BAB 2

LANDASAN TEORI

2.1 Tinjauan Perusahaan

2.1.1 Sejarah Perusahaan

Pada awal tahun 2005 percetakaan di kota cilacap belum ramai dan hanya beberapa percetakaan yang mampu bertahan menjalankan usahanya sampai sekarang. Seiring waktu banyak toko-toko yang mulai marak dan investor dari berbagai kota pun berdatangan untuk membuka lahan bisnisnya di kota cilacap ini.

Pada awal tahun 2007 ada sekumpulan anak muda yang memiliki kreatifitas, semangat untuk maju dan dari berbagai latar belakang yang berbeda berkumpul dan berbincang-bincang mengenai peluang usaha yang bagus di kota Cilacap. Ada yang mengatakan bahwa bisnis cetak digital di cilacap akan ramai dikarenakan lom ada yang mempunyai mesin cetak digital di wilayah tersebut. Kemudian mencari tempat yang tepat untuk membuka usaha tersebut yaitu di jalan bali no.67 Cilacap.

Tempat berdirinya CV.Bali67 tersebut awalnya hanya sebuah rumah dengan nama pemilik Hj. Siti Rodjiyah, kemudian direnofasi bagian rumah depan untuk usaha percetakaan. Pada tanggal 7 Februari 2007 anak-anak tersebut berkumpul dan berunding untuk menentukan dewan direksi, dan menentukan layanan apa saja yang akan diberikan oleh CV. Bali67 *Advertising* ini. Kemudian pada tanggal 9 Februari 2007, Djonny W.M., Donni Tri Prasetio, dan Lukmannul Hakim berangkat menuju notaris Naimah di jalan Jend. Ahmad

Yani No. 147 untuk mengesahkan perusahaan tersebut menjadi CV. Bali67 *Advertising*. Perusahaan tersebut merupakan percetakaan pertama di Cilacap yang melayani cetak digital.

2.1.2 Logo Perusahaan



Gambar 2.1 Logo Perusahaan

Nama CV. Bali67 *Advertising* berasal dari alamat perusahaan bertempat, dan advertising diambil karena layanan yang diberikan oleh CV. Bali67 advertising adalah berupa percetakan dan periklanan.

2.1.3 Visi

Menjadi perusahaan teknologi informasi dan komunikasi visual yang terdepan di Banyumas pada umumnya dan kota Cilacap pada khususnya.

2.1.4 Misi

Mengembangkan kreatifitas, meningkatkan mutu layanan, cepat, akurat dan tepat waktu dalam hal teknologi informasi dan komunikasi visual.

2.1.5 Tempat dan Kedudukan Perusahaan

Lokasi Usaha :

Jl. Bali No.67 Kebonmanis, Cilacap Utara, Cilacap 53235 - Indonesia

No Telp : (0282)545538

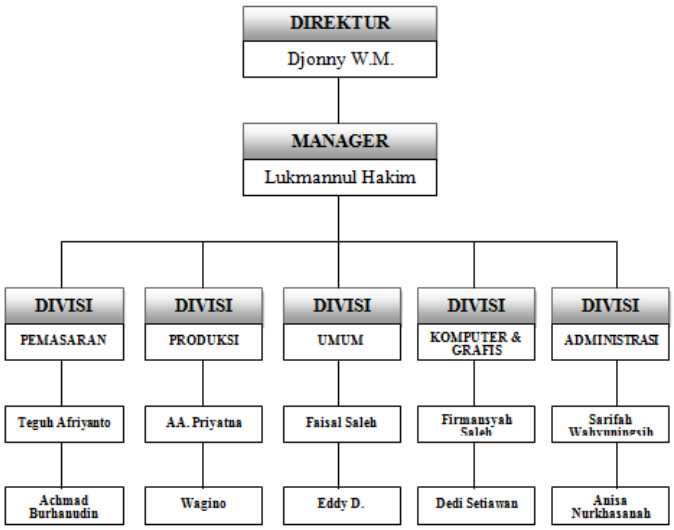
2.1.6 Bentuk dan Badan Hukum Perusahaan

CV. Bali67 *Advertising* merupakan perusahaan persekutuan komanditer (CV). CV. Bali67 *Advertising* didirikan oleh beberapa orang yang bergabung dan memberikan modal untuk membuka usaha percetakaan bersama-sama yang merupakan percetakan dan *digital printing* pertama kali didirikan di kota cilacap. Menurut Ibu Naimah, SH, MH. Selaku notaris yang beralamat di Jl. Jend. Ahmad Yani No.147 (1000) Cilacap, CV. Bali67 *Advertising* menandatangani akta pendirian pada tanggal 7 Februari 2007 oleh pengurus perusahaan Djonny Widyaputra Markam dan terdaftar di pengadilan pada tanggal 9 februari 2007.

2.1.7 Bidang Pekerjaan Perusahaan

Pelayanan yang diberikan oleh CV. Bali67 *Advertising* adalah percetakan dan sablon, *digital printing*, profil perusahaan, presentasi multimedia, desain grafis dan logo, *audio video auditing*, *gift dan souvenir*.

2.1.8 Struktur Organisasi Perusahaan



Gambar 2.2 Struktur Organisasi

2.2 Landasan Teori

2.2.1 Konsep Dasar Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.[6]

Dari pengertian diatas dapat diambil suatu kesimpulan bahwa suatu sistem merupakan elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

2.2.1.1 Karakteristik Sistem

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu memiliki komponen-komponen *(components)*, batas sistem *(boundary)*, lingkungan luar sistem *(environment)*, penghubung *(interprest)*, masukan *(input)*, keluaran *(output)*, pengolah *(process)* dan sasaran *(objective)* dan tujuan *(goal)* [1].

1. Komponen Sistem (*System Components*)

   Komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sistem baik besar maupun kecil, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu yang lebih besar yang disebut *supra system.*

2. Batas Sistem (*System Boundary*)

   Batas sistem merupakan daerah-daerah yang membatasi antara satu sistem dengan sistem lainnya dengan lingkungan luarnya. Batas sistem ini

memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menujukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*System Environment*)

   Lingkungan luar sistem dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem yang dapat bersifat menguntungkan dan dapat pula merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup sistem.

4. Penghubung Sistem

   Penghubung sistem merupakan media penghubung antara satu sub sistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung akan terjadi interaksi antar subsistem, sehingga membentuk satu kesatuan.

5. Masukan Sistem *(System Input)*

   Masukan adalah suatu energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Contoh *maintenance input* di dalam sistem komputer adalah program, yang digunakan untuk mengoperasikan komputer. Sedangkan *signal input* adalah energi yang diproses untuk mendapatkan keluaran. Contoh *signal input* di dalam sistem komputer adalah data, yang dapat diolah menjadi Informasi.

6. Keluaran Sistem *(System Output)*

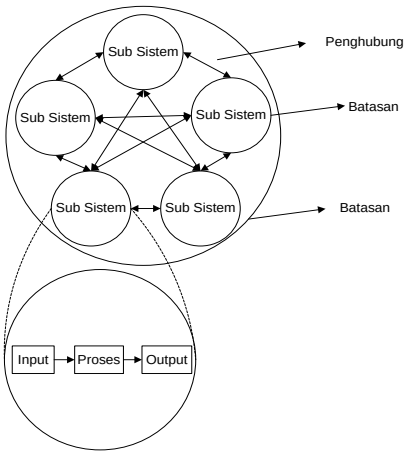
   Keluaran (*Output*) merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

7. Pengolah Sistem (*System Process*)

   Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya, yang bertugas untuk merubah masukan menjadi keluaran.

