

2. LANDASAN TEORI

2.1. Peramalan

2.1.1. Definisi Peramalan

Langkah awal dalam suatu perencanaan produksi dan persediaan adalah mengetahui besar permintaan di masa mendatang yang akan masuk ke perusahaan. Tindakan untuk mengetahui besar permintaan di masa mendatang (atau secara umum kejadian-kejadian di masa mendatang) disebut tindakan peramalan atau *forecasting*.

Deretan data-data historis yang berjalan menurut waktu disebut *time series*. Terdapat 5 komponen plot data *time series*, yaitu :

a. *Level*

b. *Trend*

Pola data *trend* adalah pola data di mana terdapat kecenderungan naik atau turun dari *time series* selama selang waktu tertentu.

c. *Seasonal Variations* (Variasi Musiman)

Pola data *seasonal* adalah pola data yang berulang dari tahun ke tahun atau dengan periode perulangan lebih pendek, misalnya 3 bulan atau 6 bulan.

Periode perulangan ini berkaitan erat dengan kondisi alam (musim hujan, panas, dingin), tradisi (hari raya, tahun baru), jadwal pembukaan sekolah (tahun ajaran baru / semesteran).

d. *Cycle*

Pola data *cycle* adalah pola data yang berulang setelah beberapa tahun, antara 2 hingga 10 tahun.

Siklus ini seringkali terjadinya sangat panjang dan dapat disebabkan oleh beberapa faktor.

Misal : siklus bisnis : resesi, depresi, dan lain-lain.

e. *Irregular / Random Fluctuations*

Pola data *irregular* adalah pola data yang berfluktuasi dengan tidak teratur, tidak mempunyai pola, yang sering disebabkan oleh kejadian-kejadian yang tidak terduga.

Misal : kebakaran, gempa bumi, dan lain-lain.

Kejadian-kejadian ini disebut “ *noise* ” atau *residual*.

Dengan mengetahui komponen yang ada pada data-data historis, maka dapat ditentukan metode peramalan apa yang dapat dipakai.

2.1.2. Metode Dekomposisi

Metode *Dekomposisi* digunakan untuk meramalkan *time series* yang menunjukkan adanya pola *trend* dan pengaruh musiman. Ide dasar dari metode ini adalah menguraikan (*decompose*) *time series* ke dalam beberapa faktor yaitu : *trend*, *seasonal*, siklus dan acak (Elsayed dan Boucher, 1994). Untuk metode ini digambarkan sebagai berikut :

$$\text{Model} = Y_t = TR_t \times SN_t \times CL_t \times IR_t \quad (2.1)$$

Dengan :

Y_t = data *time series* pada periode t

TR_t = komponen *trend* pada periode t

SN_t = komponen musiman pada periode t

CL_t = komponen siklus pada periode t

IR_t = komponen acak pada periode t

Langkah-langkah yang digunakan :

1. Untuk menghilangkan variasi *seasonal* dan acak, digunakan metode *Moving Average* atau *Centered Moving Average*. Efek yang tertinggal adalah *trend* dan siklus sehingga $tr_t \times cl_t = \text{tetap}$.
2. Dari model $Y_t = TR_t \times SN_t \times CL_t \times IR_t$

$$SN_t \times IR_t = \frac{Y_t}{TR_t \times CL_t}$$

Estimasi :

$$sn_t \times ir_t = \frac{y_t}{tr_t \times cl_t}$$

(2.2)

3. Dilakukan proses Normalisasi yaitu

$$Sn_t = \overline{sn}_t \left(\frac{L}{\sum_{t=1}^L \overline{sn}_t} \right)$$

Di mana : L = periode MA

$$\overline{sn}_t = \frac{1}{m} \sum_{k=0}^{m-1} Sn_{t+k.L} \times (I_{t+k.L}) \quad (2.3)$$

4. Estimasi komponen tertentu dilakukan proses deseasonalisasi untuk mengetahui efek dari *trend*. Indeks *deseasonal* pada periode $t = d_t$

$$d_t = \frac{y_t}{sn_t} \quad (2.4)$$

5. Indeks deseasonalisasi ini kemudian diplot untuk mengetahui polanya.
6. Jika garis lurus (*straight line*) maka diasumsikan adanya *trend* linier dan dimodelkan dengan regresi linier.

