BAB II

LANDASAN TEORI

2.1.1 Pengertian Sistem

Sistem adalah kumpulan atau group dari sub sistem, bagian, komponen apapun baik phisik ataupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. (Azh02)

Sistem dikelompokan menjadi 2 (dua) macam, yaitu yang menekankan pada komponen-komponen atau elemen-elemen dan sistem yang menekankan pada prosedur. Sistem yang menekankan pada prosedur yang dijelaskan menurut (Jog99) yaitu sebagai berikut :

"S*istem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu"*.

Dengan melihat definisi *sistem* diatas maka dapat dikatakan bahwa *sistem* adalah suatu kerangka kerja terpadu yang terdiri dari prosedur-prosedur yang saling berhubungan dan bekerja sama untuk melakukan suatu kegiatan demi pencapaian suatu tujuan organisasi.

2.1.2 Karakteristik Sistem

Model umum sebuah sistem adalah input, proses dan output. Hal ini merupakan konsep sebuah sistem yang sangat sederhana. Sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran.

Selain itu pula sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

Adapun karakteristik yang dimaksud adalah sebagai berikut :

a. Komponen Sistem (*Components*)

   Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan.

b. Batasan Sistem (*Boundary*)

   Ruang lingkup sistem merupakan daerah yang membatasi antara sistem yang lainnya atau sistem dengan lingkungan luarnya.

c. Lingkungan Luar Sistem (*Environtment*)

   Bentuk apapun yang ada diluar lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut, disebut dengan lingkungan luar sistem.

d. Penghubung sistem (*Interface*)

   Sebagai media yang menghubungkan sistem dengan sub-sistem yang lainnya disebut penghubung sistem atau interface.

e. Masukan Sistem (*Input*)

   Energi yang dimasukkan ke dalam sistem sistem disebut masukan sistem, uang dapat berupa pemeliharaan (*maitenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem (*Output*)

   Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

g. Pengolah Sistem (*Proses*)

   Suatu sistem dapat mempunyai suatu proses yang akan merubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

   Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik.

2.2.1. Pengertian Sistem Informasi

Informasi adalah hasil pengolahan data yang memberilan arti dan manfaat, jadi informasi merupakan hasil dari pengolahan data, akan tetapi tidak semua hasil dari pengolahan tersebut bisa menjadi informasi, hasil pengolahan data yang tidak memberikan makna atau arti serta tidak bermanfaat bagi seseorang bukanlah merupakan informasi bagi orang tersebut (Azh02)

Menurut (Dav85) *"Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau yang akan dating"*.

2.2.2. Kualitas informasi

Kualitas informasi tergantung dari tiga hal yaitu :

1. Akurat

   Informasi harus bebas dari kesalahan dan tidak bisa atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

2. Tepat waktu

   Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan dalam mengambil keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi infomasi. mengirimkan.

3. Relevan

   Informasi tersebut mempunyai manfaat untuk pemakainya, relevansi informasi bagi tiap-tiap orang dengan yang lainnya.

2.2.3. Nilai Informasi

Nilai informasi ditentukan dari 2 (dua) hal yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Lebih lanjut sebagai besar informasi tidak dapat persis ditaksir keuntungannya dengan suatu nilai uang tetapi dapat ditaksir nilai efektifitasnya.

2.3. Sistem Informasi

Sistem informasi adalah kumpulan dari sub-sub sistem baik phisk maupun non phisik yang saling berhubungan atau sama dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna (Azh02).

Sistem informasi menurut Laudon merupakan komponen – komponen yang saling berhubungan dan bekerja sama untuk mengumpulkan, memproses, menyimpan dan menyebarkan informasi untuk mendukung pengambilan

keputusan, koordinasi, pengendalian dan untuk memberikan gambaran aktifitas didalam perusahaan. (Azh02).

Sistem informasi menurut Mc Keown merupakan gabungan dari computer dan user yang mengelola perubahan data menjadi informasi serta menyimpan data dan informasi tersebut. (Azh02).

Sistem informasi adalah sekumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu. Dalam sistem informasi diperlukannya klasifikasi alur informasi, hal ini disebabkan keanekaragaman kebutuhan akan suatu informasi oleh pengguna informasi. Kriteria dari sistem informasi antara lain, fleksibel, efektif dan efisien.

("http://id.wikipedia.org/wiki/Sistem_Informasi)

2.3.1 Konsep Dasar Sistem Informasi

Informasi sangat dibutuhkan oleh kalangan manajer untuk mengambil keputusan, keputusan ini berguna untuk memecahkan suatu masalah. Dalam pemecahan masalah tersebut manajer memerlukan suatu sistem informasi yang tepat dan akurat.

(Dav85) melalui sistem informasi manajemen mendefinisikan **:** *"Sebuah sistem mesin manusia atau mesin yang terpadu (integrated), untuk menyajikan informasi guna mendukung fungsi operasi, manajemen, dan pengambilan keputusan dalam sebuah organisasi*

2.3.2 Komponen Sistem Informasi

Untuk membentuk suatu sistem informasi ada beberapa komponen yang harus dimiliki antara lain sebagai berikut :

1. Perangkat keras (*hardware*)

   Perangkat keras (hardware) adalah komponen fisik berupa peralatan input, peralatan proses dan peralatan output.

2. Perangkat lunak (*software*)

   Perangkat lunak (*software*) adalah instruksi-instruksi yang membuat komputer sebagai perangkat keras melakukan pekerjaan tertentu.