8. Sasaran Sistem (*System Objective*)

   Suatu sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*). Suatu operasi sistem akan berguna dan berhasil apabila mencapai sasaran atau tujuannya. Sasaran sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.



Gambar 2.3 Karakteristik system

2.2.1.2 Jenis Sistem

Ada berbagai tipe sistem berdasarkan kategori:

1. Atas dasar keterbukaan yaitu :
   1. Sistem terbuka, dimana pihak luar dapat mempengaruhinya.
   2. Sistem tertutup.

2. Atas dasar komponen yaitu :
    1. Sistem fisik, dengan komponen materi dan energi.
    2. Sistem non-fisik atau konsep, berisikan ide-ide.

2.2.2 Data dan Informasi

"Data adalah apapun dan semua fakta yang dikumpulkan, disimpan, dan diproses oleh suatu sistem informasi, sedangkan informasi adalah data yang telah diatur dan diproses sehingga dapat memiliki arti" (Romney, 2009, p.27)

Menurut Romney (2009), informasi adalah data yang telah diatur dan diproses sehingga memberikan manfaat bagi para penggunanya. Informasi dibutuhkan untuk mengambil keputusan atau untuk meningkatkan proses pengambilan keputusan. Ketika kuantitas dan kualitas informasi meningkat, user dapat membuat keputusan yang lebih baik.

2.2.2.1 Data

Data adalah kumpulan dari fakta-fakta, kejadian-kejadian yang dapat berupa simbol, angka, huruf, dan lain-lain yang berguna bagi suatu pengolahan data (*process*) atau sebagai masukan (*input*) bagi suatu proses.

2.2.2.1.1 Model Data

Data yang disimpan menggambarkan beberapa aspek dari suatu organisasi. Model data adalah himpunan deksripsi data level tinggi yang dikonstruksi untuk menyembunyikan beberapa detail dari penyimpanan level rendah. Beberapa manajemen basis data didasarkan pada model data relasional, model data hirarkis, atau model data jaringan.

2.2.2.1.1.1 Model Data Hirarkis

Model hirarkis biasa disebut model pohon, karena menyerupai pohon yang dibalik. Model ini menggunakan pola hubungan orang tua-anak. Setiap simpul (biasa dinyatakan dengan lingkaran atau kotak) menyatakan sekumpulan medan. Simpul yang terhubung ke simpul pada level di bawahnya disebut orang tua. Setiap orang tua bisa memiliki satu (hubungan 1:1) atau beberapa anak (hubungan 1:N), tetapi setiap anak hanya memiliki satu orang tua. Simpul – simpul yang dibawahi oleh simpul orang tua disebua anak. Simpul orang tua yang tidak memiliki orang tua disebut akar. Simpul yang tidak mempunyi anak disebut daun. Adapun hubungan antara anak dan orang tua disebut cabang.

2.2.2.1.1.2 Model Data Jaringan

Model jaringan distandarisasi pada tahun 1971 oleh *Data Base Task Group* (*DBTG*). Itulah sebabnya disebut model *DBTG*. Model ini juga disebut model *CODASYL* (*Conference on Data System Languages*), karena *DBTG* adalah bagian dari *CODASYL*.

Model ini menyerupai model hirarkis, dengan perbedaan suatu simpul anak bisa memilki lebih dari satu orang tua. Oleh karena sifatnya demikian, model ini bias menyatakan hubungan 1:1 (satu arang tua punya satu anak), 1:M (satu orang tua punya banyak anak), maupun N:M (beberapa anak bisa mempunyai beberapa orangtua). Pada model jaringan, orang tua diseut pemilik dan anak disebut anggota.

2.2.2.1.1.3 Model Data Relasional

Model relasional adalah model data yang paling banyak digunakan saat ini. Pembahasan pokok pada model ini adalah relasi, yang dimisalkan sebagai

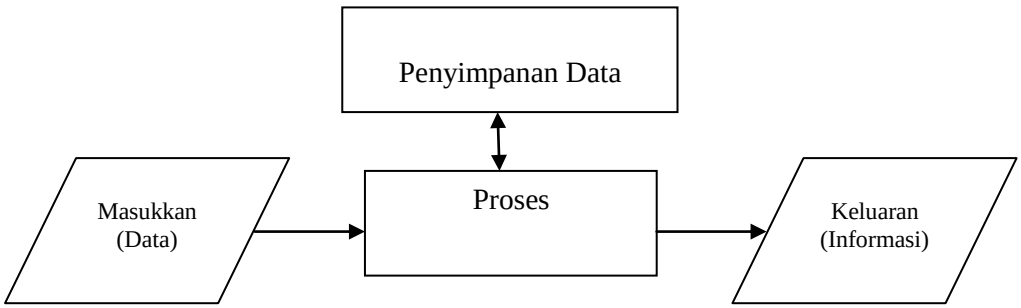
himpunan dari record. Deskripsi data dalam istilah model data disebut skema. Pada model relasional, skema untuk relasi ditentukan oleh nama, nama dari tiap field (atau atribut atau kolom), dan tipe dari tiap field.

2.2.2.2 Informasi

Dari suatu pendapat dikatakan bahwa:

"*Informasi adalah data yang diolah menjadi suatu bentuk tertentu yang lebih berguna dan lebih berarti bagi penerimanya".*

Berikut akan diperlihatkan gambar mengenai hubungan antara data dengan informasi :



Gambar 2.4 Perubahan data menjadi informasi dalam sistem informasi

Informasi merupakan hasil pengolahan dari sebuah model, formasi, organisasi, ataupun suatu perubahan bentuk dari data yang memiliki nilai tertentu, dan bisa digunakan untuk menambah pengetahuan bagi yang menerimanya. Dalam hal ini, data bisa dianggap sebagai obyek dan informasi adalah suatu subyek yang bermanfaat bagi penerimanya. Informasi juga bisa disebut sebagai hasil pengolahan atau pemrosesan data.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata, tetapi data merupakan bentuk yang masih mentah dan belum dapat memberikan arti banyak

bagi pemakai, sehingga perlu diolah lebih lanjut untuk menghasilkan informasi yang dibutuhkan. Dengan kata lain informasi adalah hasil dari pengolahan data.

Informasi dapat berupa dokumen, laporan, atau jawaban suatu pertanyaan. Dokumen merupakan catatan transaksi atau data perusahaan, sedangkan laporan merupakan informasi yang digunakan untuk membantu penagambilan keputusan yang baik.

