BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

Pada tinjauan perusahaan ini akan membahas sekilas tentang sejarah berdirinya perusahaan dan struktur organisasi yang ada di dalamnya.

2.1.1 Sejarah Berdirinya CV. DIAN'A

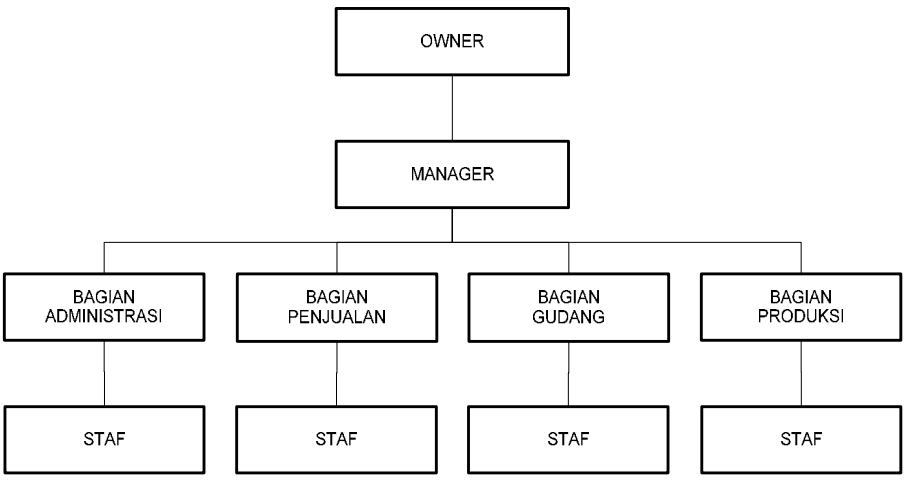
Berawal dari keinginan untuk berhenti dari pekerjaan disalah satu perusahaan BUMN, Bapak Ano Suparno menggagas pendirian perusahaan CV. DIAN'A. Perusahaan yang didirikan pada tahun 1990, pada awalnya hanya bergerak dalam bidang pengadaan alat-alat *tell struk*. Dengan modal yang minim usaha pun dimulai dengan fasilitas yang minim pula, pemasaran gerilya dan promosi mulut ke mulut.

Rome wasn't built on one day, demikian juga sesuatu yang besar dimulai dari langkah-langkah yang kecil. Dibutuhkan lebih dari sekadar keberanian, idealisme, dan kerja keras. Sejalan dengan berjalannya waktu dan bertambahnya relasi, akhirnya perusahaan pun mengalami pengembangan usaha dalam bidang pengadaan alat-alat listrik. Perkembangan perusahaan CV. DIAN'A sangat dirasakan mengalami kemajuan yang pesat. Hal ini dapat dilihat dari banyaknya jumlah permintaan barang dari konsumen/pelanggan yang berasal dari berbagai kota di luar daerah.

2.1.2 Struktur Organisasi

Setiap perusahaan mempunyai struktur organisasi yang merupakan bentuk aturan kerjasama antar manusia dan fasilitas lainnya yang digunakan untuk mencapai tujuan yang telah ditetapkan. Organisasi dapat diartikan sebagai wadah untuk melakukan kerjasama orang-orang yang berada dalam organisasi tersebut. Struktur Organisasi perusahaan merupakan kerangka kerja yang menggambarkan adanya pelimpahan wewenang dan tanggung jawab dari setiap tindakan dalam struktur tersebut.

Pengaturan dan pembagian tugas dapat berjalan dengan baik jika ada garis komunikasi yang baik antara pimpinan dan bawahannya, juga sebaliknya. Serta masing-masing bagian mengerti akan tugas dan tanggung jawabnya. Dengan demikian perlu adanya struktur organisasi agar seluruh kegiatan dalam perusahaan dapat berjalan dengan baik dan lebih efisien.



Gambar 2.1 Struktur Organisasi Perusahaan CV. DIAN'A

2.2 Landasan Teori

Pada landasan teori ini diuraikan sejumlah teori untuk membantu dan memecahkan permasalahan yang ada. Beberapa landasan teori tersebut meliputi konsep dasar dan definisi-definisi yang berkaitan dengan perangkat lunak yang digunakan serta faktor-faktor pendukung dalam pelaksanaan perancangannya.

2.2.1 Konsep Dasar Sistem

Dalam keseharian secara sadar atau tidak manusia terlibat dalam sebuah sistem, sistem transportasi, sistem pendidikan, sistem pernafasan, sistem informasi, dan lain-lain. Pengertian Sistem terbagi dua yaitu dilihat dari pendekatan yang menekankan pada prosedur dan pendekatan yang menekankan pada elemen atau komponennya.

Menurut **[Jog99]** pendekatan sistem yang menekankan pada prosedur mendefinisikan sistem sebagai : ”*Jaringan kerja dan prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan sasaran tertentu*”.

Adapun pendekatan sistem yang menekankan pada elemen atau komponennya mendefinisikan sistem sebagai : ”*Kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu*”. **[Jog99]**

Dari kedua pengertian di atas dapat disimpulkan bahwa pengertian sistem adalah kumpulan elemen-elemen atau jaringan kerja dan prosedur-prosedur yang saling berinteraksi satu sama lain untuk mencapai tujuan atau sasaran tertentu.

2.2.1.1 Karakteristik Sistem

Menurut **[Jog99]** suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu :

1. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Sebesar apapun setiap sistem, selalu mengandung komponen-komponen atau subsitem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu, dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem

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

3. Lingkungan Luar Sistem

   Lingkungan luar (*environment*) sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi yang harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya yang mengalir dari suatu subsistem ke subsistem yang lainnya. Keluaran (*output*) dari suatu subsistem akan menjadi masukan (*input*) untuk subsistem yang lainnya dengan melalui penghubung.