$$TR_t = \beta_0 + \beta_1 t$$

Dengan estimasi dari TR_t adalah tr_t

$$tr_t = b_0 + b_1 t, \text{ dimana}$$

$$b_0 = \frac{\sum_{t=1}^n d_t}{n} - b_1 \frac{\sum_{t=1}^n t}{n}$$

$$b_1 = \frac{n \sum_{t=1}^n t d_t - \left(\sum_{t=1}^n t \right) \left(\sum_{t=1}^n d_t \right)}{n \sum_{t=1}^n t^2 - \left(\sum_{t=1}^n t \right)^2}$$

(2.5)

Jika tidak berupa garis lurus, maka dapat digunakan persamaan regresi lain yang sesuai (misal : kuadratik, eksponensial, dll).

7. Membuat estimasi $CL_t \times IR_t$

$$CL_t \times IR_t = \frac{y_t}{TR_t \times SN_t}$$

Estimasi:

$$cl_t \times ir_t = \frac{y_t}{tr_t \times sn_t} \quad (2.6)$$

Berdasarkan pengalaman, jika periode data adalah bulanan atau kuartalan maka nilai CL_t dapat dihitung sebagai berikut :

$$cl_t = \frac{cl_{t-1}ir_{t-1} + cl_tir_t + cl_{t+1}ir_{t+1}}{3}$$

CL_t adalah MA dengan 3 periode dari $CL_t \times IR_t$ (2.7)

8. Menghitung IRt dengan persamaan :

$$ir_t = \frac{cl_t \times ir_t}{cl_t} \quad (2.8)$$

Jika tidak ada pola dari komponen acak maka : $CL_t \times IR_t = 1$, sehingga $F_t = TR_t \times SN_t \times CL_t$. Bila tidak ada siklus maka $F_t = TR_t \times SN_t$.

2.1.3. Perhitungan Besar Kesalahan Peramalan.

Untuk mengevaluasi harga parameter peramalan, digunakan ukuran kesalahan peramalan. Harga parameter kesalahan peramalan yang terbaik adalah harga yang memberikan nilai kesalahan peramalan yang kecil. Adapun indikator akurasi dari peramalan yaitu :

a. MAD (*Mean Absolute Deviation*)

$$MAD = \frac{\Sigma (absolute Forecast error)}{n} \quad (2.9)$$

b. MSE (*Mean Square Error*)

$$MSE = \frac{\Sigma (Forecast error)^2}{n} \quad (2.10)$$

c. MAPE (*Mean Absolute Percentage Error*)

$$MAPE = \frac{\Sigma (Forecast error / data actual) * 100}{n} \quad (2.11)$$

d. Validasi “ *Tracking Signal* ”

Salah satu cara untuk mengontrol peramalan untuk memastikan bahwa metode peramalan yang digunakan akan menghasilkan peramalan yang baik

adalah dengan menggunakan *tracking signal*. Yang merupakan perhitungan mengenai sebaik apakah metode peramalan yang digunakan dalam memprediksi data aktualnya.

Tracking signal dihitung sebagai jumlah kesalahan peramalan (*running sum forecast error / RSFE*) dibandingkan dengan nilai MAE. Secara umum *tracking signal* dirumuskan sebagai berikut :

$$TS = \frac{RSFE}{MAE} \quad (2.12)$$

Di mana :

$$RSFE = \Sigma (\text{data aktual periode } i - \text{data peramalan periode } i)$$

$$MAE = \frac{\Sigma | \text{kesalahan peramalan} |}{n} = \frac{\Sigma |e_i|}{n}$$

Nilai positif *tracking signal* menunjukkan bahwa data aktual masih lebih besar dibandingkan data peramalannya. Sedangkan nilai negatif *tracking signal* berarti bahwa data aktual lebih kecil dibandingkan dengan data peramalannya. Sebuah *tracking signal* yang baik adalah nilai positif dan negatif dari kesalahan peramalan harus seimbang sehingga pusat atau titik tengah *tracking signal* mendekati nilai nol, dengan batas penerimaan ± 4

2.2. Sistem Persediaan

Sistem pengendalian persediaan dapat diartikan sebagai serangkaian kebijaksanaan dan pengendalian untuk memonitor tingkat persediaan dan menentukan titik persediaan yang harus dijaga, kapan persediaan harus disediakan dan berapa pesanan yang harus dilakukan dengan tujuan untuk menentukan dan menjamin tersedianya sumber daya yang tepat dalam jumlah dan waktu yang tepat untuk meminimumkan jumlah total biaya yang dikeluarkan (Handoko, 1994).