3. Sumber daya manusia (*brainware*)

   Sumber daya manusi (*brainware*) adalah user atau pengoperasi sistem

4. Data

   Data adalah fakta-fakta, perkiraan-perkiraan, pendapat-pendapat yang belum memiliki arti kegunaan

5. Prosedur

   Prosedur adalah instruksi-instruksi yang digunakan dalam mengoperasikan sistem.

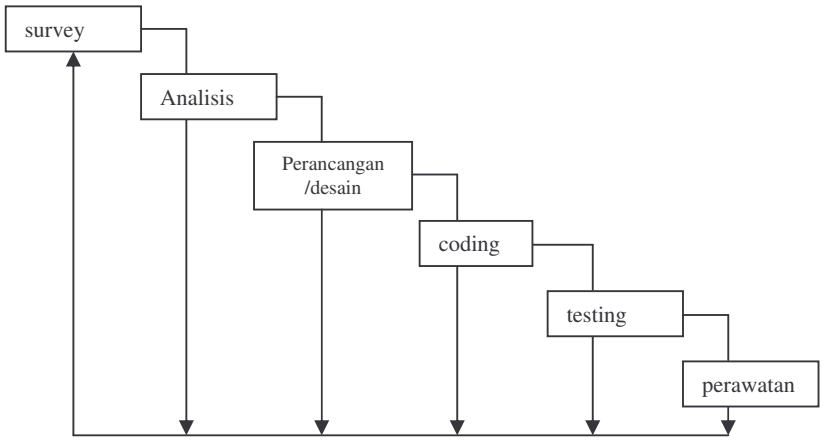
2.4 Metodologi Pengembangan Sistem

Metodologi pengembangan sistem yang di buat dalam sistem ini adalah metodologi pengembangan Waterffall.

2.4.1. Metode Waterfall

Metode Waterfall hanyalah salah satu dari ratusan metode pengembangan sistem informasi yang ada di dunia saat ini dan yang paling populer pada saat sistem informasi pertama kali berkembang. Karena tahapan-tahapan tersebut mengikuti suatu pola yang teratur dan dilakukan secara *top-down,* metode ini sering disebut sebagai pendekatan air terjun (*waterfall approach*).

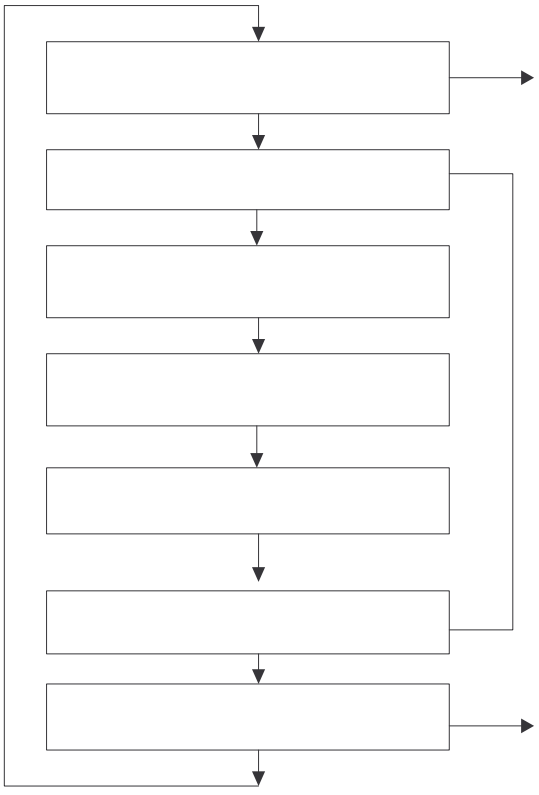
Bagi seorang analis profesional yang bergerak dalam bidang pengembangan sistem informasi, maka mengetahui berbagai metode yang berkembang seputar pengembangan sistem informasi merupakan hal yang sangat penting, agar dapat memilih metode mana. Pada Gambar dibawah menunjukkan aktivitas-aktivitas/tahapan-tahapan yang dilakukan pada metode pengembangan *Waterfall*.



Gambar 2.1 *Tahapan perancangan metode waterfall*
Sumber: (Jog95)

2.4.2. Metode SDLC (*System Development Lifecycle*)

Menurut Jogiyanto Pengembangan sistem (*System Development Lifecycle*) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Tahapan utama siklus hidup pengembangan sistem terdiri dari perencanaan sistem *(Systems Planning)*, Analisis sistem *(Systems Analysis)*, Desain Sistem *(Systems Design)*, Seleksi sistem *(Systems Selection),* Implementasi Sistem *(Systems Implementation)* dan Perawatan Sistem *(Systems Maintenance).*



Gambar 2.2 *Tahapan perancangan metode SDLC*

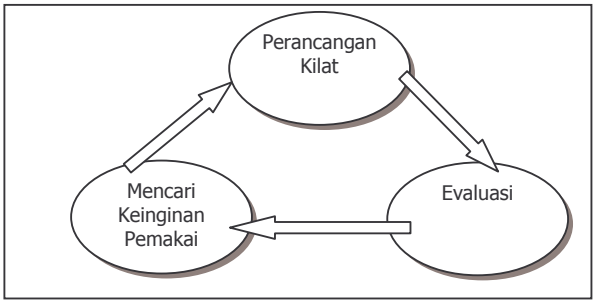
Sumber: (Jog95)

Kebijakan dan Perencanaan Si

Analisis Sistem

2.4.3. Metode Prototyping

Metode *protyping* yaitu metode pengembangan sistem yang menggunakan pendekatan untuk membuat program dengan cepat dan bertahap sehingga dapat dievaluasi oleh pemakai.



