BAB II

LANDASAN TEORI

2.1. Definisi Sistem

Terdapat beberapa definisi dari sistem, definisi sistem menurut kamus *webster's Unabridged* yang diungkapkan oleh Drs. Zulkifli Amsyah, MLS adalah :

"Elemen-elemen yang saling berhubungan membentuk satu kesatuan atau organisasi".

Sedangkan menurut L. ACKOF sistem didefinisikan sebagai:

"Setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya."

Dari kedua definisi sistem yang telah dikemukakan diatas, dapat diambil sebuah kesimpulan bahwa sistem adalah komponen atau elemen-elemen yang mempunyai hubungan dan dengan keadaan saling ketergantungan satu sama lain membentuk satu kesatuan atau organisasi.

2.1.1. Elemen Sistem

Pada umumnya suatu sistem terdiri dari lima elemen subsistem, yaitu:

1. Tujuan

   Tujuannya adalah hal yang ingin kita capai dimana dengan sistem informasi yang ada kita dapat memudahkan orang lain dalam mendapatkan informasi yang kita sajikan,sehingga terjadi interaksi antara kita dan masyarakat umum yang saling menguntungkan.

2. Masukan(input)

   Yang dimaksud masukan adalah Sekumpulan data yang akan kita olah menjadi sebuah informasi yang nantinya akan kita sajikan bagi masyarakat.

3. Keluaran(Output)

   Keluaran atau output adalah informasi-informasi yang dapat dengan mudah di peroleh, dimengerti dan dimanfaatkan oleh masyarakat.

4. Proses

   Proses adalah suatu kegiatan dimana kita mengolah seluruh data yang ada untuk menghasilkan suatu informasi .

5. Mekanisme Pengendalian

   Mekanisme Pengendalian adalah pengawasan jalannya seluruh kegiatan operasional sistem mulai dari input, proses sampai keluarannya untuk menghindari terjadinya kesalahan-kesalahan.

6. Umpan Balik

   Umpan balik adalah Masukan atau Pendapat yang diberikan oleh masyarakat mengenai sistem informasi yang kita buat sehingga kita dapat memperbaiki dan meningkatkan kinerja dari sistem informasi yang kita buat tersebut.

2.1.1. Karakteristik Sistem

Menurut Jogiyanto HM, suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu :

a. *Komponen Sistem (Components)*

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. *Batas Sistem (Boundary)*

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

c. *Lingkungan Luar Sistem (Environments)*

   Lingkungan luar suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan sistem yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar sistem yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

d. *Penghubung Sistem (Interface)*

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan subsistem berinteraksi dengan sub sistem yang lainnya membentuk satu kesatuan. Keluaran *(output)* dari satu subsistem memungkinkan menjadi masukan *(input)* untuk subsistem yang lainnya melalui media penghubung.

e. *Masukan Sistem (Input)*

Masukan adalah energi yang dimasukan ke dalam sistem. Masukan dapat berupa masukan perawatan *(maintenance input)* dan masukan sinyal *(signal input)*. M*aintenance input* adalah energi yang dimasukan supaya sistem tersebut dapat beroperasi. S*ignal input* adalah energi yang diproses untuk mendapatkan keluaran. Sebagai contoh didalam sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputer dan data adalah *signal input* untuk diolah menjadi informasi.

f. *Keluaran Sistem (Output)*

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.. Keluaran dapat merupakan masukan untuk subsistem yang lain.

g. *Pengolah Sistem (Process)*

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

h. *Sasaran Sistem (Objectives) atau tujuan (Goal)*

Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.3. Klasifikasi Sistem

Menurut Jogiyanto HM, sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut:

a. *Sistem Abstrak (abstrack system) dan Sistem Fisik (Physical system)*

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

Sistem fisik merupakan sistem yang ada secara fisik.

b. *Sistem Alamiah (Natural system) dan Sistem Buatan Manusia (Human made system)*

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia.

Sistem buatan manusia adalah sistem yang dirancang oleh manusia.

c. *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 keluaran dari sistem dapat diramalkan.

Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d. *Sistem Tertutup (Closed system) dan Sistem Terbuka (Open system)*

Sistem tertutup merupakan sistem yang tidak berhubungan dengan lingkungan lainnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup.

Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya.

2.2. Definisi informasi

Informasi adalah kumpulan data yang telah diolah sehingga mempunyai arti untuk bisa diterjemahkan dan diterima oleh penerima informasi tersebut. Informasi dibagi menjadi dua yaitu, informasi statis dan informasi dinamis.

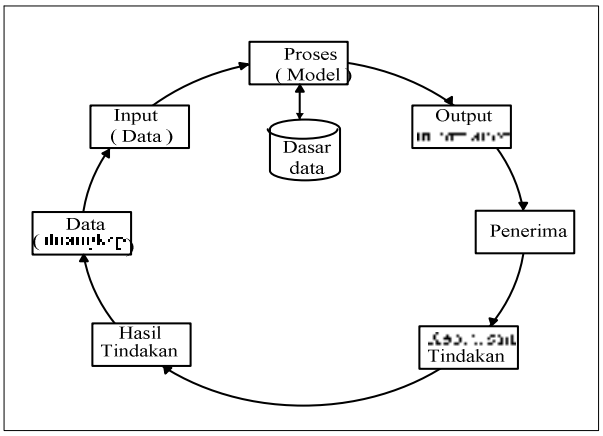
Informasi statis adalah informasi yang bersifat statik yang artinya isi dari informasi tersebut tidak dapat berubah dan tetap sesuai dengan data dari dalam informasi tersebut. Sedangkan informasi dinamis adalah kebalikan dari informasi statis yaitu data-datanya dapat berubah.

Informasi dapat diperoleh melalui:

- Pustaka (ilmiah, semi-ilmiah, popular)
- Media massa (cetak, radio, televise)
- Lisan (wawancara, telepon)
- Tulisan (surat, fax, katalog)

2.3. Definisi sistem informasi

Istilah sistem informasi merupakan suatu pengumpulan data yang terorganisasi beserta tatacara penggunaanya yang mencakup lebih jauh daripada sekedar penyajian. Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tatacara penggunaannya. Keberhasilan suatu sistem informasi yang dikukur berdasarkan maksud pembuatannya tentu bergantung pada tiga faktor utama, yaitu keserasian dan mutu data, pengorganisasian data, dan tatacara penggunaanya (Cook, 1977).



Gambar 2.1 Siklus Informasi

[Sumber : Analisis dan Desain Sistem Informasi, Jogianto HM, 1999].

Pada gambar 2.1 dapat dijelaskan bahwa data informasi yang bermanfaat bagi penerima sehingga menghasilkan informasi yang jelas, akurat dan tepat waktu. Pertama-tama data dimasukkan kedalam model *(proses)* yang umumnya memiliki urutan proses tertentu, setelah diproses maka akan menghasilkan informasi yang bermanfaat bagi penerima sebagai dasar dalam mengambil suatu keputusan atau tindakan tertentu. Dari hasil keputusan atau tindakan tersebut maka akan

menghasilkan kejadian yang dapat digunakan kembali sebagai data yang nantinya akan dimasukkan ke dalam model (*proses*). Data informasi tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model *(proses)* dan seterusnya hingga membentuk suatu siklus informasi *(information cycle)*.

Menurut Budi Sutedjo, kualitas dari suatu informasi tergantung dari beberapa hal, yaitu :

a. Akurat

   Informasi harus bebas dari kesalahan-kesalahan yang biasa terjadi dan harus jelas maksud dan tujuan dari informasi tersebut, sehingga *output* (keluaran) bisa dipertanggungjawabkan.

b. Tepat waktu

   Informasi yang datang pada pemakai tidak boleh terlambat karena informasi yang terlambat tidak akan mempunyai nilai di dalam pengambilan suatu keputusan.

c. Relevan

   Informasi yang didapat mempunyai manfaat bagi pemakai dan informasi yang diterima pemakai dapat berbeda-beda.

d. Kesempurnaan informasi

   Kesempurnaan informasi menjadi faktor penting karena kesempurnaan berperan sebagai pendukung faktor pertama diatas, dimana informasi disajikan lengkap tanpa pengurangan, penambahan atau pengubahan.

e. Mudah dan murah

Kesulitan cara mendapatkan dan besar atau kecil biaya untuk memperoleh informasi juga menjadi bahan pertimbangan. Jika cara dan biaya untuk memperoleh informasi sulit dan mahal, maka orang akan menjadi tidak berminat untuk memperoleh informasi, atau mencari alternatif substitusi informasi tersebut.