Beberapa terminologi dalam sistem persediaan :

a. Demand (permintaan)

Dapat bersifat *deterministic* atau *probabilistic*, statis atau dinamis. Dalam perhitungannya digunakan laju permintaan yang menyatakan jumlah permintaan per satuan waktu.

b. *Lead time dan replenishment rate*

Lead time menyatakan selang waktu di antara saat peletakan order dan saat item pesanan masuk ke *inventory*.

Lead time ini dapat bersifat *deterministic* atau *probabilistic*, dan konstan atau berbeda-beda menurut waktu.

Replenishment rate menyatakan laju pengisian kembali atau laju bertambahnya *inventory*.

c. *Reorder Level*

Reorder level menyatakan tingkat *inventory* di mana order harus dilakukan kembali. *Reorder level* merupakan fungsi dari permintaan selama *lead time* (*lead time demand*)

d. *Safety Stock* (Cadangan Pengaman)

Safety stock menyatakan bagian dari *inventory* yang digunakan sebagai cadangan untuk mencegah terjadinya kekurangan persediaan (*stockout*) oleh karena ketidakpastian dalam *demand* ataupun proses *supply*. *Safety stock* juga merupakan fungsi dari *lead time demand*.

2.2.1. Komponen-komponen Biaya Persediaan

Biaya-biaya yang mempengaruhi kebijaksanaan persediaan terbagi dalam 3 kelompok utama (Handoko, 1994) :

a. Biaya penyimpanan (*Holding Cost*)

Yaitu biaya- biaya yang bervariasi secara langsung dengan jumlah/ kuantitas persediaan. Biaya penyimpanan satu periode waktu akan semakin besar bila jumlah yang dipesan semakin banyak. Yang termasuk dalam biaya penyimpanan adalah:

- Biaya fasilitas penyimpanan seperti, biaya penerangan, biaya pendinginan, dan sebagainya.
- Biaya modal, yaitu alternatif pendapatan atas dana yang diinvestasikan dalam persediaan tersebut.
- Biaya depresiasi/ keusangan.
- Biaya asuransi persediaan.

b. Biaya kehabisan atau kekurangan bahan (*Shortage Cost*)

Biaya kekurangan bahan adalah biaya yang paling sulit diperkirakan.

Biaya ini timbul bilamana persediaan tidak mencukupi adanya permintaan bahan. Yang termasuk biaya kekurangan bahan adalah:

- Biaya kehilangan penjualan.
- Biaya kehilangan langganan.
- Biaya pemesanan khusus.
- Biaya terganggunya operasi.

c. Biaya pemesanan (*Ordering Cost*)

Yaitu biaya yang harus ditanggung oleh perusahaan bilamana setiap kali suatu bahan dipesan. Umumnya biaya pemesanan tidak tergantung dari kuantitas pemesanan, tetapi tergantung dari berapa kali pesanan itu dilakukan. Yang termasuk dalam biaya pemesanan adalah:

- Biaya pemrosesan pemesanan
- Biaya telepon/ faks.
- Biaya surat- menyurat.
- Biaya pengepakan.
- Biaya pemeriksaan.
- Biaya transportasi.
- Dan lain- lain.

2.2.2. Parameter-parameter dalam Sistem Persediaan

Parameter-parameter yang digunakan dalam sistem persediaan adalah sebagai berikut :

D = Laju permintaan untuk setiap periode perencanaan permintaan di masa datang

A = Biaya pemesanan setiap kali pesan

H = Biaya penyimpanan per unit per periode perencanaan (misal per bulan)

c = Harga beli per satuan item

r^* = Menyatakan titik pemesanan kembali di mana pada saat itu pemesanan harus dilakukan kembali untuk mengisi persediaan

Q^* = Kuantitas pemesanan dalam satuan barang

TC = Total biaya persediaan per periode perencanaan

Terdapat beberapa kebijakan dalam pengendalian persediaan yang berkaitan dengan penentuan waktu peletakan pesanan dan jumlah pemesanan ekonomis yang harus dilakukan yaitu : *periodic review policy*, *order up to R policy*, *base stock policy* dan *continuous review policy* (Elsayed dan Boucher, 1994).

