BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Pendefinisian 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 akan dijelaskan menurut **[1] JOG** 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"*.

Sedangkan pendekatan yang lebih menekankan pada elemen atau komponennya adalah:

> *"Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tersebut"*.

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.

2.1.1 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. Pengertian Informasi

Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian *(event)* yang nyata *(fact)* yang digunakan untuk mengambil keputusan.

[2] GOR mendefinisikan informasi sebagai berikut :

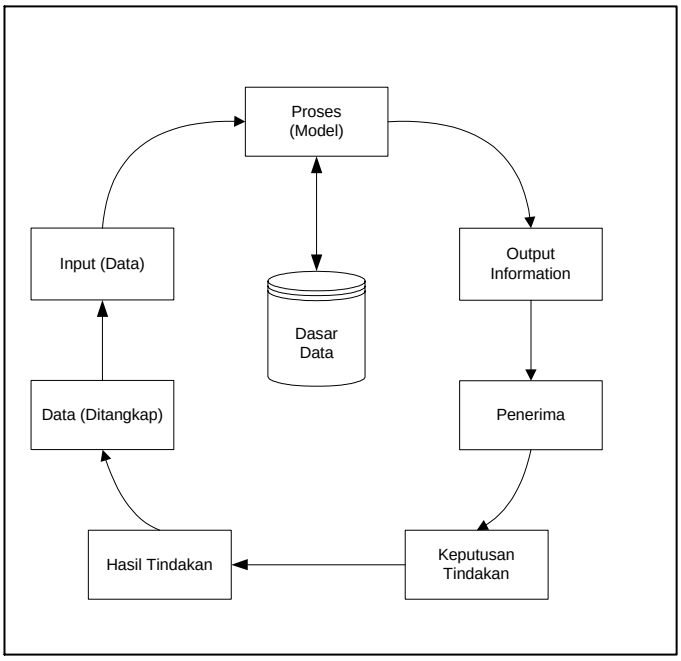
"Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau yang akan dating".

Informasi merupakan keluaran *(output)* dari proses data Output ini harus sudah tersusun degan baik dan mempunyai arti bagi yang menerimanya sehingga fungsi ini dapat dirasakan secara utuh dan relevan guna mencapai suatu keputusan yang inginkan.

2.2.1 Siklus Informasi

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

menghasilkan informasi. Informasi tersebut akan diterima penerima informasi untuk membuat keputusan dan melakukan tindakan.



Gambar 2.1 Siklus Informasi

Sumber :[6] Tagor M. Simatupang, 1994. *Teori Sistem*,

2.2.2 Kualitas informasi

Kualitas informasi tergantung dari tiga hal yaitu :

1. Akurat

Informasi harus bebas dari kesalahan-kesalahan dan tidak bias 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. Pengertian Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, tehnologi, media, prosedur-prosedur dan pengendalian yang ditunjukan untuk mendapatkan jalur komunikasi penting, memproses tipe transakasi rutin tertentu, memberi sinyal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan yang cerdik.

2.3.1 Konsep Dasar Sistem Informasi

Informasi sangat dibutuhkan oleh kalangan manajer untuk mengambil keputusan, keputusan ini berguna untuk memecahkan suatu masalah. Dalampemecahan masalah tersebut manajer memerlukan suatu sistem

informasi yang tepat dan akurat. **[2] GOR** 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 Sistem Informasi Keuangan

Sistem informasi keuangan pada SMUN 1 Ciamis adalah suatu system yang membantu dalam pengelolaan pencatatan sumber keuangan yang didapatkan dengan jumlah uang yang dikeluarkan oleh sekolah, dengan tujuan dianhtaranya:

1) Membantu staff pegawai Tata Usaha (TU) dalam mengelola dan mencatat setiap kegiatan yang berhubungan dengan penerimaan dan pengeluaran keuangan yang terjadi di sekolah menjadi lebih mudah dan cepat
2) Diharapkan dengan adanya Sistem Informasi Keuangan sekolah ini akan meminimalkan kesalahn yang ada.
3) Membuat laporan keuangan sekolah menjadi lebih cepat dan tepat, sehingga akan mengasilkan informasi yang benar dan dapat dipertanggungjawabkan.