Nilai suatu informasi ditentukan oleh dua hal, yaitu manfaat dan biaya mendapatkan informasi. Suatu informasi dikatakan bernilai jika manfaat lebih efektif dibandingkan dengan biaya mendapatkanya. Keuntungan dari sebuah informasi sebagian besar informasi tidak dapat diperkirakan dengan satuan nilai uang, tetapi dapat diperkirakan dengan nilai efektifitas informasi tersebut.

2.4 Metode Analisis Dan Perancangan Terstruktur

Didalam tahap perancangan suatu sistem diperlukan teknik-teknik penyusunan sistem yang difungsikan untuk menganalisa dan mendokumentasikan data yang mengalir di dalam sistem tersebut.

2.4.1. Flow map

Dalam menganalisis sebuah sistem flow map diperlukan sebagai salah satu cara bagaimana alur data yang berjalan dari sistem tersebut. Dari flow map tersebut dapat mengetahui divisi-divisi perusahaan yang terlibat dalam system tersebut, dokumen-dokumen apa saja yang mengalir serta dapat mengetahui kelemahan-kelemahan maupun keunggulan dari sistem yang diterapkan oleh perusahaan.

Flow Map merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan. Flow Map digunakan untuk menganalisis bagaimana hubungan antara sub kerja yang akan menggerakkan sistem. Setelah diketahui bagian-bagian yang terlibat dalam sistem, maka akan diketahui berapa jumlah entitas yang terkait dengan sistem yang dianalisis dan dirancang.

Secara garis besar suatu flowmap digambarkan sebagai pemetaaan hubungan antara bagian-bagian kerja melalui dokumen, baik berupa laporan maupun formulir yang dinotasikan melalui simbol-simbol.

Beberapa simbol yang digunakan didalam flowmap antara lain :

A. Dokumen

Simbol ini menunjukan dokumen input dan output baik untuk proses manual, mekanik atau komputer

B. Kegiatan manual

Simbol ini menunjukan pekerjaan manual yang dilakukan oleh orang/user.

C. Simpanan *offline*

Simbol ini menunjukan file non – komputer yang diarsipkan.

D. Proses

Simbol ini menunjukan kegiatan proses dari operasi program komputer.

E. Simpanan data

Simbol ini menunjukan tempat penyimpanan data

F. Penghubung

Simbol ini menunjukan penghubung kehalaman yang sama atau kehalaman lain

2.4.2. Diagram kontek

Merupakan model grafis yang memperlihatkan sistem dalam bentuk paling umum/global dan digunakan untuk mendefinisikan serta memperlihatkan lingkup atau batas sistem yang akan ditelaah.

Diagram Konteks selalu mengandung satu proses saja. Proses ini mewakili proses dari seluruh sistem. Diagram Konteks menggambarkan hubungan input / output antara sistem dengan kesatuan luar (*eksternal entity*).

2.4.3. Data flow diagram

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur. DFD dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir.

DFD disebut logical DFD karena dapat menguraikan dan menganalisis data yang mengalir melalui sistem, baik secara manual maupun otomatis, termasuk proses data, penyimpanan data. Selain itu penggambaran tranformasi dari data masukan menjadi data keluaran melalui proses sedemikian rupa sehingga fapat ditampilkan logikanya secara mandiri tanpa memperhatikan komponen fisik. Salah satu notasi DFD yang sering digunakan adalah notasi yang dikenalkan oleh Edward Yourdon.

2.5 *Software* Sistem

Software sistem merupakan *background* program yang memungkinkan software aplikasi dapat berfungsi pada peralatan *hardware* sistem komputer [Charles S.P.,'96].

Software sistem dapat dibedakan menjadi 3 bagian, yaitu [Charles S.P.,'96] **:**

a. *Operating System*, adalah kumpulan utama dari program yang mengatur aktivitas sistem komputer. Contoh : *Windows '98*, *Macintosh System Software*, MS-DOS, UNIX, dan *Penpoint*.

