

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Sistem**

Menurut Herlambang dan Tanuwijaya (2005), definisi sistem dapat dibagi menjadi dua pendekatan, yaitu pendekatan secara prosedur dan pendekatan secara komponen. Berdasarkan pendekatan prosedur, sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu. Sedangkan berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Dalam perkembangan sistem yang ada, sistem dibedakan menjadi dua jenis, yaitu sistem terbuka dan sistem tertutup. Sistem terbuka merupakan sistem yang dihubungkan dengan arus sumber daya luar dan tidak mempunyai elemen pengendali. Sedangkan sistem tertutup tidak mempunyai elemen pengontrol dan dihubungkan pada lingkungan sekitarnya.

#### **2.2 Sistem Informasi**

Sistem Informasi merupakan organisasi kumpulan orang, data, proses-proses, dan teknologi informasi yang saling berhubungan dalam mengumpulkan, memproses, menyimpan, dan menyediakan sebagai *output* (yaitu) informasi yang dibutuhkan untuk mendukung sebuah organisasi. Teknologi Informasi merupakan bentuk teknologi masa kini yang menggambarkan kombinasi dari teknologi komputer (baik hardware maupun software) dengan teknologi telekomunikasi (seperti data, gambar, dan jaringan suara) (Suryantoro, 2007).

### 2.3 Manajemen Produksi

Produksi adalah kegiatan perusahaan untuk menghasilkan barang atau jasa dan bahan-bahan atau sumber-sumber faktor produksi dengan tujuan untuk dijual lagi. Sedangkan manajemen merupakan kegiatan usaha untuk mencapai tujuan dengan mengkoordinir kegiatan orang lain. Jadi manajemen produksi adalah kegiatan untuk mengatur agar dapat menciptakan dan menambah kegunaan suatu barang dan jasa. Dengan kata lain bagaimana memelihara antar **hubungan** dari semua variable dan sedapat mungkin memandang keseluruhan proses sebagai suatu sistem terpadu. Bila semua berjalan dengan **baik**, akan **dihadarkan** keluaran (*output*) berupa produk atau jasa yang memenuhi standar kualitas dan biaya.

Secara umum fungsi produksi adalah mengelola bahan baku atau bahan penolong/bahan pembantu menjadi barang jadi/jasa yang akan memberikan pendapatan bagi perusahaan.

Ada 4 (empat) macam fungsi produksi yang utama yaitu:

a. Proses.

Proses dalam artian merupakan metode/teknik yang digunakan untuk pengolahan bahan yang digunakan dalam proses produksi.

b. Jasa-jasa.

Berupa badan pengorganisasian untuk penetapan teknik-teknik sehingga proses dapat dipergunakan secara efektif.

c. Perencanaan.

Merupakan suatu hubungan/korelasi dengan organisasi dari kegiatan produksi untuk suatu jangka waktu terentu.

d. Pengawasan.

Pengawasan berfungsi untuk menjamin bahwa maksud atau tujuan mengenai penggunaan bahan yang dipergunakan pada kenyataannya telah dilaksanakan dengan baik (Ahyari, 1986).

#### 2.4 Jadwal Induk Produksi

Menurut (Gasperz, 1998) Jadwal induk produksi adalah suatu set perencanaan yang mengidentifikasi kuantitas dari item tetentu yang dapat dan akan dibuat oleh suatu perusahaan manufaktur (dalam satuan waktu)

##### 1. Make to order

*Make to order* adalah tipe industri yang membuat produk hanya untuk memenuhi pesanan. Produk-produk dari lingkungan make to order biasanya baru dikerjakan atau diselesaikan setelah menerima pesanan pelanggan.

Ciri-ciri Make to Order :

- a. Inputnya bahan baku
- b. Biasanya untuk *supply Item* dengan banyak jenis
- c. Harga cukup mahal
- d. Perlu keahlian khusus.
- e. Komponen bisa dibeli untuk persediaan.

Contoh : Mobil balap, super komputer, alat berat, restoran.