Gambar 2.3 *Tahapan perancangan metode Prototyping*
Sumber: (Jog95)

Prototyping dimulai dengan pengumpulan kebutuhan, pengembang dan pelanggan bertemu dan mendefinisikan obyektif keseluruhan dari perangkat lunak, mengidentifikasikan segala kebutuhan yang diketahui, dan area garis besar dengan definisi yang jelas merupakan suatu keharusan, kemudian dilakukan 'perancangan kilat'. Perancangan kilat membawa kepada konstruksi sebuah prototipe. Prototipe tersebut dievaluasi dan dipakai untuk membagi kebutuhan pengembangan perangkat lunak.

Iterasi terjadi pada saat *prototipe* disetel untuk memenuhi kebutuhan pelanggan, dan pada saat yang sama memungkinkan pengembang untuk secara lebih baik memahami apa yang harus dilakukan.

Secara ideal *prototipe* berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila *prototipe* yang sedang bekerja dibangun pengembang harus mempergunakan fragmen-fragmen program yang ada yang memungkinkan program yang bekerja untuk dimunculkan secara cepat

2.5 Metode Analisis dan Perancangan Terstruktur

Salah satu pendekatan yang digunakan dalam suatu analisis dan desain adalah pendekatan terstruktur. Suatu pendekatan yang bekerja dari sudut pandang yang lebih tinggi menuju tingkat lebih rendah yang lebih rinci, dimana keinginan pemakai disajikan dalam diagram aliran data. Desain terstruktur adalah implementasi secara fisik dan pembagian struktur modular secara hirarki dengan pendekatan atas bawah. Beberapa alat bantu yang digunakan dalam analisis dan desain tersetruktur antara lain

2.5.1. Diagram konteks (*Context Diagram)*

Merupakan diagram tingkat atas, yaitu diagram yang paling tidak detil dari sebuah sistem informasi yang menggambarkan aliran – aliran data kedalam dan keluar entitas-entitas ekternal. Entitas-entitaseksternal adalah entitas yang terletak diluar sistem yang mengirim data atau yang menerima data dari sistem tersebut.

2.5.2 Diagram aliran data (*Data Flow Diagram*)

Diagram aliran data adalah salah satu alat bantu utama yang digunakan oleh sistem analist. Diagram aliran data sebagai alat Bantu pemodelan dipopulerkan oleh DeMarco (1978) dan Gane dan Sarson (1979) melalui metodologi sistem terstruktur mereka. Diagram aliran data sering digunakan untuk menggambarkan

suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa pertimbangan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan.

2.5.3. Kamus Data (*Data Dictionaries)*

Kamus Data adalah katalog fakta tentang data dan kebutuhankebutuhan informasi dari sustu sistem informasi. Dengan menggunakan kamus data, analis sistem dapat mendefinisikan data disistem dengan lengkap. Pada tahap analisis, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir disistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Pada tahap perancangan digunakan untuk merancang input dan merancang laporan-laporan dan database.

2.5.4 Normalisasi

Normalisasi merupakan sebuah tehnik logical desain basis data/database, tehnik pengelompokan atribut suatu relasi sehingga membentuk struktur relasi yang baik (tanpa redudansi).

Bentuk-bentuk normalisasi adalah :

a. Bentuk normalisasi tahap pertama

Bentuk ini dapat dilihat dari ciri-ciri yaitu satu record demi satu record dalam flat file (file data atau rata), data dibentuk dalam satu record demi satu record dan nilai value fieldnya berupa *atomic value*, tidak hanya set atribut yang berulang-ulang.

Aturan :

a. Mendefinisikan atribut kunci
b. Tidak adanya group berulang
c. Semua atribut bukan kunci tergantung pada atribut kunci

Contoh:

Barang (Kode_ cucian, Nama_ cucian, Jenis_cucian)

Penjualan (No_Nota, Kode_ Cucian)

Tabel 2.1 Contoh bentuk normalisasi tahap pertama

Kode_cucian	No_Nota	Nama_Cucian	Jenis_Cucian
01.001	001	Jeans	panjang
02.002	002	Kemeja	panjang

b. Bentuk normalisasi tahap kedua

Bentuk normalisasi tahap kedua mempunyai syarat, yaitu bentuk data telah memenuhi kriteria bentuk normal satu.

Aturan :

a. Sudah memnuhi dalam bentuk normal kesatu
b. Sudah tidak ada ketergantungan parsial, dimana seluruh *field* hanya tergantung pada sebagian *field* kunci.

Contoh:

Relasi Penjualan

Tabel 2.2 Contoh bentuk normalisasi tahap kedua

Kode_Cucian	Nama_Cucian	Jenis_Cucian
01.001	Jeans	Kemeja

Relasi Barang

Tabel 2.3 Contoh bentuk normalisasi tahap ketiga

Kode_Cucian	Nama_Cucian
01.001	Jeans

c. Bentuk normalisasi tahap ketiga

Untuk menjadi bentuk normalisasi bentuk ketiga haruslah dalam bentuk normal kedua dan semua atribut bukan primer tidak hanya hubungan transitif, contoh pada tahap kedua sudah menjadi bentuk normal ketiga karena semua atribut tergantung pada *primary key*.

Aturan :

a. Sudah berada dalam bentuk normal kedua
b. Tidak ada ketergantungan transitif (dimana *field* bukan kunci tergantung pada *field* bukan kunci lainnya).