Terdapat enam karakteristik yang membuat informasi menjadi berguna dan berarti :

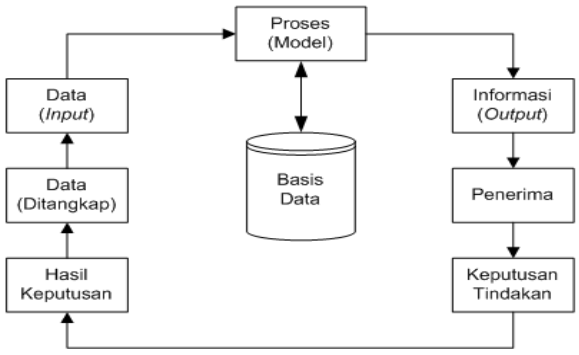
1. *Relevant,* informasi adalah relevan bila dapat mengurangi ketidakpastian, meningkatkan kemampuan pengambilan keputusan dalam membuat prediksi, atau memastikan, membenarkan perkiraan.
2. *Reliable,* informasi adalah *reliable* bila bebas dari kesalahan atau bias dan secara cepat menampilkan kejadian atau aktifitas organisasi.
3. *Complete,* Informasi adalah complete apabila dapat mencakup aspek-aspek penting dari kejadian atau aktifitas yang diukurnya.
4. *Timely,* Informasi adalah *timely* bila dapat menyediakan tepat waktu bagi para pembuat keputusan untuk menggunakannya dalam membuat keputusan.
5. *Understandable,* informasi ditampilkan dengan format yang dapat dibaca dan dimengerti oleh pemakai.
6. *Verifiable,* Informasi adalah *verifiable* bila dua orang yang berpengetahuan masing-masing menghasilkan informasi yang sama.

2.2.2.2.1 Nilai Informasi

Nilai Informasi (*value of information*) ditentukan oleh dua hal yaitu manfaat dan biaya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efekif dibandingkan dengan biaya mendapatkannya.

2.2.2.2.2 Siklus Informasi

Agar data dapat menjadi lebih berarti dan berguna dalam bentuk Informasi, maka perlu diolah melalui suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus ini disebut dengan siklus Informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*).



Gambar 2.5 Siklus informasi

2.2.3 Konsep Dasar Sistem Informasi

Menurut *Robert A. Leitch* dan *K. Roscoe Davis* di dalam bukunya *Accounting Informatioon Systems* mendefinisikan sistem informasi sebagai berikut:

> *"Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan*".

Sedangkan menurut *Susanto Azhar*:

> "*Sistem Informasi adalah kumpulan dari sub-sub sistem komponen baik phisik maupun non phisik yang saling berhubungan satu sama lain dan bekerjasama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna*".

Dari definisi diatas dapat diambil kesimpulan bahwa sistem informasi merupakan perpaduan antara manusia, alat teknologi, media, prosedure dan pengendalian yang bertujuan untuk menata jaringan komunikasi sehingga dapat membantu dalam pengambilan keputusan yang tepat. Kegiatan yang terdapat pada sistem informasi antara lain :

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

Sistem informasi dalam sebuah sistem meliputi pemasukan data (*input*) kemudian diolah melalui suatu model dalam pemrosesan data, dan hasil informasi

akan ditangkap kembali sebagai suatu input dan seterusnya sehingga membentuk siklus informasi yang dapat diperoleh dari sistem informasi sebagai sistem khusus dalam organisasi untuk mengolah informasi tersebut.

2.2.3.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building blok*) yaitu:

a. *Hardware* yaitu suatu perangkat keras dalam komputer yang kita bisa sentuh dan rasakan.
b. *Software* yaitu suatu perangkat lunak di dalam komputer yang berfungsi untuk mengoperasikan suatu aplikasi di dalam sistem komputer.
c. Data yaitu sekumpulan karakter yang diterima sebagai masukan (*input*) untuk sistem informasi dan disimpan serta diolah.
d. Prosedur yaitu suatu urutan pekerjaan tata usaha yang biasanya melibatkan beberapa orang dalam satu bagian atau lebih, dan disusun untuk menjamin adanya perlakuan yang seragam terhadap transaksi-transaksi perusahaan yang terjadi.
e. *User* yaitu orang yang terlibat dalam sistem informasi seperti operator, pemimpin sistem informasi, dan sebagainya.

2.2.3.2 Tujuan Sistem Informasi

Sistem Informasi memiliki beberapa tujuan, yaitu:

1. Integrasi sistem
    a. Menghubungkan sistem individu/kelompok
    b. Pengkolektifan data dan penyambungan secara otomatis
    c. Peningkatan koordinasi dan pencapaian sinergi

2. Efisiensi pengelolaan
    a. Penggunaan basis data dalam upaya kesamaan keadministrasian data.
    b. Pengelolaan data berkaitan dengan karakteristik Informasi.
    c. Penggunaan dan pengambilan Informasi.
3. Dukungan keputusan untuk manajemen
    a. Melengkapi Informasi guna kebutuhan proses pengambilan kebutuhan
    b. Akuisisi Informasi eksternal melalui jaringan komunikasi
    c. Ekstraksi dari Informasi internal yang terpadu.

2.2.3.3 Manfaat Sistem Informasi

Sistem Informasi memiliki beberapa manfaat, yaitu:

1. Menghemat tenaga kerja
2. Peningkatan efisiensi
3. Mempercepat proses
4. Perbaikan dokumentasi
5. Pencapaian standar
6. Perbaikan keputusan

2.2.4 Manajemen Persediaan

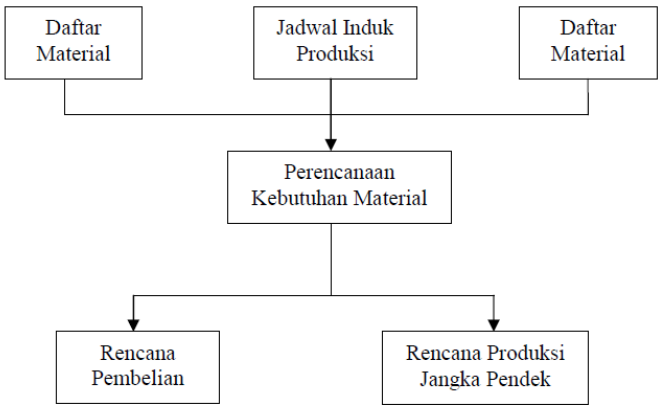
Persediaan adalah bahan atau barang yang disimpan yang akan digunaka untuk memenuhi tujuan tertentu, misalnya untuk proses produksi atau perakitan , untuk dijual kembali, dan untuk suku cadang dari suatu peralatan atau mesin. Persediaan dapat berupa bahan mentah, bahan pembantu, barang dalam proses dan barang jadi [3].

Sistem pengendalian persediaan dapat didefinisikan sebagai serangkaian kebijakan pengendalian untuk menentukan tingkat persediaan yang harus dijaga,

kapan pesanan untuk menambah persediaan harus dilakukan dan berapa bessar pesanan harus diadakan. Sistem ini menentukan dan menjamin tersedianya persediaan yang tepat dalam *kuantitas* dan waktu yang tepat.

2.2.5 *Material Requirment Planning (MRP)*

Material Requirement Planning (MRP) dapat didefinisikan sebagai suatu teknik atau set prosedur yang sistematis dalam penentuan kuantitas serta waktu dalam proses pengendalian kebutuhan bahan terhadap komponen-komponen permintaan yang saling bergantungan. (*Dependent demand items*). (Gaspersz, 1998).[5]



Gambar 2.6 Sistem MRP

Strategi produksi digunakan untuk mengantisipasi agar terjadi keseimbangan antara pemasok material dan kebutuhan aktual pesanan. Ada dua kategori yang disarankan yakni *make to stock* dan *make to order*. *Make to stock* hanya dipakai untuk standard product sedangkan *make to order* digunakan pada kedua definisi produk yakni standard product dan custom product. Perbedaan pada strategi produksi *make to order* digambarkan pada tenggang waktu pengiriman untuk proses item produk dan persediaan material. Item produksi *make to stock*

telah tersedia dan berada pada gudang barang jadi untuk mengantisipasi permintaan aktual lainnya. Pada *make to stock* definisi produk yang digunakan adalah standard product. Produk dapat dilakukan dengan waktu pengiriman yang lebih pendek karena telah tersedia di gudang barang jadi sehingga tenggang waktu (lead time) lebih kecil dari *make to order*.

