

BAB III

LANDASAN TEORI

Dalam bab ini akan dijelaskan landasan teori yang digunakan untuk mendukung penyusunan laporan kerja praktek ini. Landasan teori yang akan dibahas meliputi permasalahan-permasalahan atau prosedur-prosedur yang berlaku saat ini serta beberapa pengertian tentang ilmu yang berkaitan dengan permasalahan.

3.1 Sistem

Menurut Jogiyanto (1989:23), sistem merupakan kumpulan dari elemen-elemen yang satu dengan yang lain berinteraksi dan bersama-sama beroperasi untuk mencapai tujuan tertentu. Sistem mempunyai peran yang sangat besar dalam menentukan berjalan tidaknya suatu lembaga atau perusahaan. Hal ini dikarenakan setiap perusahaan akan selalu berdasarkan pada suatu sistem dalam menjalankan aktifitas sehari-harinya.

Suatu sistem dapat dirumuskan sebagai suatu totalitas himpunan yang terdiri dari bagian-bagian yang mana antara satu dengan yang lainnya saling berinteraksi dan bersama-sama beroperasi guna mencapai suatu tujuan tertentu didalam suatu lingkungan. Bagian-bagian atau subsistem tersebut merupakan suatu kompleksitas tersendiri, tapi dalam kebersamaan mencapai suatu tujuan berlangsung secara harmonis dalam keteraturan yang pasti.

3.2 Sistem Informasi

Menurut Leitch (1983:86), sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolah

transaksi atau informasi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut Leman (1998:3-4) komponen sistem informasi terdiri dari *hardware*, *software*, data, manusia, dan prosedur. Kegiatan dalam sistem informasi mencakup :

1. *Input*, menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.
2. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
3. *Output*, suatu kegiatan untuk menghasilkan laporan dari proses diatas tersebut.
4. Penyimpanan, suatu kegiatan untuk memelihara dan menyimpan data.
5. *Control*, suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

3.3 Persediaan

Handoko (1999:333) menyatakan bahwa persediaan menunjukkan segala sesuatu atau sumber daya-sumber daya organisasi yang disimpan dalam antisipasinya terhadap pemenuhan permintaan, sedangkan menurut Skousen (2001:360) menyatakan bahwa "persediaan adalah nama yang diberikan untuk barang-barang, baik yang dibuat atau dibeli untuk dijual kembali dalam bisnis normal". Kemudian Ridwan (2002:258) menyatakan bahwa: "persediaan meliputi semua barang atau bahan yang diperlukan dalam proses produksi dan distribusi yang menunggu untuk diproses lebih lanjut atau dijual". Jenis-jenis persediaan menurut Handoko (1999:334) dapat dibedakan atas:

1. Persediaan bahan mentah (*raw materials*), yaitu persediaan barang-barang berwujud seperti baja, kayu, dan komponen-komponen lainnya yang digunakan dalam proses produksi. Bahan mentah dapat diperoleh dari sumber-sumber alam atau dibeli dari para *supplier* dan atau dibuat sendiri oleh perusahaan untuk digunakan dalam proses produksi selanjutnya.
2. Persediaan komponen-komponen rakitan (*purchased parts components*), yaitu persediaan barang-barang yang terdiri dari komponen-komponen yang diperoleh dari perusahaan lain, di mana secara langsung dapat dirakit menjadi suatu produk.
3. Persediaan bahan pembantu atau penolong (*supplier*), yaitu persediaan barang-barang yang diperlukan dalam proses produksi tetapi tidak merupakan bagian atau komponen barang jadi.
4. Persediaan barang dalam proses (*work in process*), yaitu persediaan barang-barang yang merupakan keluaran dari tiap-tiap bagian dalam proses produksi atau yang telah diolah menjadi suatu bentuk tetapi masih perlu dilebih lanjut menjadi barang jadi.
5. Persediaan barang jadi (*finished goods*), yaitu persediaan barang-barang yang telah selesai diproses atau diolah dalam pabrik dan siap untuk dijual dan dikirim kepada langganan.

3.4 Analisis dan Perancangan Sistem

Menurut Kendall (2003:7), analisis dan perancangan sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi. Analisis sistem dilakukan dengan tujuan untuk dapat

mengidentifikasi dan mengevaluasi permasalahan yang terjadi dan kebutuhan yang diharapkan, sehingga dapat diusulkan perbaikannya. Perancangan sistem merupakan penguraian suatu sistem informasi yang utuh ke dalam bagian komputerisasi yang dimaksud, mengidentifikasi dan mengevaluasi permasalahan, menentukan kriteria, menghitung konsistensi terhadap kriteria yang ada, serta mendapatkan hasil atau tujuan dari masalah tersebut serta mengimplementasikan seluruh kebutuhan operasional dalam membangun aplikasi.

