BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Sistem memiliki dua pendekatan yang dapat digunakan untuk mendefinisikan arti sistem itu sendiri, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemen sistem.

Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang berupa urutan kegiatan yang saling berhubungan dan berkumpul bersama-sama untuk mencapai tujuan tertentu. Prosedur adalah rangkaian operasi klerikal (tulis menulis) yang melibatkan beberapa orang di dalam satu atau lebih departemen yang digunakan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi serta untuk menyelesaikan suatu kegiatan tertentu.

Sedangkan pendekatan sistem yang lebih menekankan pada komponen/elemennya mendefinisikan sistem sebagai kumpulan komponen yang saling berkaitan dan bekerja sama untuk mencapai suatu tujuan tertentu.

> *System : In data processing, a collection of men, machines, and method organized to accomplish a set of specific functions.*

Dengan kata lain, sistem dalam pemrosesan data merupakan kumpulan manusia, mesin, dan metode yang terorganisir untuk menyelesaikan suatu fungsi yang spesifik. Definisi sistem menurut **Webster's Unabridged Dictionary** adalah elemen-elemen yang saling berhubungan membentuk satu-kesatuan atau organisasi.

Sistem harus memiliki minimum empat elemen utama, yaitu :

1. *Input*/Masukan

   Menurut JOG[3] *input* atau masukan adalah :

   "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. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran".

   Sebagai contoh di dalam sistem komputer : program adalah *maintanence input* yang digunakan untuk mengoperasikan komputer, sedangkan data adalah *signal input* yang digunakan untuk menghasilkan informasi.

2. Pemroses/Pengolah

   Pemroses/pengolah merupakan bagian sistem yang akan mengubah masukan menjadi keluaran.

3. *Output*/Pengeluaran

   Output/pengeluaran yaitu hasil dari energi yang diolah dan diklasifikasikan menjadi sesuatu yang berguna.

4. Umpan Balik/Kontrol

   Umpan balik/kontrol yaitu pengendalian sistem yang memperbaiki kinerja sistem secara kontinue.

Sistem dapat diklasifikasikan dari beberapa sudut pandangan, diantaranya adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*physical system*).

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, sedangkan sistem fisik adalah sistem yang ada secara fisik.

2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*).

   Sistem alamiah adalah sistem yang terjadi melalui proses alam dan tidak dibuat oleh manusia, sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia dan melibatkan interaksi antara manusia dengan mesin.

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 diantara bagian-bagiannya dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi dengan tepat, karena mengandung unsur probabilitas.

4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

   Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya (kenyataanya tidak ada sistem yang bener-benar tertutup, yang ada hanya *relatively closed system*), sedangkan sistem terbuka merupakan sistem yang saling berhubungan dan terpengaruh dengan lingkungan luarnya.

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu :

1. Komponen (*component*)

   Komponen terdiri dari sejumlah komponen yang saling berinteraksi dan bekerja sama membentuk suatu kesatuan. Komponen-komponen dapat terdiri dari beberapa subsistem atau subbagian (setiap subsistem tersebut memiliki fungsi khusus dan akan mempengaruhi proses sistem secara keseluruhan). Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut dengan *supra system*.

2. Batas Sistem (*boundary*)

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

3. Lingkungan Luar Sistem (*environment*)

   Lingkungan luar sistem adalah apapun yang ada diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus tetap dijaga dan dipelihara, sedangkan yang merugikan sistem harus ditahan dan dikendalikan, kalau tidak akan menganggu kelangsungan hidup dari sistem tersebut.

4. Penghubung (*interface*)

   Penghubung suatu sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Keluaran (*output*) dari suatu subsistem akan

menjadi masukan (*input*) bagi subsistem yang lainnya dengan melalui penghubung sistem ini.

5. Masukan (*input*)

   Masukan adalah 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. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran

6. Keluaran (*output*)

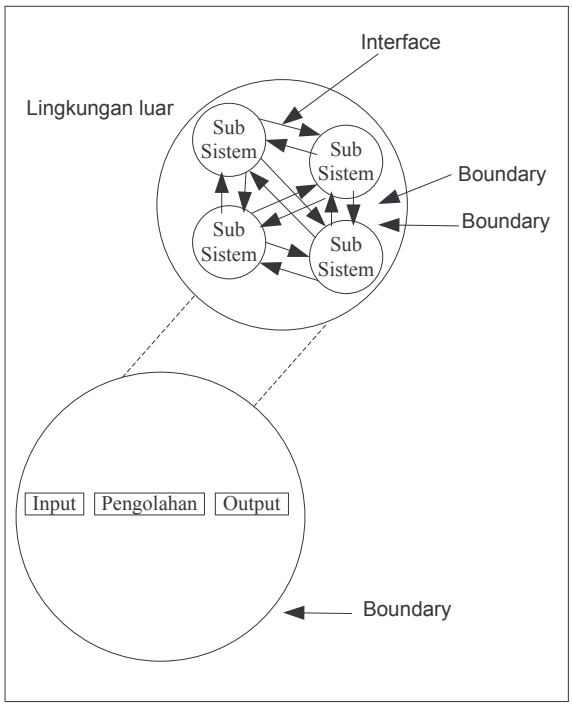
   Output/pengeluaran yaitu hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada *supra system*.

7. Pengolahan (*process*)

   Pemroses/pengolah merupakan bagian sistem yang akan mengubah masukan menjadi keluaran.

8. Sasaran (*objective*) dan Tujuan (*goal*)

   Suatu sistem pasti mempunyai tujuan dan sasaran. Sasaran dari sistem sangat menentukan sekali input yang dibutuhkan sistem dan output yang akan dihasilkan sistem. Sistem dikatakan berhasil jika mengenai sasaran atau tujuannya.



Gambar 2.1 Karakteristik Suatu Sistem
(sumber : Jogiyanto, HM, 2001, *Analisis dan Desain Sistem Informasi,* Andi Offset, Yogyakarta)

2.2 Konsep Dasar Informasi

Informasi adalah hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*events*) yang nyata (*fact*) yang digunakan untuk pengambilan keputusan.

Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum atau data-item. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (*events*) adalah sesuatu yang terjadi pada saat yang tertentu. Kesatuan nyata (*fact and entity*) berupa suatu obyek nyata seperti tempat, benda, dan orang yang betul-betul ada dan terjadi.

Data merupakan bentuk yang masih mentah dan belum dapat bercerita, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk

dihasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf-huruf atau alphabet, angka-angka, suara, sinyal, gambar, dan sebagainya.

Kualitas dari informasi (*quality of information*) tergantung dari tiga hal, yaitu :

1. Akurat (*Accurate*)

   Informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas dalam mengutarakan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dpat merubah atau merusak informasi tersebut.

2. Tepat waktu (*timeliness*)

   Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan.

3. Relevan (*relevance*)

   Informasi tersebut mempunyai manfaat bagi pemakainya. Relevan untuk tiap-tiap orang berbeda.

2.3 Konsep Dasar Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal

yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang tepat.

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

1. Blok masukan (*input block*)

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

2. Blok model (*model block*)

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

3. Blok keluaran (*output block*)

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

4. Blok teknologi (*technology block*)

   Teknologi merupakan “kotak alat” (*tool-box*) dari pekerjaan sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, serta membantu pengendalian dari sistem keseluruhan. Teknologi terdiri dari dua bagian utama, yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*). Perangkat lunak merupakan program yang membuat perankat keras dapat bekerja dengan menginstruksikannya untuk memproses sesuai

dengan model yang diterapkan. Sedangkan perangkat keras terdiri dari bermacam-macam alat yang menyediakan dukungan fisik untuk blok-blok lainnya.

5. Blok basis data (*database block*)

   Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, disimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (*control block*)

   Agar sistem informasi dapat berjalan sesuai dengan yang diinginkan, maka perlu diterapkan pengendalian-pengendalian di dalamnya. Beberapa pengendali perlu diterapkan dan dirancang untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.

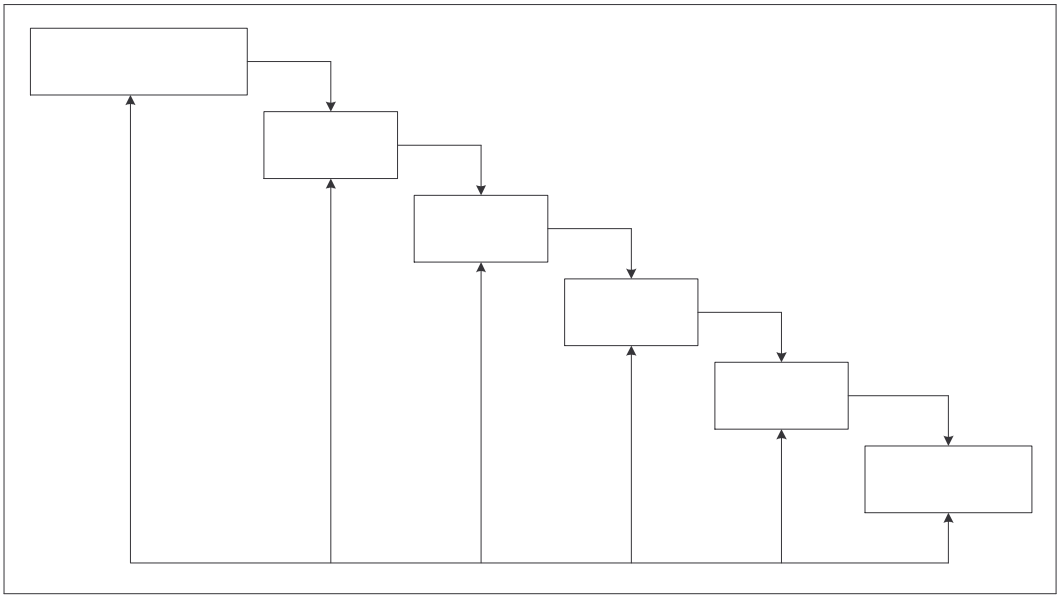
2.4 Metode Pengembangan Sistem

Dalam pengembangan sistem, paradigma yang digunakan adalah *WaterFall model.* Langkah-langkah yang perlu dilakukan antara lain adalah sebagai berikut :

1. *System Engineering*, adalah tahap rekayasa system yang ada pada tahap ini menentukan kebutuhan-kebutuhan pada system yang akan dibangun.

2. Analisis, tahap menganalisis hal-hal yang diperlukan dalam pelaksanaan proyek pembuatan atau pengembangan perangkat lunak atau *software*.
3. Desain, tahap penerjemahan dari keperluan-keperluan yang telah dianalisis ke dalam bentuk yang lebih mudah dan dimengerti oleh pengguna *(user).*
4. *Coding*, tahap penerjemahan data atau pemecahan masalah perangkat lunak yang telah direncang ke dalam bahasa pemrograman yang telah ditentukan.
5. Testing, tahap pengujian terhadap program yang telah dibuat.
6. *Implementation* adalah tahap implementasi dan pemeliharaan perangkat lunak dari keseluruhan system bila terjadi kesalahan pada program.

Dari penjelasan di atas dapat digambarkan sebagai berikut :



Gambar 2.2 *Waterfall Model* (*Classic Life Cycle*)
(Sumber : Jogiyanto, HM, 2001, *Analisis dan Desain Sistem Informasi,* Andi Offset, Yogyakarta)

2.5 Analisis dan Perancangan Terstruktur

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponen dengan maksud untuk

mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, dan hambatan-hambatan yang terjadi dalam kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan.

Perancangan sistem adalah proses perancangan, pengembangan sistem, pendefinisian kebutuhan-kebutuhan fungsional dan persiapan untuk sistem yang akan dibentuk, alat pemodelan perancangan sistem adalah :

2.5.1 Diagram Konteks

Context diagram adalah bagian dari *Data Flow Diagram* (DFD) yang berfungsi memetakan model lingkungan, yang direpresentasikan dengan lingkungan tinggal yang mewakili sistem.