##### 2. Pemilihan item-item MPS

Faktor utama lain yang perlu diperhatikan dalam mendesain MPS adalah pemilihan item-item MPS. Pemilihan item-item yang dijadwalkan melalui MPS juga perlu mendapat perhatian khusus. Pemilihan item-item ini penting, karena tidak hanya mempengaruhi bagaimana MPS beroperasi, tetapi juga

mempengaruhi bagaimana sistem perencanaan dan pengendalian manufakturing secara keseluruhan beroperasi. Terdapat beberapa kriteria dasar yang mengatur pemilihan item-item dalam MPS, yaitu:

- a. Item-item yang dijadwalkan seharusnya merupakan produk akhir, kecuali ada pertimbangan yang jelas menguntungkan untuk menjadwalkan item-item yang lebih kecil daripada produk akhir seperti: *super family*, *super modular*, atau *super planning bills* lainnya. Penjadwalan produk-produk akhir dalam MPS menyebabkan itu menjadi sama seperti : *final assembly schedule* (FAS).
- b. Jumlah item-item MPS seharusnya sedikit, karena manajemen tidak dapat membuat keputusan yang efektif terhadap MPS apabila jumlah item MPS terlalu banyak.
- c. Seharusnya memungkinkan untuk meramalkan permintaan dari item-item MPS (kecuali item itu adalah *make to order*). Item-item yang dijadwalkan harus berkaitan erat dengan item-item yang dijual.
- d. Item-item yang dipilih harus dimasukkan dalam perhitungan kapasitas produksi yang dibutuhkan.
- e. Item-item MPS harus memudahkan dalam penerjemahan pesanan-pesanan pelanggan ke dalam pembuatan produk yang akan dikirim.

Berikut ini akan dikemukakan penjelasan singkat berkaitan dengan informasi yang ada dalam MPS.

### 1. *On Hand*

Adalah posisi inventori awal yang secara fisik tersedia dalam stok, yang merupakan kuantitas dari item yang ada dalam stok.

## 2. Time Periods for Display

Adalah banyaknya periode waktu yang ditampilkan dalam format MPS.

Dalam gambar 2 ditampilkan periode waktu 6 minggu (dengan asumsi PTF = 4 minggu). Banyaknya periode waktu dalam perencanaan MPS ini sering disebut sebagai horizon perencanaan MPS.

## 3. Actual Orders

Merupakan pesanan-pesanan yang diterima dan bersifat pasti (*certain*).

## 4. Projected Available Balances ( PAB )

Merupakan proyeksi on-hand inventory dari waktu ke waktu selama horizon perencanaan MPS, yang menunjukkan status inventori yang diproyeksikan pada akhir dari setiap periode waktu dalam horizon perencanaan MPS. Dalam buku-buku teks yang lain, PAB juga disebut sebagai *Projected On Hand Balance*. Berdasarkan informasi PAB, berbagai kebijaksanaan dan tindakan korektif dapat diambil untuk perbaikan terus-menerus dari proses manufakturing. PAB dinyatakan melewati PTF hanya sebagai informasi saja, sementara MPS dan ATP tidak direncanakan melewati PTF (*planning time fences*).

## 5. Available To Promise ( ATP )

Merupakan informasi yang sangat berguna bagi departemen pemasaran untuk mampu memberikan jawaban yang tepat terhadap pertanyaan pelanggan tentang: "Kapan pesanan selesai?". Nilai ATP memberikan informasi tentang berapa banyak item atau produk tertentu yang dijadwalkan pada periode waktu itu tersedia untuk pesanan pelanggan, sehingga berdasarkan informasi ini bagian pemasaran dapat membuat

janji yang tepat kepada pelanggan. ATP dapat juga dihitung secara kumulatif untuk memberikan informasi tentang cumulative ATP pada periode waktu tertentu.

| MPS ( Jadwal Induk Produski ) |  | Time Periods (weeks) |   |   |   |   |   |
|-------------------------------|--|----------------------|---|---|---|---|---|
|                               |  | 1                    | 2 | 3 | 4 | 5 | 6 |
| OHI                           |  |                      |   |   |   |   |   |
| Actual Order                  |  |                      |   |   |   |   |   |
| Avaiable TO Promise           |  |                      |   |   |   |   |   |
| Comulative ATP                |  |                      |   |   |   |   |   |

