *password* dan hak akses mereka, kemudian berkomunikasi dengan perusahaan dengan dilindungi sistem *firewall*, yaitu teknologi yang bertujuan untuk membatasi pihak asing dalam mengakses informasi internal perusahaan. Struktur hubungan internet dan intranet dapat dilihat pada Gambar 2.2.

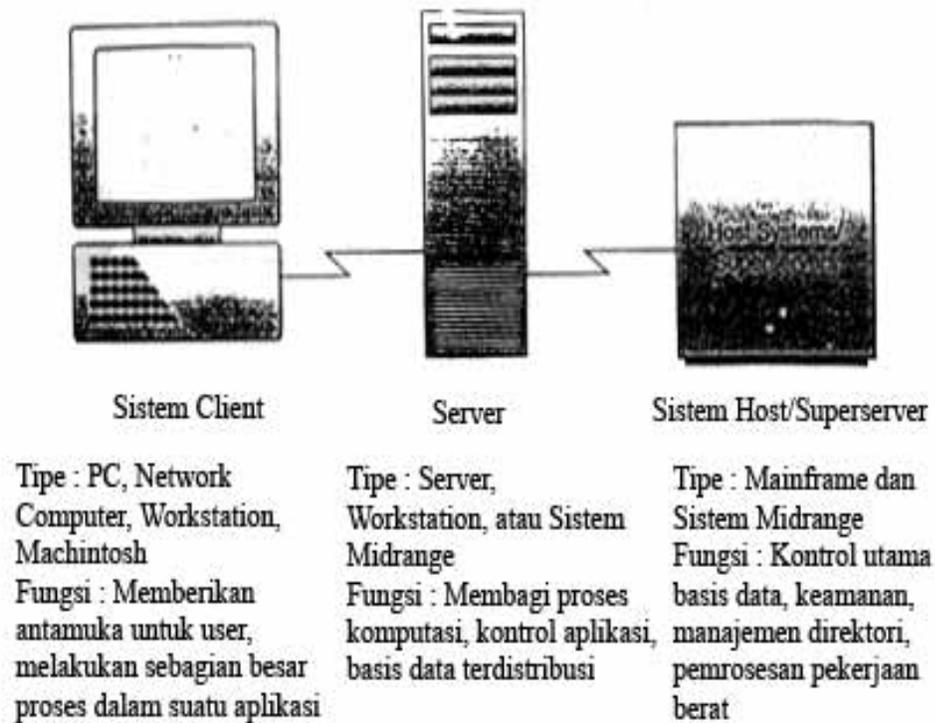


**Gambar 2.2 Struktur Hubungan Internet dan Intranet**

### 2.7.3 Server dan Web Server

Menurut Nugroho (2004, p5) secara global *server* dapat diartikan sebagai pusat dan difungsikan sebagai “pelayan” yang berguna untuk pengiriman data

dan / atau penerimaan data serta mengatur pengiriman dan penerimaan data di antara komputer-komputer yang tersambung atau dengan kata lain *server* berfungsi menyediakan pelayanan terhadap klien.



**Gambar 2.3 Sistem Client Server**

Sedangkan *web server* menurut Nugroho (2004, p6) adalah sebuah bentuk *server* yang khusus digunakan untuk menyimpan halaman *website* atau *home page*. Komputer dapat dikatakan sebagai *web server* jika komputer tersebut memiliki suatu program server yang disebut *Personal Web Server* (PWS). PWS ini difungsikan agar halaman *web* yang ada di dalam sebuah komputer *server* dapat dipanggil oleh komputer klien. Bentuk hubungan *server, client* dan *host* dapat dilihat pada Gambar 2.3.

Macam-macam *web server* :

- Apache (*Open Source*)
- Xitami
- IIS
- PWS

#### 2.7.4 WWW (*World Wide Web*)

*World Wide Web* merupakan sebuah sistem dari *server* internet yang mendukung khususnya dokumen-dokumen yang sudah diformat. Dokumen yang diformat dalam sebuah *script* disebut HTML yang mendukung adanya hubungan dengan dokumen lain, sehingga *user* dapat melompat dari satu halaman dokumen ke halaman lain dengan meng-klik pada *hotspot*. WWW juga dapat diartikan sebagai sekelompok dokumen multimedia yang saling terhubung menggunakan *hypertext link*. (Anonymous,2002).

