# BAB II

# LANDASAN TEORI

## 2.1 Sistem

Terdapat dua kelompok pendekatan didalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponennya atau elemennya. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai berikut :

Menurut Warren D. Stallings, Jr dalam buku *fundamentals of system analysis* mendefinisikan sistem sebagai berikut :

> *Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu*[7] .

Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi didalam sistem. Prosedur didefinisikan oleh Richard F. Neuschel dalam buku Management By System sebagai berikut :

> *Prosedur adalah suatu urutan operasi klerikal (tulis-menulis) biasanya melibatkan beberapa orang didalam satu atau lebih departemen, yang diterapkan utuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi* [6].

Pendekatan sistem yang lebih menekankan pada elemennya mendefinisikan sistem sebagai kumpulan elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu

Jadi sistem adalah sekumpulan komponen yang saling berhubungan untuk mencapai tujuan yang sama, dimana sistem menerima input kemudian memprosesnya dan kemudian sistem akan mengeluarkan output.

### 2.1.1 Karakteristik Sistem

Menurut Roger S. Pressman suatu sistem mempunyai karakteristik atau sifat-sifat tertentu [4], yaitu mempunyai :

a. Komponen (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi dan bekerjasama membentuk suatu kesatuan. Komponen-komponen terdiri dari beberapa subsistem, dimana subsistem tersebut memiliki fungsi khusus dan akan mempengaruhi proses sistem secara keseluruhan.

b. Batas Sistem (*Boundary*)

Merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

c. Lingkungan Luar Sistem

Adalah apapun diluar batas dari sistem yang dapat mempengaruhi operasi sistem. Linkungan luar dapat bersifat menguntungkan ataupun merugikan. Lingkungan yang menguntungkan harus tetap dijaga dan dipelihara. Sebaliknya lingkungan yang merugikan harus ditahan dan dikendalikan, jika tidak ingin terganggu kelangsungan hidup sistem.

d. Penghubung (*Interface*)

Merupakan media penghubung antar subsisten yangmemungkinkan sumber daya mengalir dari suatu subsistem lainnya. Keluaran (*Output*) dari suatu subsistem akan menjadi masukan (*Input*) untuk subsistem lainnya melalui penghubung untuk mengintegrasikan subsistem-subsistem tersebut menjadi satu kesatuan.

e. Masukan (*Input*)

Adalah energi yang dimasukkan kedalam suatu sistem yang dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukan perawatan adalah energi yang dimasukan supaya sistem dapat beroperasi, sedangkan masukan sinyal adalah energi yang diproses untuk mendapatkan keluaran. Sebagai contoh dalam sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputer.

f. Keluaran (*Output*)

Adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dari sisa pembuangan. Keluaran dapat merupakan masukan

untuk subsistem yang lain. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil pembuangan, sedangkan informasi adalah keluarkan yang dibutuhkan.

g. Pengolah (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran. Suatu sistem pembelian dan penjualan akan mengolah data-data transaksi menjadi laporan-laporan yang dibutuhkan oleh perusahaan.

h. Sasaran (*Objectivies*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objectivies*). Jika suatu sistem tidak mempunyai suatu tujuan, maka sistem tersebut tidak berguna. Tujuan sistem sangat menentukan bagi masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

## 2.2 Informasi

Informasi sangat erat kaitannya dengan data, informasi berasal dari data. Data adalah hasil, hal, peristiwa atau kenyataan lain apapun yang mengandung sesuatu pengetahuam yang dijadikan dasar guna menyusun keterangan, pembuat kesimpulan atau penetapan. Data adalah ibarat barang mentah yang melalui pengolahan tertentu menjadi informasi, untuk lebih jelasnya didefinisikan oleh Gordon B. Davis dalam bukunya yang di kutip oleh Malayu S. P. Hasibuan.

*Data adalah bahan mentah bagi informasi, dirumuskan sebagai kelompok lambang tidak acak yang menunjukan jumlah-jumlah , tindakan, tindakan, hal-hal, dan sebagainya* [2].

Sedangkan Jogiyanto mendefinisikan informasi sebagai berikut :

*Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya* [1].

Perancangan suatu program aplikasi terdiri dari satu kesatuan sistem. Informasi yang baik adalah informasi yang berkualitas ditentukan oleh hal-hal sebagai beikut :

a. Akurat

   Informasi harus bebas dari kesalahan dan tidak menyesatkan.

b. Tepat pada waktunya.

   Informasi yang datang pada penerima tidak boleh terlambat.

c. Relevan

   Informasi tersebut mempunyai manfaat/hubungan bagi pemakainya.

d. Dapat dipercaya

   Informasi yang diberikan dapat dipercaya kebenarannya dan mempunyai data-data yang lengkap dan jelas sumber-sumber datanya.