Gambar 2.1 Bentuk Umum dari *Master Production Schedule*

## 2.5 Penjadwalan proses

Penjadwalan proses merupakan kumpulan kebijaksanaan dan mekanisme disistem operasi yang berkaitan dengan urutan kerja yang dilakukan sistem komputer. Penjadwalan bertugas untuk memutuskan proses yang harus berjalan dan juga menentukan kapan dan berapa lama proses itu berjalan (Arum, 2010). Adapun kriteria-kriteria yang dipertimbangkan dalam proses penjadwalan:

1. Adil (fairness).

Proses yang diperlukan sama dan tidak ada suatu proses yang tidak kebagian layanan untuk diproses.

2. Efisiensi (efisiensi).

Efisiensi atau utilitas pemrosesan dihitung dengan perbandingan (ratio) waktu sibuk.

3. Waktu tanggap (response time).

Waktu tanggap berbeda untuk:

- Sistem interaktif, Didefinisikan sebagai waktu yang dihabiskan dari saat karakter terakhir dari perintah dimasukkan atau transaksi sampai pertama muncul dilayar. Waktu tanggap ini disebut terminal response time.
- Sistem waktu nyata, Didefinisikan sebagai waktu dari saat kejadian (internal atau eksternal) sampai instruktur pertama rutin layanan yang dimaksud dieksekusi, disebut even response time.

4. Turn arround time.

Adalah waktu yang menghabiskan dari saat program atau job mulai masuk ke sistem sampai proses diselesaikan sistem, diekspresikan sebagai penjumlahan waktu eksekusi (waktu pelayanan job) dan waktu menunggu, yaitu: Turn Arround Time = waktu eksekusi + waktu menunggu.

5. Throughput.

Adalah jumlah kerja yang dapat diselesaikan dalam satu unit waktu. Cara untuk mengekspresikan throughput adalah dengan jumlah job pemakai yang dapat dieksekusi dalam satu unit / interval waktu.

### 2.5.1 Algoritma Penjadwalan

Menurut Kusumadewi (2000:69) jenis-jenis algoritma berdasarkan penjadwalan terbagi atas:

a. Non Preemptive

Suatu algoritma penjadwalan dimana proses yang sedang running tidak bisa dihentikan sementara, jadi harus running terus sampai selesai. Yang termasuk algoritma non preemptive sceduling adalah:

1. FIFO (*First In First Out*) atau FCFS (*First Come First Serve*)
  2. SJF (*Shortest Job First*)
  3. HRN (*Hight Ratio Next*)
  4. MFQ (*Multiple Feedback Queues*)
- b. Preemptive

Algoritma penjadwalan memungkinkan beberapa proses yang sedang running, bisa dihentikan sementara. Algoritma ini bertujuan agar sistem lebih responsif dan dapat mengerjakan proses yang lain secara bergantian. Yang termasuk algoritma preemptive scheduling menggunakan konsep:

1. RR (*Round Robin*)
2. SRF (*Short Remaining First*)
3. PS (*Priority Scheduling*)
4. GS (*Guaranteed Scheduling*)

Klasifikasi selain berdasarkan dapat atau tidaknya suatu proses diambil secara paksa adalah klasifikasi adanya prioritas pada proses, yaitu:

1. Algoritma penjadwalan tanpa prioritas.
2. Algoritma penjadwalan berprioritas, terdiri dari :
  - a. Berprioritas statik
  - b. Berprioritas dinamis

### 2.5.2 Dynamic Priority

Menurut Kusumadewi (2000:73) *dynamic priority Scheduling* merupakan suatu penjadwalan dimana prioritas selalu berubah-ubah secara dinamis sesuai kondisi yang mana tiap-tiap proses dilengkapi dengan prioritas, pesanan

dialokasikan untuk proses yang memiliki prioritas lebih tinggi. Sebagai contoh ada 3 proses iklan P1, P2, dan P3 dengan lama proses :