5. Masukan Sistem

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masalah perawatan (*maintenance input*) dan masukkan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem

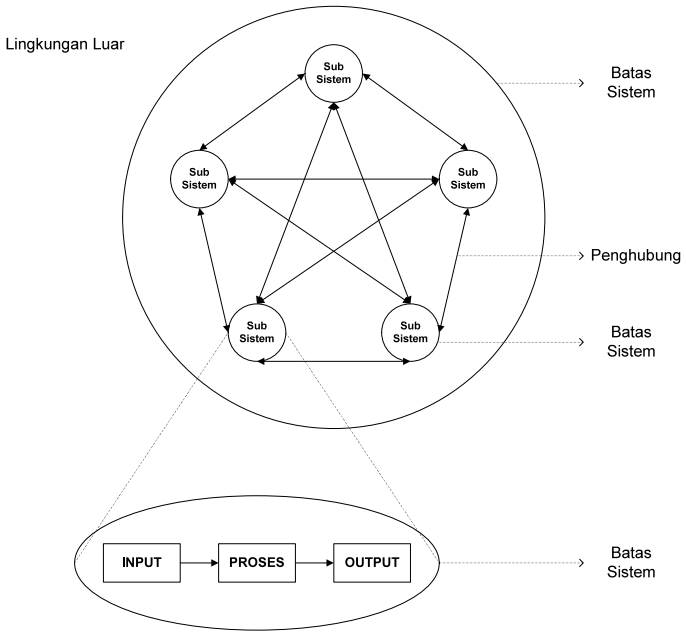
Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukkan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukkan menjadi keluaran. Suatu sistem produksi akan mengolah masukkan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan lain yang dibutuhkan oleh manajemen.

8. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan masukkan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



Gambar 2.2 Karakteristik Sistem

2.2.1.2 Klasifikasi Sistem

Menurut [Jog99] sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya adalah sebagai berikut:

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstrack system*) dan sistem fisik (*physical system*). Sistem abstrak adalah sistem yang berupa pemikiran/ide-ide yang tidak tampak secara fisik. Sistem fisik merupakan sistem yang ada secara fisik.

2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut dengan *human-machine system* atau ada yang menyebutnya dengan *man machine system*.
3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi antara bagiannya sudah dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.
4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*). Sistem tertutup merupakan sistem yang tidak berhubungan dengan lingkungan luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukkan dan menghasilkan keluaran untuk lingkungan luar/subsistem yang lainnya.

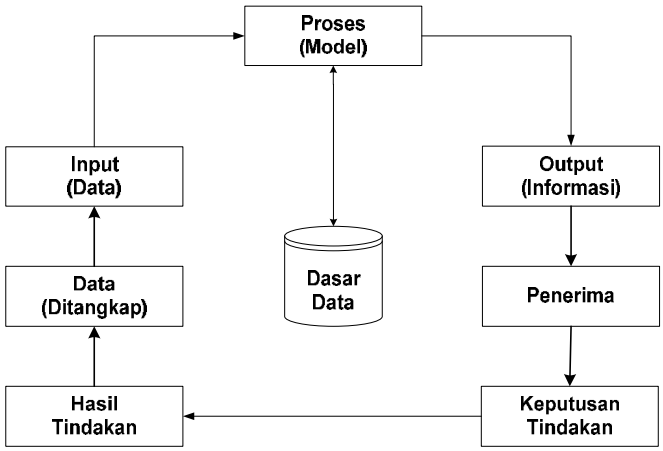
2.2.2 Konsep Dasar Informasi

Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi, sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan luruh, kerdil dan akhirnya berakhir.

"Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya" **[Jog99].** Sumber dari informasi itu sendiri ialah data. Data merupakan bentuk jamak dari bentuk tunggal atau data-item. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian-kejadian nyata yang sering terjadi adalah perbuatan dari suatu nilai yang disebut dengan transaksi.

2.2.2.1 Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data yang diolah melalui suatu model menjadi suatu informasi, penerima kemudian menerima informasi tersebut, 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, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus.



Gambar 2.3 Siklus Informasi

2.2.2.2 Kualitas Informasi

Kualitas dari suatu informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat waktunya (*timeliness*) dan relevan (*relevance*).

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bisa atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.
2. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi. Dewasa ini mahalnya nilai informasi disebabkan harus cepatnya nilai informasi tersebut didapat, sehingga diperlukan teknologi-teknologi mutakhir untuk mendapatkan, mengolah dan mengirimkannya.

3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya, dengan cara mengurangi ketidakpastian, menaikkan kemampuan untuk memprediksi, atau menegaskan ekspektasi semula.

2.2.2.3 Nilai Informasi

Nilai dari informasi (*Value of Information*) ditentukan dari dua hal yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Akan tetapi perlu diperhatikan bahwa informasi yang digunakan di dalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Sebagian informasi tidak dapat ditaksir keuntungannya dengan suatu nilai tetapi dapat ditaksir nilai keefektifannya.

2.2.3 Konsep Dasar Sistem Informasi

Informasi merupakan hal yang sangat penting didalam mengambil keputusan. Informasi tersebut didapatkan dari sistem informasi (*information system*) atau disebut juga dengan *processing systems* atau *information generating/processing systems*. Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

2.2.3.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*), dan blok kendali (*control block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.

1. Blok Masukan

   Input mewakili data yang masuk ke dalam sistem informasi, input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

   Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan dibasis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

   Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

   Teknologi merupakan “kotak alat” (*tool-box*) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan

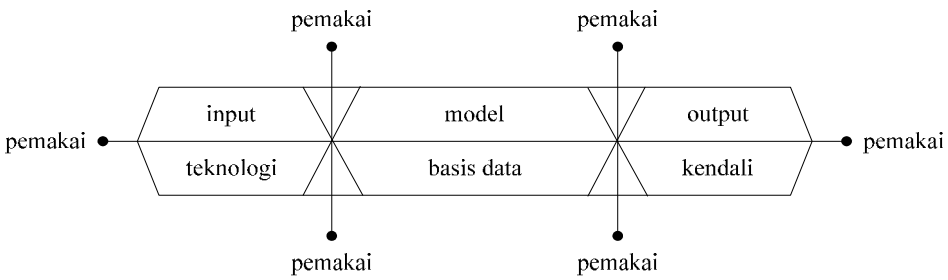
dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data

   Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas.

6. Blok Kendali

   Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana, temperatur, kecurangan , kegagalan , kesalahan , ketidakefisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.



Gambar 2.4 Komponen Sistem Informasi

2.2.4 Basis Data

Basis data terdiri dari dua kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Prinsip utama Basis data adalah pengaturan data atau arsip. Dan tujuannya adalah kemudahan dan kecepatan dalam pengambilan kembali data atau arsip.

2.2.4.1 Karakteristik Basis Data

Basis data memiliki karakteristik utama yaitu ;

1. Data yang sama dapat diakses secara serentak (*concurrency acces*) oleh beberapa pemakai untuk berbagai kegunaan yang berbeda.
2. Data tidak tergantung pada struktur atau *strategy access* dari program aplikasi atau data bersifat transparan terhadap program aplikasi.
3. Data memiliki integritas (akurasi dan validasi) yang terkendali. Strategi akses terhadap data bersifat *logic* menyebabkan basis data berbeda dengan file-file yang lainya. Interaksi basis data bersifat fisik artinya user atau penguna sangat bergantung pada struktur data yang dimilikinya.