Menurut McLeod (2001, p75) *World Wide Web* adalah ruang informasi di internet tempat dokumen-dokumen hypermedia disimpan dan dapat diambil melalui suatu skema alamat yang unik. Internet menyediakan arsitektur jaringan, dan *web* menyediakan metode untuk menyimpan dan mengambil dokumen-dokumennya.

#### 2.7.5 HTML (*Hypertext Markup Language*)

HTML adalah blok pembangun yang fundamental dari WWW (*World Wide Web*). *Hypertext* adalah sebuah bentuk dari teks yang mengandung relasi ke

teks lainnya, sehingga menghubungkan teks satu sama lain secara tidak linear. HTML adalah suatu bentuk bahasa yang tidak memiliki format khusus yang dikembangkan oleh W3C, berdasarkan SGML (*standard generalized markup language*). Dokumen *hypertext* adalah teks file standar yang mengandung kode-kode yang dibuat untuk penandaan (*markup*) logis. Dokumen-dokumen ini dapat dibuat dengan menggunakan editor-editor teks sederhana atau alat bantu yang memiliki tingkat kompleksitas yang tinggi.

HTML bukan sebuah bahasa pemrograman, tetapi sebuah bahasa penandaan (*markup*). HTML hanya menggambarkan struktur logis dari sebuah dokumen dibandingkan presentasi dari dokumen.

Dokumen-dokumen HTML disimpan di dalam *web server*, yang mendistribusikan file-file ke *web browser* yang terhubung ke *server* tersebut. File-file HTML kemudian ditampilkan kepada pengguna seperti apa adanya, dan juga diinterpretasi oleh *web browser*, yang menciptakan halaman *web* menurut instruksi-instruksi yang merupakan bagian dari kode HTML (Amor, 2002, p429).

HTML mudah dipelajari dan diimplementasikan, sehingga dapat digunakan oleh perusahaan manapun untuk mempresentasikan dirinya secara multimedia dan sangat interaktif. HTML juga dapat mengurangi biaya secara signifikan. Hal ini juga dapat mengurangi penggunaan kertas yang digunakan untuk mencetak tulisan (*printing*) (Amor, 2002, p434).

### 2.7.6 Web Browser

*Web Browser* merupakan *software* yang dapat membaca dan menginterpretasikan dokumen HTML (Ellsworth, 1995, p32).

Pemakaian *World Wide Web* menuntut pemakaian program *browser* untuk melihat file, melihat atau men-*download image* dan *movie*, atau mendengarkan atau men-*download* file-file suara. *Browser* dapat mengakses file HTML dan juga file jenis lain yang menggunakan *File Transfer Protocol* (FTP), Gopher, *Net News Transfer Protocol* (NNTP), Telnet, dan piranti lainnya. Beberapa *browser* yang terkenal: Microsoft Internet Explorer, Netscape, Mosaic, Mozilla, dan Opera.

### 2.7.7 TCP/IP

TCP/IP (*Transmission Control Protocol / Internet Protocol*) adalah sebuah protokol sistem telekomunikasi yang sangat luas digunakan untuk arsitektur jaringan internet. TCP/IP terdiri dari lima lapisan / *layer* protokol, yaitu protokol aplikasi, *host-to-host transport layer*, protokol internet (IP), antarmuka jaringan, dan *layer* fisik. TCP/IP digunakan oleh internet dan juga oleh semua intranet dan ekstranet. Banyak perusahaan dan organisasi lainnya yang mengkonversi jaringan *client / server-nya* menjadi TCP/IP (O'Brien, 2000, p158).

### 2.7.8 Enkripsi

Enkripsi adalah proses perubahan data digital dengan menggunakan algoritma matematis khusus atau kunci menjadi kode acak sebelum data tersebut

dikirim. Enkripsi data telah menjadi metode penting untuk memproteksi data dan sumber daya jaringan komputer lainnya khususnya di internet, intranet dan ekstranet. Password, pesan, file dan data lainnya dapat dienkripsi menjadi data acak (Obrien, 2000, p474).