## 2.3 Sistem Informasi

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen didalam pengambilan keputusan. Pertanyaanya adalah darimana

informasi tersebut bisa didapatkan?. informasi dapat diperoleh dari sistem informasi (*information systems*) atau disebut juga dengan *processing systems* atau *information processing systems* atau *information-generating systems*. Sistem informasi didefinisikan oleh Robert A. Leitch dan K Roescoe Davis sebagai "*suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi yang menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan*"[5].

### 2.4 *Purchase Order*

*Purchase Order* (PO) didefinisikan oleh wikipedia sebagai sebuah dokumen yang dikeluarkan pembeli kepada penjual [9]. Sebuah *Purchase Order* (PO) biasanya terdiri dari nomor *Purchase Order*, tanggal pengiriman, ketentuan pembayaran, dan list dari produk / *service* beserta jumlah dan harga satuannya.

### 2.5 *Supplier*

*supplier* didefinisikan oleh http://worldnet.priceton.edu sebagai seseorang yang bisnisnya memasok komoditas atau jasa tertentu. Sedangkan www.maxwidemman.com mendefinisikan *supplier* sebagai organisasi yang mensupply barang atau jasa kepada konsumen [8]. Supplier juga dikenal dngan sebutan kontraktor, penjual, *subcontractor*, atau *vendor*.

## 2.6 Oracle

Tiga dekade yang lalu Larry Ellison melihat ada kesempatan emas saat dia mengerjakan deskripsi *protoype database* relasional. Bersama dua orang pendiri lainnya yaitu Bob Miner dan Ed Oates mereka mengkomersilkan model database relasional menggunakan SQL. Saat ini oracle telah menjadi standar khusus untuk pembangunan aplikasi dan database enterprise.

*Oracle Developer* adalah sebuah produk dari oracle yang cukup mudah digunakan untuk membangun aplikasi database. Dengan menggunakan sebuah *application generator* sebagai sebuah *tool* untuk membangun sebuah aplikasi maka waktu yang digunakan untuk menuliskan *statement* program dapat diminimalkan, memberikan waktu lebih banyak untuk membuat prosedur-prosedur yang cukup rumit sebagai pendukung jika perlu dan fleksibel untuk diubah.

Pada *Oracle Developer*, setiap form mengandung satu atau lebih *block*. Data dapat di-*query, insert, update,* dan *delete* didalam sebuah item yang terkorespondensi dengan kolom yang terdapat dalam tabel. Item selalu ditampakan dilayar sehingga *user* dapat berinteraksi dengan data. Standar GUI (*Graphical User Interface*) *Oracle Developer* dapat mempresentasikan data dalam sebuah *interface* dengan *wujud text item, display item, check box*, dan *radio group*.

### 2.6.1 *Oracle Form Builder*

*Oracle Form Builder* adalah aplikasi untuk membuat *form* dalam *oracle developer*. Ketika bekerja dengan *oracle developer*, kita tidak dapat terlepas dari

*Appllication Generator* yang menghasilkan sebuah form. Aplikasi *Form* adalah aplikasi yang mempresentasikan data didalam format online yang bisa terdiri atas satu atau lebih *layout window*.

Sebuah form dalam *oracle* memiliki kemampuan untuk membuat master detail dimana terdapat form sebagai record master dan terdapat form sebagai record detail dari form diatas.

### 2.6.2 *Oracle Report Builder*

*Oracle Report Builder* adalah aplikasi untuk membuat *report* dalam *Oracle Developer*. Sebuah *report* adalah tampilan halaman yang berorientasi menampilkan data dimana *user* dapat memilih data apa yang akan ditampilkan namun *user* tidak dapat mengubah data tersebut.

Tipe-tipe report yang dapat ditampilkan dalam *Oracle Developer* :

1. *Tabular*

   Digunakan untuk menampilkan data dari tabel. Apa yang ada ditabel akan ditampilkan begitu saja.

2. *Mailing Label*

   Digunakan untuk menampilkan record secara berulang-ulang dimasing-masing halaman yang telah kita definisikan.

3. *Form Letter*

Digunakan untuk menampilkan sebuah area kosong yang dapat diisi oleh suatu record yang kita kehendaki. Contohnya nama penerima report dan jabatannya.

4. *Master-Master*

Digunakan untuk menampilkan dua kelompok record atau lebih secara bersamaan dimana tidak ada relasi antara satu dan lainnya.

5. *Master-Detail*

Digunakan untuk menampilkan dua kelompok record atau lebih secara bersamaan dimana terjadi relasi diantar kelompok record tersebut.

6. *Matrix* atau *Crosstab*

Digunakan untuk menampilkan perpotongan antara pertemuan dua record sehingga menampilakn suatu nilai.