Kebijakan *continuous review* selalu memonitor secara kontinu tingkat persediaan dan pesanan sebesar Q dilakukan pada saat tingkat persediaan mencapai titik pemesanan kembali. Waktu pemesanan tidak bergantung pada panjang periode waktu tertentu melainkan tergantung pada tingkat persediaan yang mencapai titik pemesanan kembali.

2.2.3. Model Persediaan

Model pengendalian persediaan yang akan digunakan yaitu model statis EOQ

Asumsi yang digunakan untuk EOQ (Elsayed dan Boucher, 1994) :

1. *Demand* konstan dan pasti
2. Tidak memperhitungkan diskon
3. Tidak ada perubahan harga
4. *Order preparation cost* and *carrying cost* konstan dan sudah diketahui
5. Setiap pesanan diterima dalam sekali pengiriman dan langsung dapat digunakan
6. Tidak ada *shortage*
7. Digunakan untuk perencanaan jangka panjang

Tujuan dari EOQ ini yaitu untuk menentukan jumlah pemesanan yang optimal sehingga meminimumkan total biaya persediaan, di mana total biaya persediaan (TC) dirumuskan sebagai :

§ TC = biaya pemesanan + biaya penyimpanan + biaya pembelian

$$TC = \frac{AD}{Q} + H \frac{Q}{2} + D.C \quad (2.13)$$

$$H = i x c \quad (2.14)$$

$$Q = \sqrt{\frac{2AD}{H}} \quad (2.15)$$

2.3. Sistem Informasi Manajemen

2.3.1. Pengertian Sistem

Suatu sistem dapat diartikan sebagai sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Subsistem sebenarnya hanyalah sistem di dalam suatu sistem.

2.3.2. Pengertian Informasi

Data harus dibedakan dengan fakta. Data terdiri dari fakta-fakta dan angka-angka yang secara relatif tidak berarti bagi pemakai. Informasi adalah data yang telah diproses atau data yang memiliki arti.

2.3.3. Pengertian Manajemen

Manajemen terdiri dari proses atau kegiatan yang menjelaskan apa yang dilakukan manajer pada operasi organisasi mereka, yaitu merencanakan, mengorganisasikan, memprakarsai dan mengendalikan operasi.

Mereka mengorganisasikan tugas-tugas yang diperlukan untuk rencana operasional, menyusun tugas ini ke dalam kelompok yang homogen, dan menetapkan pendelegasian wewenang. Mereka mengendalikan prestasi kerja dengan menentukan norma-norma prestasi kerja dan menghindari penyimpangan terhadap norma atau standard tersebut.

2.3.4. Sistem Informasi Manajemen

Sistem informasi manajemen biasanya disebut dengan MIS (*Management Information System*) yang merupakan suatu sistem terintegrasi guna menyediakan informasi yang mendukung perencanaan, pengendalian (kontrol) dan operasi dari suatu kelompok organisasi.

2.4. Flow Chart (Diagram Alir)

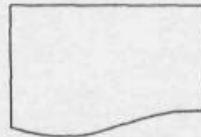
Flow Chart merupakan suatu diagram yang menunjukkan alir di dalam program atau prosedur sistem secara logika. *Flow Chart* digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. *Flow Chart* pada dasarnya menggunakan satu kumpulan simbol-simbol standar untuk menjelaskan prosedur sebuah proses dalam sebuah organisasi maupun perusahaan serta untuk menjelaskan aliran data dalam sebuah sistem tertentu dalam organisasi tersebut.

Flow Chart System merupakan diagram yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Diagram tersebut menjelaskan urutan-urutan dari prosedur-prosedur yang ada pada sistem. Simbol-simbol yang digunakan antara lain:

a. Simbol dokumen

Menunjukkan dokumen input dan output baik untuk operasi manual, mekanik atau komputer.

Simbol :



b. Simbol operasi manual

Menunjukkan pekerjaan manual (tidak menggunakan alat)

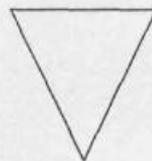
Simbol :



c. Simbol penyimpanan file

Menunjukkan file non komputer yang diarsip.