## 2.8 Perancangan Database

### 2.8.1 Database

(Connolly, 2002, p14) *Database* adalah suatu sistem penyimpanan data yang tersusun atas sekumpulan data-data yang secara logika saling terkait yang dirancang untuk memenuhi kebutuhan informasi perusahaan. Model *database* relasional adalah sistem yang paling banyak digunakan karena struktur logikalnya yang sederhana. Pada model relasional, seluruh data disusun secara logikal dalam relasi-relasi atau tabel. Setiap relasi terdiri dari baris dan kolom, dan kolom dari relasi yang diberi nama tertentu disebut atribut. Sedangkan baris dari relasi disebut *tuple* dan setiap *tuple* (baris) memiliki satu nilai untuk setiap atribut. *Database* yang tabel-tabelnya saling terhubung dikatakan memiliki relasi. Karena tidak ada relasi yang memiliki dua *tuple* yang sama, maka setiap baris dapat diidentifikasi secara unik dengan menggunakan *primary key*. Munculnya sebuah atribut dalam beberapa relasi dapat merepresentasikan hubungan antar *tuple* dari relasi-relasi tersebut.

Menurut McLeod (2001, p258) *database* adalah suatu koleksi data komputer yang terintegrasi, diorganisasikan, dan disimpan dengan suatu cara yang memudahkan pengambilan kembali Dua tujuan utama dari konsep *database*

adalah meminimumkan pengulangan data dan mencapai independensi data. Pengulangan data (*data redundancy*) atau duplikasi data artinya data yang sama disimpan dalam beberapa file. Sedangkan independensi data adalah kemampuan untuk membuat perubahan dalam struktur data tanpa membuat perubahan pada program yang memproses data. Independensi data dicapai dengan menempatkan spesifikasi data dalam tabel dan kamus yang terpisah secara fisik dari program. Program mengacu pada tabel untuk mengakses data. Perubahan pada struktur data hanya dilakukan sekali, yaitu dalam tabel. Ketika perusahaan mengadopsi konsep *database*, hirarki data menjadi:

- *Database*
- File
- Catatan
- Elemen Data

File-file tersendiri dapat tetap ada, mewakili komponen-komponen utama dari *database*, namun organisasi fisik dari data tidak menghambat pemakai. Tersedia berbagai cara untuk mengintegrasikan isi dari file-file yang memiliki hubungan logis.

Pemakai *database* dapat berupa orang atau program aplikasi. Orang biasanya menggunakan *database* dari terminal dan mengambil data dan informasi dengan menggunakan *query language*. *Query* adalah permintaan informasi dari *database*, dan *query language* adalah bahasa khusus yang *user-friendly* yang memungkinkan komputer menjawab *query*.

### 2.8.2 Database Language

Suatu *data sublanguage* terdiri dari dua bagian DDL (*Data Definition Language*) dan DML (*Data Manipulation Language*). DML biasanya digunakan untuk menetapkan bagan suatu *database* dan DML digunakan untuk keduanya yaitu untuk membaca dan memperbaharui *database* tersebut. Bahasa ini disebut *data sublanguages* sebab mereka tidak meliputi semua kebutuhan perhitungan seperti kondisi pernyataan atau iteratif pernyataan yang menyajikan bahasa pemrograman tingkat tinggi (Connolly, 2001, p40).

### 2.8.3 DDL (Data Definition Language)

DDL merupakan suatu bahasa yang memperbolehkan DBA (*Database Administrator*) atau *user* untuk membuat dan memberi nama suatu entiti, atribut, dan hubungan (*relationship*) yang dibutuhkan oleh suatu aplikasi, digunakan bersama-sama serta mempunyai hubungan yang terintegrasi dan batasan-batasan keamanan (*security constraints*).

Bagan *database* ditetapkan oleh sekumpulan set definisi yang dinyatakan dalam suatu bahasa khusus dikenal sebagai *Data Definition Language* (Connolly, 2001, p40).

### 2.8.4 DML (Data Manipulation Language)

DML merupakan suatu bahasa yang menyediakan sekumpulan set dari beberapa operasi yang mendukung dasar dari operasi-operasi manipulasi pada sebuah data yang terdapat dalam *database*.

Operasi–operasi dari *Data Manipulation Language* terdiri dari (Connolly, 2001, p41)

- Memasukkan data baru ke dalam suatu *database*.
- Memodifikasi suatu data yang telah tersimpan dalam suatu *database*.
- Mengakses suatu data yang terdapat dalam suatu *database*.
- Menghapus suatu data yang terdapat dalam suatu *database*.