3.5 Entity Relationship Diagram (ERD)

Menurut Marlinda (2004:28) *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. Relasi adalah hubungan antar *entity* yang berfungsi sebagai hubungan yang mewujudkan pemetaan antar *entity*.

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:

1. *Conceptual Data Model* (CDM)

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

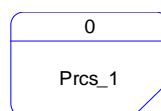
2. *Physical Data Model (PDM)*

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

3.6 Data Flow Diagram (DFD)

Menurut Kristanto (2003:55), *Data Flow Diagram (DFD)* adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersebut disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.

Data Flow Diagram merupakan suatu metode pengembangan sistem yang terstruktur (*structure analysis and design*). Penggunaan notasi dalam *data flow diagram* sangat membantu untuk memahami suatu sistem pada semua tingkat kompleksitas. Pada tahap analisa, penggunaan notasi ini dapat membantu dalam berkomunikasi dengan pemakai sistem untuk memahami sistem secara logika. Di dalam *data flow diagram*, terdapat empat simbol yang digunakan yaitu *process*, *external entity*, *data store*, dan *data flow*. Simbol *process* digunakan untuk melakukan suatu perubahan berdasarkan data yang diinputkan dan menghasilkan data dari perubahan tersebut. Simbol *process* dapat digambarkan sebagai bentuk berikut:



Gambar 3.1 Process

Pada bentuk gambar *process*, bagian atas berisi nomor untuk identitas proses. Suatu proses dengan nomor 0 (nol atau kosong) menandakan bahwa

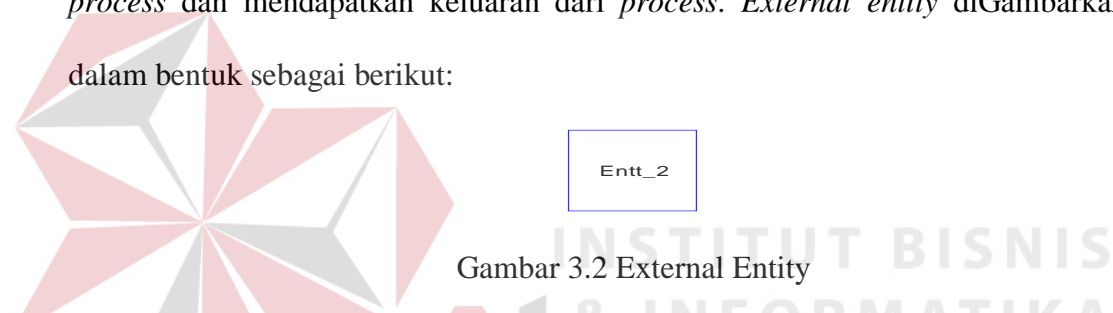
proses tersebut adalah sebuah *context diagram*. Diagram ini merupakan level tertinggi dari DFD yang menggambarkan hubungan sistem dengan lingkungan luarnya. Pembuatan *context diagram* dapat dilakukan dengan terlebih dahulu menentukan nama sistemnya, menentukan batasan dari sistem, dan menentukan terminator yang diterima atau diberikan daripada sistem untuk kemudian dilakukan penggambaran.

Nomor 1, 2, 3, dan seterusnya menandakan bahwa proses tersebut diartikan sebagai proses level-0 (nol) yang merupakan hasil turunan atau *decompose* dari proses *context diagram*. Proses level-0 membahas sistem secara lebih mendetil, baik dipandang dari segi kegiatan dari sebuah bagian, alur data yang ada, maupun *database* yang digunakan didalamnya. Pembuatannya dapat dilakukan dengan cara menentukan proses utama yang ada dalam sistem, menentukan alur data yang diterima dan diberikan masing-masing proses daripada sistem sambil memperhatikan konsep keseimbangan (alur data yang masuk atau keluar dari suatu level harus sama dengan alur data yang masuk dan keluar pada level berikutnya), memunculkan *data store* sebagai sumber maupun tujuan data (*optional*), menggambarkan diagram level-0, menghindari perpotongan arus data, dan melakukan pemberian nomor pada proses utama (nomor tidak menunjukkan urutan proses).

Nomor 1.1, 1.2, 2.1, 2.2, dan seterusnya merupakan sebuah proses turunan atau *decompose* dari proses level-0 yang disebut sebagai proses level-1 (satu). Proses level-1 menggambarkan detil kerja dari sebuah bagian dalam sebuah sistem. Penggambarannya dilakukan dengan cara menentukan proses yang lebih kecil (sub-proses) dari proses utama yang ada di level-0, menentukan apa yang

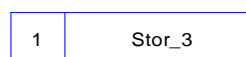
diterima atau diberikan masing-masing sub-proses daripada sistem dan tetap memperhatikan konsep keseimbangan, memunculkan *data store* sebagai sumber maupun tujuan alur data (*optional*), menggambar DFD level-1, dan berusaha untuk menghindari perpotongan arus data. Hasil turunan akhir disebut sebagai *the lowest level*, dimana hasil akhir ini tergantung dari kompleksitas sistem yang ada.