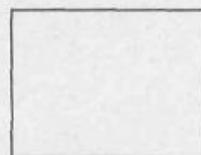
Simbol :



d. Simbol proses

Menunjukkan kegiatan proses secara komputerisasi

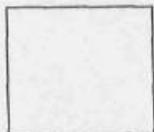
Simbol :



e. Simbol operasi luar

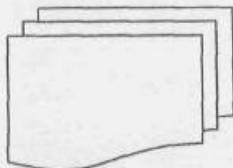
Menunjukkan operasi yang dilakukan di luar proses operasi komputer.

Simbol :



f. Simbol salinan dari sebuah dokumen

Simbol :



g. Simbol aliran dokumen atau pemrosesan

Menunjukkan aliran yang normal arahnya ke kanan atau ke bawah.

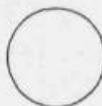
Simbol :



h. Simbol penghubung *flow chart* yang terputus

Menghubungkan aliran proses pada halaman yang sama.

Simbol :



2.5. Data Flow Diagram

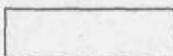
Data flow diagram merupakan diagram yang menggunakan notasi-notasi dalam menggambarkan arus dari sistem. Pada DFD penggambaran arus data dalam sistem terlihat jelas dan terstruktur. Selain itu DFD juga merupakan dokumentasi dari sistem yang baik.

Simbol yang digunakan dalam DFD antara lain :

a. *External entity* (kesatuan luar) atau *boundary* (batas sistem)

External entity merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem.

Simbol :



b. *Data flow* (arus data)

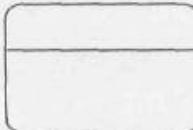
Arus data mengalir di antara proses, simpanan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem

dan dapat berbentuk formulir atau dokumen, laporan tercetak, masukan untuk komputer, komunikasi ucapan, surat-surat atau memo dan lain sebagainya.

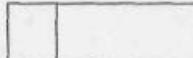
Simbol : 

c. *Process* (proses)

Suatu proses adalah kegiatan atau kerja yang akan dilakukan oleh orang, mesin atau komputer dari suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang keluar dari proses.

Simbol : 

d. *Data store* (simpanan data)

Simbol : 

2.6. *Data Base*

Data base adalah kumpulan informasi dari suatu obyek. Obyek ini dapat nyata dan abstrak. Contoh untuk obyek nyata yaitu data *inventory*, obyek abstrak yaitu merk dagang.

Pada data yang semakin banyak dan beragam, *data base* dapat berfungsi sebagai :

- a. Mengelompokkan data berdasarkan jenisnya
- b. Menyimpan informasi untuk berbagai keperluan

2.6.1. Sistem Manajemen *Data Base* (DBMS)

DBMS adalah kelompok paket program yang mempunyai fungsi antara lain:

- a. Mengelola file-file *data base*
- b. Mengakses *file-file*
- c. Memperbaiki data sesuai yang diinginkan
- d. Menambah dan menghapus *record*
- e. Mengorganisasi kembali data

2.6.2. Komponen DBMS

- a. Minimal satu orang yang menjadi pemiliknya dan bertanggungjawab *data base* tersebut.
- b. Serangkaian aturan dan hubungan yang menentukan dan mengatur interaksi antara berbagai unsur dari *data base*.
- c. Manusia yang memasukkan data ke dalam *data base* tersebut.
- d. Organisasi yang mengeluarkan data dari *data base* tersebut.
- e. *Data base*-nya sendiri.

2.6.3. Tujuan Data Base

- a. Menyediakan tempat penyimpanan massal untuk data yang terkait.
- b. Membuat pemakainya mudah mengakses data.
- c. Memungkinkan respon yang segera atas permintaan dari pemakai.
- d. Menghapus data yang berlebihan.

2.6.4. Model Data Base

- a. *Entity set*, yang merupakan kumpulan *entity* dengan fungsi dan karakteristik yang sama.
- b. *Relation set*, yang merupakan kumpulan semua *relationship* antara satu *entity set* dengan beberapa *entity set* lainnya untuk memberi informasi bagi suatu individu.
- c. *Attribute*, merupakan definisi dari kelompok data / identifikasi dari *field*.
- d. *Key attribute*, memiliki karakteristik *key attribute* harus menunjukkan keunikan dan jika satu *key attribute* dihilangkan maka tidak menjamin keunikannya lagi.