2.5.5. Relasi Tabel (Entinity Relation Tabel)

ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antara penyimpanan ERD digunkan unutk memodelkan struktur data dan hubungan antar data, karena hal ini relative kompleks. Dengan ERD kita dapat menguji model dengan mengabaikan proses yang harus dilakukan. Dan dengan ERD kita mencoba menjawab pertanyaan Seperti : data apa yang kita perlukan? Bagaimana data yang satu berhubungan dengan yang lain?

1. Entity:adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan

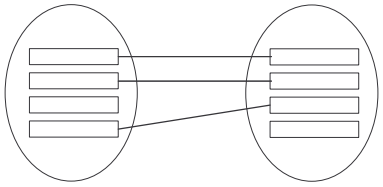
dibuat. Sebagai contoh pelanggan, pekerja dan lain-lain. Seandainya A adalah seorang pekerja maka A adalah isi dari pekerja, sedangkan B adalah seorag pelaggan maka B adalah isi dari pelanggan. Karena itu harus dibedakan antara entity sebagai bentuk umum darideskripsi tertentu dan isi entity seperti AdanB

2. Atribut : Entiti mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter entity. Misalnya atribut nama pekerja dari entity pekerja, setiap ERD bisa terdapat lebih dari satu atribut.
3. Hubungan : *Relationship,* Sebagaimana halnya entity maka dalam hubungan pun harus dibedakan antara hubungan atau bentuk hubungan antar entity dengan isi dari hubungan itu sendiri. Misalnya dalam kasus hubungan antara entity siswa dan entity mata_kuliah adalah mengikuti, sedangkan isi hubungannya dapat berupa nilai_ujian.

Jumlah maksimum entitas yang terjadi diantara dua himpunan entitas (misalnya A dan B) dapat berupa:

1. Satu ke Satu *(One to One),*

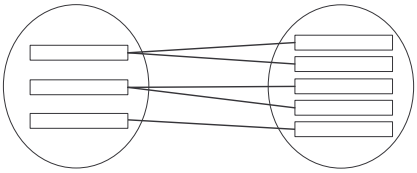
Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A



Gambar 2.4 *One to one Relationship 2 file*
Sumber : (Fat99)

2. Satu ke Banyak *(One to Many)*

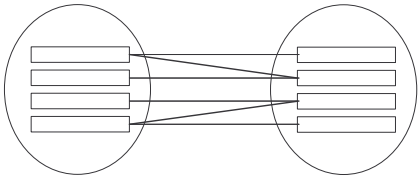
Yang berarti setiap entitas pada himpunan entitas dapat berhubungan dengan banyak entitas pada himpunan entitas B tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A



Gambar 2.5 *One to Many Relationship 2 file*
Sumber : (Fat99)

3. Many to many Relationship 2 file

Hubungan *many to many relationshi*p adalah sebuah relasi antara suatu *file* dengan *file* yang keduanya mempunyai relasi banyak berbanding banyak.



Gambar 2.6 *Many to many Relationship 2 file*
Sumber : (Fat99)

2.6 Perancangan 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 memanipulasi. *Database* merupakan salah satu komponen yang paling dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai. Penerapan *database* dalam sistem informasi disebut dengan *Database system*. Sistem basis data, adalah suatu

sistem informasi yang mengintergrasikan dari kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuat tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

a. Field

Suatu field menggambarkan suatu atribut dari *record* yang menunjukan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya. Kumpulan dar *field* membentuk suatu *record*. Ada tiga hal yang penting dari suatu *field*, yaitu :

1. Nama dari *field (field name)*

   Field harus diberi nama untuk membedakan *field* yang satu dengan *field* lain

2. Representasi dari *field (field represention)*

   Representasi dari *field* menunjukan tipe dari *field (field type)* serta lebar dari *field (field width)*. Filed dapat bertipe numerik atau huruf. Lebar dari *field* menunjukan ruang maksimum dari *field* yang dapat diisi dengan karakter-karakter data.

3. Nilai dari *field (field value)*

   Nilai dari *field* menunjukan isi dari *fiel*d untuk masing-masing *record*

b. Record

Kumpulan dari *field* membentuk suatu *record. Record* menggambarkan suatu unit data individu yang tertentu. Kumpulan dari *record* membentuk suatu file.

c. File

File terdiri dari record-record yang menggambarkan satu kesatuan data yang sejenis.

d. Database

Kumpulan dari file membentuk suatu *database.*

Penyusunan suatu *database* digunakan untuk mengatasi masalah-masalah penyusunan data, yaitu :

a. Kesulitan pengaksesan

b. Banyak pemakai *(Multiple user)*

c.. Masalah keamanan *(Secuity)*

d. Masalah kesatuan *(Intergration)*

e. Masalah kebebasan data *(Independence)*

Sistem Manajemen Database *(Database Management Sistem/DBMS)* adalah satu rangkaian program-program yang mengelola sebuah database dan menyediakan mekanisme-mekanisme melalui jenisjenis data dapat disimpan, dicari kembali, dan diubah. Terdapat banyak jenis DBMS yang membuat paket perangkat lunak ini sebagai komponen yang mempunyai nilai khusus pada sistem informasi. Dengan mengetahui karakteristiknya kita dapat mengetahui bagaimana DBMS berkerja dan digunakan sebagai berikut :

1. Pemakaian Bersama

   Dengan menggunakan sistem manajemen database setiap unit dalam organisasi dapat mengunakan data yang ada dalam database secara bersama-sama.