   Windows '98

   Windows '98 dilengkapi fasilitas-fasilitas yang dapat mempermudah kerja user, yaitu **:**

   - *Built-in Networking*, fasilitas jaringan terintegrasi di sistem operasi.
   - Pengamanan level pemakai *pass-through.*
   - *Registry,* semua informasi mengenai sistem disatukan sehingga mempermudah pengelolaan sistem.

   Pada *Windows '98* antarmuka dengan sistem dilakukan sebagai berikut **:**

   - Kebanyakan dilakukan melalui fasilitas berbasis grafik : *menu, icon.*
   - Tersedia *Task bar* yang mempermudah perpindahan aplikasi yang saat itu dijalankan sistem operasi.
   - Terdapat jendela untuk mengetikkan baris perintah secara langsung.

Keluarga *Windows* menyediakan kumpulan *system Call (API)* yang dipanggil program aplikasi seperti memanggil program atau subrutin. Keluarga *Windows* dikembangkan dengan C, *system call* berbentuk fungsi bahasa C. Meskipun demikian kita dapat memanggil dalam bahasa apapun, kompilator atau interpreter menerjemahkan menjadi panggilan ke sistem operasi *Windows.*

Kumpulan *system call* di keluarga *Windows* disebut WIN32 API. Bentuk fungsi, struktur, pesan, makro dan antarmuka konsisten dan seragam untuk semua keluarga *Microsoft* berbasis 32 bit.

Penciptaan dan pengelolaan antarmuka pemakai, juga fitur-fitur *Windows*, yaitu *Carets, Clipboard, Cursors, Hooks, Icons,* dan *Menus.*

Kendali umum pada *Windows* terdapat di *commond control library*, COMCTL32.DLL. *Common control* DLL berisi antarmuka pemrograman sehingga aplikasi dapat menciptakan dan memanipulasi *controls* juga menerima masukan *controls.*

b. *Language Translator*, adalah suatu program sistem yang berfungsi untuk mengkonversikan program aplikasi ke dalam bahasa mesin. Contoh : *Language Translator* pada COBOL dan BASIC.
c. *Utility Programs*, adalah suatu program yang menjelaskan atau memperluas kegunaan dari sistem operasi suatu sistem komputer. Contoh : *Backup utilities, Spooling utilities*, dan *Data compression utilities*.

2.6 *Software* Aplikasi

Software aplikasi merupakan suatu perangkat yang memungkinkan pemakai *(user)* memahami sistem komputer [Charles S.P.,'96]. Berikut beberapa contoh dari software aplikasi **:**

1. *Visual Basic* 6.0

Pemrograman *Visual Basic* adalah salah satu paket pemrograman visual yang handal dalam membangun aplikasi-aplikasi berbasis *windows*. Dimana dalam *Visual Basic* pembuatan aplikasi dimulai dengan memperkirakan kebutuhan, merancang tampilan program terlebih dahulu dan selanjutnya diikuti dengan pembuatan kode untuk program tersebut. Komponen paket program *Visual Basic* adalah **:**

a. *Window utama* (main windows)

   Pada *window* utama ini semua kegiatan pembuatan program dilakukan. Menu-menu untuk perancangan program terdapat pada *window* ini, yang digunakan selama *design time.*

b. *Window toolbox*

   Window ini berisi beberapa peralatan/komponen untuk perancangan aplikasi selama *design time*. Kontrol-kontrol yang terdapat pada *window* ini merupakan gabungan dari kontrol standar yang digunakan oleh aplikasi *windows* dan kontrol-kontrol tambahan yang disediakan untuk menyelesaikan tugas-tugas pemrograman yang juga disebut dengan *ActiveX Control*. Kontrol-kontrol yang terdapat pada *window* ini dapat ditambah maupun dikurangi

sesuai dengan kebutuhan pemakai. Kontrol tambahan ini biasanya disediakan perusahaan partai ketiga dimana kontrol-kontrol tersebut dibuat untuk meningkatkan unjuk kerja program misalnya dalam membuat sebuah aplikasi *Database* yang cukup rumit.

c. *Window* properti

Window ini digunakan untuk mengatur sifat objek yang terdiri dari *Form* atau kontrol yang digunakan selama *design time*. Isi *window* properti selalu berubah sesuai dengan *form* atau kontrol yang sedang aktif saat itu.

