

BAB III

LANDASAN TEORI

3.1 Sistem Informasi

Menurut Joseph (1993:3-4), sistem informasi berasal dari dua kata yaitu sistem dan informasi. Sistem adalah suatu kerangka kerja yang sangat terpadu serta mempunyai satu sasaran atau lebih. Informasi berbeda dengan data, data adalah keadaan yang ada dan belum diproses lebih lanjut, sedangkan informasi adalah data-data yang telah diolah dan bernilai bagi penggunanya. Maka sistem informasi adalah suatu kerangka kerja di mana sumber daya manusia dan teknologi dikoordinasikan untuk mengubah *input* (data) menjadi *output* (informasi) guna mencapai sasaran perusahaan.

3.2 Aplikasi

Menurut Jogiyanto (2005:126), perangkat lunak aplikasi adalah program yang ditulis dan diterjemahkan oleh *language software* untuk menyelesaikan suatu aplikasi tertentu.

Dari pengertian di atas, dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data.

3.3 Data

Menurut Edhy Sutarna (2004:4), data adalah bahan keterangan tentang kejadian-kejadian nyata antara fakta-fakta yang dirumuskan dalam sekelompok lambang tertentu yang menunjukkan jumlah, waktu, dan tindakan.

3.4 Informasi

Menurut Jogiyanto (2005 : 8) Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

3.5 Analisa Sistem

Menurut Jogiyanto (2006) analisa sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan serta hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Tahap analisa sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem.

3.6 Desain Sistem

Setelah tahap analisa sistem dilakukan, maka telah diperoleh gambaran jelas mengenai apa yang harus dikerjakan. Tahapan selanjutnya memikirkan bagaimana membuat sistem yang baru tersebut. Menurut Jogiyanto (1990:197) desain sistem dapat diartikan sebagai berikut:

- a. Tahap setelah analisa dari siklus pengembangan sistem.
- b. Pendefinisian dari kebutuhan fungsional perusahaan.
- c. Persiapan untuk rancang bangun implementasi.
- d. Menggambarkan bagaimana suatu sistem dibentuk.
- e. Berupa gambaran perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan memiliki fungsi.


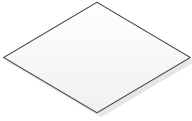
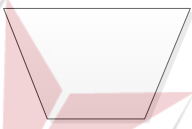

- f. Menyangkut konfigurasi dari komponen-komponen perangkat lunak (*Software*) dan perangkat keras (*Hardware*).

3.6.1 Document Flow

Menurut Sudarmo (2008), *document flow* adalah bagan yang menunjukkan alur dalam program ataupun prosedur sistem secara fisik. Bagan alur digunakan terutama untuk alat bantu komunikasi dan dokumentasi. Bagan alur sistem digambar dengan menggunakan simbol-simbol antara lain sebagai berikut :

Tabel 3.1 Simbol-simbol pada *System Flow*

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
1.		Dokumen	Untuk menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau komputer.
2.		Proses Komputerisasi	Menunjukkan kegiatan dari operasi program komputer.
3.		Database	Untuk menyimpan data.
4.		Penghubung	Menunjukkan hubungan di halaman yang sama.
5.		Penghubung Halaman	Menunjukkan hubungan

No.	Simbol	Nama Simbol <i>Flowchart</i>	Fungsi
		Lain	di halaman lain.
6.		Terminator	Menandakan awal/ akhir dari suatu sistem.
7.		Decision	Menggambarkan logika keputusan dengan nilai <i>true</i> atau <i>false</i> .
8.		Kegiatan Manual	Untuk menunjukkan pekerjaan yang dilakukan secara manual.
9.		Simpanan <i>Offline</i>	Untuk menunjukkan file non-komputer yang diarsip urut angka.

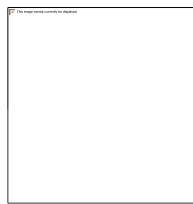
3.6.2 Data Flow Diagram (DFD)

Menurut Kendall (2003: 241), *Data Flow Diagram* menggambarkan pandangan sejauh mungkin mengenai masukan, proses dan keluaran sistem, yang berhubungan dengan masukan, proses, dan keluaran dari model sistem yang dibahas. Serangkaian diagram aliran data berlapis juga bisa digunakan untuk merepresentasikan dan menganalisis prosedur-prosedur mendetail dalam sistem. Prosedur-prosedur tersebut yaitu konseptualisasi bagaimana data-data berpindah di dalam organisasi, proses-proses atau transformasi dimana data-data melalui, dan apa keluarannya. Jadi, melalui suatu teknik analisa data terstruktur yang

disebut *Data Flow Diagram*, penganalisis sistem dapat merepresentasi proses-proses data di dalam organisasi. Menurut Kendall (2003: 265), dalam memetakan *Data Flow Diagram*, terdapat beberapa simbol yang digunakan antara lain:

a. *External Entity*

Suatu *external entity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat.