2. Hubungan antar Data

   Sering terdapat beberapa hubungan antara beberapa item data untuk keperluan tertentu, misalnya sesorang akan memproses informasi penting, DBMS harus sanggup membuat hubungan antara beberapa item data dan menggunakan hubungan tersebut bila pemakai meminta jenis data tertentu.

3. Redudancy

   Dengan database diharapkan tidak terjadi kelebihan (redundancy) atau duplikasi penyimpanan data yang sama dalam satu organisasi. Redundancy adalah penyimpanan item data yang sama lebih dari satu lokasi fisik. Umunya suatu data tertentu hanya disimpan pada satu file tetapi dapat dihubungkan dengan data pada lain file.

4. Independensi antara program dan data

   Dalam menggunakan DBMS, bila terjadi modifikasi database tidak berarti harus melakukan perubahan terhadap semua program, namun yang perlu diubah adalah filenya saja.

5. Pembuatan Laporan

   Kemampuan penting dalam DBMS adalah memproduksi laporan yang rumit, karena dapat memasukan data dari beberapa file yang berkaitan.

6. Akses bahasa pertanyaan

   Suatu sistem pertanyaan menyediakan untuk menghadapi permintaan tidak terduga.

7. Pendekatan Biasa dan Pengawasan

   Karakteristik lain dari DBMS adalah mendorong pendekatan biasa dan pengawasan pada pengembangan serta penggunaan aplikasi. Prosedur yang benar harus diikuti bilamana membuat, mengakses, meng-update atau memelihara suatu database.

8. Membuat Aplikasi

   Sebagai tambahan pada keuntungan yang sudah ada, suatu aplikasi dapat dibuat dengan cepat dengan menggunakan DBMS. Tanpa DBMS pendesain

harus memasukan semua detail kedalam proses penyimpanan dan pencarian kembali, yang sebenarnya dapat ditangani oleh DBMS secara otomatis.

2.6.1. Struktur Database

Ada dua kategori dasar hubungan-hubungan yang diperlukan untuk menerapkan kemungkinan jenis hubungan, antara lain :

1. Struktur pohon

   Karakteristik pohon adalah hubungan-hubungan "satu kebanyak". Sebuah pohon terdiri dari elemen-elemen hirarki berupa cabang (tempat cabang tumbuh), satu cabang diidentikan sebagai catatan. Contoh dari struktur pohon adalah hubungan antara file orang tua dan file anak.

2. Struktur jaringan

   Pada struktur pohon cabang yang lebih rendah hanya mempunyai satu cabang diatasnya, tetapi dalam struktur jaringan cabang yang lebih rendah bisa mempunyai banyak cabang diatasnya.

2.6.2. Jenis Database

Adapun jenis database terdiri dari tiga jenis, yaitu :

1. Database Hirarki

   Sistem manajemen database hirarki menyediakan data yang dapat ditampilkan dengan struktur pohon.

2. Database Jaringan

   Sistem manajemen database jaringan dapat mempersiapkan struktur jaringan,

3. Database Relasional

Sistem manajemen database relasional lebih fleksibel disbanding dengan kedua DBMS hirarki dan jaringan, karena database relasional mudah didesain dan digunakan. Kebanyakan suatu DBMS relasional tidak mensyaratkan pendesain untuk mengspesifikasikan bentuk hubungan-hubungan ketika database mulai dibuat. Hubungan-hubungan tersebut baru perlu dispesifikasikan bila database digunakan untuk mendapatkan informasi atau untuk keperluan pembuatan laporan.

2.7 Jaringan

Jaringan merupakan sistem syaraf dari operasi perusahaan, manajemen jaringan yang ditujukan untuk mengurangi peluang kegagalan jaringan terutama melalui perencanaan dan pengendalian.

a. Perencanaan Jaringan

Perencanaan jaringan terdiri dari semua yang diarahkan untuk mengantisipasi kebutuhan jaringan perusahaan. Tiga komponen utamanya adalah :

1. Perencanaan kapasitas menganalisis dan merencanakan *volume* lalu lintas yang bisa ditangani jaringan.
2. Perencanaan staff membantu menentukan jumlah orang yang diperlukan untuk mengelola suatu jaringan dan tingkat keahlian yang diperlukan.
3. Pemantauan kinerja menganalisa waktu respons pada suatu tingkat lalu lintas tertentu untuk mengantisipasi efek dari kemungkinan perubahan kinerja jaringan.

b. Pengendalian Jaringan

Pengendalian jaringan meliputi pemantauan jaringan setiap hari untuk memastikan bahwa jaringan tetap pada tingkat operasi yang diinginkan. Pengendalian jaringan meliputi prosedur – prosedur seperti deteksi kegagalan, isolasi kegagalan, dan pemulihan jaringan. Sistem pengendalian jaringan yang ideal memberi peringatan kepada operator mengenai kegagalan atau potensi kegagalan jaringan mengidentifikasikan komponen – komponen yang bermasalah, dan membantu memperbaiki masalah.

Untuk mencapai dan memelihara pengandalian jaringan, perusahaan memerlukan standar operasi, strategi rancangan, prosedur pengendalian kegagalan, dan metodologi pemecahan masalah. Elemen ini harus diperbaharui sesuai yang terjadi dalam teknologi komputer, jaringan atau organisasi.

2.7.1 Jenis-Jenis Jaringan Komputer

1 Local Area Network (LAN)

Jaringan LAN adalah jaringan yang menghubungkan beberapa komputer dalam suatu lokal area (biasanya dalam satu gedung atau antar gedung).