Berikut ini merupakan simbol-simbol yang ada dalam *diagram context*:

1. Proses

   Merupakan komponen utama dalam model ini karena proses dinamakan sebagai suatu fungsi dan tranformasi dari masukan *(Input)* maupun keluaran *(Output)*, dalam hal ini sejumlah masukan dapat menjadi satu keluaran ataupun sebaliknya.

2. Entitas

   Merupakan suatu komponen berikutnya yang direpresentasikan dengan menggunakan persegi panjang yang mewakili suatu entity luar dimana sistem berkomunikasi. Biasanya notasi seperti dilambangkan dengan orang ataupun kelompok.

3. Aliran Data

   Merupakan suatu komponen yang dipresentasikan dengan mempergunakan suatu panah yang menuju dari ataupun ke proses, hal ini digunakan untuk menggambarkan gerakan paket data ataupun informasi dari satu bagian kepada bagian lain dari suatu sistem yang dimana penyimpannya mewakili lokasi penyimpanan data.

2.5.2 Data Flow Diagram

Data Flow Diagram (DFD) adalah model yang menggambarkan sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data. *Data Flow Diagram* (DFD) merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured analysis and design*). *Data Flow Diagram* (DFD) juga merupakan dokumentasi dari sistem yang baik

Beberapa simbol yang digunakan di *Data Flow Diagram* (DFD) antara lain :

a. Kesatuan Luar *(External Entity)*

Setiap sistem pasti mempunyai batas sistem yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem yang lain yang berada di lingkungan luar yang akan memberikan *input* atau menerima *output* dari sistem. Kesatuan luar ini kebanyakan adalah salah satu dari berikut ini :

1. Suatu kantor, departemen atau divisi dalam perusahaan tetapi di luar sistem yang sedang dikembangkan.
2. Orang atau sekelompokan orang yang diorganisasi tetapi diluar sistem yang sedang dikembangkan.
3. Suatu organisasi atau orang yang berada diluar organisasi seperti misalnya langganan, pemasok.
4. Sistem informasi yang diluar sistem yang sedang dikembangkan.
5. Sumber asli dari suatu transaksi.
6. Penerima akhir dari suatu laporan yang dihasilkan oleh sistem.

b. Arus Data

Arus data di *Data Flow Diagram* (DFD) diberi simbol suatu panah. Arus data ini mengalir diantara proses, simpanan data dan kesatuan luar . arus data ini menunjukkan arus dari data yang dapat berupa msukan untuk sistem atau hasil dari proses sistem.

c. Proses

Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses.

d. Simpanan Data (*Data Store*)

Simpanan data merupakan simpanan dari data yang dapat berupa sebagai berikut ini :

1. Suatu file atau database di sistem komputer.
2. Suatu arsip atau catatan manual .
3. Suatu kotak tempat data di meja seseorang.

4. Suatu tabel acuan manual.
5. Suatu agenda atau buku.

Simpanan data di *Data Flow Diagram* (DFD) dapat disimbolkan dengan sepasang garis horizontal paralel yang kedua ujungnya terbuka atau salah satu ujungnya tertutup.

2.5.3 Kamus Data

Kamus data / *system Data Dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan kamus data, analisis sistem dapat mendefinisikan data yang mengalir dari dalam sistem dengan lengkap. Pada tahap analisis sistem, kamus data digunakan sebagai alat komunikasi antara analis sistem dengan pemakai sistem.

Data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatatnya. Untuk keperluan seperti ini , maka kamus data harus membuat hal-hal sebagai berikut :

1. Alias adalah nama lain dari data, dapat dituliskan bila nama lain ini ada. Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk organisasi/departemen dengan lainnya.
2. Bentuk data yang mengalir dapat berupa dokumen dasar/formulir, dokumen hasil cetakan komputer, laporan tercetak, tampilan dilayar monitor, variabel, parameter, dan *field*.
3. Arus data menunjukkan darimana data mengalir dan kemana data akan dituju.

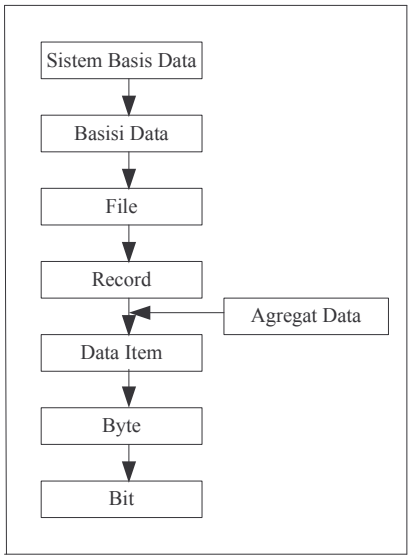
4. Penjelasan digunakan untuk memperjelas tentang makna dari arus data yang dicatat di kertas data, maka bagian penjelasan dapat diisi tentang arus data tersebut.
5. Periode, menunjukkan kapan terjadinya arus data tersebut. Periode digunakan untuk mendefinisikan kapan *input* data harus dimasukkan ke sistem, kapan proses data program harus dilakukan dan laporan-laporan harus dihasilkan.
6. Volume, yang dicatat adalah volume rata-rata dan volume puncak dari arus dokumen.
7. Struktur data, menunjukkan arus data yang dicatat dikamus data yang terdiri dari item-item.

2.6 Skema Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. *Database* merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai. Penerapan *database* dalam sistem informasi disebut dengan *database system.* Sistem basis data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

Berdasarkan tingkat kompleksitas nilai data, tingkatan data dapat disusun dalam sebuah hirarki, mulai dari yang sederhana hingga paling komplek. Susunan

atau hirarki data hingga tersusun suatu sistem basis data ditunjukkan sebagaimana gambar 2.3 :



Gambar 2.3 Hirarki Data
(Sumber : Sutanta, Edhy, 2004, *Sistem Basis Data,* Graha Ilmu, Yogyakarta)

1. Basis data, merupakan sekumpulan dari bermacam-macam tipe *record* yang memiliki hubungan antar *record* dan nilai data terhadap objek tertentu.
2. Berkas atau *file*, merupakan sekumpulan *record* sejenis secara relasi yang tersimpan dalam media penyimpan sekunder.
3. *Record*, merupakan sekumpulan *field* atau atribut atau data item yang saling berhubungan terhadap objek tertentu.
4. *Field* atau atribut atau data item, merupakan unit terkecil yang disebut data, yaitu sekumpulan byte yang mempunyai makna.
5. Byte adalah bagian terkecil yang dialamatkan dalam memori. Byte nerupakan sekumpulan bit yang secara konvensional terdiri atas kombinasi delapan bit yang menyatakan sebuah karakter dalam memori.