Make to order adalah membuat suatu produk sesuai dengan pesanan. Pada strategi produksi *make to order* definisi produk yang digunakan adalah standard product dan custom product.Variasi yang mungkin timbul pada make to order adalah bagaimana mengantisipasi level persediaan komponen atau material, serta bagaimana dapat memenuhi order-order yang masuk ke perusahaan. *Make to order* dapat dibagi atas : *Assembly to order, build to order, completely make to order dan engineer to order*.

2.2.5.1 Tujuan MRP

Tujuan dari MRP untuk menghasilkan informasi persediaan yang mampu digunakan untuk mendukung melakukan tindakan secara tepat dalam melakukan produksi. Agar MRP dapat berfungsi dan dioperasionalisasikan dengan efektif. Ada beberapa persyaratan dan asumsi yang harus dipenuhi. Adapun persyaratan yang dimaksud adalah : (*Gaspersz, 1998*) [3]

a) Tersedianya Jadwal Induk Produksi (*Master Production Schedule*), yaitu suatu rencana produksi yang menetapkan jumlah serta waktu suatu produk akhir harus tersedia sesuai dengan jadwal yang harus diproduksi. Jadwal Induk Produksi ini biasanya diperoleh dari hasil peramalan kebutuhan melalui tahapan perhitungan perencanaan

produksi yang baik, serta jadwal pemesanan produk dari pihak konsumen.

b) Setiap item persediaan harus mempunyai identifikasi yang khusus. Hal ini disebabkan karena biasanya *MRP* bekerja secara komputerisasi dimana jumlah komponen yang harus ditangani sangat banyak, maka pengklasifikasian atas bahan, bagian atas bahan, bagian komponen, perakitan setengah jadi dan produk akhir haruslah terdapat perbedaan yang jelas antara satu dengan yang lainnya.

c) Tersedianya struktur produk pada saat perencanaan. Dalam hal ini tidak diperlukan struktur produk yang memuat semua item yang terlibat dalam pembuatan suatu produk apabila itemnya sangat banyak dan proses pembuatannya sangat komplek. Walaupun demikian, yang penting struktur produk harus mampu menggambarkan secara gamblang langkah-langkah suatu produk untuk dibuat, sejak dari bahan baku sampai menjadi produk jadi.

d) Tersedianya catatan tentang persediaan untuk semua item yang menyatakan status persediaan sekarang dan yang akan datang.

Suatu sistem *MRP* pada dasarnya bertujuan untuk merancang suatu sistem yang mampu menghasilkan informasi untuk mendukung aksi yang tepat baik berupa pembatalan pesanan, pesan ulang, atau penjadwalan ulang.

Aksi ini sekaligus merupakan suatu pegangan untuk melakukan pembelian dan/ atau produksi. Ada 4 macam yang menjadi ciri utama *MRP*, yaitu: (Nasution,1992) :

a. Mampu menentukan kebutuhan pada saat yang tepat, kapan suatu pekerjaan akan selesai / material harus tersedia untuk memenuhi permintaan produk yang dijadwalkan berdasarkan MPS yang direncanakan.
b. Menentukan kebutuhan minimal setiap item, dengan menentukan secara tepat sistem penjadwalan.
c. Menentukan pelaksanaan rencana pemesanan, dengan memberikan indikasi kapan pemesanan atau pembatalan suatu pesanan harus dilakukan.
d. Menentukan penjadwalan ulang atau pembatalan atas suatu jadwal yang sudah direncanakan.Apabila kapasitas yang ada tidak mampu memenuhi pesanan yang dijadwalkan pada waktu yang dikehendaki, maka MRP dapat memberikan indikasi untuk melaksanakan rencana penjadwalan ulang (jika mungkin) dengan menentukan prioritas pesanan yang realistis. Seandainya penjadwalan ulang ini masih tidak memungkinkan untuk memenuhi pesanan, maka pembatalan terhadap suatu pesanan harus dilakukan.

2.2.5.2 Input MRP

Ada 3 Input yangdibutuhkan dalam konsep MRP yaitu (Nasution,1992) [9] :

1. Jadwal Induk Produksi (*Master production schedule*). Merupakan suatu rencana produksi yang menggambarkan hubungan antara kuantitas setiap jenis produk akhir yang diinginkan dengan waktu penyediaannya.

2. Struktur Produk *(Product structure Record & Bill of Material).*Merupakan kaitan antara produk dengan komponen penyusunnya. Informasi yang dilengkapi untuk setiap komponen ini meliputi :
    a. Jenis komponen.
    b. Jumlah yang dibutuhkan.
    c. Tingkat penyusunannya.

    Selain ini ada juga masukan tambahan seperti :

    1. Pesanan komponen dari perusahaan lain yang membutuhkan
    2. Peramalan atas item yang bersifat tidak bergantungan.
3. Status Persediaan *(Inventory Master File atau Inventory Status Record)* Menggambarkan keadaan dari setiap komponen atau material yang ada dalam persediaan, yang berkaitan dengan :
    a. Jumlah persediaan yang dimiliki pada setiap periode (*on hand inventory*)
    b. Jumlah barang dipesan dan kapan akan datang (*on order Inventory*)
    c. Waktu ancang – ancang (*lead time*) dari setiap bahan.

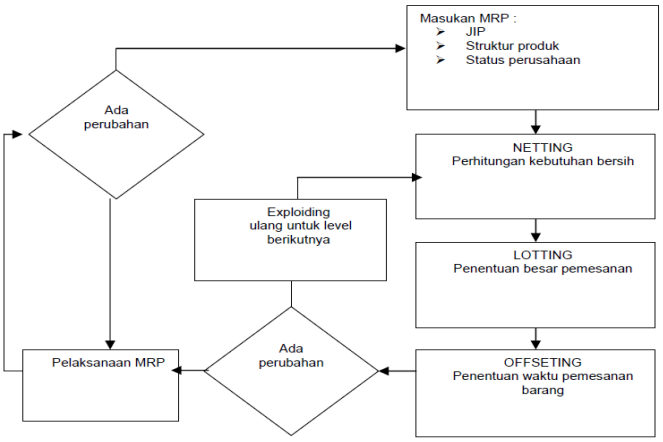
    Status persediaan ini harus diketahui untuk setiap bahan atau item dan diperbaharui setiap terjadi perubahan untuk menghindari adanya kekeliruan dalam perencanaan.

2.2.5.3 Proses MRP

Langkah - langkah dasar dalam penyusunan Proses *MRP* (Nasution,1992) [9] :

1) *Netting* (kebutuhan bersih) : Proses perhitungan kebutuhan bersih untuk setiap periode selama horison perencanaan.

2) *Lotting* (kuantitas pesanan) : Proses penentuan besarnya ukuran jumlah pesanan yang optimal untuk sebuah item, berdasarkan kebutuhan bersih yang dihasilkan.
3) *Offsetting* (rencana pemesanan) : Bertujuan untuk menentukan kuantitas pesanan yang dihasilkan proses lotting. Penentuan rencana saat pemesanan ini diperoleh dengan cara mengurangkan saat kebutuhan bersih yang harus tersedia dengan waktu ancang-ancang (*Lead Time*).
4) *Exploding*: Merupakan proses perhitungan kebutuhan kotor untuk tingkat (level) yang lebih bawah dalam suatu struktur produk, serta didasarkan atas rencana pemesanan.