2 Metropolitan Area Network (MAN)

MAN merupakan jaringan yang menghubungkan beberapa jaringan komputer dalam wilayah yang lebih luas, biasanya digunakan oleh sebuah korporasi, jaringan komputer dalam satu kota.

3 Wide Area Network (WAN)

Jaringan WAN menghubungkan beberapa WAN dari beberapa kota atau Negara yang berbeda. WAN biasanya terhubung *via* satelit.

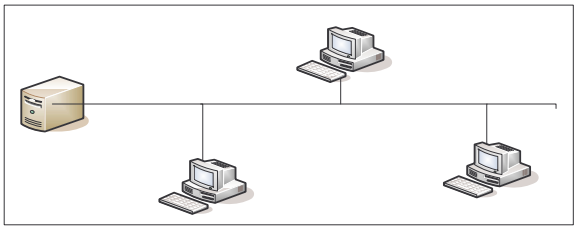
4 Jaringan Tanpa Kabel *(Wireless)*

Jaringan tanpa kabel mempunyai berbagai manfaat,yang telah umum dikenal adalah kantor *portable*.

2.7.2. Topologi Jaringan Komputer

Topologi Jaringan adalah gambaran secara fisik dari pola hubungan antara komponen-komponen jaringan, yang meliputi *server, workstation, hub* dan pengkabelannnya. Terdapat tiga macam topologi jaringan umum digunakan, yaitu Bus, Star dan Ring.

1 Topologi Bus juga dikenal sebagai *Linear Bus*, bentuk ini menghubungkan beberapa *node* dalam jalur data dan masing-masing *node* dapat melakukan tugas-tugas operasi yang berbeda. Seperti hanya *loop network*, tidak hanya *central node* dan semua node mempunyai status yang sama. Keunggulan topologi Bus adalah pengembangan jaringan atau penambahan workstation baru dapat dilakukan dengan mudah tanpa mengganggu *workstation* lain. Kelemahan dari topologi ini adalah bila terdapat gangguan di sepanjang kabel pusat maka keseluruhan jaringan akan mengalami gangguan.

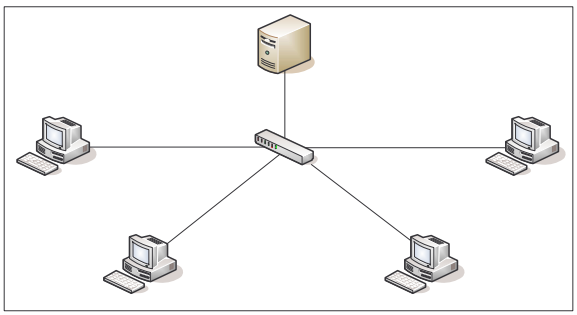


Gambar 2.7 *Topologi Bus*

Sumber : (Jak99)

2 Topologi Star yaitu beberapa *node* dihubungkan dengan suatu *node* pusat *(central node atau host node)* yang membentuk jaringan seperti bentuk bintang *(star)*. Semua komunikasi ditangani dan diatur langsung oleh *central node*. *Central node* melakukan semua tanggung jawab untuk mengatur arus informasi diantara *node* yang lainnya. Jika *node* yang satu ingin berkomunikasi dengan *node* yang lain, maka harus melewati *central node*. *Central node* biasanya berupa komputer besar *(large computer)* yang dihubungkan dengan node lainnya yang berupa beberapa terminal atau komputer mikro melalui suatu *link.*

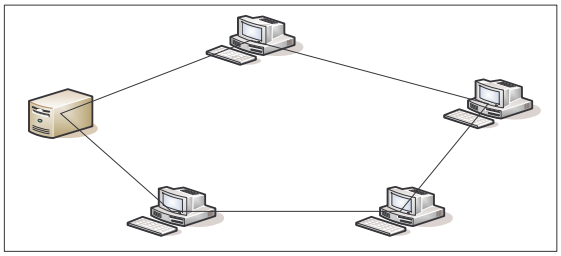
Keuntungan dari topologi ini paling fleksibel, control pusat, tidak menggangu jaringan lain serta control pusat. Akan tetapi memiliki kelemahan juga yaitu boros kabel dan perlu penangan khusus.



Gambar 2.8 Topologi Star
Sumber : (Jak99)

3 Topologi Ring merupakan gabungan bentuk loop network dan bus network. Jika salah satu node tidak berfungsi atau rusak, maka tidak akan mempengaruhi komunikasi node yang lain karena terpisah dari jalur data. Hal ini berbeda dengan loop network, bila salah satu node rusak, maka akan mempengaruhi node yang lainnya. Keunggulan topologi Ring adalah

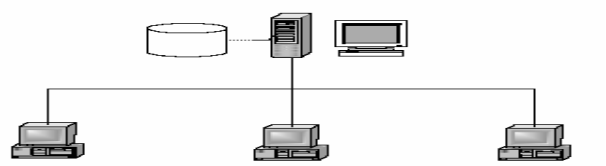
tidak terjadinya *collision* atau tabrakan pengiriman data seperti pada topologi Bus, karena hanya satu node dapat mengirimkan data suatu saat.