6. Bit adalah sistem biner yang terdiri atas dua macam nilai, yaitu 0 dan 1. sistem biner merupakan dasar yang dapat digunakan untuk komunikasi antara manusia dan mesin (komputer) yang merupakan serangkaian komponen elektronik dan hanya dapat membedakan dua macam keadaan, yaitu ada tegangan dan tidak ada tegangan yang masuk ke rangkaian tersebut.

2.6.1 Kunci Relasi

Kunci relasi diperlukan dalam rangka untuk pengaksesan data dari dalam relasi atau untuk menyusun kerelasian antar relasi. Kunci relasi merupakan satu atau gabungan atribut yang bersifat unik yang dapat digunakan untuk mengidentifikasi atau membedakan setiap *record* dalam relasi. Dengan demikian, kunci relasi harus bersifat unik, artinya nilai-nilai elemen data atau entri dalam atribut yang digunakan sebagai kunci relasi tidak boleh ada yang sama untuk seluruh *record* dalam relasi.

Berdasarkan macamnya kunci relasi terdiri atas :

1. Kunci kandidat (*candidate key* atau ck), yaitu satu atau gabungan minimal atribut yang bersifat unik yang dapat digunakan untuk mengidentifikasi atau membedakan setiap *record* dalam relasi. Dalam setiap relasi minimal mempunyai sebuah kunci kandidat.
2. Kunci primer (*primary key* atau pk), yaitu bagian atau salah satu dari *candidate key* yang dipilih atau digunakan sebagai kunci utama untuk mengidentifikasi atau membedakan setiap *record* dalam relasi. Dalam setipa relasi harus mempunyai *primary key* dan jumlahnya hanya satu buah. *Primary key* harus unik dan tidak boleh null.

3. Kunci alternatif (*alternate key* atau ak), yaitu bagian dari *candidate key* yang tidak dipilih atau digunakan sebagai *primary key*. Dalam setiap relasi tidak harus mempunyai *alternate key*. Hal ini bergantung pada jumlah *candidate key* yang ada. Jika jumlah dalam sebuah relasi lebih dari satu, maka salah satu akan digunakan sebagai *candidate key* dan satu yang lainnya menjadi *alternate key*. Tetapi apabila sebuah relasi hanya memiliki sebuah *candidate key*, maka ia akan digunakan sebagai *primary key* dan tidak ada *alternate key*.
4. Kunci penghubung atau sering juga disebut sebagai kunci tamu atau kunci asing (*foreign key* atau fk), yaitu satu atau gabungan sembarang atribut yang menjadi *primary key* dalam relasi lain yang mempunyai hubungan secara logik. *Foreign key* tidak harus dimiliki dalam relasi. Jika *foreign key* muncul dalam sebuah relasi, maka *foreign key* tersebut akan menunjukkan adanya kerelasian antar relasi dalam basis data. Dalam kerelasian antar relasi tersebut, relasi yang mengacu atau mereferensi pada relasi lain disebut sebagai relasi anak, sedangkan relasi yang menjadi acuan atau referensi disebut sebagai relasi induk. Foreign key atau *primary key* tersebut harus mempunyai tipe dan ukuran data yang sama.

2.6.2 Normalisasi

Menurut FAT[2] Normalisasi dapat membantu pencegahan penciptaan struktur tabel yang kurang *fleksibel* atau mengurangi ketidak efisienan pada penyimpanan. Tujuan dari normalisasi adalah untuk mengurangi anomali atau penyimpangan informasi pada proses *insert*, *update*, dan *delete*. Penyimpangan ini

harus dipecahkan pada beberapa tabel karena dapat menyulitkan *validasi* dan *duplikasi* data. Depedensi merupakan konsep yang mendasari normalisasi. Dependensi menjelaskan nilai suatu attribut yang menentukan nilai attribut lainnya. Dependensi ini kelak menjadi acuan bagi pendekomposisian data ke dalam bentuk yang paling efisien.

Adapun bentuk-bentuk normal suatu table adalah sebagai berikut :

1. Bentuk Normal Pertama

   Suatu relasi dikatakan dalam bentuk normal pertama jika dan hanya jika atomik, yaitu tidak ada attribut berulang pada suatu skema relasi dan tidak duplikasi, yaitu nilai dari attribut tidak ada yang bernilai ganda.

2. Bentuk Normal Kedua

   Suatu relasi dikatakan dalam bentuk normal kedua jika sudah memenuhi bentuk normal pertama dan semua attribut bukan kunci memiliki dependensi sepenuhnya terhadap kunci primer atau tidak ada ketergantungan fungsi sebagian (*Partial Functional Dependency*).

3. Bentuk Normal Ketiga

   Suatu relasi dikatakan dalam bentuk normal ketiga jika sudah memenuhi bentuk normal kedua dan setiap attribut bukan kunci tidak memiliki dependensi transitif terhadap kunci primer.

Dari penjelasan-penjelasan diatas dapat diambil kesimpulan umum bahwa normalisasi dilakukan untuk menghindarkan *redudansi field-field* yang ada.

2.6.3 ERD *(Entity Relationship Diagram)*

Entity Relationship Diagram (ERD) adalah komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari dunia nyata yang ditinjau.

Entity Relationship Diagram (ERD) terdiri dari empat macam relasi dalam hubungan atribut dalam satu file di antaranya adalah sebagai berikut :

1. *One to One*

   Hubungan antara file pertama dengan file kedua adalah satu berbanding satu.

2. *One to Many*

   Hubungan antara file pertama dengan file kedua adalah satu berbanding kebanyak.