2.5. Metode Pengembangan Sistem

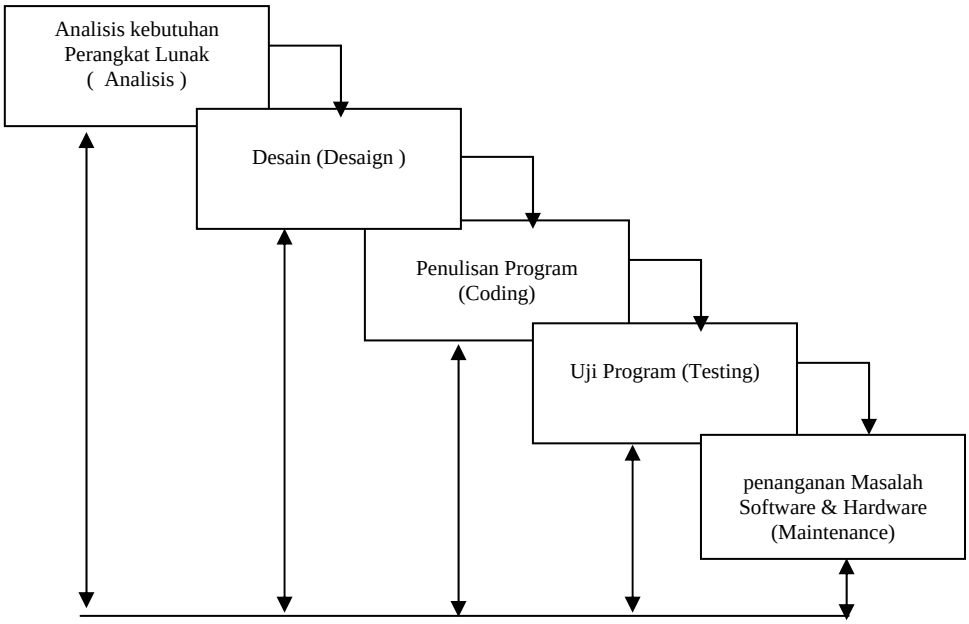
Metode pengembangan sistem merupakan aturan yang digunakan untuk mengembangkan suatu sistem informasi atau pedoman bagaimana dan apa yang harus dikerjakan selama pengembangan sistem.

2.5.1. Sistem Development Life Cycle (SDLC)

Metode SDLC 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 yang paling tepat untuk dapat digunakan dalam menangani masalah system informasi suatu organisasi/perusahaan. Pada Gambar dibawah menunjukkan aktivitas-aktivitas/tahapan-tahapan yang dilakukan pada metode pengembangan *Waterfall*.

Adapun Proses kegiatan dapat dilihat dari gambar dibawah ini :



Gambar 2.2 *Waterfall Model*

Sumber : Pressman Roger.S, 1997 : 37

Prosedure dari metode waterfall adalah sebagai berikut :

1. *Analisys*, tahapan dimana menganalisis hal-hal yang diperlukan dalam pelaksanaan proyek pembuatan atau pengembangan perangkat lunak dan bertujuan untuk memahami sistem yang ada, mengidentifikasi masalah dan mencari solusinya.

2. *Design*, tahap ini merupakan tahap penterjemahan dari keperluan atau data yang telah dianalisis kedalam bentuk yang mudah di mengerti oleh pemakai.

3. *Coding*, yaitu menterjemahkan data atau pemecahan masalah yang dirancang kedalam bahasa pemograman yang telah ditentukan.

4. *Testing*, setelah program selesai dibuat maka tahapan berikutnya adalah ujicoba terhadap program tersebut.