Gambar 3.1 Simbol *External Entity*

b. *Data Flow*

Data Flow atau aliran data disimbolkan dengan data tanda panah. Aliran data menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses.



Gambar 3.2 Simbol *Data Flow*

c. *Process*

Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan.



Gambar 3.3 Simbol *Process*

d. *Data Store*

Data store adalah simbol yang digunakan untuk melambangkan proses penyimpanan data.

Gambar 3.4 Simbol *Data Store*

3.6.3 *Entity Relational Diagram (ERD)*

Entity relationship diagram (ERD) adalah gambaran pada sistem dimana di dalamnya terdapat hubungan antara *entity* beserta relasinya. *Entity* merupakan sesuatu yang ada dan terdefiniskan di dalam suatu organisasi, dapat abstrak dan nyata. Untuk setiap *entity* biasanya mempunyai *attribute* yang merupakan ciri *entity* tersebut. *Attribute* yaitu uraian dari entitas dimana mereka dihubungkan atau dapat dikatakan sebagai *identifier* atau *descriptors* dari entitas.

Entitas digolongkan menjadi *independent* atau *dependent entity*. *Independent entity* adalah apa yang tidak bersandar pada yang lain sebagai identifikasi. Suatu *dependent entity* adalah apa yang bersandar pada yang lain sebagai identifikasi.

Menurut Marlinda (2004: 28), *attribute* sebagai kolom di sebuah relasi mempunyai macam-macam jenis *attribute* yaitu :

a. *Key Attribute*

Attribute ini merupakan *attribute* yang unik dan tidak dimiliki oleh *attribute* lainnya, misalnya *entity* mahasiswa yang *attribute*-nya NIM.

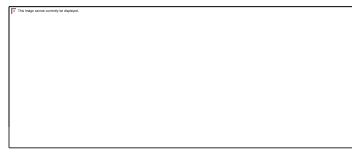


Gambar 3.5 Key Attribute

b. *Particial key Attribute*

Adalah *Attribute* yang tidak menjadi atau merupakan anggota dari *Key*

Primer. Misalnya antara Cabang (toko) dan kode cabang.



Gambar 3.6 Particial Key Attribute

c. *Single Vallue Atribute*

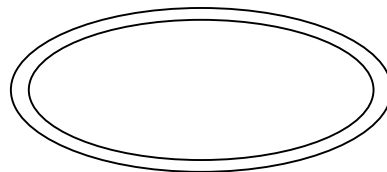
Atribute yang hanya memiliki satu nilai harga, misalnya *entity* mahasiswa dengan *atribute*-nya Umur (Tanggal lahir).



Gambar 3.7 Single Value Attribute

d. *Multi Vallue Atribute*

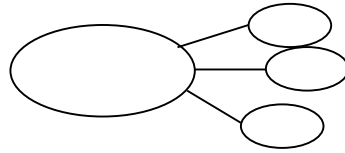
Atribute yang banyak memiliki nilai harga, misalnya *entity* mahasiswa dengan *atribute*-nya pendidikan (SD, SMP, SMA).



Gambar 3.8 Multi Value Attribute

e. *Composite Attribute*

Attribute yang memiliki dua harga, misalnya nama besar (nama kerja) dan nama kecil (nama asli)



Gambar 3.9 *Composite Attribute*

f. *Derived Attribute*

Attribute yang nilai-nilainya diperoleh dari pengolahan atau dapat diturunkan dari table *Attribute* atau table lain yang berhubungan.



Gambar 3.10 *Derived Attribute*

Model *Entity - Relationship* (ER) menurut Peter (1976), sebagai cara untuk mempersatukan pandangan basis data jaringan dan relasional. Langkah sederhana dari model ER adalah model data konseptual yang memandang dunia nyata sebagai kesatuan (*entities*) dan hubungan (*relationship*).

Komponen dasar model merupakan diagram *entity-relationship* yang digunakan untuk menyajikan objek data secara *visual*. *Entity Relationship Diagram* mengilustrasikan struktur logis dari basis data yang mempunyai metodologi sebagai berikut:

Tabel 3.2 Ilustrasi Pembuatan ERD

Proses	Keterangan
1. Menentukan Entitas	Menentukan peran, kejadian, lokasi, hal

Proses	Keterangan
	nyata, dan konsep dimana pengguna akan menyimpan data.
2. Menentukan Relasi	Tentukan hubungan antara pasangan entitas menggunakan matriks relasi.
3. Gambar ERD Sementara	Entitas digambarkan dengan kotak dan relasi dengan garis yang menghubungkan entitas.
4. Isi Kardinalitas	Tentukan jumlah kejadian dari satu entitas untuk sebuah kejadian pada entitas yang berhubungan.
5. Tentukan Kunci Utama	Tentukan atribut yang mengidentifikasi satu dan hanya satu kejadian pada masing-masing entitas.
6. Gambar ERD berdasar Kunci	Hilangkan relasi <i>Many-to-Many</i> dan masukkan <i>primary</i> dan kunci tamu pada masing-masing entitas.
7. Menentukan Atribut	Tuliskan <i>field-field</i> yang diperlukan oleh sistem.
8. Pemetaan Atribut	Pasangkan atribut dengan satu entitas yang sesuai pada masing-masing atribut.
9. Gambar ERD dengan Atribut	Aturlah ERD dari langkah 6 dengan menambahkan entitas atau relasi yang ditemukan pada langkah 8.

Proses	Keterangan
10. Periksa Hasil	Apakah ERD sudah menggambar sistem yang akan dibangun.

Entity Relationship Diagram ini diperlukan agar dapat menggambarkan hubungan antar *entity* dengan jelas, dapat menggambarkan batasan jumlah *entity* dan partisipasi antar *entity*, mudah dimengerti pemakai dan mudah disajikan oleh perancang *database*. Untuk itu, *entity relationship diagram* dibagi menjadi dua jenis model, yaitu:

a. *Conceptual Data model*

Conceptual Data model (CDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara konseptual.

b. *Physical Data Model* (PDM)

Physical Data Model (PDM) adalah jenis model data yang menggambarkan hubungan antar tabel secara fisikal.

3.7 Teori Yang Terkait Dengan Penyelesaian Masalah

3.7.1 Penilaian Kinerja

Penilaian kinerja atau yang biasa disebut *performance appraisal*, pada kamus Manajemen Sumber Daya Manusia (SDM) atau Perilaku Organisasi (Tunggal, 1997) adalah suatu proses organisasi dalam menilai performa individu.

Sedangkan, Blanchard dan Spencer (1982), menyebutkan bahwa penilaian kinerja merupakan proses organisasi yang mengevaluasi prestasi kerja karyawan terhadap pekerjaannya.

Esensinya, manajer atas dan karyawan secara formal melakukan evaluasi secara terus-menerus. Kebanyakan dari mereka mengacu pada prestasi kerja sebelumnya dan mengevaluasi untuk mengetahui apa yang akan dilakukan selanjutnya. Ketika prestasi kerja tidak memenuhi syarat, maka manajer harus mengambil tindakan, demikian juga apabila prestasi kerjanya bagus, maka perilakunya perlu dipertahankan.

3.7.2 *Outsourcing*

Pengertian tentang *outsourcing* terus berkembang dari masa ke masa seiring dengan perkembangan teknologi, perkembangan industri, serta perkembangan pemikiran tentang *outsourcing*.

Outsourcing berasal dari bahasa Inggris yang berarti “alih daya”. *Outsourcing* memiliki nama lain, yaitu *contracting out*, yang merupakan sebuah pemindahan operasi dari satu perusahaan ke tempat lain. Hal ini biasanya dilakukan untuk memperkecil biaya produksi atau untuk memusatkan perhatian kepada hal lain. Di negara-negara maju seperti Eropa dan Amerika, pemanfaatan *outsourcing* sudah sangat mengglobal, sehingga menjadi sarana bagi perusahaan untuk lebih berkonsentrasi pada *core business*-nya dan fokus pada keunggulan produk servis.

Secara umum, *outsourcing* diartikan sebagai pemindahan atau pendelegasian beberapa proses bisnis kepada suatu badan penyedia jasa, di mana badan penyedia jasa tersebut melakukan proses administrasi dan manajemen berdasarkan definisi serta kriteria yang telah disepakati.

Menurut Sehat Damanik (2006), secara sederhana, *outsourcing* dipahami sebagai pendelegasian operasi dan manajemen harian dari suatu proses bisnis kepada pihak luar (perusahaan penyedia jasa *outsourcing*). Melalui pendelegasian, pengelolaan tidak lagi dilakukan oleh perusahaan, melainkan dilimpahkan kepada perusahaan jasa *outsourcing*.

Dari pengertian tersebut, dapat disimpulkan bahwa *outsourcing* adalah suatu bentuk perjanjian kerja antara perusahaan pengguna jasa dengan perusahaan penyedia jasa, di mana perusahaan pengguna jasa meminta kepada perusahaan penyedia jasa untuk menyediakan tenaga kerja yang diperlukan agar bekerja di perusahaan pengguna jasa dengan membayar sejumlah uang dan gaji tetap dibayarkan oleh perusahaan penyedia jasa.

