BAB II

LANDASAN TEORI

2.1 Konsep Persediaan

Inventory secara umum dapat dibagi menjadi empat bagian proses bisnis utama, yaitu :

1. ***Purchasing***

   Purchasing atau pembelian barang merupakan unit bisnis yang melayani transaksi pembelian barang. Proses bisnis *purchasing* terbagi menjadi beberapa sub proses bisnis antara lain: *Purchase Request*, *Purchase Order*, *Receiving*, *Purchase Return*.

2. ***Sales***

   Sales atau penjualan barang merupakan unit bisnis yang melayani transaksi penjualan barang kepada pelanggan. Proses bisnis *sales* terbagi menjadi sub proses bisnis antara lain: *Sales Order*, *Sales Return*.

3. ***Distribution***

   Distribution atau pengiriman merupakan transaksi pendistribusian barang kepada pelanggan maupun pendistribusian barang antar gudang. Proses bisnis *distribution* terbagi menjadi beberapa sub proses bisnis antara lain: *Shipment*, *Load Sheet*, *Location Transfer*.

4. ***Controlling Stock***

   Controlling stock merupakan proses pengecekkan kesesuaian *stock* barang pada gudang dengan sistem. Proses bisnis *controlling stock*

terbagi menjadi beberapa sub proses bisnis antara lain: *Inventory Stock Take*, *In Debit Adjusment*, *In Credit Adjusment*, *Item Balance*, *Valuation Item*.

2.1.1 Pengertian Persediaan

"*Inventory* atau persediaan merupakan simpanan material yang berupa bahan mentah, barang dalam proses dan barang jadi. Dari sudut pandang sebuah perusahaan maka persediaan adalah investasi modal yang dibutuhkan untuk menyimpan material pada kondisi tertentu". Sumayang (2003:197).

Barang persediaan adalah sejumlah material yang disimpan dan dirawat menurut aturan tertentu dalam tempat persediaan agar selalu dalam keadaan siap pakai dan ditatausahakan dalam buku perusahaan. Indrajit dan Djokopranoto (2003:4).

Sediaan (*inventory*) adalah sejumlah bahan atau barang yang tersedia untuk digunakan sewaktu-waktu di masa yang akan datang. Sediaan terjadi apabila jumlah bahan atau barang yang diadakan (dibeli atau dibuat sendiri) lebih besar daripada jumlah yang digunakan (dijual atau diolah sendiri).

2.1.2 Penjualan

Penjualan adalah kegiatan pemasok atau penjual untuk menjalankan bisnis, guna memenuhi kebutuhan pasar, menetapkan harga, mendistribusikan serta mempromosikan melalui proses pertukaran agar memuaskan konsumen dalam mencapai tujuan perusahaan, sedangkan menurut Jogiyanto (2001:1) Penjualan adalah transaksi perubahan nilai barang menjadi nilai uang atau nilai piutang.

2.1.3 Pembelian

Menurut Carter (2001:1) Pembelian adalah proses untuk mendapatkan suatu barang dengan menukar sesuatu, tetapi prosesnya mempunyai peran dan lingkup

serta tujuan dan strategi untuk tujuan tertentu, sedangkan menurut Wijaya, (2002:127), Pembelian adalah menerima suatu barang dengan menyerahkan sesuatu kepada orang lain dengan menberikan sesuatu sebagai syarat/ perjanjian yang telah disepakati.

2.1.4 Manajemen Persediaan

Pengendalian terhadap persediaan atau *inventory control* adalah aktifitas mempertahankan jumlah persediaan pada tingkat yang dikehendaki. Pada produk barang, pengendalian *inventory* ditekankan pada pengendalian material. Pada produk jasa, pengendalian diutamakan sedikit pada material dan banyak pada jasa pasokan karena konsumsi sering kali bersamaan dengan pengadaan jasa sehingga tidak memerlukan persediaan. Sumayang (2003:197).

Harus ada keseimbangan antara mempertahankan tingkat *inventory* yang tepat dengan pengaruh keuangan minimum terhadap pelanggan. Jika investasi sangat besar akan mengakibatkan biaya modal yang sangat besar, sehingga akan mengakibatkan juga biaya operasi yang tinggi. Sumayang (2003:197).

Pengendalian tingkat persediaan bertujuan mencapai efisiensi dan efektifitas optimal dalam penyediaan material. Dalam pengertian di atas, usaha yang perlu dilakukan dalam manajemen persediaan secara garis besar sebagai berikut:

1. Menjamin terpenuhinya kebutuhan operasi
2. Membatasi nilai seluruh investasi
3. Membatasi jenis dan jumlah material
4. Memanfaatkan seoptimal mungkin material yang ada

2.1.5 Prinsip Manajemen Persediaan

Mengenai persediaan barang ada konsep pengelolaaan yang harus dianut, yaitu penentuan jumlah dan jenis barang yang disimpan dalam persediaan haruslah sedemikian rupa sehingga produksi dan operasi perusahaan tidak terganggu, tetapi di lain pihak sekaligus harus dijaga agar biaya investasi yang timbul dari penyediaan barang tersebut seminimal mungkin.

Prinsip tersebut memang selaras dengan prinsip ekonomi yaitu, menghasilkan keluaran tertentu dengan biaya seminimal mungkin atau dengan biaya tertentu menghasilkan keluaran semaksimal mungkin. Jika melihat prinsip persediaan tersebut, maka jelas bahwa diperlukan perpaduan antara dua hal yang sangat bertolak belakang. Indrajit dan Djokopranoto (2003:10-11).

2.1.6 Tujuan Persediaan

Tujuan utama dari persediaan bahan baku adalah menghubungkan pemasok dengan pabrik. Sumayang (2003:201-203). Ada tiga alasan mengapa persediaan diperlukan:

1. **Menghilangkan pengaruh ketidakpastian**

   Untuk menghadapi ketidakpastian maka pada sistem persediaan ditetapkan persediaan darurat yang dinamakan *safety stock.*

   a. Apabila permintaan telah diketahui maka persediaan barang dalam proses dan barang jadi akan disesuaikan dengan permintaan. Dalam hal ini tidak perlu ada persediaan dan apabila ada gejolak permintaan akan diteruskan kebagian produksi dan begian produksi akan berusaha mengatasi gejolak permintaan ini.

b. Tetapi sesungguhnya *safety stock* dapat mengatasi hal seperti ini tanpa ikut campur bagian produksi. Demikian juga dengan persediaan bahan baku yang akan menyerap seandainya ada gejolak dari pemasok.

2. Memberi waktu luang untuk pengelolaan produksi dan pembelian

Kadang-kadang lebih ekonomis memproduksi barang dalam proses atau barang jadi dalam jumlah besar atau dalam jumlah paket yang kemudian disimpan sebagai persediaan. Selama persediaan masih ada maka proses produksi dihentikan dan akan dimulai lagi bila diketahui persediaan hampir habis. Pertimbangan ini memberikan kemudahan, yaitu:

a. Memberikan kemungkinan untuk menyebarkan dan meratakan beban biaya investasi pada sejumlah besar produk.
b. Memungkinkan penggunaan satu peralatan untuk menghasilkan bermacam-macam jenis produk.

3. Untuk mengantisipasi perubahan pada *demand* dan *supply*

Persediaan disiapkan untuk mengadapi beberapa kondisi yang menunjukkan perubahan *demand* dan *supply*.

a. Bila ada perkiraan perubahan harga dan persediaan bahan baku.
b. Sebagai persiapan menghadapi promosi pasar di mana sejumlah besar barang jadi disimpan menunggu penjualan tersebut.
c. Perusahaan yang melakukan produksi dengan jumlah output tetap akan mengalami kelebihan produk, pada kondisi permintaan rendah atau pada kondisi musim lesu atau *low season*.

2.1.7 Klasifikasi Barang Persediaan

Secara fisik, item persediaan dapat dikelompokkan kedalam lima kategori (Baroto, 2002:52), yaitu sebagai berikut:

1. **Bahan Mentah (*raw materials*)**

   Yaitu barang-barang berwujud seperti baja, atau bahan-bahan mentah lainnya yang diperoleh dari sumber-sumber alam, atau dibeli dari pemasok, atau diolah sendiri oleh perusahaan.

2. **Komponen Rakitan (*parts/components*)**

   Yaitu barang-barang yang terdiri atas bagian-bagian (*parts*) yang diperoleh dari perusahaan lain atau hasil produksi sendiri untuk digunakan dalam pembuatan barang jadi atau barang setengah jadi.

3. **Barang Setengah Jadi (*work in process*)**

   Yaitu barang-barang keluaran dari tiap operasi produksi atau perakitan yang telah memiliki bentuk lebih kompleks daripada komponen.

4. **Barang Jadi (*finished good*)**

   Adalah barang-barang yang telah selesai diproses dan siap untuk di distribusikan ke konsumen.

5. **Bahan Pembantu (*supplies material*)**

   Adalah barang-barang yang diperlukan dalam proses pembuatan atau perakitan barang.

2.1.8 Fungsi Persediaan

Inventory memiliki fungsi tersendiri, sebagaimana tujuan dari diadakannya persediaan, berikut ini merupakan fungsi-fungsi utamanya:

1. Menghilangkan resiko keterlambatan pengiriman bahan
2. Menyesuaikan dengan jadwal produksi
3. Menghilangkan resiko kenaikan harga
4. Menjaga persediaan bahan yang dihasilkan secara musiman
5. Mengantisipasi permintaan yang dapat diramalkan
6. Mendapatkan keuntungan dari quantity discount
7. Komitmen terhadap pelanggan

2.2 *Inventory Multi Warehouse*

Inventory Multi Warehouse merupakan sebuah pengelolaan persediaan yang melibatkan lebih dari satu gudang.