## 2.9 Alat Bantu Aplikasi Berbasis Web

### 2.9.1 PHP

Menurut Bunafit (2004, p139) PHP adalah singkatan dari PHP *Hypertext Preprocessor*. PHP merupakan bahasa program yang berbentuk *script* yang diletakkan di dalam *server web*. PHP telah diciptakan terutama untuk kegunaan *web* dan dapat menghubungkan *query database* serta menggunakan perintah-perintah sederhana / *simple task* yang dapat diluruskan dalam 3 atau 4 baris kode saja. PHP adalah bahasa pemrograman yang baru dibangun sekitar tahun 1994 / 1995. PHP dapat menggantikan *static website* yang menggunakan HTML ke *dynamic web pages* yang berfungsi secara otomatis seperti ASP, CGI dan sebagainya.

Keuntungan menggunakan PHP adalah sebagai berikut :

- PHP merupakan program *open source* (tidak memerlukan biaya lisensi)
- *Multi-platform* (dapat dijalankan pada sistem operasi yang berbeda-beda)
- Adanya penggunaan *session* / sesi

## 2.9.2 MySQL

Menurut Allen dan Hornberger (2002, p220) MySQL merupakan bahasa pemrograman *open-source* yang paling populer dan banyak digunakan di lingkungan Linux. Kepopuleran ini karena ditunjang oleh performansi *query* dari *database*-nya yang jarang bermasalah.

Nugroho (2004, p29) mengemukakan, MySQL (*My Structure Query Language*) adalah sebuah program pembuat *database* yang bersifat *open source*, artinya siapa saja dapat menggunakannya secara bebas.

MySQL sebenarnya produk yang berjalan pada *platform* Linux. Karena sifatnya yang *open source*, MySQL dapat berjalan pada semua *platform* baik Windows maupun Linux. Selain itu, MySQL juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multi-user* (banyak pengguna). Saat ini *database* MySQL telah digunakan hampir oleh semua pemrogram *database*, terlebih dalam pemrograman *web*.

Kelebihan lain dari MySQL adalah penggunaan bahasa *query* yang dimiliki SQL (*Structured Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur dan telah distandarisasi untuk semua program pengakses *database* seperti Oracle, PostgreSQL, SQL Server, dan lain-lain.

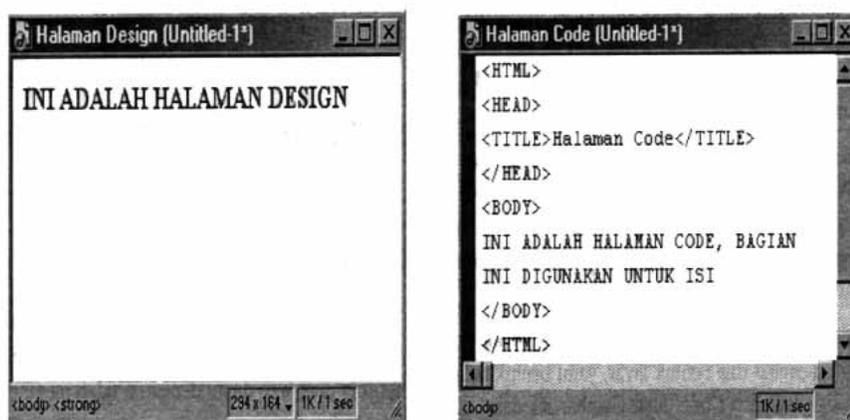
Sebagai sebuah program penghasil *database*, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain (*interface*). MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open source* seperti PHP maupun yang tidak, yang ada pada *platform* Windows seperti Visual Basic, Delphi, dan lainnya.

### 2.9.3 Macromedia Dreamweaver MX

Dreamweaver MX adalah suatu bentuk program editor *web* yang dibuat oleh Macromedia. Dengan program ini seorang *programer web* dapat dengan mudah membuat dan mendesain *web*-nya.

Sebagai editor Dreamweaver MX mempunyai sifat WYSIWYG, artinya apa yang kamu lihat akan kamu peroleh (*What You See Is What You Get*). Dengan kelebihan ini, seorang *programer* dapat langsung melihat hasil buaatannya tanpa harus dibuka di *browser*.

Seperti program editor-editor *web* lain, Dreamweaver MX juga memiliki dua bentuk layar, yaitu bentuk halaman *Design* dan halaman *Code* (dapat dilihat pada Gambar 2.4). Hal ini akan mempermudah *programer* dalam menambahkan *script* berbasis PHP maupun JavaScript. Dreamweaver MX selain mendukung pembuatan web yang berbasis HTML (*Hyper Text Markup Language*), juga dapat mendukung program-program *web* yang lain diantaranya PHP, ASP, Perl, JavaScript, dan lain-lain. (Nu groho, 2004, p91).