2.2.4.2 Operasi Dasar Basis Data

Operasi-operasi dasar yang dapat kita lakukan berkenaan dengan basis data dapat meliputi :

1. Pembuatan basis data baru (*create database*)
2. Penghapusan basis data (*drop database*)
3. Pembuatan file atau tabel baru ke suatu basis data (*create table*)
4. Penghapusan file atau tabel dari suatu basis data (*drop table*)
5. Penambahan atau pengisian data baru ke sebuah file atau tabel di sebuah basis data (*insert*)
6. Pengamabilan data dari sebuah file atau tabel (*retriev*e atau *search*)
7. Pengubahan data dari sebuah file atau tabel (*update*)
8. Penghapusan data dari sebuah file atau tabel (*delete*)

2.2.4.3 Bahasa Basis Data

Sistem basis data menyediakan bahasa pendefinisian data *(Data Definition Language-DDL)* untuk menentukan skema basis data dan bahasa manipulasi data *(Data Manipulation Language-DML)* untuk menyatakan *query* dan *update* basis data. Pada praktiknya, DDL dan DML bukan merupakan dua bahasa yang terpisah melainkan membentuk bagian bahasa basis data, seperti yang umum digunakan pada bahasa SQL.

2.2.4.4 *Data Definition Language* (DDL)

Skema basis data ditentukan sekumpulan definisi yang dinyatakan dengan bahasa tertentu yang disebut *Data Definition Language* (DDL). Sebagai contoh, pernyataan berikut dalam bahasa SQL mendefinisikan tabel rekening.

```
Create table rekening
    (no_rekening char(10),
     Saldo integer)
```

Eksekusi pernyataan DDL di atas akan membuat tabel rekening. Sebagai tambahan, pernyataan itu menambah kumpulan tabel yang disebut kamus data atau direktori data.

Sistem basis data akan mengecek kamus data sebelum membaca atau memodifikasi data sebenarnya. Struktur penyimpanan dan metode akses yang digunakan oleh sistem basis data dengan menggunakan sekumpulan pernyataan dalam tipe DDL tertentu disebut bahasa penyimpanan dan definisi data. Pernyataan ini menentukan penerapan detail skema basis data, yang sering disembunyikan dari pengguna. Nilai yang disimpan dalam basis data harus sesuai dengan batasan tertentu. Sistem basis data mengecek batasan setiap basis data diubah.

2.2.4.5 *Data Manipulation Language* (DML)

Manipulasi data adalah :

1. Pengambilan informasi yang disimpan dalam basis data
2. Penempatan informasi baru dalam basis data

3. Penghapusan informasi dari basis data
4. Modifikasi informasi yang disimpan dari basis data

Data Manipulation Language (DML) adalah bahasa yang memungkinkan pengguna mengakses atau memanipulasi data seperti yang diatur oleh model data. Ada dua tipe DML, yaitu :

1. DML Prosedural

   DML Prosedural mengharuskan pengguna untuk menentukan data yang dibutuhkan dan bagaimana mendapatkannya.

2. DML Deklaratif

   DML Deklaratif (disebut juga DML non-prosedural), mengharuskan pengguna menentukan data yang dibutuhkan. DML deklaratif lebih mudah dipelajari dan digunakan dibandingkan DML prosedural. Namun, karena pengguna tidak harus menentukan cara mendapatkan data, sistem basis data harus mencari alat yang efektif untuk mengakses data. Komponen DML dari bahasa SQL adalah non-prosedural.

2.2.4.6 Organisasi File Basis Data

Penekanan file pada basis data adalah kemampuan untuk mengakses data secara cepat dan efisien dalam menggunakan media simpanan luarnya. Faktor yang dapat mempengaruhi hal ini adalah organisasi dari file basis data. Organisasi file basis data ini mencoba meningkatkan struktur dari data agar integrasi data antara satu file dengan file lainnya, dengan menunjukan hubungan antara data yang ada :

1. Struktur data berjenjang (*hierarchial data structure*) atau disebut juga dengan nama data pohon (*tree data structure*) menunjukan hubungan data membentuk suatu jenjang seperti pohon.
2. Struktur data jaringan (*network data structure*) disebut juga dengan plex data struktur. Pada struktur data pohon tiap-tiap node tidak dapat mempunyai lebih dari satu orang tua, maka struktur data jaringan ini, tiap-tiap node memiliki banyak orang tua.
3. Struktur data hubungan adalah meletakkan semua hubungan dalam sebuah tabel dua dimensi. Data dalam model ini dapat diidentifikasi dalam hubungan nyata terhadap data item yang ada.

2.2.5 *Database Management System* (DBMS)

Sistem basis data adalah suatu sistem menyusun dan mengelola record-record menggunakan komputer untuk menyimpan serta memelihara data opersional lengkap sebuah organisasi sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Manajemen Sistem Basis Data DBMS adalah perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar. DBMS dapat menjadi alternatif penggunaan secara khusus untuk aplikasi, semisal penyimpanan data dalam file dan menulis kode aplikasi yang spesifik untuk pengaturannya. DBMS memiliki beberapa sifat-sifat antara lain :

1. Mengolah file-file yang saling berhubungan.

2. Program yang disediakan meliputi fungsi untuk :
    a. Memanipulasi data yang telah ada dalam file.
    b. Mengorganisasi dan mengontrol data dalam jumlah yang besar.
    c. Memasukan data atau write data.
    d. Melakukan *backup*, *recovery* dan *loging* terhadap data.
    e. Menyederhanakan dan memperluas pengamanan data.

2.2.6 Normalisasi

Normalisasi merupakan tahapan perancangan dalam membangun basis data relasional yang tidak secara langsung berkaitan dengan model data, tetapi dengan menerapkan sejumlah aturan dan kriteria standar untuk menghasilkan struktur tabel yang normal. Adapun langkah-langkah untuk menghasilkan struktur tabel yang normal, diantaranya yaitu :

1. Bentuk *Unnormallized*

    Bentuk *unnormal* adalah bentuk tabel dengan mencantumkan semua *field* data yang ada.

2. Bentuk Normal Pertama / 1 NF (*First Normal Form*)

    Bentuk normal pertama terpenuhi jika sebuah tabel tidak memiliki atribut bernilai banyak atau lebih dari satu atribut dengan domain nilai yang sama.

3. Bentuk Normal Kedua / 2 NF (*Second Normal Form*)

    Bentuk normal kedua terpenuhi jika pada sebuah tabel semua atribut yang tidak termasuk dalam *key primer* memiliki ketergantungan fungsional pada *key primer* secara utuh.