2.2.1 Gudang

Gudang adalah suatu tempat yang digunakan untuk menyimpan barang, yang dijadikan sebagai objek transaksi bisnis, baik oleh sebuah badan tertentu maupun perseorangan.

2.2.2 Jenis Gudang

1. ***Central Warehouse* (Gudang Pokok)**

   Merupakan sebuah gudang yang berfungsi sebagai penerimaan, penyimpanan dan pengiriman barang.

2. ***Retailer Warehouse* (Gudang Pengecer)**

   Merupakan gudang yang berfungsi sebagai aktivitas penjualan langsung, biasanya adalah toko yang menjual barang kepada konsumennya.

3. ***Distribution Warehouse* (Gudang Distribusi)**

   Merupakan sebuah gudang yang berfungsi sebagai pendistribusian.

2.3 *Economic Order Quantity* (EOQ)

Menurut Aminudin (2005:67) Dalam persoalan persediaan dikenal beberapa model. Masing-masing model mempunyai karakteristik tersendiri sesuai dengan parameter persoalan. Pada dasarnya model persediaan dibagi menjadi dua kelompok utama, yaitu model deterministik dan model stokastik. Model deterministik semua nilai parameter-parameternya diasumsikan diketahui dengan pasti sedangkan model stokastik nilai-nilai parameternya tidak diketahui dengan pasti, berupa nilai-nilai acak. Model EOQ dapat dikelompokkkan sebagai berikut:

1. Model EOQ Deterministik Statik
2. Model EOQ Deterministik Dinamik
3. Model EOQ Probabilistik Statik
4. Model EOQ Probabilistik Dinamik (Stokastik)

Sementara itu berikut ini jenis-jenis model persediaan EOQ Deterministik Statik:

1. Model EOQ Klasik (Sederhana) atau *Single Item*
2. Model EOQ *Back Order*
3. Model EOQ *Fixed Production Rate*
4. Model EOQ *Quantity Discount*

2.3.1 Model EOQ Klasik

Model jumlah pesanan terhemat (*Economic Order Quantity*) yang sering juga dinamakan model ukuran tumpuk sederhana digunakan dalam menentukan jumlah barang yang akan dipesan untuk setiap kali pemesanan serta jumlah biaya

pengadaan bahan-bahan. EOQ menunjukkan jumlah barang yang harus dipesan setiap kali pemesanan agar biaya persediaan keseluruhan menjadi minimum.

"Konsep perhitungan EOQ ini juga berdasarkan pemikiran yang cukup logis dan sederhana sebagai berikut makin sering pengisian kembali persediaan itu dilakukan, persediaan rata-rata akan semakin kecil, dan ini mengakibatkan biaya penyediaan barang akan makin kecil juga. Tetapi, di lain pihak makin sering pengisian kembali persediaan itu dilakukan, maka biaya pemesanan akan semakin besar pula. Oleh karena itu, dicari suatu keseimbangan yang paling ekonomis atau paling optimal dari dua hal yang saling bertentangan tersebut. Untuk mencari titik keseimbangan inilah maksud dari rumus EOQ". Indrajit dan Djokopranoto (2003:55).

Metode ini disebut juga dengan metode ukuran lot atau *lot size method* yang digunakan untuk pengelolaan *independent demand inventory* dan didasarkan pada asumsi sebagai berikut. Aminudin (2005:67-70):

1. Kecepatan permintaan tetap dan terus menerus serta diketahui.
2. Waktu antara pemesanan sampai dengan pesanan datang (*lead time*) harus tetap.
3. Tidak pernah ada kejadian persediaan habis atau *stock out*.
4. Material dipesan dalam paket atau lot dan pesanan datang pada waktu yang bersamaan dan tetap dalam bentuk paket.
5. Harga per unit tetap dan tidak ada pengurangan harga walaupun pembelian dalam jumlah volume yang besar.
6. Besar *carrying cost* tergantung secara garis lurus dengan rata-rata jumlah persediaan.
7. Besar *ordering cost* atau *set up cost* tetap untuk setiap lot yang dipesan dan tidak tergantung pada jumlah item pada setiap *lot*.
8. *Item* yaitu produk semacam dan tidak berhubungan dengan produk lain.

9. *Stockout* tidak diperbolehkan, karena *demand* dan *lead time* diketahui

10. Barang yang dipesan dan disimpan hanya barang sejenis (homogeny)

Sedangkan parameter-parameter yang digunakan adalah:

k = *ordering cost* per pemesanan

A = jumlah barang yang dibutuhkan dalam 1 periode

c = *procurement cost* per unit barang yang dipesan

h = *holding cost* per unit nilai persediaan

T = waktu antara pemesanan

Total annual cost = ordering cost + holding cost

Dari model persoalan persediaan ini akan dicari berapa jumlah pemesanan (Q), sehingga *total annual cost* mencapai minimum. *Order point* adalah satu dimana siklus persediaan yang baru dimulai dan yang lama berakhir. Setiap siklus persediaan mempunyai periode T, artinya setiap T satuan waktu pemesanan kembali dilakukan dan ini tergantung pada Q. lamanya T sama dengan proporsi kebutuhan selama satu periode (A). Aminudin (2005:67-70).

$$T = \frac{Q}{A}$$

$$\text{Frekuensi pemesanan} = \frac{A}{Q} \qquad \textit{Annual ordering cost} = \left(\frac{A}{Q}\right)k$$

Komponen biaya kedua adalah *holding cost*, yang ditentukan oleh jumlah dan lamanya barang disimpan. Setiap waktu jumlah barang berkurang sehingga perlu diperhatikan tingkat persediaan rata-rata.

$$\boxed{\text{Rata-rata persediaan} = \frac{Q}{2}}$$

Holding cost dihitung berdasarkan satuan nilai persediaan dan *procurement cost* (c), sehingga:

Holding cost per unit barang = hc

$$Total\ annual\ cost = \left(\frac{A}{Q}\right)k + hc\left(\frac{Q}{2}\right) \qquad Annual\ holding\ cost = \text{hc}\left(\frac{Q}{2}\right)$$

Total annual cost minimum dapat dicari dengan menentukan berapa jumlah pemesanan (Q). *Total annual cost* mencapai minimum jika antara fungsi *annual order cost* dan *total annual holding cost* berharga sama. Secara matematis ditulis:

$$hc\left(\frac{Q}{2}\right) = \left(\frac{A}{Q}\right)k$$

$$Q = \sqrt{\frac{2Ak}{hc}}$$

Dalam menerapkan EOQ ada beberapa biaya yang harus dipertimbangkan dalam penentuan jumlah pembelian atau keuntungan, diantaranya :

1. Biaya Pemesanan

   Biaya pemesanan merupakan biaya yang akan langsung terkait dengan kegiatan pemesanan yang dilakukan perusahaan. Biaya pemesanan berfluktuasi bukan dengan jumlah barang yang dipesan, tetapi dengan frekuensi pesanan. Biaya pesan tidak hanya terdiri dari biaya yang eksplisit, tetapi juga biaya kesempatan (*Opportunity Cost*). Misalnya, waktu yang terbuang untuk memproses pesanan, menjalankan administrasi pesanan dan sebagainya. Contoh biaya tersebut antara lain: Biaya persiapan, Biaya telepon, Biaya pengiriman dan Biaya pembuatan

faktur. Rumus biaya pemesanan menurut Heizer (2005:73) adalah sebagai berikut:

$$BiayaPesan = \frac{D}{Q} xS$$

Keterangan :

Q = Jumlah Barang setiap pesan.

D = Permintaan barang persediaan, dalam unit.

S = Biaya pemesanan untuk setiap pesanan.

2. Biaya Penyimpanan

Biaya penyimpanan merupakan biaya yang harus ditanggung oleh perusahaan sehubungan dengan adanya barang yang disimpan dalam gudang perusahaan. Biaya simpan akan berfluktuasi dengan tingkat persediaan. Beberapa contoh biaya penyimpanan antara lain: Biaya pemeliharaan, Biaya asuransi, Biaya kerusakan dalam penyimpanan, Biaya sewa gedung, dan Biaya fasilitas penyimpanan. Menurut Heizer (2005:71) biaya penyimpanan dirumuskan sebagai:

$$BiayaPenyimpanan = \frac{Q}{2} xH$$

Keterangan:

Q = Jumlah barang setiap pemesanan H = Biaya penyimpanan

Sehingga dalam menentukan biaya persediaan ada dua (2) jenis biaya yang berubah-ubah dan harus dipertimbangkan. Pertama berubah-ubah sesuai dengan frekuensi pesanan yaitu biaya pesan. Kedua biaya yang berubah-ubah sesuai

dengan besar kecilnya persediaan yaitu biaya penyimpanan. Selanjutnya menentukan total biaya persediaan (TC) dengan menjumlahkan biaya pesan dan biaya simpan. Adapun rumusnya sebagai berikut :

$$TC = \frac{D}{Q}S + \frac{Q}{2}H$$

Keterangan:

TC = Total biaya persediaan

Q = Jumlah barang setiap pesan

D = Permintaan tahunan barang persediaan dalam unit

S = Biaya pemesanan untuk setiap pesan

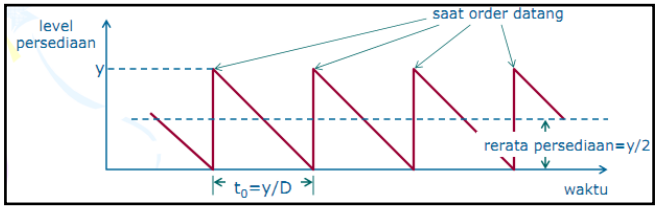
H = Biaya penyimpanan per unit per tahun

Didefinisikan dalam bentuk grafik sebagai berikut:

y = kuantitas *order* (unit)

D = permintaan (unit/waktu)

t0 = panjang siklus pengorderan (waktu)



Gambar 2.1 EOQ *Inventory*

[**Sumber**: Indrajit, Eko Richardus dan Djokopranoto, Richardus. 2003. *Manajemen Persediaan*. Jakarta: PT Gramedia Widiasarana Indonesia.]