External entity disimbolkan dengan bentuk persegi yang digunakan untuk menggambarkan pelaku-pelaku sistem yang terkait, dapat berupa orang-orang, organisasi maupun instansi. *External entity* dapat memberikan masukan kepada *process* dan mendapatkan keluaran dari *process*. *External entity* digambarkan dalam bentuk sebagai berikut:



Gambar 3.2 External Entity

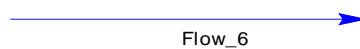
Data store digunakan sebagai media penyimpanan suatu data yang dapat berupa *file* atau *database*, arsip atau catatan manual, lemari *file*, dan tabel-tabel dalam *database*. Penamaan *data store* harus sesuai dengan bentuk data yang tersimpan pada *data store* tersebut, misalnya tabel pelanggan, tabel detail penjualan, tabel detail pembelian, dan lain-lain. *Data store* digambarkan dalam bentuk simbol sebagai berikut:



Gambar 3.3 Data Store

Data flow merupakan penghubung antara *external entity* dengan *process* dan *process* dengan *data store*. *Data flow* menunjukkan aliran data dari satu titik ke titik lainnya dengan tanda anak panah mengarah ke tujuan data. Penamaan *data*

flow harus menggunakan kata benda, karena di dalam *data flow* mengandung sekumpulan data. *Data flow* digambarkan dengan bentuk simbol sebagai berikut:



Gambar 3.4 Data Flow

3.7 Database

Menurut Marlinda (2004:1), *database* adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir/dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya. Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai), masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data).

3.8 Database Management System

Menurut Marlinda (2004:6), *Database Management System* (DBMS) merupakan kumpulan *file* yang saling berkaitan dan program untuk pengelolanya. Basis Data adalah kumpulan datanya, sedang program pengelolanya berdiri sendiri dalam suatu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data. Bahasa-bahasa yang terdapat dalam DBMS adalah:

1. *Data Definition Language (DDL)*

Pola skema basis data dispesifikasikan dengan satu set definisi yang diekspresikan dengan satu bahasa khusus yang disebut DDL. Hasil kompilasi perintah DDL adalah satu set tabel yang disimpan di dalam *file* khusus yang disebut *data dictionary/directory*.

2. *Data Manipulation Language (DML)*

Bahasa yang memperbolehkan pemakai mengakses atau memanipulasi data sebagai yang diorganisasikan sebelumnya model data yang tepat.

3. *Query*

Pernyataan yang diajukan untuk mengambil informasi. Merupakan bagian DML yang digunakan untuk pengambilan informasi.

DBMS memiliki fungsi sebagai berikut:

1. *Data Definition*

DBMS harus dapat mengolah pendefinisian data.

2. *Data Manipulation*

DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data.

3. *Data Security dan Integrity*

DBMS dapat memeriksa *security* dan *integrity* data yang didefinisikan oleh DBA.

4. *Data Recovery dan Concurrency*

a. DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan *disk*, dan sebagainya.

- b. DBMS harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat yang bersamaan.

5. *Data Dictionary*

DBMS harus menyediakan *data dictionary*.

3.9 Microsoft Office Visio 2003

Menurut Rahman (2004:1) Produk Microsoft Visio tahun 2003 adalah *Visio Standart* dan *Visio Professional*. Kedua produk ini memakai *format file* yang umum, yang berarti Anda dapat membagi diagram Gambar Visio Anda dengan pengguna Visio lain, tanpa memperhatikan produk yang Anda digunakan. Visio 2003 merupakan hasil pengembangan dari Visio versi sebelumnya, seperti Visio 2000/2002, Visio 5 dan Visio Technical. Dibandingkan dengan Visio versi sebelumnya, Visio 2003 jauh lebih lengkap dan terintegrasi. Visio 2003 termasuk dalam rangkaian program *Microsoft Office* yang dalam keadaan default terinstal dalam folder instalasi Visio 11 yang berada pada folder *Microsoft Office*.

Visio Standart merupakan solusi Gambar atau diagram yang membantu profesional bisnis, seperti manajer proyek, sales dan pemasaran, person HR, staf administrasi, di dalam memvisualkan ide pekerjaan sehari-harinya. *Visio Profesional* merupakan perangkat yang membantu profesional teknik, seperti spesialis IT, *developer*, dan mahasiswa teknik di dalam memvisualisasikan ide, informasi, dan sistem serta *prototipe* suatu objek. *Visio Profesional* juga menyediakan solusi diagram bisnis.