4. Bentuk Normal Ketiga / 3 NF (*Third Normal Form*)

   Bentuk normal ketiga terpenuhi jika dan hanya jika semua atribut bukan kunci memiliki *dependensi* transitif terhadap kunci primer.

5. *Boyce-Codd Normal Form* (BCNF)

   Suatu tabel dikatakan berada pada BCNF jika dan hanya jika semua ketergantungan fungsional dengan notasi X ? Y, maka X harus merupakan *superkey* pada tabel tersebut.

6. Bentuk Normal Keempat / 4 NF (*Fourth Normal Form*)

   Suatu tabel dikatakan berada pada normal keempat jila tidak mengandung dua atribut atau lebih yang bernilai banyak.

7. Bentuk Normal Kelima / 5 NF (*Fifth Normal Form*)

   Bentuk normal kelima berkaitan dengan ketergantungan relasi antar tabel.

2.2.7 Perancangan dan Pengujian Sistem

2.2.7.1 Perancangan Data

Perancangan data atau desain data merupakan aktivitas pertama dari empat aktivitas desain yang dilakukan selama rekayasa perangkat lunak. Pengaruh struktur data pada struktur program dan kompleksitas prosedural menyebabkan desain data berpegaruh penting terhadap kualitas perangkat lunak.

2.2.7.2 Perancangan Antarmuka

Desain *interface* memfokuskan diri pada tiga area perhatian, yaitu :

1. Desain *interface* antara modul-modul perangkat lunak.

2. Desain *interface* antara perangkat lunak dan prosedur dan konsumen informasi.bukan manusia lainnya (entitas eksternal).
3. Desain *interface* antara seorang manusia (pemakai) dan komputer.

2.2.7.3 Perancangan Prosedural

Desain prosedural terjadi setelah data dan desain *interface* dibangun. Spesifikasi prosedural diperlukan untuk menetapkan detail-detail algoritma yang akan dinyatakan dalam suatu bahasa ibu, seperti bahasa Inggris. Akan tetapi, semua anggota organisasi pengembangan perangkat lunak menggunakan bahasa ibu (paling tidak secara teori), orang di luar domain perangkat lunak dapat lebih memahami spesifikasi tersebut, dan tidak ada pelajaran baru yang diperlukan.

2.2.7.4 Pengujian

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan memresentasikan kajian pokok dari spesifiaksi, desain, dan pengkodean. Sasaran pengujian menyatakan sejumlah aturan diantaranya :

1. Pengujian (*test case*) adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
2. Pengujian yang baik adalah pengujianyang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
3. Pengujian harus dimulai “dari yang terkecil” dan berkembang ke pengujian “yang besar”.
4. Pengujian yang mendalam tidak mungkin.

Testabilitas perangkat lunak secara sederhana adalah seberapa mudah sebuah program komputer dapat diuji. Serangkaian karakteristik yang membawa kepada perangkat lunak yang dapat diuji :

1. Operabilitas

   “Semakin baik dia bekerja, semakin efisien dia dapat diuji”.

2. Observabilitas

   “Apa yang anda lihat adalah apa yang anda uji”.

3. Kontrolabilitas

   “Semakin baik kita dapat mengontrol perangkat lunak, semakin banyak pengujian yang dapat diotomatisasi dan dioptimalkan”

4. Dekomposabilitas

2.2.7.5 Pengujian *Black Box*

Metode yang dilakukan dengan cara memasukkan hasil yang dikeluarkan oleh sistem tersebut sama dengan data yang diisikan ke dalam sistem atau hasil yang dikeluarkan sistem itu sesuai dengan hasil perhitungan yang ada di dalam sistem.

Pengujian *black box* didesain untuk mengungkap kesalahan pada persyaratan fungsional tanpa mengabaikan kerja *internal* dari suatu program. Teknik pengujian *black box* berfokus pada domain informasi dari perangkat lunak, dengan melakukan *test case* dengan mempartisi domain *input* dan *output* dari suatu program dengan cara yang memberikan cakupan pengujian yang mendalam **[Pre97]**.

Metode pengujian *black box* bukan merupakan alternatif dari metode pengujian *white box*, melainkan merupakan pendekatan yang saling melengkapi, misalnya menemukan bagian kesalahan yang berbeda dari metode pengujian *white box*. Metode pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1 Fungsi-fungsi yang tidak benar atau hilang.

2 Kesalahan antarmuka.

3 Kesalahan dalam struktur data atau akses basis data.

4 Kesalahan kinerja atau perilaku.

5 Kesalahan inisialisasi dan terminasi.

2.2.7.6 Pengujian *White Box*

Metode *White box* sering disebut juga dengan metode pengujian *glass box*, adalah metode desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *test case*. Dengan menggunakan metode pengujian *white box*, perekayasa sistem dapat melakukan sistem yang:

1 Memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali.

2 Menggunakan semua keputusan logis pada sisi *true* dan *false.*

3 Mengeksekusi semua *loop* pada batasan perekayasa dan pada batas operasional perekayasa.

4 Menggunakan struktur data internal untuk menjamin validasinya [Pre97].

2.2.8 *Tools* Analisis dan Perancangan Sistem

2.2.8.1 *Document Flowmap* (Bagan Alir Dokumen)

Bagan alir dokumen menggambarkan aliran dokumen dan informasi antar *area* pertanggungjawaban di dalam sebuah organisasi. Bagan alir ini menelusuri sebuah dokumen dari asalnya sampai tujuannya. Secara rinci bagan alir ini menunjukkan dari mana dokumen tersebut berasal, distribusinya, tujuan digunakannya dokumen tersebut dan lain-lain. Bagan alir ini bermanfaat untuk menganalisis kecukupan prosedur pengawasan dalam sebuah sistem. Bagan alir dokumen disebut juga bagan alir formulir yang merupakan yang menunjukkan arus dari laporan dan formulir termasuk tembusannya.

2.2.8.2 *Context Diagram* (CD)

Diagram konteks merupakan diagram yang paling awal, yang digunakan untuk menggambarkan sistem dan lingkungan luar sistem yang saling berhubungan. Yang digambarkan dengan lingkaran tunggal yang mewakili seluruh sistem. Sistem dibatasi oleh *boundary*. Dalam diagram konteks hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks.

2.2.8.3 *Data Flow Diagram* (DFD)

Model analisis harus dapat mencapai tiga sasaran utama, yaitu menggambarkan apa yang dibutuhkan oleh pelanggan, membangun dasar bagi pembuatan disain perangkat lunak, membatasi serangkaian persyaratan yang dapat divalidasi begitu perangkat lunak dibangun.