Hal-hal yang perlu diperhatikan sebelum menghitung EOQ:

D : Besar laju permintaan (*demand rate*) dalam unit per tahun.

S : Biaya setiap kali pemesanan (*ordering cost*) per pesanan

C : Biaya per unit dalam harga (rupiah) per unit

i : Biaya pengelolaan (*carrying cost*) adalah persentase terhadap nilai persediaan per tahun.

Q : Ukuran paket pesanan (*lot size*) dalam unit

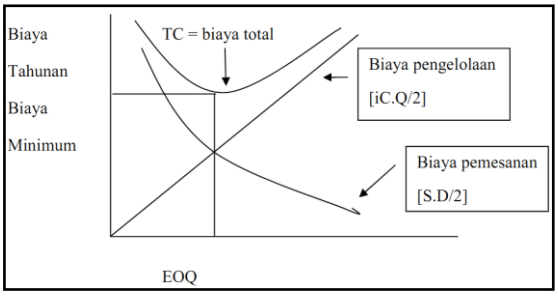
TC : Biaya total persediaan dalam rupiah per tahun.

H : *Holding Cost* (ic)

Biaya pemesanan per tahun (*Ordering cost*): OC = S (D/Q)

Biaya pengelolaan persediaan per tahun (*Carrying cost*): CC = ic (Q/2)

Maka, total biaya persediaan: TC = S (D/Q) + ic (Q/2)



Gambar 2.2 Keseimbangan EOQ

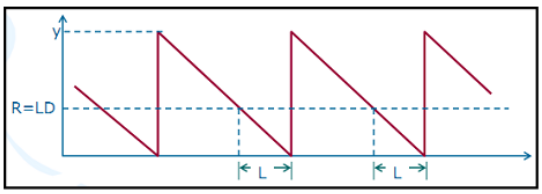
[**Sumber**: Aminudin. 2005. *Prinsip-Prinsip Riset Operasi*. Jakarta: Erlangga.]

Terjadi keseimbangan antara *carrying cost* dan *ordering cost* dan menghitung jumlah paket pesanan ekonomis atau EOQ yaitu Q dihitung dari:

$$\mathbf{Q = \sqrt{(2SD)/ic}}$$

2.3.2 Ekstensi Model EOQ Klasik *Lead Time*

Kenyataannya, kedatangan *order* tidak secara bersamaan, ada selang waktu dari pengorderan sampai kedatangan barang yang disebut sebagai *lead time* (L). Sementara itu pengorderan harus dilakukan ketika persediaan tinggal sejumlah kebutuhan selama *lead time* = LD = *reorder point* (R).



Gambar 2.3 EOQ *Inventory Lead Time*

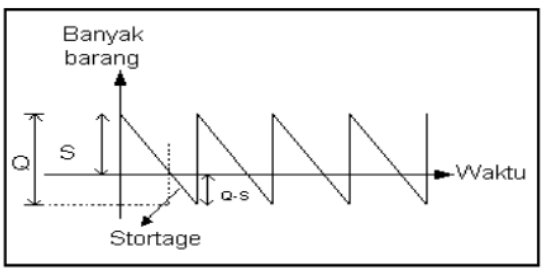
[**Sumber**: Indrajit, Eko Richardus dan Djokopranoto, Richardus. 2003. *Manajemen Persediaan*. Jakarta: PT Gramedia Widiasarana Indonesia.]

2.3.3 Model EOQ *Back Order*

Model ini mendeskripsikan EOQ yang berusahaa untuk tidak kehilangan penjualan yang mana akan menyebabkan pelanggan menunggu. Asumsinya adalah sebagai berikut:

1. Pesanan untuk diambil kemudian lazim disebut *back order*. Dengan kata lain sebuah toko mengijinkan adanya *back order* apabila ia tetap menjual suatu barang yang meskipun sudah tidak ada di gudang (tingkat persediaan barang nol).
2. Model EOQ dengan asumsi pembeli dapat memesan barang kembali walaupun pesanan tersebut akan dipenuhi kemudian, sehingga dalam model ini ada biaya yang harus dikeluarkan akibat kehabisan *stock* (Cs).

Sebagai pembanding berikut ini merupakan gambar grafik EOQ-nya:



Gambar 2.4 EOQ *Back Order*

[**Sumber**: Aminudin. 2005. *Prinsip-Prinsip Riset Operasi*. Jakarta: Erlangga.]

Contoh Kasus:

Sebuah toko minuman mampu menjual 5200 peti bir setiap tahun. Setiap peti menanggung biaya $ 20 untuk sampai ke gudang. Penyalur meminta bayaran $ 100 untuk pemesanan. Modal kerja yang dimiliki toko tsb dipinjam dari bank dengan bunga 10% pertahun, selain itu pemilik toko harus membayarkan atas barang yg disimpannya sebesar 5 % dari nilai persediaan rata-rata serta asuransi sebesar 5 %. Disisi lain pembeli akan menunggu sampai barang tsb ada, walaupun barang tersebut sedang tidak tersedia. Andaikan toko tsb dibebani $ 0,01 per peti per hari sebagai hukuman karena tidak dapat memenuhi permintaan langganan, maka tentukan Qo dan So dari kasus di atas?

Jawaban:

D = 5200 per Tahun; Co = $ 10; Cs = 3,65 per tahun

Ch1 =20 % = 0,2 [bunga bank + asuransi + pajak simpanan] x 20 = (0,2)(20) = 4 + 20 = $ 24

$$Qo = \sqrt{\frac{2CoD}{Ch}}\left(\frac{Cs + Ch}{Cs}\right) = \sqrt{\frac{2.100 .5200}{24}}\left(\frac{3,65 + 24}{3,65}\right) = \sqrt{328033 ,1} = 572,7 \approx 573 \text{ peti}$$

$$So = \sqrt{\frac{2CoD}{Ch}}\sqrt{\frac{Cs}{Cs + Ch}} = \sqrt{\frac{2.100 .5200}{24}}\sqrt{\frac{3,65}{(3,65 + 24)}} = \sqrt{43333 ,3} \quad \sqrt{0,132} = 74,9 = 75$$

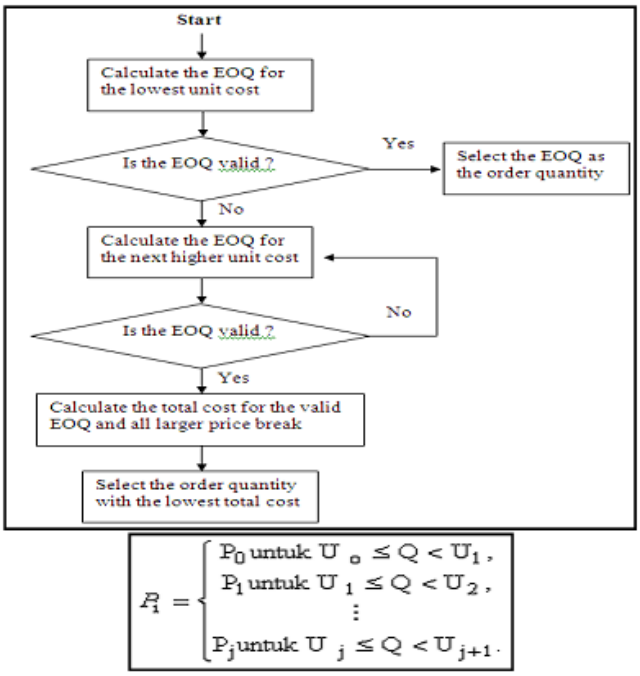
To = Qo /D = 573/5200 = 0,11 tahun = 40 hari

Ternyata apabila toko tsb mengijinkan adanya back order maka kebijaksanaan persediaan yang optimal = 573 peti yg dipesan setiap 40 hari. Diantara yg dipesan tersebut, hanya 75 peti yang disimpan sebagai persediaan, selebihnya 498 peti dipergunakan untuk memenuhi permintaan yang belum terpenuhi (back order).

2.3.4 Model EOQ *Quantity Discount*

Model EOQ ini memiliki prosedur menentukan jumlah yang dipesan sebagai berikut:

1. Mulai dengan harga yang paling rendah, hitung EOQ pada setiap harga, sampai diperoleh EOQ yang *valid*
2. Hitung biaya total tahunan untuk EOQ yang *valid* dan semua *price break quantities* yang lebih besar dari EOQ (*price-break quantity* adalah jumlah terkecil yang akan mendapat harga diskon)
3. Pilih *quantity* dengan biaya total minimum pada Langkah 2.



$$P_i = \begin{cases} P_0 \text{ untuk } U_0 \le Q < U_1, \\ P_1 \text{ untuk } U_1 \le Q < U_2, \\ \vdots \\ P_j \text{ untuk } U_j \le Q < U_{j+1}. \end{cases}$$

Gambar 2.5 Flow Chart EOQ *Quantity Discount*

[**Sumber**: Aminudin. 2005. *Prinsip-Prinsip Riset Operasi*. Jakarta: Erlangga.]