**Gambar 2.4** Layar *Design* dan *Coding*

**Dreamweaver MX**

## 2.10 System Development Life Cycle (SDLC)

Menurut O'brien (2000, p383) *System Development Life Cycle* (SDLC) adalah aplikasi penerapan dari penemuan permasalahan (*problem solving*) yang didapat dari pendekatan sistem (*system approach*) menjadi pengembangan dari solusi sistem informasi terhadap masalah bisnis. Tahapan SDLC dapat dilihat pada Gambar 2.5.

Menurut Turban (2000, p477) *System Development Life Cycle* (SDLC) atau Siklus Hidup Pengembangan Sistem adalah metode pengembangan sistem tradisional yang digunakan sebagian besar organisasi saat ini. SDLC adalah kerangka kerja (*framework*) yang terstruktur yang berisi proses-proses sekuensial di mana sistem informasi dikembangkan.

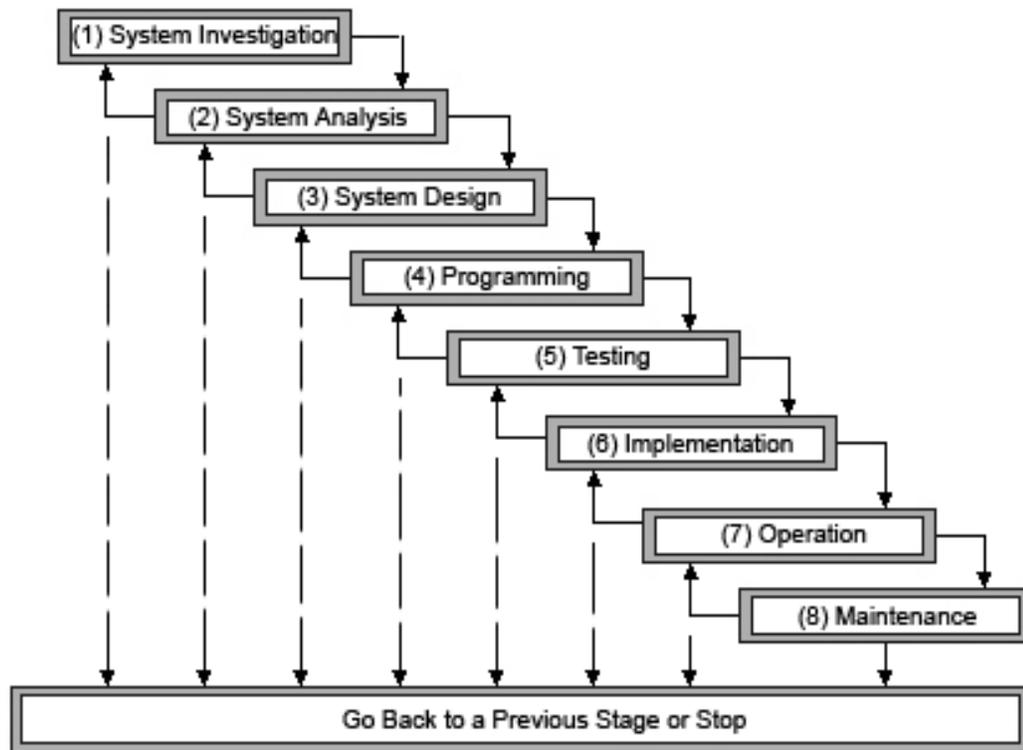
Pada masa lalu, pendekatan *waterfall* digunakan di mana tugas dalam satu tahap harus diselesaikan secara sempurna sebelum pekerjaan berlanjut ke tahap berikutnya. Tetapi sekarang, perancang sistem dapat berjalan maju dan mundur di semua tahap selama hal itu diperlukan.

Tahap-tahap dalam SDLC adalah : 1. Investigasi sistem 2. Analisa sistem 3. Perancangan sistem 4. Pemrograman 5. Testing 6. Implementasi 7. Operasi 8. Perawatan.

Berikut adalah detil dari masing-masing tahap :

1. Investigasi sistem : Berisi studi kelayakan. Studi kelayakan digunakan untuk menentukan kemungkinan suksesnya proyek pengembangan sistem yang diajukan dan menentukan kelayakan teknis, ekonomi, dan perilaku proyek. Studi kelayakan dibagi atas tiga tahap, yaitu :

- Kelayakan teknis : Menentukan apakah *hardware*, *software*, dan komponen-komponen komunikasi dapat dikembangkan atau didapat untuk memecahkan permasalahan bisnis, serta menentukan apakah teknologi yang dimiliki perusahaan dapat memenuhi objektifitas kinerja proyek.