Diagram aliran data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram aliran data adalah memudahkan pemakai atau user yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan. Beberapa simbol yang digunakan dalam *Data Flow Diagram* (DFD) antara lain:

1. Entitas Luar (*External Entity*)

   Sesuatu yang berada di luar sistem, tetapi ia memberikan data kedalam sistem atau memberikan data dari sistem, disimbolkan dengan suatu kotak notasi. *External Entity* tidak termasuk bagian dari sistem. Bila sistem informasi dirancang untuk satu bagian maka bagian lain yang masih terkait menjadi *external entity*.

2. Arus Data (*Data Flow*)

   Arus data merupakan tempat mengalirnya informasi dan digambarkan dengan garis yang menghubungkan komponen dari sistem. Arus data ditunjukkan dengan arah panah dan garis diberi nama atas arus data yang mengalir. Arus data ini mengalir diantara proses, *data store* dan menunjukan arus data yang berupa masukan untuk sistem atau hasil proses sistem.

3. Proses (*Process*)

   Proses merupakan apa yang dikerjakan oleh sistem. Proses dapat mengolah data atau aliran data masuk menjadi aliran data keluar. Proses berfungsi mentransformasikan satu atau beberapa data keluaran sesuai dengan spesifikasi yang diinginkan. Setiap proses memiliki satu atau beberapa masukan serta menghasilkan satu atau beberapa data keluaran.

4. Simpanan Data (*Data Store*)

Simpanan data merupakan tempat penyimpanan data yang ada dalam sistem. *Data store* dapat disimbolkan dengan dua garis sejajar atau dua garis dengan salah satu sisi samping terbuka. Proses dapat mengambil data dari atau memberikan data ke simpanan data (*database*).

2.2.8.4 *Entity Relational Diagram* (ERD)

Entity Relationship Diagram (ERD) merupakan suatu diagram yang digunakan untuk menghubungkan antar elemen (*relational condition*), dimana pada tahap selanjutnya dapat diimplementasikan kedalam bentuk tabel relasi.

ERD hanya berfokus pada data, dengan menunjukkan “jaringan data” yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi dimana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya diusulkan oleh Peter Chen untuk desain sistem *database* relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diidentifikasikan untuk ERD : objek data, atribut, hubungan dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka.

Kardinalitas model data harus dapat merepresentasikan jumlah peristiwa dari objek di dalam hubungan yang diberikan. Kardinalitas merupakan spesifikasi dari sejumlah peristiwa dari satu [objek] yang dapat dihubungkan ke sejumlah peristiwa dari [objek] yang lain. Dengan mempertimbangkan semua kombinasi dari ‘satu’ dan ‘banyak’, dua [objek] dapat dihubungkan sebagai:

1. Satu-ke-satu (1:1)

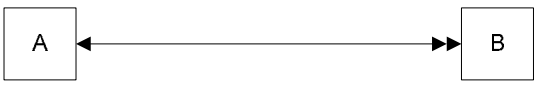
   Tingkat hubungan satu-ke-satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.



Gambar 2.5 Hubungan Relasi Satu Ke Satu

2. Satu-ke-banyak (1:N) atau banyak-ke-satu (N:1)

   Tingkat hubungan satu-ke-banyak adalah sama dengan banyak-ke-satu. Tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua. Sebaliknya satu kejadian pada entitas yang kedua hanya dapat mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.



Gambar 2.6 Hubungan Relasi Satu Ke Banyak

3. Banyak-ke-banyak (N:N)

   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 2.7 Hubungan Relasi Banyak Ke Banyak

2.2.8.5 *Data Dictionary* (Kamus Data)

Salah satu komponen kunci dalam sistem manajemen *database* adalah *file* khusus yang disebut kamus data (*data dictionary*). Kamus data merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Pada tahap analisis dan perancangan, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Pada tahap perancangan, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database*. Kamus data dibuat berdasarkan arus data yang ada di DFD. Arus data di DFD sifatnya global, hanya ditunjukkan nama arus datanya saja. Keterangan lebih lanjut tentang struktur data dari arus data di DFD secara lebih rinci dapat dilihat di kamus data. Kamus data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya, maka kamus data harus memuat nama arus data, alias, bentuk data, arus data, penjelasan, periode, volume, dan struktur data.

Simbol-simbol yang ada dalam kamus data adalah sebagai berikut :

= artinya adalah disusun atas

+ artinya adalah dan

( ) artinya adalah data opsional

[|] artinya adalah memilih salah satu alternatif

{ }^{n} artinya adalah pengulangan ke-*n* dari

* * artinya adalah komentar tidak dibatasi

Contoh penulisan dalam kamus data adalah sebagai berikut :

Tgl_beli = *tgl-bulan-tahun*

Item = kode_brg+qty+harga

Kode_brg = | 0000 | 0001 | 0002 | … | 9999 |

Qty = *999*

Harga = *999.999,99*

2.2.9 Jaringan Komputer

2.2.9.1 Tipe-tipe Jaringan Komputer

1. *Local Area Network* (LAN)

Local Area Network (LAN) adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung. Beberapa model konfigurasi LAN biasanya berupa satu komputer yang dijadikan sebuah *file server* yang digunakan untuk menyimpan perangkat lunak (*software*). Komputer-komputer yang terhubung ke dalam sebuah *file server* biasanya disebut *workstation*. Kemampuan *workstation* lebih kurang di bawah dari *file server*-nya dan mempunyai aplikasi lain di dalam media penyimpanannya selain untuk jaringan.

2. *Metropolitan Area Network* (MAN)

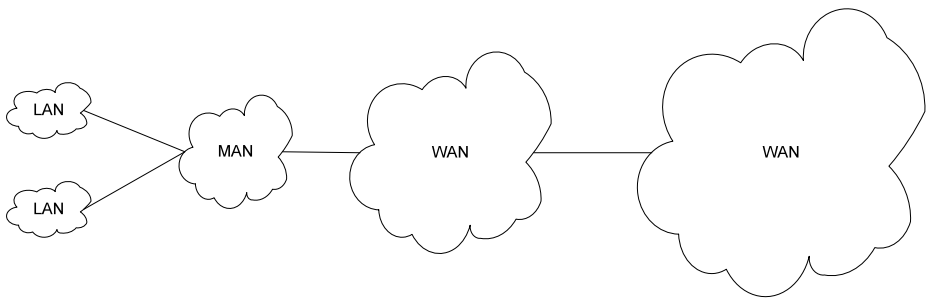
   Metropolitan Area Network (MAN) merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan.