2.4 *Lead Time* (Waktu Tunggu)

Untuk menjamin kelancaran proses bisnis perusahaan perlu memperhatikan jangka waktu antara saat mengadakan pemesanan dengan saat penerimaan barang-barang yang dipesan kemudian dimasukkan kedalam gudang. Lamanya waktu antara mulai dilakukannya pemesanan barang sampai dengan kedatangan barang yang dipesan dinamakan *lead time*. Barang yang datang terlambat mengakibatkan

kekurangan barang. Sedangkan barang yang datang lebih awal dari waktu yang telah ditentukan akan memaksa perusahaan memperbesar biaya penyimpanan.

Faktor-faktor yang perlu diperhatikan dalam menetukan *lead time* adalah:

1. *Stock Out Cost*

   Stock Out Cost adalah biaya-biaya yang terpaksa dikeluarkan karena keterlambatan datangnya barang.

2. *Extra Carrying Cost*

   Extra Carrying Cost adalah biaya-biaya yang terpaksa dikeluarkan karena keterlambatan barang datang lebih awal.

2.5 *Reorder Point* (ROP)

Merupakan sebuah titik dimana suatu pemesanan baru harus dilakukan. Hal ini dipengaruhi oleh *lead time*, yaitu waktu yang dibutuhkan untuk menerima pesanan setelah pemesanan dilakukan atau persiapan dimulai.

Jika model EOQ yang diterapkan, maka faktor terpenting adalah *lead time*. *Lead time* adalah jarak waktu antara melakukan *order* hingga *order* datang. Adanya *lead time* membuat kita harus menentukan waktu pemesanan. Pada model EOQ *lead time* diketahui dengan pasti. Namun pada kenyataannya, baik permintaan maupun *lead time* sama-sama tidak pasti. Oleh karena itu, waktu pemesanan suatu barang harus mempertimbangkan ketidakpastian dari dua aspek tersebut. Berikut ini cara perhitungan untuk mendapatkan nilai ROP:

$$\mathbf{ROP = Q \times Lt}$$

Q : Jumlah kebutuhan

Lt : *Lead Time*

2.6 Persediaan Pengamanan (*Safety Stock*)

Menurut Taylor (2005:364), persediaan cadangan adalah persediaan yang disimpan untuk mengantisipasi permintaan pelanggan yang sulit diketahui dengan pasti. Stok cadangan disimpan untuk memenuhi permintaan yang besar.

Persediaan pengaman atau sering pula disebut *safety stock* adalah persediaan yang dilakukan untuk mengantisipasi unsur ketidakpastian permintaan dan penyediaan. Apabila persediaan pengaman tidak mampu mengantisipasi ketidakpastian tersebut, maka akan terjadi kekurangan persediaan (*stockout*).

Faktor-faktor yang menentukan besarnya *safety stock* adalah :

1. **Penggunaan bahan baku rata-rata.**

   Salah satu dasar untuk memperkirakan penggunaan bahan baku selama periode tertentu, khusunya selama periode pemesanan adalah rata-rata penggunaan bahan baku pada masa sebelumnya.

2. **Faktor waktu.**

   Lead time adalah lamanya waktu antara mulai dilakukannya pemesanan bahan-bahan sampai dengan kedatangan bahan-bahan yang dipesan.

3. **Persediaan antisipasi.**

   Persediaan antisipasi disebut sebagai *stabilization stock* merupakan persediaan yang dilakukan untuk menghadapi fluktuasi permintaan yang sudah dapat diperkirakan sebelumnya.

4. **Persediaan dalam pengiriman (*transit stock*).**

   Persediaan dalam pengiriman disebut *work-in process stock* adalah persediaan yang masih dalam pengiriman, yaitu :

a. *External transit stock* adalah persediaan yang masih berada dalam transportasi.

b. *Internal transit stock* adalah persediaan yang masih menunggu untuk diproses atau menunggu sebelum dipindahkan.

ROPs = ROP + (Qmaks –Qr) x Lt

ROPs : *reorder point* dengan stok pengaman

ROP : *reoder point* semula (sebelum *safety stock*)

Qmaks : jumlah permintaan (penggunaan) maksimal

Qr : jumlah permintaan rata-rata

Lt : *Lead time*

2.7 Konsep Dasar Sistem

Dalam buku Al-Bahra (2005:2) yang berjudul Analisis dan Desain Sistem Informasi mendefinisikan bahwa:

"Sistem terdapat dua kelompok pendekatan sistem, yaitu sistem yang lebih menekankan pada prosedur dan elemennya. Prosedur didefinisikan sebagai suatu urutan-urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa yang harus dikerjakan, siapa yang mengerjakan, kapan dikerjakan, dan bagaimana mengerjakannya".

2.7.1 Pengertian Sistem

Bambang Harianto (2004:59-68) Sistem adalah sekumpulan objek atau elemen yang saling berinteraksi untuk mencapai satu tujuan tertentu. Beberapa prinsip umum sistem adalah sebagai berikut:

1. Sistem selalu merupakan bagian dari sistem yang lebih besar, sekaligus sistem tersebut dapat dipartisi menjadi sub-sistem yang lebih kecil

2. Sistem yang lebih terspesialisasi akan kurang dapat beradaptasi untuk menghadapi keadaan-keadaan yang berbeda.
3. Lebih besar ukuran sistem, maka memerlukan sumberdaya yang lebih banyak untuk operasi dan pemeliharaannya.
4. Sistem senantiasa mengalami perubahan, tumbuh dan berkembang.

Hanif Al Fatta (2007:3) mengemukakan: Sistem adalah kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi dan saling bergantung sama lain, sedangkan menurut Kristanto (2008:1), Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu. Sistem adalah kumpulan/grup dari subsistem/bagian/komponen apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. Azhar Susanto (2004:18).

Sistem yang tengah dikaji adalah berada di dalam batas, sementara di luar batas adalah lingkungan dimana sistem berinteraksi. Lingkungan didefinisikan sebagai kondisi-kondisi yang melingkupi, di luar batas lingkungan mempengaruhi dan dipengaruhi sistem, tapi tidak dapat dikendalikan oleh sistem. Karena batas-batas dari sistem berbasis komputer sering sulit didefinisikan maka pengembang, *client* dan pemakai sistem harus bekerja sama dalam memutuskan mengenai apa-apa yang termasuk di dalam sistem. Maka yang dimaksud dengan Sistem adalah sekumpulan objek atau elemen yang dipandang sebagai keseluruhan dan

dirancang untuk mencapai satu sasaran, memiliki batas dimana sistem berada didalamnya dan di luar batas adalah sebagai lingkungannya.

Dari penjabaran definisi diatas maka dapat digambarkan suatu sistem dengan menetukan komponen-komponennya, bagaimana bagian-bagian tersebut berhubungan dan bagaimana ciri-ciri dari tujuan yang harus dicapai.

2.7.2 Karakteristik Sistem

Al-Bahra (2005:3) menerangkan bahwa suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yaitu:

1. **Komponen Sistem (*Component*)**

   Komponen sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian dari sistem.

2. **Batasan Sistem (*Boundary*)**

   Batas sistem 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 dan menunjukkan ruang lingkup dari sistem tersebut.

3. **Lingkungan Luar Sistem (*Environment*)**

   Lingkungan luar sistem dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian

harus dijaga dan dipelihara. Sedangkan lingkungan luar sistem yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan menganggu kelangsungan hidup dari sistem.

4. **Penghubung Sistem (*Interface*)**

   Penghubung merupakan media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya.

5. **Masukan Sistem (*Input*)**

   Masukan sistem adalah energi yang dimasukkan kedalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal maintenance input yaitu energi yang dimasukkan supaya sistem tersebut dapat berjalan. Sinyal input adalah energi yang diproses untuk mendapatkan keluaran dari sistem.

6. **Keluaran Sistem (*Output*)**

   Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

7. **Pengolahan Sistem (*Process*)**

   Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

8. **Sasaran Sistem (*Goal*)**

   Suatu sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan

berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

2.7.3 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya. Karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan kedalam beberapa sudut pandang. Berikut akan dipaparkan mengenai klasifikasi sistem. Al-Bahra (2005:6).

1. **Sistem Abstrak dan Sistem Fisik**
   a. Sistem abstrak: Sistem yang berupa pemikiran/ide-ide yang tidak tampak secara fisik.
   b. Sistem fisik: Sistem yang ada secara fisik.
2. **Sistem Alamiah dan Sistem Buatan**
   a. Sistem alamiah: Sistem yang terjadi karena proses alam tidak dibuat oleh manusia.



Gambar 2.6 Sistem Tata Surya

[**Sumber**: http://blog.unsri.ac.id/userfiles/09071003013.doc]

   b. Sistem buatan manusia: Sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin

disebut dengan *human-machine sistem* atau ada yang menyebut dengan *man-machine sistem.* Contoh: Sistem Akuntansi.

3. **Sistem Tertentu dan Sistem Tak tentu**
   a. Sistem tertentu: Beroperasi dengan tingkah laku yang sudah dapat diprediksi.
   b. Sistem tak tentu: Sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. **Sistem Tertutup dan Sistem Terbuka**
   a. Sistem tertutup: Sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya.
   b. Sistem terbuka: Sistem yang berhubungan dan terpengruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lain.

2.8 Konsep Dasar Informasi

Informasi merupakan hal yang sangat penting di dalam sebuah sistem. Jika sebuah sistem mengolah informasi yang salah maka penerima informasi akan susah untuk mengambil keputusan masa kini atau masa yang akan datang.

2.8.1 Data dan Informasi

Data adalah deskripsi dari sesuatu dan kejadian yang kita hadapi (*the description of things and events that we face).* Definisi data yang lain adalah merupakan kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Al-Bahra (2005:8).