**Gambar 2.5 SDLC (System Development Live Cycle)**

- Kelayakan ekonomi : Menentukan apakah proyek adalah resiko keuangan yang dapat diterima dan apakah organisasi dapat membiayai pengeluaran dan waktu yang dibutuhkan untuk menyelesaikan proyek.
  - Kelayakan perilaku : Berhubungan dengan isu-isu manusia pada proyek. Semua proyek pengembangan sistem memberikan perubahan di dalam organisasi, dan manusia biasanya takut akan perubahan.
2. Analisa sistem : Analisa sistem adalah penentuan permasalahan bisnis yang ingin diselesaikan oleh organisasi dengan sistem informasi. Tahap ini

menentukan permasalahan bisnis, mengidentifikasi sebab-sebabnya, menentukan solusi, dan mengidentifikasikan kebutuhan informasi yang akan digunakan untuk memenuhi solusi. Analisa sistem menghasilkan beberapa hal di bawah ini :

- Kekuatan dan kelemahan dari sistem yang telah ada.
  - Fungsi-fungsi yang harus dimiliki sistem baru untuk memecahkan permasalahan bisnis.
  - Kebutuhan informasi pengguna (*user*) untuk sistem yang baru.
3. Perancangan sistem : Perancangan sistem menggambarkan bagaimana sistem dapat memenuhi tugasnya. Secara umum tahap perancangan sistem terbagi atas dua bagian :

- Perancangan spesifikasi logika : Menyatakan apa yang akan dilakukan sistem. Perancangan spesifikasi logika meliputi keluaran (*output*), masukan(*input*), antarmuka pemakai (*user interface*), proses, *database*, telekomunikasi, kontrol, keamanan dan tugas IS (sistem informasi).
- Perancangan spesifikasi fisik: Menyatakan bagaimana sistem akan menjalankan fungsi-fungsinya. Perancangan spesifikasi fisik meliputi *hardware*, *software*, *database*, alat-alat telekomunikasi, personil dan prosedur.

Dengan demikian produk-produk yang dihasilkan pada tahap ini adalah perancangan :

- Keluaran (*output*), masukan (*input*), dan antarmuka pemakai (*user interface*) sistem.

- *Hardware, software, database*, alat-alat telekomunikasi, personil dan prosedur.
  - Bagaimana komponen-komponen di atas dapat diintegrasikan.
4. Pemrograman : Pemrograman meliputi translasi atau penerjemahan dari perancangan spesifikasi ke dalam kode komputer.
  5. *Testing* : *Testing* bertujuan untuk melihat apakah kode komputer akan memberikan hasil yang diinginkan dan diharapkan di dalam kondisi tertentu. *Testing* dirancang untuk mendeteksi kesalahan-kesalahan di dalam kode komputer. Kesalahan-kesalahan ini dibagi ke dalam dua tipe:
    - Kesalahan Sintaks (seperti kesalahan pengetikan kata-kata atau kesalahan penempatan tanda koma), mudah ditemukan dan tidak perlu dicari dengan menjalankan program.
    - Kesalahan Logika, mengharuskan program untuk dijalankan, tetapi hasilnya keluarannya salah.
  6. Implementasi : Implementasi adalah proses konversi dari sistem lama ke dalam sistem yang baru. Organisasi menggunakan empat jenis strategi konversi :
    - Konversi Parallel : Sistem lama dan sistem baru dijalankan secara simultan dalam periode tertentu.
    - Konversi Langsung (*Direct*) : Sistem lama dihentikan dan sistem baru dijalankan pada periode tertentu.