3. *Many to one*

   Hubungan antara file pertama dengan file kedua adalah banyak berbanding ke satu.

4. *Many to Many*

   Hubungan antara file pertama dengan file kedua adalah banyak berbanding ke banyak.

2.6.4 Tabel Relasi

Integrity constrain table atau tabel relasi berisikan daftar keterkaitan data yang terjadi dan langkah atau batasan apa yang berlaku bila terjadi *insert*, *update*, dan *delete*.

Kardinalitas suatu relasi data dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka, dimana angka 1 untuk relasi satu ke satu, 1 dan N untuk relasi dari satu ke banyak atau N ke N untuk relasi banyak ke banyak. Untuk menentukan atribut sebagai *key* dilakukan dengan menggaris bawahi nama atribut tersebut.

2.7 Arsitektur Aplikasi

Arsitektur aplikasi disini menjelaskan mengenai definisi jaringan komputer, jenis-jenis jaringan komputer, topologi jaringan komputer, dan manfaat jaringan komputer.

2.7.1 Pengertian Jaringan Komputer

Jaringan komputer adalah sebuah kumpulan komputer dan peralatan lainnya yang terhubung dalam satu kesatuan. Informasi dan data bergerak melalui kabel-kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data.

Komponen dari suatu jaringan komputer yang terhubung dengan jaringan adalah *Node dan Link***.** *Node* adalah titik yang dapat menerima *input* data ke dalam jaringan atau menghasilkan *output* informasi, *node* juga dapat berupa sebuah *printer* atau alat-alat cetak lainnya atau komputer *mikro* sampai komputer raksasa atau dapat disebut sebagai modem. Sedangkan *Link* adalah kanal atau jalur transmisi untuk arus informasi. *Link* dapat berupa kabel, sistem gelombang mikro, laser atau sistem satelit.

2.7.2 Jenis-Jenis Jaringan Komputer

Menurut FAT[2] jaringan komputer dibagi menjadi lima jenis, yaitu:

a. *Local Area Network* (LAN)

Local Area Network (LAN), merupakan jaringan milik pribadi di dalam sebuah gedung yang berukuran sampai beberapa kilometer. LAN seringkali digunakan untuk menghubungkan komputer-komputer pribadi dan *workstation* dalam kantor suatu perusahaan atau pabrik-pabrik untuk pemakaian bersama sumber daya dan saling bertukar informasi.

b. *Metropolitan Area Network* (MAN)

Metropolitan Area Network (MAN), pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya menggunakan teknologi yang sama dengan LAN. MAN dapat mencakup kantor-kantor perusahaan yang letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum.

c. *Wide Area Network* (WAN)

Wide Area Network (WAN), jangkauannya mencakup daerah geografis yang luas, seringkali mencakup sebuah negara bahkan benua. WAN terdiri dari kumpulan mesin-mesin yang bertujuan untuk menjalankan program-program (aplikasi) pemakai.

d. Internet

Sebenarnya terdapat banyak jaringan didunia ini, seringkali menggunakan perangkat keras dan perangkat lunak yang berbeda-beda . Orang yang terhubung ke jaringan sering berharap untuk bisa berkomunikasi dengan orang lain yang terhubung ke jaringan lainnya. Keinginan seperti ini memerlukan hubungan antar

jaringan yang seringkali berbeda. Biasanya untuk melakukan hal ini diperlukan sebuah mesin yang disebut *gateway* guna melakukan hubungan dan melaksanakan terjemahan yang diperlukan, baik perangkat keras maupun perangkat lunaknya. Kumpulan jaringan yang terinterkoneksi seperti ini yang disebut dengan internet.

e. JaringanTanpa Kabel

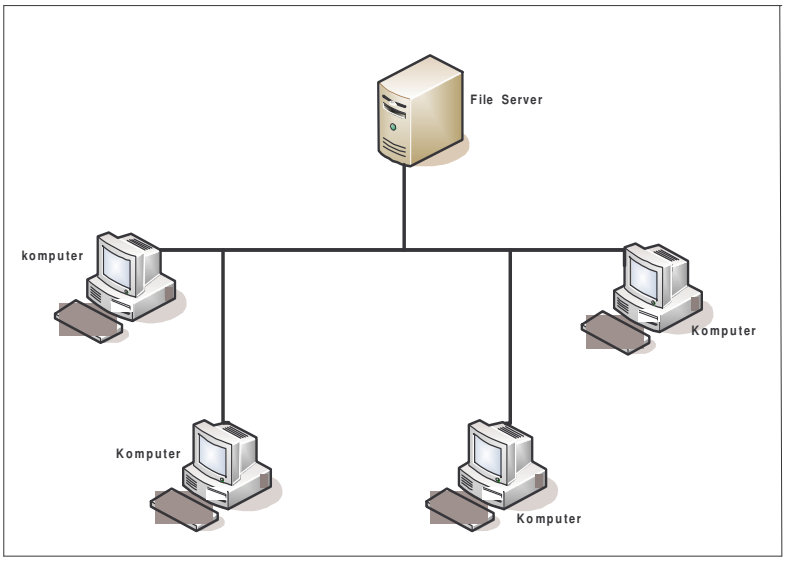
Jaringan tanpa kabel merupakan suatu solusi terhadap komukasi yang tidak bisa dilakukan dengan jaringan yang menggunakan kabel. Saat ini jaringan tanpa kabel sudah sangat banyak digunakan dengan memanfaatkan jasa satelit dan mampu memberikan kecepatan akses yang lebih cepat dibandingkan dengan jaringan yang menggunakan kabel.