Informasi merupakan salah satu sumberdaya penting dalam suatu organisasi, digunakan sebagai bahan pengambilan keputusan. Kadir dan Terra (2003:546). Menurut Gorgon. B. Davis (1985) dalam buku Al-bahra (2005:8) mendefinisikan informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang.

Sumber informasi adalah data. Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Informasi diperoleh setelah data-data mentah diproses atau diolah. Agar informasi yang dihasilkan menjadi lebih berharga, maka harus memenuhi kriteria sebagai berikut: Al-Bahra (2005:9)

1. Informasi harus akurat, sehingga mendukung pihak manajemen dalam mengambil keputusan.
2. Informasi harus relevan, terasa bermanfaat bagi yang membutuhkan.
3. Informasi harus tepat waktu, sehingga tidak ada keterlambatan pada saat dibutuhkan.

2.8.2 Pengolahan Data

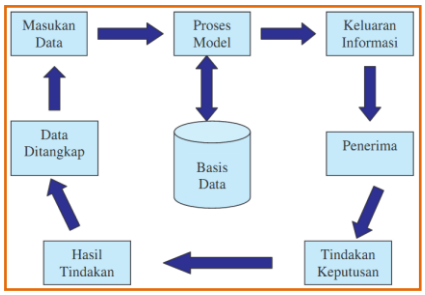
Pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan. Ada beberapa operasi yang dilakukan dalam pengolahan data, antara lain sebagai berikut: Al-Bahra (2005:9)

1. Data masukan, adalah kumpulan data transaksi ke sebuah pengolahan data medium.

2. Data transformasi, merupakan penghitungan/pengelompokan terhadap kelompok-kelompok tertentu. Beberapa bentuk data transformasi :
   a. Kalkulasi operasi aritmatik terhadap data *field*
   b. Menyimpulkan proses akumulasi beberapa data
   c. Melakukan klasifikasi terhadap data group-group tertentu
3. Informasi keluaran, adalah proses menampilkan informasi.

2.8.3 Siklus Informasi

Untuk memperoleh informasi yang bermanfaat bagi penerimanya, maka perlu dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. Al-Bahra (2005:11)



Gambar 2.7 Siklus Informasi

[**Sumber** : Al-bahra bin Ladjamudin. 2005. *Analisis dan Desain Sistem Informasi.* Graha Ilmu. Yogyakarta.]

2.8.4 Kualitas Informasi

Kualitas informasi sangat dipengaruhi atau ditentukan oleh hal-hal sebagai berikut. Al-Bahra (2005:11):

1. **Relevan (*relevancy*)**, yaitu sejauh mana tingkat relevansi informasi tersebut terhadap kenyataan kejadian masa lalu, masa sekarang dan kejadian yang akan datang.

2. **Akurat (*accuracy*)**, yaitu suatu informasi dikatakan berkualitas jika seluruh kebutuhan informasi telah tersampaiakan serta pesan yang disampaiakan sudah lengkap sesuai dengan yang diinginkan oleh user.
3. **Tepat Waktu (*timelines*)**, informasi yang datang pada penerima tidak boleh terlambat. Keterlambatan suatu informasi bisa berakibat fatal bagi suatu organisasi atau pemakainya hal ini dikarenakan informasi merupakan landasan dalam pengambilan keputusan.
4. **Ekonomis (*economy*)**, informasi yang dihasilkan harus mempunyai daya jual yang tinggi dan biaya operasional yang harus dikeluarkan untuk menghasilkan informasi tersebut harus minimal, informasi tersebut juga mapu memberikan dampak yang luas terhadap laju pertumbuhan ekonomi dan teknologi informasi.
5. **Efisien (*efficiency*)**, informasi yang berkualitas harus memiliki kalimat yang sederhana dan mudah dimengerti, tapi bisa memberikan makna yang mendalam.
6. **Dapat dipercaya (*reliability*)**, informasi yang didapat harus dari sumber yang bisa dipercaya.

2.9 Konsep Dasar Sistem Informasi

Sistem informasi dapat didefinisikan sebagi suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

2.9.1 Definisi Sistem Informasi

Menurut Sutabri (2004 : 36) “Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian

yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan".

Sedangkan menurut Kristanto (2003:6) mengemukakan Sistem informasi adalah kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti baik yang menerimanya.

Dikutip dari buku Al-Bahra (2005:13), sistem informasi dapat didefinisikan sebagai berikut:

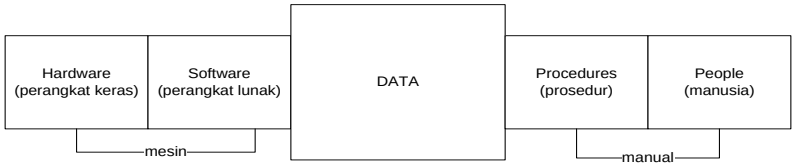
1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan / atau untuk mengendalikan organisasi.
3. Suatu sistem dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.9.2 Komponen Sistem Informasi

Adapun beberapa elemen / komponen dalam sistem informasi dalam buku Al-bahra (2005:14) dapat diklasifikasikan sebagai berikut:

1. *Hardware* dan *software* yang berfungsi sebagai mesin
2. *People* dan *procedures* yang merupakan manusia dan tata cara menggunakan mesin.

3. *Data* merupakan jembatan penghubung antara manuia dan mesin agar terjadi suatu proses pengolahan data.
4. *Input, Output, Process, Storage,* dan *Control*.



Gambar 2.8 Lima Komponen Sistem Informasi

[**Sumber**: Al-bahra bin Ladjamudin. 2005. *Analisis dan Desain Sistem Informasi.* Graha Ilmu. Yogyakarta.]

2.9.3 Kegiatan Sistem Informasi

Adapun kegiatan sistem informasi berdasarkan buku Al-Bahra (2005:22) :

1. **Input:**

   Menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.

2. **Proses:**

   Menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang berilai tambah.

3. **Output:**

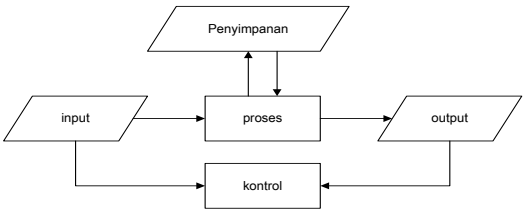
   Suatu kegiatan untuk menghasilkan laporan dari proses diatas tersebut.

4. **Penyimpanan:**

   Suatu kegiatan untuk memelihara dan menyimpan data.

5. **Kontrol:**

   Suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

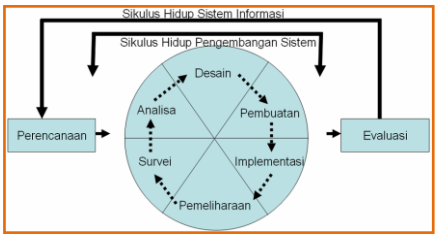


Gambar 2.9 Kegiatan Sistem Informasi

[**Sumber**: Azhar, Susanto. 2004. *Sistem Informasi Manajemen : Konsep dan pengembangannya*. Lingga Jaya. Bandung]

2.9.4 Siklus Hidup Sistem Informasi

Siklus hidup sistem informasi dimulai dari perencanaan, pengembangan (survei, analisa, desain, pembuatan, implementasi, pemeliharaan) dan dievaluasi secara terus menerus untuk mendapatkan apakah sistem informasi tersebut masih layak diaplikasikan, jika tidak, sistem informasi tersebut akan diganti dengan yang baru dan dimulai dari perencanaan kembali.



Gambar 2.10 Siklus Hidup Sistem Informasi

[**Sumber**: Kendal. 2003. *Analisis dan Perancangan Sistem Edisi ke-5.* Jakarta: Pearson Education Asia Pte. Ltd. dan PT Prenhallindo.]

2.10 Metode Pengembangan *Prototype*

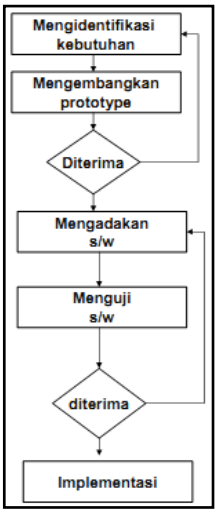
Adapun metode pengembangan yang digunakan adalah *prototype*, dalam buku Kendal (2003:221):

"*Prototyping* yaitu suatu teknik pengumpulan data yang sangat berguna untuk mengumpulkan informasi tertentu mengenai syarat-syarat tertentu mengenai syarat-syarat informasi pengguna secara cepat. *Prototyping* yang efektif seharusnya dilakukan pada awal-awal siklus pengembangan sistem, yakni selama fase penetapan syarat-syarat."

Sistem *prototype* merupakan bagian operasional dari sistem secara keseluruhan yang akan dibangun. *Prototype* merupakan suatu cara yang baik untuk mendapatkan umpan balik mengenai sistem yang diajukan dan mengenai bagaimana sistem tersebut tersedia untuk memenuhi kebutuhan informasi pengguna. Ada empat petunjuk yang harus diamati saat mengintegrasikan *prototype* kedalam fase penetapan siklus hidup pengembangan sistem, yakni:

1. Bekerja sesuai modul
2. Membangun *prototype* dengan cepat
3. Memodifikasi *prototype* dengan iterasi yang berurutan
4. Menekankan antar muka pengguna.

Model prototipe (*prototyping model*), dimulai dengan pengumpulan kebutuhan dan perbaikan, desain cepat, pembentukan prototipe, evaluasi pelanggan terhadap prototipe, perbaikan prototipe dan produk akhir.