Gambar 2.7 Langkah dan Proses *MRP*

2.2.5.4 Teknik Penentuan *Lotting* dalam MRP

Proses penentuan besarnya ukuran jumlah pesanan yang optimal untuk sebuah item, berdasarkan kebutuhan bersih yang dihasilkan dari masing masing periode horison perencanaan dalam *MRP (material requirment Planning)* [8]. Didalam ukuran lot ini ada beberapa pendekatan yaitu:

1. Menyeimbangkan ongkos pesan (*set up cost*) dan ongkos simpan.

    a) Biaya pemesanan (*order cost*) adalah biaya yang dikaitkan dengan usaha untuk mendapatkan bahan atau bahan dari luar. Biaya pemesanan dapat berupa biaya penulisan pemesanan, biaya proses pemesanan, biaya materai / perangko, biaya faktur, biaya pengetesan, biaya pengawasan, dan biaya transportasi. Sifat biaya pemesanan ini adalah semakin besar frekuensi pembelian semakin besar biaya pemesanan.

    b) Biaya Penyimpanan.

    Komponen utama dari biaya simpan (*carrying cost*) terdiri dari :

    a) Biaya Modal, meliputi : biaya yang diinvestasikan dalam persediaan, gedung, dan peralatan yang diperlukan untuk mengadakan dan memelihara persediaan.

    b) Biaya Simpan, meliputi : biaya sewa gudang, perawatan dan perbaikan bangunan, listrik, gaji, personel keamanan, pajak atas persediaan, pajak dan asuransiperalatan, biaya penyusutan dan perbaikan peralatan. Biaya tersebut ada bersifat tetap (*fixed*), variabel, maupun semi *fixed* atau semi variabel.

2. Menggunakan konsep jumlah pesanan tetap.

3. Dengan jumlah periode pemesanan tetap.

Terdapat beberapa alternatif teknik yang digunakan dalam menentukan ukuran Lot yaitu sebagai berikut :

Eqonomic Order Quantity (EOQ) adalah teknik pemesanan dalam manajemen pengadaan yaitu cara perhitungan pemesanan bahan baku sekali pesan atau berangsur dengan biaya paling minimum [9].

Variabel-variabel berikut ini akan digunakan untuk menentukan biaya pesan, biaya simpan, dan menghitung kuantitas pemesanan optimal :

(Q* atau EOQ).

Q = Jumlah satuan per pesanan

Q* = EOQ

D = Kebutuhan bahan baku (*Annual Demand*)

S = Biaya pesan per pesanan (*Setup/Ordering Cost*)

H = Biaya simpan/unit/hari (*Holding/Carrying Cost*)

$$Q^* = EOQ = \sqrt{\frac{2.D.S}{H}}$$

Waktu optimal antar pesanan (T)

$$T^* = \frac{Q^*}{D}$$

Jumlah pesanan optimal (N)

$$N = \frac{D}{Q^*}$$

OI = (*Current Inventori* + SR) – NR

OI = *Onhand Inventory* (merupakan proyeksi persediaan yaitu jumlah persediaan pada akhir suatu periode dengan memperhitungkan jumlah persediaan yang ada ditambah dengan jumlah item yang akan diterima atau dikurangi dengan jumlah item yang dipakai/dikeluarkan dari persediaan pada periode itu)

SR = *Schedule Receipt* (jumlah item yang akan diterima pada suatu periode tertentu berdasarkan pesanan yang telah dibuat)

Current Inventory = (jumlah material yang secara fisik tersedia dalam gudang pada awal periode)

NR = *Net Requirement* (jumlah kebutuhan bersih dari suatu item yang diperlukan untuk dapat memenuhi kebutuhan kasar pada suatu periode)

Kelebihan : Merupakan teknik yang mudah yang memasukkan parameterbiaya dan teknik yang menentukan trade off antara biaya pesan,set up dan ongkos simpan.

Kekurangan : Metode ini mengabaikan kemungkinan permintaan yang akan datang pada MRP. Teknik ini bukan teknik eksak sehingga sering mengakibatkan adanya sisa dari persediaan sehingga akan meningkatkan ongkos simpan.

2.2.5.5 Output MRP

Keluaran MRP sekaligus juga mencerminkan kemampuan dan ciri dari MRP, yaitu : (Gaspersz, 1998)

a. *Planned Order Schedule* (Jadwal Pesanan Terencana) adalah penentuan jumlah kebutuhan material serta waktu pemesanannya untuk masa yang akan datang.
b. *Order Release Report* (Laporan Pengeluaran Pesanan) berguna bagi pembeli yang akan digunakan untuk bernegosiasi dengan pemasok, dan berguna juga bagi manejer manufaktur, yang akan digunakan untuk mengontrol proses produksi.

b. *Changes to planning Orders* (Perubahan terhadap pesanan yang telah direncanakan) adalah yang merefleksikan pembatalan pesanan, pengurangan pesanan, pengubahan jumlah pesanan.

c. *Performance Report* (Laporan Penampilan) suatu tampilan yang menunjukkan sejauh mana sistem bekerja, kaitannya dengan kekosongan stock dan ukuran yang lain.

2.2.6 Basis Data (*Database*)

Berikut ini akan dikemukakan definisi dari basis data :

"Basis data (Database) adalah sekumpulan informasi bermanfaat yang diorganisasikan kedalam tata cara yang khusus".

Basis data (*database*) adalah kumpulan dari berbagai data yang saling berhubungan satu sama lainnya. Basis data tersimpan di perangkat keras, serta dimanipulasi dengan menggunakan perangkat lunak. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur, dan batasan dari data atau informasi yang akan disimpan. *Database* merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna. Penyusunan basis data meliputi proses memasukkan data kedalam media penyimpanan data, dan diatur dengan menggunakan perangkat Sistem Manajemen Basis Data (*Database Management System - DBMS*).

"Managemen Sistem Basis Data (Database Management System / DBMS) adalah perangkat lunak yang di desain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar".

Manipulasi basis data meliputi pembuatan pernyataan (*query*) untuk mendapatkan informasi tertentu, melakukan pembaharuan atau penggantian

(*update*) data, serta pembuatan laporan dari data. Tujuan utama DBMS adalah untuk menyediakan tinjauan abstrak dari data bagi *user*. Jadi system menyembunyikan informasi mengenai bagaimana data disimpan dan dirawat, tetapi data tetap dapat diambil dengan efisien. Pertimbangan efisiensi yang digunakan adalah bagaimana merancang struktur data yang komplek, tetapi tetap dapat digunakan oleh pengguna yang masih awam, tanpa mengetahui kompleksitas stuktur data.

Mengapa basis data penting? Basis data menjadi penting karena munculnya beberapa masalah bila tidak menggunakan data yang terpusat, seperti adanya duplikasi data, hubungan antar data tidak jelas, organisasi data dan *update* menjadi rumit. Jadi tujuan dari pengaturan data dengan menggunakan basis data adalah :

1. Menyediakan penyimpanan data untuk dapat digunakan oleh organisasi saat sekarang dan masa yang akan datang.
2. Cara pemasukan data sehingga memudahkan tugas operator dan menyangkut pula waktu yang diperlukan oleh pemakai untuk mendapatkan data serta hak-hak yang dimiliki terhadap data yang ditangani.
3. Pengendalian data untuk setiap siklus agar data selalu *up-to-date* dan dapat mencerminkan perubahan spesifik yang terjadi di setiap sistem.
4. Pengamanan data terhadap kemungkinan penambahan, modifikasi, pencurian dan gangguan-gangguan lain.