- Konversi Pilot : Sistem baru dijalankan hanya di satu bagian tertentu dari organisasi. Setelah sistem baru berjalan baik, sistem akan diperkenalkan ke bagian yang lain dari organisasi.
  - Konversi Fase : Konversi fase memperkenalkan komponen-komponen dari sistem baru, seperti modul-modul individual, dalam beberapa tahapan. Setiap modul dijalankan, dan jika modul telah berjalan baik, modul lain diperkenalkan sampai seluruh sistem beroperasi.
7. Operasi : Setelah konversi, sistem baru akan dijalankan dalam periode waktu tertentu sampai sistem itu tidak lagi sesuai dengan kondisi tertentu. Setelah sistem baru distabilkan, akan dilakukan audit selama sistem dijalankan untuk memperlihatkan kemampuan sistem dan menentukan apakah sistem telah digunakan secara benar.
8. Perawatan (*Maintenance*) : Perawatan sistem terbagi atas tiga tipe :
- Melakukan *debugging* program, yaitu proses yang secara berkesinambungan dijalankan dalam masa hidup sistem.
  - Memperbaharui (*updating*) sistem untuk memenuhi perubahan kondisi bisnis.
  - Menambahkan fungsi baru pada sistem. Proses ini meliputi penambahan fitur baru pada sistem yang ada tanpa mengganggu kerjanya.

## 2.11 Delapan Aturan Emas Perancangan *User Interface* (Antar Muka Pemakai)

Menurut Shneiderman (1998, p74), ada delapan aturan emas dalam perancangan antar muka (*interface*) yang harus dipenuhi yaitu :

1. Berusaha untuk tetap konsisten : Urutan kejadian harus konsisten pada situasi-situasi yang serupa. Menu, layar dan perintah harus memiliki terminologi yang identik. Perancang juga harus menjaga konsistensi warna, *layout*, kapitalisasi, jenis huruf, dan sebagainya.
2. Memungkinkan *frequent user* untuk menggunakan jalan pintas (*shortcut*) : Umumnya *user* yang sering menggunakan aplikasi ingin kecepatan respon yang tinggi. Untuk itu penggunaan abrasi, kunci spesial dan perintah tersembunyi sangat dibutuhkan.
3. Menawarkan umpan balik yang informatif : Untuk setiap aksi dari *user* sistem harus memberikan umpan balik yang informatif. Untuk aksi minor, respon bersifat lebih umum. Untuk aksi mayor, respon bersifat lebih substansial.
4. Rancang dialog untuk menghasikan keadaan akhir : Umpan balik atas akhir dari suatu proses dan aksi akan sangat membantu dan juga *user* akan mendapat sinyal untuk melakukan aksi lainnya.
5. Memberikan pencegahan kesalahan dan penanganan kesalahan yang sederhana : Sedapat mungkin sistem didesain sedemikian rupa agar *user* tidak sering melakukan kesalahan yang serius, jika *user* melakukan kesalahan sistem harus dapat mendeteksi kesalahan dan menawarkan instruksi yang sederhana, konstruktif dan spesifik untuk penyembuhan (*recovery*).
6. Mengizinkan pembalikan (*undo*) aksi yang mudah : Dalam suatu waktu *user* mungkin tidak sengaja melakukan aksi yang tidak diinginkan dan ingin melakukan pembatalan. Sistem harus memberikan fungsi pembatalan

sebanyak mungkin agar *user* merasa nyaman dan tidak takut dalam mencoba dan memakai sistem.

7. Mendukung fokus internal dari suatu kontrol : Kontrol yang kuat sangat baik digunakan pada sistem, sehingga *user* dapat menguasai sistem secara cepat dan mudah. Sistem yang tidak terduga dan sulit dalam melakukan aksi dapat menyulitkan *user*.
8. Mengurangi beban memori jangka pendek : Keterbatasan memory pada manusia harus ditanggulangi dengan meminimalisasi proses penyimpanan memori dengan tampilan yang sederhana, konsolidasi tampilan halaman jamak dan pengurangan frekuensi perubahan *windows*.

## 2.12 Teori Diagram

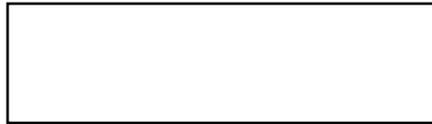
### 2.12.1 Data Flow Diagram (DFD)

(Syahroni, 2003) Untuk memudahkan penggambaran suatu sistem yang ada atau sistem yang baru yang akan dikembangkan secara logika tanpa memperhatikan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan, maka perancang sistem menggunakan diagram aliran data atau *Data Flow Diagram* (DFD).