Gambar 2.11 Metode *Prototype*

[**Sumber**: Kendal. 2003. *Analisis dan Perancangan Sistem Edisi ke-5.* Jakarta: Pearson Education Asia Pte. Ltd. dan PT Prenhallindo

Aktivitas atau langkah-langkah metode *prototype* adalah sebagai berikut:

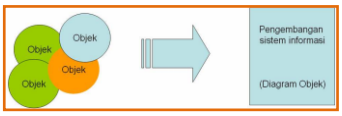
1. Mengidentifikasi kebutuhan: analisa terhadap kebutuhan calon *user*
2. *Quick design*: pembuatan desain global untuk membentuk s/w
3. *Build prototype*: pembuatan s/w *prototype* termasuk pengujian dan penyempurnaan
4. Evaluasi pelanggan
5. Pembuatan & implementasi: pembuatan sebenarnya termasuk *design*, *coding*, dan *testing*.

2.11 Metodologi Berorientasi Objek

Metodologi ini diperkenalksan sekitar tahun 1990 sebagai pelengkap untuk pemrograman yang terlebih dahulu telah mengadopsi metode berorientasi objek. Beberapa alat dan teknik yang dapat digunakan antara lain *dynamic* dan *static object-oriented model*, *state transisition diagram* dan *case scenario*.

Fokus utama metodologi ini pada objek, dengan melihat suatu sistem terdiri dari objek yang saling berhubungan dengan beberapa cara untuk mencapai suatu tujuan. Objek dapat digambarkan sebagai benda, orang, tempat dan sebaginya yang mempunyai atribut dan *method*. Obyek dalam '*software analysis & design*' adalah sesuatu berupa konsep (*concept*), benda (*thing*), dan sesuatu yang membedakannya dengan lingkungannya. Secara sederhana obyek adalah mobil, manusia, alarm dan lain-lainnya. Tapi obyek dapat pula merupakan sesuatu yang abstrak yang hidup didalam sistem seperti tabel, *database*, *event*, *system messages*. Obyek dikenali dari keadaannya dan juga operasinya. Sebagai contoh

sebuah mobil dikenali dari warnanya, bentuknya, sedangkan manusia dari suaranya. Ciri-ciri ini yang akan membedakan obyek tersebut dari obyek lainnya.



Gambar 2.12 Fokus Utama *Object Oriented*

[**Sumber**: Harianto, Bambang. 2004. *Rekayasa Sistem Berorientasi Objek*. Bandung: Informatika.]

Object Oriented Programming merupakan cara atau metode baru dalam membuat program. Langkah membuat program dengan teknik OOP, pertama kali mengetahui objeknya, kemudian atribut (*field*/*data*) dan perilakunya (fungsi). Berbeda dengan pemrograman prosedural (konvensional), membuat program memakai aspek *input*, proses, dan *output*. OOP adalah cara berpikir, pandangan, atau paradigm baru untuk membuat program/merancang sistem dengan memperhatikan objek, cirri objek, dan perilakunya. OOP ini sangat berbeda dengan program prosedural yang fokusnya aspek *input*, proses, dan *output*.

2.11.1 Karakteristik OOAD

Dalam pendekatan berorientasi objek ada 4 pilar utama yang harus dipahamai dalam pendekatan berorientasi objek yaitu karakteristik. Karakteristik (ciri) suatu program termasuk OOAD/OOP, apabila terdapat abstraksi, pembungkusan (*encapsulation*), polymorphisme, dan turunan (*inheritance*).

1. ***Abstraction***

   Kemampuan untuk menjadikan dalam bentuk yang lebih sederhana. Hal ini juga dikenal dalam metodologi pendekatan struktur yaitu dekomposisi seperti menyerderhanakan suatu sistem dalam bentuk *Context Diagram*.

2. *Encapsulation*

Merupakan suatu karakteristik OOAD dimana program terbungkus (jadi satu) data dan perilaku, artinya lebih memperhatikan aspek internal daripada aspek eksternal. Contoh: dalam program terdapat tombol *button close* didalamnya ada *method system.exit(0)* untuk keluar dari sistem java. Berbeda dengan metodologi terdahulu, metodologi ini menggabungkan atribut dan fungsi / proses kedalam suatu objek yang disebut dengan *encapsulation*. Setiap objek dapat "menyembunyikan" kompleksitasnya dan berhubungan dengan objek lain dengan mengirim "pesan / *message*" yang dapat dikenal dan diproses oleh objek penerima. Contoh: Pada dunia nyata, seorang ibu rumah tangga menanak nasi dengan menggunakan *rice cooker*, ibu tersebut menggunakannya hanya dengan menekan tombol. Tanpa harus tahu bagaimana proses itu sebenarnya terjadi. Disini terdapat penyembunyian informasi milik *rice cooker*, sehingga tidak perlu diketahui seorang ibu. Dengan demikian menanak nasi oleh si ibu menjadi sesuatu yang menjadi dasar bagi konsep information *hiding*.

3. *Polymorphisme*

Dengan kata lain suatu mekanisme yang memungkinkan suatu objek memiliki semua atau sebagian definisi dari objek induk. Menurut Bambang Hariyanto (2007:67) *Polymorphism* berasal dari kata Poly yang artinya banyak dan morph yang artinya bentuk. Jadi polymorphism adalah kemampuan suatu atribut atau method dapat berubah dalam

berbagai bentuk dalam implementasi. Contoh Pada obyek mobil, walaupun minibus dan truk merupakan jenis obyek mobil yang sama, namun memiliki juga perbedaan. Misalnya suara truk lebih keras dari pada minibus, hal ini juga berlaku pada obyek anak (*child*) melakukan metoda yang sama dengan algoritma berbeda dari obyek induknya. Hal ini yang disebut *polymorphism*, teknik atau konsep dasar lainnya adalah ruang lingkup/pembatasan. Artinya setiap obyek mempunyai ruang lingkup kelas, atribut, dan metoda yang dibatasi.

4. ***Inheritance***

   Merupakan suatu karakteristik OOAD di mana suatu kelas (*parent/base class*) dapat diturunkan ke kelas lain (*child/derived class*), sehingga kelas anak dapat memiliki data atau perilaku kelas orangtuanya. Contoh dengan beberapa buah mobil yang mempunyai kegunaan yang berbeda-beda. Ada mobil bak terbuka seperti truk, bak tertutup seperti sedan dan minibus. Walaupun demikian obyek-obyek ini memiliki kesamaan yaitu teridentifikasi sebagai obyek mobil, obyek ini dikatakan obyek induk (*parent*). Sedangkan minibus obyek anak (*child*), berarti semua operasi yang berlaku pada mobil berlaku pada minibus.

2.11.2 Ciri-Ciri *Object Oriented Programming* (OOP)

Menurut Bambang Hariyanto dalam bukunya yang berjudul Rekayasa Perangkat Lunak Berorientasi Objek, bahwa *Object Oriented* memiliki cirri-ciri sebagai berikut: Bambang Hariyanto (2007:67)

1. **Objek**

   Bentuk baik yang nyata atau tidak, seperti manusia, hewan, benda, konsep, aliran, dan lain-lain. Objek merupakan inisiasi (turunan langsung) dari suatu kelas.

2. **Kelas**

   Kumpulan objek yang memiliki kemiripan perilaku (*method*), cirri atau karakteristik (*property*). Contoh objek orang dari kelas manusia, potongan sebagai berikut: Manusia orang1=new manusia(“esson”);

3. **Method**

   Perilaku dari objek atau kelas tertentu. Merupakan perwujudan aksi atau tindakan dari dunia nyata di dalam pemrograman komputer.

4. **Konstruktor**

   Suatu fungsi yang dideklarasikan atau didefinisikan di dalam kelas, konstruktor harus mempunyai nama yang sama dengan fungsinya. Konstruktor dijalankan bersamaan dengan terciptanya kelas tersebut. Dalam suatu kelas bias terdapat lebih dari satu konstruktor. Konstruktor seperti method tetapi tidak mengembalikan nilai dan dapat didefinisikan tanpa parameter atau memakainya.

5. **De-konstruktor**

   Fungsi yang dideklarasikan dalam kelas, nama sama dengan nama fungsinya. Tetapi dijalankan bersamaan dengan dimusnahkannya kelas tersebut.

6. **Karakteristik / properties**

   Ciri yang dimiliki oleh suatu objek, karakteristik ini juga sebagai pembeda objek satu dengan objek lainnya dalam kelas yang sama (konsep individu).

7. **Variabel**

   Tempat menampung data sementara, dalam pemrograman objek biasanya disebut data, sedangkan dalam pemrograman prosedural sering disebut dengan variabel.

8. **Data**

   Istilah lain dari variabel dalam OOP. Dalam pemrograman java bisa juga disebut *field*, data member atau *instance variable*.

9. **Hak akses (*access attribute*)**

   Hak akses digunakan untuk dapat menentukan *data member* mana yang dapat digunakan oleh kelas lain, dan mana yang tidak dapat digunakan. Hak akses ini sangat penting dalam membuat program turunan kelas.

   a. *Public*

      Data member atau *variable* dapat diakses dari kelas mana saja

   b. *Protected*

      Dapat mengakses *data member* dari kelas dalam *package* yang sama dan subkelasnya

   c. *Private*

      Kelas yang data membernya memakai *private* hanya dapat digunakan oleh kelas bersangkutan, tidak dapat digunakan kelas lain

d. Tidak disebutkan

Data member dapat diakses dari kelas dalam *package* yang sama.