Window properti terdiri dari beberapa bagian **:**

1. Bagian untuk memilih objek.
2. Bagian untuk mengatur sifat objek.
3. Bagian untuk memilih properti sebuah objek.

Setiap objek yang dibuat, secara otomatis akan diberi nama sebagai sebuah objek oleh *Visual Basic* dan akan memiliki properti *default*.

d. *Window project*

Window ini berisi informasi tentang sebuah *project* yang sedang dibangun, sangat berguna untuk manajemen *project*. Dalam membangun sebuah *project* yang besar biasanya melibatkan banyak modul yang berisikan bagian-bagian program yang telah dibagi-bagi agar mempermudah manajemen program dan pengecekan kesalahan. Maka daftar modul yang terlibat dalam pembangunan *project* akan ditampilkan dalam *window* ini.

e. *Window form*

Window ini digunakan untuk membuat program dan digunakan untuk menggambarkan kontrol-kontrol yang diambil dari *window toolbox*.

Kode-kode program yang akan diletakkan pada objek yang akan menggunakan kode tersebut pada setiap kejadian yang kita inginkan. Kode tersebut akan dijalankan setiap terjadi sesuatu atas objek tersebut selama *Running Time*. Setiap kejadian yang terjadi atas objek tersebut selanjutnya kita sebut *Event Hendler*.

Setiap aplikasi *Windows* selalu digerakkan oleh pesan (*message*). Pesan ini dikirimkan oleh *window* ke aplikasi dan aplikasi memberikan respon karena pesan yang diterimanya. Cara ini merupakan teknik yang dilakukan oleh *windows* untuk implementasi aplikasi-aplikasi yang berada dalam lingkungannya, terutama untuk manajemen sistem supaya beberapa program dapat dijalankan pada saat yang bersamaan (*multitasking*). Pesan-pesan tersebut ada jika terdapat kejadian-kejadian (*event*) apakah itu berasal dari aplikasi itu sendiri atau dari sistem. Ada kejadian yang khusus terdapat pada satu objek tetapi ada juga yang umum terdapat pada setiap objek.

Pada *Visual Basic* sebuah *project* terdiri atas beberapa *file*. Jika program semakin kompleks maka program akan terdiri dari beberapa *form. Form-form* akan diletakkan pada *file-file* terpisah. Untuk aplikasi yang paling sederhana *Visual Basic* akan membuat dua *file* yaitu *file Project* (*.VBP) untuk menyimpan

informasi proyek yang dibuat dan sebuah *file form* (*.FRM) untuk menyimpan informasi *form* dan kode-kode yang terdapat pada *form* tersebut.

Visual basic selain digunakan untuk merancang juga merupakan sebuah *compiler*. Dengan *compiler* ini, program dapat dibuat menjadi sebuah *file* yang dapat berdiri sendiri tanpa membutuhkan program *Visual Basic* untuk menjalankannya.

Program–program yang dibuat dengan menggunakan *Visual Basic* merupakan sebuah aplikasi yang kompleks. Walaupun program yang dirancang berukuran kecil, tetapi untuk menjalankan program ini diperlukan *file-file* pembantu untuk menjalankannya. Beberapa *file* ini harus disertakan dalam pendistribusian aplikasi yang telah jadi. *Visual Basic* telah menyediakan suatu *tools* untuk mempermudah proses instalasi ini. *Tools* tersebut adalah *Application Setup Wizard.* Cara menggunakan program ini sangat sederhana, karena program ini dengan pintar mengetahui tindakan apa saja yang akan dilakukan. Tetapi dalam kondisi-kondisi tertentu Anda akan diminta untuk membantu proses instalasi ini.

Objek User interface yang digunakan oleh *Visual Basic* sama dengan yang dimiliki oleh *windows*. Bagi pemrogram yang membuat aplikasi *window*, *objek user interface* disebut sebagai kontrol. Kontrol-kontrol ini digunakan sebagai media input bagi pemakai yang tujuannya untuk media berinteraksi antara pemakai dengan program yang dijalankan. Objek ini dapat berupa suatu *tools* yang digunakan untuk masukan (*input)* ataupun keluaran (*output*), atau hanya sebagai pesan *(message)* yang ditampilkan ke layar monitor.