3.8 Konsep Dasar Basis Data

3.8.1 Database

Menurut Yuswanto (2005), *database* merupakan sekumpulan data yang berisi informasi yang saling berhubungan. Pengertian ini sangat berbeda antara *database* relasional dan non-relasional. Pada *database* non-relasional, sebuah *database* hanya sebuah file.

Menurut Marlinda (2004), *database* adalah suatu susunan/ kumpulan data operasional lengkap dari suatu organisasi/ perusahaan yang dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.

3.8.2 Sistem Basis Data

Menurut Marlinda (2004), sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk

menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi/ perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses pengambilan keputusan.

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu perangkat keras (*hardware*), sistem operasi (*operating system*), basis data (*database*), pemakai (*user*), dan aplikasi perangkat lunak yang bersifat opsional.

Adapun beberapa keuntungan dari penggunaan sistem basis data adalah :

- a. Mengurangi redudansi data, yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga pembaruan dilakukan berulang-ulang.
- b. Menjaga konsistensi data.
- c. Keamanan data dapat terjaga.
- d. Integritas dapat dipertahankan.
- e. Data dapat digunakan bersama-sama.
- f. Menyediakan *recovery*.
- g. Memudahkan penerapan standarisasi.
- h. Data bersifat mandiri (*data independence*).
- i. Keterpaduan data terjaga, memelihara data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengontrolan kerangkapan data dan pendidikan keselarasan data.

Kerugian sistem basis data adalah :

- a. Diperlukan tempat penyimpanan yang besar.
- b. Diperlukan tenaga yang terampil dalam mengolah data.
- c. Perangkat lunaknya relatif mahal.

- d. Kerusakan sistem basis data yang dapat mempengaruhi departemen/ bagian yang terkait.

3.8.3 Database Management System (DBMS)

Menurut Merlinda (2004:6), *Database Management System (DBMS)* merupakan kumpulan file yang saling berkaitan dan program untuk pengelolanya. Basis data adalah kumpulan data, sedangkan program pengelolanya berdiri sendiri dalam satu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data.

3.9 Website

Menurut Hidayat (2009), *website* adalah suatu situs atau dapat diartikan sebagai kumpulan-kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam/ gambar gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat *statis* atau *dinamis* yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman *web* dengan halaman *web* lainnya disebut *Hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *Hypertext*.

3.10 HyperText Markup Language (HTML)

Menurut Prasetio (2005), *HyperText Markup Language (HTML)* adalah bahasa yang mempelopori hadirnya *web* dan *internet*. Bahasa ini merupakan bahasa pemrograman yang digunakan oleh sebagian besar situs web yang dikunjungi oleh setiap orang. *HTML* saat ini dikenal oleh hampir semua komputer yang ada di dunia dan merupakan universal untuk membuat sebuah dokumen. *HTML* tidak memiliki variasi format terbaik dan bahkan tidak menjamin bahwa

halaman web yang yang dibuat sama persis di setiap *browser*, tetapi perlu diingat bahwa tanpa *HTML*, tidak akan ada *internet*.

3.11 *Hypertext Preprocessor (PHP)*

Menurut Anhar (2009), *Hypertext Preprocessor (PHP)* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada server. *PHP* adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti dapat disesuaikan sesuai keinginan *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu terbaru. Semua *script PHP* dieksekusi pada *server* di mana *script* tersebut dijalankan.

3.12 *My Structure Query Language (MySQL)*

Menurut Anhar (2010), *My Structure Query Language (MySQL)* adalah salah satu *Database Management System (DBMS)* dari sekian banyak *DBMS* seperti *Oracle*, *MS SQL*, *Postagre SQL*, dan lain-lain. Pemrograman *PHP* juga sangat mendukung dengan penggunaan *database MySQL*.

Keunggulan dari *MySQL* adalah cepat dan mudah digunakan. *MySQL* semula berkembang karena memerlukan *SQL Server* yang dapat mengatasi sebuah perintah *database*.

3.13 *Intranet*

Menurut Johnsen (2004), *intranet* merupakan suatu bentuk pemanfaatan teknologi *internet* dalam sebuah organisasi/perusahaan. Dalam sebuah perusahaan, *intranet* dapat dioperasikan dalam sebuah *Local Area Network*

(LAN), dan untuk skala yang lebih besar, *intranet* dapat dioperasikan pada *Wide Area Network* (WAN).

Untuk membangun *intranet* ada beberapa hal yang harus disediakan, yaitu infrastruktur dan aplikasi. Infrastruktur yang harus disediakan adalah LAN yang menggunakan TCP/IP, dan sebuah komputer server yang akan dijadikan sebagai *web server*, dan aplikasi berbasis *web* yang mendukung semua aktivitas pada *intranet* yang dibuat. Sedangkan pada komputer *client* harus terinstal *web browser*.