2.12 *Unified Modeling Language*

Untuk mendapatkan spesifikasi perangkat lunak yang sesuai dengan keinginan dan kebutuhan pengguna, para pengembang melakukan pemodelan-pemodelan secara visual. Langkah ini sering dinamakan pemodelan visual. Pemodelan visual adalah proses penggambaran informasi-informasi secara grafis dengan notasi-notasi yang telah disepakati sebelumnya. Nugroho (2005:16-17).

Tujuan utama pemodelan visual adalah untuk memungkinkan adanya komunikasi antara pengguna, pengembang, penganalisis, *tester*, *manager*, dan siapapun yang terlibat dalam proyek, menunjukkan interaksi antara pengguna dengan sistem, obyek-obyek dalam sistem, dan antar sistem itu sendiri. Roques (2004:1-5). Untuk melakukan pemodelan sistem atau perangkat lunak, kita akan menggunakan notasi-notasi UML (*Unified Modeling Language*) yang kita gambarkan lewat sarana perangkat lunak *Rational Rose*.

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*. Martin (2005:1).

Pada tahap analisis, meliputi usaha untuk mengetahui apa kemampuan sebuah sistem yang diinginkan pengguna dan pelanggan dari sebuah perangkat lunak. Beberapa teknik yang dapat membantu dalam tahapan analisis. Martin (2005:44):

Untuk itulah rancangan dibangun untuk memberikan gambaran. Semakin kompleks sebuah sistem semakin penting penggunaan teknik pemodelan yang baik. Dengan menggunakan model, diharapkan pengembangan piranti lunak dapat memenuhi semua kebutuhan pengguna dengan lengkap, cepat dan tepat. Adapun diagram utama dalam UML adalah:

1. *Use case Diagram*

   Digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. *Use case* merepresentasikan operasi-operasi yang dilakukan oleh *actor*. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnya. *Actor* yang melakukan operasi dihubungkan dengan garis lurus ke *use case*.

2. *Class Diagram*

   Merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class diagram* menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

3. *Statechart Diagram*

   Untuk memodelkan perilaku *objects* di dalam sistem.

4. *Activity Diagram*

   Untuk memodelkan perilaku *Use Cases* dan *objects* di dalam *system*

5. *Sequence Diagram*

   Menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*: interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.

6. *Collaboration Diagram*

   Dipakai untuk memodelkan interaksi antar *object* di dalam sistem. Berbeda dengan *sequence diagram* yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, *collaboration* diagram lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh *object*.

7. *Component Diagram*

   Untuk memodelkan komponen *object*.

8. *Deployment Diagram*

   Untuk memodelkan distribusi aplikasi.

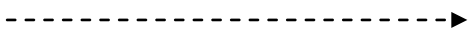
Sedangkan pada tahapan perancangan / desain, dapat menggunakan notasi yang lebih banyak dan lebih tepat dalam notasi. Berikut adalah beberapa teknik dalam tahap perancangan (Martin 2005:45).

Pemodelan adalah proses merancang piranti lunak sebelum melakukan pengkodean, yang diwujudkan dalam bentuk bahasa model, model ini mampu

menggambarkan persoalan dalam beberapa sudut pandang. Piranti lunak dapat dianalogikan seperti pembuatan *blueprint* pada pembangunan gedung. Membuat model dari sebuah sistem yang kompleks sangatlah penting, karena kadang kita sulit memahami sistem yang akan di bangun tersebut seperti apa, dan bagaimana cara kerjanya.

2.12.1 *Relationship*

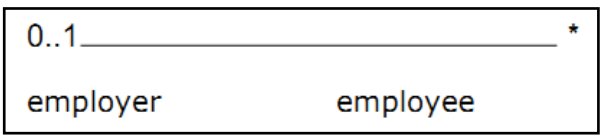
Pertama, sebuah *dependency* adalah hubungan semantik antara dua benda/things yang mana sebuah benda berubah mengakibatkan benda satunya akan berubah pula. Umumnya sebuah *dependency* digambarkan sebuah panah dengan garis terputus-putus seperti terlihat dalam gambar di bawah ini.



Gambar 2.13 *Dependency*

[**Sumber:** Fowler, Martin. 2005. Panduan Singkat Bahasa Pemodelan Objek Standar. Yogyakarta: Andi.]

Kedua, sebuah *association* adalah hubungan antar benda struktural yang terhubung diantara obyek. Kesatuan obyek yang terhubung merupakan hubungan khusus, yang menggambarkan sebuah hubungan struktural diantara seluruh atau sebagian. Umumnya *assosiation* digambarkan dengan sebuah garis yang dilengkapi dengan sebuah label, nama, dan status hubungannya.



Gambar 2.14 *Association*

[**Sumber:** Fowler, Martin. 2005. Panduan Singkat Bahasa Pemodelan Objek Standar. Yogyakarta: Andi.]

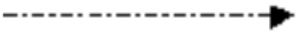
Ketiga, sebuah *generalization* adalah menggambarkan hubungan khusus dalam obyek anak/*child* yang menggantikan obyek *parent* / induk . Dalam hal ini, obyek anak memberikan pengaruhnya dalam hal struktur dan tingkah lakunya kepada obyek induk.



Gambar 2.15 *Generalization*

[**Sumber:** Fowler, Martin. 2005. Panduan Singkat Bahasa Pemodelan Objek Standar. Yogyakarta: Andi.]

Keempat, sebuah *realization* merupakan hubungan semantik antara pengelompokkan yang menjamin adanya ikatan diantaranya. Hubungan ini dapat diwujudkan diantara *interface* dan kelas atau *elements*, serta antara *use cases* dan *collaborations*.



Gambar 2.16 *Realization*

[**Sumber:** Fowler, Martin. 2005. Panduan Singkat Bahasa Pemodelan Objek Standar. Yogyakarta: Andi.]

2.13 Jaringan Komputer

Sub bab ini akan membahas mengenai jaringan komputer, yang berisi mengenai topologi, jenis dan konfigurasi jaringan komputer yang mungkin dibutuhkan dalam perancangan sistem maupun aplikasi.

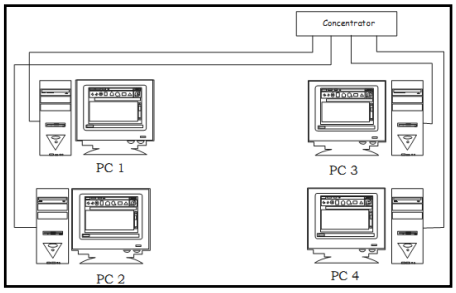
2.13.1 Terminologi Dasar Jaringan Komputer

Jaringan komputer dibangun dalam bentuk dan ukuran berbeda-beda, bergantung kondisi dan kebutuhan individu yang menyelenggarakan. Tahun demi tahun, industri *networking* berkembang demikian pesat sehingga ditemukan

beragam tipe dan desain. Inilah yang disebut dengan *networking terminology*. Keanekaragaman ini semakin memberi alternatif bagi kita untuk membangun jaringan sesuai dengan rancangan yang dikehendaki. Berikut adalah beberapa varian *"area network"*, seperti yang dijelaskan oleh Rahmat (2003:2) dalam bukunya yang berjudul Panduan Membangun Jaringan Komputer Untuk Pemula.

1. **LAN (*Local Area Network*)**

   Merupakan rancangan dasar jaringan komputer, dan dari sinilah kita dapat memahami rancangan-rancangan lainnya. LAN berupa 2 komputer atau lebih yang dihubungkan satu sama lain melalui sebuah media (kabel jaringan, komunikasi wireless, dan lain-lain) sehingga setiap node komputer dapat saling melakukan akses. LAN mengkoneksikan *device-device* jaringan secara langsung dalam jarak dekat.



Gambar 2.17 *Local Area Network* (LAN)

[**Sumber**: Rafiudin, Rahmat. 2003. *Panduan Membangun Jaringan Komputer Untuk Pemula.* PT Elex Media Komputindo. Jakarta]

2. **WAN (*Wide Area Network*)**

   Wide Area Networks (WAN) adalah jaringan yang lingkupnya biasanya sudah menggunakan sarana satelit ataupun kabel bawah laut sebagai contoh keseluruhan jaringan suatu Bank yang ada di Indonesia ataupun

yang ada di negara-negara lain menggunakan sarana WAN. Sebuah Bank yang ada di Bandung bisa menghubungi kantor cabangnya yang ada di Jepang, hanya dalam beberapa menit. Biasanya WAN agak rumit dan sangat kompleks, menggunakan banyak sarana untuk menghubungkan antara LAN dan WAN ke dalam komunikasi global seperti Internet. (Rahmat 2003).

3. ***Metropolitan Area Network*** **(MAN)**

   Sebuah MAN, biasanya meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu provinsi.

4. ***Global Area Network*** **(GAN)**

   GAN merupakan suatu jaringan yang menghubungkan negara-negara di seluruh dunia. Contoh yang sangat baik dari GAN ini adalah Internet.

2.13.2 Topologi Jaringan

Topologi menggambarkan strukutur jaringan, atau bagaimana sebuah jaringan didesain. Terdapat dua definisi topologi (Rahmat 2003:):

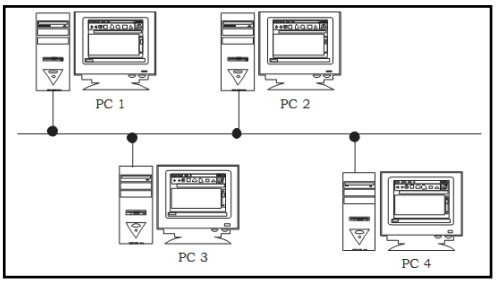
1. *Physical topology*, merupakan layout *actual* kabel-kabel jaringan
2. *Logical topology*, mendefinisikan bagaimana media diakses oleh host.