Dalam basis data sistem informasi digambarkan dalam model *entity relationship* (*E-R*). Bahasa yang digunakan dalam basis data yaitu

1. *DDL* (*Data Definition Language*)

   Merupakan bahasa definisi data yang digunakan untuk membuat dan mengelola objek *database* seperti *database*, tabel dan *view*.

2. *DML* (*Data Manipulation Language*)

   Merupakan bahasa manipulasi data yang digunakan untuk memanipulasi data pada objek *database* seperti table.

3. *DCL* (*Data Control Language*)

   Merupakan bahasa yang digunakan untuk mengendalikan pengaksesan Data.

2.2.6.1 Istilah Basis Data

Istilah dalam basis data ada beberapa istilah seperti :

1. Elemen Data

   Salah satu nilai tunggal dengan satu petunjuk nama dan deskripsi karakteristik seperti tipe (*char, int, varchar*) dan panjang karakter atau digit.

2. *Item* Data

   Merupakan referensi nama dan himpunan karakteristik elemen-elemen data yang menggambarkan suatu atribut. Atau merupakan tempat menyimpan setiap atribut dari sebuah entitas. Contoh : *Item* data IdMahasiswa dapat dikarakteristik dengan nomor digit 9 dengan nilai antara 000000001 sampai 999999999.

3. *Field*

   Merupakan lokasi penyimpanan untuk salah satu elemen data. Atau suatu elemen yang memiliki atribut dan harga dan merupakan unit informasi terkecil yang bisa diakses. Contoh : *field* IdMahasiswa.

4. *Record*

   Lokasi penyimpanan yang terbuat dari rangkain *field* yang berisi elemen-elemen data yang menggambarkan beberapa entitas.

5. *File*

   Sekumpulan *record* dari tipe tunggal yang berisi elemen-elemen data yang menggambarkan himpunan entitas. Contoh : *File* mahasiswa yang berisi satu *record* untuk tiap mahasiswa dalam sistem.

6. Akses Data

   Merupakan satu cara dimana suatu program mengakses secara fisik *record-record* dalam *file* penyimpan.

2.2.6.2 Normalisasi

"Normalisasi adalah suatu proses untuk mengorganisasikan file, untuk menghubungkan group elemen yang berulang-ulang dan merubah bentuk database dari struktur pohon menjadi struktur hubungan".

Aturan-aturan normalisasi dinyatakan dalam istilah bantuk normal. Yang merupakan suatu aturan yang dikenakan pada relasi-relasi dalam basis data yang harus dipenuhi oleh relasi tersebut pada *level-level* normalisasi.

Beberapa *level* yang biasa digunakan pada normalisasi adalah :

1. Bentuk normal pertama (1NF).

   Bentuk normal pertama adalah suatu relasi dikatakan dalam bentuk normal pertama jika dan hanya jika setiap atribut bernilai tunggal untuk setiap baris.

2. Bentuk normal kedua (2NF)

   Bentuk normal kedua didefinisikan berdasarkan dependensi fungsional. Suatu relasi berada dalam bentuk normal kedua jika dan hanya jika berada pada bentuk normal pertama dan semua atribut bukan kunci memiliki dependensi sepenuhnya terhadap kunci primer. Dalam ungkapan yang lebih praktis, bentuk normal kedua mensyaratkan setiap atribut bergantung kunci primer.

3. Bentuk normal ketiga (3NF).

   Bentuk normal ketiga adalah suatu relasi dikatakan dalam bentuk normal ketiga jika berada dalam bentuk normal kedua dan setiap atribut bukan kunci tidak memiliki dependensi transitif terhadap kunci primer.

4. Bentuk normal *Boyce-Codd* (BCNF).

   Bentuk normal *Boyce-Codd* adalah suatu relasi disebut memenuhi bentuk normal Boyce-Codd dan hanya jika semua penentu (*determinant*) adalah kunci kandidat (atribut yang bersifat unik). BCNF merupakan bentuk normal sebagai perbaikan terhadap 3NF, tetapi tidak untuk sebaliknya karena bentuk normal ketiga pun mungkin masih mengandung *anomaly* sehingga masih perlu dinormalisasi lebih lanjut.

2.2.7 *E-R Diagram*

Entity Relationship Diagram atau biasa dikenal dengan diagram E-R secara grafis menggambarkan isi sebuah database. Diagram ini memiliki dua komponen utama yaitu entity dan relasi. Untuk melambangkan fungsi diatas maka digunakan simbol-simbol yang bisa dilihat pada daftar simbol.

Elemen-elemen *Entity Relationship Diagram* adalah sebagai berikut:

1. Entity (Entitas)

   Pada E-R diagram, entity digambarkan dengan sebuah bentuk persegi panjang. Entity adalah sesuatu apa saja yang ada didalam sistem, nyata maupun abstrak dimana data tersimpan. Entitas diberi nama dengan kata benda dan dapat dikelompokan dalam empat jenis nama, yaitu : orang, benda, lokasi kejadian (terdapat unsur waktu didalamnya).

2. *Relationship* (Relasi)

   Pada E-R diagram, relationship dapat digambarkan dengan sebuah bentuk belah ketupat. Realationship adalah hubungan alamiah yang terjadi antara entitas. Pada umumnya relationship diberi nama dengan kata kerja dasar, sehinga memudahkan untuk melakukan pembacaan relasinya.

3. Atribut

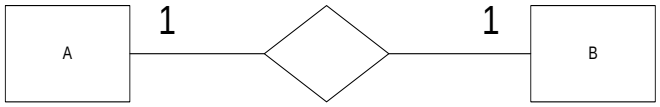
   Seacara umum atribut adalah sifat atau karakteristik dari tiap entitas maupun tiap relationship. Maksudnya adalah sesutau yang menjelaskan apa sebenarnya yang dimaksud entitas maupun relationship, sehingga sering dikatakan bahwa atribut adalah elemen dari setiap entitas dan relationship.

4. Kardinalitas

   Kardinalitas relasi menunjukan jumlah maksimum tupel yang dapat berelasi dengan entitas yang lainnya. Dari sejumlah kemungkinan banyaknya hubungan yang terjadi dari entitas, kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari entitas yang satu ke entitas yang lainnya dan begitu juga sebaliknya. Macam-macam kardinalitas relasi, yaitu :

a. One to one Relationship

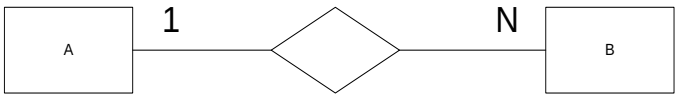
Tingkat hubungann satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.