Tabel 2.1 Contoh Penjadwalan Dynamic Priority Scheduling

| Proses | Waktu/ jam | Prioritas |
|--------|------------|-----------|
| P1     | 6          | 2         |
| P2     | 4          | 1         |
| P3     | 3          | 4         |

Maka didapat waktu proses seperti Tabel 2.2 berikut:

Tabel 2.2 Waktu Proses Penjadwalan Dynamic Priority Scheduling

|              |         |         |         |
|--------------|---------|---------|---------|
| Waktu Proses | 4<br>P2 | 6<br>P1 | 3<br>P3 |
|--------------|---------|---------|---------|

Dari tabel di atas diketahui bahwa tugas P2 memiliki prioritas lebih tinggi dari pada P1, maka P2 akan dierjakan terlebih dahulu, dan dilanjutkan dengan P1 dan P3.

### 2.5.3 Shortest job first scheduling

Menurut Kusumadewi (2000) Algoritma *Shortest Job First Scheduling* sangat optimal karena memberikan rata-rata waktu tunggu lebih kecil dibandingkan algoritma penjadwalan yang lain dengan cara memindahkan job-job pendek di depan job-job yang panjang, sehingga akan mengurangi waktu tunggu.

Untuk memperjelasnya dapat dilihat di contoh berikut.

Misalkan ada 4 job yaitu A, B, C, D masing – masing waktu kedatangan sama yaitu pada  $t = 0$ , dan lama proses job berturut – turut : 8, 4, 4, 4.

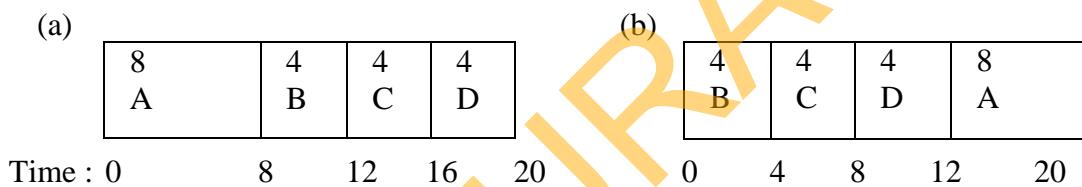
Tabel 2.3. Contoh penjadwalan shortest job first

| PROSES | WAKTU |
|--------|-------|
| A      | 8     |
| B      | 4     |
| C      | 4     |
| D      | 4     |

Jika urutan pengjerjaannya :

- a. Job A, B, C, D
- b. Job B, C, D, A

Maka proses pengjerjaanya adalah sebagai berikut :



- a. Dengan pengjerjaan job berdasarkan urutan (a) maka berturut – turut waktu yang dibutuhkan untuk proses A, B, C, D adalah 8, 12, 16, 20 sehingga dapat dihitung waktu rata – rata =  $( 8 + 12 + 16 + 20 ) / 4 = 14$
  - b. Bila job yang dikerjakan berdasarkan (b), yaitu dengan shortest job first, maka waktu yang dibutuhkan untuk proses B, C, D, A adalah 4, 8, 12, 20 atau rata – rata =  $( 4 + 8 + 12 + 20 ) / 4 = 11$
- berdasarkan contoh di atas, bahwa algoritma shortest job first scheduling lebih optimal.

## 2.6 SQL Server 2005

SQL Server 2005 merupakan aplikasi database produk microsoft yang memiliki fitur-fitur baru sehingga membuatnya menjadi platform database yang

sempurna, khususnya dalam mengelola data yang berskala besar (Yuswanto, 2008).

SQL Server 2005 perangkat lunak Relational Database Management System (RDBMS) yang handal. Didesain untuk mendukung proses transaksi yang besar seperti online order entry, inventory, akuntansi atau manufaktur. SQL Server 2005 dapat dijalankan pada Windows 2000 Pro SP2, Windows 2000 Server SP4, Windows XP Professional SP2 atau Windows 2003 Server SP1. SQL Server 2005 membutuhkan Windows installer 3.1 yang dapat diperoleh pada saat instalasi Visual Studio 2005. SQL Server 2005 memiliki fasilitas tambahan yang menyebabkannya memiliki kemampuan penuh dalam *e-Commerce*, antara lain *reporting* dan *analysis services*.