2.7.3 Topologi Jaringan Komputer

Yang dimaksud dengan topologi jaringan adalah susunan fisik tentang bagaimana *node-node* tersebut saling dihubungkan. Ada tiga topologi yang sering digunakan:

1. Topologi Bus

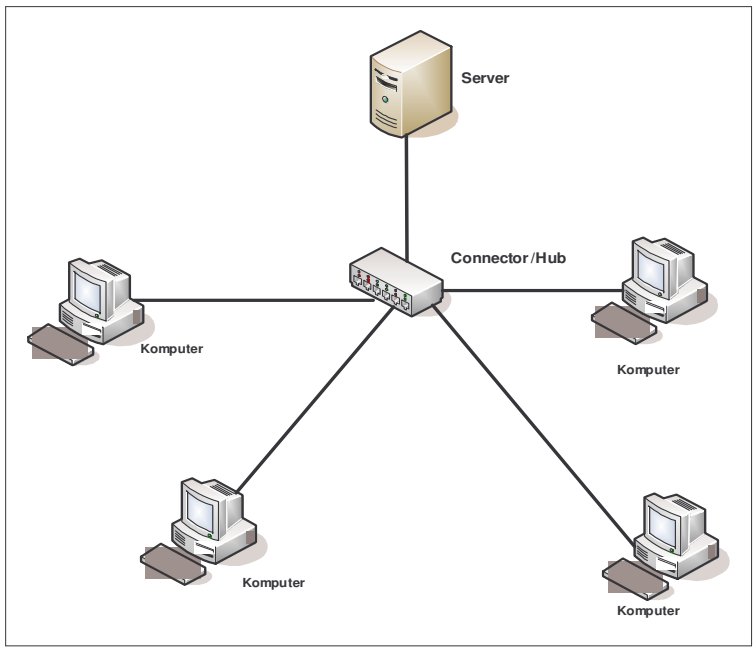
Pada topologi Bus digunakan sebuah kabel tunggal atau kabel pusat di mana seluruh *workstation* dan *server* dihubungkan. Keunggulan topologi Bus adalah pengembangan jaringan atau penambahan *workstation* baru dapat dilakukan dengan mudah tanpa mengganggu *workstation* lain. Kelemahan dari topologi bus ini adalah bila terdapat gangguan di sepanjang kabel pusat maka keseluruhan jaringan akan mengalami gangguan.



Gambar 2.4 Topologi Bus
(Sumber: Dede Sopandi, 2005, *Instalasi dan Konfigurasi Jaringan Komputer*, CV. Informatika, Bandung)

2. Topologi Star

Pada topologi Star, masing-masing *workstation* dihubungkan secara langsung ke *server* atau *hub*. Keunggulan dari topologi tipe Star ini adalah bahwa dengan adanya kabel tersendiri untuk setiap *workstation* ke *server*, maka *bandwidth* atau lebar jalur komunikasi dalam kabel akan semakin lebar sehingga akan meningkatkan unjuk kerja jaringan secara keseluruhan. Dan juga bila terdapat gangguan di suatu jalur kabel maka gangguan hanya akan terjadi dalam komunikasi antara *workstation* yang bersangkutan dengan *server*, jaringan secara keseluruhan tidak mengalami gangguan. Kelemahan dari topologi Star adalah kebutuhan kabel yang lebih besar dibandingkan dengan topologi lain.

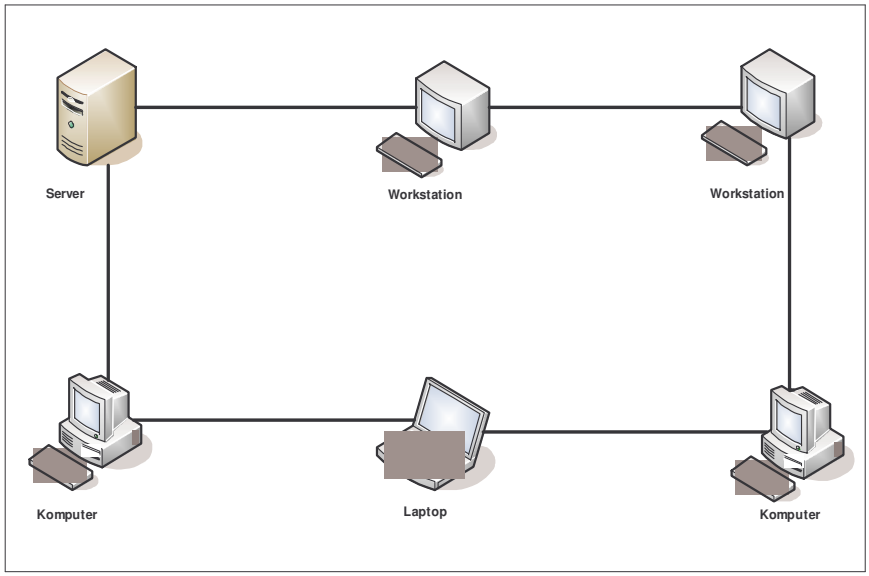


Gambar 2.5 Topologi Star
(Sumber: Dede Sopandi, 2005, *Instalasi dan Konfigurasi Jaringan Komputer*, CV. Informatika, Bandung)

3. Topologi Ring

Di dalam topologi Ring semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin. Tiap *workstation* ataupun *server* akan menerima dan melewatkan informasi dari satu komputer ke komputer lain, bila alamat- alamat yang dimaksud sesuai maka informasi diterima dan bila tidak informasi akan dilewatkan.

Kelemahan dari topologi ini adalah setiap *node* dalam jaringan akan selalu ikut serta mengelola informasi yang dilewatkan dalam jaringan, sehingga bila terdapat gangguan di suatu *node* maka seluruh jaringan akan terganggu. Keunggulan dari topologi Ring adalah tidak terjadinya *collision* atau tabrakan pengiriman data seperti pada topologi Bus, karena hanya satu node dapat mengirimkan data pada suatu saat.