Adapun topologi fisik yang umum digunakan dalam membangun sebuah jaringan adalah :

1. **Topologi *Linear Bus***

   Topologi sebuah *Linear BUS* / garis lurus terdiri dari satu jalur kabel utama (*backbone*) dimana pada masing-masing ujungnya diberikan

sebuah *terminator*. Semua *nodes* pada jaringan (*file server, workstation*, dan perangkat lainnya) terkoneksi ke sebuah kabel utama.

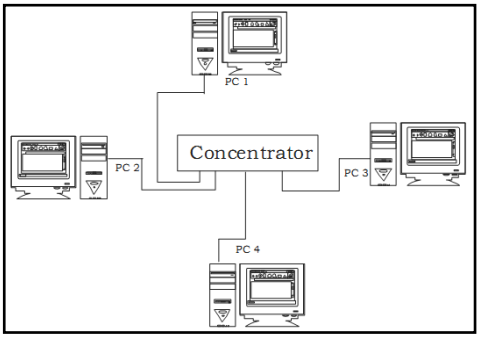


Gambar 2.18 Topologi Linear Bus

[**Sumber**: Rafiudin, Rahmat. 2003. *Panduan Membangun Jaringan Komputer Untuk Pemula.* PT Elex Media Komputindo. Jakarta.]

2. Topologi *Star*

Topologi model ini didesain dimana setap *node* (*file server, workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator* baik berupa HUB ataupun SWITCH. Data yang terkirim ke jaringan akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengontrol keseluruhan fungsi jaringan. Dia juga bertindak sebagai *repeater* dalam skala kecil dari aliran data. Konfigurasi pada jaringan model ini menggunakan kabel *Twisted Pair*. Gambar 2.19 menunjukkan topologi jaringan *Star*.

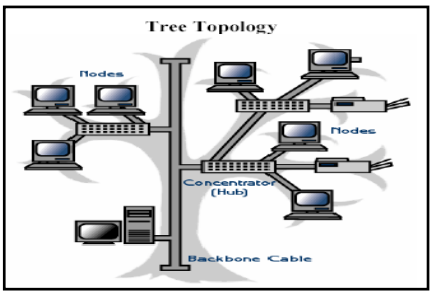


Gambar 2.19 Topologi Star

[**Sumber**: Rafiudin, Rahmat. 2003. *Panduan Membangun Jaringan Komputer Untuk Pemula.* PT Elex Media Komputindo. Jakarta.]

3. Topologi *Tree*

Sebuah topologi model ini merupakan perpaduan antara topologi *Linear BUS* / Garis Lurus dan Bintang, yang terdiri dari kelompok-kelompok dari *workstation*.

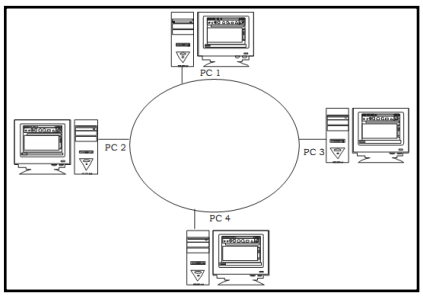


Gambar 2.20 Topologi Tree

[**Sumber**: http://blog.unsri.ac.id/userfiles/09071003013.doc]

4. Topologi *Ring*

Dalam topologi *Ring* semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin.



Gambar 2.21 Topologi Ring

[**Sumber**: Rafiudin, Rahmat. 2003. *Panduan Membangun Jaringan Komputer Untuk Pemula.* PT Elex Media Komputindo. Jakarta.]

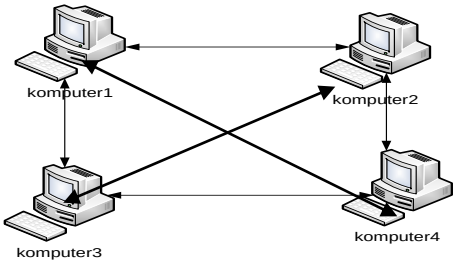
2.13.3 Tipe dan Arsitektur Jaringan

Untuk bisa membangun sebuah jaringan komputer, maka perlu dipahami tipe dan arsitektur jaringan sesuai dengan kondisi tempat. Hal ini penting karena tipe dan arsitekrur sebuah jaringan menentukan perangkat apa yang harus disediakan untuk membangun jaringan tersebut. Wahana Komputer dan Andi (2001:3).

Menurut fungsi komputer pada sebuah jaringan dalam buku pintar penanganan jaringan komputer (2001:3), maka tipe jaringan komputer dapat dibedakan menjadi dua tipe, yaitu:

1. **Jaringan *peer to peer***

   Yaitu, setiap komputer yang terhubung pada jaringan dapat berkomunikasi dengan komputer-komputer lain secara langsung tanpa melalui komputer perantara. Pada jaringan tipe ini sumber daya komputer terbagi pada seluruh komputer yang terhubung dalam jaringan tersebut, baik yang sumber daya yang berupa perangkat keras maupun perangkat lunak dan datanya.



Gambar 2.22 Tipe Jaringan *Peer to Peer*

[Sumber: Wahana Komputer dan Andi. 2001. *Buku Pintar Penanganan Jaringan Komputer*. Wahana Komputer dan Andi. Semarang, Yogyakarta.**]**

2. Jaringan *Client – Server*

Dalam buku pintar Penanganan Jaringan Komputer (2001:4) pada jaringan *client-server* terdapat sebuah komputer yang berfungsi *server* sedangkan komputer-komputer yang lain berfungsi sebagai *client*. Sesuai namanya maka maka komputer *server* berfungsi dan bertugas melayani seluruh komputer yang terdapat dalam jaringan tersebut. Adapun pelayanan yang diberikan komputer *server* ini adalah:

a. *Disk Sharing*, yaitu berupa penggunaan kapasitas disk secara bersama-sama pada komputer *client*.
b. *Print sharing*, berupa penggunaan *printer* secara bersamaan.
c. Penggunaan perangkat lain secara bersama demikian pula dengan data dan sistem aplikasi yang ada.
d. Mengatur keamanan jaringan dan data dalam jaringan.
e. Mengatur dan mengontrol hak dan waktu akses perangkat-perangkat yang ada dalam jaringan.

Pada sebuah jaringan komputer dimungkinkan untuk digunakannya lebih dari satu komputer *server*, bahkan dengan kemampuan dan fasilitas yang

berbeda. Dalam sebuah jaringan biasanya workstation menggunakan komputer yang memiliki kemampuan lebih rendah dari komputer *server*, meskipun tidak selalu demikian. Beberapa jaringan multi *user*, seperti komputer *mainframe*, menggunakan dumb terminal sebagai *workstation*.

2.14 Perangkat Lunak Pendukung

Berikut ini merupakan penjelasan mengenai perangkat-perangkat lunak pendukung yang penulis pakai dalam merancang maupum membangun aplikasi.

2.14.1 JAVA

Bahasa Java diciptakan oleh James Gosling dan Patrick Naughton dalam suatu proyek dari SUN Microsystem. Pada mulanya ingin diberi nama OAK dari pohon yang terdapat pada kantor James Gosling, tetapi kata OAK telah ada pada sun microsystem, maka diberi nama java (terinspirasi minum kopi). Browser pertama yang dapat membaca java adalah Hot Java. Setelah Browser Netscape dari perusahaan Netscape Navigator dan IE dari perusahaan Microsoft Inc dapat membaca script java, maka bahasa java semakin popular. Vendor-vendor lain, seperti IBM, Oracle, Symantec, Inprise (Borland Inc), dan perusahaan-perusahaan mobile, seperti Nokia, Sony Ericsson, Motorola, dan Samsung juga mengadopsi teknologi java.

2.14.2 MySQL

Menurut Peranginangin (2006:381) "MySQL merupakan suatu *Relational Database Management System* (RDBMS) yang mendukung *database* yang terdiri dari sekumpulan relasi atau tabel". MySQL bekerja pada berbagai sistem operasi

dan banyak bahasa. MySQL bekerja dengan cepat dan baik dengan data yang besar karena menggunakan suatu format standar SQL bahasa data yang terkenal.

2.14.3 Rational Rose

Menurut Adi Nugroho (2005:20-21) dalam bukunya yang berjudul Rational Rose untuk Pemodelan Berorientasi Objek, menerangkan bahwa Rational Rose adalah kakas (tools) pemodelan visual untuk pengembangan sistem berbasis objek yang sangat handal untuk digunakan sebagai bantuan bagi para pengembang dalam melakukan analisis dan perancangan sistem. Rational Rose digunakan untuk melakukan pemodelan sistem sebelum pengembangan menulis kode-kode dalam bahasa pemrograman tertentu. Rational Rose mendukung pemodelan bisnis yang membantu para pengembang untuk memahami sistem secara komprehensif. Rational Rose juga membantu analisis sistem dengan cara pengembang membuat diagram use case untuk melihat fungsionalitas sistem secara keseluruhan sesuai dengan harapan dan keinginan pengguna, juga menuntut pengembang untuk mengembangkan *Interaction Diagram* untuk melihat bagaimana objek-objek saling bekerjasama dalam menyediakan fungsionalitas yang diperlukan.

Dalam Rational Rose, pemodelan adalah cara melihat sistem dari berbagai sudut pandang mencakup semua program yang dikenal dalam UML, aktor-aktor yang terlibat dalam sistem, *use-case*, objek-objek, kelas-kelas, komponen-komponen, serta simpul-simpul penyebaran (*deployment node*).