Gambar 2.9 *Topologi Ring*
Sumber : (Jak99)

2.8 Sistem *Client – Server*

Client berfungsi untuk menjalankan aplikasi basis data, dimana *client* akan menangani setiap aktivitas yang dikehendaki para pemakai dan mengupayakan agar semua proses 'sebisa mungkin' ditangani sendiri. Jika ada proses yang harus melibatkan data yang tersimpan pada basis data, barulah *client* melakukan 'kontak' dengan *server*. *Server* merupakan terminal yang menjalankan DBMS dan berisi basis data sebagai tempat *user* berinteraksi dengan sistem.



Gambar 2.10 *Sistem Client - Server Sederhana*

Sumber : (Jak99)

2.8.1 Cara Kerja Sistem Client Server

Biasanya sebuah server melayani beberapa komputer client, walaupun ada juga yang hanya melayani satu *client* saja. *Client* merupakan sebuah komputer desktop yang terhubung ke jaringan. Apabila pemakai ingin memakai informasi, bagian aplikasi client mengeluarkan permintaan yang dikirim melalui jaringan

kepada server. *Server* kemudian menjalankan permintaan dan mengirim informasi ke komputer *client*.

Pada proses server berperan sebagai aplikasi yang mengelola sumber daya milik bersama seperti *database*, *printer,* atau jalur komunikasi selain menjalankan tugasnya sebagai *back end*. Sistem client atau server yaitu pusat pemrosesan data. Sedangkan proses client meliputi program – program untuk mengirimkan pesan permintaan pada server serta melakukan pengaksesan pada data seperti mengedit, menghapus atau menambah data. Karena itu program pada client adalah aplikasi *front end* yang digunakan sebagai antar muka *interface* bagi pemakai untuk berinteraksi dengan server. Selain itu client juga melayani pemakaian sumber data lokal seperti *monitor, keyboard,* dan perangkat lokal lainnya.

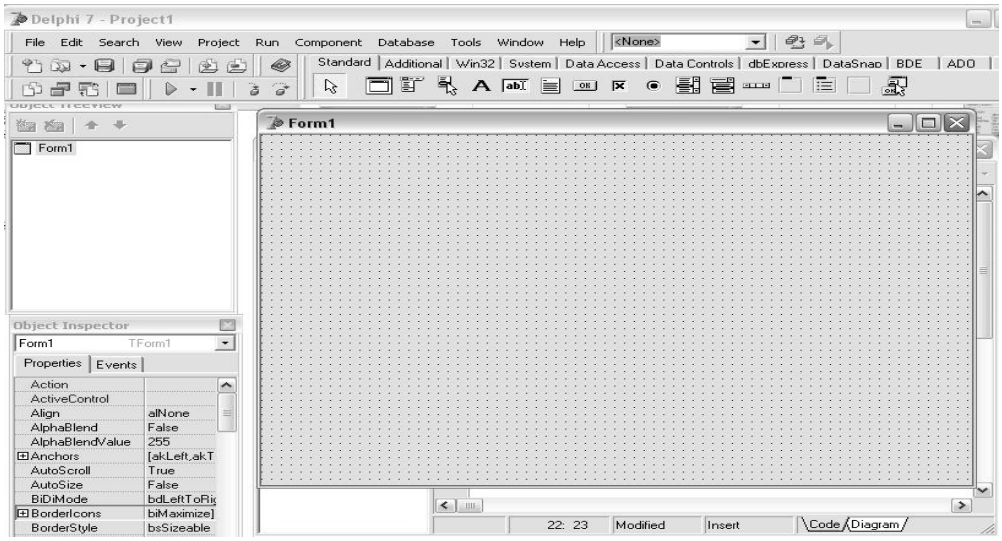
Sistem client server bekerja yaitu dengan adanya permintaan dari komputer client dan hasil dari permintaan yang dikirim adalah bagaimana membuat sistem *database* pada komputer server dengan aplikasi *database* nya mengembangkan aplikasi sebagai antarmuka bagi pemakai pada komputer client bahasa pemograman yang dipilih menghubungkan aplikasi tersebut dengan *database* jarak jauh pada server.

2.9 Perangkat Lunak Pendukung

Perangkat lunak pendukung didalam sistem ini menggunakan Borland Delphi 7.0 dan Database yang di gunakan yaitu SQL Server

2.9.1 Sekilas Tentang Borland Delphi 7.0

Menurut (Kad04) adalah merupakan sebuah piranti pengembangan aplikasi berbasis windows yang dikeluarkan oleh Borland International. Perangkat lunak ini sangat terkenal dilingkungan pengembang aplikasi karena mudah untuk dipelajari dan dapat digunakan untuk menangani berbagai hal, dari aplikasi matematika, permainan, hingga database. Pada penanganan database, Delphi menyediakan fasiitas yang memungkinkan pemrogram dapat berinteraksi dengan database seperti dBase, Paradox, Oracle, SQL Server dan Access.Menyediakan komponen-komponen dan bahasa pemrograman yang andal, sehingga memungkinkan penggunanya untuk membuat program apikasi sesuai dengan keinginan, dengan tampilan dan kemampuan yang canggih.



Gambar 2.11 *Tampilan Utama Delphi 7*

Seperti layaknya software *visual* lain Delphi juga memiliki IDE (*Integreted Development Environment*) atau lingkungan pengembangan tersendiri.

IDE Delphi menyediakan berbagai jendela yang akan sering digunakan dalam pengembangan aplikasi antara lain yaitu :

Menu bar

Menu bar atau menu utama digunakan untuk memilih operasi-operasi tertentu seperti menyimpan project, membuka project, dan lain-lain.