3. *Wide Area Network* (WAN)

   Wide Area Network (WAN) dirancang untuk menghubungkan komputer-yang terletak pada cakupan geografis yang luas, seperti hubungan dari satu kota ke kota lain dalam suatu negara. Jaringan WAN biasanya dimiliki dan dioperasikan sebagai suatu jaringan publik.

4. *Global Area Network* (GAN)

   Global Area Network (GAN) merupakan jaringan yang menghubungkan negara-negara diseluruh dunia. Contoh yang sangat serign dijumpai dari GAN ini adalah *internet.*



Gambar 2.8 Interaksi antara LAN, MAN, WAN, dan GAN

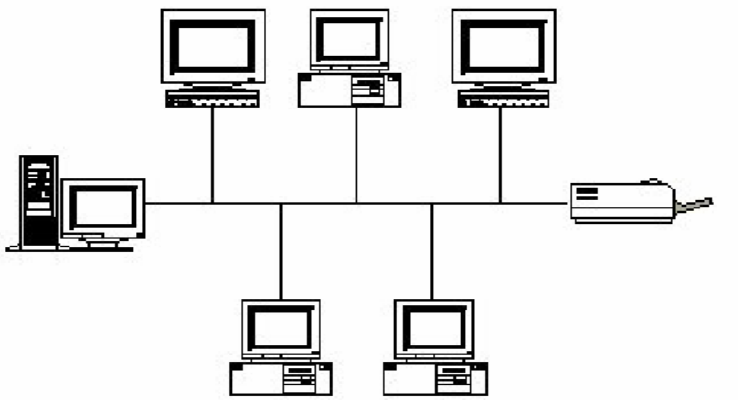
2.2.9.2 Topologi Jaringan Komputer

Topologi secara fisik dari suatu jaringan lokal kepada konfigurasi kabel, komputer, dan perangkat lainnya. Berikut ini adalah topologi fisik yang digunakan dalam jaringan lokal, diantaranya :

1. *Linear Bus* (Garis Lurus)

Topologi *linear bus* 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 ke sebuah kabel utama (*backbone*).

- Kelebihan :
  - Mudah dalam mengkonfigurasi komputer atau perangkat lain ke dalam sebuah kabel utama.
  - *Layout* kabel sederhana dan hemat.
- Kekurangan :
  - Seluruh jaringan akan mati apabila ada kerusakan pada kabel utama (*backbone*).
  - Membutuhkan termintor pada kedua sisi dari kabel utama.
  - Kepadatan lalu lintas.
  - Sangat sulit mengidentifikasi permasalahan jika jaringan rusak.
  - Diperlukan repeater untuk jarak jauh.

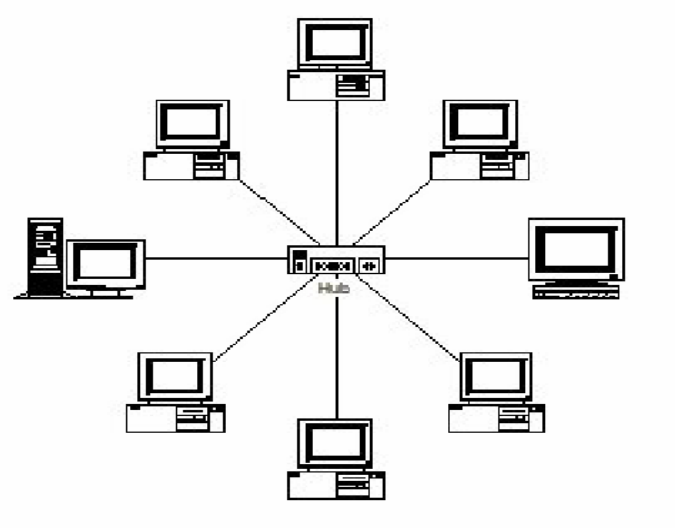


Gambar 2.9 Topologi *Linear Bus* (Garis Lurus)

2. *Star* (Bintang)

Topologi ini dirancang yang mana setiap nodes (*file server*, *workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator* yang mengatur dan mengendalikan keseluruhan fungsi jaringan, dan juga bertindak sebagai *repeater* (penguat aliran data). Konfigurasi jaringan ini menggunakan kabel *Twisted Pair*, dan dapat pula kabel *coaxial* atau kabel *fiber optic.*

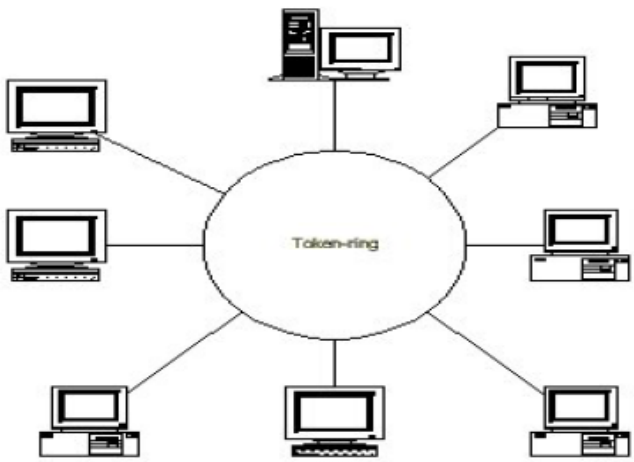
- Kelebihan :
  - Pemasangan/perubahan stasiun sangat mudah dan tidak mengganggu jaringan lain.
  - Kontrol terpusat.
  - Mudah mendeteksi dan isolasi kesalahan/kerusakan.
  - Mudah dalam pengelolaan jaringan.
  - Fleksibel.
- Kekurangan :
  - Membutuhkan lebih banyak kabel dari pada topologi *linear Bus.*
  - Lebih mahal dibandingkan topologi *Linear Bus*, karena pengadaan biaya untuk *concentrator*, dan bilamana *concentrator* rusak maka semua nodes yang terkoneksi tidak dapat terdeteksi.



Gambar 2.10 Topologi *Star* (Bintang)

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.* Kelebihannya hemat kabel dan kekurangannya peka terhadap kesalahan dan pengembangan jaringan lebih kaku.



Gambar 2.11 Topologi *Ring* (Cincin)

4. *Tree* (Pohon)

Topologi ini merupakan perpaduan topologi *Linear Bus* dan *Star*, yang mana terdiri dari kelompok-kelompok *workstion* dengan konfigurasi *star* yang terkoneksi ke kabel utama yang menggunakan topologi *linear Bus*, sehingga dapat memungkinkan pengembangan dan konfigurasi jaringan sesuai dengan kebutuhan.