Gambar 2.6 Topologi Ring
(Sumber: Dede Sopandi, 2005, *Instalasi dan Konfigurasi Jaringan Komputer*, CV. Informatika, Bandung)

2.7.4 Manfaat Jaringan Komputer

Manfaat dari jaringan komputer adalah sebagai berikut :

1. *Resource Sharing*, dapat menggunakan sumberdaya yang ada secara bersama-sama. Misal seorang pengguna yang berada 100 km jauhnya dari suatu data, tidak mendapatkan kesulitan dalam menggunakan data tersebut, seolah-olah data tersebut berada didekatnya. Hal ini sering diartikan bahwa jaringan komputer mangatasi masalah jarak.
2. *Reliabilitas* tinggi, dengan jaringan komputer kita akan mendapatkan *reliabilitas* yang tinggi dengan memiliki sumber-sumber alternatif persediaan. Misalnya, semua file dapat disimpan atau di *copy* ke dua, tiga atau lebih komputer yang terkoneksi ke jaringan. Sehingga bila salah satu mesin rusak, maka salinan di mesin yang lain bisa digunakan.
3. Menghemat uang, komputer berukuran kecil mempunyai rasio harga/kinerja yang lebih baik dibandingkan dengan komputer yang besar. Komputer besar

seperti *mainframe* memiliki kecapatan kira-kira sepuluh kali lipat kecepatan jaringan komputer kecil/pribadi. Akan tetap, harga *mainframe* seribu kali lebih mahal dari komputer pribadi. Ketidak seimbangan rasio harga/kinerja dan kecepatan inilah membuat para perancang sistem untuk membangun sistem yang terdiri dari jaringan komputer-komputer pribadi.

2.8 Pengertian Client/Server

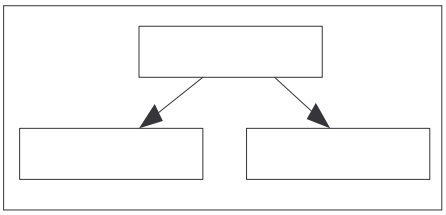
Sistem *client-server* mempunyai dua komponen utama yaitu komputer *client* dan komputer *server*. *Server* merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintan-permintaan dari komputer *client* dan bertindak sebagai *server database* yang menyimpan data. *Client* yaitu komputer atau *workstation* yang melakukan pengiriman permintaan-permintaan data pada *server* kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya. Selain itu client juga mempunyai kemampuan untuk mengubah, melihat, dan mencetak data.

Sistem *client-server* merupakan suatu sistem *client* komputer yang melibatkan prosedur-prosedur *client* yang meminta suatu pelayanan data kepada komputer *server* yang menyediakan layanan data tersebut, sehingga *client* maupun *server* sama-sama melakukan pekerjaan. Dengan adanya kombinasi *client* dan *server* ini maka kumpulan dari modul-modul program tidak dieksekusi dalam memori yang sama namun terbagi dalam komputer *client*-server, dengan arsitektur seperti ini modul yang memanggil menjadi *client* yang meminta suatu layanan data dan modul yang dipanggil menjadi *server* yang menyediakan layanan data.

Hal ini menjadikan konfigurasi bagi komputer *client* dan komputer *server* bisa berbeda seperti kapasitas memori, kecepatan *processor* atau alat masukan dan keluaran yang disesuaikan dengan fungsi kerja dari elemen-elemen tersebut. Bagi *server* yang menjalankan tugas pengolahan suatu *database* digunakan suatu konfigurasi yang khusus menangani tugasnya tersebut dengan sistem operasi yang dikhususkan bagi *server* seperti Windows NT server dan Windows 2000 server, sedangkan komputer *client* menggunakan konfigurasi yang umum bagi sebuah komputer *desktop* yang terhubung ke jaringan dengan sistem operasi seperti Windows 98, Windows ME, Windows XP, dan yang lainnya.

Sistem *client-server* berjalan seutuhnya pada dua sistem yang berbeda. Biasanya satu buah *server* melayani beberapa komputer *client* atau ada juga yang hanya melayani satu *client* saja. *Client* maupun sebuah komputer *desktop* yang terhubung ke jaringan. Apabila pemakai ingin mengakses informasi bagian aplikasi, *client* mengeluarkan permintaan yang dikirimkan melalui jaringan kepada *server* (*upload*). *Server* kemudian menjalankan permintaan dan mengirimkan informasi kembali kepada *client* (*download*).

Proses *server* berperan sebagai aplikasi yang mengelola sumber daya nilai bersama (*shared resource*) seperti *database*, *pointer* atau jalur komunikasi selain menjalankan tugasnya sebagai *back-end*, *system client-server*, yaitu pusat pemrosesan data. Sedangkan proses *client* meliputi program-program untuk mengirimkan pesan permintaan *server* serta melakukan pengaksesan pada data seperti merubah, menghapus atau menambah data. Karena itu program pada *client* adalah aplikasi *front-end* yang digunakan sebagai antar muka (*interface*) bagi pemakai.



Gambar 2.7 Sistem *Client-Server*

(Sumber: Dede Sopandi, 2005, *Instalasi dan Konfigurasi Jaringan Komputer*, CV. Informatika, Bandung)

Tugas dari *Client* adalah :

1. Membuat koneksi *network* TCP (*Transfer Control Protocol*) dengan *server*.
2. Menerima *input* dari *user*.
3. Memformat kembali *input* dari *user* kemudian mengubahnya ke dalam bentuk format standar dan mengirimkannya pada *server*.
4. Menerima *output* dari *server* dalam format standar.
5. Mengubah format *output* tadi untuk ditampilkan pada layar.

Sedangkan tugas server adalah :

1. Menginformasikan software jaringan bahwa komputer itu siap menerima koneksi.
2. Menunggu permintaan dalam bentuk format standar.
3. Melaksanakan permintaan tersebut.
4. Mengirim kembali hasil ke client dalam bentuk format satandar.
5. Menunggu permintaan selanjutnya.

2.9 Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan adalah bahasa pemrograman Delphi beserta Database SQL Server 2000.

2.9.1 Sekilas Tentang Delphi

Delphi adalah bahasa pemrograman terstruktur yang dibuat dengan basis visual atau grafik windows, dan merupakan versi lanjutan dari turbo pascal. Delphi merupakan bahasa pemrograman yang mempunyai cakupan yang luas.

Delphi menyediakan fasilitas pemrograman yang dibagi dalam dua kelompok yaitu object dan bahasa pemrograman. Secara ringkas object adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat. Object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman dapat disebut sebagai kumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu.