5. *Mainteneance*, yakni penerapan secara keseluruhan disertai pemeliharaan jika terjadi perubahan struktur, baik dari segi software maupun hardware.

2.6. 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 :

2.6.1 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.

2.6.2 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.

2.6.3. 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.

2.6.4 Normalisasi

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

1. Normalisasi Pertama

Aturan :

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

2. Normalisasi Kedua

Aturan :

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

3. Normalisasi Ketiga

Aturan :

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

2.6.5. 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 dari deskripsi tertentu dan isi entity seperti A dan B
2. Atribut : Entiti mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter entity. Misalnya atribut nama pekerja dari entity pekerja, setiap ERD bias terdapat lebih dari satu atribut.
3. Hubungan : *Relationship,* Sebagaimana halnya entity maka dalam hubunganpun 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.

2.7 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* yang lainnya.

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 secarabersama-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.

2.7.1. 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.

2.7.2. 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, misalnya aplikasi dimana orang tua boleh

mempunyai banyak anak tetapi seorang anak hanya dapat mempunyai satu orang tua.

2. Database Jaringan

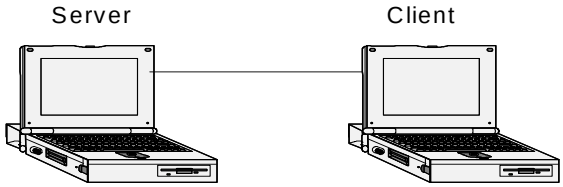
Sistem manajemen database jaringan dapat mempersiapkan struktur jaringan, misalnya dimana seorang mahasiswa boleh mempunyai banyak mata kuliah.

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. Hubunganhubungan tersebut baru perlu dispesifikasikan bila database digunakan untuk mendapatkan informasi atau untuk keperluan pembuatan laporan.

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 2.3

Sistem *Client - Server* Sederhana

2.9. Perangkat Lunak Pendukung

2.9.1 Sekilas Tentang Delphi7

Menurut Abdul Kadir dalam bukunya yang berjudul Dasar pemrograman delphi 5.0, pada awal tahun 1993 anggota *Borland International Incorporation* yang dipimpin oleh Andes Hejlberg merancang pemrograman yang berbasis *Windows* yang lebih mudah dan diberi nama *Visual Foo* yang akhirnya diganti menjadi *Delphi*.

Borland Delphi dibuat pada tahun 1995, kemudian pada bulan Maret 1996 *Borland International Corporation* meluncurkan *Delphi 2.0* yang dikhususkan untuk sistem operasi *windows* 95 dan *windows NT*, dan dipromosikanj menjadi salah satu bahasa pemrograman terbaik. Pada tahun 1997 serta tahun 1998 *Borland International Corporation* merilis *Delphi 3.0* dan *Delphi 4.0*. Kemudian pada tahun 1999 barulah *Borland International Corporation* meluncurkan *Borland Delphi 5.0*. dan hingga sekarang sudah ada *Borland Delphi7*

Borland Delphi menggunakan sistem yang disebut RAD (*Rapid Aplication Development*) yaitu perangkat pengembang yang mampu dengan mudah dan cepat menghasilkan program aplikasi. Delphi di sini adalah sebuah aplikasi yang

digunakan untuk pengembanagan dengan memanfaatkan keistimewaan konsep-konsep antar muka grafis dalam *microsoft windows*.

Seperti layaknya software *visual* lain Delphi juga memiliki IDE (*Integreted Development Environment*) atau lingkungan pengembangan tersendiri. Pada IDE terdapat beberapa fasilitas untuk melakukan desain, pengkodean maupun *debugging*. IDE Delphi terdiri dari beberapa bagian utama, antara lain:

1. *Main Windows* (Menu Utama)

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

2. *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*.