- Kelebihan :
  - Proses konfigurasi dilakukan dari titik ke titik pada setiap segmen.
  - Didukung oleh banyak perangkat keras dan perangkat lunak.
- Kekurangan :
  - Keseluruhan panjang kabel pada setiap segmen dibatasi oleh tipe kabel yang digunakan.
  - Jika jaringan utama rusak, maka seluruh segmen ikut rusak.
  - Sangat relatif sulit untuk dikonfigurasi dan proses pengkabelannya dibandingkan topologi jaringan lainnya.

2.2.9.3 Manfaat Jaringan Komputer

Manfaat yang didapat dari membangun jaringan komputer adalah sebagai berikut:

1. *Sharing Resources*

   Sharing resources bertujuan agar seluruh program, peralatan lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan komputer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai.

2. Media Komunikasi

   Jaringan komputer memungkinkan terjadinya komunikasi antara pengguna, baik untuk *teleconference* maupun untuk mengirim pesan/informasi yang penting lainnya.

3. Integrasi Data

   Jaringan komputer dapat mencegah ketergantungan pada komputer pusat. Setiap proses data dapat didistribusikan ke tempat lainnya. Oleh sebab inilah maka dapat terbentuk data yang terintegrasi sehingga dengan demikian memudahkan pemakai untuk memperoleh dan mengolah informasi setiap saat.

4. Pengembangan dan Pemeliharaan

   Dengan adanya jaringan komputer, maka perkembangan peralatan dapat dilakukan dengan mudah dan menghemat biaya. Jaringan komputer bisa memudahkan pemakai dalam merawat *hardware*-nya.

5. Keamanan Data

   Sistem jaringan komputer memberikan perlindungan terhadap data. Jaminan keamanan data diberikan melalui pengaturan hak akses para pemakai dan password, serta teknik perlindungan terhadap *harddisk* sehingga data mendapatkan perlindungan yang efektif.

6. Sumber Daya Lebih Efisien dan Informasi Terkini

   Dengan adanya pemakaian sumber daya secara bersama-sama maka pemakai bisa mendapatkan hasil dengan maksimal informasi yang diakses selalu terbaru, karena setiap ada perubahan yang terjadi dapat secara langsung diketahui oleh setiap pemakai.

2.2.9.4 Media Transmisi

Kabel adalah penghubung untuk mengirim informasi dari satu komputer ke komputer lain. Kabel yang dipilih adalah berdasarkan dengan topologi jaringan, protokol jaringan, dan ukurannya. Hal ini sangat penting diketahui karena kesuksesan jaringan bergantung dari semua hal tersebut. Adapun tipe-tipe kabel yang digunakan di dalam jaringan lokal diantaranya :

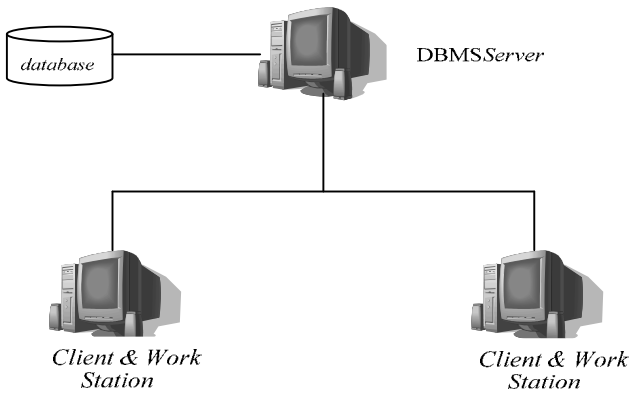
1. Kabel Unshielded Twisted Pair (UTP)
2. Kabel Shielded Twisted Pair (STP)
3. Kabel Koaksial
4. Kabel Fiber Optic

2.2.9.5 Sistem *Client-Server*

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*. *Client* selanjutnya mengupayakan agar semua proses “sebisa mungkin” ditangani sendiri. Bila ada proses yang harus melibatkan data yang tersimpan pada basis data barulah *client* mengadakan hubungan dengan *server*.

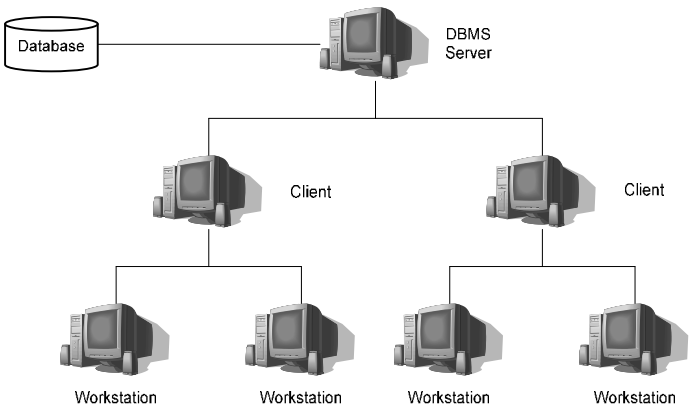
Pada sistem *Client Server* untuk memenuhi kebutuhan, *client* akan mengirimkan *message* (perintah) *query* pengambilan data. Selanjutnya, *server*

yang menerima *message* tersebut akan menjalankan *query* tersebut dan hasilnya akan dikirimkan kembali ke *client*. Dengan begitu transfer datanya jauh lebih efisien. Adapun bentuk dari Sistem *Client Server* yang sederhan adalah sebagai berikut :



Gambar 2.9 Sistem *Client-Server* Sederhana

Disamping bentuk *client server* sederhana terdapat pula bentuk *client server* yang lebih komplek yang digambarkan sebagai berikut :



Gambar 2.10 Sistem *Client-Server* Komplek

Dari kedua gambar di atas, dapat dilihat adanya dua macam implementasi sistem *client server*. Bentuk yang sederhana dapat diterapkan pada sebuah jaringan komputer lokal (LAN) dimana fungsi *client* (untuk menangani sebagian

besar proses pengolahan data seperti perhitungan, perulangan, pembandingan, dan lain-lain.) dan fungsi *work station* (untuk menangani interaksi dengan pemakai, menerima data masukan dan menayangkan hasil pengolahan) disatukan.

Adanya pemisahan fungsi *client* dan fungsi *server*, disamping meningkatkan kompleksitas tersendiri dalam pembangunan aplikasi secara keseluruhan, juga menimbulkan kelemahan lain, yaitu aktivitas pemasangan aplikasi yang tidak praktis. Bila terdapat perubahan/perbaikan aplikasi basis data maka harus mengulangi pekerjaan instalasi disemua mesin *client* yang digunakan. Sedangkan pada varian sistem *client server* yang lebih kompleks, aplikasi basis data tidak ditempatkan di setiap *work station,* tetapi dipasang pada setiap *client* yang jumlahnya jauh lebih sedikit.. Tidak bagaimana *work station* yang diaktifkan dan dinonaktifkan oleh para pemakai, *client-client* tersebut (sebagaimana juga DBMS *server)* harus selalu dalam keadaan aktif dan terkoneksi dalam sebuah jaringan yang lebih besar (WAN). Dengan begitu tahap instalasi aplikasi dapat dilakukan secara jarak jauh (*remote)* dari lokasi lain, sehingga kelemahan dari sisi instalasi dapat diatasi.