Gambar 2.8 *One to One Relationship*

b. One to many Relationship

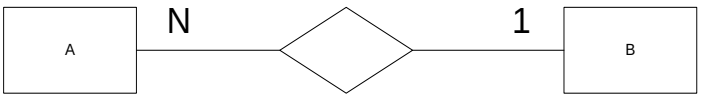
Tingkat hubungan satu ke banyak adalah untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.9 *One to Many Relationship*

c. Many To One Relationship

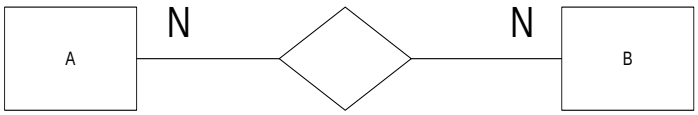
Untuk banyak kejadian pada entitas yang pertama hanya dapat mempunyai satu hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.10 *Many to One Relationship*

d. Many to many Relationship

Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya, baik dilihat dari sisi entitas yang pertama maupun dilihat dari sisi yang kedua.



Gambar 2.11 *Many to Many Relationship*

5. *Key* (Kunci)

Sebuah atribut atau set atribut yang nilainya mengidentifikasikan entitas secara unik dalam suatu entitas. *Key* memiliki beberapa jenis sesuai dengan kegunaannya masing-masing, yaitu *primary key* (kunci utama), *foreign key* (kunci tamu).

2.2.8 DFD (Data Flow Diagram)

Diagram Alir Data atau *Data Flow Diagram* (DFD) adalah suatu model yang menjelaskan arus data mulai dari pemasukan sampai dengan keluaran data. Tingkatan DFD dimulai dari diagram konteks yang menjelaskan secara umum suatu sistem atau batasan sistem aplikasi yang akan dikembangkan. Kemudian DFD dikembangkan menjadi DFD tingkat 0 atau level 0 dan kemudian DFD level 0 dikembangkan lagi menjadi level 1 dan selanjutnya sampai sistem tersebut tergambarkan secara rinci menjadi tingkatan-tingkatan lebih rendah lagi.

DFD merupakan penurunan atau penjabaran dari diagram konteks. Dalam pembuatan DFD harus mengacu pada ketentuan sebagai berikut :

1. Setiap penurunan level yang lebih rendah harus mempresentasikan proses tersebut dalam spesifikasi proses yang jelas.
2. Penurunan dilakukan apabila memang diperlukan.
3. Tidak semua bagian dari sistem harus ditunjukkan dengan jumlah level yang sama.

Simbol-simbol yang digunakan dalam Data Flow Diagram menurut notasi Yourdan adalah sebagai berikut :

1. Proses

Proses adalah simbol pertama data flow diagram. Proses dilambangkan dengan lingkaran, dimana proses ini menunjukan bagian dari sistem yang mengubah satu atau lebih input dan output. Nama proses dituliskan dengan satu kata, singkatan atau kalimat sederhana.

2. Aliran Data

Aliran Data digambarkan dengan tanda panah. Aliran data juga digunakan untuk menunjukan bagian-bagian informasi dari satu bagian ke bagian lain. Pembagian nama untuk aliran ini menunjukan sebuah arti untuk sebuah aliran. Untuk kebanyakan sistem yang dibuat, aliran data sebenarnya mengambarkan data yakni angka, huruf, pesan, floating point, dan macam-macam informasi lainnya.

3. Simpanan Data

Simpanan data digunakan sebagai penyimpanan bagi paket-paket data. Notasi penyimpanan data digambarkan dengan garis horizontal yang pararel. Simpanan data merupakan simpanan data dari data yang berupa suatu file atau database di sistem komputer ataupun berupa arsip atau catatan manual. Nama dari simpanan data menunjukan nama filenya.

4. Terminator

Terminator digambarkan dengan sebuah kotak yang menggambarkan kesatuan luar (*eksternal entitty*) yang berhungan dengan sistem. Kesatuan luar merupakan kesatuan (*entity*) dilingkungan luar sistem yang dapat berupa orang, Organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau output dari sistem.

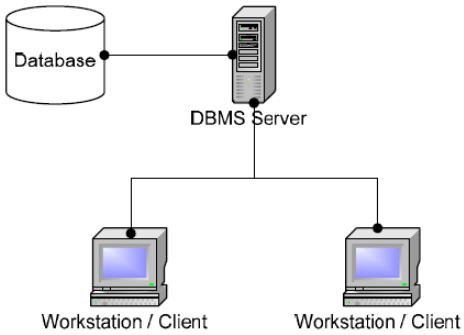
2.2.9 Sistem *Client-Server*

Sistem *client-server* dapat mempunyai dua atau tiga tingkat *setup* (pengaturan). Terhadap sebuah komputer yang berfungsi sebagai *server* (pusat) dan beberapa komputer lainnya yang bersifat sebagai *client*. Beberapa bentuk pelayanan yang dapat diberikan komputer *server* antara lain :

1. Mengontrol hak akses perangkat-perangkat yang ada dalam jaringan
2. Mengatur keamanan data dalam jaringan
3. Penggunaan perangkat, data dan sistem aplikasi komputer secara bersama

Sistem *client-server* atau disebut juga sistem tersentralisasi diterapkan pada sebuah sistem jaringan. Sistem *client-server* ini ditujukan untuk mengatasi kelemahan-kelemahan yang terdapat pada sistem sebelumnya. Sistem *client-server* terdiri dari dua komponen utama yaitu *client* dan *server*. *Client* berisi aplikasi basis data dan *server* berisi DBMS dan basis data. Setiap aktivitas yang dikehendaki para pemakai akan lebih dulu ditangani oleh *client*. Bila ada proses yang harus melibatkan data yang tersimpan pada basis data barulah *client* melakukan hubungan dengan *server*.

Pada sistem *client-server* untuk memenuhi kebutuhan *client* akan mengirimkan *message* (perintah) *query* pengambilan data. Selanjutnya, *server* yang menerima *message* tersebut akan menjalankan *query* tersebut dan hasilnya akan dikirim kembali ke *client*. Dengan begitu transfer datanya jauh lebih efisien. Adapun bentuk dari sistem *client-server* yang sederhana adalah sebagai berikut :



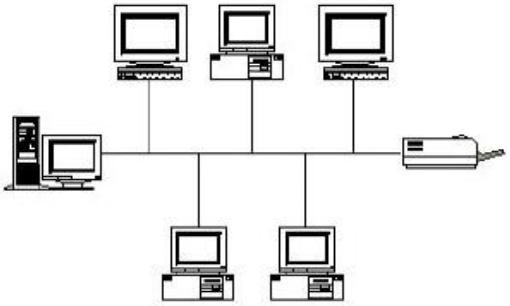
Gambar 2.12 Bentuk *Client-Server*

2.2.9.1 Topologi Fisik

Berikut ini akan dijelaskan mengenai topologi fisik yang digunakan di dalam jaringan lokal diantaranya :

1. *Linear Bus* (Garis Lurus)

   Topologi *Linear Bus* (Garis lurus) terdiri dari satu jalur kabel utama dimana pada masing-masing ujungnya diberikan sebuah *terminator*. Semua nodes pada jaringan (*file server*, *workstation*, dan perangkat lainnya) terkoneksi sebuah kabel utama (*backbone*). Jaringan-jaringan *Ethernet* dan *Local Talk* menggunakan topologi linear ini.

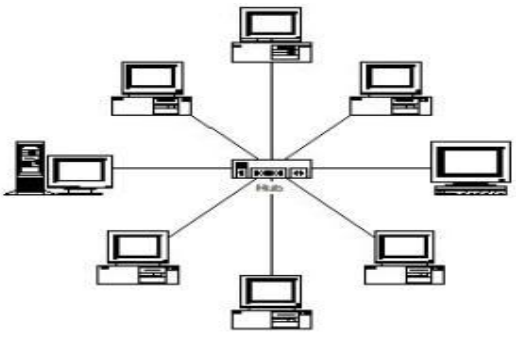


Gambar 2.13 Topologi *Linear Bus*

2. *Star* (Bintang)

Topologi model ini dirancang, yang mana setiap nodes (*file server*, *workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator*.