SQL didefinisikan sebagai bahasa pemrograman untuk memproses basis data, yang dapat dipergunakan untuk mendefinisikan basis data, memanipulasi data yang ada di dalam basis data dan untuk melakukan administrasi penggunaan basis data.

SQL pertama kali didefinisikan oleh E.F.Codd dan IBM San Jose Laboratory pada tahun 1970 dengan nama *Structured English Query Language* (SEQUEL). Pada perkembangannya SEQUEL diubah namanya menjadi SQL dan oleh American National Standard Institute (ANSI) didefinisikan sebagai SQL-92 dan distandarisasi oleh Internasional Standard Organization (ISO) sebagai ISO sebagai ISO/IEC 9057:1992, “*Database Languange SQL*”.

SQL termasuk keluarga *Fourth Generation Language* (4GL) dimana perintah-perintah SQL merupakan deskripsi dari hasil yang diinginkan. Dalam ANSI-SQL, perintah-perintah SQL dibagi menjadi 6 kategori, yaitu:

## 1. *Data Query Language*

*Data-Query Language* dipergunakan untuk mengambil data yang diperlukan dari basis data.

## 2. *Data Manipulation Language*

Digunakan untuk melakukan manipulasi data di dalam tabel. Perintah-perintah *Data-Manipulation Language* ini adalah *Insert* untuk mengisi data baru, *Update* untuk mengubah data lama dengan data baru, *Delete* untuk menghapus data yang ingin dihapus.

## 3. *Transaction Processing Language*

Digunakan untuk menentukan apakah data telah diperbarui dengan mempergunakan *Data-Manipulation Language* dan akan disimpan secara permanen didatabase atau tidak. Perintah-perintah SQL yang termasuk di dalam kelompok ini adalah *Begin Transaction, Commit dan Rollback*.

## 4. *Data Control Language*

Digunakan untuk menentukan hak akses individu atau grup dalam mempergunakan database. Perintah-perintah SQL yang termasuk dalam kelompok ini adalah *Grant dan Revoke*.

## 5. *Data Definition Language*

Digunakan untuk membuat tabel baru dalam database (*Create table*), menambah file indeks (*Create Index*), hubungan antar table (*Primary Key, Foreign key, Preference*) serta menghapus tabel dan indeks (*Drop Table, Drop Index*).

## 6. Cursor Control Language

Digunakan untuk memproses record dari tiap-tiap table. Perintah-perintahnya adalah *Fetch Into, Update Where Current*.

## 2.7 Analisa dan Perancangan Sistem

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.

Menurut Kendall dan Kendall (2003) Analisa dan Perancangan Sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi.

Berikut ini adalah proses dalam analisis dan perancangan sistem yang terbagi menjadi dua, yaitu:

### 2.7.1 Entity Relationship 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 terdefinisi dalam suatu organisasi, dapat

abstrak dan nyata. Untuk setiap *entity* biasanya mempunyai *atribute* yang merupakan ciri *entity* tersebut. Relasi adalah hubungan antar *entity* yang berfungsi sebagai hubungan yang mewujudkan pemetaan antar *entity*.

Menurut Marlinda (2004), *Atribute* adalah kolom sebuah relasi. Macam-macam *atribute* yaitu:

a. *Simple Attribute*

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

b. *Composite Attribute*

*Composite attribute* adalah *attribute* yang memiliki dua nilai harga, misalnya nama besar (nama keluarga) dan nama kecil (nama asli).

c. *Single Value Attribute*

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

d. *Multi Value Attribute*

*Multi value attribute* adalah *attribute* yang banyak memiliki nilai harga, misalnya *entity* mahasiswa dengan *attribute*-nya pendidikan (SD, SMP, SMA).

e. *Null Value Attribute*

*Null value attribute* adalah *attribute* yang tidak memiliki nilai harga, misalnya *entity* tukang becak dengan *attribute*-nya pendidikan (tanpa memiliki ijazah).