Setiap kontrol memiliki properti seperti warna, tulisan, posisi dan lain-lain. Dan terkadang memiliki properti yang sama walaupun kontrolnya berbeda namun adakalanya memiliki properti khusus. Pengaturan properti kontrol dapat dilakukan dari *Window* kontrol, namun dapat juga dilakukan dengan menuliskan properti tersebut di dalam *source* kodenya.

Form merupakan dasar dari setiap aplikasi yang dikembangkan dengan menggunakan *Visual Basic*. Kontrol-kontrol, tulisan atau hasil program diletakkan pada *form* ini. Dari *form* dapat dirancang beberapa *file windows*, antara lain **:**

Window Utama, *Window* Anak(*Child*), *Window* Dialog, *Window Multiple Document Interface (MDI).*

Form juga seperti sebuah kontrol yang memiliki properti yang dapat digunakan dan diubah sesuai keperluan.

Visual Basic adalah bahasa pemrograman yang bekerja dalam lingkungan *Windows*. *Visual Basic* juga merupakan sarana (*tool*) untuk menghasilkan program-program aplikasi berbasis *Windows* (*Object Oriented Programming*).

Visual Basic merupakan pengembangan terakhir dari BASIC (*Beginner's All-Purpose Symbolic Instruction Code*). *Visual Basic* masih tetap mempertahankan beberapa sintaks atau format penulisan program yang dipakai oleh BASIC.

Ada tiga jenis *Visual Basic* versi 6.0. yaitu **:**

a. *Standard Edition /Learning Edition*

Merupakan versi standar yang sudah mencakup berbagai sarana dasar dari *Visual Basic* 6.0 untuk mengembangkan aplikasi berbasis *Windows,* dan merupakan versi yang digunakan pada pembuatan *software* aplikasi ini.

b. *Professional Edition*

Versi ini memberikan berbagai sarana tambahan yang dibutuhkan oleh para *programmer* profesional.

c. *Enterprise Edition*

Versi ini dikhususkan untuk para *programmer* yang ingin mengembangkan aplikasi *remote computing* atau *client/server.* Biasanya versi ini digunakan untuk membuat aplikasi pada jaringan.

2. *Microsoft Access* 2000

Pada *Microsoft Access* 2000 terdapat tempat utama yang dipergunakan untuk bekerja yaitu *Database Window*. *Database Window* terdiri dari beberapa bagian, meliputi **:**

a. *Tables*, adalah kumpulan data, merupakan komponen utama dari sebuah *database.*
b. *Queries*, digunakan untuk mengatur data mana saja yang terdapat pada suatu tabel yang perlu ditampilkan.
c. *Forms*, digunakan untuk mengatur tampilan data di layar monitor.
d. *Reports*, digunakan untuk mengatur tampilan data yang akan dicetak dengan *printer.*

e. *Macros*, merupakan fasilitas untuk mengotomatisasi sekaligus menghemat waktu yang diperlukan dalam pembuatan aplikasi *database.*

f. *Modules*, berguna dalam pembuatan aplikasi *database* tingkat lanjut.

Pada *access* telah tersedia berbagai cara yang dapat digunakan untuk pembuatan tabel. Salah satu pilihannya adalah dengan menggunakan *Design View*. Modus *Design View* akan menampilkan nama *field* (*Field Name*), jenis datanya (*Data Type*), serta sebuah kolom *Description* (keterangan) yang merupakan kolom opsional (boleh tidak diisi).

Access menyediakan berbagai pilihan jenis data, meliputi *Text, Memo, Number, date/Time, Currency, AutoNumber, yes/No, OLE Object, Hyperlinks*, dan *Lookup Wizard.*

Bagian bawah dari *Design View* menampilkan properti bagi setiap *field*. Komponen-komponen *properties* hampir sama untuk berbagai jenis data, kecuali **:**

a. Ukuran *field (FieldSize)* hanya berlaku untuk jenis data *Text* dan *Number,* sementara maknanya pun berbeda pada keduanya.

b. Letak desimal *(Decimal Places)* hanya berlaku untuk *Number* dan *Currency*.

c. Properti *Format* bekerja secara berbeda untuk jenis data yang berlainan.