*Data Flow Diagram* merupakan alat yang cukup populer sekarang, karena dapat menggambarkan aliran data di dalam suatu sistem dengan terstruktur dan jelas. Dalam menggambarkan sistem perlu dilakukan pembentukan simbol, berikut ini simbol-simbol yang sering digunakan dalam *Data Flow Diagram* (DFD) :

### 1. *External entity* (kesatuan luar) atau *boundary* (batasan)

Setiap sistem pasti memiliki batas sistem yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* bagi lingkungan luarnya. Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang berada di lingkungan luarnya yang akan memberikan input serta menerima output dari sistem. Suatu kesatuan luar dapat disimbolkan dengan notasi kotak dapat dilihat pada Gambar 2.6 berikut :



**Gambar 2.6 Notasi *External Entity***

### 2. *Data flow* (aliran data)

Aliran data pada diagram aliran data diberi simbol panah seperti pada Gambar 2.7. Aliran data ini mengalir di antara proses, penyimpanan data dan kesatuan luar. Aliran data ini menunjukkan arus atau aliran data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem dan dapat berbentuk sebagai berikut ini :

- Formulir atau dokumen yang digunakan.
- Laporan tercetak yang dihasilkan oleh sistem.
- Tampilan atau *output* di layar komputer yang dihasilkan oleh sistem.
- Masukan oleh komputer.
- Komunikasi ucapan.
- Surat-surat atau memo.

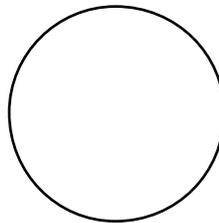
- Data yang dibaca atau direkam pada suatu file.
- Surat isian yang dicatat pada buku agenda.
- Transmisi data dari satu komputer ke komputer yang lain.



**Gambar 2.7 Notasi Aliran Data**

### 3. Proses

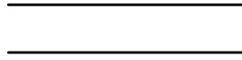
Suatu proses adalah kegiatan atau kerja yang dilakukan orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Untuk *physical data flow diagram* (PDFD), proses dapat dilakukan oleh orang, mesin atau komputer. Sedangkan untuk *logical data flow diagram* (LDFD), suatu proses hanya menunjukkan proses dari komputer. Suatu proses dapat ditunjukkan dengan simbol lingkaran atau dengan simbol empat persegi panjang dengan sudut-sudutnya yang tumpul. Berikut ini Gambar 2.8 untuk simbol untuk proses :



**Gambar 2.8 Notasi Proses**

### 4. Data Store

*Data Store* adalah tempat penyimpanan data yang digunakan sistem. Proses dapat mengambil data dari atau memberikan data ke *data store*. Berikut ini Gambar 2.9 untuk notasi *data store* :



**Gambar 2.9 Notasi Data Store**

### 2.12.2 State Transition Diagram (STD)

*State Transition Diagram* (STD) merupakan sebuah perilaku model yang bergantung pada sekumpulan keadaan sistem, dimana keadaan tersebut adalah setiap modus perilaku sistem yang diamati (Yourdan, 1989, p259-260).

Komponen-komponen utama dalam STD adalah:

1. Keadaan sistem (*System state*)

Penjelasan : Merupakan keadaan yang terjadi di dalam sistem pada waktu tertentu. Keadaan sistem dilambangkan dengan bentuk segi empat seperti Gambar 2.10



**Gambar 2.10 Simbol State**

2. Perubahan keadaan (*Change of state*)

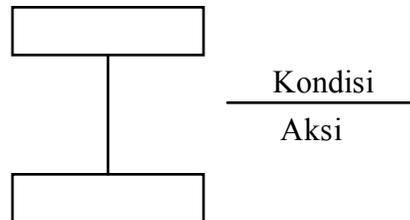
Penjelasan : Perubahan keadaan yang digambarkan dengan garis panah yang menghubungkan 2 keadaan yang berkaitan seperti pada Gambar 2.11.



**Gambar 2.11 Simbol Perubahan State**

3. Kondisi dan aksi (*Condition and action*)

Penjelasan : Kondisi merupakan penyebab suatu keadaan menjadi berubah, sedangkan aksi ialah apa yang dilakukan oleh sistem bila terjadi perubahan keadaan. Aksi juga dapat dikatakan sebagai reaksi terhadap kondisi seperti pada Gambar 2.12.



**Gambar 2.12 Kondisi dan Aksi**

### 2.12.3 Entity Relationship Diagram (ERD)

ERD merupakan diagram hubungan antar entitas dan digunakan untuk mengetahui hubungan dari setiap tabel dalam *database*. Hubungan dalam ERD umum terbagi menjadi 3 tipe, yaitu *one to one*, *one to many* dan *many to many*. Hubungan ini ditentukan oleh *primary key*, *foreign key* dan *candidate key* yang ada pada tabel *database*.