Gambar 2.12 *Menu Bar*

1. *Toolbar*

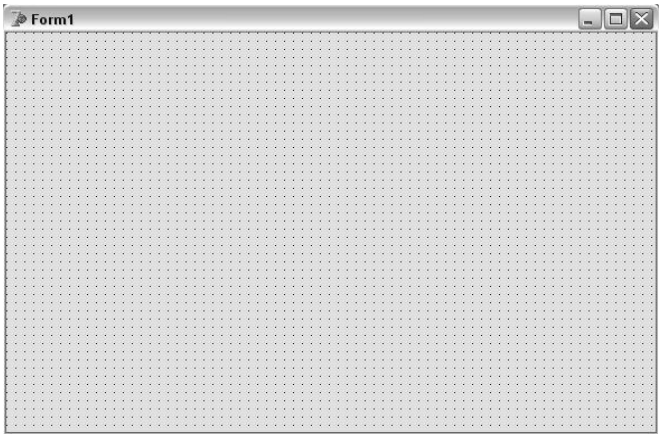
   Fasilitas ini dapat mempercepat pengaksesan perintah-perintah dalam pemrograman, dengan mengklik tombol-tombol dalam toolbar ini untuk melakukan aksi tertentu.



Gambar 2.13 *Toolbar*

2. *Form Designer*

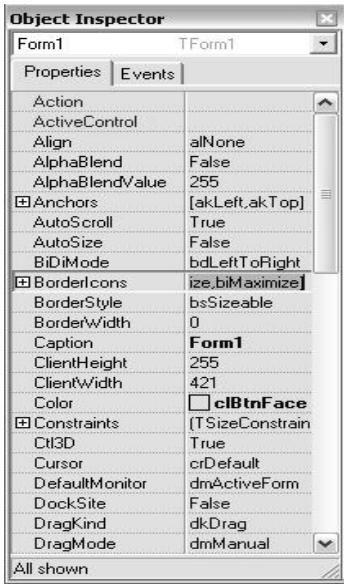
   Suatu form atau tampilan yang dapat diisi dengan desain tampilan database program yang akn dibuat.



Gambar 2.14 *Form Designer*

3. Object Inspector

Object Inspector terdiri dari dua bagian, yaitu *Properties* dan *Events*. Pada Properties bisa mengatur berbagai *property* dari *object* atau komponen yang akan digunakan.



Gambar 2.15 *Object Inspector*

4. *Component Pallete*

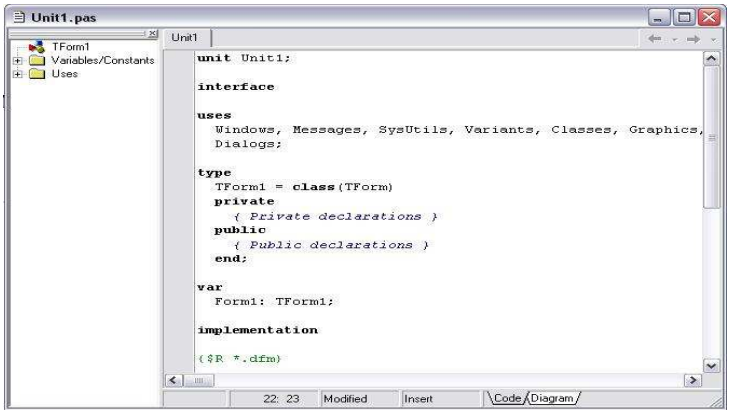
Component palette merupakan bagian yang digunakan untuk meletakkan berbagai komponen yang digunakan untuk aplikasi yang menggunakan *Borland Database Engine* (BDE) akan diletakkan pada page BDE, atau komponen yang digunakan untuk aplikasi yang membutuhkan *Interbase* akan diletakkan pada *Page Interbase* dan *Interbase Admin*.



Gambar 2.16 *Component Pallete*

5. *Code Editor*

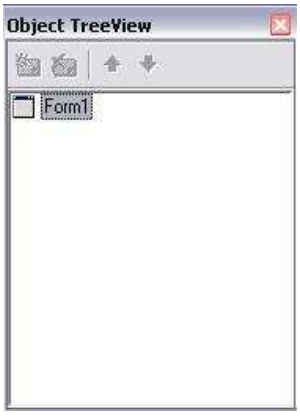
Pada *Code editor* ini dapat dituliskan kode dan memasukkan semua unit yang akan dipakai untuk program yang dibuat.



Gambar 2.17 *Code Editor*

6. *Object TreeView*

Merupakan sebuah diagram pohon yang menggambarkan hubungan logis antara komponen visual dan non visual yang terletak pada form, dan module atau frame.



Gambar 2.18 *Object TreeView*

2.9.2 Sekilas Tentang SQL Server

Menurut (Mar00) SQL Server merupakan salah satu dari sejumlah bahasa pemograman database (DBMS) yang bersaing merebut popularitas bersama-sama dengan database, FoxPro, FoxBase, dan sebagainya. SQL Server kini mulai menjauhkan diri dan melangkah jauh lebih kedepan, terutama dengan munculnya versi SQL Server 2000.

Menentukan bahasa nama yang terbaik untuk aplikasi database akan bersifat sangat subyektif. Namun, biasanya dukungan akan bahasa SQL (structure query language) karena dalam kecepata, pemakaian memori, mudak tidaknya program dan daya tampung data menjadi kriteria utama.

SQL Server mampu menampung database cukup besar dan dukungannya terhadap bahasa SQL, SQL Server juga memberikan dukungan terhadap sistem database terdistribusi.