Delphi adalah cara yang paling sederhana untuk membangun aplikasi berbasis windows dan produktifitas dari perangkat lunak yang terdiri dari 5 atribut penting, yaitu :

1. Kualitas dari lingkungan pengembangan visual.
2. Kecepatan kompiler dibandingkan dengan kompleksitasnya.
3. Kekuatan dari bahasa pemrograman dibandingkan dengan kompleksitasnya.
4. Fleksibilitas dari arsitektur basis data.
5. Pola desain dari pemakai yang dijadikan oleh *frame work*nya.

Khusus untuk pemrograman data base, delphi menyediakan object yang sangat kuat, canggih dan lengkap sehingga memudahkan dalam merancang, membuat, dan menyelesaikan aplikasi. Selain itu Delphi juga dapat menangani data dalam berbagai format *database*, misalnya format MS. Access, Oracle,

Foxpro, Informix, DB2, dan lain-lain. Format *database* yang dianggap asli dari Delphi adalah paradox dan dbase.

2.9.2 Microsoft SQL Server 2000

SQL Server 2000 mempunyai fungsi utama untuk memproses *query* dan transaksi *database* yang diberikan oleh aplikasi *client* dengan performansi yang baik.

Dalam Microsoft SQL Server 2000, database bukanlah sebuah file tetapi merupakan sebuah konsep logis yang berisi sekumpulan objek-objek yang berhubungan. Misalnya sebuah database berisi data, struktur *database*, index, sekuritas, *view*, dan *stored procedure*. Objek-objek dalam sebuah database :

1. Tabel. Objek yang berisi tipe-tipe data dan data mentah
2. Kolom. Sebuah tabel berisi kolom-kolom untuk menampung data. Kolom mempunyai sebuah tipe dan nama yang unik
3. Tipe data. Sebuah kolom mempunyai sebuah tipe data. Tipe-tipe yang dapat dipilih adalah karakter, numerik, tanggal, boolean, dan lain-lain.
4. *Stored Procedure*. Merupakan perintah-perintah SQL yang membentuk makro. Dengan menjalankan *stored procedure* berarti menjalankan perintah-perintah SQL di dalam sebuah *procedure.*
5. *Trigger*. *Stored Procedure* yang diaktifkan pada saat data ditambahkan, diubah, atau dihapus dari database. *Trigger* dipakai untuk menjamin aturan integritas di dalam database.
6. *Rule*. Diberlakukan pada kolom sehingga data yang dimasukkan harus sesuai dengan aturan.

7. *Primary Key*. Menjamin setiap baris data unik, dapat dibedakan dari data lain.
8. *Foreign Key*. Kolom-kolom yang mengacu *primary key* pada tabel lain. *Primary key* dan *foreign key* dipakai untuk menghubungkan sebuah tabel dengan tabel lain.
9. *Konstrain*. Mekanisme integritas data yang berbasis *server* dan diimplementasikan oleh sistem.
10. *Default*. Dinyatakan pada *field* (kolom) sehingga jika kolom tersebut tidak diisi data, maka diisi dengan nilai *default.*
11. *View*. *Query* yang memakai beberapa tabel, dan disimpan di dalam database. *View* dapat memilih beberapa kolom dari sebuah tabel atau menghubungkan beberapa tabel. *View* dapat dipakai untuk menjaga keamanan data.
12. Index. Membantu mengorganisasikan data sehinggga query menjadi lebih cepat.

2.10 Persediaan

Persediaan adalah sebagai suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual atau digunakan dalam suatu periode tertentu, atau persediaan barang-barang yang masih dalam pengerjaan/proses produksi, ataupun persediaan bahan baku yang menunggu dalam suatu proses produksi.

Menurut RAN[8] persediaan merupakan “bahan-bahan bagian-bagian yang disediakan dan bahan-bahan dalam proses yang terdapat dalam perusahaan untuk proses produksi, serta barang-barang jadi/produk yang disediakan untuk memenuhi permintaan dari konsumen atau langganan setiap waktu”.

2.10.1 Jenis-Jenis Persediaan

Menurut fungsi, persediaan dibagi menjadi tiga jenis, yaitu :

1. *Batch Stock/Lot Size Inventory*

   Persediaan yang diadakan karena membeli atau membuat bahan-bahan atau barang-barang dalam jumlah yang lebih besar dari jumlah yang dibutuhkan saat itu.

2. *Fluctuation Stock*

   Persediaan yang diadakan untuk menghadapi fluktuasi permintaan konsumen yang tidak dapat diramalkan.

3. *Anticipation Stock*

   Persediaaan yang diadakan untuk menghadapi fluktuasi permintaan yang dapat diramalkan, berdasarkan pola musiman yang terdapat dalam satu tahun untuk menghadapi penggunaan atau penjualan atau permintaan yang meningkat.

2.10.2 Fungsi-Fungsi Persediaan

Fungsi-fungsi persediaan adalah :

1. Fungsi *Decoupling*

   Fungsi *Decoupling* adalah persediaan yang memungkinkan perusahaan dapat memenuhi permintaan langganan tanpa tergantung pada *suplier*. Persediaan bahan mentah diadakan agar perusahaan tidak akan sepenuhnya tergantung pada pengadaannya dalam hal kualitas dan waktu pengiriman.

2. Fungsi *Economic Lot Sizing*

   Persediaan *Lot Size* ini mempertimbangkan penghematan-penghematan atau potongan pembelian, biaya pengangkutan per unit menjadi lebih murah dan sebagainya. Hal ini disebabkan karena perusahaan melakukan pembelian dalam kuantitas yang lebih besar dibandingkan dengan biaya-biaya yang timbul karena besarnya persediaan.

3. Fungsi Antisipasi

   Apabila perusahaan menghadapi *fluktuasi* permintaan yang diperkirakan dan diramalkan berdasarkan pengalaman atau data-data masa lalu, yaitu permintaan musiman. Dalam hal ini perusahaan dapat mengadakan persediaan musiman (*seasional inventories*).

Disamping itu, perusahaan juga sering menghadapi ketidakpastiaan jangka waktu pengiriman dan permintaan akan barang-barang selama periode tertentu. Dalam hal ini, perusahaan memerlukan persediaan ekstra yang disebut persediaan pengaman (*safety stock inventories*).