3. *Toolbar*

4. *Object TreeView*

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

5. *Code Explorer*

   Pada jendela *Code Explorer* ini akan ditampilkan semua tipe, variable, dan *routline* yang didefinisikan pada unit. Untuk tipe yang kompleks seperti

kelas, *Code Explorer* akan menampilkan semua informasi termasuk daftar field, properties dan method.

6. *Object Inspector*

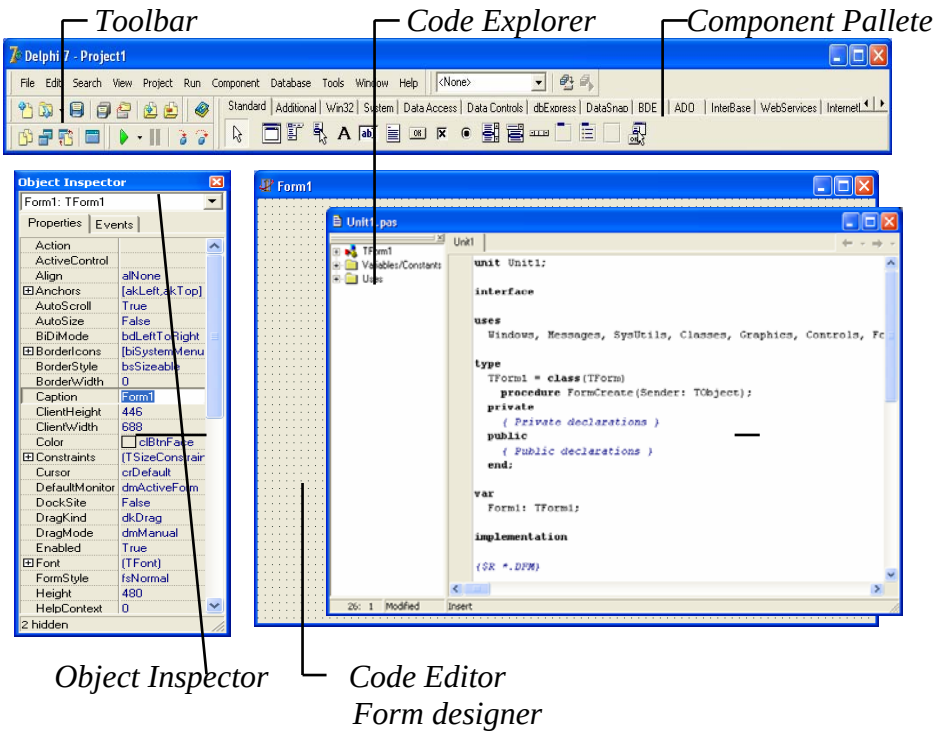
   Object Inspector terdiri dari dua bagian, yaitu *Properties* dan *Events*. Pada Properties bisa mengatur berbagai *property* dari *object* atau komponen yang akan digunakan.

7. *Form Designer*

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

8. *Code Editor*

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



Gambar 24
Bagian- bagian dari IDE1
Borland Delphi 7

2.9.2 Sekilas Tentang Program SQL Server 2000

SQL Server merupakan salah satu dari sejumlah bahasa pemrograman database (DBMS) yang bersaing merebut popularitas bersama-sama dengan dbase, Foxpro, FoxBase, QuickSilver dll. SQL Server kini mulai menjauhkan diri dan melangkah jauh lebih ke depan, terutama dengan munculnya versi SQL Server 2000. Menentukan bahasa mana yang terbaik untuk aplikasi database akan bersifat sangat subyektif. Namun, biasanya dukungan akan bahasa SQL *(structure query language)*, kriteria kecepatan, pemakaian memori, mudah tidaknya program, dan daya tampung data menjadi kriteria utama. Selain keutamaan SQL Server sebagai penampung database cukup besar dan dukungannya terhadap bahasa SQL, SQL server juga memberikan dukungan terhadap sistem database terdistribusi yaitu *client/server*.