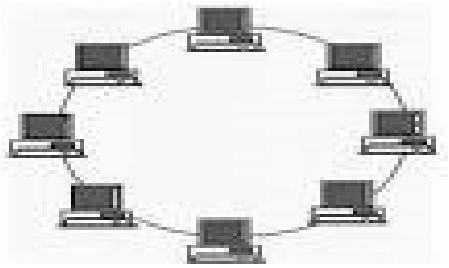
Data yang dikirim ke jaringan lokal akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengendalikan keseluruhan fungsi jaringan, dan juga bertindak sebagai repeter (penguat aliran data). Konfigurasi pada jaringan model ini menggunakan kabel *Twisted Pair*, dan dapat diguanakan pula kabel *coaxial* atau kabel *fiber optic*.



Gambar 2.14 Topologi *Star*

3. *Ring* (Cincin)

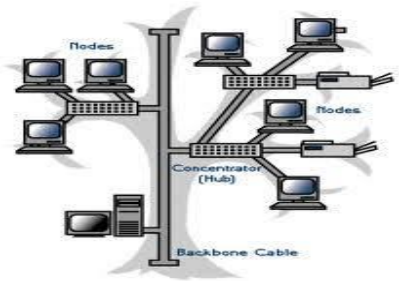
Topologi *Ring* (Cincin) menggunakan teknik konfigurasi yang sama dengan topologi *star* tetapi pada topologi ini terlihat bahwa jalur media transmisi menyerupai suatu lingkaran tertutup menyerupai cincin (lingkaran), sehingga diberi nama topologi bintang dalam lingkaran atau *star-wired ring*.



Gambar 2.15 Topologi *Ring*

4. *Tree* (Pohon)

Topologi model ini merupakan perpaduan antara topologi *Linear Bus* dan *Star*, yang mana terdiri dari kelompok-kelompok dari *workstation* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *Linear Bus*. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada, dan memungkinkan untuk mengkonfigurasi jaringan sesuai dengan kebutuhan.



Gambar 2.16 Topologi *Tree*

2.3 *Software* Pendukung

Software pendukung yang digunkan dalam pembuatan aplikasi ini adalah *Rad Studio Delphi 2010* dan *MySQL Server 5*.

2.3.1 RAD STUDIO DELPHI XE

Delphi adalah suatu program berbasis bahasa *Pascal* yang berjalan dalam lingkungan *Windows*. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut *RAD* yang telah membuat pemrograman menjadi lebih mudah. *Delphi* adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman *Object Oriented Programming* (OOP). Secara ringkas, objek adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (*visual*). *Object* biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu.

Khusus untuk pemrograman *database*, *Delphi* menyediakan objek yang sangat kuat, canggih, dan lengkap, sehingga memudahkan pemrogram dalam merancang, membuat dan menyelesaikan aplikasi *database* yang diinginkan. Selain itu, Delphi juga dapat menangani data dalam berbagai format *database*, misalnya format *MS-Access, SyBase, Oracle, FoxPro, Informix, DB2, MySQL* dan lain-lain. Format database yang dianggap asli dari Delphi adalah Paradox dan dBase.

2.3.2 MySQL

2.3.2.1 Pengertian MySQL

MySQL adalah suatu perangkat lunak database relasi (Relational Database Management System atau RDBMS), seperti halnya ORACLE, Postgresql, MS SQL, dan sebagainya. MySQL AB menyebut produknya sebagai database open source terpopuler di dunia. Berdasarkan riset dinyatakan bahwa bahwa di platform Web, dan baik untuk kategori open source maupun umum, MySQL adalah database yang paling banyak dipakai. Menurut perusahaan pengembangnya, MySQL telah terpasang di sekitar 3 juta komputer. Puluhan hingga ratusan ribu situs mengandalkan MySQL bekerja siang malam memompa data bagi para pengunjungnya.

2.3.2.2 Fiture MySQL

Kalau di seri 3.22 MySQL mulai diadopsi banyak orang dan meningkat populasi penggunanya, maka di seri 3.23 dan 4.0-lah terjadi banyak peningkatan dari sisi teknologi. Ini tidak terlepas dari tuntutan pemakai yang semakin mengandalkan MySQL, namun membutuhkan fitur-fitur yang lebih banyak lagi.

Seri 3.23. Di seri 3.23 MySQL menambahkan tiga jenis tabel baru: pertama MyISAM, yang sampai sekarang menjadi tipe tabel default; kedua BerkeleyDB, yang pertama kali menambahkan kemampuan transaksi pada MySQL; dan ketiga InnoDB, primadona baru yang potensial.

Seri 4.x. Di seri yang baru berjalan hingga 4.0 tahap alfa ini, pengembang MySQL berjanji akan menjadikan MySQL satu derajat lebih tinggi lagi. Fitur-fitur yang sejak dulu diminta akan dikabulkan, seperti subselek (di 4.1), union (4.0), foreign key constraint (4.0 atau 4.1—meski InnoDB sudah menyediakan ini di

3.23.x), stored procedure (4.1), view (4.2), cursor (4.1 atau 4.2), trigger (4.1). MySQL AB tetap berdedikasi mengembangkan dan memperbaiki MySQL, serta mempertahankan MySQL sebagai database open source terpopuler.

2.3.2.3 Keunggulan MySQL

Ada beberapa keungulan dari MySQL diantaranya adalah:

1. **(**MySQL begitu popular di kalangan Web adalah karena ia memang cocok bekerja di lingkungan tersebut. Pertama, MySQL tersedia di berbagai platform Linux dan berbagai varian Unix. Sesuatu yang tidak dimiliki Access, misalnya—padahal Access amat popular di platform Windows. Banyak server Web berbasiskan Unix, ini menjadikan Access otomatis tidak dapat dipakai karena ia pun tidak memiliki kemampuan client-server/networking.
2. Fitur-fitur yang dimiliki MySQL memang yang biasanya banyak dibutuhkan dalam aplikasi Web. Misalnya, klausa LIMIT SQL-nya, praktis untuk melakukan paging. Atau jenis indeks field FULLTEXT, untuk full text searching. Atau sebutlah kekayaaan fungsi-fungsi builtinnya, mulai dari memformat dan memanipulasi tanggal, mengolah string, regex, enkripsi dan hashing. Yang terakhir misalnya, praktis untuk melakukan penyimpanan password anggota situs.
3. MySQL memiliki overhead koneksi yang rendah. Soal kecepatan melakukan transaksi atau kinerja di kondisi load tinggi mungkin bisa diperdebatkan dengan berbagai benchmark berbeda, tapi kalau soal yang satu ini MySQL-lah juaranya. Karakteristik ini membuat MySQL cocok bekerja dengan aplikasi CGI, di mana di setiap request skrip akan

melakukan koneksi, mengirimkan satu atau lebih perintah SQL, lalu memutuskan koneksi lagi. Cobalah melakukan hal ini dengan Interbase atau bahkan Oracle. Maka dengan load beberapa request per detik saja server Web/database Anda mungkin akan segera menyerah karena tidak bisa mengimbangi beban ini.