Deskripsi *relationship* yang terjadi adalah :

- 1-1, satu *entity* dari satu *entity set* tertentu berhubungan dengan satu *entity* dari satu *entity set* tertentu.
- 1-N atau N-1, satu *entity* dari satu *entity set* tertentu berhubungan dengan N *entity* dari satu *entity set* tertentu atau sebaliknya.
- N-M atau M-N, N *entity* dari satu *entity set* tertentu berhubungan dengan M *entity* dari satu *entity set* tertentu atau sebaliknya.

Relationship yang terjadi antara *entity* ini dapat bersifat *obligatory* maupun *non-obligatory*. Di mana *obligatory* adalah keadaan di mana semua *record* dalam *attribute* memiliki relasi dengan *record* dari *attribute* lainnya (tidak ada yang menganggur). Sedangkan *non-obligatory* adalah keadaan di mana ada satu atau lebih *record* dalam suatu *attribute* tidak memiliki relasi dengan *record* dari *attribute* lainnya. Keadaan *obligatory* ini ditunjukkan oleh simbol *entity* yang memiliki tanda titik. Sedangkan simbol *entity* tanpa tanda titik menunjukkan keadaan *non-obligatory*.



Gambar

Entity Obligatory dan Entity Non-obligatory

Berdasarkan *relationship* antara *entity* ini, yang merupakan *binary relationship* atau relasi yang memiliki dua *entity*, maka dapat dilakukan *design relationship*, yaitu :

a. *Relation* untuk *Binary relationship* derajat 1 : 1

- Rule 1: Ketika derajat dari suatu *binary relationship* 1 : 1 di mana kedua *entity*-nya bersifat *obligatory*, maka hanya dibutuhkan satu buah *relation* saja.
- Rule 2 : Ketika derajat dari suatu *binary relationship* 1 : 1 di mana salah satu *entity*-nya bersifat *non-obligatory*, maka hanya dibutuhkan 2 buah *relation*.
- Rule 3 : Ketika derajat dari suatu *binary relationship* 1 : 1 di mana salah kedua *entity*-nya bersifat *non-obligatory*, maka hanya dibutuhkan 3 buah *relation*

b. *Relation* untuk *Binary relationship* derajat 1 : N

- Rule 4 : Jika derajat *binary relationship*-nya 1 : N atau N : 1 di mana sisi N nya memiliki sifat *obligatory*, maka hanya dibutuhkan 2 buah *relation*. Hal ini tidak memandang apakah sisi satunya adalah *obligatory* maupun *non-obligatory*.
- Rule 5 : Jika derajat *binary relationship*-nya 1 : N atau N : 1 di mana sisi N nya memiliki sifat *non-obligatory*, maka dibutuhkan 3 buah *relation*. Hal

ini tidak memandang apakah sisi satunya adalah *obligatory* maupun *non-obligatory*.

c. *Relation* untuk *Binary relationship* derajat M : N

- Rule 6 : Jika derajat *binary relationship*-nya M : N , maka harus dibutuhkan 3 buah *relation*, yaitu 2 untuk masing-masing *entity* dan 1 untuk *relationship*-nya.

2.6.5. Skema *Data Base*

Skema *data base* adalah kumpulan dari skema relasi yang telah ada. Dimana perlu juga menentukan *key attribute* di dalam skema *data base* tersebut. *Key attribute* dalam hal ini adalah beberapa *attribut* dari relasi yang dapat dipergunakan untuk mengidentifikasi keunikan dari sistem.

Ketergantungan antara satu *attribut* terhadap *attribut* lainnya atau sering disebut *Functional Dependencies*. Yang terdiri dari ketergantungan penuh, parsial, dan transitif.

Prinsip-prinsip normalisasi :

- Tidak ditemukan adanya *attribut* yang repetitive.
- Memenuhi aturan pertama dan tidak ditemukan ketegantungan parsial, yaitu semua *attribut non key attribute* tergantung secara penuh terhadap *key attribute*.
- Memenuhi aturan pertama dan kedua, dan tidak ada ketergantungan transitif.

2.6.6. Struktur *Data Base*

Struktur *data base* biasanya berisikan *file*, *field* dan *type*. Dalam 1 *file* ada beberapa baris yang disebut *record*. Jadi *record* adalah kumpulan dari *field-field*. Kolom di dalam *file* disebut *field*. Tiap *field* harus memiliki nama yang disebut *field name* dan memiliki *type*. Typenya adalah numerik, karakter, *logical* dan data.