7. *Data File*

Digunakan untuk mentransfer data ke tool yang berbeda.

8. *Graph*

Digunakan untuk menampilkan sebuah grafik didalam sebuah report.

### 2.6.3 *SQL *Plus*

SQL *Plus merupakan salah satu *Relational Database Management System* (*RDBMS*) yang terbukti memiliki kinerja sanagat baik. SQL *Plus merupakan bagian dari Oracle . Sesuai namanya SQL *Plus menggunakan bahasa SQL.

Didalam Oracle 9i, kita menuliskan SQL dalam bentuk *command line* untuk berinteraksi dengan *database*.

## 2.7 Basis Data

### 2.7.1 Definisi Basis Data

Basis data didefinisikan sebagai kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan dengan cepat dan mudah.

Tujuan dari desain basis data adalah untuk menentukan data-data yang dibutuhkan dalam sistem, sehingga informasi yang dihasilkan dapat terpenuhi dengan baik. Perancangan *database* yang digunakan adalah untuk memudahkan dalam mengetahui *file-file database* yang digunakan dalam perancangan sistem, sekaligus untuk mengetahui hubungan antara file dari *database* tersebut.

Ada beberapa istilah atau definisi yang digunakan dalam sistem manajemen basis data, antara lain :

a. Entitas

   Orang, tempat, kejadian (konsep) yang informasinya direkam.

b. Atribut

   Setiap entitas mempunyai atribut atau sebutan untuk mewakili suatu entitas.

c. Nilai / isi data

   Data aktual informasi yang disimpan pada tiap data.

d. *Record / Tuple*

Kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu atau seseorang.

e. *File*

Kumpulan *file-file* yang mempunyai panjang elemen yang sama, atribut sama, namun berbeda isi datanya.

f. *Database*

Kumpulan *file-file* yang mempunyai kaitan antara *file-file* dengan *file* lainnya sehingga membentuk satu bangunan data untuk menginformasikan satu perusahaan atau instansi dalam batasan tertentu.

g. *Database Management System (DBMS)*

Kumpulan *file* yang saling berkaitan bersama dengan program untuk pengelolaanya.

### 2.7.2 Diagram E-R

Diagram E-R atau *Entity Relationship Diagram* (ERD) adalah diagram grafikal yang menggambarkan keseluruhan struktur logik dari sebuah basis data. Pada model ini semua data yang ada pada dunia nyata diterjemahkan atau ditransformasikan dengan memanfaatkan sejumlah perangkat konseptual menjadi sebuah diagram data.

Secara terjemahan dalam bahasa Indonesia, *Entity Relationship Diagram* adalah diagram relasi atau keterhubungan entitas. Dari model *Entity Relationship*

*Diagram* akan didapatkan data-data yang dibutuhkan sistem. Dengan begitu maka akan didapatkan pula kejelasan aktivitas dilakukan dalam sistem.

Didalam *Entity Relationship Diagram* (ERD) dikenal beberapa komponen, yaitu sebagai berikut :

a. Entitas

   Adalah suatu objek yang memuliki hubungan dengan objek lain. Didalam ERD digambarkan dengan bentuk persegi panjang.

b. Hubungan

   Dimana entitas dapat berhubungan dengan entitas lain, hubungan ini disebut dengan *entity relationship* yang digambarkan dengan garis.

Ada empat bentuk relasi dasar pada database, yaitu :

1. *One-to-One*

   Artinya satu data memiliki satu data pasangan.

2. *One-to Many*

   Artinya satu data memiliki beberapa data pasangan.

3. *Many-to-One*

   Artinya beberapa data memiliki satu data pasangan.

4. *Many-to-Many*

   Artinya beberapa data memiliki beberapa data pasangan.

### 2.7.3 *Structured Query Language* (SQL)

Pada bulan Juni tahun 1970 Dr. E. F. Codd mempublikasikan makalah "*A Relational Model of Data for Large Shared Data Banks*" dalam jurnal asosiasi mesin komputer (*Association of Computer Machinery* / ACM). Komunitas ACM menerima model Codd's sebagai sesuatu yang dapat ditetapkan sebagai model untuk sistem manajemen relasional database (*RDBMS*). Setelah itu dikembangkan suatu bahasa yang dinamakan *Structured English Query Language* ("SEQUEL") oleh IBM Corporation, Inc. SEQUEL kemudian menjadi SQL (walaupun masih sering dieja sebagai "sequel"). Pada tahun 1979, Relational Software Inc (yang sekarang dikenal sebagai Oracle *Corporation*) mengenalkan bentuk pertama komersialisasi SQL. Saat ini, SQL diterima sebagai bentuk standar bahasa *RDBMS*.