ERD 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 ERD 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.

#### 2.7.2 **Data Flow Diagram (DFD)**

Pada tahap ini, penggunaan notasi dapat membantu komunikasi dengan pemakai/user sistem untuk memahami sistem tersebut secara logika. Diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem ini dikenal dengan nama *Diagram Arus Data (Data Flow Diagram)*. DFD berfungsi untuk menggambarkan proses aliran data yang terjadi di dalam sistem dari tingkat yang tertinggi sampai yang terendah, yang memungkinkan untuk melakukan dekomposisi, mempartisi atau membagi sistem kedalam bagian-bagian yang lebih kecil dan yang lebih sederhana.

DFD fokus pada aliran data dari dan ke dalam sistem serta memproses data tersebut (Kendall dan Kendall, 2003). Simbol-simbol dasar dalam DFD antara lain:

##### a. *External Entity*

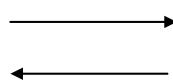
Suatu *External Entity* atau entitas merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat. Gambar 2.2 merupakan simbol entitas dalam DFD dalam model Gane dan Sarson.



Gambar 2.2 Simbol *External Entity*

b. *Data Flow*

*Data Flow* atau aliran data disimbolkan dengan tanda panah. *Data Flow* menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses. Gambar 2.3 merupakan simbol *Data Flow*.

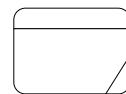


Gambar 2.3 Simbol Data Flow

c. *Process*

Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan.

Gambar 2.4 merupakan simbol *process*.



Gambar 2.4 Simbol *Process*

d. *Data Store*

*Data Store* adalah simbol yang digunakan untuk melambangkan proses penyimpanan data. Gambar 2.5 merupakan simbol file penyimpanan/*data store*.



Gambar 2.5 Simbol Data Store

## 2.8 Konsep Dasar Basis Data

Kumpulan data yang saling berhubungan (relasi). Relasi biasanya ditunjukan dengan kunci dari tiap file yang ada. Dalam satu file terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu *record* terdiri dari *field* yang saling berhubungan menunjukan bahwa *field* tersebut dalam satu pengertian yang lengkap dan direkam dalam satu *record*.

## 2.9 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 mengambil keputusan.

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu Perangkat Keras (*Hardware*), Sistem Operasi (*Operating System*), Basis Data (*Database*), Sistem (Aplikasi atau Perangkat Lunak) Pengelola Basis Data (DBMS), Pemakai (*User*), dan Aplikasi (Perangkat Lunak) lain (bersifat opsional).

Tabel 2.4 Keuntungan dan Kerugian Sistem Basis Data

| Keuntungan Sistem Basis Data                                                                                                            | Kerugian Sistem Basis Data               |
|-----------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------|
| Mengurangi Kerangkapan data, yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga update dilakukan berulang-ulang | Diperlukan tempat penyimpanan yang besar |

| Keuntungan Sistem Basis Data                                                               | Kerugian Sistem Basis Data                          |
|--------------------------------------------------------------------------------------------|-----------------------------------------------------|
| Mencegah ketidakkonsistenan                                                                | Diperlukan tenaga yang terampil dalam mengolah data |
| Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang | Perangkat lunaknya mahal                            |

## 2.10 Database

Menurut Yuswanto (2008), *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 merupakan sebuah file.

Menurut Marlinda (2004), *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).

Menurut Marlinda (2004), *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.

Berikut ini merupakan Bahasa-bahasa yang terdapat dalam DBMS , yakni:

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.

Fungsi-fungsi dari DBMS terdiri dari 3 yaitu *Data Definition*, *Data Manipulation*, *Data Security* dan *Integrity*. Berikut adalah penjelasan dari ketiga fungsi tersebut

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*. adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan DD analis sistem dapat mendefinisikan data yang mengalir disistem dengan lengkap. Pada tahap analisis sistem, DD digunakan sebagai alat komunikasi antara analis sistem dengan pemakai sistem tentang data yang mengalir ke sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.