2.2.10 Borland Delphi

Borland Delphi atau yang biasa disebut Delphi saja, merupakan sarana aplikasi visual. Bahasa pemrograman yang digunakan adalah bahasa pemrograman Pascal atau yang kemudian disebut juga bahasa pemrograman Delphi. Delphi merupakan generasi penerus dari Turbo Pascal. Turbo Pascal yang diluncurkan pada tahun 1983 dirancang untuk dijalankan pada sistem operasi

DOS (yang merupakan sistem operasi yang paling banyak digunakan pada saat itu). Sedangkan Delphi yang diluncurkan pertama kali tahun 1995 dirancang untuk beroperasi dibawah sistem operasi windows.

Kelebihan yang dimiliki Borland Delphi diantaranya :

1. Form dan komponen-komponennya dapat dipakai ulang dan dikembangkan.
2. Mampu mengakses VBX.
3. Tersedia template aplikasi dan template form.
4. Meliliki lingkungan pengembangan visual yang dapat diatur sesuai kebutuhan.
5. Menghasilkan file terkompilasi yang berjalan lebih cepat.
6. Kemampuan mengakses data dari berbagai macam format.

2.2.11 My SQL Server

My SQL Server adalah sistem bisnis data yang memakai perintah-perintah *Transact-SQL* untuk mengirimkan perintah dari komputer *client* ke komputer *server*. *Transact-SQL* adalah bahasa yang dikembangkan oleh Microsoft dengan menambahkan dialek-dialek tertentu. My SQL Server berisi database, mesin database, dan aplikasi yang diperlukan untuk mengolah data dan komponen-komponennya.

Berikut ialah beberapa contoh perintah-perintah SQL :

1. Menambahkan data (*Insert*)

   Perintah SQL yang digunakan untuk menambah data ke database adalah perintah *INSERT* . Perintah ini mempunyai bentuk umum sebagai berikut:

   INSERT [INTO] nama_tabel [(daftar field/kolom)] VALUES
   (daftar_data)

2. Memilih/Mengakses Data (*Select*)

   Perintah SQL yang digunakan untuk memilih data adalah perintah *SELECT*. Bentuk umum perintah ini adalah sebagai berikut:

   SELECT daftar_select FROM daftar_tabel
   [WHERE kondisi_pencarian]
   [GROUP BY daftar_group_by]
   [HAVING kondisi_pencarian]
   [ORDER BY daftar_order[ASC¦DESC]]

3. Mengedit/Mengubah Data (*Update*)

   Perintah SQL yang digunakan untuk mengedit data adalah perintah *UPDATE SET*. Perintah ini mempunyai bentuk umum sebagai berikut:

   UPDATE nama_table SET field1=databaru1 [data2=databaru2]
   [WHERE kondisi_update]

4. Menghapus Data (*Delete*)

   Perintah SQL yang digunakan untuk menghapus data adalah perintah *DELETE FROM*. Perintah ini mempunyai bentuk umum sebagai berikut:

   DELETE FROM tabel_hapus WHERE kondisi_hapus

2.2.12 Koneksi Delphi dengan My SQL

Telah diketahui bahwa BDE (Borland Delphi Engine) adalah salah satu teknologi akses data yang sudah sangat populer di Delphi. Hingga saat ini BDE belum dapat mengakses server My SQL secara langsung seperti yang dilakukan pada *interbase* BDE dapat mengakses server SQL seperti My SQL Server dengan memanfaatkan driver ODBC buatan Microsoft, tetapi driver tersebut tidak dapat mengakses basis data My SQL.

dbExpress merupakan komponen yang disediakan oleh Delphi untuk membangun aplikasi *client-server*. dbExpress tidak mendukung *software* database *desktop* yang menjadi fokus BDE dan ADO. dbExpress menyediakan dukungan akses data terhadap *interbase*. My SQL Server, Oracle, My SQL, Post Gress, Sybase, dan DB2.Borland menganjurkan agar aplikasi basis data *client-server* dibuat menggunakan dbExpress.

2.2.13 Pengertian Penjualan

Penjualan adalah merupakan salah satu ujung tombak dari perusahaan, sebab tujuan utama bagi perusahaan setelah perancangan menejemen dibuat secara baik sesuai dengan sistem-sistem yang akurat penjulanlah yang menjadi utama untuk memperoleh laba dari hasil penjualan sebagai kelangsungan hidup perusahaan itu dapat dipertahankan dan direalisasikan. Jadi dengan adanya penjualan dapat tercipta suatu proses pertukaran barang dan atau jasa antara penjual dengan pembeli.

Kegiatan menjual terbagi menjadi dua cara yaitu :

1. Penjualan kredit yaitu penjualan barang yang dilakukan dengan cara mengirim barang sesuai dengan pesanan dari pembeli dan tidak ada pembayaran langsung yang terjadi dalam jangka waktu tertentu dan perusahaan mempunyai tagihan pada pembeli barang.
2. Penjualan tunai adalah apabila perusahaan tersebut menjadi produknya secara langsung pada pembeli dan dibayar pada saat itu juga oleh pembeli.

Jenis-jenis penjualan dikelompokkan menjadi :

1. *Trade Selling*

   Trade selling dapat terjadi bilamana produsen dan pedagang besar mempersilahkan pengecer-pengecer berusaha memperbaiki distributor produk-produk mereka.

2. *Missionary Selling*

   Dalam *missionary selling*, penjualan berusaha ditingkatkan dengan mendorong pembeli untuk membeli barang-barang dari penyalur perusahaan.

3. *Technical Selling.*

   Technical selling berusaha meningkatkan penjualan dengan pemberian saran dan nasehat kepada pembeli akhir dari barang dan jasanya.

4. *New Bussines Selling.*

   New bussines selling berusaha membuka transaksi baru dengan merubah calon pembeli menjadi pembeli.

5. *Responsive Selling.*

   Jenis penjualan ini tidak akan menciptakan penjualan yang terlalu besar meskipun layanan yang baik dan hubungan pelanggan yang menyenangkan dapat menjurus kepada pembeli ulang, misalnya para pengemudi yang mengantarkan susu, para pelayan ditoko serba ada.